

# Análise Espectral de Superfícies e Aplicações em Computação Gráfica

Fernando Ferrari de Goes<sup>1</sup>

23 de Julho de 2009

## Banca Examinadora:

- Siome Klein Goldenstein  
Instituto de Computação - UNICAMP (Orientador)
- Marcelo Siqueira  
Departamento de Informática e Matemática Aplicada - UFRN
- Jorge Stolfi  
Instituto de Computação - UNICAMP
- Wu Shin-Ting  
Faculdade de Engenharia Elétrica e de Computação - UNICAMP (Suplente)
- Hélio Pedrini  
Instituto de Computação - UNICAMP (Suplente)

---

<sup>1</sup>Suporte financeiro Bolsa FAPESP (processo 06/58007-7) 2007–2009

© Fernando Ferrari de Goes, 2009.  
Todos os direitos reservados.

# Resumo

Em computação gráfica, diversos problemas consistem na análise e manipulação da geometria de superfícies. O operador Laplace-Beltrami apresenta autovalores e autofunções que caracterizam a geometria de variedades, proporcionando poderosas ferramentas para o processamento geométrico. Nesta dissertação, revisamos as propriedades espectrais do operador Laplace-Beltrami e propomos sua aplicação em computação gráfica. Em especial, introduzimos novas abordagens para os problemas de segmentação semântica e geração de atlas em superfícies.

# Abstract

Many applications in computer graphics consist of the analysis and manipulation of the geometry of surfaces. The Laplace-Beltrami operator presents eigenvalues and eigenfunctions which characterize the geometry of manifolds, supporting powerful tools for geometry processing. In this dissertation, we revisit the spectral properties of the Laplace-Beltrami operator and apply them in computer graphics. In particular, we introduce new approaches for the problems of semantic segmentation and atlas generation on surfaces.

# Agradecimentos

Este momento encerra não apenas o meu mestrado, mas sim um ciclo de cinco anos. Um período que remete ao meu terceiro ano de graduação, quando procurei o Prof. Siome para minha primeira iniciação científica. Esses anos foram de muita dedicação e muitas conquistas, as quais só foram possíveis graças à colaboração de muitos. Por isso, gostaria de agradecer profundamente ao Prof. Siome pela amizade e liberdade que me permitiu navegar entre várias áreas até achar aquela que mais me interessava, e também ao Prof. Luiz Velho pelo exemplo, ensinamentos, confiança e as muitas e muitas oportunidades que me ofereceu. Também quero agradecer aos amigos de graduação, do LIV e do VISGRAF-IMPA. Em especial, ao Thiago, Yuri e Samuel que sempre arranjavam um colchão para eu passar minhas temporadas no IMPA. Agradeço principalmente à minha família, a grande responsável pela minha luta. Quero agradecer aos meus pais por tudo e mais um pouco que fizeram, sempre pensando no meu bem sem medir esforços, permitindo que eu pudesse me dedicar completamente aos meus sonhos. À minha irmã Vivian, por ser sempre amiga e companheira e olhar pelos meus pais enquanto morava fora. E à minha namorada Juliana, a fonte eterna da minha inspiração. Por fim, quero agradecer ao Instituto de Computação, à Universidade Estadual de Campinas e à FAPESP pela infra-estrutura e o suporte financeiro.

# Sumário

<b>Resumo</b>	<b>vi</b>
<b>Abstract</b>	<b>vii</b>
<b>Agradecimentos</b>	<b>viii</b>
<b>1 Introdução</b>	<b>1</b>
<b>2 Fundamentos da Análise Espectral</b>	<b>3</b>
2.1 Notações Básicas . . . . .	3
2.2 Operadores para Métodos Espectrais . . . . .	4
2.3 Operador Laplace-Beltrami . . . . .	6
2.4 Propriedades . . . . .	7
2.4.1 Generalização da Análise de Fourier . . . . .	7
2.4.2 Invariância a Isometrias . . . . .	8
2.4.3 Unicidade dos Espectros . . . . .	9
2.4.4 Suavidade das Autofunções . . . . .	10
2.4.5 Domínios Nodais . . . . .	10
2.4.6 Autovalores e Medidas Globais . . . . .	10
2.5 Discretização do operador Laplace-Beltrami para Malhas Triangulares . . . . .	11
2.5.1 Malhas Triangulares . . . . .	11
2.5.2 Fórmula da Cotangente . . . . .	11
2.5.3 Derivação . . . . .	13
2.5.4 Problema de Autovalores e Autovetores Generalizado . . . . .	14
<b>3 Aplicações</b>	<b>16</b>
3.1 Projeto de Filtros . . . . .	16
3.2 Descritores Globais e Locais . . . . .	17
3.3 Detecção de Simetrias . . . . .	18
3.4 Segmentação . . . . .	19

3.5	Remalhamento . . . . .	20
<b>4</b>	<b>Segmentação Hierárquica de Objetos Articuláveis</b>	<b>21</b>
4.1	Motivação . . . . .	21
4.2	Trabalhos Relacionados . . . . .	22
4.3	Teoria . . . . .	24
4.3.1	Distância de Difusão . . . . .	24
4.3.2	Estrutura Medial . . . . .	26
4.4	Algoritmo . . . . .	27
4.5	Resultados . . . . .	30
<b>5</b>	<b>Geração de Atlas Baseado em Estruturas Globais para Modelos Orgânicos</b>	<b>35</b>
5.1	Motivação . . . . .	35
5.2	Trabalhos Relacionados . . . . .	36
5.3	Método VSA . . . . .	38
5.4	Notação . . . . .	39
5.5	Algoritmo . . . . .	39
5.6	Resultados . . . . .	45
5.7	Aplicações . . . . .	48
5.7.1	Remalhamento Quadrangular . . . . .	48
5.7.2	Deformação baseada em Cella . . . . .	49
<b>6</b>	<b>Conclusão</b>	<b>50</b>
	<b>Bibliografia</b>	<b>52</b>

# Lista de Tabelas

2.1	Tempo de execução para computar os 200 primeiros autovalores e os autovetores do operador Laplace-Beltrami em malhas triangulares. . . . .	15
4.1	Tempo de execução do algoritmo de segmentação. . . . .	33

# Lista de Figuras

2.1	Autofunções do operador Laplace-Beltrami. Os números indicam os índices das autofunções. Os modelos usados são, de cima para baixo, retângulo com condições de borda de Neumann, retângulo com condições de borda de Dirichlet e superfície sem borda. As autofunções são ilustradas pelas cores variando de azul para vermelho, Nas duas primeiras linhas, as autofunções também são representadas como funções de altura. . . . .	7
2.2	Exemplos de harmônicas esféricas. Cada autofunção corresponde a um autovalor diferente e, por isso, possui um número diferente de domínios nodais. . . . .	8
2.3	Comparação dos autovetores obtidos, respectivamente, com o operador Laplace-Beltrami combinatório e com a fórmula da cotangente (Eq. 2.21). . . . .	12
3.1	Filtros aplicados sobre as coordenadas 3D das superfícies. Na primeira linha, os números indicam quantos coeficientes da transformada de Fourier foram utilizados para reconstruir cada exemplo. Na segunda linha, os gráficos mostram os filtros utilizados. . . . .	17
3.2	Espectro de seis poses do modelo de um cavalo e de um gorila. Os gráficos contém, respectivamente, 100, 50 e 20 autovalores. . . . .	18
3.3	Pontos correspondentes em superfícies obtidos através de simetrias Euclidianas dos vetores <i>GPS</i> . As imagens enumeradas ilustram as autofunções utilizadas nos vetores <i>GPS</i> . Na linha superior, as simetrias são detectadas por reflexões das autofunções. Na linha inferior, as simetrias preservam os valores das autofunções. . . . .	19
3.4	Segmentações induzidas pelas linhas nodais das autofunções do operador Laplace-Beltrami. Os números indicam os índices das autofunções utilizadas. . . . .	20
3.5	Complexos Morse-Smale construídos a partir de autofunções do operador Laplace-Beltrami. Os números indicam os índices das autofunções utilizadas. . . . .	20

4.1	Resultados do nosso algoritmo de segmentação. . . . .	24
4.2	Propagação de frentes a partir de cada estrutura medial. Esferas indicam os pontos das estruturas mediais. As cores representam os isovalores da distância de difusão entre as estruturas mediais e os pontos da superfície, variando de azul para vermelho. Note que as cores mudam rapidamente ao passar pelas concavidades. . . . .	25
4.3	Esferas representam os pontos das estruturas mediais. (a) A função <i>ADD</i> da região e sua estrutura medial. As cores indicam os valores da função <i>ADD</i> , crescendo de azul para vermelho. (b) As estruturas mediais e os segmentos iniciais. (c) As estruturas mediais e os segmentos finais. . . . .	28
4.4	Diferentes cortes da hierarquia de segmentos dos modelos <i>raptor</i> e <i>dinopet</i> . . . . .	28
4.5	A árvore hierárquica de segmentos do modelo <i>dog</i> . Cada cor indica um nível de detalhe diferente. . . . .	30
4.6	Várias poses e os segmentos consistentes gerados pelo nosso método. . . . .	32
4.7	Segmentações consistentes em escalas sucessivas de sete poses de um modelo humano escaneado [1]. As imagens mostram os segmentos do primeiro e do segundo nível de detalhe, respectivamente. . . . .	32
4.8	(a-c) Nosso método é robusto a ruído. As imagens comparam as segmentações do modelo original com duas versões modificadas por ruídos moderados e exagerados. (d) Nosso método também é robusto a pequenos circuitos topológicos. (e) Entretanto, falha com eventos grandes. . . . .	33
4.9	Esqueletos extraídos a partir dos segmentos. As esferas azuis indicam os centros de massa dos segmentos e as esferas cinzas indicam os núcleos das articulações dos personagens. . . . .	34
5.1	Cartas da perna esquerda do modelo <i>raptor</i> . (A) Computar cartas em cada segmento separadamente resulta em estruturas desalinhadas. Nosso método controla o alinhamento ordenando a dependência entre os segmentos. Ordenações diferentes podem induzir resultados diferentes. (B) parte inferior da perna restringe parte superior; (C) parte superior da perna restringe parte inferior. . . . .	40
5.2	Cartas do pescoço do modelo <i>raptor</i> . O pescoço é restrito pela cabeça por quatro cortes. (A) Inicialização das partições usando os triângulos adjacentes aos cortes. (B) Funções harmônicas computadas a partir dos cortes. (C) Triângulos pré-atribuídos para cada partição durante a etapa de colagem. (D) Cartas finais. Triângulos brancos representam as sementes dos <i>proxies</i> . . . . .	41

5.3	Cartas do maxilar do modelo <i>raptor</i> . (A) Cartas iniciais. A mandíbula impõe três cortes sobre o maxilar. (B) As cores indicam o erro de distorção entre as partições e seus respectivos <i>proxies</i> , aumentando de azul para vermelho. (C) Inserção de um novo <i>proxy</i> inicializado com o triângulo de maior erro. (D) Cartas finais após as iterações. . . . .	43
5.4	Cartas para a parte superior do rabo do modelo <i>raptor</i> . A parte inferior do rabo e o quadril impõem seis cortes (três cada). (A) Partições iniciais e grafo dual. (B) Cartas finais após contrair as arestas em vermelho. . . . .	43
5.5	(A) Segmentação semântica computada com o método do Capítulo 4. (B) Atlas final. Observe o alinhamento das curvas geométricas cruzando as estruturas semânticas. (C) Distorção da métrica $\mathcal{L}^{2,1}$ entre as cartas e os respectivos <i>proxies</i> . . . . .	45
5.6	Exemplos de atlas alinhados a estruturas gerados pelo nosso método. Em cada exemplo, mostramos a segmentação e a vista de frente e por trás do atlas. . . . .	46
5.7	Comparação dos nossos resultados com os métodos WYSIWYG [100], Iso-Charts [109] e VSA [16]. As imagens comparativas são cortesia de [100]. . . . .	47
5.8	Aplicação do atlas estruturado para remalhamento quadrangular. As imagens da esquerda mostram os atlas utilizados. As imagens centrais indicam as direções dos gradientes dos potenciais computados pelo método [98]. Nas imagens da direita, mapeamos uma textura de xadrez usando os campos potenciais como coordenadas. . . . .	48
5.9	Aplicação do atlas estruturado para a deformação de superfícies baseado em celas. Da esquerda para direita: nosso atlas estruturado, a cela extraída do atlas e deformações. . . . .	49

# Capítulo 1

## Introdução

Modelagem geométrica é o ramo da computação gráfica que desenvolve ferramentas para a análise e o processamento de objetos gráficos. Estas ferramentas são essenciais em diversas aplicações, entre as quais podemos citar geração de atlas e parametrização, adaptação e remalhamento de superfícies, extração de esqueleto e animação. Para ilustrar a importância dessas ferramentas, vejamos o problema de animação de superfícies. Nesta aplicação, são necessários mecanismos simples e intuitivos para manipular a superfície preservando ao máximo a geometria original. Assim, as ferramentas de análise revelam as características da superfície a serem conservadas durante a animação, como volume e curvatura, enquanto as técnicas de processamento permitem o mapeamento da interação do artista em deformações que respeitem as restrições anteriores.

As ferramentas de modelagem são caracterizadas por três aspectos. O primeiro deles é a escala de atuação. Entende-se por escala o tamanho da região em que a ferramenta age, podendo variar de medidas locais até globais. O segundo aspecto é a invariância da ferramenta a grupos de transformações, como transformações rígidas e afins, isometrias e homotopias. A invariância a transformações rígidas, por exemplo, garante que a análise em questão seja independente da rotação e translação submetida ao objeto. O último aspecto refere-se aos modelos matemáticos utilizados para discretizar ferramentas definidas em domínios contínuos. Estes modelos devem garantir comportamento assintótico convergente aos valores contínuos, além de serem robustos à presença de ruídos.

O desenvolvimento de ferramentas de modelagem também está fortemente vinculado ao domínio do objeto a ser manipulado. Nesta dissertação, em particular, trabalhamos com variedades bidimensionais compactas, possivelmente com bordas. Estes objetos correspondem a superfícies regulares e orientáveis imersas em  $\mathbb{R}^3$ . Também adotamos malhas triangulares como nosso modelo discreto. No entanto, vale ressaltar que todos os conceitos aqui apresentados são facilmente generalizados para outras discretizações, como nuvem de pontos, e para variedades de qualquer dimensão.

Tópicos em geometria diferencial apresentam uma ampla formulação para o estudo

de superfícies. Seguindo a descrição feita em [23], uma superfície é determinada unicamente pela combinação da primeira e segunda forma fundamental mais uma transformação rígida. A primeira forma fundamental, também chamada de tensor de métrica, indica a geometria intrínseca do objeto, fornecendo meios para medir comprimentos de arco e a área da superfície e, assim, caracterizá-la sob uma isometria. A segunda forma fundamental, por sua vez, analisa a geometria extrínseca, informando como a superfície é imersa em  $\mathbb{R}^3$  e, conseqüentemente, distinguindo as isometrias. Por fim, para posicionar a superfície em relação a um sistema de coordenadas, é necessário especificar uma rotação e uma translação.

A caracterização de superfícies com os três termos anteriores proporciona as ferramentas básicas utilizadas em modelagem geométrica. Estas ferramentas são definidas pela combinação das formas fundamentais em um operador denominado operador de forma. O operador de forma é determinado em cada ponto da superfície e identifica medidas locais e invariantes a transformações rígidas, incluindo curvaturas gaussianas e média e curvaturas e direções principais. Estas medidas diferenciais já foram discretizadas para vários objetos gráficos. Especialmente em malhas triangulares [71, 72], tais valores são aplicados em problemas como simplificação [34], suavização [22] e simulação [31]. No entanto, a localidade dos termos diferenciais está acompanhada de limitações numéricas, já que a maioria das superfícies é discretizada de maneira irregular, com densidade de amostragem não uniforme.

Para suprir estas dificuldades, recentes pesquisas introduzem métodos espectrais como uma representação alternativa para superfícies [108]. Estas técnicas consistem na definição de novos operadores sobre superfícies como, por exemplo, matrizes de afinidade, e procedem computando seus autovalores e autofunções. O espectro destes operadores complementa a análise de superfícies de duas maneiras. Em primeiro lugar, este espectro oferece uma representação multi-escala para superfícies, expandindo conceitos locais para medidas globais. Em segundo lugar, as técnicas espectrais são robustas computacionalmente, pois se baseiam em métodos numéricos capazes de lidar com amostragens irregulares.

Motivada pelas vantagens dos métodos espectrais, esta dissertação de mestrado propõe a análise espectral em superfícies a partir do operador Laplace-Beltrami. Estudamos este operador em especial porque ele complementa os conceitos de geometria diferencial, originando ferramentas multi-escala e invariantes a isometrias. Além disso, o seu espectro contém informações tanto sobre a geometria quanto sobre a topologia da superfície. Os seus autovalores, por exemplo, são suficientes para calcular a área e o comprimento das bordas da superfície. Já as suas autofunções generalizam a análise de Fourier para variedades, proporcionando decomposições em frequência de sinais definidos em superfícies.

# Capítulo 2

## Fundamentos da Análise Espectral

Neste capítulo, discutimos os conceitos básicos relacionados à análise espectral do operador Laplace-Beltrami. Assim, justificamos o uso de métodos espectrais desenvolvidos a partir desse operador como ferramentas para modelagem geométrica. Em especial, demonstramos que o operador Laplace-Beltrami apresenta informações invariantes a isometrias e abrangendo diversas escalas. Por fim, descrevemos a discretização do operador Laplace-Beltrami para malhas triangulares e revisamos um método numérico robusto e eficiente para computar seus autovalores e autovetores.

### 2.1 Notações Básicas

Antes de começarmos, precisamos adotar alguns notações. Denotamos por  $\mathcal{M}_m^n$  a variedade compacta de dimensão  $m$  imersa em  $\mathbb{R}^n$  ( $m \leq n$ ). As suas bordas são indicadas por  $\partial\mathcal{M}_m^n$ . As superfícies, por exemplo, correspondem a variedades  $\mathcal{M}_2^3$ .

Dado um ponto  $x_0 \in \mathcal{M}_m^n$ , a vizinhança deste ponto pode ser representada pela parametrização  $x = \mathbf{x}(\mathbf{u}) : \mathbb{R}^m \rightarrow \mathbb{R}^n$  (letras em negrito indicam vetores). Calculando as derivadas parciais de  $\mathbf{x}(\mathbf{u})$ , definimos como  $G = [g_{ij}]_{m \times m} = [\mathbf{x}_{u_i}^T \mathbf{x}_{u_j}]$  o tensor de métrica de  $\mathcal{M}_m^n$  em  $x_0$ . O determinante do tensor de métrica é indicado por  $|G|$  e sua matriz inversa por  $G^{-1} = [g^{ij}]$ . O tensor de métrica determina a medida infinitesimal do volume  $mD$  de  $\mathcal{M}_m^n$  através de

$$dx = \sqrt{|G|} \prod_{i=1 \dots m} du_i. \quad (2.1)$$

Exemplificando novamente com superfícies, temos o tensor de métrica em  $\mathcal{M}_2^3$  dado por

$$G = \begin{bmatrix} \mathbf{x}_{u_1}^T \mathbf{x}_{u_1} & \mathbf{x}_{u_1}^T \mathbf{x}_{u_2} \\ \mathbf{x}_{u_2}^T \mathbf{x}_{u_1} & \mathbf{x}_{u_2}^T \mathbf{x}_{u_2} \end{bmatrix}, \quad (2.2)$$

e a medida infinitesimal de área igual a  $dx = \sqrt{|G|}du_1du_2$ . As derivadas parciais  $\mathbf{x}_{u_1}$  e  $\mathbf{x}_{u_2}$  também indicam o vetor normal  $n = \mathbf{x}_{u_1} \times \mathbf{x}_{u_2} / \|\mathbf{x}_{u_1} \times \mathbf{x}_{u_2}\|$ .

Por fim, indicamos por  $L^2(\mathcal{M}_m^n)$  o espaço de Lebesgue definido em  $\mathcal{M}_m^n$ . O espaço de Lebesgue é um espaço de medida que opera sobre funções  $f : \mathcal{M}_m^n \rightarrow \mathbb{R}$ , caracterizando-se pelo produto interno

$$\langle f, g \rangle = \int_{\mathcal{M}_m^n} f(x)g(x)dx. \quad (2.3)$$

Uma função  $f$  pertence a  $L^2(\mathcal{M}_m^n)$  se e somente se sua norma for finita,

$$\|f\|^2 = \int_{\mathcal{M}_m^n} f(x)^2dx < \infty. \quad (2.4)$$

## 2.2 Operadores para Métodos Espectrais

Os métodos espectrais se baseiam na análise de autovalores e autofunções. Especificado um domínio  $\mathcal{M}_m^n$ , estes pares são computados a partir de operadores  $\mathcal{A} : L^2(\mathcal{M}_m^n) \rightarrow L^2(\mathcal{M}_m^n)$  que respeitam duas propriedades básicas. Dado as funções  $f, g \in L^2(\mathcal{M}_m^n)$  e um escalar  $\alpha$ , o operador  $\mathcal{A}$  deve ser:

- **Linear:**  $\mathcal{A}(f + \alpha g) = \mathcal{A}(f) + \alpha\mathcal{A}(g)$  e
- **Auto-Adjunto:**  $\langle \mathcal{A}(f), g \rangle = \langle f, \mathcal{A}(g) \rangle$ .

Com estas propriedades, podemos definir os teoremas que justificam e motivam o uso de métodos espectrais. O primeiro deles se refere à decomposição do operador  $\mathcal{A}$ .

**Teorema 1** (Teorema Espectral). *Seja  $\mathcal{A}$  um operador linear e auto-adjunto em  $L^2(\mathcal{M}_m^n)$ . A decomposição espectral do operador  $\mathcal{A}$  consiste em autofunções  $\{\phi^i\}$  ortogonais e de norma unitária,  $\|\phi^i\| = 1$ , associadas a autovalores reais  $\{\lambda^i\}$  e satisfazendo*

$$\mathcal{A}(\phi^i) = \lambda^i\phi^i. \quad (2.5)$$

As autofunções de  $\mathcal{A}$  compõem uma base ortonormal para  $L^2(\mathcal{M}_m^n)$ . Logo, podemos reescrever qualquer função  $f \in L^2(\mathcal{M}_m^n)$  por  $f = \sum_i \langle f, \phi^i \rangle \phi^i$ . Os autovalores de  $\mathcal{A}$ , por sua vez, indicam a importância de cada autofunção na imagem resultante desse operador. Isto é facilmente verificado ao reescrever  $\mathcal{A}(f) = \sum_i \lambda^i \langle f, \phi^i \rangle \phi^i$ .

Uma interpretação complementar dos autovalores e autofunções do operador  $\mathcal{A}$  pode ser obtida através do coeficiente de Rayleigh-Ritz, o qual é definido por

$$R_{\mathcal{A}}(f) = \frac{\langle f, \mathcal{A}(f) \rangle}{\|f\|^2}. \quad (2.6)$$

Dada uma ordenação crescente dos autovalores de  $\mathcal{A}$ ,  $\{\lambda^0 \leq \dots \leq \lambda^i \leq \dots\}$ , identificamos a autofunção  $\phi^i$  como aquela que minimiza  $R_{\mathcal{A}}$  sendo ortogonal ao subespaço gerado pelas autofunções  $\{\phi^j\}_{j < i}$ . Já o autovalor  $\lambda^i$  representa o valor mínimo de  $R_{\mathcal{A}}$ . Este resultado, na realidade, segue como um corolário do Teorema 2.

**Teorema 2** (Teorema Min-Max (Courant-Fischer)). *Dada a decomposição de um operador linear e auto-adjunto  $\mathcal{A}$  em autovalores  $\{\lambda^0 \leq \lambda^1 \leq \dots \lambda^i \leq \dots\}$  e suas respectivas autofunções  $\{\phi^i\}$  e sendo  $\mathcal{V}_k \subset L^2(\mathcal{M}_m^n)$  qualquer subespaço de dimensão  $k$ , temos que*

$$\lambda^k = \min_{\mathcal{V}_k} \max_{f \in \mathcal{V}_k} R_{\mathcal{A}}(f). \quad (2.7)$$

Com exceção de algumas soluções analíticas, os métodos espectrais são computados sobre modelos discretos. A discretização consiste na amostragem do domínio  $\mathcal{M}_m^n$  em um conjunto de pontos  $\{p_i\}$ . Consequentemente, o operador  $\mathcal{A}$  é aproximado por uma matriz (que indicamos por  $A$ ) e suas autofunções por autovetores. Para se preservar a topologia de  $\mathcal{M}_m^n$ , as amostras são associadas com arestas ponderadas e, assim, a variedade passa a ser representada por um grafo  $\mathcal{G}$ . Os operadores discretos em  $\mathcal{G}$  apresentam a forma geral  $A = D - W$ , conhecida como Laplaciano de grafos [14]. Nesta forma matricial,  $W$  é uma matriz simétrica com elemento  $w_{ij}$  igual a zero, caso  $p_i$  e  $p_j$  não estejam conectadas, ou igual ao peso da aresta existente entre  $p_i$  e  $p_j$ . Já  $D$  é uma matriz diagonal com  $d_{ii} = \sum_j w_{ij}$ .

Diversos operadores podem ser definidos especificando diferentes matrizes  $W$  [108]. Em [50], por exemplo, o peso das arestas corresponde a uma combinação de distâncias geodésicas e medidas de concavidade em superfícies, enquanto, em [65], o peso compara as curvaturas principais entre pontos vizinhos de uma superfície.

Na área de aprendizado de máquina, a matriz  $W$  também é conhecida como matriz de afinidade. Uma classe especial de matriz de afinidade é a matriz de Gram, na qual a afinidade (o peso) entre duas amostras está relacionada ao tensor de métrica de  $\mathcal{M}_m^n$ . Vários métodos espectrais se baseiam nas matrizes de Gram para calcular coordenadas para as amostras que aproximem espaços lineares de baixa dimensão, preservando ao máximo as afinidades iniciais. Exemplos destes métodos são *Multi-Dimensional Scaling* (MDS) [11], *IsoMaps* [97], *Linear Laplacian Embedding* [85], *Laplacian EigenMaps* [5] e *kernel PCA* [90].

Para ilustrar melhor o uso das matrizes de Gram, vejamos a sua aplicação em superfícies. Neste caso, a matriz  $A$  contém em cada elemento  $a_{ij}$  a distância geodésica entre  $p_i$  e  $p_j$ . Partindo do Teorema 1, podemos escrever  $a_{ij} = \sum_l \lambda^l \phi_i^l \phi_j^l$ , onde  $|\lambda^l| \geq |\lambda^{l+1}|$ . Logo, podemos aproximar os elementos de  $A$  truncando a expressão anterior com apenas os  $k$  maiores autovalores, em módulo, e seus autovetores. Agora, definindo para cada amostra  $p_j$  a nova coordenada  $\mathbf{y}_j = \left[ \sqrt{\lambda^i} \phi_j^i \right]_{i=1}^k$ , obtemos a melhor aproximação em um

espaço linear de dimensão  $k$  da afinidade (não-) linear contida em  $A$ . Este resultado foi utilizado em [109] com  $k = 2$  para parametrizar superfícies no plano e, em [49] com  $k = 3$ , para computar poses canônicas para objetos articuláveis.

Nesta dissertação, adotamos uma perspectiva diferente sobre operadores de métodos espectrais. Ao invés de avaliarmos a implicação de diferentes pesos  $W$  sobre a decomposição espectral do operador  $A$ , analisamos um único operador definido continuamente em  $\mathcal{M}_m^n$ . O nosso estudo se concentra no operador Laplace-Beltrami, o qual generaliza o Laplaciano de  $\mathbb{R}^n$  para variedades não-lineares. Este operador também abrange e unifica conceitos de diferentes áreas, incluindo geometria, física e análise de sinais. Por fim, discutimos os modelos discretos do operador Laplace-Beltrami e como estes modelos preservam suas propriedades espectrais.

## 2.3 Operador Laplace-Beltrami

O operador Laplace-Beltrami (LBop) em uma variedade  $\mathcal{M}_m^n$  é definido por

$$\Delta = -\nabla \circ \nabla = -\frac{1}{\sqrt{|G|}} \sum_{i,j=1\dots m} \frac{\partial}{\partial x_i} \left( g^{ij} \sqrt{|G|} \frac{\partial}{\partial x_j} \right). \quad (2.8)$$

Esta expressão é simplificada quando aplicada a espaços Euclidianos  $\mathcal{M}_n^n \subset \mathbb{R}^n$ . Já que o tensor de métrica em  $\mathbb{R}^n$  equivale à matriz identidade, o LBop se reduz a uma somatória de segunda derivadas,

$$\Delta_{\mathbb{R}^n} = - \sum_{i=1\dots n} \frac{\partial^2}{\partial x_i^2}. \quad (2.9)$$

Pela Eq. 2.8, é fácil verificar que o LBop é um operador linear e auto-adjunto. Além disso, o LBop também é um operador local (analisando somente uma pequena vizinhança ao redor de cada ponto) e semipositivo ( $\langle f, \Delta f \rangle \geq 0$ ).

Seguindo o Teorema 1, a decomposição espectral do LBop é determinada por

$$\Delta \phi^i = \lambda^i \phi^i. \quad (2.10)$$

Quando  $\partial \mathcal{M}_m^n \neq \emptyset$ , a Eq. 2.10 requer condições de borda. No caso de condições de borda de Dirichlet, temos que  $\phi^i(x) = 0, \forall x \in \partial \mathcal{M}_m^n$ . Já no caso de condições de borda de Neumann, temos que  $\frac{\partial \phi^i}{\partial n} = 0, \forall x \in \partial \mathcal{M}_m^n$ .

Os autovalores do LBop compõem uma sequência infinita e crescente  $\{0 = \lambda^0 < \lambda^1 \leq \dots \uparrow \infty\}$ . Já suas autofunções  $\{\phi^i\}$  correspondem aos modos de vibração em  $\mathcal{M}_m^n$ . Esta interpretação se deve ao fato de que a Eq. 2.10 equivale à equação de Helmholtz estacionária, a parte independente do tempo da equação de onda  $\Delta u = u_{tt}$ . Na Figura 2.1, ilustramos as autofunções para diferentes superfícies. Nas duas primeiras linhas, utilizamos os dois tipos de condições de borda para um plano retangular e representamos

suas autofunções como funções de altura. Na terceira linha, mostramos as autofunções para uma superfície sem borda através de um intervalo de cores crescendo de azul para vermelho.

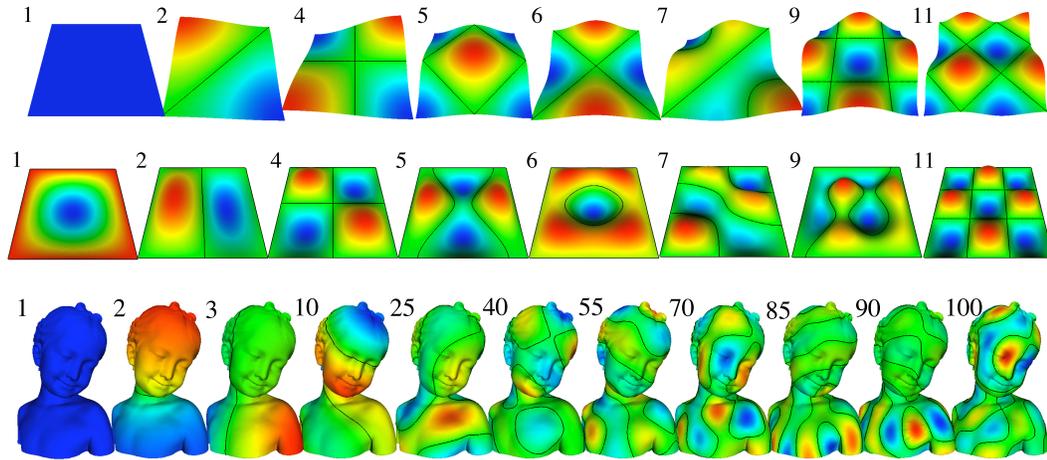


Figura 2.1: Autofunções do operador Laplace-Beltrami. Os números indicam os índices das autofunções. Os modelos usados são, de cima para baixo, retângulo com condições de borda de Neumann, retângulo com condições de borda de Dirichlet e superfície sem borda. As autofunções são ilustradas pelas cores variando de azul para vermelho. Nas duas primeiras linhas, as autofunções também são representadas como funções de altura.

## 2.4 Propriedades

Nesta seção, descrevemos algumas propriedades espectrais do LBop [13, 84]. Estas propriedades são válidas para o modelo contínuo de qualquer variedade  $\mathcal{M}_m^n$  e unificam conceitos de geometria, física e análise de sinais. Em particular, a aplicação dessas propriedades em superfícies permite o desenvolvimento de ferramentas de modelagem multi-escala, invariante a isometrias e robustas numericamente.

### 2.4.1 Generalização da Análise de Fourier

A análise de Fourier oferece as ferramentas clássicas para o estudo de sinais em  $\mathbb{R}^n$ . No caso da reta real, esta análise introduz as funções  $\sin(\omega x)$  e  $\cos(\omega x)$  como uma nova base ortogonal para o espaço  $L^2(\mathbb{R})$ . Assim, ao projetar os sinais nestas bases, informações antes expressas espacialmente passam a ser representadas como a contribuição de diferentes frequências  $\omega$ . A função que mapeia frequências em contribuições é denominada transformada de Fourier e é fundamental em problemas como remoção de ruídos, compressão e codificação.

Partindo da Eq. 2.9, podemos constatar que as bases da análise de Fourier são autofunções do LBop em  $\mathbb{R}^n$  com autovalores  $\omega^2$ . Exemplificando com a reta real, temos

$$\Delta_{\mathbb{R}} \sin(\omega x) = -\frac{d^2}{dx^2} \sin(\omega x) = \omega^2 \sin(\omega x), \quad (2.11)$$

$$\Delta_{\mathbb{R}} \cos(\omega x) = -\frac{d^2}{dx^2} \cos(\omega x) = \omega^2 \cos(\omega x). \quad (2.12)$$

Portanto, podemos estender a construção da análise de Fourier para qualquer variedade  $\mathcal{M}_m^n$  usando as autofunções do LBop como generalizações dos senos e cossenos e os autovalores do LBop como o quadrado da frequência. Um exemplo desta generalização é as harmônicas esféricas (Figura 2.2), que correspondem às autofunções do LBop em uma esfera. Também vale ressaltar que, apesar do LBop ser um operador local, suas autofunções revelam informações em diversas escalas. Enquanto as autofunções de maior autovalor são mais locais, as autofunções de menor autovalor indicam informações globais.

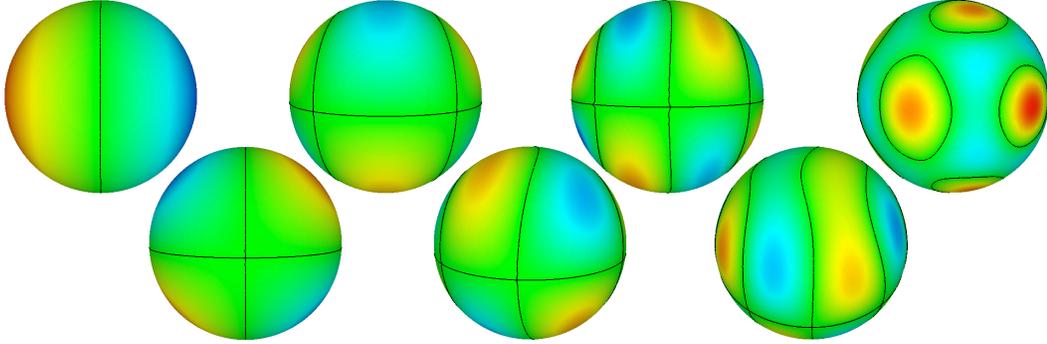


Figura 2.2: Exemplos de harmônicas esféricas. Cada autofunção corresponde a um autovalor diferente e, por isso, possui um número diferente de domínios nodais.

### 2.4.2 Invariância a Isometrias

De acordo com a Eq. 2.8, verificamos que a definição do LBop em  $\mathcal{M}_m^n$  depende somente de seu tensor de métrica  $G$ . O tensor de métrica é responsável pelo cálculo do volume  $mD$  de uma região  $\mathcal{D} \subset \mathcal{M}_m^n$  (Eq. 2.13) e das distâncias geodésicas entre pontos de  $\mathcal{M}_m^n$  (Eq. 2.14).

$$V_{\mathcal{D}} = \int_{\mathcal{D}} dx = \int_{\mathcal{D}} \sqrt{|G|} \prod_{i=1 \dots m} du_i, \quad (2.13)$$

$$geod(x, y) = \min_{\gamma} \int_{\gamma} \sqrt{\langle \gamma', \gamma' \rangle_G} dt, \quad (2.14)$$

onde  $\gamma$  é qualquer arco em  $\mathcal{M}_m^n$  que liga os pontos  $x$  e  $y$ ,  $\gamma'$  é a sua derivada em relação ao parâmetro  $t$  e  $\langle \cdot, \cdot \rangle_G$  é o produto interno definido por  $G$ .

Seja  $\Upsilon : \mathcal{M}_m^n \rightarrow \mathcal{M}_m^{n'}$  uma aplicação isométrica, ou seja, uma aplicação bijetora que conserva as medidas de volume  $mD$  e as distâncias geodésicas. Para que haja esta conservação entre  $\mathcal{M}_m^n$  e  $\mathcal{M}_m^{n'}$ , é necessário que o tensor de métrica não seja alterado por  $\Upsilon$ . Se  $G = G'$ , consequentemente, o LBop é igual em ambas as variedades. Logo, concluímos que o LBop e seus autovalores são invariantes a isometrias. Já suas autofunções são invariantes a isometrias sob uma rotação do espaço  $L^2(\mathcal{M}_m^n)$ . No caso de um autovalor  $\lambda^i$  de multiplicidade unitária, por exemplo, essa rotação faz com que  $\phi^i \circ \Upsilon = -\phi^i$  ou  $\phi^i$ .

Em superfícies, a invariância a isometrias oferece um resultado complementar à geometria diferencial [23]. Para entendermos isto, lembremos que uma superfície é caracterizada pela primeira forma fundamental (geometria intrínseca), pela segunda forma fundamental (geometria extrínseca) e por uma transformação rígida. As medidas diferenciais são determinadas a partir do operador de forma, o qual atua sem depender da transformação rígida. Para isso, o operador de forma combina a primeira e a segunda forma fundamental, distinguindo isometrias através de imersões em  $\mathbb{R}^3$ . Por outro lado, o LBop permite a análise da superfície sob uma isometria, separando a ação de cada forma fundamental. Portanto, dadas as coordenadas 3D de uma superfície, o operador de forma remove a transformação rígida dessas coordenadas e o LBop, de maneira complementar, remove a segunda forma fundamental (imersão em  $\mathbb{R}^3$ ) do operador de forma.

A invariância a isometrias também é um recurso muito interessante para o estudo do movimento de objetos animados. Isto é motivado pelo fato de que, ao deformar a superfície desses modelos, as distâncias entre seus pontos são aproximadamente preservadas. Logo, diferentes poses de um objeto podem ser representadas como diferentes imersões de uma isometria e a interpolação entre essas poses corresponde a transformações isométricas.

### 2.4.3 Unicidade dos Espectros

Até então, vimos que o espectro do LBop é invariante a isometrias e análogo ao quadrado da frequência para variedades. Estas duas propriedades motivaram pesquisadores a se perguntarem *Can we hear the shape of a drum?* [47, 78]. Em outras palavras, esta questão discute se os autovalores do LBop identificam univocamente uma isometria. Em [30, 15], foi provado que existem alguns poucos pares de variedades planares isoespectrais, i.e., com o mesmo espectro. No entanto, a raridade destes casos ainda permite o uso dos autovalores como uma assinatura de isometrias.

Por outro lado, pela construção do LBop, temos que a combinação de seus autovalores e autofunções determina univocamente isometrias. As autofunções do LBop distinguem os poucos casos isoespectrais através de suas informações locais, comparando as variedades ponto a ponto. Nas Seções 3.2 e 4.3.1, apresentamos duas possíveis combinações dos autovalores e das autofunções do LBop.

### 2.4.4 Suavidade das Autofunções

Dada uma função  $f \in L^2(\mathcal{M}_m^n)$ , a suavidade de  $f$  é medida através da *energia de Dirichlet*

$$E_D = \int_{\mathcal{M}_m^n} \|\nabla f(x)\|^2 dx = \langle f, \Delta f \rangle. \quad (2.15)$$

Observe que  $E_D$  corresponde ao numerador do coeficiente de Rayleigh-Ritz (Eq. 2.6). Logo, assim como discutido na Seção 2.2, temos que as autofunções  $\phi^i$  do LBop são as funções de norma unitária mais suaves em  $L^2(\mathcal{M}_m^n)$ , com suavidade igual a  $\lambda^i$ . Isto indica que as autofunções possuem pontos críticos (máximos, mínimos e selas) uniformemente distribuídas sobre  $\mathcal{M}_m^n$ . Esta característica faz com que as autofunções sejam ótimas candidatas para análise topológica de variedades através da construção de grafos de Reeb e de complexos Morse-Smale [81].

### 2.4.5 Domínios Nodais

Outra propriedade topológica interessante sobre as autofunções do LBop é a configuração de seus domínios nodais. Seja  $\phi^i$  a  $i$ -ésima autofunção do LBop em  $\mathcal{M}_m^n$ . Em [18, 43], é provado que as linhas nodais de  $\phi^i$ , i.e.,  $\{x \in \mathcal{M}_m^n : \phi^i(x) = 0\}$ , dividem a variedade em, no máximo,  $i$  domínios. Estas linhas fazem uso da suavidade das autofunções, estando distribuídas sobre a variedade de maneira regular e, assim, compondo domínios uniformes. Os contornos em preto das Figuras 2.1 e 2.2 mostram exemplos dos domínios nodais de várias autofunções.

### 2.4.6 Autovalores e Medidas Globais

A análise, em conjunto, dos autovalores do LBop também revela medidas geométricas sobre sua variedade. Para explicarmos esta relação, precisamos primeiro definir o *kernel* de calor em  $\mathcal{M}_m^n$ . A equação de difusão de calor em uma variedade  $\mathcal{M}_m^n$  é definida por

$$\Delta u + u_t = 0, \quad (2.16)$$

sendo  $u(t, x) : \mathbb{R}^+ \times \mathcal{M}_m^n \rightarrow \mathbb{R}$  e dadas a condição inicial  $u(0, x) = u_0(x)$  e, se necessário, as condições de borda. A solução da Eq. 2.16 pode ser escrita por

$$u(t, x) = \int_{\mathcal{M}_m^n} \mathcal{K}(t, x, y) u_0(y) dy, \quad (2.17)$$

onde  $\mathcal{K}(t, x, y)$  é o *kernel* de calor (também chamado de solução fundamental de calor). O *kernel* de calor, por sua vez, é determinado em função dos autovalores e autofunções do LBop em  $\mathcal{M}_m^n$ ,

$$\mathcal{K}(t, x, y) = \sum_i e^{-\lambda^i t} \phi^i(x) \phi^i(y). \quad (2.18)$$

As medidas globais em  $\mathcal{M}_m^n$  podem ser feitas restringindo o *kernel* de calor através do traço de calor,

$$Z(t) = \int_{\mathcal{M}_m^n} \mathcal{K}(t, x, x) = \sum_i e^{-\lambda^i t}. \quad (2.19)$$

Ao expandir  $Z(t)$  assintoticamente em  $t \rightarrow 0^+$ , obtemos

$$\lim_{t \rightarrow 0^+} Z(t) = (4\pi t)^{-\frac{m}{2}} \left( \sum_{i=0}^n b_i t^{\frac{i}{2}} + O\left(t^{\frac{n+1}{2}}\right) \right). \quad (2.20)$$

Cada coeficiente  $b_i$  está associado a uma medida geométrica de  $\mathcal{M}_m^n$ . No caso de superfícies, o termo  $b_0$  indica a área de  $\mathcal{M}_2^3$ , enquanto  $b_1$  é igual ao comprimento de  $\partial\mathcal{M}_2^3$ .

## 2.5 Discretização do operador Laplace-Beltrami para Malhas Triangulares

Agora, descrevemos a discretização do LBop preservando suas propriedades do domínio contínuo. Concentramos nosso estudo em superfícies aproximadas por malhas triangulares. Esta escolha se deve ao fato de que malhas triangulares correspondem à maioria dos objetos utilizados em computação gráfica. Porém, vale ressaltar que o LBop em superfícies também foi discretizado para nuvens de pontos [7] e para malhas quadrangulares [63]. Em [6], a versão discreta do LBop foi generalizada para variedades de qualquer dimensão. Para isso, este método se baseia em núcleos Gaussianos com estimativas de espaços tangentes para ponderar a influência entre cada par de amostras.

### 2.5.1 Malhas Triangulares

Uma malha triangular  $\mathcal{M}$  é composta por um grafo  $\mathcal{G}$  de vértices  $\mathcal{V}$  e arestas  $\mathcal{E}$ , no qual os vértices são associados a coordenadas 3D. As arestas de  $\mathcal{G}$  são estruturadas em faces triangulares  $\mathcal{F}$ . Deste modo,  $\mathcal{M}$  corresponde a um complexo simplicial 2D [21, 37] que, no limite, converge à superfície  $\mathcal{M}_2^3$ . Daqui em diante, indicamos por  $\mathcal{V}(i)$  os vértices vizinhos ao vértice  $i$  e por  $\mathcal{F}(i)$  as faces que contém  $i$ . Em uma aresta  $\mathbf{e}_{ij} = (i, j) \in \mathcal{E}$ , os ângulos dos seus vértices opostos são dados por  $\theta_{ij}$  e  $\theta_{ji}$ . Por último, denotamos a versão discreta do LBop pela matriz  $L$ .

### 2.5.2 Fórmula da Cotangente

Várias derivações já foram apresentadas para o LBop em malhas triangulares. A maioria delas resulta na mesma expressão, conhecida como *fórmula da cotangente*. Nela, o LBop é aproximado pela matriz

$$L = B^{-1}Q, \quad (2.21)$$

sendo  $B$  a matriz diagonal relacionada com a distribuição dos vértices  $\mathcal{V}$  em  $\mathcal{M}_2^3$ , e  $Q$  uma matriz esparsa e simétrica com elementos

$$q_{ij} = \begin{cases} 0 & (i, j) \notin \mathcal{E}, \\ 0.5 (\cot \theta_{ij} + \cot \theta_{ji}) & (i, j) \in \mathcal{E}, \\ -\sum_{k \in \mathcal{V}(i)} q_{ik} & i = j. \end{cases} \quad (2.22)$$

Originalmente, a fórmula da cotangente foi derivada em [77] com  $B$  igual à matriz identidade  $I$ . Em [22], foi proposta a normalização de cada linha de  $Q$  usando  $b_{ii} = -q_{ii}$ . Somente em [72], a matriz  $B$  foi ajustada de modo a convergir ao LBop contínuo. Para isso, os autores se basearam em propriedades diferenciais e concluíram que  $b_{ii}$  corresponde à área  $A_i$  da região ao redor do vértice  $i$ . Em [21, 37], esta associação entre a matriz  $B$  e a área ao redor de cada vértice foi confirmada utilizando conceitos de cálculo exterior discreto (DEC). Assim, foi provado que o uso correto de  $B$  permite a aproximação do LBop em malhas irregulares, com amostragem não-uniforme. Xu [103] analisou a convergência das derivações anteriores para vários tipos de malhas e em diversas superfícies. Em [104], o mesmo autor propôs uma pequena modificação na área  $A_i$  para melhorar a convergência da matriz  $L$  em esferas. Recentemente, Vallet e Lévy [99] utilizaram métodos de elementos finitos (FEM) lineares para discretizar o LBop. Por fim, em [80], a precisão de  $L$  foi aperfeiçoada utilizando FEM de bases cúbicas.

Outras abordagens definiram  $L$  através de variações da fórmula da cotangente. Em [96, 48], o LBop foi aproximado usando apenas a conectividade de sua malha. Assim, a Eq. 2.21 se reduz à forma matricial  $Q = D - W$  (Seção 2.2), recebendo o nome de LBop combinatório. Em [40], os elementos da matriz  $L$  foram substituídos pelos *mean value weights*. Já em [8], a matriz  $L$  é calculada a partir de uma triangulação de Delaunay intrínseca da malha  $\mathcal{M}$ . Em [58], a Eq. 2.21 foi ajustada por  $0.5(L + L^T)$  para que a matriz final fosse simétrica. Wardetzky et al. [101] compararam as diferentes possibilidades do LBop e demonstraram que é impossível discretizar o LBop gerando uma matriz que seja, ao mesmo tempo, simétrica, de elementos positivos, local e de precisão linear. A Figura 2.3 compara os autovetores obtidos com o LBop combinatório e com a Eq. 2.21.

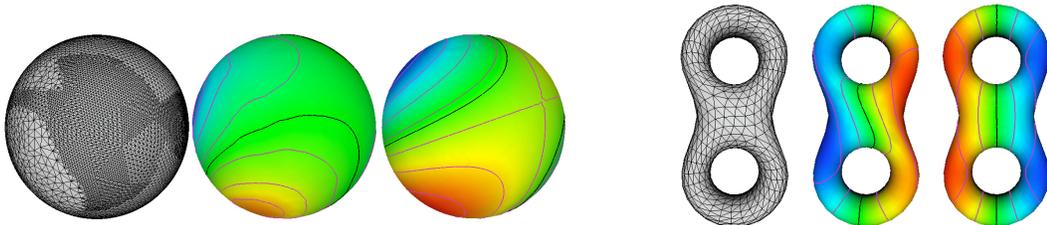


Figura 2.3: Comparação dos autovetores obtidos, respectivamente, com o operador Laplace-Beltrami combinatório e com a fórmula da cotangente (Eq. 2.21).

### 2.5.3 Derivação

Neste trabalho, utilizamos a discretização do LBop apresentada em [21, 99]. Assim, nossa matriz  $L$  segue às Eq. 2.21 e 2.22, com  $b_{ii} = \sum_{t \in \mathcal{F}(i)} A_t/3$ . A seguir, revisamos de maneira simplificada como obter a fórmula da cotangente a partir da decomposição espectral do LBop (Eq. 2.10). Para isso, combinamos o método de elementos finitos (FEM) e o método de volumes finitos (FVM).

Nossa derivação parte da aproximação do espaço de funções  $L^2(\mathcal{M}_2^3)$  pelo espaço  $L^2(\mathcal{M})$ , no qual uma função  $f$  é discretizada pelo vetor  $\mathbf{f} = [f_i]_{i \in \mathcal{V}}$ . Seguindo FVM, interpolamos os valores do vetor  $\mathbf{f}$  através das funções *box*  $\{\delta^i\}_{i \in \mathcal{V}}$ . Estas funções compõem uma base ortogonal de  $L^2(\mathcal{M})$ ,

$$\langle \delta^i, \delta^j \rangle = \begin{cases} 0 & i \neq j, \\ A_i & i = j. \end{cases} \quad (2.23)$$

Assim, dado um ponto  $x \in \mathcal{M}_2^3$ , calculamos  $f(x) \simeq \sum_{i \in \mathcal{V}} f_i \delta^i(x)$ . Consequentemente, o produto interno entre funções  $f, g \in L^2(\mathcal{M}_2^3)$  é aproximado por um produto interno de vetores dependente da matriz  $B$ ,

$$\langle f, g \rangle = \int_{\mathcal{M}_2^3} f(x)g(x)dx \simeq \sum_{i \in \mathcal{V}} f_i g_i A_i = \mathbf{f}^T B \mathbf{g} = \langle \mathbf{f}, \mathbf{g} \rangle_B. \quad (2.24)$$

Fixando um vértice  $i$  e um par de autovalor e autofunção  $(\lambda^j, \phi^j)$ , podemos calcular o produto interno entre  $\delta^i$  e os dois lados da Eq. 2.10,  $\langle \Delta \phi^j, \delta^i \rangle = \lambda^j \langle \phi^j, \delta^i \rangle$ . Repetindo este cálculo para as funções  $\delta^i$  de todos os vértices  $i \in \mathcal{V}$ , obtemos um sistema linear cuja solução corresponde à discretização  $\Phi^j$  da autofunção  $\phi^j \simeq \sum_{i \in \mathcal{V}} \Phi_i^j \delta^i$ ,

$$Q \Phi^j = \lambda^j B \Phi^j. \quad (2.25)$$

Os elementos da matriz  $B$  valem  $b_{ik} = \langle \delta^i, \delta^k \rangle$ . Logo, pela Eq. 2.23, temos que  $B$  é uma matriz diagonal contendo a área ao redor de cada vértice de  $\mathcal{M}$ . Quanto à matriz  $Q$ , os seus elementos valem

$$q_{ik} = \langle \Delta \delta^i, \delta^k \rangle = \int_{\mathcal{M}_2^3} \nabla \delta^i(x)^T \nabla \delta^k(x) dx. \quad (2.26)$$

Como as funções  $\delta^i$  são contínuas de ordem zero, adotamos a abordagem de FEM e aproximamos  $\nabla \delta^i$  em partes constantes utilizando o gradiente das coordenadas baricêntricas do vértice  $i$  em cada triângulo  $t = \{i, k, l\} \in \mathcal{F}(i)$ . As coordenadas baricêntricas,  $\beta_t^i$ , correspondem a funções interpolantes do tipo *hat*. Logo, temos

$$\nabla \delta^i|_t \simeq \nabla \beta_t^i = -\frac{\mathbf{e}_{kl}^\perp}{2A_t}, \quad (2.27)$$

onde  $\mathbf{v}^\perp$  significa que o vetor  $\mathbf{v}$  está rotacionado por  $\pi/2$  no sentido anti-horário sobre o plano tangente de  $t$ . Além disso, lembremos que  $\nabla\beta_t^i + \nabla\beta_t^j + \nabla\beta_t^k = 0$ , pois as coordenadas baricêntricas são uma partição da unidade em  $t$ , e que  $A_t = 0.5\|e_{kl}\|\|e_{li}\|\sin\theta_{ik}$ . Baseado nestes fatos, substituímos a Eq. 2.27 na Eq. 2.26 e obtemos

$$q_{ik} \simeq \sum_{t \in \mathcal{F}} \int_t \left( \nabla\delta^i|_t(x)^T \nabla\delta^k|_t(x) \right) dx \simeq \sum_{t \in \mathcal{F}} \nabla\beta_t^{iT} \nabla\beta_t^k A_t. \quad (2.28)$$

Quando  $i \neq k$ , esta expressão se reduz a

$$q_{ik} \simeq \frac{\mathbf{e}_{kl_1}^T \mathbf{e}_{l_1 i}}{4A_{t_1}} + \frac{\mathbf{e}_{l_2 k}^T \mathbf{e}_{il_2}}{4A_{t_2}} = 0.5(\cot\theta_{ik} + \cot\theta_{ki}). \quad (2.29)$$

E quando  $i = k$ , temos

$$q_{ii} \simeq \sum_{t \in \mathcal{F}(i)} \nabla\beta_t^{iT} \nabla\beta_t^i A_t = \sum_{t=\{i,k,l\} \in \mathcal{F}(i)} \left[ \nabla\beta_t^{iT} (-\nabla\beta_t^k - \nabla\beta_t^l) \right] A_t = - \sum_{k \in \mathcal{V}(i)} q_{ik}. \quad (2.30)$$

Assim, comprovamos a fórmula da cotangente (Eq. 2.22). Por fim, partindo da Eq. 2.25, é fácil verificar que o LBop em  $\mathcal{M}$  é aproximado pela Eq. 2.21.

#### 2.5.4 Problema de Autovalores e Autovetores Generalizado

Os autovalores e autovetores  $(\lambda^i, \Phi^i)$  da matriz  $L$  são determinados através da Eq. 2.25. Este sistema corresponde ao problema de autovalores e autovetores generalizado. Como  $Q$  é uma matriz simétrica e  $B$  é definida positiva, temos garantias numéricas de que  $(\lambda^i, \Phi^i)$  é sempre real. Vale ressaltar, novamente, que os autovetores de  $L$  são  $B$ -ortonormais.

Aproveitando o fato de  $Q$  ser uma matriz esparsa (número de elementos não nulos igual a  $|\mathcal{E}| \in O(|\mathcal{V}|)$ ), o problema de autovalores e autovetores generalizado pode ser resolvido eficientemente através do algoritmo iterativo denominado *método Lanczos*. A biblioteca pública ARPACK<sup>1</sup> possui a implementação deste algoritmo.

O método Lanczos é uma extensão do método de potência para computar vários autovetores em uma única rotina. A idéia central é gerar um vetor aleatório  $\mathbf{v}_0$  e, iterativamente, realizar a multiplicação  $\mathbf{v}_{i+1} = L\mathbf{v}_i$ . Para um número de iterações  $n$  suficientemente grande,  $\mathbf{v}_n$  converge para o autovetor de maior autovalor, em módulo, de  $L$ . Esta convergência, por sua vez, depende diretamente do crescimento do espectro de  $L$ . Logo, para computar os  $k$  primeiros autovetores de  $L$ , a convergência do método Lanczos depende da diferença entre o maior e o  $(k+1)$ -ésimo maior autovalor de  $L$ . A complexidade de tempo do método Lanczos é superlinear em relação ao número de autovalores  $k$  e linear em relação à multiplicação por  $L$ . Quanto à memória, o seu custo é na ordem de  $O(k|\mathcal{V}|)$ .

<sup>1</sup><http://www.caam.rice.edu/software/ARPACK/>

No caso do LBop, estamos interessados principalmente nas informações geométricas mais suaves da superfície, i.e., as informações de baixa frequência. Por isso, precisamos computar os autovetores de menor autovalor. No entanto, como a convergência do método Lanczos depende do espectro de  $L$ , calcular seus menores autovalores requer mais iterações e pode até não convergir caso a diferença entre os menores autovalores seja muito pequena.

Com o intuito de garantir a convergência para os autovetores de baixa frequência e, ao mesmo tempo, tornar este cálculo eficiente, adotamos a estratégia introduzida em [99]. Nela, a matriz  $L$  é decomposta em pequenas bandas através da transformação espectral *shift-invert*. Esta transformação consiste em deslocar o espectro por um valor  $\sigma$  e invertê-lo. Assim, os autovalores próximos de  $\sigma$  se tornam os maiores, em módulo, e mais estáveis autovalores de um novo sistema,

$$(Q - \sigma B)^{-1} B \Phi^i = \frac{1}{\lambda^i - \sigma} \Phi^i. \quad (2.31)$$

Para que esta transformação seja utilizada junto ao método Lanczos, é preciso inverter a matriz  $(Q - \sigma B)$ . Utilizando novamente a esparsidade de  $Q$ , podemos resolver esta inversão com custo  $O(|\mathcal{V}|)$  através da fatoração LU. Assim, a cada iteração do algoritmo, resolvemos um sistema linear  $(Q - \sigma B)\mathbf{x} = \mathbf{y}$ .

Em nossos experimentos, computamos os autovalores e autovetores da matriz  $L$  em bandas de tamanho 50. Além disso, computamos fatorações LU das matrizes  $(Q - \sigma B)$  através da biblioteca SuperLU <sup>2</sup>. A Tabela 2.1 relata o tempo de execução para calcular 200 pares de autovalores e autovetores em malhas de diversos tamanhos. Os testes foram feitos em uma máquina 2.4GHz Intel Core 2 Duo com 2GB RAM.

Modelo	$ \mathcal{V} $	$ \mathcal{F} $	Tempo (s)
bitorus	766	1.5k	1
dinopet	4k	8k	2.7
bumpy	5.5k	11k	6.3
human	8k	16k	7.4
dog	9.5k	19k	9.2
ajax	10k	20k	11.7
bimba	15k	30k	17.7
grid	16k	32k	20.6
armadillo	23k	46k	23.7
elephant	42k	84.5k	46.6
dinosaur	56k	112k	63.9
gargolye	85.5k	171k	100.9

Tabela 2.1: Tempo de execução para computar os 200 primeiros autovalores e os autovetores do operador Laplace-Beltrami em malhas triangulares.

<sup>2</sup><http://crd.lbl.gov/~xiaoye/SuperLU/>

# Capítulo 3

## Aplicações

Neste capítulo, revisamos as recentes aplicações em computação gráfica que se baseiam na análise espectral do operador Laplace-Beltrami de superfícies. Para cada um dos problemas, discutimos as vantagens da abordagem espectral e reproduzimos seus resultados.

### 3.1 Projeto de Filtros

Inspirado em técnicas de processamento de sinais, Vallet e Lévy [99] utilizaram os autovalores e as autofunções do operador Laplace-Beltrami para computar a transformada de Fourier em superfícies. Dado um sinal  $f$  definido na superfície, a sua transformada de Fourier corresponde aos coeficientes  $\{F_i = \langle f, \phi^i \rangle\}$ . Manipulando estes coeficientes, podemos projetar filtros que alteram no domínio da frequência atributos das superfícies, tais como cor e textura. Em especial, Vallet e Lévy [99] aplicaram a transformada de Fourier sobre as coordenadas 3D das superfícies. Com isso, técnicas de suavização baseadas no fluxo de curvatura [96, 22] foram generalizadas. Para ilustrar esta idéia, a primeira linha da Figura 3.1 mostra a remoção de ruídos de uma esfera através de filtros passa-baixa com bandas de diferentes tamanhos. Observe que estes filtros atacam somente as altas frequências, enquanto métodos anteriores, como [96, 22], atenuam todo o espectro, danificando as baixas frequências. Filtros sobre as coordenadas 3D também podem ser usados para remodelar as superfícies. Na segunda linha da Figura 3.1, mostramos alguns exemplos obtidos alterando as frequências baixas e intermediárias de uma superfície.

Em [66], os coeficientes das coordenadas 3D também foram utilizados para esconder registros, como marcas d'água, em malhas 3D com mínima distorção geométrica. Em [82, 83], a transformada de Fourier foi incorporada a métodos variacionais lineares [10], acelerando a deformação interativa de superfícies.

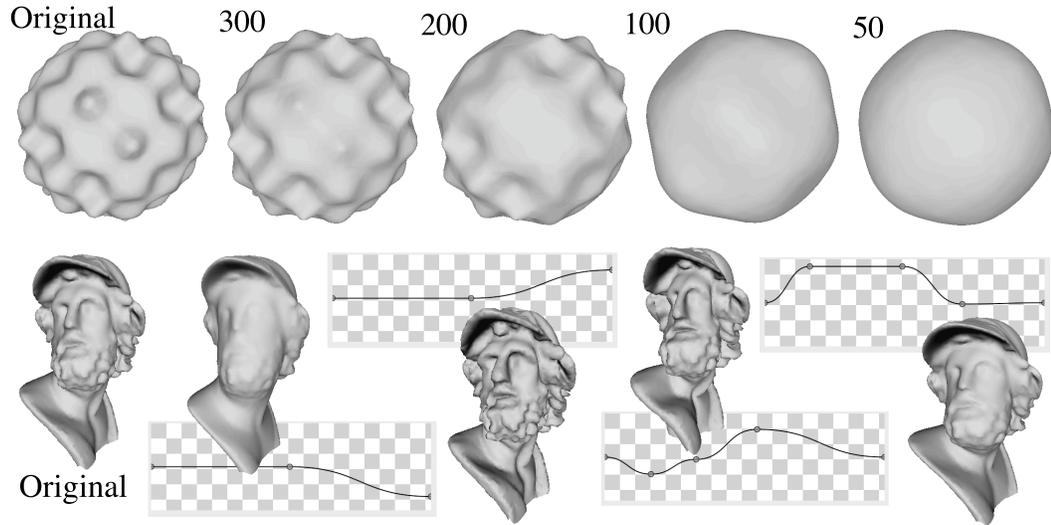


Figura 3.1: Filtros aplicados sobre as coordenadas 3D das superfícies. Na primeira linha, os números indicam quantos coeficientes da transformada de Fourier foram utilizados para reconstruir cada exemplo. Na segunda linha, os gráficos mostram os filtros utilizados.

## 3.2 Descritores Globais e Locais

Conforme discutido na Seção 2.4.3, os autovalores do operador Laplace-Beltrami caracterizam variedades sob isometrias. Motivados por este fato, Reuter et al. [80] utilizaram o espectro do operador Laplace-Beltrami como um descritor global de superfícies, denominando-o como o DNA de formas. Estes descritores permitem com que objetos semelhantes sejam identificados e classificados. A Figura 3.2 exemplifica o uso desses descritores comparando os espectros de seis poses do modelo de um cavalo com os espectros de seis modelos de um gorila. Vale observar que, como o tensor de métrica varia linearmente de acordo com a área do objeto, os autovalores não são invariantes à escala. Logo, para comparar superfícies de tamanhos diferentes, normalizamos os autovalores multiplicando-os pela área das superfícies.

Em [86], os autovalores e as autofunções do operador Laplace-Beltrami foram combinados, compondo um descritor local e invariante a isometrias. Com este fim, esta abordagem definiu para cada ponto  $x \in \mathcal{M}_m^n$  um novo vetor de coordenadas denominado *Global Point Signature (GPS)* e indicado por

$$GPS(x) = \left[ \dots \frac{1}{\sqrt{\lambda^i}} \phi^i(x) \dots \right]. \quad (3.1)$$

Semelhante aos métodos espectrais baseados em matrizes de Gram (Seção 2.2), esta assi-

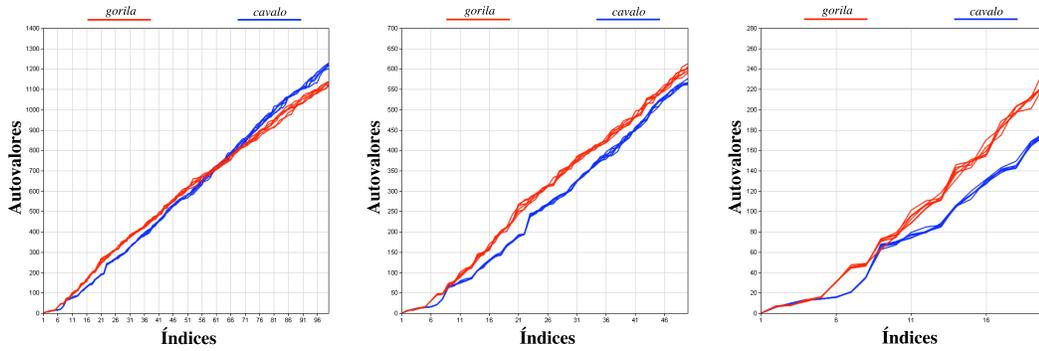


Figura 3.2: Espectro de seis poses do modelo de um cavalo e de um gorila. Os gráficos contém, respectivamente, 100, 50 e 20 autovalores.

natura representa um espaço linear cujo produto interno equivale à afinidade entre pontos indicada pela solução fundamental da equação de Poisson,

$$\mathcal{K}(x, y) = \sum_i \frac{\phi^i(x)\phi^i(y)}{\lambda^i}. \quad (3.2)$$

Dada uma transformação isométrica  $\Upsilon : \mathcal{M}_m^n \rightarrow \mathcal{M}_m^{n'}$ , as coordenadas  $GPS \circ \Upsilon$  não são necessariamente iguais às coordenadas  $GPS$ . Isto acontece porque a transformação  $\Upsilon$  pode causar rotações sobre subespaços de autofunções com mesmo autovalor. No entanto, como rotações são transformações ortogonais, é fácil verificar que

$$\|GPS(x) - GPS(y)\| = \|GPS(\Upsilon(x)) - GPS(\Upsilon(y))\|, \forall x, y \in \mathcal{M}_m^n. \quad (3.3)$$

A partir deste fato, Rustamov [86] definiu um descritor global invariante a isometrias computando o histograma das distâncias Euclidianas entre os vetores  $GPS$ . Por conter as informações locais das autofunções, este descritor consegue distinguir inclusive os raros casos de variedades isoespectrais.

### 3.3 Detecção de Simetrias

Simetrias globais em superfícies correspondem a um caso especial de isometrias em que  $\mathcal{M}_m^{n'} = \mathcal{M}_m^n$ . Em [75], este fato foi analisado utilizando as coordenadas  $GPS$ . Partindo da Eq. 3.3, este trabalho provou que simetrias intrínsecas de uma superfície se transformam em simetrias Euclidianas no espaço definido pelos vetores  $GPS$ . Em outras palavras, isto quer dizer que simetrias globais podem ser detectadas verificando as rotações existentes em cada autofunção do operador Laplace-Beltrami.

Para simplificar esta análise, Ovsjanikov et al. [75] se concentraram somente nas autofunções com autovalores de multiplicidade unitária. Assim, detectar simetrias  $\Upsilon : \mathcal{M}_m^n \rightarrow$

$\mathcal{M}_m^n$  corresponde a identificar o conjunto de autofunções tais que  $\phi^i \circ \Upsilon = -\phi^i$ . Baseando-se novamente na Eq. 3.3, essas autofunções podem ser identificadas e agrupadas em diferentes isometrias  $\Upsilon$  através da minimização da expressão

$$\mathcal{E}(\Upsilon) = \int_{\mathcal{M}_2^3} \min_y \|GPS(x) - GPS(\Upsilon(y))\|^2 dx. \quad (3.4)$$

A Figura 3.3 exemplifica pontos correspondentes obtidos a partir de uma simetria detectada pela Eq. 3.4. As imagens enumeradas ilustram as autofunções utilizadas nos vetores  $GPS$ . Na linha superior, mostramos as autofunções tais que  $\phi^i \circ \Upsilon = -\phi^i$ , enquanto, na linha inferior, temos as autofunções  $\phi^i \circ \Upsilon = \phi^i$ .

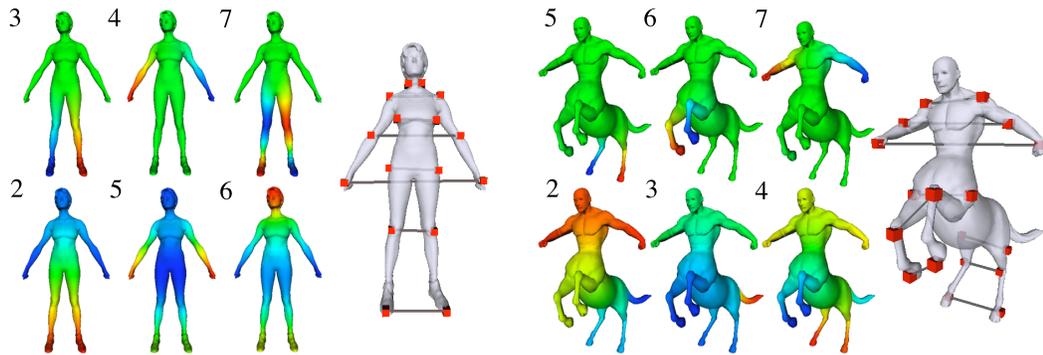


Figura 3.3: Pontos correspondentes em superfícies obtidos através de simetrias Euclidianas dos vetores  $GPS$ . As imagens enumeradas ilustram as autofunções utilizadas nos vetores  $GPS$ . Na linha superior, as simetrias são detectadas por reflexões das autofunções. Na linha inferior, as simetrias preservam os valores das autofunções.

### 3.4 Segmentação

Em [79], as autofunções do operador Laplace-Beltrami foram apontadas como uma biblioteca de ferramentas para a análise de superfícies. Em particular, este trabalho verificou que as linhas nodais de grupos seletos de autofunções segmentam superfícies em regiões significativas. No entanto, não foi apresentando nenhum método para identificar estes grupos de autofunções. A Figura 3.4 ilustra essa constatação, mostrando a segmentação resultante da combinação das linhas nodais de várias autofunções. É importante destacar que, no Capítulo 4, apresentamos um novo método de segmentação que também explora as autofunções do operador Laplace-Beltrami.

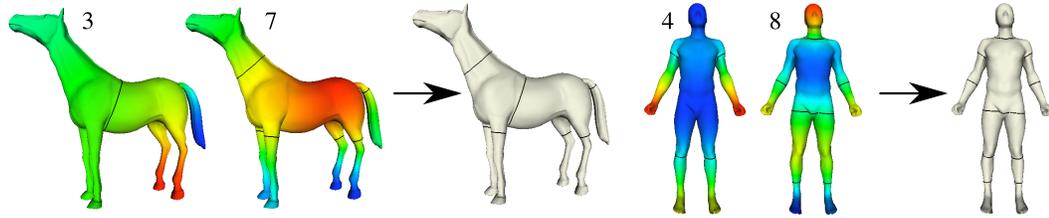


Figura 3.4: Segmentações induzidas pelas linhas nodais das autofunções do operador Laplace-Beltrami. Os números indicam os índices das autofunções utilizadas.

### 3.5 Remalhamento

Na Seção 2.4, vimos que as autofunções do operador Laplace-Beltrami representam funções suaves e que o número de domínios nodais em cada uma delas depende diretamente do seu respectivo autovalor. Motivados por estas propriedades, Dong et al. [24] apresentaram um método espectral para o remalhamento quadrangular de superfícies. Esta técnica utiliza uma autofunção pré-selecionada do operador Laplace-Beltrami para construir um complexo Morse-Smale interligando seus pontos críticos. Como os pontos críticos são uniformemente espaçados, as células desse complexo se aproximam a quadrados. Além disso, o número de células pode ser controlado escolhendo autofunções de diferentes autovalores. Assim, cada célula pode ser reamostrada, gerando uma nova malha quadrangular. A Figura 3.5 mostra exemplos de autofunções do operador Laplace-Beltrami e uma estimativa inicial de seus complexos Morse-Smale.

Apesar da simplicidade, essa técnica de remalhamento segue apenas informações intrínsecas da superfície. Em [42], foi apresentada uma extensão desse método para que o usuário possa orientar as arestas da nova malha. Para isso, o cálculo das autofunções é integrado a um conjunto de restrições e o complexo Morse-Smale é construído a partir do resultado da otimização deste novo sistema.

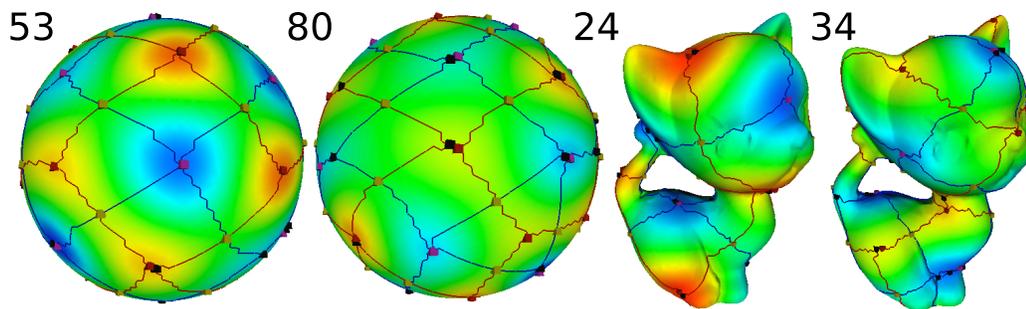


Figura 3.5: Complexos Morse-Smale construídos a partir de autofunções do operador Laplace-Beltrami. Os números indicam os índices das autofunções utilizadas.

## Capítulo 4

# Segmentação Hierárquica de Objetos Articuláveis

A seguir, apresentamos a aplicação da análise espectral do operador Laplace-Beltrami no problema de segmentação semântica de objetos articuláveis. Utilizamos os autovalores e as autofunções do operador Laplace-Beltrami em superfícies para introduzir dois novos conceitos essenciais para o desenvolvimento da nossa técnica. Primeiro, definimos a família de métricas multi-escala denominada distância de difusão. Em seguida, introduzimos uma nova estrutura medial capaz de identificar segmentos significativos. Nosso método resulta em uma hierarquia de segmentos ordenados pelo seu nível de detalhe, consistentes a deformações e numericamente robusto a ruído, amostragens irregulares e pequenos eventos topológicos. O trabalho descrito neste capítulo resultou no artigo [19].

### 4.1 Motivação

Articulação de personagens é uma tarefa fundamental e muito custosa em sistemas de animação. Métodos de segmentação, por sua vez, podem assistir tais tarefas orientando a construção de esqueletos que controlem a animação do modelo 3D. Isto é possível pois as técnicas de segmentação identificam regiões significativas e articuláveis. Além disso, a segmentação também pode representar diferentes níveis de detalhe da superfície. Enquanto nos níveis globais, os segmentos indicam regiões funcionais, como braços e pernas, os segmentos de níveis locais correspondem a componentes rígidos, reproduzindo os ossos de um esqueleto.

Com o intuito de assistir técnicas de animação, introduzimos neste capítulo um novo método de segmentação para objetos articuláveis. Nossa abordagem recupera articulações organizando-as em uma estrutura hierárquica baseada em duas ideias centrais. Primeiro, investigamos a família de métricas multi-escala denominada *distância de difusão* [17]. A distância de difusão apresenta propriedades interessantes, como robustez a ruído e a

eventos topológicos, destaque de regiões côncavas e invariância a deformações isométricas. Em seguida, apresentamos uma generalização das estruturas mediais [74]. Em cada nível de detalhe, decompos os modelos computando uma bijeção entre as estruturas mediais e os segmentos.

## 4.2 Trabalhos Relacionados

Segmentação de malhas é uma ferramenta muito usada em aplicações gráficas como, por exemplo, simplificação, parametrização e classificação de formas. Revisões sobre tais métodos estão disponíveis em [91, 3, 92]. Nossa técnica lida, especificamente, com o problema de segmentação de objetos articuláveis em regiões significativas.

Vários métodos de segmentação são baseados em estudos de cognição. Pesquisas em cognição [38, 39] relatam que a percepção humana separa uma forma em partes salientes, ajustando os cortes em regiões côncavas. Para reproduzir tais resultados, alguns métodos incorporam medidas de concavidade em técnicas de clusterização [68, 94, 76, 64]. Em particular, Katz e Tal [50] extraíram esqueletos utilizando uma clusterização *fuzzy* com cortes de grafos e uma métrica que combina distâncias geodésicas e ângulos diedrais. Lee et al. [57] também exploraram estudos de cognição e apresentaram um detector semi-automático de contornos salientes em modelos 3D. Para isso, este método aplica a técnica *geometric-snakes* sobre os pontos com curvatura mínima.

Atributos locais, como ângulos diedrais e curvaturas, são sensíveis a ruído e à resolução da malha. Consequentemente, os métodos anteriores sofrem de instabilidade numérica. A nossa abordagem, por outro lado, reproduz os resultados de cognição com robustez numérica graças ao uso da distância de difusão. Por construção, a distância de difusão enfatiza as concavidades dos modelos 3D, medindo a taxa de caminhos Brownianos existentes entre pontos da superfície (Seção 4.3.1). Este esquema de integração também ameniza a interferência de ruído e amostragem.

Alguns métodos de segmentação identificam regiões significativas através da detecção de pontos característicos. Hilaga et al. [35] definiram um ponto característico como o máximo local da função *Distância Geodésica Média (AGD)*. Geometricamente, a função *AGD* mede o isolamento de cada ponto da superfície. Em [49], os pontos característicos da função *AGD* foram usados como âncoras em um procedimento de reflexão esférica para, assim, extrair os componentes centrais de modelos 3D. Em [106], os segmentos foram decompostos através da análise da variação da área ao redor de cada ponto característico. Zhang e Liu [107, 65] computaram segmentos recursivamente através de uma busca linear guiada por métricas de saliências e partindo de pontos característicos. Em [107], os pontos característicos foram estimados aproximando a função *AGD* através de um pequeno conjunto de faces de referência. Em [65], a seleção dos pontos foi aperfeiçoada através da análise do contorno da projeção espectral da superfície.

Apesar do fato dos pontos característicos identificarem segmentos, um segmento pode não ser associado a um ponto característico. Logo, métodos baseados nestes pontos podem falhar, criando uma hierarquia de detalhes incompleta. Nós evitamos estes problemas introduzindo uma nova estrutura para identificar os segmentos em cada nível de detalhe. Nossa estrutura é uma generalização das estruturas mediais propostas em [74]. Originalmente, as estruturas mediais foram computadas como curvas fechadas em regiões tubulares. Aqui, definimos uma estrutura medial como o conjunto de pontos com o menor valor médio da distância de difusão (Seção 4.3.2).

Técnicas mais recentes aproximam segmentos significativos por componentes convexos. Em [61], uma abordagem iterativa é proposta gerando segmentações hierárquicas e esqueletos em várias resoluções através do método de decomposição convexa aproximada [60]. Kreavoy et al. [52] utilizaram segmentação para criar um sistema de modelagem a partir da combinação de componentes de diferentes objetos. Os modelos são decompostos em uma hierarquia de componentes quase convexos através do método descrito em [53]. Como segmentos convexos aproximam componentes rígidos, a hierarquia de segmentos é construída a partir de elementos locais. Isto faz necessário um passo extra para juntar segmentos e, assim, recuperar regiões mais globais. A nossa abordagem contrapõe-se a isso computando uma hierarquia de detalhes por demanda. Começamos com elementos globais e, de acordo com a aplicação, geramos elementos menores.

Métodos baseados em exemplos [73, 1, 44, 89] aprendem esqueletos a partir de um conjunto de poses de referência. Estes métodos decompõem todas as poses simultaneamente, garantindo resultados consistentes. Nosso algoritmo, por sua vez, computa segmentos consistentes analisando apenas uma pose do modelo. Esta consistência ocorre porque a distância de difusão é invariante a deformações isométricas. Como já discutido na Seção 3.2, Rustamov [86] utilizou uma variação da distância de difusão como descritor de formas e, combinado com a clusterização *k-means*, também gerou segmentos consistentes. Porém, enquanto este descritor representa somente uma escala de detalhe da superfície, nós exploramos todo o espaço de escala oferecido pela família de distâncias de difusão.

Por último, Shapira et al. [93] apresentaram um algoritmo de segmentação baseado na função *Diâmetro de Forma (SDF)*. Esta função associa estimativas de volumes locais com os pontos da superfície. Assim, seus valores são robustos a deformações e permitem segmentações consistentes. Os segmentos são computados agrupando os valores do histograma da *SDF* através de um modelo de mistura de Gaussianas. O número de Gaussianas utilizado indica o nível da hierarquia, ao invés de se referir ao número de segmentos. Conseqüentemente, diferentes níveis contêm segmentos independentes e, por isso, podem formar uma hierarquia que não corresponda, topologicamente, a uma árvore.



Figura 4.1: Resultados do nosso algoritmo de segmentação.

## 4.3 Teoria

A seguir, apresentamos os dois conceitos essenciais para o nosso método de segmentação. Primeiro, revisamos a família de distância de difusão e discutimos algumas propriedades relevantes. Depois, introduzimos a definição de estruturas mediais e descrevemos uma bijeção entre segmentos e estruturas mediais.

### 4.3.1 Distância de Difusão

A distância de difusão foi apresentada inicialmente para sistemas dinâmicos estocásticos e aplicada para o problema de redução de dimensionalidade e parametrização de dados [17]. Neste contexto, dada uma cadeia de Markov definida em um conjunto de dados  $\mathcal{X}$ , a distância de difusão entre dois pontos  $x$  e  $y$  no tempo  $t$  é definida por:

$$\mathcal{D}_t^2(x, y) = \int_{\mathcal{X}} (p^t(x, z) - p^t(y, z))^2 d\mu(z), \quad (4.1)$$

onde  $p^t(\cdot, \cdot)$  é a probabilidade de se transitar entre os dois pontos em, no máximo,  $t$  passos, e  $\mu(\cdot)$  é a densidade dos pontos em  $\mathcal{X}$ .

Deste modo, a distância de difusão indica a conectividade entre os pontos de  $\mathcal{X}$ . Esta medida é pequena caso haja bastante caminhos aleatórios de tempo menor ou igual a  $t$ . Se o número de caminhos aleatórios for pequeno, a distância se torna grande.

Seguindo o Capítulo 2, nosso domínio é restrito a um conjunto de dados correspondente a variedades compactas e contínuas  $\mathcal{M}_m^n$ . Na nossa aplicação, estamos interessados

especialmente em superfícies  $\mathcal{M}_2^3$ . Em processos estocásticos [4], sabe-se que caminhos aleatórios em  $\mathcal{M}_m^n$  equivalem, no limite, a caminhos Brownianos e a função de probabilidade  $p^t(\cdot, \cdot)$  converge no *kernel* de calor  $\mathcal{K}(t, \cdot, \cdot)$  de  $\mathcal{M}_m^n$ . Logo, a distância de difusão entre pontos de  $\mathcal{M}_m^n$  mede a densidade de caminhos Brownianos na escala de tempo  $t$ ,

$$\mathcal{D}_t^2(x, y) = \int_{\mathcal{M}_m^n} (\mathcal{K}(t, x, z) - \mathcal{K}(t, y, z))^2 dz. \quad (4.2)$$

Este esquema de integração oferece propriedades desejáveis para o problema de segmentação de superfícies. Primeiramente, suas medidas são robustas a ruídos e a pequenos circuitos topológicos. Opondo-se a distâncias geodésicas, a distância de difusão também apresenta um crescimento rápido em regiões em que a área é comprimida. Deste modo, concavidades em  $\mathcal{M}_2^3$  são enfatizadas (Figura 4.2).

Outra vantagem da distância de difusão é a análise multi-escala proporcionada pelo parâmetro  $t$ , expandindo a distância de difusão como uma família de métricas. Em intervalos de tempo longos, a distância de difusão avalia estruturas globais de  $\mathcal{M}_2^3$ , enquanto, em intervalos mais curtos, as medidas se tornam locais e detectam pequenos detalhes.

Revisitando a Eq. 2.18, temos que o *kernel* de calor  $\mathcal{K}(t, \cdot, \cdot)$  é determinado pelos autovalores e autofunções  $(\lambda^i, \phi^i)$  do operador Laplace-Beltrami. Como este operador é invariante a isometrias, garantimos que, ao deformar uma superfície  $\mathcal{M}_2^3$  mantendo suas distâncias geodésicas, os pares  $(\lambda^i, \phi^i)$ , o *kernel* de calor e a família de distâncias de difusão são preservados. Ou seja, a distância de difusão é invariante a isometrias. Esta propriedade possibilita que nosso método segmente diferentes poses de um objeto de maneira consistente.

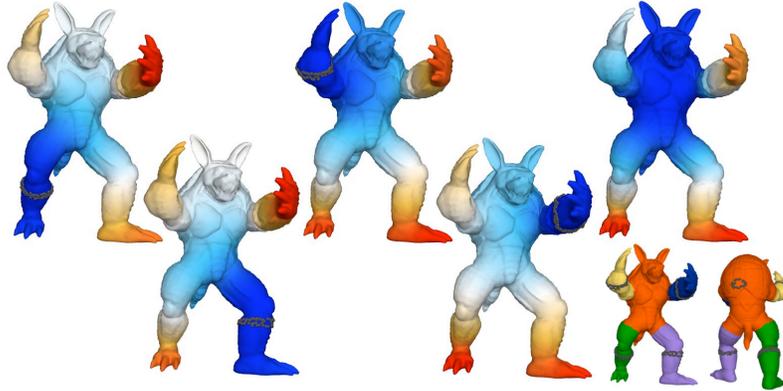


Figura 4.2: Propagação de frentes a partir de cada estrutura medial. Esferas indicam os pontos das estruturas mediais. As cores representam os isovalores da distância de difusão entre as estruturas mediais e os pontos da superfície, variando de azul para vermelho. Note que as cores mudam rapidamente ao passar pelas concavidades.

Substituindo a Eq 2.18 na Eq. 4.2, podemos reescrever a distância de difusão em termos dos *mapas de difusão* [17],

$$\Psi_t(x) = \left[ \dots e^{-\lambda^i t} \phi^i(x) \dots \right]. \quad (4.3)$$

Assim, a distância de difusão equivale a uma distância Euclidiana entre vetores de dimensão infinita,

$$\begin{aligned} \mathcal{D}_t^2(x, y) &= \int_{\mathcal{M}_m^n} \left[ \left( \sum_i e^{-\lambda^i t} \phi^i(z) \phi^i(x) \right) - \left( \sum_i e^{-\lambda^i t} \phi^i(z) \phi^i(y) \right) \right]^2 dz = \\ &= \int_{\mathcal{M}_m^n} \left[ \sum_i e^{-\lambda^i t} \phi^i(z) (\phi^i(x) - \phi^i(y)) \right]^2 dz = \\ &= \int_{\mathcal{M}_m^n} \sum_{i,j} \left[ e^{-(\lambda^i + \lambda^j)t} \phi^i(z) \phi^j(z) (\phi^i(x) - \phi^i(y)) (\phi^j(x) - \phi^j(y)) \right] dz = \\ &= \sum_{i,j} e^{-(\lambda^i + \lambda^j)t} (\phi^i(x) - \phi^i(y)) (\phi^j(x) - \phi^j(y)) \left( \int_{\mathcal{M}_m^n} \phi^i(z) \phi^j(z) dz \right) = \\ &= \sum_{i,j} e^{-(\lambda^i + \lambda^j)t} (\phi^i(x) - \phi^i(y)) (\phi^j(x) - \phi^j(y)) \langle \phi^i, \phi^j \rangle = \\ &= \sum_i e^{-2\lambda^i t} (\phi^i(x) - \phi^i(y))^2 = \|\Psi_t(x) - \Psi_t(y)\|^2. \end{aligned} \quad (4.4)$$

Como os autovalores  $\{\lambda^i\}$  aumentam rapidamente, podemos truncar os mapas de difusão e aproximar a distância de difusão usando apenas os primeiros termos da soma. Daqui em diante, referimos por  $\Psi_t(\cdot)$  ao vetor com  $e^{-\lambda^i t} > 0.1$ .

### 4.3.2 Estrutura Medial

Agora, discutimos como identificar os componentes significativos de uma região  $\mathcal{R} \subset \mathcal{M}$ . Para isso, usamos a distância de difusão para formular uma bijeção entre as estruturas mediais e os segmentos de  $\mathcal{R}$ . Em um sentido da bijeção, computamos as estruturas mediais a partir dos segmentos de  $\mathcal{R}$ . Já no sentido oposto, computamos os segmentos a partir das estruturas mediais. No restante desta seção, indicamos por  $t_{\mathcal{R}}$  a escala de tempo da distância de difusão suficiente para destacar as concavidades globais de  $\mathcal{R}$ , filtrando pequenos detalhes.

#### Segmentos $\rightarrow$ Estruturas Mediais

Dado um segmento  $\mathcal{R}_i \subset \mathcal{R}$ , caracterizamos  $\mathcal{R}_i$  pela sua estrutura medial. Em [74], estruturas mediais foram introduzidas como curvas fechadas equidistantes das bordas de regiões tubulares. Aqui, definimos a estrutura medial como o conjunto de pontos distantes das concavidades de seu respectivo segmento. Note que a estrutura medial difere do eixo

medial, pois o primeiro se refere a pontos sobre a superfície do segmento, enquanto o segundo é composto por pontos no interior do volume de  $\mathcal{R}_i$ .

Para computar a estrutura medial de  $\mathcal{R}_i$ , propomos a avaliação da função *Distância de Difusão Média* (*ADD*):

$$ADD(x) = \frac{1}{A_{\mathcal{R}_i}} \int_{\mathcal{R}_i} \mathcal{D}_{t_{\mathcal{R}}}^2(x, y) dy. \quad (4.5)$$

Semelhante à função *AGD* [35], a função *ADD* apresenta valores altos para os pontos de borda e das extremidades, e valores baixos para pontos no centro do segmento. Como a *ADD* herda as propriedades da distância de difusão, os pontos com baixo valor correspondem a pontos distantes das concavidades (Figura 4.3a). Portanto, definimos a estrutura medial  $\mathcal{S}_i$  do segmento  $\mathcal{R}_i$  por:

$$\mathcal{S}_i = \left\{ x \in \mathcal{R}_i : ADD(x) = \min_{y \in \mathcal{R}_i} ADD(y) \right\}. \quad (4.6)$$

### Estruturas Mediais $\rightarrow$ Segmentos

Dadas as estruturas mediais  $\{\mathcal{S}_i\}$  de  $\mathcal{R}$ , aplicamos uma competição de frentes para recuperar os segmentos  $\{\mathcal{R}_i\}$ . Cada frente  $\mathcal{F}_i$  é propagada de acordo com o crescimento dos isovalores  $d$  da distância de difusão relativa a  $\mathcal{S}_i$ ,

$$\mathcal{F}_i(d) = \left\{ x \in \mathcal{R} : \min_{y \in \mathcal{S}_i} \mathcal{D}_{t_{\mathcal{R}}}^2(x, y) = d \right\}. \quad (4.7)$$

A competição começa a partir de todas as estruturas mediais,  $\mathcal{F}_i(0)$ , e acaba quando as frentes se encontram.

Durante a propagação, construímos cada segmento  $\mathcal{R}_i$  como a região percorrida pela respectiva frente  $\mathcal{F}_i$ . Como as concavidades em  $t_{\mathcal{R}}$  são distantes das estruturas mediais, as fronteiras dos segmentos tendem a se posicionar nas concavidades. A Figura 4.2 ilustra com cores a propagação de frentes a partir de várias estruturas mediais. Observe que os isovalores  $d$  aumentam rapidamente nas concavidades mais relevantes do modelo.

## 4.4 Algoritmo

Baseado nos conceitos da Seção 4.3, apresentamos nosso algoritmo de segmentação para malhas triangulares. Aqui, usamos as mesmas notações da Seção 2.5 e aproximamos funções em  $\mathcal{M}_2^3$  por vetores  $\mathbf{f}$  em  $\mathcal{M}$ , onde  $f_i$  é o valor de  $f$  no vértice  $i$ . Para os pontos dentro dos triângulos, aproximamos  $f$  utilizando interpolação linear e coordenadas baricêntricas.

Começamos nosso algoritmo computando, em uma etapa de pré-processamento, os pares de autovalores e autovetores  $(\lambda^i, \Phi^i)$  do operador Laplace-Beltrami de  $\mathcal{M}$  (Eq. 2.10).

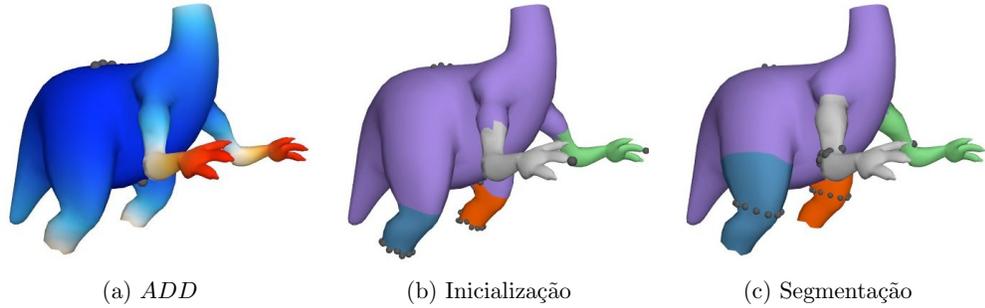


Figura 4.3: Esferas representam os pontos das estruturas mediais. (a) A função  $ADD$  da região e sua estrutura medial. As cores indicam os valores da função  $ADD$ , crescendo de azul para vermelho. (b) As estruturas mediais e os segmentos iniciais. (c) As estruturas mediais e os segmentos finais.

Para isso, seguimos novamente a Seção 2.5 e utilizamos o método numérico proposto em [99].

Dada uma região  $\mathcal{R} \subset \mathcal{M}$ , primeiro selecionamos a escala de tempo  $t_{\mathcal{R}}$ . Exploramos o fato de que áreas maiores tendem a ter estruturas globais e definimos

$$t_{\mathcal{R}} = \frac{1}{2\lambda^2} \frac{A_{\mathcal{R}}}{A_{\mathcal{M}}}, \quad (4.8)$$

onde  $A_{\mathcal{R}}$ ,  $A_{\mathcal{M}}$  são as áreas de  $\mathcal{R}$  e  $\mathcal{M}$ , respectivamente. A constante  $1/2\lambda^2$  faz com que a seleção do tempo seja invariante ao tamanho de  $\mathcal{M}$ . Usamos o segundo autovalor  $\lambda^2$  pois  $\lambda^1 = 0$ . Em seguida, computamos os mapas de difusão  $\Psi_{t_{\mathcal{R}}}$  para todos os vértices de  $\mathcal{R}$ .

Já para decompor a região  $\mathcal{R}$ , computamos a bijeção entre as suas estruturas mediais e seus segmentos. Para isso, propomos um algoritmo iterativo com uma fase de inicialização e uma rotina principal.

Durante a inicialização, computamos o número de segmentos em  $\mathcal{R}$  e uma estimativa inicial das estruturas mediais  $\{\mathcal{S}_i\}$  (Figura 4.3b). Começamos atribuindo a estrutura



Figura 4.4: Diferentes cortes da hierarquia de segmentos dos modelos *raptor* e *dinopet*.

medial de  $\mathcal{R}$ ,  $\mathcal{S}_{\mathcal{R}}$ , como a primeira estrutura medial,  $\mathcal{S}_1$ . Recursivamente, selecionamos o ponto mais distante  $v \in \mathcal{R}$  das estruturas mediais anteriores. Se a distância de  $v$  for maior que um limiar, adicionamos  $v$  como uma nova estrutura medial, caso contrário, interrompemos a inicialização. Quando o vértice  $v$  está em uma borda, estendemos a estrutura medial sobre a borda. Em nossos experimentos, fixamos o limiar para  $\kappa \bar{\mathcal{D}}$ , onde  $\kappa \in [1, 2]$  e  $\bar{\mathcal{D}}$  é a média da distância de difusão entre os pontos de  $\mathcal{R}$  e  $\mathcal{S}_{\mathcal{R}}$ ,

$$\begin{aligned} \bar{\mathcal{D}} &= \frac{1}{A_{\mathcal{R}}} \int_{\mathcal{R}} (\min_{u \in \mathcal{S}_{\mathcal{R}}} \mathcal{D}_t^2(x, u)) dx \simeq \\ &\simeq \frac{1}{A_{\mathcal{R}}} \left[ \sum_{v \in \mathcal{R}} (\min_{u \in \mathcal{S}_{\mathcal{R}}} \mathcal{D}_t^2(v, u)) A_v + \right. \\ &+ \sum_{f \in \mathcal{R}} \frac{A_f}{6} \langle \Psi_t(f_3) - \Psi_t(f_1), \Psi_t(f_3) - \Psi_t(f_2) \rangle - \\ &\left. - \sum_{f \in \mathcal{R}} \frac{A_f}{6} (\|\Psi_t(f_3) - \Psi_t(f_1)\|^2 + \|\Psi_t(f_3) - \Psi_t(f_2)\|^2) \right]. \end{aligned} \quad (4.9)$$

Na rotina principal, refinamos as estruturas mediais  $\{\mathcal{S}_i\}$  iterativamente até que a bijeção com os segmentos  $\{\mathcal{R}_i\}$  convirja (Figura 4.3c). Em cada iteração, primeiro competimos as frentes a partir das estruturas mediais para compor novos segmentos. Em seguida, usamos estes segmentos para atualizar as estruturas mediais. O algoritmo termina quando a porcentagem de vértices de  $\mathcal{R}$  mudando de segmento é inferior a uma tolerância (0 – 5%).

Para competir as frentes, criamos um *heap*  $H$  contendo todos os pares  $(f, i)$ , onde  $f$  é uma face candidata a ser percorrida pela frente  $\mathcal{F}_i$ . Estes pares são ordenados em  $H$  pela distância máxima entre os vértices de  $f$  e a estrutura medial  $\mathcal{S}_i$ ,

$$\max_{v \in f} \left( \min_{u \in \mathcal{S}_i} \mathcal{D}_{t_{\mathcal{R}}}^2(v, u) \right). \quad (4.10)$$

Inicializamos  $H$  com todas as faces adjacentes aos vértices das estruturas mediais atuais  $\{\mathcal{S}_i\}$ . A cada passo, removemos o par  $(f, i)$  com a menor distância. Se  $f$  não estiver em nenhum segmento, adicionamo-no em  $\mathcal{R}_i$ , senão, descartamo-no. Depois, atualizamos  $H$  com as faces vizinhas de  $f$  em  $\mathcal{R}$ . A competição termina quando o *heap* ficar vazio.

Para atualizar as estruturas mediais, computamos os vértices com valor mínimo da função *ADD* em cada segmento  $\mathcal{R}_i$ ,

$$\begin{aligned} ADD(x) &= \|\Psi_t(x)\|^2 - 2 \left\langle \Psi_t(x), \frac{\int_{\mathcal{R}} \Psi_t(y) dy}{A_{\mathcal{R}}} \right\rangle + \frac{\int_{\mathcal{R}} \|\Psi_t(y)\|^2 dy}{A_{\mathcal{R}}} \simeq \\ &\simeq \|\Psi_t(x)\|^2 - 2 \left\langle \Psi_t(x), \frac{\sum_{v \in \mathcal{R}} \Psi_t(v) A_v}{A_{\mathcal{R}}} \right\rangle + Q_{\mathcal{R}}, \end{aligned} \quad (4.11)$$

onde

$$Q_{\mathcal{R}} = \frac{1}{6A_{\mathcal{R}}} \sum_{f \in \mathcal{R}} A_f \left( \sum_i \|\Psi_t(f_i)\|^2 + \langle \Psi_t(f_i), \Psi_t(f_{i+1 \bmod 3}) \rangle \right). \quad (4.12)$$

Como a malha pode ter resoluções irregulares, as estruturas mediais podem ser subamostradas. Superamos estes casos adicionando uma tolerância  $\varepsilon$  ao redor do valor mínimo de  $ADD$ . Esta tolerância deve ser próxima de zero e adaptada à forma de  $\mathcal{R}_i$ . Neste trabalho, utilizamos  $\varepsilon = 0.01\overline{ADD}$ , onde  $\overline{ADD}$  é o valor médio da função  $ADD$  em  $\mathcal{R}_i$ ,

$$\overline{ADD} = \frac{1}{A_{\mathcal{R}}} \int_{\mathcal{R}} ADD(x) dx \simeq 2 \left( Q_{\mathcal{R}} - \left\| \frac{\sum_{v \in \mathcal{R}} \Psi_t(v) A_v}{A_{\mathcal{R}}} \right\|^2 \right). \quad (4.13)$$

Aplicamos nosso algoritmo recursivamente, começando com  $\mathcal{R} = \mathcal{M}$ . Assim, os segmentos resultantes compõem uma árvore hierárquica. Esta hierarquia também revela uma ordenação global/local dos segmentos. Por fim, a segmentação final corresponde a qualquer corte desta árvore (Figura 4.4). O Algoritmo 1 resume o nosso método.

## 4.5 Resultados

Testamos nosso método com uma diversidade de modelos e poses. A Figura 4.1 mostra uma coleção de resultados. Em cada exemplo, indicamos os segmentos com uma paleta de cores diferente.

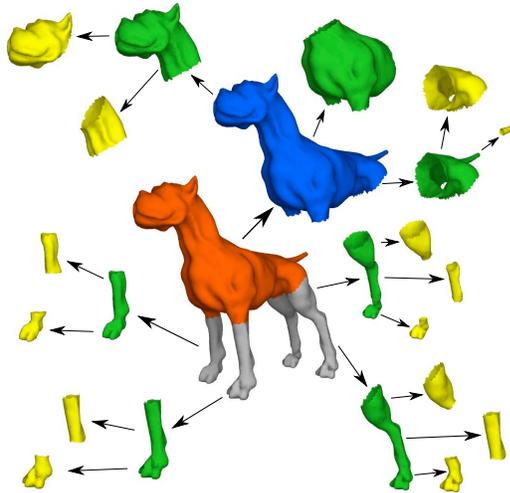


Figura 4.5: A árvore hierárquica de segmentos do modelo *dog*. Cada cor indica um nível de detalhe diferente.

Na Figura 4.5, ilustramos um exemplo da hierarquia de segmentos. Note que os segmentos reproduzem as partes salientes, como sugere os estudos de cognição. Os primeiros segmentos correspondem a regiões funcionais (em verde). Já os segmentos mais refinados correspondem a componentes rígidos (em amarelo). Os segmentos também revelam a estrutura do modelo e, assim, possibilitam a combinação de diferentes níveis de detalhe. Na Figura 4.4, apresentamos possíveis cortes da hierarquia de segmentos dos modelos

```

Input: malha triangular  $\mathcal{M}$ 
Output: segmentos  $\{\mathcal{S}_i\}$ 

1 compute os primeiros pares  $(\lambda^i, \Phi^i)$  da matriz  $L$  (Eq 2.21)
2  $\mathcal{R} = \mathcal{M}$ 
3 begin
4   selecione  $t_{\mathcal{R}}$  (Eq. 4.8)
5   compute  $\Psi_{t_{\mathcal{R}}}$  em  $\mathcal{R}$  (Eq. 4.3)
   // Inicialização
6    $\mathcal{S}_1 = \mathcal{S}_{\mathcal{R}}$ 
7    $v =$  vértice mais distante de  $\mathcal{S}_1$ 
8   while  $dist(v) > \kappa \bar{D}$  do
9     if  $v \in \partial \mathcal{R}$  then
10      |  $\mathcal{S}_i =$  borda que contém  $v$ 
11     else
12      |  $\mathcal{S}_i = v$ 
13     |  $v =$  vértice mais distante de  $\{\mathcal{S}_j\}_{j \leq i}$ 

   // Rotina Principal
14  while  $\{(\mathcal{R}_i, \mathcal{S}_i)\}$  não convergem do
   // Estruturas Mediais  $\rightarrow$  Segmentos
15  inicializa heap  $H$  com faces vizinhas de  $\{\mathcal{S}_i\}$ 
16  while  $H \neq \emptyset$  do
17    |  $(f, i) =$  topo de  $H$ 
18    | adicione  $f$  em  $\mathcal{R}_i$ 
19    | insira em  $H$  faces vizinhas de  $f$ 

   // Segmentos  $\rightarrow$  Estruturas Mediais
20  foreach  $\mathcal{R}_i \subset \mathcal{R}$  do
21    | compute  $ADD$  em  $\mathcal{R}_i$ 
22    | foreach  $v \in \mathcal{R}_i$  do
23    | | if  $(ADD(v) \leq ADD_{min} + \varepsilon)$  then adicione  $v$  em  $\mathcal{S}_i$ 

24  foreach  $\mathcal{R}_i \subset \mathcal{R}$  do
25    |  $\mathcal{M} = \mathcal{R}_i$ 
26    | goto linha 2
27 end

```

**Algoritmo 1:** Pseudocódigo do nosso método de segmentação.

*raptor* e *dinopet*. Em particular no modelo *dinopet*, particionamos os braços, as pernas e a cabeça na primeira imagem, dividimos apenas o seu corpo na segunda imagem e, na terceira imagem, identificamos inclusive os dedos dos pés e das mãos.

A Figura 4.6 ilustra a consistência dos nossos resultados para uma variedade de deformações em três modelos distintos. Na Figura 4.7, demonstramos que nossos segmentos também são consistentes em escalas sucessivas. É importante comentar que o nosso método gerou resultados consistentes, sendo que cada pose tem uma malha diferente.

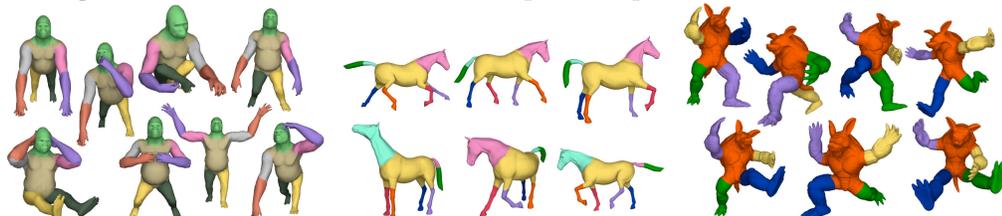


Figura 4.6: Várias poses e os segmentos consistentes gerados pelo nosso método.

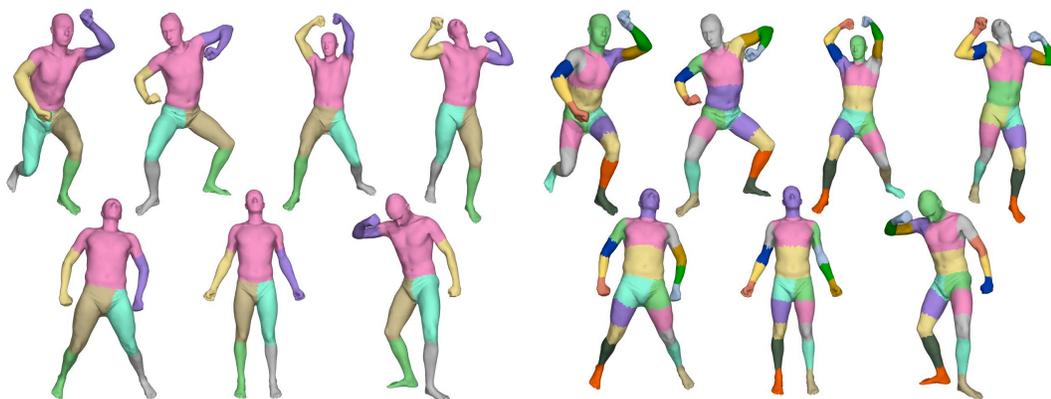


Figura 4.7: Segmentações consistentes em escalas sucessivas de sete poses de um modelo humano escaneado [1]. As imagens mostram os segmentos do primeiro e do segundo nível de detalhe, respectivamente.

Na Figura 4.8(a-c), mostramos a robustez a ruído do nosso algoritmo. Perturbamos os vértices do modelo *cat* com ruídos moderados e exagerados e conseguimos resultados similares à segmentação do modelo original. Na Figura 4.8d, mostramos que o nosso método também é robusto a pequenos circuitos topológicos, separando as pernas do modelo *homer*. Entretanto, a nossa técnica falha quando o evento topológico é grande, conforme indicado na Figura 4.8e. Neste caso, o evento altera a distância de difusão e, por isso, a segmentação identifica apenas os dedos remanescentes da mão.

A Figura 4.9 ilustra esqueletos extraídos a partir das segmentações. Os esqueletos são compostos por arestas ligando os centros de massa dos segmentos (esferas azuis) com suas articulações adjacentes (esferas cinzas).

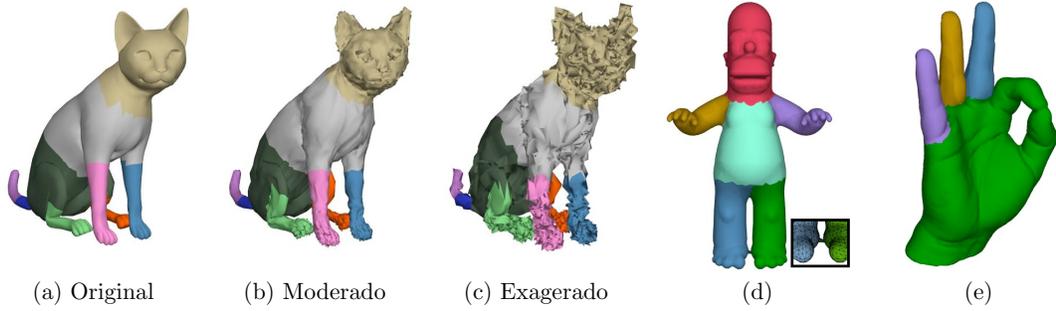


Figura 4.8: (a-c) Nosso método é robusto a ruído. As imagens comparam as segmentações do modelo original com duas versões modificadas por ruídos moderados e exagerados. (d) Nosso método também é robusto a pequenos circuitos topológicos. (e) Entretanto, falha com eventos grandes.

Por fim, a Tabela 4.1 reporta o tempo de execução do nosso método em uma máquina 2.4GHz Intel Core 2 Duo com 2GB RAM. Repare que o número de iterações para que as estruturas mediais e os segmentos converjam para uma bijeção não é maior que 10. Portanto, a complexidade de tempo do nosso algoritmo depende somente do custo de uma iteração da rotina principal, o qual é, no pior caso,  $O(|\mathcal{F}| \log |\mathcal{F}|)$ . O passo mais caro do nosso método é o cálculo dos pares de autovalores e autovetores. Porém, estes são computados em uma fase de pré-processamento e, usando a técnica proposta por [99], o custo é reduzido para ordem de segundos (Tabela 2.1).

Modelo	$ \mathcal{V} $	$ \mathcal{F} $	# Iterações	Tempo (s)
dinopet	4k	8k	5	0.9
human	8k	16k	5	3.0
dog	9.5k	19k	6	4.7
armadillo	23k	46k	3	6.7
elephant	42k	84.5k	2	13.4
dinosaur	56k	112k	2	29.9

Tabela 4.1: Tempo de execução do algoritmo de segmentação.

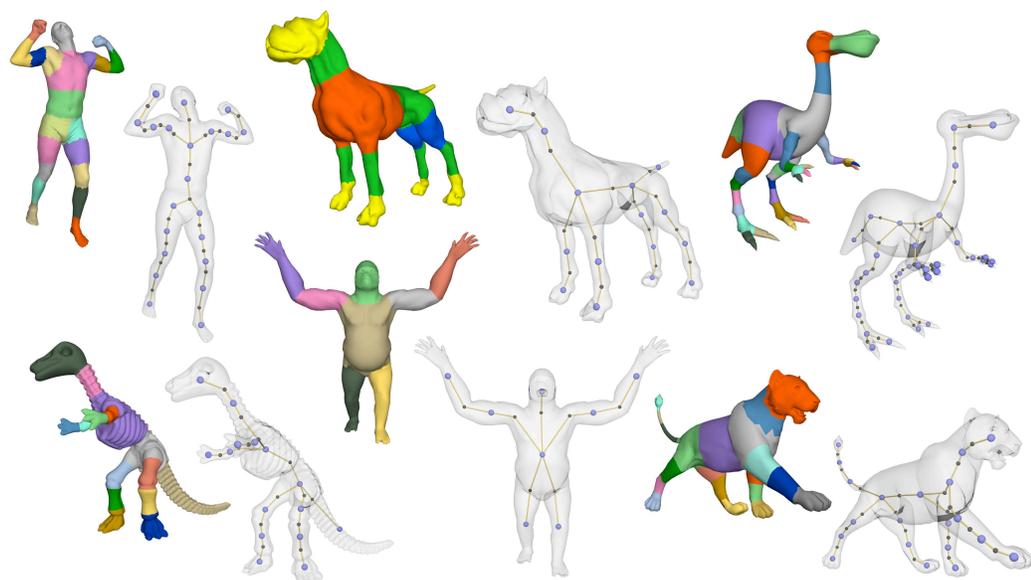


Figura 4.9: Esqueletos extraídos a partir dos segmentos. As esferas azuis indicam os centros de massa dos segmentos e as esferas cinzas indicam os núcleos das articulações dos personagens.

## Capítulo 5

# Geração de Atlas Baseado em Estruturas Globais para Modelos Orgânicos

Motivados pelos segmentos significativos apresentados no Capítulo 4, introduzimos agora uma nova técnica para construção de atlas em superfícies. Este método combina informações semânticas com características geométricas e, assim, recupera as estruturas globais de modelos 3D. Concentramos nossa abordagem em modelos orgânicos, já que nestes as estruturas semânticas são predominantes. O trabalho descrito neste capítulo resultou no artigo [20].

### 5.1 Motivação

Pela definição de variedades, superfícies podem ser decompostas em regiões homeomorfas ao disco. Estas regiões são denominadas cartas e, em conjunto, compõem o atlas da superfície. A geração de atlas proporciona muitas vantagens para as aplicações gráficas. Sua importância se deve ao fato de que as cartas de um atlas simplificam a análise e o processamento da superfície. Em sistemas de pintura 3D, por exemplo, um atlas permite o mapeamento de superfícies com geometrias complexas em texturas planares.

A versatilidade dos atlas pode ser expandida capturando as estruturas representativas das superfícies. Preservando tais estruturas, os atlas se tornam capazes de orientar algoritmos de remalhamento, por exemplo, codificando o *layout* final das novas malhas, ou então oferecem interessantes malhas de controle para técnicas de multirresolução e de deformação de espaço.

Métodos convencionais geram atlas adotando uma perspectiva quantitativa. A ideia central é balancear um número pequeno de cartas com parametrizações suaves e de baixa distorção. Com este fim, os atlas são gerados minimizando energias globais, tais como

métricas de distorção de área e de ângulo. Geometricamente, a ocorrência de distorções está relacionada com regiões de alta curvatura. Por esta razão, vários métodos orientam suas cartas em relação às direções principais de curvatura das superfícies. Entretanto, as características geométricas não são suficientes para expressar todas as estruturas chaves de uma superfície, principalmente quando esta representa modelos orgânicos. Como discutido em [100], o atlas ideal é aquele que combina características geométricas com a interpretação semântica do objeto.

Para peças mecânicas, veículos e móveis, as estruturas relevantes tipicamente carregam um significado de engenharia. Assim, medidas de curvatura e convexidade mais ângulos diedrais bastam para detectar suas principais estruturas. Porém, abordagens puramente geométricas são menos aptas para formas orgânicas. Modelos animados são dominados por articulações e saliências e, por isso, requerem uma análise funcional.

As técnicas existentes agregam estruturas semânticas em atlas apenas através de intervenção manual. Consequentemente, a criação de atlas persiste como uma tarefa extremamente tediosa e custosa. Por outro lado, conforme relatado no Capítulo 4, a última década introduziu uma vasta literatura para o problema de segmentação semântica. Guiados por estudos de cognição, estes métodos resultam em decomposições de modelos 3D em regiões significativas. No contexto de geração de atlas, esses segmentos complementam as estruturas geométricas e, assim, compõem atlas semelhantes aos que um artista faria.

A seguir, apresentamos uma nova estratégia para gerar atlas de modelos orgânicos. Nosso método computa o atlas considerando tanto as estruturas geométricas quanto as semânticas. Assim, garantimos cartas com baixa distorção e um atlas com *layout* intuitivo. Nossa abordagem é fundamentada na ideia de que regiões significativas vizinhas possuem estruturas geométricas alinhadas. Incorporamos este alinhamento usando uma simples modificação do método VSA [16]. Entre outras vantagens, o método VSA faz com que o nosso algoritmo seja guiado por uma otimização, a qual pode ser realizada automaticamente ou em sessões interativas. Aplicamos e comparamos nosso algoritmo com o estado da arte para vários modelos e demonstramos que nossos atlas estruturados podem auxiliar aplicações gráficas existentes, como os problemas de remalhamento quadrangular e de deformação de superfícies baseada em celas.

## 5.2 **Trabalhos Relacionados**

Muitas das técnicas que geram atlas estão relacionadas com o mapeamento de textura em superfícies. Normalmente, estes métodos também estão vinculados com o problema de parametrização de cartas em planos. A seguir, revisamos somente os geradores de atlas, independentemente da parametrização utilizada. Para mais detalhes sobre parametrização, veja [41].

Nos trabalhos pioneiros sobre mapeamento de textura [33, 54], os atlas são criados através de sistemas interativos de pintura. Um dos primeiros métodos automáticos foi proposto em [67], e particiona malhas agrupando faces com normais similares. Abordagens gulosas [87, 28] também combinam normais para clusterizar faces em regiões planares. Em [95], as cartas são formadas limitando o erro de distorção do domínio paramétrico. Outros métodos minimizam distorção decompondo superfícies em regiões desenvolvíveis [46].

Lévy et al. [59] exploraram o fato de que, ao se mapear texturas, zonas de alta curvatura causam mais artefatos e apresentaram um algoritmo que alinha as cartas do atlas com as estruturas geométricas da superfície. Zhou et al. [109] aplicaram o método espectral *IsoMaps* (Seção 2.2) para criar cartas com distorção controlada e que aproximem regiões significativas. Outra técnica baseada em estruturas semânticas foi apresentada em [106]. Este trabalho primeiro segmenta malhas em partes significativas com a ajuda de funções de distâncias e, então, parametriza cada segmento em uma ou duas cartas. Recentemente, Vallet e Lévy [100] desenvolveram um esquema semi-interativo para computar atlas de textura para malhas artísticas. Este método estende a análise espectral de grafos para recuperar os eixos de simetria em malhas. Assim, as curvas sobre as malhas correspondentes a esses eixos compõem novos atlas.

Também relacionado com parametrização, algoritmos de remalhamento usam atlas para especificar a aparência das novas malhas [98]. Em [24], por exemplo, atlas quadrangulares são computados a partir de complexos Morse-Smale extraídos das autofunções do operador Laplace-Beltrami de uma dada superfície (Seção 3.5). Já em [42], mecanismos de controle foram introduzidos para permitir que interações do usuário orientem tais complexos. Outras abordagens de remalhamento [26, 88, 9, 12] criam atlas sensíveis a atributos geométricos, combinando técnicas de clusterização baseadas em normais e em distâncias geodésicas. Em particular, Marinov e Kobbelt [70] dividiram superfícies baseados no método VSA.

O método VSA [16] é uma estratégia variacional para simplificar malhas. Fundamentada no conceito de *proxies*, esta abordagem minimiza o erro de aproximação entre uma superfície e um conjunto de planos, alternando clusterização de faces e o ajuste de cada plano em relação a sua respectiva região. Esse trabalho também introduziu a métrica  $\mathcal{L}^{2,1}$ , melhorando a captura da anisotropia das superfícies. O método VSA serve como ponto de partida para a nossa proposta e, por isso, revisamo-no em mais detalhes na Seção 5.3.

A simplificação em esquemas de multirresolução também tem sido extensivamente usada para gerar atlas [56, 32, 55, 69]. Isto está relacionado ao fato de que, semelhante ao atlas, as técnicas de simplificação visam preservar os atributos geométricos das superfícies. Em especial, Khodakovsky et al. [51] otimizaram o *layout* de atlas com um critério de simplificação que pondera a geometria e a distorção das cartas.

A caracterização de peças mecânicas também apresenta técnicas que preservam estruturas geométricas. Explorando o significado de engenharia destas formas, alguns métodos decompõem superfícies em partes geométricas, como componentes convexos [52] ou primitivas geométricas [29, 2]. Em [102, 105], por exemplo, o método VSA foi expandido para aproximar regiões de um modelo por superfícies quádricas. Já abordagens alternativas detectam estruturas geométricas diretamente, baseando-se apenas em medidas geométricas, tais como ângulos diedrais [27], curvaturas principais [57] e as derivadas das curvaturas [36]. Resultados interessantes também são possíveis combinando as estratégias anteriores: primeiro os modelos são separados em partes e cada parte é analisada individualmente.

Como já dissemos, estruturas geométricas não caracterizam todos os atributos de modelos orgânicos. Este tipo de objeto também requer um conhecimento semântico que divida a superfície em regiões significativas. Uma maneira de atender esta necessidade é através de métodos de segmentação semântica. Na Seção 4.2, revisamos vários desses métodos e mostramos como a análise espectral do operador Laplace-Beltrami auxilia a resolução desse problema.

### 5.3 Método VSA

O método VSA (*Variational Shape Approximation*) é uma técnica de simplificação que otimiza um conjunto finito de planos para aproximar uma dada superfície. Para isso, a superfície é decomposta em partições  $\mathcal{P}^j$  e cada partição é vinculada a um plano. A qualidade da aproximação entre cada partição e seu plano é avaliada de acordo com uma métrica  $\mathcal{L}$ . Como a aplicação desta métrica depende diretamente da descrição do plano, o método VSA apresenta o conceito de proxy. Assim, um *proxy*  $P^j$  é definido pelo par  $(\mathbf{c}_j, \mathbf{n}_j)$  que determina o plano associado à partição  $\mathcal{P}^j$ , onde  $\mathbf{c}_j$  é um ponto do plano e  $\mathbf{n}_j$  é a normal do plano. Já a otimização dos *proxies* e das partições consiste na minimização da energia

$$\mathcal{E}(\{\mathcal{P}^j\}_{j=1}^k, \{P^j\}_{j=1}^k) = \sum_{j=1}^k \mathcal{L}(\mathcal{P}^j, P^j). \quad (5.1)$$

Para malhas triangulares  $\mathcal{M}$ , a solução da Eq. 5.1 pode ser computada através de uma variação do método de Lloyd, em que a energia  $\mathcal{E}$  é minimizada alternando dois subproblemas. No primeiro subproblema, as partições são fixadas e os *proxies* otimizados. Já no segundo subproblema, os *proxies* são fixados e as partições otimizadas. Em mais detalhes, cada iteração da otimização é dividida em três etapas: *cálculo de sementes*, *inundação* e *ajuste dos planos*. Dada uma configuração inicial das partições e dos *proxies*, começamos cada iteração computando o triângulo  $s_j$  da partição  $\mathcal{P}^j$  com o menor erro de aproximação em relação ao *proxy*  $P^j$ . Em seguida, realizamos a etapa de inundação, na qual aplicamos um crescimento de região a partir de cada semente  $s_j$  e, assim, atribuímos

o triângulo  $t$  para a partição  $P^j$  que minimiza a distância  $\mathcal{L}(t, P^j)$ . Por último, ajustamos os planos de cada *proxy* minimizando a métrica  $\mathcal{L}$  com as partições fixadas.

Em [16], foram apresentadas duas possíveis métricas para guiar a otimização. A primeira delas é a usual métrica de Hausdorff

$$\mathcal{L}^2(\mathcal{P}, P) = \int_{\mathcal{P}} \|x - \Pi(x)\|^2 dx, \quad (5.2)$$

onde  $\Pi(x)$  indica a projeção ortográfica de  $x$  no plano do *proxy*  $P$ . No limite assintótico de áreas elípticas, a aproximação resultante desta métrica tende a criar elementos com razão de aspecto igual a  $\sqrt{|\kappa_{\max}/\kappa_{\min}|}$ , onde  $\kappa_{\max}, \kappa_{\min}$  são as curvaturas principais. A segunda métrica proposta, por sua vez, é definida por

$$\mathcal{L}^{2,1}(\mathcal{P}, P) = \int_{\mathcal{P}} \|\mathbf{n}(x) - \mathbf{n}_P\|^2 dx, \quad (5.3)$$

onde  $\mathbf{n}(x)$  é a normal do ponto  $x$  e  $\mathbf{n}_P$  é a normal do *proxy*  $P$ . Por avaliar a variação da normal, esta métrica captura melhor a anisotropia da superfície, criando elementos com razão de aspecto na ordem de  $|\kappa_{\max}/\kappa_{\min}|$ .

Na próxima seção, apresentamos a nossa técnica como uma modificação do método VSA baseado na métrica  $\mathcal{L}^{2,1}$ . Logo, dada uma malha  $\mathcal{M}$ , aproximamos  $\mathcal{L}^{2,1}(t, P) \simeq \|\mathbf{n}_t - \mathbf{n}_P\|^2 A_t$ , onde  $A_t$  é a área do triângulo  $t$ , e computamos o plano que melhor aproxima uma partição  $\mathcal{P}$  através da média dos triângulos de  $\mathcal{P}$  ponderados por suas áreas. A seguir, indicamos as partições de um segmento  $\mathcal{R}_i$  por  $\mathcal{P}_i^j$  e seus *proxies* por  $P_i^j$ .

## 5.4 Notação

Novamente trabalhamos com malhas triangulares  $\mathcal{M}$ . Também reutilizamos a decomposição de  $\mathcal{M}$  em regiões significativas  $\{\mathcal{R}_i\}$ . Uma estrutura semântica  $\mathcal{S}_{ij}$  corresponde à sequência de arestas conectadas (fechadas ou não) compartilhadas entre dois segmentos adjacentes  $\mathcal{R}_i$  e  $\mathcal{R}_j$ . Já as estruturas semânticas que delimitam a região  $\mathcal{R}_i$  são indicadas por  $S_i = \{\mathcal{S}_{ij}\}$ . Durante o nosso algoritmo, particionamos cada estrutura semântica em cortes. Denotamos por  $C_{ij}$  o conjunto de cortes  $\mathcal{C}_{ij}^k \subset \mathcal{S}_{ij}$ , e por  $S_i^C \subset S_i$  as estruturas semânticas com  $C_{ij} \neq \emptyset$ . Indicamos por  $C_i = \{\mathcal{C}_i^k\} = \cup_{\mathcal{S}_{ij} \in S_i^C} \mathcal{C}_{ij}$  a coleção de cortes em  $S_i$ . Por fim, usamos  $C_{ij,i}$  para informar que os cortes  $C_{ij}$  são gerados a partir de  $\mathcal{R}_i$ .

## 5.5 Algoritmo

O objetivo do nosso algoritmo é gerar atlas preservando as estruturas geométricas e semânticas de modelos orgânicos. Para isso, estendemos o método VSA revisado na Seção 5.3. No final, obtemos um atlas que decompõe a malha de entrada  $\mathcal{M}$  em um conjunto de cartas homeomorfas a discos e com bordas alinhadas às suas estruturas.

Modelos orgânicos são extremamente dependentes de estruturas semânticas, como articulações e saliências. Estas estruturas caracterizam o contexto do objeto, indicando sua pose e seu nível de detalhe. No nosso método, incorporamos essas estruturas como um dado de entrada adicional do nosso algoritmo. Assim, possibilitamos o reuso da variedade de métodos de segmentação semântica existente. As estruturas semânticas são definidas como as fronteiras entre regiões significativas adjacentes.

Dadas as estruturas semânticas, nosso método extrai os atributos geométricos em cada região significativa. Em peças mecânicas, as estruturas geométricas podem ser identificadas em cada parte do objeto isoladamente. Já em modelos orgânicos, estas estruturas compõem longas curvas que cruzam as estruturas semânticas. Logo, para gerar o atlas, não podemos analisar os segmentos de entrada individualmente (Figura 5.1A). Ao invés, desenvolvemos mecanismos para controlar o alinhamento e a suavidade entre as estruturas geométricas que transitam entre segmentos adjacentes.

Este controle de alinhamento é baseado na seguinte formulação: dada uma estrutura semântica  $\mathcal{S}_{ij}$ , as estruturas geométricas de  $\mathcal{R}_i$  e  $\mathcal{R}_j$  se encontram em  $\mathcal{S}_{ij}$  definindo dois conjuntos de cortes,  $\mathcal{C}_{ij,i}$  e  $\mathcal{C}_{ij,j}$ , respectivamente. Assumindo que  $\mathcal{C}_{ij,j}$  seja computado primeiro, afirmamos que as estruturas geométricas estão alinhadas em  $\mathcal{S}_{ij}$  se, para todo  $\mathcal{C}_{ij,j}^k$ , existir  $\mathcal{C}_{ij,i}^l$  tal que  $\mathcal{C}_{ij,j}^k \subset \mathcal{C}_{ij,i}^l$ .

Incorporamos a restrição de alinhamento em uma simples modificação do método VSA. A seguir, descrevemos os principais passos desta modificação.

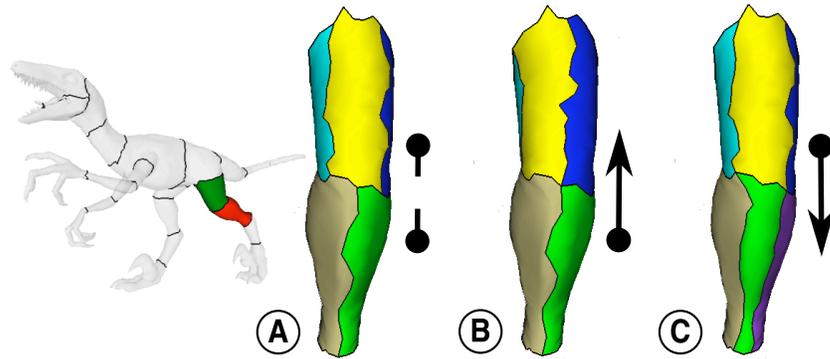


Figura 5.1: Cartas da perna esquerda do modelo *raptor*. (A) Computar cartas em cada segmento separadamente resulta em estruturas desalinhadas. Nosso método controla o alinhamento ordenando a dependência entre os segmentos. Ordenações diferentes podem induzir resultados diferentes. (B) parte inferior da perna restringe parte superior; (C) parte superior da perna restringe parte inferior.

**Ordem de Dependência.** Nossa formulação de alinhamento impõe uma dependência na ordem em que os segmentos podem ser processados. Dada uma estrutura semântica  $\mathcal{S}_{ij}$

e assumindo que as estruturas geométricas de  $\mathcal{R}_j$  são extraídas primeiro, as possibilidades de *proxies* em  $\mathcal{R}_i$  ficam restritas a  $C_{ij,j}$ . Conforme ilustrado na Figura 5.1, diferentes ordenações dos segmentos podem induzir resultados diferentes. Logo, começamos nosso algoritmo especificando uma ordenação padrão para os segmentos. Iniciamos com os segmentos próximos às extremidades do modelo e avançamos, progressivamente, para seus segmentos vizinhos, preferindo sempre segmentos de menor área.

Seguindo esta ordenação, aplicamos a nossa modificação do método VSA em cada segmento, um por um. É fácil perceber que os primeiros segmentos acessados não possuem nenhuma dependência. Por isso, nesses casos, o nosso método se reduz ao VSA original. Já os segmentos subsequentes  $\mathcal{R}_i$  possuem cortes ( $C_i \neq \emptyset$ ), logo, seus *proxies* devem particionar  $\mathcal{R}_i$  de maneira consistente com  $C_i$ .

**Funções Harmônicas.** Com o intuito de garantir esta consistência, utilizamos o número de cortes  $|C_i|$  como uma estimativa inicial do número de *proxies* em  $\mathcal{R}_i$ . Associamos cada *proxy*  $P_i^k$  com um corte  $C_i^k$ , e inicializamos suas respectivas partições  $\mathcal{P}_i^k$  com os triângulos adjacentes a  $C_i^k$ . Além disso, avaliamos a influência de cada corte  $C_i^k$  em  $\mathcal{R}_i$  (Figura 5.2A-B). Seguindo [25], computamos a influência de  $C_i^k$  através da função harmônica  $h_i^k$  resultante da equação de Laplace:

$$\begin{cases} \Delta h_i^k(x) = 0 & \forall x \in \mathcal{R}_i \setminus S_i, \\ h_i^k(x) = 1 & \forall x \in C_i^k, \\ h_i^k(x) = 0 & \text{caso contrário.} \end{cases} \quad (5.4)$$

Esta equação é discretizada em  $\mathcal{R}_i$  usando a *fórmula da cotangente* (Seção 2.5.2). Note que computamos todas as funções harmônicas em  $\mathcal{R}_i$  fatorando a matriz Laplaciana apenas uma vez e mudando o lado direito da equação para cada corte.

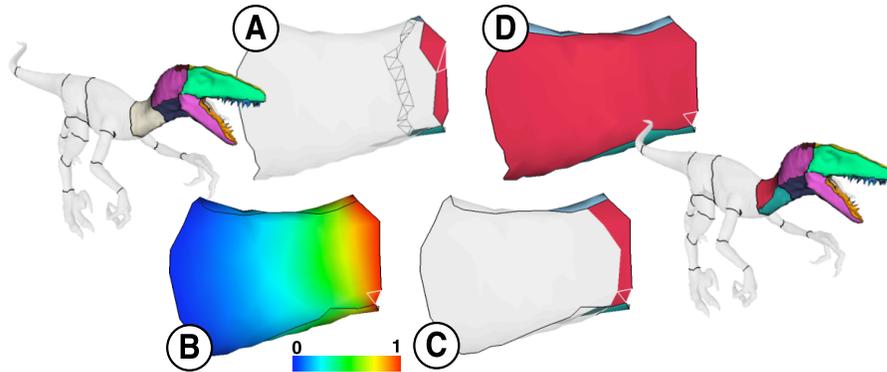


Figura 5.2: Cartas do pescoço do modelo *raptor*. O pescoço é restrito pela cabeça por quatro cortes. (A) Inicialização das partições usando os triângulos adjacentes aos cortes. (B) Funções harmônicas computadas a partir dos cortes. (C) Triângulos pré-atribuídos para cada partição durante a etapa de colagem. (D) Cartas finais. Triângulos brancos representam as sementes dos *proxies*.

**Colagem.** Conforme a descrição da Seção 5.3, o método VSA é um processo iterativo e cada iteração é dividida em três etapas: *cálculo das sementes*, *inundação* e *ajuste dos planos*. Para forçar que cortes e *proxies* sejam alinhados, introduzimos uma etapa adicional: a *colagem*. Esta etapa ocorre entre o cálculo das sementes e a inundação. A ideia dessa fase é pré-atribuir alguns triângulos de  $\mathcal{R}_i$  para garantir que cada corte seja totalmente vinculado à sua respectiva partição. Para isso, percorremos cada partição  $\mathcal{P}_i^k$  e lhe pré-atribuímos todos os triângulos com  $h_i^k(t) \geq h_i^k(s_i^k)$ , onde  $h_i^k(t)$  é a média de  $h_i^k$  nos vértices do triângulo  $t$ . Já os triângulos restantes são atribuídos normalmente durante a inundação (Figura 5.2C-D). Repetimos esta iteração de quatro fases até que os *proxies* convirjam ou um número máximo de iterações seja alcançado.

**Métrica Estendida.** Tanto as etapas de inundação e de ajuste requerem uma métrica para calcular o erro de distorção. Esta métrica também dita a extração de estruturas geométricas em  $\mathcal{R}_i$ . Como discutido na Seção 5.3, utilizamos a métrica  $\mathcal{L}^{2,1}$  (Eq. 5.3), pois esta captura melhor a anisotropia da superfície. No entanto, para que o processo de inundação seja influenciado pelos cortes, combinamos  $\mathcal{L}^{2,1}$  com as funções harmônicas:

$$\mathcal{L}_A^{2,1}(\mathcal{P}_i^k, P_i^k) = \int_{\mathcal{P}_i^k} (1 - h_i^k(x)) \|\mathbf{n}(x) - \mathbf{n}_{P_i^k}\|^2 dx. \quad (5.5)$$

Esta métrica estendida induz as sementes das partições a se aproximarem dos cortes e, consequentemente, pré-atribui uma quantidade menor de triângulos durante a colagem. Observe que a métrica estendida só é usada para *proxies* associados a cortes.

Terminadas as iterações, temos o segmento  $\mathcal{R}_i$  dividido em partições  $\{\mathcal{P}_i^k\}$  e a garantia, por construção, de que  $C_{ij,i} = C_{ij,j}$ , para todas as estruturas semânticas  $\mathcal{S}_{ij} \in S_i^C$ . As estruturas geométricas correspondem às sequências de arestas entre pares de partições adjacentes. Entretanto, estas estruturas podem não representar o nível de detalhe desejado para o segmento  $\mathcal{R}_i$ . A inicialização com  $|C_i|$  *proxies* pode causar uma indesejável sub- ou super-segmentação.

**Inserindo Proxies.** Quando  $\mathcal{R}_i$  é sub-segmentado, precisamos inserir mais *proxies*. Logo, dividimos os *proxies* com os maiores erros de distorção. Incrementalmente, inicializamos um novo *proxy* com o triângulo de pior erro pertencente ao último *proxy* dividido. Note que a métrica estendida induz novos *proxies* a se afastarem dos cortes, preservando a associação entre cortes e *proxies*. Após inserir um novo *proxy*, limpamos as partições existentes e repetimos as iterações (Figura 5.3).

**Contraindo Proxies.** No caso contrário, precisamos remover *proxies* consistentemente com os cortes  $C_i$ . Para cada estrutura restrita  $\mathcal{S}_{ij} \in S_i^C$ , devemos assegurar que  $\mathcal{C}_{ij,j}^l \subset \mathcal{C}_{ij,i}^k$ . Para manter esta restrição, desenvolvemos uma abordagem de contração gulosa. Nela, construímos um grafo dual em  $\mathcal{R}_i$  em que os nós correspondem às partições e as arestas correspondem às estruturas geométricas. Semelhante ao método proposto

em [70], ponderamos cada aresta desse grafo de acordo com a sua relevância geométrica:

$$Peso_{ab} = [1 + \mathcal{L}^2(\mathcal{P}_i^{ab}, P_i^{ab})] \times [1 + \mathcal{L}^{2,1}(\mathcal{P}_i^{ab}, P_i^{ab})]. \quad (5.6)$$

Este critério balanceia a distorção espacial e normal da partição  $\mathcal{P}_i^{ab} = \mathcal{P}_i^a \cup \mathcal{P}_i^b$  em relação ao seu *proxy* otimizado  $P_i^{ab}$ . Contraímos as arestas com o menor peso até que um critério de parada seja respeitado (Figura 5.4).

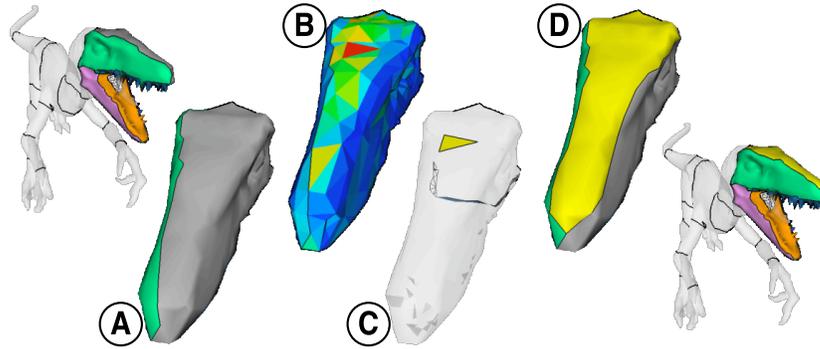


Figura 5.3: Cartas do maxilar do modelo *raptor*. (A) Cartas iniciais. A mandíbula impõe três cortes sobre o maxilar. (B) As cores indicam o erro de distorção entre as partições e seus respectivos *proxies*, aumentando de azul para vermelho. (C) Inserção de um novo *proxy* inicializado com o triângulo de maior erro. (D) Cartas finais após as iterações.

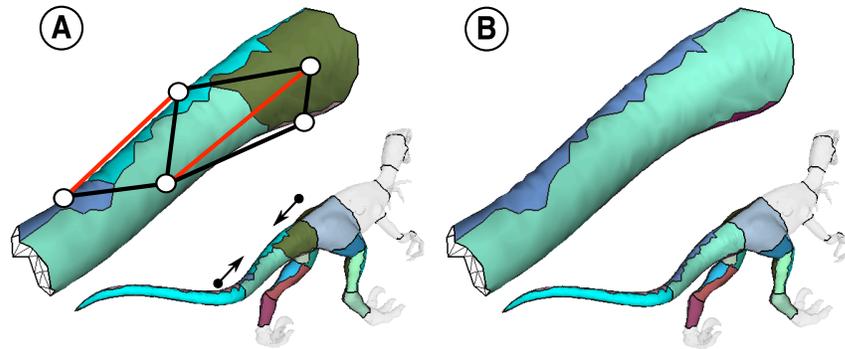


Figura 5.4: Cartas para a parte superior do rabo do modelo *raptor*. A parte inferior do rabo e o quadril impõem seis cortes (três cada). (A) Partições iniciais e grafo dual. (B) Cartas finais após contrair as arestas em vermelho.

**Pós-Processamento.** Após particionar todas as regiões de  $\mathcal{M}$ , nosso método resulta em cartas próximas a planos e alinhadas às estruturas geométricas e semânticas. Para gerar o atlas final, verificamos se todas as cartas são topologicamente equivalentes a discos.

Em nossos experimentos, apenas poucas cartas não respeitaram esta exigência. Para estes poucos casos, detectamos os vértices âncoras da partição, i.e., os vértices em que três ou mais cartas se encontram, e daí computamos a triangulação de Delaunay com restrições (CDT) [16]. Convertemos essa partição em um disco usando as menores arestas do interior da triangulação.

No Algoritmo 2, apresentamos o pseudocódigo do nosso método. Observe que nossa técnica introduz apenas um módulo extra ao método VSA. Por um lado, este módulo é responsável por controlar o alinhamento das estruturas da superfície, aperfeiçoando o método VSA para objetos orgânicos. Por outro lado, desativando esse módulo, reproduzimos o VSA original e extraímos as estruturas chaves de modelos não orgânicos.

<pre> <b>Input:</b> malha <math>\mathcal{M}</math> e segmentos <math>\{\mathcal{R}_i\}</math> <b>Output:</b> cartas <math>\{\mathcal{P}_i^k\}</math>  1 ordene os segmentos <math>\{\mathcal{R}_i\}</math> 2 <b>foreach</b> segmento <math>\mathcal{R}_i</math> na ordenação <b>do</b> 3   <b>if</b> <math>C_i \neq \emptyset</math> <b>then</b> 4     crie <math> C_i </math> <i>proxies</i> 5     associe os <i>proxies</i> <math>\{P_i^k\}</math> com os cortes <math>\{C_i^k\}</math> 6     compute as funções harmônicas <math>\{h_i^k\}</math>  7   <b>repeat</b> 8     cálculo das sementes das partições 9     <b>if</b> <math>C_i \neq \emptyset</math> <b>then</b> colagem 10    inundação 11    ajuste dos planos 12  <b>until</b> <i>num.iterações</i> &lt; <i>MAX</i>  13  <b>if</b> <math>\mathcal{R}_i</math> for sub-segmentado <b>then</b> 14    adicione um novo <i>proxy</i> 15    <b>goto</b> linha 7 16  <b>else</b> 17    crie o grafo dual 18    <b>while</b> <math>\mathcal{R}_i</math> for super-segmentado <b>do</b> 19      contraia a aresta com o menor peso  20 <b>if</b> alguma carta <math>\mathcal{P}_i^k</math> não for homeomorfa a disco <b>then</b> 21   compute CDT em <math>\mathcal{P}_i^k</math> 22   costure as bordas da carta com as menores arestas internas </pre>
--

**Algoritmo 2:** Pseudocódigo para nosso VSA modificado.

## 5.6 Resultados

Testamos nosso método com vários modelos orgânicos. Em todos os exemplos, extraímos as estruturas semânticas usando o método de segmentação apresentado no Capítulo 4. O nosso método pode ser aplicado de duas maneiras: automaticamente ou em sessões interativas. Para executá-lo automaticamente, só precisamos especificar uma quota superior para o erro de aproximação entre as partições e os *proxies*. Porém, em nossos experimentos, preferimos uma estratégia mais intuitiva e, por isso, implementamos nosso algoritmo acoplado a um ambiente interativo. Além das operações básicas de inserção e remoção de *proxies*, este sistema permite o usuário navegar entre os segmentos. Assim, a ordenação de dependência descrita na Seção 5.5 pode ser alterada, enfatizando diferentes aspectos geométricos da superfície em questão. Realizamos tais sessões interativas em uma máquina 2.4GHz Intel Core 2 Duo com 2GB RAM e gastamos menos de dois minutos em cada exemplo.

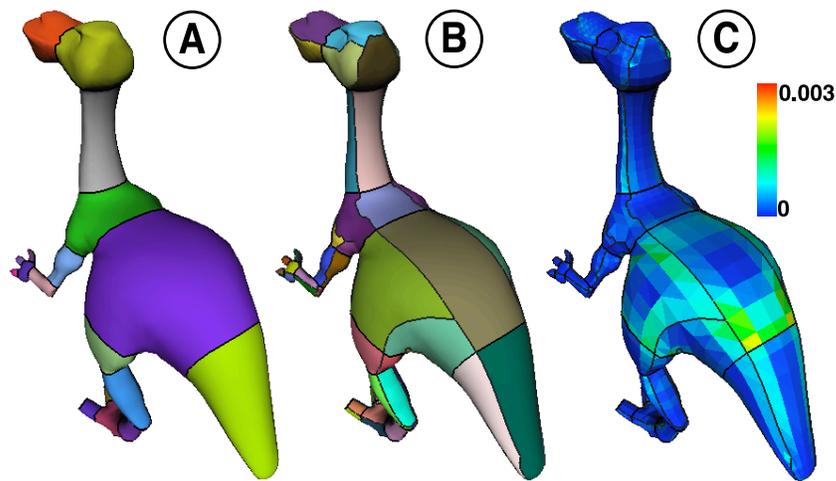


Figura 5.5: (A) Segmentação semântica computada com o método do Capítulo 4. (B) Atlas final. Observe o alinhamento das curvas geométricas cruzando as estruturas semânticas. (C) Distorção da métrica  $\mathcal{L}^{2,1}$  entre as cartas e os respectivos *proxies*.

Na Figura 5.5, discutimos as vantagens do nosso método de enriquecimento de atlas para modelos orgânicos. A primeira vantagem da nossa abordagem é o fato dela ser baseada em resultados de segmentação semântica (Figura 5.5A). Esta característica faz com que nossos atlas tenham um *layout* mais natural, reproduzindo como um artista criaria cartas em uma superfície. Além disso, a diversidade de segmentações existentes (veja alguns exemplos na Seção 4.2) expande a flexibilidade do nosso método. Assim, podemos criar atlas aptos a contextos de diferentes aplicações gráficas, desde mapeamento de textura até traçado de raio e detecção de colisões. Em segundo lugar, as estruturas

semânticas complementam as estruturas geométricas como regiões em potencial para a ocorrência de distorção da parametrização. Por isso, detectando ambas as estruturas, garantimos um atlas com distorção de métrica suave e controlada (Figura 5.5C). Por último, nosso mecanismo de alinhamento possibilita a extração de estruturas globais em objetos orgânicos. Na Figura 5.5B, por exemplo, detectamos longas estruturas atravessando o modelo *dinopet* a partir da cabeça até o rabo e também partindo das patas até o quadril. Mais exemplos de atlas gerados pelo nosso método estão disponíveis na Figura 5.6.

Comparamos nossos resultados com três outras técnicas: VSA [16], IsoCharts [109] e WYSIWYG [100]. A Figura 5.7 apresenta alguns exemplos. Em relação ao método VSA, nossa abordagem corresponde a uma extensão que cria cartas maiores e mais regulares. Comparado com o método IsoCharts, é fácil observar que nossos resultados melhoram a preservação das estruturas dos modelos, porém, contendo um número maior de cartas. Por fim, o nosso método apresenta aspectos complementares em relação ao método WYSIWYG. Ambas as técnicas são baseadas em métodos de segmentação. No entanto, nós exploramos segmentações semânticas, detectando simetrias funcionais como a correspondência entre os braços e as pernas. Por outro lado, o método WYSIWYG generaliza a técnica PCA (*Principal Component Analysis*), identificando simetrias globais como as metades de um corpo.



Figura 5.6: Exemplos de atlas alinhados a estruturas gerados pelo nosso método. Em cada exemplo, mostramos a segmentação e a vista de frente e por trás do atlas.

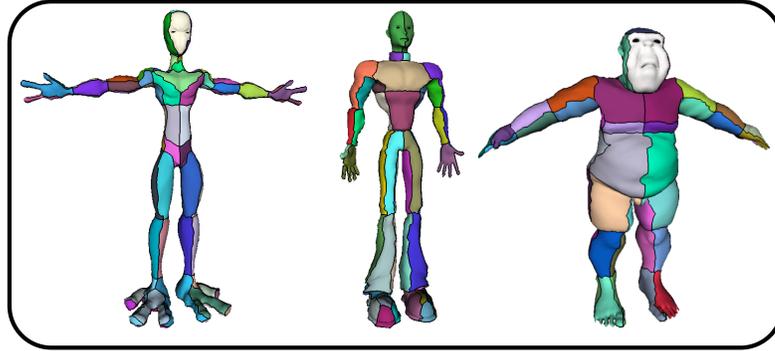
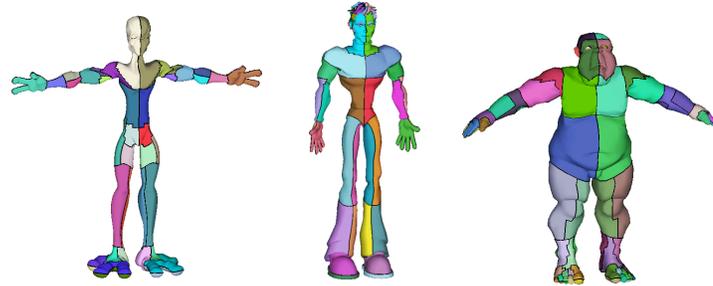
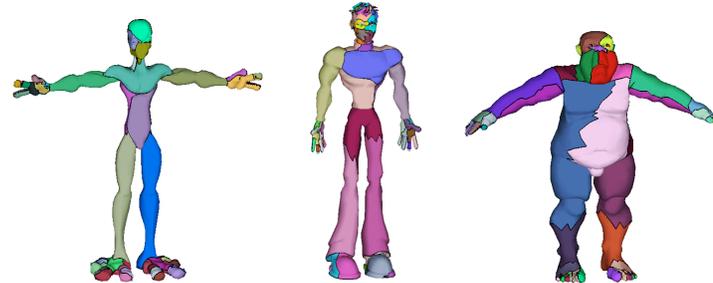
**Structure-Aware Atlas****WYSIWYG****IsoCharts****VSA**

Figura 5.7: Comparação dos nossos resultados com os métodos WYSIWYG [100], IsoCharts [109] e VSA [16]. As imagens comparativas são cortesia de [100].

## 5.7 Aplicações

Muitas aplicações gráficas funcionam melhor se as estruturas das superfícies são previamente conhecidas. Entretanto, a maioria destas aplicações propõe a detecção dessas estruturas manualmente. Nosso método, por sua vez, oferece estruturas semânticas e geométricas automaticamente. Nesta seção, mostramos duas aplicações existentes que se beneficiam com o nosso novo atlas.

### 5.7.1 Remalhamento Quadrangular

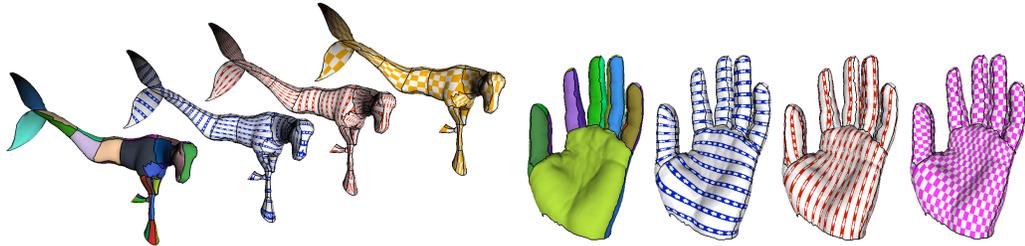


Figura 5.8: Aplicação do atlas estruturado para remalhamento quadrangular. As imagens da esquerda mostram os atlas utilizados. As imagens centrais indicam as direções dos gradientes dos potenciais computados pelo método [98]. Nas imagens da direita, mapeamos uma textura de xadrez usando os campos potenciais como coordenadas.

Tong et al. [98] introduziram um sistema linear para remalhamento quadrangular. Baseado em formas diferenciais discretas, este método codifica singularidades em linhas e em pontos através de um atlas denominado grafo de singularidade. Além de armazenar as singularidades, esse grafo também controla o *layout* e o tamanho dos quadriláteros da malha final. Para criá-lo, os autores apresentaram um algoritmo assistido em que os tensores de curvatura orientam a posição e o alinhamento das singularidades em relação às direções principais de curvatura.

Como discutimos anteriormente, os atlas baseados apenas em análise geométrica são deficientes no que se refere à preservação de estruturas semânticas, principalmente em objetos orgânicos, e a nossa abordagem sobrepõe esta limitação. Portanto, substituímos o grafo de singularidades com o nosso atlas alinhado a estruturas. A Figura 5.8 ilustra os campos potenciais resultantes do método [98] usando o nosso atlas enriquecido. O contorno de valores inteiros destes campos potenciais compõe uma malha de quadriláteros. Por fim, vale ressaltar que nenhuma modificação foi feita no algoritmo de remalhamento. Nosso atlas apenas representa uma nova alternativa para o grafo de singularidade, projetada especificamente para modelos orgânicos.

### 5.7.2 Deformação baseada em Cella

Deformação baseada em cela é uma técnica em que uma superfície é distorcida através da manipulação de uma segunda superfície, envelopando a primeira. Esta outra superfície é representada por uma malha triangular de baixa resolução, normalmente chamada de *cela*. Um ponto qualquer no interior da cela é manipulado através de uma combinação linear dos vértices da cela ponderados por um conjunto de coordenadas previamente calculado.

Apesar da variedade de trabalhos relacionados com o desenvolvimento de coordenadas, o processo de criação de celas ainda é um tópico em aberto. Recentemente, Ju et al. [45] mostraram que o *design* da cela interfere diretamente na facilidade de se controlar uma deformação. Uma cela ideal seria uma malha com poucos elementos e contendo informações semânticas sobre a superfície a ser deformada. Por esta razão, Ju et al. propuseram a construção de celas a partir de pequenas malhas padronizadas associadas a esqueletos.

Nesta mesma linha, nós propomos a criação de celas a partir dos nossos atlas estruturados. Computamos celas triangulares aplicando sobre nosso atlas a triangulação de Delaunay com restrições (CDT) [16]. Assim, nossa cela possui vértices localizados nas regiões funcionais e arestas orientadas pelas estruturas da superfície. Por fim, como a cela precisa envelopar a superfície deformável, deslocamos cada face da cela seguindo suas normais. Na Figura 5.9, mostramos celas extraídas a partir dos nossos atlas e algumas deformações realizadas com as coordenadas Green [62].

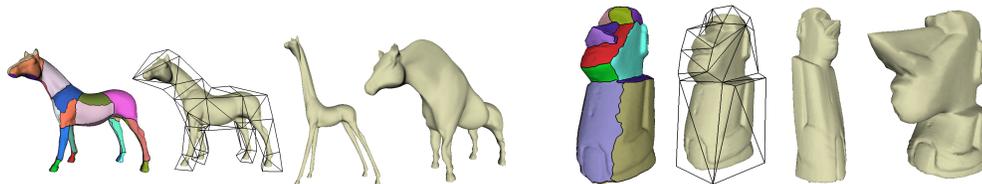


Figura 5.9: Aplicação do atlas estruturado para a deformação de superfícies baseado em celas. Da esquerda para direita: nosso atlas estruturado, a cela extraída do atlas e deformações.

# Capítulo 6

## Conclusão

Este projeto de mestrado introduziu uma nova abordagem para a modelagem geométrica de superfícies. Apoiados na análise espectral do operador Laplace-Beltrami, discutimos, desenvolvemos e ilustramos ferramentas elegantes para o estudo e processamento da geometria e topologia de superfícies. Estas técnicas complementam conceitos diferenciais através de análises multi-escala e invariantes a isometrias e se aplicam em diversos problemas de computação gráfica, incluindo segmentação, geração de atlas, remalhamento e animação.

Motivamos e justificamos o uso do domínio espectral do operador Laplace-Beltrami revisando as definições e propriedades relacionadas aos seus autovalores e às suas autofunções. Também comparamos as vantagens do operador Laplace-Beltrami em relação a outros métodos espectrais. Em particular, mostramos que o operador Laplace-Beltrami generaliza e unifica o conhecimento de diversas áreas, tais como análise de sinais, teoria espectral de grafos e aprendizado de máquina. Além disso, apresentamos um estudo teórico e prático da discretização do operador Laplace-Beltrami em superfícies representadas por malhas triangulares. Comprovamos que os pares de autovalores e autofunções podem ser computados em malhas com amostragens não-uniformes na ordem de segundos, convergindo e preservando as propriedades contínuas.

Aplicamos a análise espectral do operador Laplace-Beltrami no problema de segmentação hierárquica de objetos articulados. Baseados na teoria espectral, introduzimos dois novos conceitos: a distância de difusão e a estrutura medial. Através destes, desenvolvemos um algoritmo iterativo que resulta em uma hierarquia de segmentos, revelando a estrutura de objetos articulados. Nosso método garante várias propriedades interessantes, como invariância a deformações isométricas e robustez a ruídos, malhas irregulares e pequenos eventos topológicos.

Inspirados pelas segmentações semânticas, também desenvolvemos uma nova estratégia para construir atlas na superfície de objetos orgânicos. Nossa técnica gera atlas preser-

vando tanto as estruturas semânticas entre os segmentos quanto as estruturas geométricas características das superfícies. Para isso, expandimos o método VSA e a métrica  $\mathcal{L}^{2,1}$  [16] incorporando mecanismos que controlem o alinhamento entre as estruturas semânticas e geométricas. Assim, garantimos atlas com baixa distorção geométrica e um *layout* intuitivo e artístico. Estes atlas também foram utilizados em aplicações como remalhamento quadrangular e deformação de superfícies baseada em celas.

Quanto a trabalhos futuros, acreditamos que a investigação das propriedades espectrais do operador Laplace-Beltrami pode auxiliar ainda mais a resolução de problemas em computação gráfica. Em especial, estamos interessados na detecção de simetrias parciais e de correspondências entre superfícies semelhantes. Também planejamos estudar a animação de objetos através de transformações isométricas e a relação entre as auto-funções do operador Laplace-Beltrami e diferentes imersões de sua superfície em  $\mathbb{R}^3$ .

# Referências Bibliográficas

- [1] D. Anguelov, D. Koller, H.-C. Pang, P. Srinivasan, and S. Thrun. Recovering articulated object models from 3D range data. In *Proc. of the 20th conf. on uncertainty in artificial intelligence*, pages 18–26, 2004.
- [2] M. Attene, B. Falcidieno, and M. Spagnuolo. Hierarchical mesh segmentation based on fitting primitives. *Vis. Comput.*, 22(3):181–193, 2006.
- [3] M. Attene, S. Katz, M. Mortara, G. Patane, M. Spagnuolo, and A. Tal. Mesh segmentation - a comparative study. In *Proc. of the IEEE Intl. Conf. on Shape Modeling and Appl.*, page 7, 2006.
- [4] R. Bass. *Probabilistic Techniques in Analysis*. Springer-Verlag, 1995.
- [5] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computations*, 15(6):1373—1396, 2003.
- [6] M. Belkin, J. Sun, and Y. Wang. Constructing laplace operator from point clouds in  $R^d$ . Technical report, 2008.
- [7] M. Belkin, J. Sun, and Y. Wang. Discrete laplace operator for meshed surfaces. In *ACM-SIAM Symp. on Discrete Algorithms*, pages 1031–1040, 2009.
- [8] A. I. Bobenko and B. A. Springborn. A discrete laplace-beltrami operator for simplicial surfaces. arXiv:math.DG/0503219.
- [9] I. Boier-Martin, H. Rushmeier, and J. Jin. Parameterization of triangle meshes over quadrilateral domains. In *Proc. of the Eurographics Symp. on Geometry Processing*, pages 193–203, 2004.
- [10] M. Botsch and O. Sorkine. On linear variational surface deformation methods. *IEEE Transactions on Visualization and Computer Graphics*, 14(1):213–230, 2008.

- [11] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. *Proc. National Academy of Sciences (PNAS)*, 103(5):1168–1172, 2006.
- [12] N. A. Carr., J. Hoberock, K. Crane, and J. C. Hart. Rectangular multi-chart geometry images. In *Proc. of the Eurograph. Symp. on Geometry Processing*, pages 181–190, 2006.
- [13] I. Chavel. *Eigenvalues in Riemannian Geometry*. Academic Press, 1984.
- [14] F. R. K. Chung. *Spectral Graph Theory*. AMS, 1997.
- [15] B. Cipra. You can't always hear the shape of a drum. *What's happening in the Mathematical Sciences*, 1, 1993.
- [16] D. Cohen-Steiner, P. Alliez, and M. Desbrun. Variational shape approximation. In *SIGGRAPH*, pages 905–914. ACM, 2004.
- [17] R. R. Coifman and S. Lafon. Diffusion maps. *Applied and Comp. Harmonic Analysis: Special issue on Diffusion Maps and Wavelets*, 21:5–30, 2006.
- [18] R. Courant and D. Hilbert. *Methods of Mathematical Physics*, volume 1. Interscience Publishers, 1953.
- [19] F. de Goes, S. Goldenstein, and L. Velho. A hierarchical segmentation of articulated bodies. *Computer Graphics Forum*, 27(5):1349–1356, 2008.
- [20] F. de Goes, S. Goldenstein, and L. Velho. Structure-aware atlas for organic models. *submitted*, 2009.
- [21] M. Desbrun, E. Kanso, and Y. Tong. Discrete differential forms for computational modeling. In *SIGGRAPH ASIA courses*, pages 1–17. ACM, 2008.
- [22] M. Desbrun, M. Meyer, P. Schröder, and A. Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. In *SIGGRAPH*, pages 317–324. ACM, 1999.
- [23] M. P. do Carmo. *Geometria Diferencial de Curvas e Superfícies*. SBM, 2004.
- [24] S. Dong, P.-T. Bremer, M. Garland, V. Pascucci, and J. C. Hart. Spectral surface quadrangulation. *ACM Trans. Graph.*, 25(3):1057–1066, 2006.
- [25] S. Dong, S. Kircher, and M. Garland. Harmonic functions for quadrilateral remeshing of arbitrary manifolds. *Comput. Aided Geom. Des.*, 22(5):392–423, 2005.

- [26] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle. Multiresolution analysis of arbitrary meshes. In *SIGGRAPH*, pages 173–182. ACM, 1995.
- [27] R. Gal, O. Sorkine, N. Mitra, and D. Cohen-Or. iWIRES: An analyze-and-edit approach to shape manipulation. *SIGGRAPH*, 2009. to appear.
- [28] M. Garland, A. Willmott, and P. S. Heckbert. Hierarchical face clustering on polygonal surfaces. In *Proc. of the Symp. on Interactive 3D graphics*, pages 49–58. ACM, 2001.
- [29] N. Gelfand and L. Guibas. Shape segmentation using local slippage analysis. In *Proc. of the Eurographics Symp. on Geometry Processing*, pages 214–223, 2004.
- [30] C. Gordon, D. Webb, and S. Wolpert. Isospectral plane domains and surfaces via riemannian orbifolds. *Inventiones Mathematicae*, 110:1–22, 1992.
- [31] E. Grinspun, A. N. Hirani, M. Desbrun, and P. Schröder. Discrete shells. In *Proc. of the ACM SIGGRAPH/Eurographics Symp. on Computer Animation*, pages 62–67. Eurographics Association, 2003.
- [32] I. Guskov, K. Vidimče, W. Sweldens, and P. Schröder. Normal meshes. In *SIGGRAPH*, pages 95–102. ACM, 2000.
- [33] P. Hanrahan and P. Haeberli. Direct WYSIWYG painting and texturing on 3d shapes. In *SIGGRAPH*, pages 215–223. ACM, 1990.
- [34] P. S. Heckbert and M. Garland. Optimal triangulation and quadric-based surface simplification. *Comput. Geom. Theory Appl.*, 14(1-3):49–65, 1999.
- [35] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii. Topology matching for fully automatic similarity estimation of 3d shapes. In *SIGGRAPH*, pages 203–212. ACM, 2001.
- [36] K. Hildebrandt, K. Polthier, and M. Wardetzky. Smooth feature lines on surface meshes. In *Proc. of Eurographics Symp. on Geometry Processing*, pages 85–90, 2005.
- [37] A. N. Hirani and J. E. Marsden. *Discrete exterior calculus*. PhD thesis, California Institute of Technology, 2003.
- [38] D. D. Hoffman and W. A. Richards. Parts of recognition. *Cognition*, 18:65–96, 1984.
- [39] D. D. Hoffman and M. Singh. Saliency of visual parts. *Cognition*, 63:29–78, 1997.

- [40] K. Hormann and M. S. Floater. Mean value coordinates for arbitrary planar polygons. *ACM Trans. Graph.*, 25(4):1424–1441, 2006.
- [41] K. Hormann, B. Lévy, and A. Sheffer. Mesh parameterization: theory and practice. In *SIGGRAPH courses*, 2007.
- [42] J. Huang, M. Zhang, J. Ma, X. Liu, L. Kobbelt, and H. Bao. Spectral quadrangulation with orientation and alignment control. In *SIGGRAPH Asia*, pages 1–9. ACM, 2008.
- [43] D. Jakobson, N. Nadirashvili, and J. Toth. Geometric properties of eigenfunctions. *Russian Mathematical Surveys*, 56(6):1085–1105, 2001.
- [44] D. L. James and C. D. Twigg. Skinning mesh animations. In *SIGGRAPH*, pages 399–407. ACM, 2005.
- [45] T. Ju, Q.-Y. Zhou, M. van de Panne, D. Cohen-Or, and U. Neumann. Reusable skinning templates using cage-based deformations. In *SIGGRAPH Asia*, pages 1–10. ACM, 2008.
- [46] D. Julius, V. Kraevoy, and A. Sheffer. D-charts: Quasi-developable mesh segmentation. *Comput. Graph. Forum*, 24(3):581–590, 2005.
- [47] M. Kac. Can you hear the shape of a drum? *Amer. Math. Monthly*, 73, 1966.
- [48] Z. Karni and C. Gotsman. Spectral compression of mesh geometry. In *SIGGRAPH*, pages 279–286. ACM, 2000.
- [49] S. Katz, G. Leifman, and A. Tal. Mesh segmentation using feature point and core extraction. *The Visual Computer (Pacific Graph.)*, 21(8-10):649–658, 2005.
- [50] S. Katz and A. Tal. Hierarchical mesh decomposition using fuzzy clustering and cuts. In *SIGGRAPH*, pages 954–961. ACM, 2003.
- [51] A. Khodakovsky, N. Litke, and P. Schröder. Globally smooth parameterizations with low distortion. *ACM Trans. Graph.*, 22(3):350–357, 2003.
- [52] V. Kraevoy, D. Julius, and A. Sheffer. Model composition from interchangeable components. In *15th Pacific Conf. on Comp. Graph. and Appl.*, pages 129–138, 2007.
- [53] V. Kraevoy and A. Sheffer. Variational, meaningful shape decomposition. In *SIGGRAPH Technical Sketches*, page 50. ACM, 2006.

- [54] V. Krishnamurthy and M. Levoy. Fitting smooth surfaces to dense polygon meshes. In *SIGGRAPH*, pages 313–324. ACM, 1996.
- [55] A. Lee, H. Moreton, and H. Hoppe. Displaced subdivision surfaces. In *SIGGRAPH*, pages 85–94. ACM, 2000.
- [56] A. W. F. Lee, W. Sweldens, P. Schröder, L. Cowsar, and D. Dobkin. MAPS: multiresolution adaptive parameterization of surfaces. In *SIGGRAPH*, pages 95–104, 1998.
- [57] Y. Lee, S. Lee, A. Shamir, D. Cohen-Or, and H.-P. Seidel. Mesh scissoring with minima rule and part salience. *Comp. Aided Geom. Design*, 22(5):444–465, 2005.
- [58] B. Lévy. Laplace-beltrami eigenfunctions: Towards an algorithm that understands geometry. In *IEEE International Conference on Shape Modeling and Applications, invited talk*, 2006.
- [59] B. Lévy, S. Petitjean, N. Ray, and J. Maillot. Least squares conformal maps for automatic texture atlas generation. *ACM Trans. Graph.*, 21(3):362–371, 2002.
- [60] J.-M. Lien and N. M. Amato. Approximate convex decomposition of polyhedra and its applications. *Comp. Aided Geom. Design*, 25(7):503–522, 2008.
- [61] J.-M. Lien, J. Keyser, and N. M. Amato. Simultaneous shape decomposition and skeletonization. In *Proc. of the ACM Symp. on solid and physical modeling*, pages 219–228, 2006.
- [62] Y. Lipman, D. Levin, and D. Cohen-Or. Green coordinates. *ACM Trans. Graph.*, 27(3):1–10, 2008.
- [63] D. Liu, G. Xu, and Q. Zhang. A discrete scheme of laplace-beltrami operator and its convergence over quadrilateral meshes. *Comput. Math. Appl.*, 55(6):1081–1093, 2008.
- [64] R. Liu and H. Zhang. Segmentation of 3D meshes through spectral clustering. In *12th Pacific Conf. on Computer Graphics and Appl.*, pages 298–305, 2004.
- [65] R. Liu and H. Zhang. Mesh segmentation via spectral embedding and contour analysis. *Computer Graphics Forum (Special Issue of Eurographics)*, 26, 2007.
- [66] Y. Liu, B. Prabhakaran, and X. Guo. A robust spectral approach for blind watermarking of manifold surfaces. In *Proc. of the 10th ACM workshop on Multimedia and security*, pages 43–52. ACM, 2008.

- [67] J. Maillot, H. Yahia, and A. Verroust. Interactive texture mapping. In *SIGGRAPH*, pages 27–34. ACM, 1993.
- [68] A. P. Mangan and R. T. Whitaker. Partitioning 3D surface meshes using watershed segmentation. *IEEE Trans. on Vis. and Comp. Graphics*, 5(4):308–321, 1999.
- [69] M. Marinov and L. Kobbelt. Automatic generation of structure preserving multiresolution models. *Comput. Graph. Forum*, 24(3):479–486, 2005.
- [70] M. Marinov and L. Kobbelt. A robust two-step procedure for quad-dominant remeshing. *Comput. Graph. Forum*, 25(3):537–546, 2006.
- [71] M. Meyer and A. Barr. *Discrete differential operators for computer graphics*. PhD thesis, California Institute of Technology, 2004.
- [72] M. Meyer, M. Desbrun, P. Schröder, and A. Barr. Discrete differential geometry operators for triangulated 2-manifolds. In *VisMath*, 2002.
- [73] A. Mohr and M. Gleicher. Building efficient, accurate character skins from examples. *ACM Trans. Graph.*, 22(3):562–568, 2003.
- [74] M. Mortara, G. Patanè, M. Spagnuolo, B. Falcidieno, and J. Rossignac. Plumber: a method for a multi-scale decomposition of 3D shapes into tubular primitives and bodies. In *Proc. of the ninth ACM symp. on solid modeling and applications*, pages 339–344, 2004.
- [75] M. Ovsjanikov, J. Sun, and L. Guibas. Global intrinsic symmetries of shapes. *Computer Graphics Forum*, 27(5):1341–1348, 2008.
- [76] D. L. Page, A. Koschan, and M. Abidi. Perception-based 3D triangle mesh segmentation using fast marching watersheds. In *Proc. of the Computer Vision and Pattern Recognition*, pages 27–32, 2003.
- [77] U. Pinkall, S. Des Juni, and K. Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics*, 2:15–36, 1993.
- [78] M. Reuter. Can one hear shape? In *International Congress of Industrial and Applied Mathematics*, 2007.
- [79] M. Reuter, S. Biasotti, D. Giorgi, G. Patanè, and M. Spagnuolo. Discrete laplace-beltrami operators for shape analysis and segmentation. *Computers & Graphics*, 33(3):381–390, 2009. to appear.

- [80] M. Reuter, F.-E. Wolter, and N. Peinecke. Laplace-beltrami spectra as "shape-dna" of surfaces and solids. *Computer-Aided Design*, 38(4):342–366, 2006.
- [81] M. Reuter, F.-E. Wolter, M. Shenton, and M. Niethammer. Laplace-beltrami eigenvalues and topological features of eigenfunctions for statistical shape analysis. *Computer-Aided Design*, 2009. to appear.
- [82] G. Rong, Y. Cao, and X. Guo. Spectral mesh deformation. *Vis. Comput.*, 24(7):787–796, 2008.
- [83] G. Rong, Y. Cao, and X. Guo. Spectral surface deformation with dual mesh. In *Proc. of International Conference on Computer Animation and Social Agents*, pages 17 – 24, 2008.
- [84] S. Rosenberg. *The Laplacian on a Riemannian Manifold*. Cambridge Un. Press, 1997.
- [85] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.
- [86] R. M. Rustamov. Laplace-beltrami eigenfunctions for deformation invariant shape representation. In *Proc. of the fifth Eurographics Symp. on Geometry Processing (SGP)*, pages 225–233, 2007.
- [87] P. V. Sander, J. Snyder, S. J. Gortler, and H. Hoppe. Texture mapping progressive meshes. In *SIGGRAPH*, pages 409–416. ACM, 2001.
- [88] P. V. Sander, Z. Wood, S. J. Gortler, J. Snyder, and H. Hoppe. Multi-chart geometry images. In *Proc. of the Eurograph. Symp. on Geometry Processing*, pages 146–155, 2003.
- [89] S. Schaefer and C. Yuksel. Example-based skeleton extraction. In *Proc. of the fifth Eurographics Symp. on Geometry Processing*, pages 153–162, 2007.
- [90] B. Scholköpf, E. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, 1998.
- [91] A. Shamir. A formulation of boundary mesh segmentation. In *Proc. of the 2nd Intl. Symp. on 3D Data Proc., Visualization, and Transmission*, pages 82–89, 2004.
- [92] A. Shamir. Segmentation and shape extraction of 3D boundary meshes. In *State-of-art Report: Eurographics*, pages 137–149, 2006.

- [93] L. Shapira, A. Shamir, and D. Cohen-Or. Consistent mesh partitioning and skeletonisation using the shape diameter function. *Visual Comp.*, 24(4):249–259, 2008.
- [94] S. Shlafman, A. Tal, and S. Katz. Metamorphosis of polyhedral surfaces using decomposition. *Computer Graphics Forum*, 21(3):219–228, 2002.
- [95] O. Sorkine, D. Cohen-Or, R. Goldenthal, and D. Lischinski. Bounded-distortion piecewise mesh parameterization. In *Proc. of the Conf. on Visualiz.*, pages 355–362, 2002.
- [96] G. Taubin. A signal processing approach to fair surface design. In *SIGGRAPH*, pages 351–358. ACM, 1995.
- [97] J. B. Tenenbaum and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.
- [98] Y. Tong, P. Alliez, D. Cohen-Steiner, and M. Desbrun. Designing quadrangulations with discrete harmonic forms. In *Proc. of the fourth Eurographics Symp. on Geometry Processing*, pages 201–210, 2006.
- [99] B. Vallet and B. Lévy. Spectral geometry processing with manifold harmonics. *Computer Graphics Forum (Proc. Eurographics)*, 27(2), 2008.
- [100] B. Vallet and B. Lévy. What you seam is what you get. Technical report, INRIA - ALICE Project Team, 2009.
- [101] M. Wardetzky, S. Mathur, F. Kälberer, and E. Grinspun. Discrete Laplace operators: No free lunch. In *Symposium on Geometry Processing*, pages 33–37, 2007.
- [102] J. Wu and L. Kobbelt. Structure recovery via hybrid variational surface approximation. *Comput. Graph. Forum*, 24(3):277–284, 2005.
- [103] G. Xu. Discrete laplace-beltrami operator on sphere and optimal spherical triangulations. *Int. J. Comput. Geometry Appl*, 16:75–93, 2004.
- [104] G. Xu. Convergence analysis of a discretization scheme for gaussian curvature over triangular surfaces. *Computer Aided Geometric Design*, 23(2):193–207, 2006.
- [105] D.-M. Yan, Y. Liu, and W. Wang. Quadric surface extraction by variational shape approximation. In *Proc. of Geometric Modeling and Processing*, pages 73–86, 2006.
- [106] E. Zhang, K. Mischaikow, and G. Turk. Feature-based surface parameterization and texture mapping. *ACM Trans. on Graph.*, 24(1):1–27, 2005.

- [107] H. Zhang and R. Liu. Mesh segmentation via recursive and visually salient spectral cuts. In *Proc. of Vision, Modeling, and Visualization*, pages 429–436, 2005.
- [108] H. Zhang, O. van Kaick, and R. Dyer. Spectral mesh processing. *Computer Graphics Forum*, 2009. to appear.
- [109] K. Zhou, J. Snyder, B. Guo, and H.-Y. Shum. Iso-charts: stretch-driven mesh parameterization using spectral analysis. In *Proc. of the Eurographics Symp. on Geometry Processing*, pages 45–54. ACM, 2004.