

Lucia Darsa

Deformação e Metamorfose de Objetos Gráficos

Dissertação apresentada ao Departamento de Informática da PUC-Rio como parte dos requisitos para obtenção do título de Mestre em Informática: Ciência da Computação.

Orientador: Prof. Jonas de M. Gomes
(Visgraf/IMPA)

Co-orientador: Prof. Bruno Feijó
(DI/PUC-Rio)

Departamento de Informática

Pontifícia Universidade Católica do Rio de Janeiro

Rio de Janeiro, 14 de julho de 1994.

A meus pais e ao Júlio.

Agradecimentos

Agradeço ao prof. Jonas de Miranda Gomes, meu orientador, por ter me ajudado a utilizar a Matemática como uma ferramenta para uma compreensão melhor da Computação Gráfica e do Processamento de Imagens. A Luiz Henrique de Figueiredo pelo exemplo de estilo sucinto e objetivo simplesmente inigualável. Ao projeto Visgraf e seus integrantes, pelos uso dos equipamentos e ambiente propício à pesquisa. Ao Departamento de Informática da Puc-Rio, e aos projetos ICAD/TecGraf, em especial, aos professores Bruno Feijó e Marcelo Gattass pela paciência e determinação que possibilitaram a realização deste trabalho em colaboração com o IMPA. E, finalmente, a Bruno Costa por sua presença constante que permitiu a conclusão deste trabalho.

Resumo

A metamorfose é simplesmente uma mudança completa de forma, estrutura ou substância de algo. Esta definição de dicionário corresponde às três principais formas de informação que devem ser representadas em modelos de objetos: geometria, topologia e propriedades. Esta dissertação visa descrever e analisar o problema de metamorfose em \mathbb{R}^n , através do uso de objetos suficientemente gerais para abranger, em essência, todas as formas clássicas de representação em \mathbb{R}^2 e \mathbb{R}^3 . Técnicas existentes são analisadas como sub-casos da generalização, integrando o tratamento dado aos previamente dissociados problemas de metamorfose 2D e 3D, e permitindo o compartilhamento de algumas das técnicas. Uma nova abordagem para metamorfose 3D, baseada em uma técnica 2D, e uma implementação são descritas.

Abstract

Metamorphosis is defined as a complete change of shape, structure or substance of something. This common sense dictionary definition corresponds to the three main forms of information that must be represented in object models: geometry, topology and properties. This dissertation attempts to describe and analyze the metamorphosis problem in \mathbb{R}^n , through the use of objects general enough to encompass, in essence, all classical forms of representation in \mathbb{R}^2 and \mathbb{R}^3 . Existing techniques are analyzed as subcases of the generalization, integrating the treatment given to the previously dissociated 2D and 3D metamorphosis problems, and allowing the sharing of some of the techniques. A new approach for 3D metamorphosis, based on a 2D technique, and an implementation are described.

(An English version is available from the author: lucia@visgraf.impa.br)

Sumário

1 Introdução	1
1.1 Aplicações de Metamorfose	1
1.2 Uma Visão Integrada	4
2 Morphing em R^n	5
2.1 Conceitos Fundamentais	5
2.2 Transformações de Domínio	7
Transformações de Domínio Parametrizadas	8
2.3 Transformações de Contra-domínio	12
Transformações de Combinação	14
Transformações de Contra-domínio Parametrizadas	16
2.4 Morphing	17
3 Morphing em R^2	18
3.1 Modelos 2D e suas Representações	18
Imagens Digitais	19
Desenhos	20
Representações 2D por Bordo e Volume	21
3.2 Transformações de Domínio e Contra-domínio	23
Especificação e Implementação	24
3.3 Técnicas de Combinação de Formas 2D	25
Blending Linear Simples	26
Blending Físico	26
Combinação de Estruturas Combinatórias 2D	28
3.4 Técnicas de Warping de Imagens	28
Warping por Malhas de Triângulos	29
Warping por Malhas de Splines em Dois Passos	30
Warping Baseado em Campo	33
4 Morphing em R^3	36
4.1 Modelos 3D e suas Representações	36
4.2 Transformações 3D de Domínio e Contra-domínio	38
4.3 Técnicas de Warping 3D	39
Deformação por Formas Livres	39
Warping por Malha de Triângulos	41
Warping Baseado em Campo	42
Warping por Malhas de Splines em Dois Passos	43
4.4 Técnicas de Blending 3D	45
Blending de Objetos Volumétricos Implícitos	45

Blending de Volumes por Fourier Escalonado	46
Combinação de Estruturas Combinatórias	47
Blending Físico	48
5 Implementação	50
5.1 Decisões de Implementação	50
Representação	50
Técnicas de Warping e Blending	51
5.2 Visualização	52
5.3 Tópicos de Interface com o Usuário	53
6 Conclusão e Trabalho Futuro	54
6.1 Trabalho Futuro	55
7 Referências Bibliográficas	56

Lista de Figuras

Figura 1-1: Visualização de transformações 3D: (a) técnicas 2D; (b) técnicas 3D.	3
Figura 2-1: Funções diferentes com a mesma superfície de nível 0.	6
Figura 2-2: Um objeto g criado por uma transformação de domínio de f .	7
Figura 2-3: Transformação de domínio: expansão e contração.	8
Figura 2-4: Objeto de saída parametrizado: uma translação de um objeto de entrada parametrizado.	10
Figura 2-5: Objeto de saída parametrizado: uma translação variável de um objeto de entrada.	11
Figura 2-6: Um objeto de saída parametrizado: objeto de entrada e transformação são variáveis.	12
Figura 2-7: Diagrama de transformação de contra-domínio.	13
Figura 2-8: Transformação de contra-domínio de um objeto.	13
Figura 2-9: Transformação de contra-domínio de dois objetos.	14
Figura 2-10: Combinação por média de duas funções..	15
Figura 2-11: Combinações diferentes entre duas parábolas.	16
Figura 2-12: Diagrama de morphing.	17
Figura 3-1: Um objeto 2D simples.	19
Figura 3-2: Uma imagem digital.	20
Figura 3-3: Um desenho bidimensional.	21
Figura 3-4: Especificação de morphing: (a) partição; (b) características; (c) semi-automática.	24
Figura 3-5: Duas possíveis associações de vértices de referência e os blendings respectivos.	26
Figura 3-6: (a) Blending físico; (b) Software comercial [Sederberg, Greenwood 92].	27
Figura 3-7: Deformação por malha de triângulos: (a) original; (b) deformada.	30
Figura 3-8 Malhas de splines.	31
Figura 3-9: Warping de uma scanline: (a) interseção; (b) função de mapeamento.	32
Figura 3-10: Sistema de coordenadas definido por uma característica.	33
Figura 3-11: Combinação de múltiplos sistemas de coordenadas: (a) original; (b) deformado.	34
Figura 4-1: Sistema de coordenadas deformado por formas livres: (a) relaxado; (b) deformado.	40
Figura 4-2: Morphing por formas livres: (a) direto; (b) com um passo intermediário.	41
Figura 4-3: Sistema de coordenadas definidos por uma característica 3D.	42
Figura 4-4: Pipeline de warping em três passos.	44
Figura 4-6: Uma combinação de duas estruturas combinatórias: (a) projeção; (b) combinação; (c) interpolação.	48
Figura 4-7: Comparação de blendings de formas 2D: (a) físico; (b) combinação de estruturas combinatórias.	49

1 Introdução

Metamorfose é simplesmente uma mudança completa de forma, estrutura ou substância de alguma coisa [Webster's 90]. É uma forma de transformação presente em nossas vidas, geralmente envolvendo uma alteração de um ser vivo em outro. Como todos os processos da natureza, a metamorfose é um candidato natural para a ciência tentar modelar. Várias aplicações para metamorfose, vista como uma modificação completa de qualquer objeto em outro, têm sido encontradas em áreas bastante diversas. Muitas tentativas têm sido feitas para gerar e visualizar tais transformações. O uso de métodos gráficos baseados em computador com este fim é relativamente recente, e algumas abordagens potenciais foram investigadas.

A Modelagem Geométrica é uma área da Computação Gráfica que objetiva a representação e a manipulação de entidades inerentemente geométricas através do uso de modelos apropriados. Um *modelo* é uma representação de um subconjunto das características de uma dada entidade, que ajuda ou torna possível uma melhor compreensão, análise ou até mesmo previsão de comportamento ou estrutura da entidade. A Computação Gráfica, na verdade, requer modelos mais que simplesmente geométricos, i.e., o subconjunto das características da entidade representados no modelo varia de acordo com a aplicação alvo, e normalmente inclui características não geométricas. Uma *ferramenta de modelagem* é simplesmente um método de se obter uma instância (*objeto*) de um certo modelo geométrico ou gráfico [Foley et alli 90].

1.1 Aplicações de Metamorfose

As aplicações de metamorfose têm usualmente se enquadrado em duas categorias principais: aquelas usadas para gerar uma seqüência de transformação, e aquelas que criam um objeto intermediário como um fim. Obviamente, um objeto intermediário é um passo de uma seqüência de transformação, mas algumas aplicações requerem a criação de objetos que possam ser posteriormente manipulados, independentemente de terem sua origem em uma metamorfose. Em outras palavras, a metamorfose pode ser usada

para visualizar transformações, ou como uma ferramenta de modelagem, no sentido que pode ser usada para criar novos modelos a partir de modelos primitivos.

Classicamente, a metamorfose foi usada principalmente para visualizar transformações, que usualmente ocorrem originalmente em um espaço tridimensional. O estudo de transformações é especialmente relevante nos espaços 2D e 3D por razões óbvias, mas existe uma importante distinção entre situações que são intrinsecamente bidimensionais ou tridimensionais. A metamorfose entre duas letras do alfabeto, por exemplo, é naturalmente um problema 2D, enquanto a transformação entre dois seres é um problema 3D. Apesar desta clara distinção, uma certa confusão advém do fato que estamos acostumados a trabalhar em projeções 2D de um mundo 3D, tal como acontece em um desenho animado ou em um filme.

Tanto técnicas bidimensionais quanto tridimensionais podem ser usadas para visualizar a transformação de objetos 2D ou 3D. Além dos usos canônicos, técnicas 3D podem ser facilmente utilizadas para visualizar o caso particular de uma transformação 2D, e técnicas 2D podem ser usadas para visualizar uma transformação 3D, mas apenas sob certas condições: ao invés de criar uma metamorfose entre dois objetos 3D e projetar o objeto resultante em uma tela 2D para visualização, é possível projetar os objetos 3D em telas 2D e criar a metamorfose entre as duas projeções, em lugar dos objetos reais (veja Figura 1-1). Esta aproximação pode alcançar resultados aceitáveis se, entre outras restrições, os objetos forem observados de um ponto de vista similar com condições de iluminação muito semelhantes.

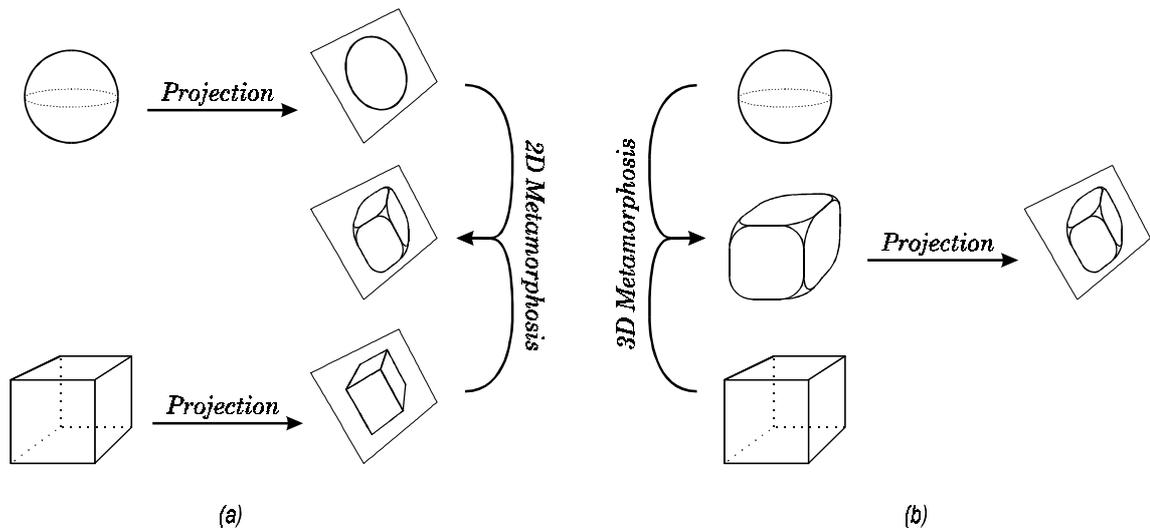


Figura 1-1: Visualização de transformações 3D: (a) técnicas 2D; (b) técnicas 3D.

Como imagens digitalizadas são uma forma destas projeções 2D do mundo 3D, as técnicas de metamorfose de imagens largamente utilizadas ocorrem em uma projeção de um espaço 3D, e portanto, não são capazes de modelar alguns efeitos de iluminação e visibilidade corretamente. O contraste entre a criação de uma metamorfose a partir de imagens ou de modelos 3D é análogo ao existente entre a digitalização de uma imagem e a sua síntese: no primeiro, uma vez que a imagem é digitalizada, não é possível alterar suas características de iluminação e visibilidade, enquanto no segundo, ainda não é possível sintetizar objetos que se assemelhem a seres vivos.

Uma redução alternativa do problema 3D para duas dimensões é também possível através do fatiamento dos objetos originais em seções retas 2D. Uma metamorfose entre cada par de seções retas correspondentes é criada, e a superfície 3D é reconstruída dessas fatias metamorfoseadas [Chen, Parent 89]. Esta abordagem, diferentemente da anterior, não é baseada na visualização da metamorfose resultante, mas ao invés, na criação de um novo modelo 3D que pode ser manipulado posteriormente.

Historicamente, o uso de técnicas e modelos de dados tridimensionais extremamente simples para a representação e a síntese de metamorfoses no espaço 3D parece ser pioneiro. Contudo, devido aos seus resultados inicialmente limitados, foi temporariamente abandonado em favor de abordagens bidimensionais, mais viáveis e de melhores resultados. De fato, a metamorfose de imagens, particularmente, tem sido

extensivamente utilizada na indústria de entretenimento para criar transformações entre objetos originalmente 3D, com resultados práticos excelentes.

Talvez inspirados pelo sucesso da metamorfose de imagens, novos caminhos para resolver o problema de metamorfose de objetos 3D diretamente no espaço tridimensional foram tentados. Estas novas técnicas mostraram um avanço considerável sobre o trabalho anterior, que restringia severamente a natureza dos objetos envolvidos na transformação. Entretanto, ainda são impostas restrições consideráveis aos objetos a serem transformados, especialmente em termos de topologia, além de faltar um controle mais detalhado sobre a transformação. Será mostrada que até mesmo as mais recentes técnicas resolvem, de fato, apenas uma parte do problema de metamorfose.

1.2 Uma Visão Integrada

A abordagem puramente pragmática dada à metamorfose levou a uma classificação de tipos de objeto relativamente artificial, com tratamentos inteiramente diferentes dados a transformações de cada tipo: imagens, desenhos estruturados 2D e sólidos 3D. Técnicas presentes na literatura foram desenvolvidas para objetos de uma certa dimensão, representados de uma certa forma, e eram freqüentemente deficientes conceitualmente.

Este trabalho procura demonstrar que todos esses problemas compartilham uma conceituação comum, que será generalizada para objetos abstratos n -dimensionais, independentemente de sua representação. Esta visão integrada irá mostrar que, a despeito de poucas idiosincrasias inerentes, algumas das técnicas e soluções existentes para alguns dos problemas podem ser estendidas para os outros, revelando o que poderia ser considerado como novas técnicas. Em particular, uma nova técnica que resolve o verdadeiro problema de metamorfose entre objetos de topologias distintas, de uma forma controlada, será apresentado com alguns resultados obtidos para uma certa representação de objeto. Mais ainda, todas as técnicas serão mostradas serem casos particulares da conceituação, e formas através das quais elas podem se beneficiar dessa generalização serão evidenciadas.

2 Morphing em \mathbb{R}^n

A transformação entre objetos n -dimensionais é um problema relevante em Computação Gráfica, com importantes casos particulares em dimensões mais baixas. O estudo conceitual do problema geral será essencial para estabelecer uma visão mais ampla dos freqüentemente utilizados casos 2D e 3D, ajudando a melhorar o tratamento dado a ambos. Neste capítulo, conceitos e definições fundamentais serão apresentados, analisando o problema de morphing para objetos bastante gerais, que serão particularizados para modelos mais práticos quando apropriado.

2.1 Conceitos Fundamentais

O estudo da transformação de objetos se inicia pela definição da classe de objetos relevantes e de um modelo apropriado para sua representação. Um modelo natural para um objeto real é uma coleção de todas as suas propriedades relevantes. Matematicamente, um objeto n -dimensional será descrito por uma função associando a cada ponto de um subconjunto do espaço Euclidiano \mathbb{R}^n uma tupla em um *espaço de propriedades*, descrevendo quaisquer propriedades relevantes do objeto. É útil definir as propriedades individualmente como

$$f^j: U \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^{r_j},$$

de tal forma que o objeto f agrega todas as suas k propriedades em uma m -tupla:

$$f: U \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^{r_1} \times \mathbb{R}^{r_2} \times \dots \times \mathbb{R}^{r_k}, \quad r_1 + r_2 + \dots + r_k = m$$
$$f = (f^1, f^2, \dots, f^k)$$

Mais simplesmente, um objeto f pode ser designado de uma forma mais curta como uma função

$$f: U \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m.$$

O espaço de objetos \mathcal{O} é portanto um espaço de funções $f: U \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$. Esta definição para um objeto irá se mostrar suficientemente flexível para acomodar uma larga classe de aplicações de Computação Gráfica e Processamento de Imagens.

É essencial ter uma propriedade do objeto que permita a extração de informação geométrica do modelo. Isto poderia ser dado uma *função de classificação de pertinência de pontos* (*point membership classification function*), i.e., uma função ($r_j = 1$) que retorna valores reais indicando a relação de um dado ponto em \mathbb{R}^n com o objeto. Por exemplo, a função de classificação de pontos para um object Ω poderia ser

$$f^1 = g: U \subseteq \mathbb{R}^n \rightarrow \mathbb{R}, \quad g(p) \begin{cases} > 0, & \text{if } p \text{ is outside } \Omega \\ = 0, & \text{if } p \text{ is on the boundary of } \Omega, \\ < 0, & \text{if } p \text{ is inside } \Omega \end{cases}$$

identificando regiões do domínio. O object Ω é definido implicitamente como $\Omega = g^{-1}(V)$, onde V é um subconjunto da imagem de g ; no exemplo acima V poderia ser $(-\infty, 0]$ para um objeto sólido, ou apenas $\{0\}$ para o bordo. No último caso, particularmente importante, o objeto Ω , dado por $g^{-1}(0)$, é a *superfície de nível 0* de g . Note que diferentes funções g podem ter exatamente a mesma superfície de nível 0, e podem ser usadas para representar o mesmo objeto, como mostrado no exemplo na Figura 2-1. A função de classificação de pontos pode também ser usada para especificar uma propriedade diferente definindo uma *máscara* do domínio, se interpretada como um campo de transparência contínuo. Outra propriedade frequentemente associada com objetos gráficos é a cor, representada por uma função tomando valores em um espaço de cor, isto é, uma propriedade $f^2: U \subseteq \mathbb{R}^n \rightarrow C$.

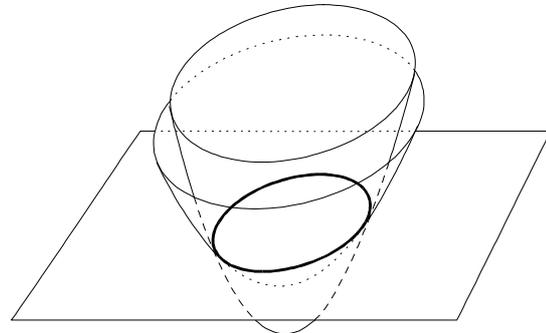


Figura 2-1: Funções diferentes com a mesma superfície de nível 0.

Um importante conceito matemático que será usado ao longo deste capítulo é o de uma *família de transformações a k-parâmetros*, uma função $f: \mathbb{R}^n \times \mathbb{R}^k \rightarrow \mathbb{R}^m$. Para cada

vetor v do *espaço de parâmetros* R^k , uma função distinta $f_v: R^n \rightarrow R^m$ é obtida, de tal forma que f efetivamente descreve uma família de funções f_v . Um espaço de parâmetros particularmente útil é o tempo, que ajuda a definir várias famílias de transformações a 1 parâmetro. Em particular, um objeto gráfico animado é uma família de transformações a 1 parâmetro

$$f: R^n \times R \rightarrow R^{r_1} \times R^{r_2} \times \dots \times R^{r_k}, \quad r_1 + r_2 + \dots + r_k = m,$$

que define uma família de objetos com coerência temporal, ou um objeto que varia com o tempo. Naturalmente, tais objetos variáveis podem ser governados por mais de um parâmetro, que não necessariamente têm um significado temporal.

2.2 Transformações de Domínio

Uma *transformação de domínio* é uma função de mapeamento $T_d: R^n \rightarrow R^n$ que produz uma mudança de coordenadas no domínio de definição de um objeto. Em geral, a relação espacial entre os pontos não é mantida por essa função de mapeamento. Uma transformação de domínio pode ser vista como uma ferramenta para criação de novos objetos deformados, como mostrado no diagrama abaixo

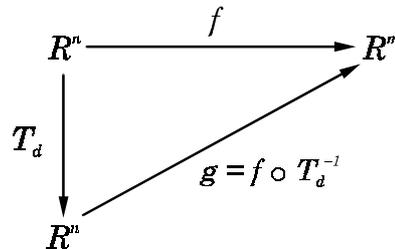


Figura 2-2: Um objeto g criado por uma transformação de domínio de f .

É útil analisar exemplos de transformações de domínio no caso unidimensional, indicadas por mudanças na abcissa do gráfico da função. Tais exemplos unidimensionais serão extensivamente utilizados neste capítulo, por serem simples de analisar e entender, possuírem as mesmas propriedades básicas, e serem fáceis de estender para dimensões mais altas.

Seja $f: \mathbb{R} \rightarrow \mathbb{R}$ um objeto definido por $f(x) = \sin(x)$ que é mostrado graficamente na Figura 2-3(a). Uma transformação de domínio poderia ser dada por $T_d(x) = \sqrt{x}$, $x \geq 0$, e o objeto resultante g , mostrado em (b), seria obtido como

$$\left. \begin{array}{l} T_d^{-1}(x) = x^2 \\ g(x) = f(T_d^{-1}(x)) \end{array} \right\} \Rightarrow g(x) = f(x^2) = \sin x^2$$

Note que o domínio é contraído para valores de x em $[0, 1/2)$, e expandido em $(1/2, +\infty)^*$.

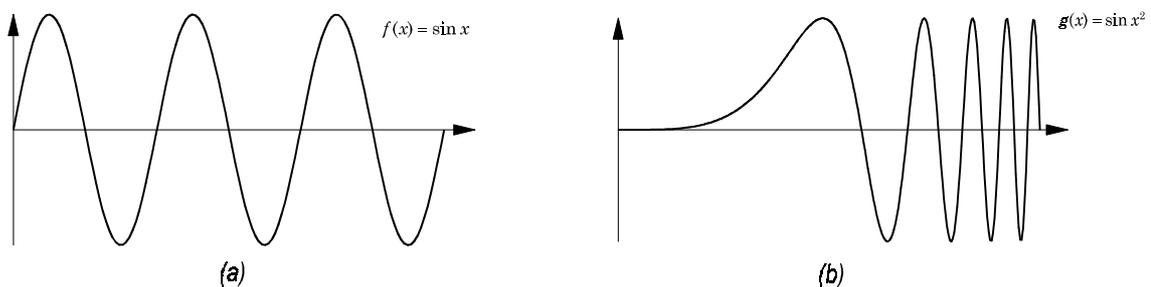


Figura 2-3: Transformação de domínio: expansão e contração

Transformações de Domínio Parametrizadas

Como pôde ser visto na Figura 2-2, a deformação de um objeto, através de uma transformação de seu domínio, envolve duas entidades: um objeto de entrada e uma transformação de domínio. Qualquer um dos dois pode variar de acordo com parâmetros apropriados, e portanto, quatro situações relativas à obtenção de um objeto parametrizado ocorrem:

- Nem o objeto de entrada nem a transformação são parametrizados;
- Apenas o objeto de entrada é parametrizado;
- Apenas a transformação é parametrizada;
- Tanto o objeto de entrada quanto a transformação são parametrizados.

* T_d é uma expansão se $|T_d(b) - T_d(a)| > |b - a|$. No exemplo, isso significa que $|b^2 - a^2| > |b - a|$. Como $|b^2 - a^2| = |b + a| |b - a|$, T_d é uma expansão se $|b + a| > 1$, e uma contração caso contrário. Para uma vizinhança de a , se $|(a + \varepsilon) + a| > 1$, T_d é uma expansão. Se $\varepsilon \rightarrow 0$, então para $|a| > 1/2$, T_d é uma expansão na vizinhança de a .

O primeiro caso resulta em um objeto de saída que não é parametrizada, exatamente como visto até o momento. Os três casos restantes serão brevemente discutidos, com exemplos unidimensionais potencialmente clarificadores.

Objeto de entrada parametrizado. O segundo caso resulta em objeto parametrizado que representa o objeto de entrada transformado pela mesma transformação T para cada vetor específico no espaço de parâmetros. Em outras palavras, o objeto original parametrizado é dado por $f: R^n \times R^k \rightarrow R^m$, que define para cada vetor v em R^k um objeto f_v . Do diagrama na Figura 2-2,

$$g_v: R^n \rightarrow R^m$$

$$g_v = f_v \circ T^{-1}.$$

Isto pode ser visto mais facilmente no exemplo unidimensional na Figura 2-4. Neste caso, o objeto de entrada é dado por $f_t(x) = x.t$ e a transformação T é uma translação dada por $T(x) = x + a$. Da composição de f_t e T^{-1} , o objeto transformado g_t é obtido como

$$\left. \begin{array}{l} T^{-1}(x) = x - a \\ g_t(x) = f_t(T^{-1}(x)) \end{array} \right\} \Rightarrow g_t(x) = f_t(x - a) = t.(x - a),$$

e o resultado—uma família de funções transladadas—é mostrado em (c).

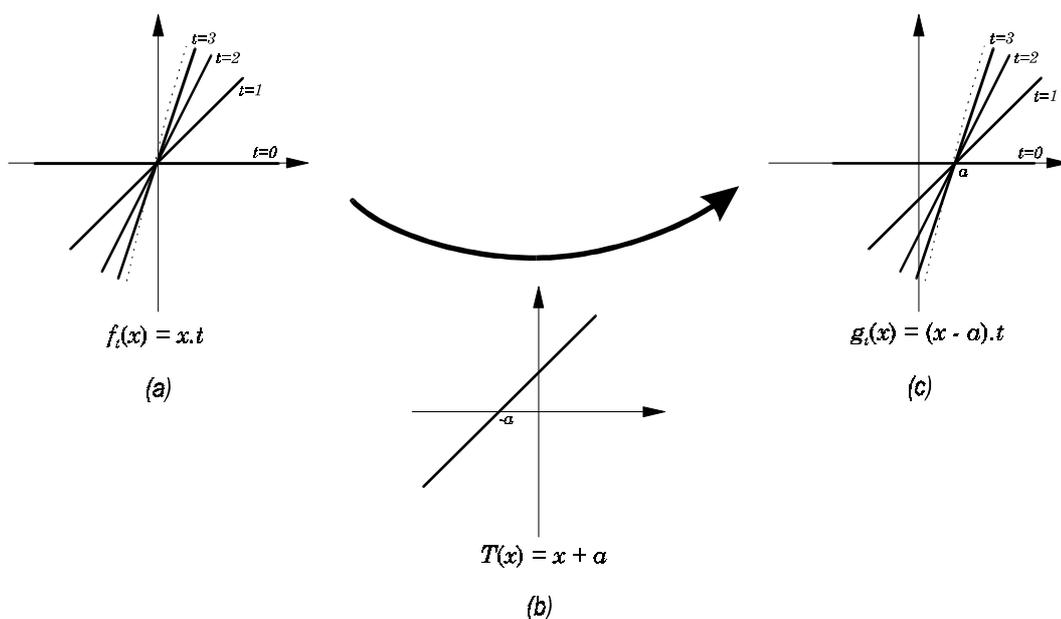


Figura 2-4: Objeto de saída parametrizado: uma translação de um objeto de entrada parametrizado

Transformação parametrizada. O caso onde apenas a transformação é parametrizada é similar. O objeto de entrada é dado simplesmente por $f: R^n \rightarrow R^m$, enquanto a transformação parametrizada é dada por $T: R^n \times R^k \rightarrow R^n$. Para cada vetor v em R^k existe uma transformação $T_v: R^n \rightarrow R^n$, e, conseqüentemente,

$$g_v: R^n \rightarrow R^m$$

$$g_v = f \circ T_v^{-1}.$$

No exemplo na Figura 2-5, o objeto unidimensional de entrada é simplesmente $f(x) = x$, e a transformação variável $T: R \times R \rightarrow R$ é dada por $T_t(x) = x + at$. O objeto transformado, mostrado em (c), é então

$$\left. \begin{array}{l} T_t^{-1}(x) = x - at \\ g_t(x) = f(T_t^{-1}(x)) \end{array} \right\} \Rightarrow g_t(x) = f(x - at) = x - at.$$

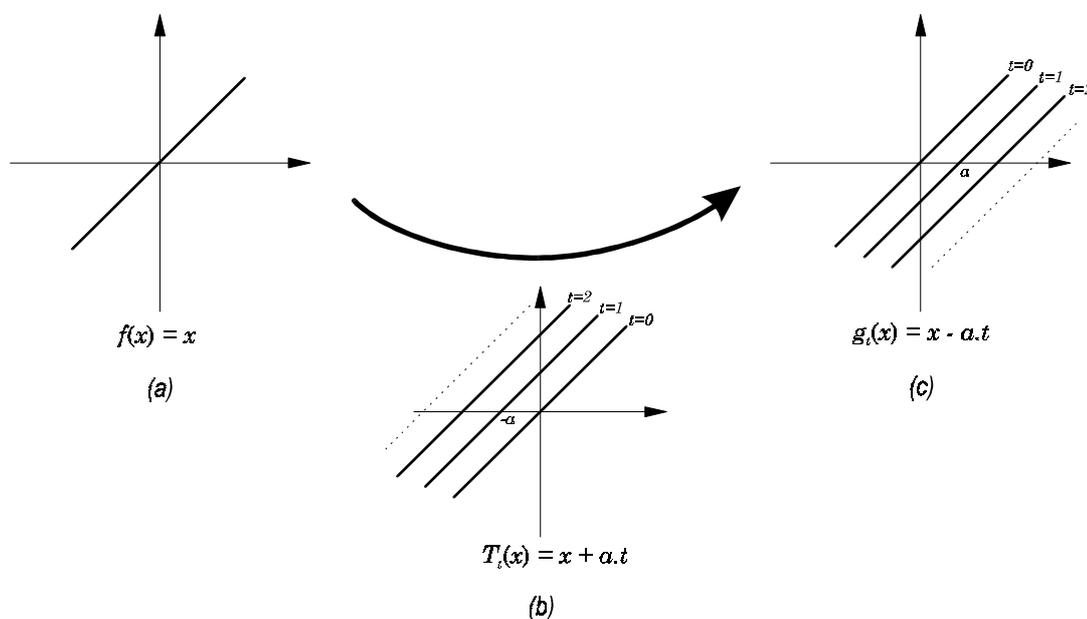


Figura 2-5: Objeto de saída parametrizado: uma translação variável de um objeto de entrada.

Objeto de entrada e transformação parametrizados. No último caso, onde tanto o objeto de entrada quanto a transformação são parametrizados, i.e., o objeto de entrada é dado por $f: R^n \times R^k \rightarrow R^m$, ou $f_u: R^n \rightarrow R^m$, e a transformação é dada por $T: R^n \times R^l \rightarrow R^n$, ou, como antes, $T_v: R^n \rightarrow R^n$. O objeto transformado é portanto uma família de transformações a $(k+l)$ parâmetros,

$$g_{uv}: R^n \rightarrow R^m$$

$$g_{uv} = f_u \circ T_v^{-1}.$$

Este é o caso do exemplo unidimensional na Figura 2-6, onde o objeto de entrada é dado por $f_u(x) = x.u$ e a transformação por $T_v(x) = x + a.v$. Portanto, o objeto de saída g_{uv} é dado por

$$\left. \begin{array}{l} T_v^{-1}(x) = x - av \\ g_{uv}(x) = f_u(T_v^{-1}(x)) \end{array} \right\} \Rightarrow g_{uv}(x) = f_u(x - av) = u(x - av),$$

que é mostrado em (c), para uma escolha particular de parâmetros u e v . Qualquer curva no espaço de parâmetros relacionando u e v pode ser usada; no exemplo, a curva $u = v$ foi selecionada. Note que (c) é uma combinação natural dos “movimentos” em (a) e (b).

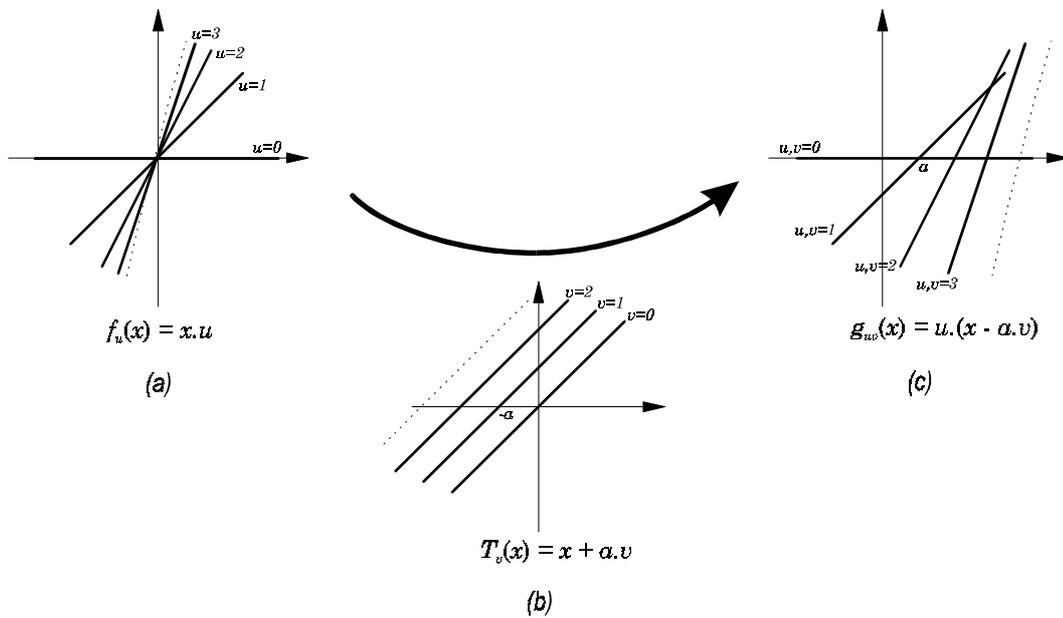


Figura 2-6: Um objeto de saída parametrizado: objeto de entrada e transformação são variáveis

2.3 Transformações de Contra-domínio

Um objeto também pode ter seu contra-domínio transformado para criar um novo objeto, ao invés de seu domínio. Neste caso, a relação espacial entre os pontos não é modificada pela transformação, que altera apenas os valores das propriedades dos objetos. Como geralmente propriedades distintas têm natureza diferente, cada propriedade deve ser transformada por uma transformação separada. Lembre-se que um objeto é definido como uma coleção de suas propriedades f como

$$f: U \subseteq R^n \rightarrow R^{r_1} \times R^{r_2} \times \dots \times R^{r_k}, \quad r_1 + r_2 + \dots + r_k = m$$

$$f = (f^1, f^2, \dots, f^k)$$

$$f^j: U \subseteq R^n \rightarrow R^{r_j},$$

Para cada propriedade a ser transformada separadamente, uma transformação de propriedade T^i é definida* e T é uma coleção de tais transformações:

$$T^i: R^{r_1} \times R^{r_2} \times \dots \times R^{r_k} \rightarrow R^{r_i},$$

* Esta definição assume que a transformação de uma propriedade depende dos valores das outras propriedades. Caso contrário, uma transformação T^i poderia ser definida como $T^i: R^m \rightarrow R^{r_i}$.

$$T: R^{r_1} \times R^{r_2} \times \dots \times R^{r_k} \rightarrow R^{r_1} \times R^{r_2} \times \dots \times R^{r_k} \text{ or, simply, } T: R^m \rightarrow R^m$$

$$T = (T^1, T^2, \dots, T^k).$$

A aplicação de uma transformação de contra-domínio a um objeto está esquematicamente apresentada no diagrama na Figura 2-7. O domínio R^n do objeto f permanece intocado e o contra-domínio é transformado por T_r . O objeto resultante g é então dado por

$$g: R^n \rightarrow R^m$$

$$g = T_r \circ f.$$

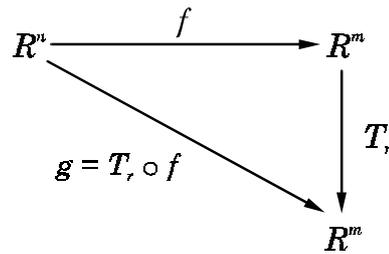


Figura 2-7: Diagrama de transformação de contra-domínio.

O simples exemplo unidimensional na Figura 2-8 sumariza a discussão acima, evidenciando o caráter de modificação de amplitude de transformações de contra-domínio. Tomando um objeto $f(x) = \sin(x)$ e uma transformação $T(x) = x/2$, o objeto de saída $g(x)$, mostrado em (c), é dado por $g(x) = T(f(x)) = \sin(x)/2$.

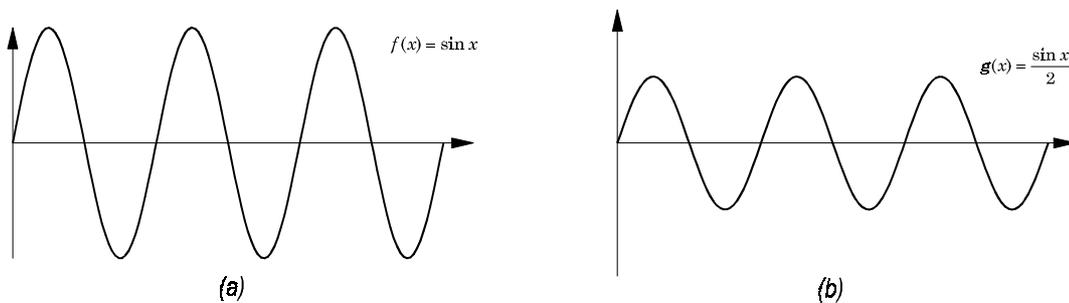


Figura 2-8: Transformação de contra-domínio de um objeto

Transformações de Combinação

É também útil, e conceitualmente equivalente, ter transformações de contra-domínio definidas como $T: R^m \times \dots \times R^m \rightarrow R^m$, onde a transformação T é uma função que mapeia vários espaços de propriedades em outro. Neste caso freqüentemente utilizado, chamado combinação, ou *blending*, o objeto de saída $g: R^n \rightarrow R^m$ é obtido a partir dos objetos de entrada f_1, f_2, \dots, f_k simplesmente como $g = T(f_1, f_2, \dots, f_k)$.

A investigação de algumas formas de blending bem conhecidas prova ser útil no desenvolvimento de técnicas de metamorfose. Para este tipo de aplicação blends geralmente resultam em transições suaves entre os objetos de entrada, e no caso mais freqüente, são usados para combinar apenas dois objetos. Por exemplo, dados dois objetos $f_1(x) = \sin(x)$ e $f_2(x) = x/3$, e uma transformação $T(x,y) = x.y$, o objeto combinado g é obtido como:

$$g(x) = T(f_1(x), f_2(x)) = T(\sin x, x/3) = \frac{x \sin x}{3}.$$

Os objetos f_1 e f_2 e sua combinação g estão ilustrados na Figura 2-9 abaixo.

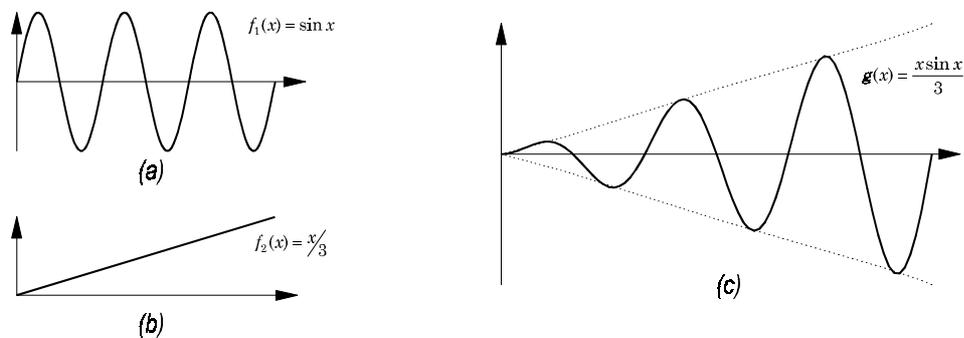


Figura 2-9: Transformação de contra-domínio de dois objetos

Funções de blending são um importante tópico de pesquisa, tendo produzido uma larga variedade de técnicas diferentes, cada uma apropriada a aplicações particulares [Rockwood, Owen 87]. Como um objeto é definido por um conjunto de propriedades distintas e combinar dois objetos implica na combinação de propriedades correspondentes, a diversidade de funções de blending é fundamental no estabelecimento de tais combinações. O blending de uma propriedade como cor certamente tem requerimentos diferentes daqueles de uma propriedade como forma.

Blending resulta em uma função onde uma certa característica é uma transição suave entre os valores da mesma característica nas funções de entrada. A função característica que deve ser combinada depende unicamente na natureza da propriedade particular do objeto, e no caso de morphing, duas situações são especialmente importantes:

- combinar valores das funções;
- combinar certas superfícies de nível das funções.

No primeiro caso, o valor da função em cada ponto é uma transição suave entre os valores das funções de entrada no mesmo ponto. No segundo, a superfície de nível zero da função é uma transição suave entre as superfícies de nível zero das funções de entrada.*

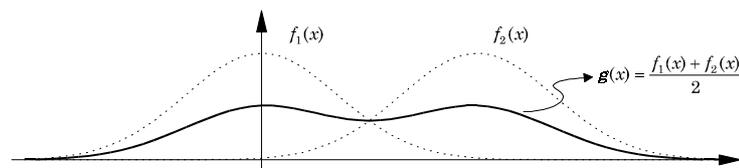


Figura 2-10: Combinação por média de duas funções

Por exemplo, uma propriedade como intensidade de cor poderia ser representada graficamente como mostrada nas curvas pontilhadas na Figura 2-10. A propriedade resultante de uma combinação de tais funções claramente tomaria valores na faixa entre uma função e a outra, como a média mostrada como uma curva grossa na figura. Por outro lado, para uma propriedade como forma—dada por, por exemplo, uma superfície de nível zero de uma função de classificação de pontos implícita—a combinação deveria resultar em uma função onde a superfície de nível zero é uma transição suave entre as superfícies de nível zeros das funções de entrada, não interessando os valores particulares das funções. A Figura 2-11 torna esta distinção mais clara, mostrando diferentes combinações de duas funções que poderiam ser interpretadas como funções de classificação de pontos. Note que enquanto o blending exponencial resulta em uma combinação natural das superfícies de nível das duas parábolas, o blending média gera

* Note que uma superfície de nível zero pode ser usada sem perda de generalidade: a superfície de nível K $f(x) = K$ coincide com a superfície de nível zero de $g(x) = f(x) - K$.

uma função com uma superfície de nível zero nula. Comparando com a Figura 2-10, este resultado claramente mostra a relação entre a técnica de blending e a interpretação dos dados a serem combinados.

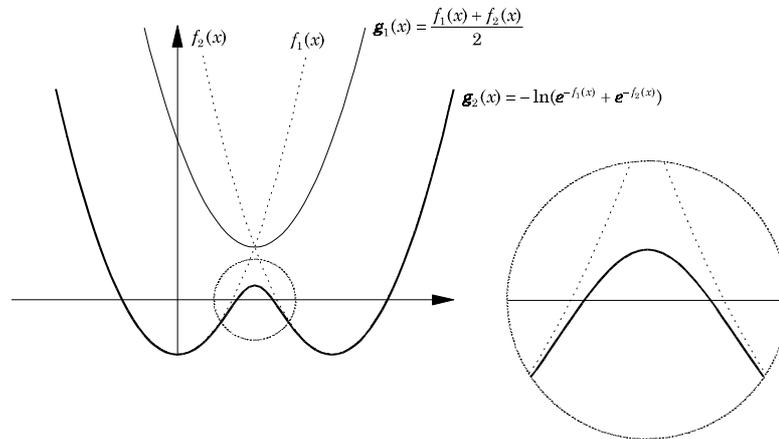


Figura 2-11: Combinações diferentes entre duas parábolas

Um fato importante a perceber é que uma mudança na forma de um objeto pode ser devido tanto a uma transformação de domínio quanto a uma transformação de contra-domínio da propriedade forma, especialmente se a forma for a única propriedade do modelo. Para morphing, warping é geralmente utilizado para grosseiramente alinhar as características, e blending de forma é usado para combinar os objetos, encaixando os detalhes mais finos.

Transformações de Contra-domínio Parametrizadas

Similarmente a transformações de domínio, transformações de contra-domínio também podem ser definidas como funções parametrizadas. Combinadas com objetos de entradas parametrizados, quatro situações, análogas àquelas discutidas na seção 2.2, surgem:

- Nem o objeto de entrada nem a transformação são parametrizados;
- Apenas o objeto de entrada é parametrizado;
- Apenas a transformação é parametrizada;
- Tanto o objeto de entrada quanto a transformação são parametrizados;

As últimas três situações resultam em objetos parametrizados que podem ser representados por famílias de transformações, como já foi feito antes.

2.4 Morphing

É naturalmente possível criar um objeto novo através da transformação tanto do domínio quanto do contra-domínio do original. Uma transformação de morphing é uma combinação de transformações de domínio e contra-domínio tal como esta, e portanto, é um mapeamento no espaço de objetos $M: O \rightarrow O$, que associa ao objeto original f um objeto transformado $M(f)$. Como mostrado no diagrama na Figura 2-12, o domínio do objeto f é transformado por uma transformação de domínio $T_d: R^n \rightarrow R^n$ e o contra-domínio de f é transformado por uma transformação de contra-domínio $T_r: R^m \rightarrow R^m$. O objeto resultante g é portanto uma composição de transformações de domínio e contra-domínio, como indicado no diagrama.

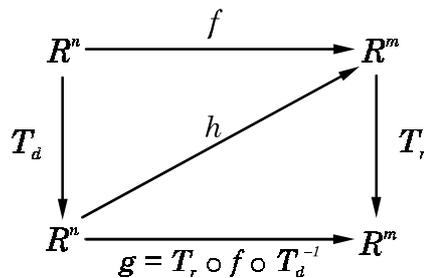


Figura 2-12: Diagrama de morphing.

Morphing é evidentemente uma generalização das transformações previamente discutidas: se T_d é a transformação identidade, o objeto g é simplesmente uma transformação de contra-domínio de f , por outro lado, se T_r é a transformação identidade, g é uma transformação de domínio de f . Note que, em algumas aplicações, o objeto g não é obtido diretamente de f . Ao contrário, um objeto intermediário h é criado em um primeiro passo por uma transformação de domínio de f , e então uma transformação de contra-domínio é aplicada à h para obter g , como se segue:

$$h = f \circ T_d^{-1}$$

$$g = T_r \circ h.$$

Obviamente, se qualquer uma das transformações que compõem uma transformação de morphing for parametrizada, o resultado será parametrizado, e, similarmente, se qualquer dos objetos de entrada for parametrizado, o resultado também será parametrizado.

3 Morphing em \mathbb{R}^2

Neste capítulo, os conceitos desenvolvidos no estudo do problema de morphing n -dimensional serão aplicadas a objetos de dimensão 2. A importância da forma de representação de tais objetos será reforçada, com ênfase em como diferentes representações foram usadas em trabalhos anteriores. Em particular, os problemas e técnicas de morphing de objetos 2D em duas das representações serão discutidos: desenhos estruturados, ou simplesmente, *desenhos*, e *imagens*. O último é agora um problema bem conhecido, com soluções bastante efetivas usadas na prática, e o primeiro também tem importantes resultados de pesquisa e aplicações práticas.

O capítulo 2 mostrou que morphing 2D e morphing em dimensões mais altas são problemas fortemente relacionados, e, de fato, compartilham uma conceituação comum. Portanto, este capítulo procura endereçar o problema bidimensional não como um fim em si, mas como um possível caminho para se detectar soluções para morphing em mais altas dimensões a partir das soluções de morphing 2D bem conhecidas.

3.1 Modelos 2D e suas Representações

Neste capítulo, a preocupação primária são os objetos bidimensionais, um caso particular dos objetos definidos no capítulo 2. Lembre-se que um objeto n -dimensional foi descrito na seção 2.1 como uma função associando a cada ponto em um subconjunto do espaço Euclidiano \mathbb{R}^n uma tupla em um espaço de propriedades, descrevendo quaisquer propriedades relevantes do objeto. Aqui, como $n = 2$, um objeto bidimensional será descrito por uma função

$$f: U \subseteq \mathbb{R}^2 \rightarrow \mathbb{R}^{r_1} \times \mathbb{R}^{r_2} \times \dots \times \mathbb{R}^{r_k}, \quad r_1 + r_2 + \dots + r_k = m$$
$$f = (f^1, f^2, \dots, f^k)$$

combinando as k propriedades do objeto, cada uma delas uma função

$$f^j: U \subseteq \mathbb{R}^2 \rightarrow \mathbb{R}^j.$$

Para objetos geométricos, uma descrição de sua forma precisa estar presente nesta definição. Usualmente, isto é feito através de uma propriedade especial que é uma função de classificação de pertinência de pontos, indicando para cada ponto no domínio de definição U do objeto se ele pertence ao objeto ou não. A Figura 3-1 mostra um objeto bidimensional simples, com um interior cinza e um bordo preto.

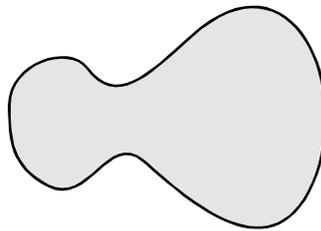


Figura 3-1: Um objeto 2D simples.

A forma como esses objetos são representados computacionalmente, no entanto, não lembra necessariamente a definição acima. Estruturas de dados apropriadas são usadas para fornecer basicamente a mesma informação geométrica (e topológica) e também valores para quaisquer outras propriedades. Há basicamente duas formas de representação que são largamente utilizadas para objetos bidimensionais na prática: *imagens digitais* e *desenhos*.

Imagens Digitais

Imagens digitais são descritas por um conjunto finito de elementos de imagem (*pixels*), cada um deles descrevendo as propriedades do objeto para uma pequena vizinhança. As propriedades são geralmente assumidas como sendo constantes por todo o pixel, embora esse não seja o caso usual. Estes pixels são normalmente dispostos em uma estrutura regular, de tal forma que a descrição da imagem é tornada mais simples, evitando a descrição da geometria de cada pixel. Um caso freqüente, onde nenhuma informação de geometria por pixel é necessária, é o de um grid retangular regular, onde todos os pixels são quadrados de mesmo tamanho. Isto permite que a imagem seja descrita de uma

forma tabular por uma matriz onde cada elemento descreve as propriedades do pixel correspondente da imagem.

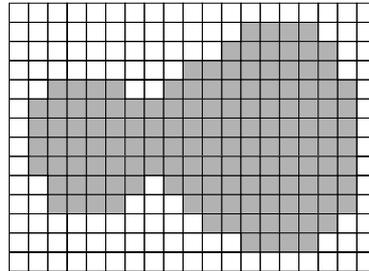


Figura 3-2: Uma imagem digital.

Para imagens digitais, a propriedade padrão associada a cada pixel é a cor ou intensidade do objeto naquela região. Esta informação é suficiente para a maior parte das aplicações simples com imagens, incluindo a sua visualização como na Figura 3-2. Uma outra propriedade interessante é a chamada canal α , que fornece, para cada pixel, uma informação de transparência que efetivamente restringe a definição da imagem a um subconjunto irregular da matriz da imagem. Onde o canal α é zero, a imagem é considerada indefinida, e os valores das propriedades destes pixels não são usados; para valores de canal α não nulos, ou o valor inteiro da propriedade naquele pixel é usado, ou as propriedades são atenuadas pelo valor do canal α . Isto é uma versão amostrada da função característica que descreve a forma da imagem, desconsiderando a atenuação de propriedade. Esta informação é fundamental na obtenção de resultados de boa qualidade em aplicações de composição de imagens digitais [Porter, Duff 84]. Outras propriedades são necessárias para aplicações mais específicas, como a informação de profundidade do pixel—usualmente chamada de *z-buffer*—que torna as imagens descrições de uma classe bastante particular de objetos 3D, útil para a composição de imagens com profundidade, usadas na visualização de objetos 3D mais genéricos.

Desenhos

Desenhos são usados em situações onde uma discretização do domínio de definição do objeto é indesejável. Como descrito em [Gomes, Velho 94], tais formas de discretização trazem uma série de problemas, a maioria deles relacionadas aos processos de amostragem e reconstrução envolvidos em transformações geométricas de objetos discretos. Uma representação por desenho evita este tipo de problema através da

descrição da fronteira do objeto com uma precisão arbitrária, por um conjunto de segmentos de linha ou curvas conectados, como mostrado na Figura 3-3. Transformações geométricas são facilmente executadas através da transformação apenas dos pontos finais dos segmentos, se os segmentos são invariantes para aquele tipo de transformação geométrica particular. Segmentos de linha são invariantes sob transformações afins, e são usualmente considerados serem invariantes sob outras transformações se os segmentos são curtos o suficiente, e a transformação é no mínimo contínua.

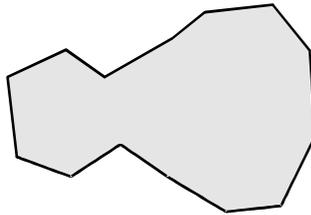


Figura 3-3: Um desenho bidimensional

Para desenhos, as propriedades são algumas vezes tomadas como constantes ao longo de todo o objeto, devido à dificuldade inerente em descrever tais propriedades localmente usando esta representação. Diferentes valores de propriedades podem ser facilmente associados a cada segmento da fronteira do desenho, mas suas definições para o interior do objeto irão requerer usualmente interpolações caras. Para cor, em particular, algumas funções de variação lenta são freqüentemente utilizadas para descrever efeitos *degradée* para todo o objeto. Uma abordagem mais pragmática é o uso de imagens combinadas com desenhos: uma imagem digital descrevendo quaisquer propriedades relevantes é mapeada para o interior do desenho, efetivamente texturizando o objeto [Heckbert 86]. Usualmente, esta textura não está sujeita a transformações geométricas, apesar do desenho estar—o efeito resultante pode ser eventualmente decepcionante, mas não existem problemas de discretização. Este tipo de associação de propriedades com desenhos são largamente utilizadas em aplicações de desenho sob o nome genérico de *gráficos orientados a objetos*.

Representações 2D por Bordo e Volume

Esta abordagem pragmática à modelagem de objetos 2D é consistente com o paradigma de representação usado na modelagem e representação de objetos 3D. Há duas representação básicas para sólidos 3D [Gomes, Velho 93]:

- **Bordo** – um sólido é representado por uma descrição de seu bordo;
- **Volume** – um sólido é representado por uma descrição de seu interior.

A descrição geométrica tanto de objetos representados por fronteira quanto por volume pode ser dada de duas maneiras:

- **Paramétrica** – o conjunto de pontos constituindo o objeto é dado *diretamente* por uma coleção de mapeamentos ou parametrizações, estabelecendo um mapeamento bijetor entre o espaço de parâmetro e o objeto.
- **Implícita** – o conjunto de pontos constituindo o objeto é dado *indiretamente* através do uso de uma função de classificação de pertinência de pontos.

Naturalmente, cada forma de descrição pode ser representada exatamente ou como uma aproximação. Os bem conhecidos modelos B-Rep, por exemplo, aproximam a fronteira de objetos por funções paramétricas lineares por partes, i.e., uma coleção de faces planares. Descrições implícitas podem ser usadas tanto para representar exatamente o interior quanto o bordo de um objeto. Um modelo volumétrico é, na verdade, uma aproximação discreta de uma descrição implícita do volume do objeto, dado por uma coleção de valores que especificam diretamente a função de classificação de pontos.

A situação bidimensional é análoga: imagens digitais são uma forma particular de aproximação discreta de uma área descrita implicitamente, e desenhos são uma forma de representação por bordo paramétrica aproximada. Objetos 2D definidos implicitamente não são usados freqüentemente, mas são certamente viáveis para alguns tipos de computação, especificamente, a classificação de pontos contra um objeto e a combinação de formas de objetos. Neste trabalho, a representação implícita será especialmente relevante por sua proximidade com a conceituação de morphing e sua possibilidade de extensão direta para 3D.* Naturalmente, é teoricamente possível converter entre algumas das representações, apesar de problemas de precisão estarem freqüentemente presentes.

* De fato, objetos implícitos 2D foram usados como tubo de ensaio para uma técnica, descrita no capítulo 5, desenvolvida para metamorfose de objetos 3D.

3.2 Transformações de Domínio e Contra-domínio

Transformações de domínio e contra-domínio práticas são bastante diferentes, dependendo diretamente da representação do objeto. A especificação, representação e computação de transformações de domínio e contra-domínio envolvem vários níveis de manipulação das estruturas de dados representando o objeto sob a transformação. Transformar o contra-domínio de um objeto—suas propriedades—pode ser mais fácil ou mais difícil que transformar o seu domínio de definição, dependendo do tipo de representação. Além disso, algumas das técnicas são inequivocamente ligadas a certas formas de representação, dependendo de algumas de suas particularidades para alcançar os resultados desejados.

O trabalho anterior em morphing 2D concentrou-se em imagens e desenhos, como notado anteriormente. Para imagens, a transformação de contra-domínio, i.e., a alteração dos valores dos pixels, é usualmente muito simples, sem nenhuma complicação além da média relativamente padrão envolvida [Costa 94]. Transformações de domínio, entretanto, são complicadas por importantes problemas de amostragem e reconstrução, e possuem uma ampla variedade de técnicas de especificação e implementação disponíveis. Em morphing de imagens, a técnica de warping usualmente indica as formas de representação possíveis, a eficiência computacional, o tipo de interface com o usuário e até mesmo, as técnicas possíveis para transformação de contra-domínio. Naturalmente, a maior parte do trabalho em morphing de imagens foi direcionada para as técnicas de warping, e por isso, essa é a área de onde outras formas de morphing têm mais a aprender. Na seção 3.4, as técnicas usadas para deformar imagens digitais serão brevemente discutidas, com ênfase naquelas mais apropriadas para extensão para dimensões mais altas. A extensão de algumas das técnicas para três dimensões será discutida no capítulo 4.

Para desenhos, poucos esforços foram feitos na transformação de contra-domínio para transformar propriedades outras que a forma do objeto, provavelmente devido à inerente dificuldade na representação destas propriedades em desenhos, como observado anteriormente. Lembre-se que uma mudança na forma de um objeto pode ser devida tanto a uma transformação de domínio quanto a uma transformação de contra-domínio da propriedade forma, como explicado na seção 2.3. É natural entender o trabalho anterior em combinação de desenhos como uma transformação de contra-domínio ao

invés de uma transformação de domínio, apesar de resultados equivalentes poderem ser obtidos por ambos. Uma transformação de domínio, contudo, alteraria não apenas a forma do contorno, mas influenciaria também a disposição dos valores de outras propriedades. Será mostrado que as transformações de domínio podem ser combinadas com essas técnicas de combinação de formas para obter transformações de morphing mais completas que são mais fáceis de controlar.

Especificação e Implementação

Em todos os processos cliente-servidor, alguma forma de especificação do resultado desejado precisa ser dado pelo cliente, de forma que o servidor seja capaz de realizar o que foi especificado. Para morphing, esta distinção é especialmente importante, tendo permitido a detecção de novas combinações de técnicas existentes.

Morphing na prática requer uma certa quantidade de especificação por parte do usuário para alcançar os resultados desejados. Técnicas totalmente automáticas não existem no momento, embora alguns esforços nessa direção tenham sido feitos. Ainda assim, a forma de especificação e a interface com o usuário são tópicos essenciais em técnicas de morphing atuais. Ainda que a forma de especificação esteja geralmente associada com a técnica sendo usada, é possível dissociar as duas, e ter diferentes formas de especificação com a mesma técnica sendo utilizada para computar a transformação [Costa et alli 92].

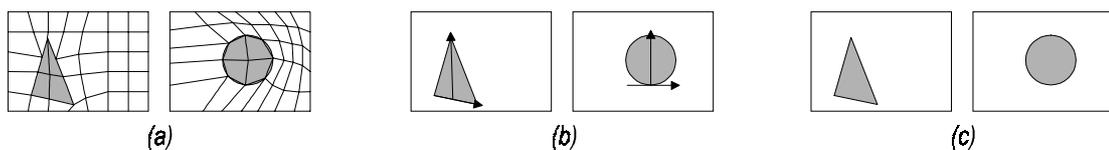


Figura 3-4: Especificação de morphing: (a) partição; (b) características; (c) semi-automática

Algumas classificações de técnicas de especificação de morphing são possíveis. Uma classificação relativamente geral é dada em [Costa et alli 92; Costa 94], onde técnicas de especificação são agrupadas como sendo baseadas em partições ou baseadas em características. Uma técnica baseada em partição descreve a deformação das características correspondentes para todo o domínio (Figura 3-4(a)); uma técnica baseada em característica descreve a deformação apenas para algumas características selecionadas (Figura 3-4(b)). Esta classificação, contudo, não é geral o suficiente para

compreender técnicas automáticas e semi-automáticas que já existem para desenhos. Para tais técnicas, a correspondência de características não está presente na especificação, já que ela é totalmente realizada pelo algoritmo. Ainda é necessário, contudo, especificar outros parâmetros que auxiliam o controle do progresso da transformação (Figura 3-4(c)).

3.3 Técnicas de Combinação de Formas 2D

Dados dois objetos 2D representados por seus bordos de uma forma aproximada, i.e., dois desenhos, o objeto das técnicas de blending, ou combinação, é obter uma transformação contínua da forma de um objeto até o outro, ou passos desta transformação. Isto corresponde a obter uma família de transformações a 1 parâmetro, ou amostras desta família para alguns valores de parâmetro, como descrito detalhadamente no capítulo 2. Como mencionado anteriormente, tais combinações de forma são consideradas transformações de contra-domínio, e podem ser precedidas por uma fase de warping para alcançar um morphing completo, com alto nível de controle

Em geral, blendings produzem pequenas alterações na forma dos desenhos, que devem ser relativamente similares. Técnicas mais avançadas podem eliminar a necessidade de uma fase de warping em algumas circunstâncias, transformando desenhos que diferem moderadamente de uma só vez, com excelentes resultados [Sederberg, Greenwood 92], mas uma fase de warping irá sempre ajudar a alcançar resultados específicos dando ao usuário um controle adicional. Estas técnicas avançadas são freqüentemente automáticas, guiadas por alguma forma de heurística, já que transformações são um problema inerentemente ambíguo: um blending entre dois objetos é um problema com múltiplas soluções (veja Figura 3-5), sem uma única resposta “correta”; a escolha da melhor solução é um problema perceptual, e a interação com o usuário é necessária para indicar a melhor alternativa para uma aplicação particular. Tentativas de definir uma métrica da qualidade de um blending permitem que uma discutível melhor solução possa ser escolhida por um algoritmo de otimização, como mostrado na seção de Blending Físico.

A estrutura combinatória dos desenhos usados para blending pode ser idêntica, i.e., os desenhos contêm exatamente o mesmo número de vértices e arestas estruturados da mesma forma, ou diferente. Em qualquer dos casos, o problema da ambigüidade

descrito ainda existe, e o alcance de uma solução “ótima” pode requerer a inserção de vértices, até mesmo para desenhos identicamente estruturados.

Blending Linear Simples

Para dois desenhos com a mesma estrutura combinatória, e assumindo que os passos intermediários terão esta mesma estrutura, um blending das suas formas é trivial. Dado um vértice de referência em cada um dos desenhos, os vértices remanescentes são seqüencialmente associados, e uma interpolação linear é realizada entre cada um desses pares de vértices. A escolha dos vértices referência é um fator importante na produção de bons resultados, como mostrado nas duas possibilidades na Figura 3-5.

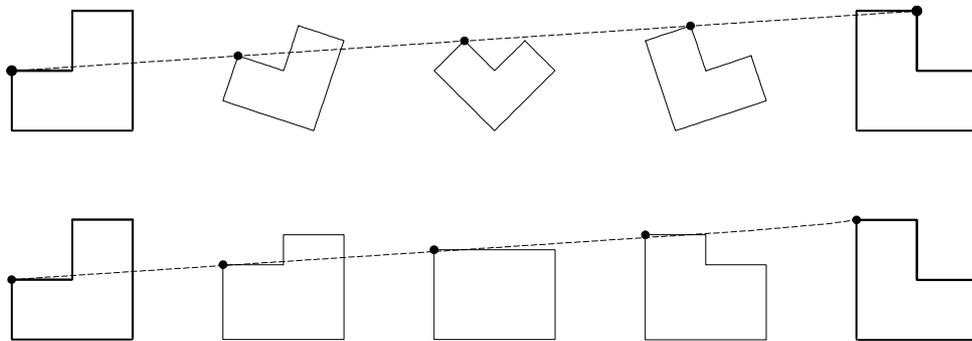


Figura 3-5: Duas possíveis associações de vértices de referência e os blendings respectivos.

Naturalmente, o caso usual são dois desenhos de entrada com estruturas combinatórias diferentes, e neste caso, para criar um blending linear, é necessário tornar as estruturas compatíveis através da inserção de novos vértices em ambas as estruturas. Como apontado em [Sederberg, Greenwood 92], a tarefa principal em blending de formas 2D é a de adicionar tais vértices de forma que a correspondência seqüencial de vértices resultante produza a combinação desejada. Várias heurísticas são possíveis, incluindo uma distribuição homogênea direta dos vértices proporcionalmente ao número total de vértices existentes, usada por pacotes comerciais de desenho populares [Corel 94], ou uma distribuição proporcional ao perímetro dos contornos.

Blending Físico

Esta técnica é de fato uma heurística elaborada que seleciona as posições para inserir os vértices de forma que a correspondência de vértices resultante produza um blending que minimize trabalho, de acordo com um modelo de energia apropriado [Sederberg,

Greenwood 92]. Os desenhos poligonais 2D são considerados como construídos de um arame de metal ideal, de tal forma que o trabalho de flexão e de extensão/contração do arame possa ser computado. O melhor blending é considerado aquele que requer a mínima quantidade de trabalho para transformar uma forma na outra, dobrando e esticando o arame.

Dada esta métrica de trabalho, o problema agora é o de otimizar as posições dos pontos inseridos, de tal forma que um mínimo de trabalho seja executado. Obviamente, este problema como apresentado é de grande complexidade, mas a restrição de inserir novos vértices somente na posição de vértices existentes, proposta em [Sederberg, Greenwood 92], reduz a complexidade do problema a $O(mn)$, onde m e n são os números de vértices de cada desenho.

Uma consequência interessante da minimização de trabalho realizado é que características relacionadas dos dois desenhos tendem a ser automaticamente associadas. Se há uma característica de um objeto que casa com uma característica do outro objeto, e essas características estão posicionadas relativamente próximas, a transformação de uma característica na outra requer uma quantidade de trabalho muito pequena. Em particular, características que não movem não requerem nenhum trabalho para serem transformadas, e portanto, não movem durante o blending. Isto é mostrado no exemplo na Figura 3-6, onde os pés permanecem estáticos, enquanto o resto é transformado. Para características que são relativamente distantes, contudo, o algoritmo tende a confundir as características, produzindo resultados não naturais. Neste caso, uma fase de warping seria extremamente favorável, por trazer as características relacionadas para posições mais próximas.

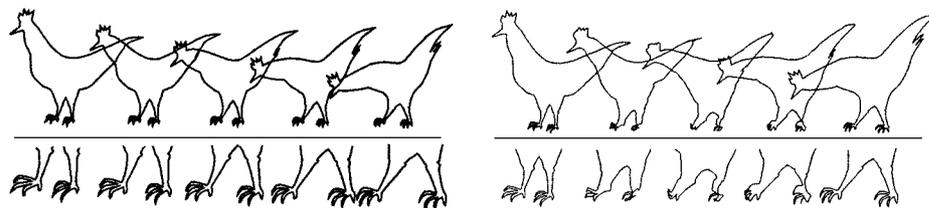


Figura 3-6: (a) Blending físico; (b) Software comercial [Sederberg, Greenwood 92].

Note que, apesar da métrica proposta por Sederberg e Greenwood poder alcançar excelentes resultados, o fato de o trabalho ser uma quantidade com sinal, que pode se

cancelar durante uma transformação, pode levar a soluções globalmente ótimas peculiares. Considere, por exemplo, a transformação no topo na Figura 3-5, onde rotação, encolhimento e esticamento são realizados e, paradoxalmente, nenhum trabalho é realizado. Com esta métrica, esta transformação é obviamente ótima, já que ela executa o blending com o menor trabalho possível. Isto é devido ao cancelamento do encolhimento com o esticamento, e ao trabalho rotacional, que não é considerado na métrica.

A procura de métricas melhores é um objetivo importante do blending físico, já que a técnica para a solução do problema de otimização permanece a mesma. Entre os melhoramentos possíveis para esta heurística estão a consideração da soma total da quantidade absoluta de trabalho necessária, o pico de trabalho realizado durante a transformação e o trabalho rotacional total.

Esta técnica está obviamente direcionada a objetos representados por bordos poligonais, apesar de uma extensão para lidar com imagens digitais ser concebível. Isto pode ser conseguido começando pela obtenção de uma descrição da borda das regiões da imagem, seja diretamente especificada pelo usuário, seja através do uso de um algoritmo de detecção de contornos. Estes contornos são então transformados com a realização do menor trabalho possível; é preciso apenas escolher um mapeamento apropriado para a região interior dos bordos.

Combinação de Estruturas Combinatórias 2D

Esta técnica de blending, originalmente desenvolvida para a combinação de objetos 3D descritos por fronteira, é descrita em detalhe no próximo capítulo, na seção Combinação de Estruturas Combinatórias. A técnica original 3D e a sua extensão para 2D são analisadas e comparadas à técnica baseada em conhecimentos físicos descrita acima.

3.4 Técnicas de Warping de Imagens

Warping de imagens tem estado em uso muito antes do advento do morphing de imagens, e várias técnicas foram desenvolvidas para unicamente deformar imagens. Técnicas de deformação de imagens apropriadas quando morphing é o objetivo são usualmente diferentes, contudo, e foram desenvolvidas mais recentemente. Isto se deve à necessidade de correlacionar características em imagens diferentes, de tal forma que o

alinhamento dessas características seja possível, e também ao requerimento de um controle da transformação mais detalhado e localizado. Como mencionado anteriormente, a transformação de contra-domínio—neste caso um cross-dissolve da propriedade cor—é usualmente muito simples para imagens, e não será discutido aqui. Uma discussão do problema completo para imagens pode ser encontrada em [Costa 94].

Há três técnicas principais conhecidas para implementação de warping de imagens: malhas de triângulos, malhas de splines em dois passos, e baseada em campo. Uma quarta técnica, a malha de splines controlada por campo originalmente apresentada em [Costa et alli 92], é o algoritmo de malha de splines em dois passos utilizado com uma especificação diferente e mais fácil de usar.

Warping por Malhas de Triângulos

Neste algoritmo, a deformação é descrita por duas malhas triangulares, uma definindo um sistema de coordenadas não deformado, e a outra uma deformação deste sistema (veja Figura 3-7). As duas malhas precisam ter a mesma estrutura combinatória, de tal forma que a correspondência entre pares de triângulos esteja bem definida. Um mapeamento precisa ser estabelecido entre os interiores de cada par de triângulos correspondentes, de tal forma que exista um mapeamento de sistemas de coordenadas global. Um mapeamento de triângulos apropriado é o uso de coordenadas baricêntricas, que são fáceis de implementar e produzem resultados aceitáveis, mas são descontínuas nas arestas de triângulos adjacentes, criando um mapeamento global linear por partes. Esquemas de interpolação de mais alta ordem podem ser usados, mas uma triangulação regular reduz as discontinuidades induzidas por coordenadas baricêntricas a um nível aceitável, por um custo muito menor.

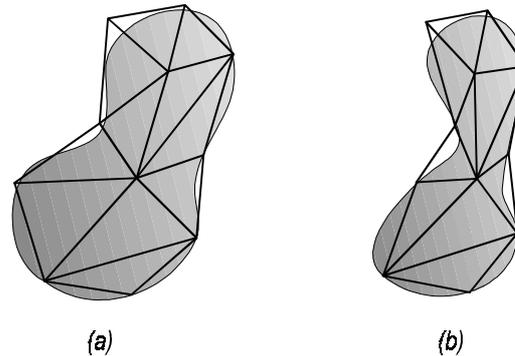


Figura 3-7: Deformação por malha de triângulos: (a) original; (b) deformada.

As malhas de triângulos usadas como entrada para este algoritmo podem ser especificadas de diversas formas. A forma mais rude é a construção direta de ambas as malhas de triângulos, garantindo a sua consistência. Obviamente, isto é um trabalho tedioso com tendência a erro, e as estruturas de dados necessárias são de manutenção inconveniente. Uma alternativa muito melhor é o uso de uma forma de especificação por características, onde as características são marcadas como pontos ou segmentos de reta, e a aplicação de um algoritmo de triangulação automática para construir a malha. Uma triangulação automática que produz triângulos regulares, como a triangulação de Delaunay [Preparata, Shamos 85], possivelmente com restrições, é uma escolha natural.

Este algoritmo pode ser usado para deformar imagens digitais através da rasterização de cada triângulo [Foley et alli 90] no estado deformado, e da aplicação de um mapeamento inverso a cada um dos pixels assim obtidos para determinar a área que lhes deu origem. Obviamente, como isto define um mapeamento de sistema de coordenadas independente da representação do objeto, este algoritmo é aplicável a outras formas de representação. Por exemplo, desenhos podem ser facilmente transformados mapeando apenas seus vértices de acordo com o mapeamento global descrito pelas malhas.*

Warping por Malhas de Splines em Dois Passos

Este algoritmo foi desenvolvido especificamente para o uso com imagens digitais, explicitamente tomando vantagem da organização usual de armazenamento de imagens

* Mapear apenas os vértices é incorreto se as arestas não são invariantes sob esta mudança de coordenadas em particular, embora os resultados sejam perfeitamente aceitáveis, e até mesmo atenuem as discontinuidades causadas pelo uso de coordenadas baricêntricas.

em computadores. Ao invés de transformar uma imagem diretamente, uma imagem intermediária é criada que é em seguida deformada para alcançar o resultado final. Isto corresponde a uma composição de duas transformações mais simples—uma contendo apenas os deslocamentos horizontais, e a outra, apenas os verticais—que é equivalente à deformação desejada. Deformações puramente horizontais ou verticais são muito mais facilmente implementadas para imagens digitais, especialmente no que se refere a evitar problemas de amostragem e reconstrução [Wolberg 90]. Contudo, este esquema em dois passos introduz um problema, chamado de *bottleneck*, que ocorre quando a imagem intermediária é deformada a tal ponto que não “sobra” informação para o segundo passo.

O algoritmo de warping por malha de splines em dois passos usa como entrada duas malhas regulares de splines, tais como as mostradas na Figura 3-8, restritas a terem splines retilíneas nas bordas da imagem, e splines de mesma direção que não se cruzam. Mais uma vez, a primeira malha define um sistema de coordenadas não deformada, e a outra descreve uma deformação deste sistema.

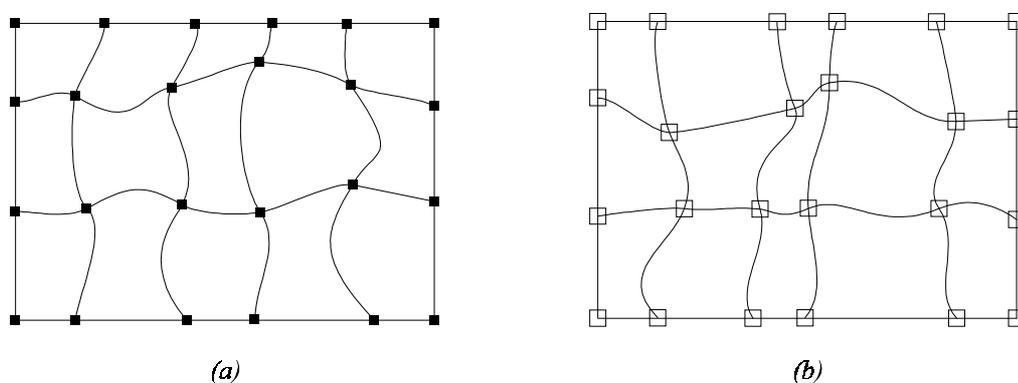


Figura 3-8 Malhas de splines.

Os dois passos são análogos, e podem ser realizados em qualquer ordem. No passo horizontal, cada scanline horizontal é interceptada com as splines verticais nos estados deformado e não deformado, como mostrado na Figura 3-9(a). O estado deformado é determinado neste passo considerando apenas os deslocamentos horizontais. Estas interseções são então interpoladas para obter a função de mapeamento da scanline, como mostrado na Figura 3-9(b), colocando-se as interseções com as splines deformadas em um eixo, e as interseções com as splines não deformadas no outro. Esta função de

mapeamento relaciona todos os pontos da scanline no sistema de coordenadas não deformado ao seu estado deformado. Este processo é repetido para cada scanline para obter uma imagem deformada horizontalmente.

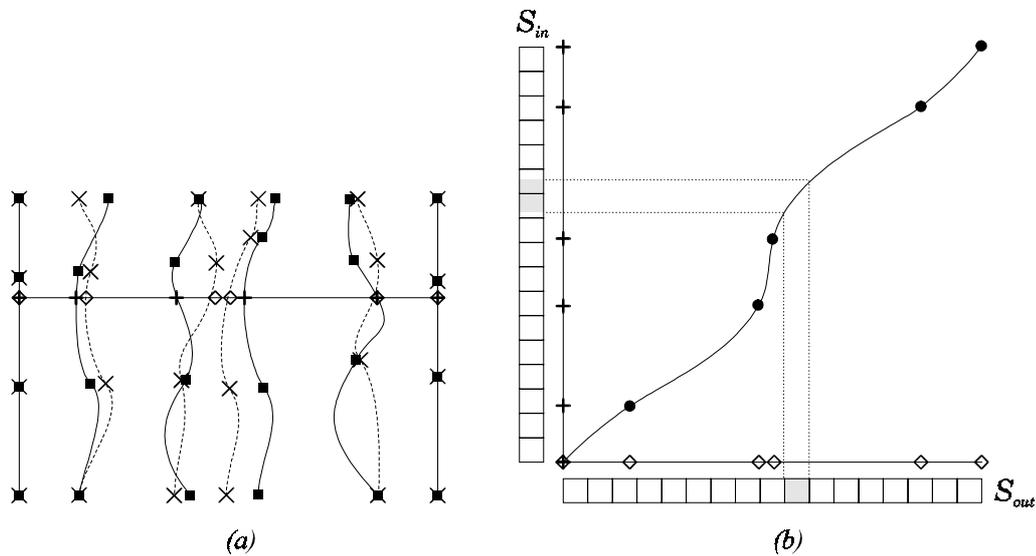


Figura 3-9: Warping de uma scanline: (a) interseção; (b) função de mapeamento.

As malhas de splines podem ser diretamente criadas e modificadas, ou indiretamente obtidas em uma tentativa de reduzir a entrada de dados desnecessária por parte do usuário. Uma possibilidade útil, dada em [Costa et alli 92], é descrever as deformações com conjuntos de segmentos de reta orientados que indicam características importantes, como será feito na próxima técnica. Estes segmentos são usados para deformar uma malha de splines relaxada—que pode ser vista como uma imagem de resolução mais baixa—que é subsequente usada como entrada para o algoritmo de warping por malhas de splines em dois passos.

É fácil ver que, apesar de sua eficiência para imagens digitais, a técnica de warping por malha de splines em dois passos é de uso limitado para outras representações. É possível usar malhas de splines para definir sistemas de coordenadas e deformar desenhos, por exemplo, mas a abordagem em dois passos é inútil nestes casos.

Warping Baseado em Campo

Esta técnica é baseada na utilização de campos de influência ao redor das principais características da imagem a ser deformada. Essas características são marcadas com pares de segmentos de reta orientadas, ou vetores, dos quais um indica o estado original não deformado e o outro, uma deformação daquela característica [Beier, Neely 92]. Cada característica tem seu próprio campo de influência definido como mostrado na Figura 3-10. A coordenada v é a distância perpendicular ao vetor, enquanto u é a distância ao longo do vetor. Note que u é normalizado de acordo com o comprimento do segmento, mas v é uma distância absoluta, de tal forma que esticando o vetor em uma direção, a vizinhança daquela característica é também esticada ao longo daquela direção, mas não perpendicularmente a ela. É possível definir u e v de forma diferente, mas Beier e Neely conduziram experimentos que indicaram a conveniência da definição aqui apresentada em termos de interface com o usuário.

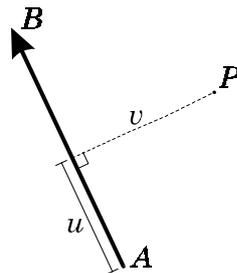


Figura 3-10: Sistema de coordenadas definido por uma característica.

Cada par de características define portanto uma mudança de coordenadas para todo o domínio e essas transformações conflitantes precisam ser combinadas de uma forma suave para produzir uma mudança de coordenadas global. Esta combinação é definida para cada ponto como uma média ponderada das posições deste ponto se mapeado de acordo com cada par de vetores, como mostrado na Figura 3-11. O ponto P é mapeado por $F_1 \rightarrow F_1'$ para P_1' e por $F_2 \rightarrow F_2'$ para P_2' ; o ponto mapeado final P' é uma média ponderada entre P_1' e P_2' .

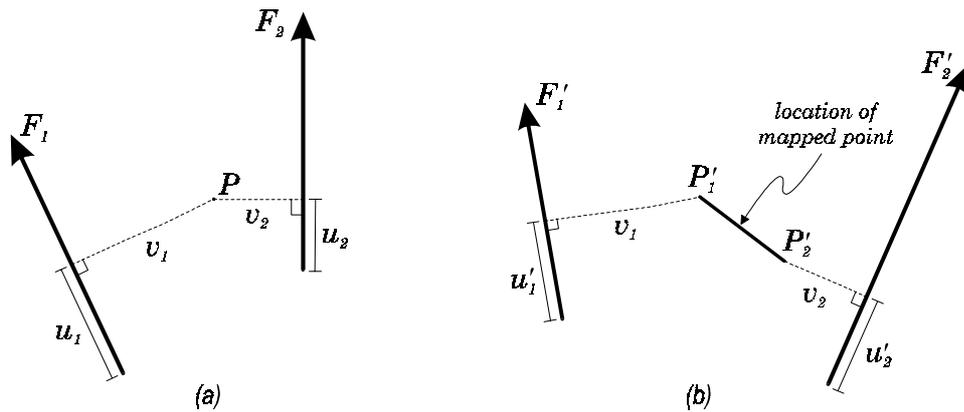


Figura 3-11: Combinação de múltiplos sistemas de coordenadas: (a) original; (b) deformado.

O peso de cada característica é inversamente proporcional à distância do ponto a essa característica, sendo definido como

$$weight = \left(\frac{l^p}{a + d} \right)^b.$$

Na equação, l é o comprimento de um vetor, d é a distância do ponto ao vetor, e a , b e p são constantes usadas para ajustar a média. A constante a pode ser vista como a aderência ao vetor: para valores próximos a zero, pontos próximos ao vetor serão mapeados exatamente como o vetor determina; para valores maiores, pontos estão mais soltos, e livres para se movimentar, mesmo que estejam sobre a característica. A importância do comprimento do vetor é controlada por p : se ele é zero, o comprimento é ignorado; caso contrário, vetores mais longos são mais importantes. A interpretação de b pode ser vista como a concentração de força de uma característica: valores grandes tornam a força do vetor grande próxima ao vetor e pequena quando distante, com decaimento bastante rápido; pequenos valores tornam esse decaimento mais lento, reduzindo localidade da característica. Note que diferentes valores de constantes podem ser associados a cada característica, apesar de talvez ser mais conveniente usar um valor global para b .

Esta definição de mapeamento de coordenadas, baseado em campos de influência ao redor das principais características da imagem, não depende de forma alguma da estrutura ou das particularidades de imagens digitais. É simplesmente uma maneira inteligente e parametrizada de combinação de diferentes transformações, resultando em

uma mudança de coordenadas global que pode ser facilmente usada para transformar objetos em qualquer representação.

4 Morphing em \mathbb{R}^3

Transformações entre sólidos tridimensionais são um importante objetivo de modelagem geométrica, e várias tentativas foram feitas para se executar tais mutações. O trabalho anterior se concentrou na transformação direta da forma de um sólido na forma do outro, investigando um número de possibilidades e produzindo uma larga faixa de resultados. A falta de interação com a pesquisa em morphing 2D, contudo, obscureceu as vantagens de se executar um passo de warping a priori, que permitiria um grau muito maior de controle por parte do usuário no casamento das características e simplificaria os requerimentos dos algoritmos de blending. Este passo de warping foi mostrado no capítulo 2 ser um parte integral da metamorfose de objetos, tanto em 2D quanto em 3D.

Neste capítulo, irá se mostrar que soluções anteriores para o problema de metamorfose tridimensional são apenas transformações de contra-domínio, ao invés de soluções completas envolvendo warping e blending. Algumas dessas soluções serão criticamente vistas como possíveis alternativas às técnicas bidimensionais mostradas no capítulo anterior, e técnicas 2D existentes serão analisadas tendo em vista possíveis extensões para 3D. Esta abordagem integrada irá resultar na introdução de várias novas técnicas para warping e blending de objetos 3D, apresentados nas seções 4.3 e 4.4.

4.1 Modelos 3D e suas Representações

A analogia entre modelos 2D e 3D e suas representações é fundamental para o compartilhamento de técnicas de morphing e algoritmos entre os casos bi- e tridimensionais. Na seção 3.1, esta similaridade natural entre as representações 2D e 3D foi delineada, mostrando que diferentes representações 2D eram sub-casos do modelo conceitual de objeto descrito na seção 2.1. Analogamente, representações práticas 3D serão mostradas ser consistentes com este mesmo modelo de objeto.

Na seção 2.1, um objeto n -dimensional foi modelado como uma função associando a cada ponto de um subconjunto do espaço Euclidiano \mathbb{R}^n , uma tupla em um espaço de

propriedades descrevendo quaisquer características relevantes do objeto. Para objetos tridimensionais, esta definição será então

$$f: U \subseteq R^3 \rightarrow R^{r_1} \times R^{r_2} \times \dots \times R^{r_k}, \quad r_1 + r_2 + \dots + r_k = m$$

$$f = (f^1, f^2, \dots, f^k)$$

onde as k propriedades do objeto são definidas como funções

$$f^j: U \subseteq R^3 \rightarrow R^{r_j}.$$

Mais uma vez, a geometria dos objetos é dada por uma propriedade especial que é uma função de classificação de pertinência de pontos, indicando para cada ponto no domínio de definição U do objeto se ele pertence ao objeto ou não.

Em geral, o estudo de representações de objetos 3D confere maior importância à representação da forma que à qualquer outra propriedade. Há, contudo, várias formas de representar tais propriedades associadas com os modelos geométricos, sob a mesma estruturação. Os largamente utilizados modelos B-Rep [Mäntylä 88] são apropriados a terem propriedades associadas a vértices, arestas ou faces; representar propriedades que variem ao longo destes itens ou no interior do objeto, contudo, é mais complexo. Similarmente, um modelo volumétrico pode facilmente ter uma propriedade associada a cada voxel, e um modelo CSG pode ter propriedades como uma parte de, ou associada a, cada primitiva. Objetos implícitos têm propriedades naturalmente associadas a cada ponto no espaço, dadas por funções da mesma forma que no modelo conceitual de objeto.

As formas de representação de objetos 3D são basicamente agrupadas em duas categorias, que foram descritas no capítulo anterior e aplicadas a objetos 2D:

- Fronteira: paramétrica ou implícita;
- Volume: paramétrico ou implícito.

Distintamente da situação 2D, todas as formas de representação, incluindo objetos implícitos, são largamente utilizadas na prática para a modelagem 3D, com diversas aplicações.

4.2 Transformações 3D de Domínio e Contra-domínio

Ao contrário da pesquisa em morphing 2D, o trabalho anterior em morphing 3D não distingue claramente entre transformações de domínio e contra-domínio. A absoluta maioria das técnicas de morphing 3D são na verdade abordagens de combinação de formas 3D, i.e., transformações de contra-domínio da propriedade forma, algumas delas com resultados valiosos que podem ser grandemente melhorados se usados em conjunção com uma transformação de domínio. Deste modo, as próximas seções irão se concentrar em um breve estudo das técnicas 2D e 3D existentes que tentam resolver as fases de warping e blending do problema de morphing 3D:

- Técnicas de deformação 3D existentes aplicáveis a morphing;
- Extensões possíveis de técnicas de warping 2D;
- Técnicas de blending de formas 3D existentes, que incluem os trabalhos anteriores em morphing 3D;
- Extensões possíveis de técnicas de blending de formas 2D.

A forma de representação do objeto tem um papel fundamental na computação de transformações de domínio e contra-domínio. Além de uma associação natural entre algumas técnicas e certas representações, há uma diferença inerente de dificuldade em computar transformações de domínio e contra-domínio para cada representação. Para objetos representados por voxels com densidade, por exemplo, um blending de forma é tão simples quanto interpolar as densidades de voxels correspondentes, analogamente a um cross-dissolve entre duas imagens digitais; a computação de uma transformação de domínio, entretanto, envolve problemas de amostragem e reconstrução, necessitando de técnicas adicionais para evitar tais efeitos. Para objetos B-Rep, a fase de warping envolve apenas o estabelecimento usual de um mapeamento de domínio e a computação é alcançada simplesmente através da transformação dos vértices do objeto; o blending de objetos, usualmente com propriedades combinatórias incompatíveis, envolve complicadas técnicas de compatibilização de estruturas. Objetos implícitos, por outro lado, dada a sua proximidade com a teoria, são relativamente simples de se lidar, tanto em termos de transformações de domínio quanto de contra-domínio, apesar da sua visualização ser certamente mais cara computacionalmente.

É possível se beneficiar das dificuldades acima descritas através do emprego de diferentes representações do mesmo objeto para cada fase. A conversão entre as representações é algumas vezes bastante complicada, apesar de algumas das conversões serem diretas em algumas direções [Gomes, Velho 92]. A implementação descrita no capítulo 5 faz uso deste fato para permitir uma visualização rápida dos resultados com uma representação, e uma visualização de alta qualidade com outra. O uso de modelos híbridos [Alvaro 86] é também uma alternativa válida.

4.3 Técnicas de Warping 3D

Das técnicas discutidas nesta seção, apenas a primeira foi desenvolvida especificamente para objetos 3D. As outras são técnicas de warping 2D existentes, para as quais uma análise da viabilidade de extensão para 3D é feita aqui.

Técnicas de warping usadas para morphing compartilham um requerimento comum: é preciso permitir ao usuário especificar correspondências entre características relevantes dos dois objetos. Uma deformação para morphing precisa ser tal que as características de cada objeto sejam deformadas para se alinharem com as características associadas. Nem todas as técnicas de deformação genéricas são apropriadas a este tipo de aplicação, já que algumas podem não ter uma maneira intrínseca de associar localmente os dois objetos, como a técnica descrita em [Barr 84].

Deformação por Formas Livres

Nesta técnica, proposta em [Sederberg, Parry 86], objetos são envolvidos por um paralelepípedo que é repartido por um reticulado regular de pontos de controle. Alterando a posição destes pontos de controle, um sistema de coordenados deformado é definido, como mostrado na Figura 4-1.

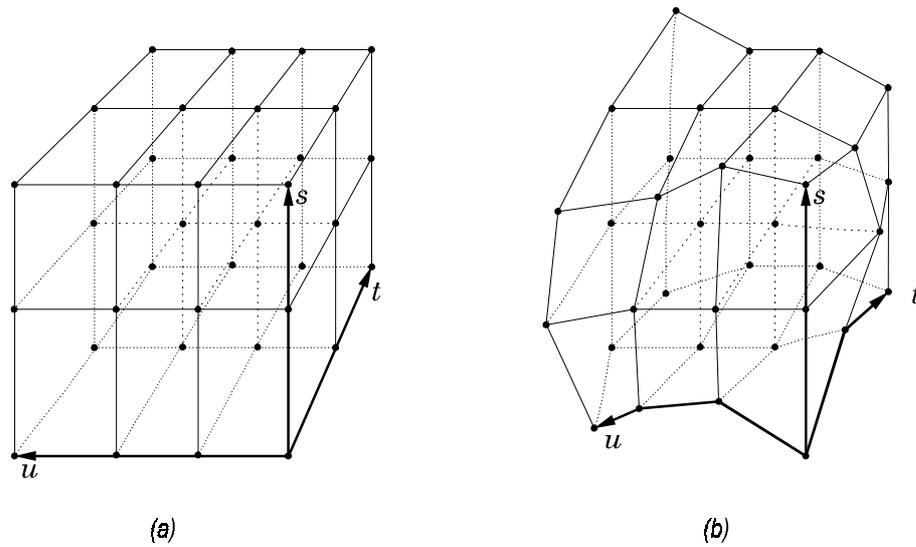


Figura 4-1: Sistema de coordenadas deformado por formas livres: (a) relaxado; (b) deformado.

O mapeamento do sistema relaxado para o sistema deformado é relativamente simples: cada ponto \mathbf{X} é decomposto em três coordenadas (s, t, u) através da projeção sobre os três eixos do paralelepípedo, e o ponto deformado \mathbf{X}' é obtido pela transformação de (s, t, u) por um polinômio de Bernstein onde os coeficientes são os pontos de controle deformados. Isto efetivamente define um mapeamento $W: R^3 \rightarrow R^3$, que é um warp do espaço Euclidiano tridimensional, como descrito no capítulo 2.

Para aplicações de morphing, entretanto, não haveria um sistema relaxado, mas dois sistemas seguindo as características dos objetos em questão, como mostrado no exemplo na Figura 4-2(a). O mapeamento precisa ser feito de um sistema de coordenadas deformada para outro sistema deformado, e isso pode ser feito com um passo em um sistema relaxado, mostrado na Figura 4-2(b). A transformação polinomial do sistema relaxado T para o sistema final F é a mesma descrita acima, mas o mapeamento do sistema inicial I para o sistema intermediário T requer a inversão do polinômio de Bernstein. Esta inversão envolve a busca de raízes de um polinômio trivariado de alto grau, que aumenta com o número de pontos de controle no reticulado.

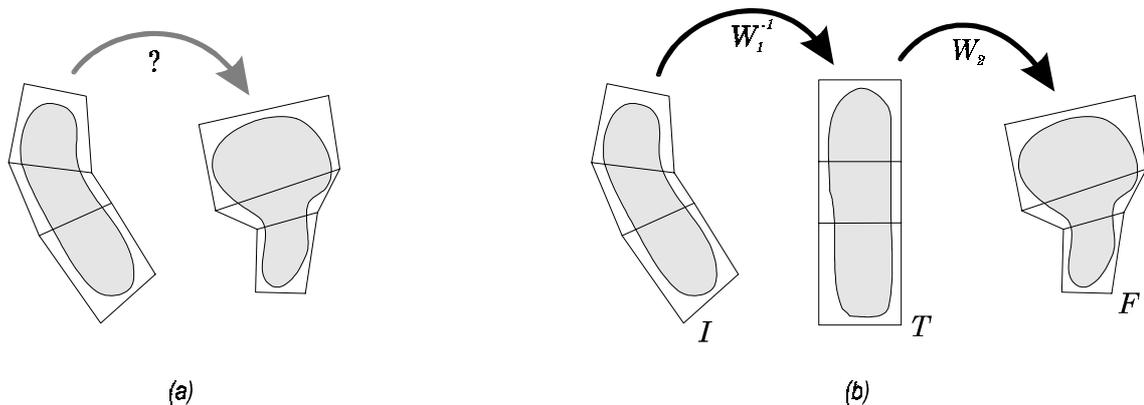


Figura 4-2: Morphing por formas livres: (a) direto; (b) com um passo intermediário.

Como esta técnica define de fato uma deformação do espaço ambiente, ela pode ser facilmente aplicada a diversas representações de objetos. Além disso, as modificações necessárias para aplicar esta técnica em 2D são mínimas: um polinômio de Bernstein bivariado seria usado ao invés, com pontos de controle bidimensionais. É claro que esta técnica de deformação é extremamente similar à deformação por malha de splines descrita no capítulo anterior, mas usando curvas de Bézier—outras bases polinomiais geram resultados correspondentes. A principal diferença é que o algoritmo de warping por malha de splines em dois passos foi descrito especificamente para imagens digitais, enquanto esta técnica simplesmente estabelece uma deformação do domínio dos objetos, sem nenhuma dependência em nenhuma forma particular de representação de objeto.

Warping por Malha de Triângulos

Esta técnica 2D pode ser naturalmente estendida para 3D: uma malha de tetraedros é dada em dois estados distintos, um definindo um sistema de coordenadas e o outro uma deformação deste. Para tetraedros correspondentes, pontos interiores podem ser facilmente mapeados usando coordenadas baricêntricas. Desta forma, este algoritmo é apropriado para qualquer representação de objeto. Em particular, para objetos volumétricos amostrados, uma rasterização 3D do interior dos tetraedros é necessária.

Descrever a deformação diretamente com malhas de tetraedros não é viável a menos que sejam obtidas algorítmicamente de alguma forma. Exatamente como no caso 2D, uma triangulação automática de pontos usados para indicar características é uma alternativa factível e fácil de usar, e uma triangulação automática pode ser usada para

gerar as malhas. Uma triangulação regular, tal como uma triangulação de Delaunay 3D, irá minimizar as discontinuidades em arestas e faces introduzidas pelo mapeamento linear dentro de cada tetraedro.

Warping Baseado em Campo

A extensão da técnica de warping 2D baseada em campo para 3D é relativamente direta. Mais uma vez, características importantes dos objetos são marcadas com segmentos de reta orientados, e uma posição diferente de cada segmento define uma deformação da característica. Em 3D, há um grau adicional de liberdade que resulta em um segmento adicional no sistema de coordenadas definido pela característica, como mostrado na Figura 4-3. Os dois vetores mostrados em negrito, especificados pelo usuário e perpendiculares, formam uma abstração de um *bone* (osso). Um conjunto de tais bones, que marcam todas as características relevantes em um objeto, é chamado de *skeleton* (esqueleto). Para obter um sistema de coordenadas Cartesiano, um eixo adicional é obtido do produto vetorial entre os dois vetores em um bone. Cada ponto pode então ser projetado em cada um desses eixos para determinar suas coordenadas nesse sistema. A coordenada u é normalizada de acordo com o comprimento do segmento, enquanto v e t são distâncias absolutas.

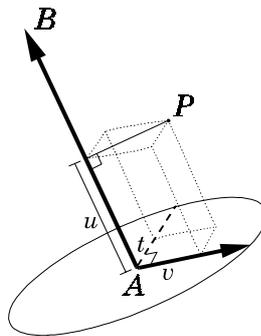


Figura 4-3: Sistema de coordenadas definidos por uma característica 3D.

O sistema de coordenadas global definido por dois esqueletos é obtido a partir de uma combinação das mudanças de coordenadas individuais definidas por cada par de ossos. A combinação é uma média ponderada dos mapeamentos de cada ponto de acordo com cada par de ossos, exatamente como o esquema usado no caso 2D. O peso de cada osso é computado como

$$weight = \left(\frac{I^p}{a + d} \right)^b$$

A interpretação das constantes a , b e p é idêntica àquela do caso 2D. A técnica resultante é expressiva e dá um controle fácil ao usuário sobre a transformação, através da manipulação das características e do ajuste das constantes. Como esta técnica simplesmente define um mapeamento de coordenadas, ela é independente da representação do objeto. Para objetos B-Rep e volumétricos implícitos, por exemplo, sua aplicação é fácil e direta. Para objetos volumétricos discretos, por outro lado, além da reamostragem necessária, esta técnica é computacionalmente cara, já que precisa ser aplicada pelo menos a cada voxel. Uma implementação desta técnica é discutida no capítulo 5.

Warping por Malhas de Splines em Dois Passos

O algoritmo original 2D é aplicável a imagens digitais; a extensão proposta aqui é correspondentemente aplicável a modelos volumétricos baseados em voxels. Ela é útil não apenas para morphing de dados volumétricos, mas também como uma ferramenta de modelagem de objetos volumétricos através de deformação, de acordo com o paradigma de modelagem do “bloco de argila” mencionado em [Sederberg, Parry 86]. A maior vantagem de modelos volumétricos é que não há um tratamento especial dado a objetos com topologias diferentes, e portanto, a topologia de um objeto pode facilmente mudar durante uma transformação.

O algoritmo 3D, warping por malhas de superfícies splines em três passos, efetua três passos sobre os dados volumétricos, cada um deles computando a transformação em apenas uma direção. A composição destes passos resulta em uma transformação completa do objeto, como mostrado no *pipeline* na Figura 4-4. Isto efetivamente reduz o problema de warping de um sólido 3D para vários problemas de warping unidimensional de filas de voxels, ou *scanlines*. Os passos do algoritmo são efetuados seqüencialmente, mas a deformação de cada scanline em cada passo é independente das outras scanlines, e portanto é apropriada para paralelização, como no algoritmo 2D.

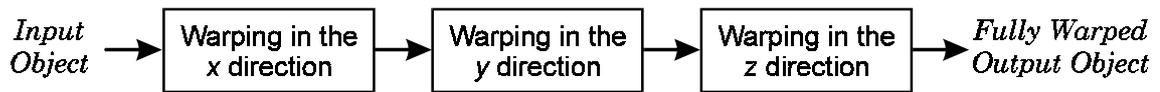


Figura 4-4: Pipeline de warping em três passos.

Cada passo do algoritmo realiza uma deformação na direção de um dos eixos Cartesianos x , y ou z . Uma malha é composta de um grid de pontos de controle C_{ijk} , com m pontos na direção x , n na direção y e p na direção z , totalizando $m.n.p$ pontos de controle, de tal forma que cada ponto de controle pode ser facilmente referido pelos seus índices inteiros. O passo para o eixo x é realizado como se segue, e os outros são análogos:

- Uma malha intermediária I é criada considerando apenas os deslocamentos dos pontos de controle na direção x da malha original S para a malha deformada D . Desta forma, os pontos de controle de I terão as mesmas coordenadas y e z dos pontos correspondentes em S e a coordenada x do ponto correspondente em D .
- Uma superfície spline S_i é passada por cada conjunto de pontos de controle de S com um índice i constante, resultando em superfícies que são “perpendiculares” na direção x . Um processo idêntico é realizado para obter a superfície de splines I_i passando pelos pontos de controle de I .
- Cada fila de voxels na direção x é interceptada com superfícies S_i e I_i , resultando em dois conjuntos de interseções colineares.
- Para cada fila de voxels, as interseções com S_i e I_i são interpoladas com uma curva spline, resultando em um função de mapeamento da scanline que é usada para mapear voxels do objeto origem para suas posições deformadas na direção x .

A minimização dos problemas de amostragem e reconstrução é agora um problema unidimensional mais simples, que é realizado scanline por scanline. Esta simplificação ajuda a tornar este algoritmo um procedimento extremamente eficiente para deformar dados volumétricos com resultados de boa qualidade.

No caso 2D, a especificação do warping usando tais técnicas baseadas em partições era um trabalho difícil, que se torna pior em três dimensões. O uso de uma especificação baseada em características tal como segmentos de reta orientados, é uma alternativa muito melhor, semelhante à técnica de warping por malhas de splines controladas por campo descrita em [Costa et alli 92]. Tem a vantagem de oferecer uma interface fácil de usar aliada a uma implementação para dados volumétricos eficiente, apropriada para paralelização.

4.4 Técnicas de Blending 3D

Esta seção apresenta duas abordagens que eram consideradas morphing, mas são, de fato, técnicas de combinação de formas. Além disso, o blending de sólidos, uma importante ferramenta da modelagem geométrica é brevemente discutido, por suas aplicações como uma transformação de contra-domínio de alta qualidade para a metamorfose de objetos volumétricos implícitos. Finalmente, uma idéia para extensão de blending de formas 2D existente para 3D é investigada.

Blending de Objetos Volumétricos Implícitos

As operações de blending são muito apropriados para objetos definidos implicitamente, produzindo um terceiro objeto implícito que é uma transição suave entre as duas superfícies interceptantes. O estudo das funções de blending usadas para combinar as funções características descrevendo os objetos é uma área de pesquisa relevante [Rockwood, Owen 87]. Uma função de blending é uma função $B(f_1, f_2, \dots, f_n)$ que toma duas ou mais funções características como parâmetros e é por sua vez descrita implicitamente.

Para aplicação em morphing, duas formas de blending são especialmente úteis: *exponencial* [Blinn 82] e *super-elíptico* [Rockwood 89]. O blending exponencial produz resultados excelentes quando as funções características são algébricas e é descrito, para duas entradas, por uma família de funções a um parâmetro

$$B_t(x, y) = -\log[(1-t)e^{-x} + te^{-y}].$$

O objeto combinado é descrito por uma função característica que é obtida simplesmente por uma composição de B com f e g , i.e., $B(f, g)$.

Similarmente, o blending super-elíptico, para dois objetos implícitos f e g , é dado pela função

$$B(f, g) = 1 - \left([1 - \alpha_1 f]_*^\beta + [1 - \alpha_2 g]_*^\beta \right)^{\frac{1}{\beta}}, \text{ where } [\cdot]_* = \max(0, \cdot).$$

Os parâmetros α e β são usados para alterar a forma do blending: α_i leva o blending em direção à primitiva i , e β controla a rigidez geral do blending.

Blending de Volumes por Fourier Escalonado

Dois objetos implícitos, descritos por funções de classificação de pertinência de pontos, podem ser facilmente combinados em um novo objeto implícito dado por uma média das duas funções de entrada. Tal combinação simples é uma forma de blending linear, com resultados que são geralmente inaceitáveis para a combinação de sólidos de boa qualidade.

Uma alternativa interessante é o blending de objetos no domínio da frequência ao invés do domínio espacial [Hughes 92]. Isto é facilmente realizado através da aplicação de uma transformada de Fourier aos objetos, seguida da combinação de suas versões transformadas com um blending linear, e então trazendo-os de volta ao domínio espacial com uma transformada de Fourier inversa.

Uma vez no domínio da frequência é possível extrair mais facilmente outras formas de informação da forma dos objetos. Componentes de alta frequência são responsáveis por pequenos detalhes na superfície do objeto, enquanto baixas frequências descrevem a forma geral do objeto. Desta forma, uma transformação escalonada, onde as diferentes faixas de frequência são transformadas a taxas diferentes, resultaria em uma transformação melhor: as altas frequências do objeto origem são gradualmente removidas, as baixas frequências são interpoladas, e finalmente as altas frequências do objeto destino são introduzidas.

Note que esta técnica pode ser facilmente usada com dados volumétricos amostrados com uma transformada de Fourier rápida (FFT), e seu uso com objetos definidos implicitamente também é possível. Entretanto, existe um controle

extremamente restrito sobre a transformação na forma de uma especificação de escalonamento, que não é suficiente para se obter resultados de qualidade apropriada.

Combinação de Estruturas Combinatórias

Esta técnica, desenvolvida para o uso com objetos B-Rep, é baseada na compatibilização das estruturas de objetos combinatorialmente diferentes [Kent et alli 92]. A idéia é alterar os dois objetos resultando em dois novos objetos que têm exatamente o mesmo número e adjacência de vértices, arestas e faces. Uma vez que essa correspondência tenha sido estabelecida, a interpolação entre os dois novos objetos é direta, necessitando apenas da interpolação geométrica entre pares de vértices correspondentes. Note que os dois objetos de entrada precisam ser topologicamente equivalentes e de *genus* 0, i.e., sem passagens através dos objetos.

O algoritmo de correspondência consiste em projetar as estruturas de ambos os modelos sobre a mesma esfera. Em seguida, as duas estruturas são combinadas através de cerceamento das faces projetadas de um modelo pelas faces projetadas do outro. A estrutura combinada é então mapeada sobre a superfície de ambos os modelos originais. Isto gera dois novos modelos, que têm exatamente a mesma forma que os dois modelos originais, mas que compartilham uma estrutura combinatória comum. Estes novos objetos usualmente terão diversas faces coplanares. Note que a forma como a projeção é feita é fundamental para o sucesso desta técnica e pode influenciar significativamente os resultados.

Esta técnica sozinha, como descrita em [Kent et alli 92], simplesmente tenta fazer as estruturas compatíveis, com nenhuma intenção de mapear partes de um objeto em partes similares dos outros. Isto é útil para objetos que não têm partes similares, ou se as partes similares estão geometricamente alinhadas a priori por uma fase de warping. De fato, a combinação desta técnica de blending com uma fase de warping resulta em uma técnica viável e eficiente para morphing de objetos B-Rep.

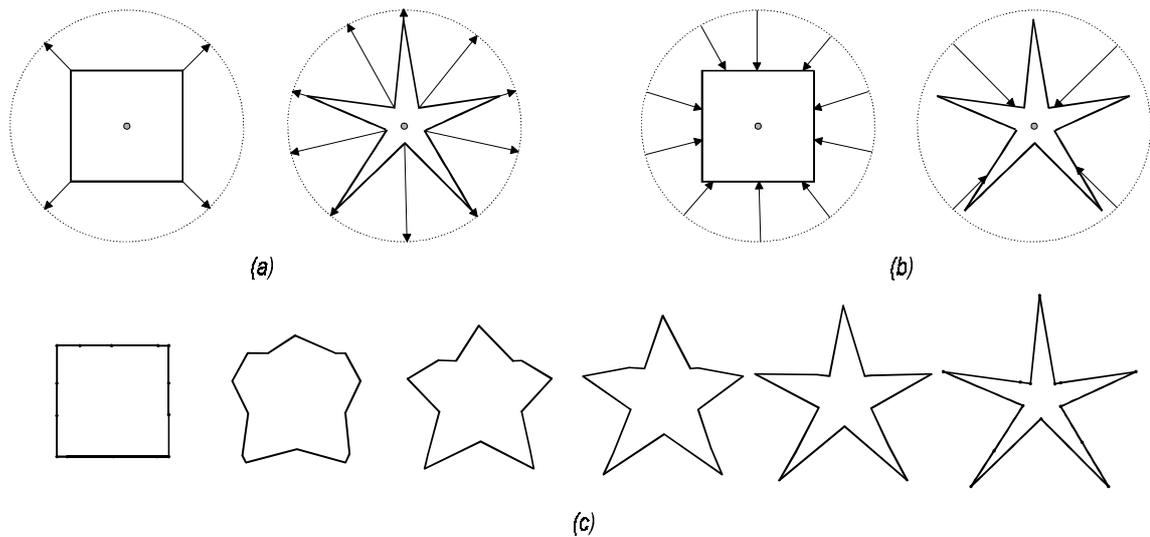


Figura 4-6: Uma combinação de duas estruturas combinatorias: (a) projeção; (b) combinação; (c) interpolação.

Esta técnica pode ser naturalmente estendida para modelos de fronteira aproximados 2D, ou desenhos. As formas poligonais 2D são projetadas sobre um círculo, as duas estruturas são combinadas e subsequentemente mapeadas de volta às formas originais. Isto resulta em dois objetos com estruturas equivalentes, mas mantendo a geometria dos originais. Um exemplo 2D, que também é útil na compreensão do caso 3D, é mostrado na Figura 4-6. Esta técnica é comparada com outras técnicas de combinação de formas na próxima seção.

Blending Físico

A extensão desta técnica 2D, descrita no capítulo anterior, para 3D é certamente não trivial. Um modelo de arame 3D pode ser usado, mas a computação do trabalho necessário para a transformação daquele modelo não é direta. Embora a medida do trabalho necessário para esticar uma aresta seja similar àquele de esticar em 2D, o trabalho de flexão em 3D é muito mais complexo.

Além disso, o problema de correspondência de vértices em 2D era limitado a estabelecer uma única associação e os outros seguiam-se seqüencialmente. Em 3D, tal associação simples é completamente impossível, e a compatibilização das estruturas combinatorias é um problema de inserção não somente de vértices e arestas, mas também de faces. O problema de otimização é muito mais complexo do que em 2D, já que

o problema 3D não pode ser restrito a simplesmente adicionar vértices na posição de vértices existentes.

Esta extensão é um problema aberto com resultados promissores. A comparação de blending físico com outras técnicas de blending de formas 2D mostra a efetividade potencial desta abordagem, se estendida para 3D.

A combinação de estruturas combinatórias 2D é de fato uma técnica com essencialmente a mesma abordagem que o blending físico 2D: o objetivo é escolher onde novos vértices devem ser adicionados a cada objeto, de tal forma que as estruturas sejam equivalentes, e uma interpolação linear produza bons resultados. Obviamente, o blending físico é um método muito mais sofisticado, com excelentes resultados que incluem o casamento de características correspondentes se as suas geometrias forem relativamente similares.

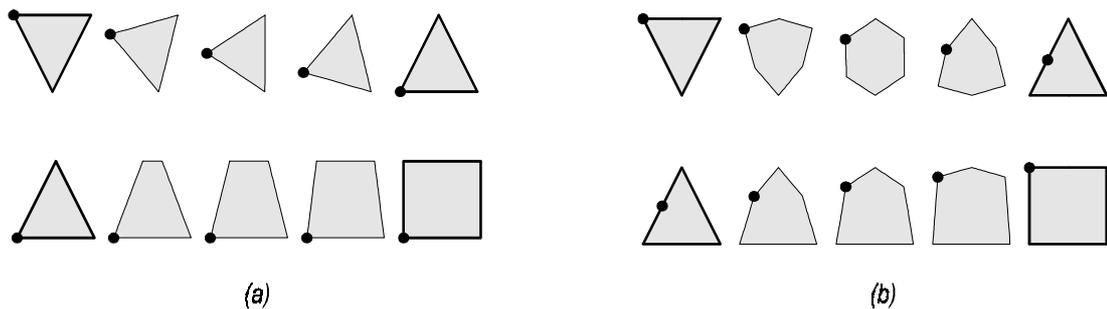


Figura 4-7: Comparação de blendings de formas 2D: (a) físico; (b) combinação de estruturas combinatórias.

A técnica de combinação 2D descrita acima, por outro lado, não consegue casar as características, produzindo bons resultados apenas se os objetos de entrada não têm uma correspondência de características natural, como os exemplos dados em [Kent et alli 92] e na Figura 4-7. Para aplicações de morphing, o casamento de características é um objetivo perceptual essencial, tornando o blending físico uma alternativa muito mais vantajosa, embora um passo de warping minimize a importância do casamento de características no blending.

5 Implementação

Este capítulo irá discutir uma implementação de um sistema de morphing 3D experimental baseado em técnicas descritas em capítulos anteriores. Isto inclui um modelo de objeto apropriado, uma extensão de uma técnica de warping 2D e blending de objetos implícitos. As suposições e decisões de implementação serão justificadas, e tópicos de interface com o usuário mencionados.

5.1 Decisões de Implementação

Representação

A primeira decisão na implementação de um sistema de morphing é a escolha de uma representação para os objetos. Há algumas escolhas convencionais: objetos B-Rep, objetos volumétricos amostrados e objetos implícitos exatos.

Objetos B-Rep seriam uma escolha natural se as ferramentas e sistemas de modelagem existentes fossem ser utilizados. Além disso, existe uma ampla disponibilidade de objetos B-Rep interessantes e prontos para o uso, que permitiriam a realização de testes mais realistas mais rapidamente, evitando a necessidade de construir os modelos. Um outro fator importante é que a visualização de objetos B-Rep é tratada diretamente por vários bons pacotes de visualização, com diferentes graus de velocidade/qualidade na síntese das imagens. Por outro lado, alterar a topologia de um objeto B-Rep durante uma transformação de morphing é uma tarefa trabalhosa, e a representação de propriedades genéricas que variem ao longo do objeto não é suportada por muitos sistemas, complicando o processo de blending das propriedades.

Objetos implícitos são mais fáceis de usar em transformações de morphing, por sua proximidade da conceituação. Transformações de domínio e contra-domínio são mais facilmente e naturalmente aplicadas a tais modelos, e como objetos de topologias diferentes são descritos da mesma forma, alterações na topologia durante a transformação são transparentes. A modelagem com objetos implícitos é menos usual, no

entanto, e assim, a disponibilidade de modelos e ferramentas de modelagem é limitada. Deve-se lembrar que objetos implícitos são uma forma geral de representação, e que é possível obter objetos implícitos a partir de um conjunto de amostras pontuais que pode advir de medições do mundo real (scanners 3D, por exemplo), vértices B-Rep, ou amostras de superfícies paramétricas, entre outros [Sclaroff, Pentland 91].

Para os propósitos deste trabalho, a possibilidade de experimentar diretamente com a teoria era altamente atrativa, e não havia necessidade dos modelos mais realistas disponíveis na forma B-Rep. Desta forma, os objetos implícitos usados nos exemplos são simples mas significativos, e a implementação pode ser feita de uma forma elegante e frugal.

O uso de objetos implícitos amostrados é um tanto mais difícil durante a fase de warping devido aos problemas de amostragem e reconstrução mencionados anteriormente. Entretanto, a simples ignorância destes problemas provê resultados aceitáveis que podem ser usados para visualizar rapidamente uma transformação. Uma vantagem é que há uma quantidade considerável de objetos volumétricos amostrados disponível, e objetos implícitos exatos podem ser trivialmente convertidos para esta representação.

Técnicas de Warping e Blending

Das técnicas de warping 3D no capítulo 4, a abordagem baseada em campo parece a mais expressiva devido à metáfora do esqueleto, uma idéia que é facilmente entendida e largamente utilizada em sistemas de animação para objetos hierárquicos. A técnica originalmente 2D tem uma extensão natural para 3D, que pode ser facilmente implementada. Além disso, pode ser usada com qualquer forma de representação.

Há uma forte relação entre a técnica de blending e o esquema de representação utilizado. Para objetos implícitos, é natural usar uma das muitas técnicas de blending de superfícies implícitas disponíveis, algumas das quais geram excelentes resultados para os propósitos de combinação de formas. Em particular, o sistema implementado utiliza atualmente funções algébricas para descrever os objetos e, deste modo, um dos melhores resultados é obtido com o blending exponencial [Blinn 82]. É trivial suportar funções genéricas com blending super-elíptico [Rockwood 89], por exemplo.

Estas decisões de implementação resultaram em uma implementação que é muito próxima da conceituação—possibilitando uma fácil experimentação—e é facilmente inteligível. Isto é expresso, por exemplo, no extrato de código abaixo, responsável pelo morphing de dois objetos implícitos:

```

/*
 * Computes the value of the characteristic function at <point>
 * of an implicit object, that is the result of a morphing of
 * two objects, each of which described by an implicit function.
 */

/* field-based warp */
bonemap (skel, GEO_INTERMED, GEO_INITIAL, &point, &warped1);
bonemap (skel, GEO_INTERMED, GEO_FINAL, &point, &warped2);

f1x = (*implicitfunction1) (&warped1);
f2x = (*implicitfunction2) (&warped2);

/* exponential blending */
fx = -log (t * exp (-f2x) + (1 - t) * exp (-f1x));

```

5.2 Visualização

Qualquer sistema de visualização capaz de sintetizar objetos implícitos pode ser usado para visualizar os resultados de um morphing entre dois objetos implícitos, já que o resultado é um objeto implícito.

Ray tracing, acoplado a um simples algoritmo de procura de raízes, é particularmente apropriado para visualização de alta qualidade dos resultados. É possível suportar não apenas o morphing da forma, mas também o blending de outras propriedades, tais como texturas de cor e parâmetros de iluminação. Não é uma técnica eficiente, mas os resultados compensam o tempo dispendido. Um sistema para visualização de morphings completos, incluindo a forma e outras propriedades, foi implementado baseado em um pacote de ray tracing anteriormente desenvolvido, descrito em [Costa, Darsa 91]. Objetos metamorfoseados foram implementados como uma nova classe de objetos do sistema, que pode ter quaisquer duas outras instâncias de objetos suportadas pelo pacote como filhos, incluindo superfícies implícitas customizadas.

Para evitar maiores atrasos no ciclo de design de uma metamorfose, um módulo de visualização rápida foi usado para visualizar grosseiramente a forma resultantes dos objetos deformados e até mesmo metamorfoseados. Isto é feito através de amostragem regular do domínio do objeto de saída: para cada amostra, o valor da função característica do objeto metamorfoseado é computado para determinar se a amostra pertence ou não ao objeto. Isto é feito exatamente como mostrado no segmento de código que foi mostrado acima.

5.3 Tópicos de Interface com o Usuário

A especificação do morphing é ainda um processo manual e, portanto, uma interface com o usuário fácil de usar é indispensável. O sistema de morphing implementado inclui algum grau de interação com o usuário na forma de um editor de *skeletons*, que é de fato, uma extensão de um editor de curvas genéricas anteriormente desenvolvido, feita através da adição de uma nova classe de curvas, *bones*. Este editor foi desenvolvido usando o pacote IUP [Levy 93], tendo necessitado do desenvolvimento de um driver IUP que integrasse Motif e a biblioteca gráfica 3D GL. O desenho e a interação 3D são tratados pela GL, enquanto os diálogos são tratados pelo IUP.

O editor permite que até cinco vistas sincronizadas sejam utilizadas sincronizadamente, e permite a edição do *skeleton* superposto a uma visão da forma dos objetos implícitos. Isto é feito através da leitura de um conjunto de amostras do objeto implícito desejado, que é considerado uma curva não editável.

6 Conclusão e Trabalho Futuro

Uma conceituação do problema de morphing para objetos genéricos definidos em um espaço Euclidiano n -dimensional foi apresentada no capítulo 2. Foi mostrado que uma transformação de morphing completa compreende não apenas uma mudança de coordenadas no domínio de definição dos objetos, mas também uma combinação dos contra-domínios dos objetos—suas propriedades. Esta visão teórica engloba uma série de características importantes que implementações de morphing se utilizam, incluindo transformações parametrizadas e locais.

A conceituação considerou objetos como entidades n -dimensionais, que podiam ser facilmente reduzidas a duas e três dimensões, permitindo que a pesquisa anterior em morphing fosse vista de uma forma integrada.

No capítulo 3, o trabalho anterior em morphing de imagens e desenhos estruturados foi discutido e mostrado ser compatível com a conceituação. Uma discussão similar foi apresentada no capítulo 4, para os esforços anteriores em morphing de objetos 3D. Em ambos os casos, uma particularização dos objetos genéricos n -dimensionais para as formas usuais de representação de objetos tornou possível a ligação da teoria com as técnicas práticas.

A abordagem integrada talvez possa ser considerada a mais relevante contribuição dessa dissertação, tendo possibilitado uma expansão da variedade de técnicas disponíveis para cada caso. No capítulo 4, diversas extensões das técnicas de warping 2D para 3D foram apresentadas, assim como uma técnica 3D estendida para 2D.

Uma dessas extensões foi implementada para experimentação e visualização de resultados, e seus detalhes e decisões de implementação são descritos no capítulo 5.

6.1 Trabalho Futuro

Diversas técnicas não previamente experimentadas foram apresentadas neste trabalho, e uma delas foi implementada. Um objetivo válido é implementar estas técnicas para verificar sua propriedade e utilidade. O algoritmo de warping por malhas de superfícies splines em três passos, por exemplo, precisa ser comparado, em termos de eficiência computacional, com outros algoritmos de warping aplicados a dados volumétricos amostrados—um resultado favorável à técnica em três passos é esperado.

A combinação de uma fase de warping com o algoritmo de combinação de estruturas combinatórias parece ser uma alternativa extremamente atraente para criar metamorfoses entre objetos B-Rep. Testes preliminares no caso 2D indicaram resultados bastante bons, embora uma análise mais profunda do caso 3D seja necessária.

Cada uma das técnicas propostas de fato possui algumas particularidades e espaço para melhorias. O algoritmo de blending físico 2D, por exemplo, pode ser melhorado de várias formas, como mencionado na seção respectiva no capítulo 3. Também, a técnica de warping baseada em campo pode se beneficiar de melhoramentos no cálculo do peso das características.

Uma melhor análise dessas técnicas irá auxiliar nas decisões de implementação necessárias ao desenvolvimento de sistemas de morphing 2D e 3D a serem utilizados em produção.

7 Referências Bibliográficas

- Alvaro 86 Isabel Navazo Alvaro (1986). *Geometric Modeling of Octree Encoded Polyhedral Objects*. Doctoral Thesis. Departament de Metodes Informatics, Universitat Politecnica de Catalunya.
- Barr 84 Alan Barr (1984). *Global and Local Deformation of Solid Primitives*. Computer Graphics, 18, pages 21-29. Proceedings of SIGGRAPH'84.
- Beier, Neely 92 Thaddeus Beier and Shawn Neely (1992). *Feature-based Image Metamorphosis*. Computer Graphics, 26, 2, pages 35-42, Proceedings of SIGGRAPH'92.
- Blinn 82 James Blinn (1982). *A Generalization of Algebraic Surface Drawing*. ACM Transactions on Graphics, 1, 3, pages 235-256.
- Chen, Parent 89 Shenchang Chen and Richard Parent (1989). *Shape Averaging and its applications to industrial design*. IEEE CG&A, 9, 1, pages 47-54.
- Corel 94 Corel Systems Corporation (1994). *Corel Draw! 4.0*.
- Costa, Darsa 91 Bruno Costa and Lucia Darsa (1991). *Ray: Um Ray Tracer Extensível e Orientado a Objetos*. Final Graduation Project. Computer Science Department, PUC-RJ. (in Portuguese)
- Costa et alli 92 Bruno Costa, Lucia Darsa and Jonas Gomes (1992). *Image Metamorphosis*. Proceedings of SIBGRAP'92, pages 19-27.
- Costa, Darsa 92 Bruno Costa and Lucia Darsa (1992). *Visionaire*. Commercial morphing software. Impulse Inc., Minneapolis.

- Costa 94 Bruno Costa (1994). *Image Deformation and Metamorphosis*. Master's Dissertation. Computer Science Department, PUC-Rio. (in Portuguese; english version available from bruno@visgrafimpa.br)
- Farin 88 G. Farin (1988). *Curves and Surfaces for Computer Aided Geometric Design*. Academic Press, San Diego.
- Figueiredo, Carvalho 91 Luiz Henrique de Figueiredo and Paulo Cezar Carvalho (1991). *Introdução à Geometria Computacional*. XVIII Colóquio Brasileiro de Matemática.
- Foley et alli 90 James Foley, Andries van Dam, Steven Feiner and John Hughes (1990). *Computer Graphics: Principles and Practice*, second edition. Addison-Wesley.
- Gomes et alli 93 J. Gomes, C. Hoffmann, V. Shapiro and L. Velho (1993). *Modeling in Graphics*. SIGGRAPH'93 Course Notes.
- Gomes, Velho 92 Jonas Gomes and Luiz Velho (1992). *Implicit Objects in Computer Graphics*. IMPA, Rio de Janeiro.
- Gomes, Velho 94 Jonas Gomes and Luiz Velho (1994). *Computação Gráfica: Imagem*. Preprint.
- Gonzalez, Wintz 87 Rafael Gonzalez and Paul Wintz (1987). *Digital Image Processing*, second edition. Addison-Wesley, Reading.
- Heckbert 86 Paul Heckbert (1986). *Survey of Texture Mapping*. CG&A, 6, 11, pages 56-67.
- Hughes 92 John Hughes (1992). *Scheduled Fourier Volume Morphing*. Computer Graphics, 26, 2, pages 43-46, Proceedings of SIGGRAPH'92.
- Kent et alli 92 James Kent, Wayne Carlson and Richard Parent (1992). *Shape Transformation for Polyhedral Objects*. Computer Graphics, 26, 2, pages 47-54, Proceedings of SIGGRAPH'92.

- Levy 93 Carlos Henrique Levy (1993). *IUP/LED: Uma Ferramenta Portátil de Interface com Usuário*. Master's Dissertation. Computer Science Department, PUC-Rio. (in Portuguese)
- Mäntylä 88 M. Mäntylä (1988). *Introduction to Solid Modeling*. Computer Science Press, Rockville, MD.
- Perlin 92 Ken Perlin (1992). *Interactive Image Warping using Delaunay Triangulation*. Preprint. New York University.
- Porter, Duff 84 Thomas Porter and Tom Duff (1984). *Compositing Digital Images*. *Computer Graphics*, 18, 3, pages:253-259. Proceedings of SIGGRAPH'84.
- Preparata, Shamos 85 F. Preparata and M. Shamos (1985). *Computational Geometry: An Introduction*. Springer-Verlag, New York.
- Rockwood 89 A. Rockwood (1989). *The Displacement Method for Implicit Blending Surfaces in Solid Models*. *ACM Transactions on Graphics*, 8, 4, pages 279-297.
- Rockwood, Owen 87 A. Rockwood and J. Owen (1987). *Using Implicit Surfaces to Blend Arbitrary Solid Models*. *Geometric Modeling: Algorithms and Trends*. SIAM.
- Scaroff, Pentland 91 Stan Scaroff and Alex Pentland (1991). *Generalized Implicit Functions for Computer Graphics*. *Computer Graphics*, 25, 4, pages 247-250. Proceedings of SIGGRAPH'91.
- Sederberg, Greenwood 92 Thomas Sederberg and Eugene Greenwood (1992). *A Physically Based Approach to 2-D Shape Blending*. *Computer Graphics*, 26, 2, pages 25-34. Proceedings of SIGGRAPH'92.
- Sederberg, Parry 86 Thomas Sederberg and Scott Parry (1986). *Free-Form Deformation of Solid Geometric Models*. *Computer Graphics* 20, 4, pages 151-160. Proceedings of SIGGRAPH'86.
- Webster's 90 *Webster's Encyclopedic Unabridged Dictionary of the English Language* (1989). Portland House.

Wolberg 90

George Wolberg (1990). *Digital Image Warping*. IEEE Computer Society Press.