

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

DENNIS GIOVANI BALREIRA

**Aplicação da Teoria das Transformações de
D'Arcy Thompson em Computação Gráfica**

Trabalho de Graduação.

Prof. Dr. Marcelo Walter
Orientador

Porto Alegre, dezembro de 2012

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos Alexandre Netto

Pró-Reitor de Graduação: Prof. Sérgio Roberto Kieling Franco

Diretor do Instituto de Informática: Prof. Luís da Cunha Lamb

Coordenador da Ciência da Computação: Prof. Raul Fernando Weber

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

*“You mustn’t allow yourself to be chained to fate, to be ruled by your genes.
Human beings can choose the kind of life that they want to live.
What’s important is that you choose life... and then live.”*
— DR. NAOMI HUNTER, METAL GEAR SOLID

AGRADECIMENTOS

À minha família, em especial à minha mãe, Maria José, por todo o apoio, carinho e dedicação em todas as etapas da minha vida. Grande parte do que eu conquisei só foi possível porque ela estava ao meu lado. Muito obrigado!

Aos meus amigos por todos os momentos que passamos juntos. É muito bom saber que sempre posso contar com eles.

Aos meus colegas de curso por toda a ajuda e amizade ao longo desses quatro anos.

Ao meu professor orientador Marcelo Walter pelo excelente apoio e interesse durante todo o desenvolvimento do Trabalho de Conclusão.

Ao Instituto de Informática pela infraestrutura, técnicos e corpo docente de altíssima qualidade, em especial ao professor Raul Fernando Weber pelos inúmeros conselhos ao longo do curso e ao professor Dante Augusto Couto Barone pela orientação durante a bolsa de iniciação científica do grupo PET Computação.

A UFRGS pela oportunidade única de expandir o conhecimento por meio de um ensino público extremamente qualificado e reconhecido internacionalmente.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS	6
LISTA DE FIGURAS	7
RESUMO	11
ABSTRACT	12
1 INTRODUÇÃO	13
1.1 Motivação	14
1.2 Objetivo	14
1.3 Organização do texto	14
2 REVISÃO BIBLIOGRÁFICA	16
2.1 Modelagem Geométrica	16
2.2 Métodos para Representação da Forma em Computação Gráfica	16
2.3 Morphing 2D	17
2.4 Deformação de Superfícies e Morphing 3D	19
2.5 Comparação entre Formas	20
2.6 Discussão	21
3 IMPLEMENTAÇÃO	23
3.1 Módulo de Modelagem	23
3.1.1 Funcionalidades	24
3.1.2 Transformações do Espaço	25
3.1.3 Polinômio do Segundo Grau com Três Variáveis	25
3.1.4 Coordenadas Polares	28
3.2 Módulo de Validação	30
3.3 Ambiente de Desenvolvimento	33
4 RESULTADOS	34
4.1 Transformações do Trabalho de Thompson	34
4.2 Discussão	34
5 CONCLUSÃO	42
REFERÊNCIAS	44

LISTA DE ABREVIATURAS E SIGLAS

2D	Duas Dimensões
3D	Três Dimensões
API	<i>Application Programming Interface</i>
GCC	<i>GNU Compiler Collection</i>
GLM	<i>Graphic Library Model</i>
GLUT	<i>OpenGL Utility Toolkit</i>
GNU	<i>GNU is Not Unix</i>
IDE	<i>Integrated Development Environment</i>
MinGW	<i>Minimalist GNU for Windows</i>
OBJ	<i>OBJ File Format</i>
OpenCV	<i>Open Source Computer Vision</i>
OpenGL	<i>Open Graphics Library</i>

LISTA DE FIGURAS

Figura 1.1:	Transformação relacionando diferentes espécies de peixes. (a) <i>Polyprion</i> (acima) e <i>Pseudopriacanthus altus</i> (abaixo); (b) <i>Scorpaena</i> (acima) e <i>Antigonia capros</i> (abaixo); (c) <i>Argyropelecus olfersi</i> (acima) e <i>Sternoptyx diaphana</i> (abaixo); (d) <i>Diodon</i> (acima) e <i>Orthogoriscus</i> (abaixo). (Figura extraída do trabalho (ARTHUR, 2006))	13
Figura 1.2:	Diagrama de visualização do processo. (a) Modelo inicial; (b) Modelo inicial transformado; (c) Visão 2D lateral de (a) e do peixe original equivalente de Thompson; (d) Visão 2D lateral de (b) e do peixe original equivalente de Thompson; (e) Percentual de diferença entre as figuras de (c); (f) Percentual de diferença entre as figuras de (d). . .	15
Figura 2.1:	(a) Transformação com um par de linhas; (b) Transformação com múltiplos pares de linhas. (Figuras extraídas do trabalho (BEIER; NEELY, 1992))	18
Figura 2.2:	Imagens do clip musical de vídeo do Michael Jackson. (a) Primeira imagem a ser transformada na segunda; (b) Segunda imagem a ser transformada na primeira; (c) Transformação intermediária da primeira para a segunda; (d) Transformação intermediária da segunda para a primeira; (e) Resultado intermediário entre as duas imagens. (Figura extraída do trabalho de (BEIER; NEELY, 1992))	18
Figura 2.3:	Representação de morphing de um elefante para uma girafa usando interpolação linear (primeira linha) e o método As-Rigid-As-Possible (segunda linha). (Figura extraída do trabalho de (ALEXA; COHENOR; LEVIN, 2000))	19
Figura 2.4:	(a) Representação do objeto a ser transformado já limitado por um cubo; (b) Objeto (a) deformado. (Figuras extraídas do trabalho (SEDERBERG; PARRY, 1986))	20
Figura 2.5:	Transformação apresentando um par de objetos nas extremidades e suas etapas intermediárias. (Figura extraída do trabalho de (KENT; CARLSON; PARENT, 1992))	20
Figura 2.6:	Resultados para comparação de um ônibus (a) e de uma flor (b) usando a distância <i>Hamming</i> . (Figura extraída do trabalho de (LANDRÉ; TRUCHETET, 2007))	21
Figura 2.7:	(a) Figura base para reconhecimento de estruturas semelhantes; (b) Modelo de comparação para a figura (a); (c) Figura (a) com os modelos similares a (b) destacados. (Figura extraída do trabalho de (HUTTENLOCHER; KLANDERMAN; RUCKLIDGE, 1993))	21

Figura 3.1:	Diagrama do módulo de modelagem. (a) Objeto inicial carregado pelo programa; (b) Exemplo de transformação de (a), gerando um novo objeto.	23
Figura 3.2:	Tela inicial do aplicativo ao carregar o modelo.	24
Figura 3.3:	Menu de interação detalhado. (a) Trackball; (b) Coeficiente r correspondente ao raio das coordenadas polares juntamente com os coeficientes do polinômio de segundo grau para o eixo x ; (c) Funcionalidades <i>Zoom</i> , <i>Reset</i> , <i>Four Viewports</i> , <i>Debug</i> e <i>Illumination</i> ; (d) Coeficientes do polinômio de segundo grau para o eixo y ; (e) Coeficientes do polinômio de segundo grau para o eixo z juntamente com o botão <i>Exit</i>	24
Figura 3.4:	(a) Modelo inicial; (b) Transformação a partir de (a) na dimensão x com coeficiente $E = 1.0$	26
Figura 3.5:	(a) Transformação a partir da figura 3.4(a) na dimensão z com coeficiente $G = 0.5$; (b) Transformação a partir da figura 3.4(a) na dimensão y com coeficiente $A = 2.0$	26
Figura 3.6:	(a) Transformação a partir da figura 3.4(a) na dimensão x com coeficiente $A = 2.0$ e variáveis <i>neg</i> desativadas; (b) Transformação a partir da figura 3.4(a) na dimensão x com coeficiente $A = 2.0$ e variáveis <i>neg</i> ativadas.	28
Figura 3.7:	(a) Transformação a partir da figura 3.4(a) com coeficiente $cr1 = 1.2$; (b) Transformação a partir da figura 3.4(a) nas dimensões x , x e y com coeficientes $C = 2.0$, $D = 0.5$ e $E = 0.5$, respectivamente, e $cr1 = -0.15$	29
Figura 3.8:	Diagrama geral de exemplo do módulo de validação.	30
Figura 3.9:	(a) Imagem de entrada 1; (b) Imagem de entrada 2.	30
Figura 3.10:	(a) Imagem 1 segmentada por limite definido; (b) Imagem 2 segmentada por limite definido.	31
Figura 3.11:	(a) União da imagem 1 com imagem 2; (b) Intersecção da imagem 1 com imagem 2.	32
Figura 3.12:	União da imagem 1 e imagem 2 menos intersecção da imagem 1 e imagem 2; O percentual de diferença encontrado foi de aproximadamente 0.45.	32
Figura 3.13:	(a) Imagem 1 menos Imagem 2; (b) Imagem 2 menos Imagem 1; Os percentuais de diferença encontradas foram de aproximadamente 0.27 e 0.31, respectivamente.	33
Figura 4.1:	(a) Modelo inicial; (b) Transformação de (a) nas dimensões x e z , com coeficientes $D = 0.65$ e $H = 2.0$, respectivamente; (c) Imagem original do peixe <i>Polyprion</i> de Thompson; (d) Imagem original do peixe <i>Pseudopriacanthus altus</i> de Thompson.	36
Figura 4.2:	(a) Imagem do modelo (a) de 4.1; (b) Imagem do modelo (b) de 4.1; (c) Percentual de diferença de (a) e (b): 0.2219; (d) Imagem do peixe <i>Polyprion</i> ; (e) Imagem do peixe <i>Pseudopriacanthus altus</i> ; (f) Percentual de diferença de (d) e (e): 0.1708; (g) Percentual de diferença de (a) e (d): 0.2992; (h) Percentual de diferença de (b) e (e): 0.2328.	36

Figura 4.3:	(a) Modelo inicial; (b) Transformação de (a) nas dimensões x e y com coeficientes $C = 1.0$ e $B = -0.5$, respectivamente, e $cr1 = 0.5$; (c) Imagem do peixe <i>Scorpaena</i> de Thompson; (d) Imagem do peixe <i>Antigonia capros</i> de Thompson.	37
Figura 4.4:	(a) Imagem do modelo (a) de 4.3; (b) Imagem do modelo (b) de 4.3; (c) Percentual de diferença de (a) e (b): 0.2679; (d) Imagem do peixe <i>Scorpaena</i> ; (e) Imagem do peixe <i>Antigonia capros</i> ; (f) Percentual de diferença de (d) e (e): 0.3373; (g) Percentual de diferença de (a) e (d): 0.3460; (h) Percentual de diferença de (b) e (e): 0.3186.	37
Figura 4.5:	(a) Modelo inicial; (b) Transformação de (a) nas dimensões x e y com coeficientes $E = 0.5$, $E = -1.4$, respectivamente; (c) Imagem do peixe <i>Argyroleucus olfersi</i> de Thompson; (d) Imagem do peixe <i>Sternoptyx diaphana</i> de Thompson.	38
Figura 4.6:	(a) Imagem do modelo (a) de 4.5; (b) Imagem do modelo (b) de 4.5; (c) Percentual de diferença de (a) e (b): 0.2556; (d) Imagem do peixe <i>Argyroleucus olfersi</i> ; (e) Imagem do peixe <i>Sternoptyx diaphana</i> ; (f) Percentual de diferença de (d) e (e): 0.2235; (g) Percentual de diferença de (a) e (d): 0.2210; (h) Percentual de diferença de (b) e (e): 0.2588.	38
Figura 4.7:	(a) Modelo inicial; (b) Transformação de (a) nas dimensões x , y e y com coeficientes $C = 4.0$, $B = -0.5$ e $E = 2.0$, respectivamente; (c) Imagem do peixe <i>Scarus sp.</i> de Thompson; (d) Imagem do peixe <i>Pomacanthus</i> de Thompson.	39
Figura 4.8:	(a) Imagem do modelo (a) de 4.7; (b) Imagem do modelo (b) de 4.7; (c) Percentual de diferença de (a) e (b): 0.2796; (d) Imagem do peixe <i>Scarus sp.</i> ; (e) Imagem do peixe <i>Pomacanthus</i> ; (f) Percentual de diferença de (d) e (e): 0.1934; (g) Percentual de diferença de (a) e (d): 0.2466; (h) Percentual de diferença de (b) e (e): 0.2757.	39
Figura 4.9:	(a) Modelo inicial; (b) Transformação de (a) nas dimensões x , y e y com coeficientes $C = -2.0$, $B = 1.0$ e $E = 1.5$, respectivamente; (c) Imagem do peixe <i>Diodon</i> de Thompson; (d) Imagem do peixe <i>Orthogoriscus</i> de Thompson.	40
Figura 4.10:	(a) Imagem do modelo (a) de 4.9; (b) Imagem do modelo (b) de 4.9; (c) Percentual de diferença de (a) e (b): 0.2388; (d) Imagem do peixe <i>Diodon</i> ; (e) Imagem do peixe <i>Orthogoriscus</i> ; (f) Percentual de diferença de (d) e (e): 0.5114; (g) Percentual de diferença de (a) e (d): 0.3470; (h) Percentual de diferença de (b) e (e): 0.3821.	40
Figura 4.11:	(a) Modelo inicial; (b) Transformação de (a) nas dimensões y , y , z e z com coeficientes $F = 0.5$, $I = 0.3$, $C = -0.5$ e $I = -0.5$, respectivamente; (c) Imagem de <i>Human Skull</i> de Thompson; (d) Imagem de <i>Cimpanzee Skull</i> de Thompson.	41
Figura 4.12:	(a) Imagem do modelo (a) de 4.11; (b) Imagem do modelo (b) de 4.11; (c) Percentual de diferença de (a) e (b): 0.1574; (d) Imagem do crânio humano; (e) Imagem do crânio do chimpanzé; (f) Percentual de diferença de (d) e (e): 0.2742; (g) Percentual de diferença de (a) e (d): 0.1595; (h) Percentual de diferença de (b) e (e): 0.2998.	41

Figura 5.1: (a) Modelo inicial; (b) Transformação de (a) nas dimensões x e y com coeficientes $B = -1.0$ e $C = 1.0$, respectivamente. 43

RESUMO

A Teoria das Transformações de D'Arcy Thompson argumenta que diferenças nas formas do corpo entre organismos podem ser simplesmente explicadas por transformações geométricas, ao invés de complicados esquemas de rearranjo dos componentes físicos. A forma de um peixe baiacu, por exemplo, pode ser transformada para a de um peixe lua por uma modificação do sistema de coordenadas retangular que descreve o peixe baiacu em um sistema curvilíneo para o peixe lua, que alonga a porção posterior do peixe.

O objetivo deste projeto é investigar a aplicabilidade desta teoria em tarefas de modelagem geométrica para computação gráfica. Foi desenvolvido um sistema de modelagem geométrica que utiliza os princípios da Teoria de Thompson. Este método permite a síntese de novas formas a partir de modificações nos sistemas de coordenadas de uma forma original. Um teste simples de comparação entre formas foi implementado para prover uma métrica de validação dos resultados. Os resultados obtidos apontam que a Teoria das Transformações pode ser uma poderosa ferramenta em tarefas de Modelagem Geométrica onde uma grande variedade de formas similares mas com variações individuais é necessária.

Palavras-chave: D'Arcy Thompson, teoria das transformações, computação gráfica.

Applying D'Arcy Thompson's Theory of Transformations in computer graphics

ABSTRACT

The D'Arcy Thompson's Theory of Transformations states that differences in the shapes of the bodies among organisms can be simply explained by geometric transformations, instead of by complicated rearrangements schemes of the physical components. The shape of a fugu fish, for instance, can be transformed to a sunfish by modifying the rectangular coordinate system that describes the fugu fish in a curvilinear system to model the sunfish, stretching the back of the fish.

The goal of this project is to investigate the applicability of this theory in geometric modeling tasks for computer graphics. It was developed a geometric modeling system that implements the principles of Thompson's Theory. This method allows the synthesis of new shapes from modifications in the coordinate systems in an original way. A simple comparison test among shapes was implemented as a metric to validate the results. The results show that the Theory of Transformations can be a powerful tool in Geometric Modeling tasks where a wide variety of similar shapes but with individual variations is needed.

Keywords: D'Arcy Thompson, theory of transformations, computer graphics.

1 INTRODUÇÃO

O estudo da teoria das transformações por D'Arcy Thompson concretizou-se com a publicação de seu livro *On Growth and Form* (D'ARCY, 1963), onde tratava a evolução como um aspecto fundamental para a forma de organismos vivos, relacionando-a com princípios físicos e matemáticos. Particularmente no capítulo XVII deste importante livro, ele argumentou que as diferenças nas formas entre espécies relacionadas de seres vivos podem ser obtidas por meio de transformações geométricas em um plano cartesiano. A ideia principal de D'Arcy Thompson permanece em muitos casos inexplorada. Mais recentemente inicia-se uma discussão sobre a relevância desse trabalho pioneiro e sua possível contribuição para entendimento genético (ARTHUR, 2006), ou seja, existe alguma relação entre as transformações geométricas propostas por Thompson e os genes que controlam as formas dos seres vivos? Esta ainda é uma pergunta em aberto.

Algumas das transformações mais impressionantes ocorrem entre tipos de peixes, conforme observado na figura 1.1. Algumas das espécies retratadas sofrem modificações simples, como é o caso do *Argyropelecus olfersi* e *Sternoptyx diaphana*, (fig. 1.1(c)) que recebem uma simples operação de deslizamento (*shear*). Outras, entretanto, requerem uma deformação mais elaborada geometricamente, como é o caso das espécies *Scorpaena* e *Antigonia capros* (fig. 1.1(b)), na qual a primeira sofre um alongamento sobre o eixo x , principalmente no meio da imagem.

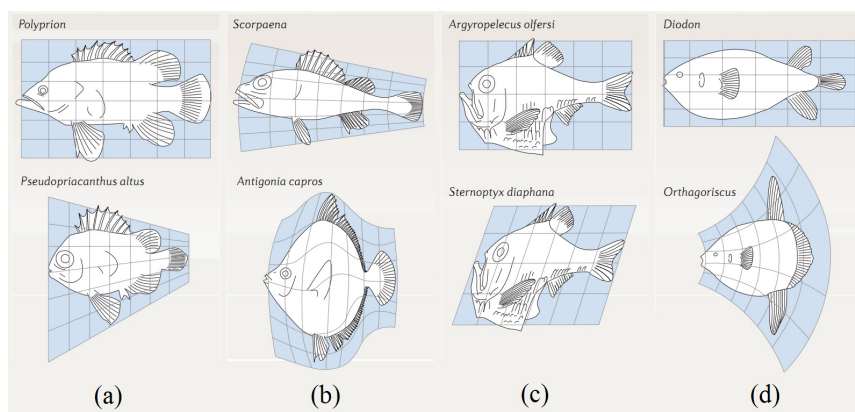


Figura 1.1: Transformação relacionando diferentes espécies de peixes. (a) *Polyprion* (acima) e *Pseudopriacanthus altus* (abaixo); (b) *Scorpaena* (acima) e *Antigonia capros* (abaixo); (c) *Argyropelecus olfersi* (acima) e *Sternoptyx diaphana* (abaixo); (d) *Diodon* (acima) e *Orthogoriscus* (abaixo). (Figura extraída do trabalho (ARTHUR, 2006))

1.1 Motivação

A obra de D'Arcy Thompson é considerada até hoje fundamental para o estudo da morfogênese (estudo da origem das formas) e permanece sem comprovação nos dias atuais, mesmo com o avanço da genética. Apesar de não comprovada, a ideia tem um potencial de exploração em tarefas de computação gráfica que será abordada nesse trabalho. Considerando que todos os desenhos e cálculos matemáticos realizados foram gerados manualmente, dado que D'Arcy Thompson não dispunha do arsenal tecnológico hoje disponível, é importante que uma exploração mais detalhada do tema seja feita. Dessa forma, é possível não só comprovar computacionalmente os resultados obtidos como também expandir o estudo, possivelmente descobrindo novas relações entre diferentes espécies.

1.2 Objetivo

O objetivo do trabalho é estudar as transformações geométricas por meio de uma implementação em computação gráfica que recebe um modelo 3D como entrada e permite modificar os pontos deste objeto globalmente, a fim de conseguir reproduzir parte dos resultados atingidos no trabalho de D'Arcy Thompson por meio da geração de um novo modelo transformado. Para validação, foi desenvolvido outro aplicativo que recebe duas imagens como entrada e aplica uma heurística simples de comparação entre elas, permitindo verificar o nível de similaridade entre os modelos escolhidos para o estudo e os objetos retratados por Thompson no modelo original. A figura 1.2 ilustra o processo completo de modelagem com a etapa de validação. Esse processo inicia-se com um modelo 3D similar (fig. 1.2(a)) ao modelo original inicial 2D de Thompson, que por meio de transformações geométricas dá origem a um novo modelo (fig. 1.2(b)), o qual deve ser semelhante ao modelo original final 2D de Thompson. Para verificar o nível de semelhança, para ambos os modelos 3D usa-se uma imagem de perfil e compara-se com fotos dos peixes propostos por Thompson (fig. 1.2(c) e fig. 1.2(d)), gerando uma figura que tem correspondência com o percentual de diferença entre ambas (fig. 1.2(e) e fig. 1.2(f)).

1.3 Organização do texto

O trabalho encontra-se dividido em cinco capítulos. O capítulo 2 aborda os trabalhos anteriores em assuntos relacionados ao tema principal, particularmente morfologia e modelagem geométrica em duas e três dimensões. O capítulo 3 apresenta o desenvolvimento da implementação e o capítulo 4 mostra os resultados obtidos a partir de diferentes objetos. Por fim, o capítulo 5 refere-se às conclusões obtidas e ideias para trabalhos futuros.

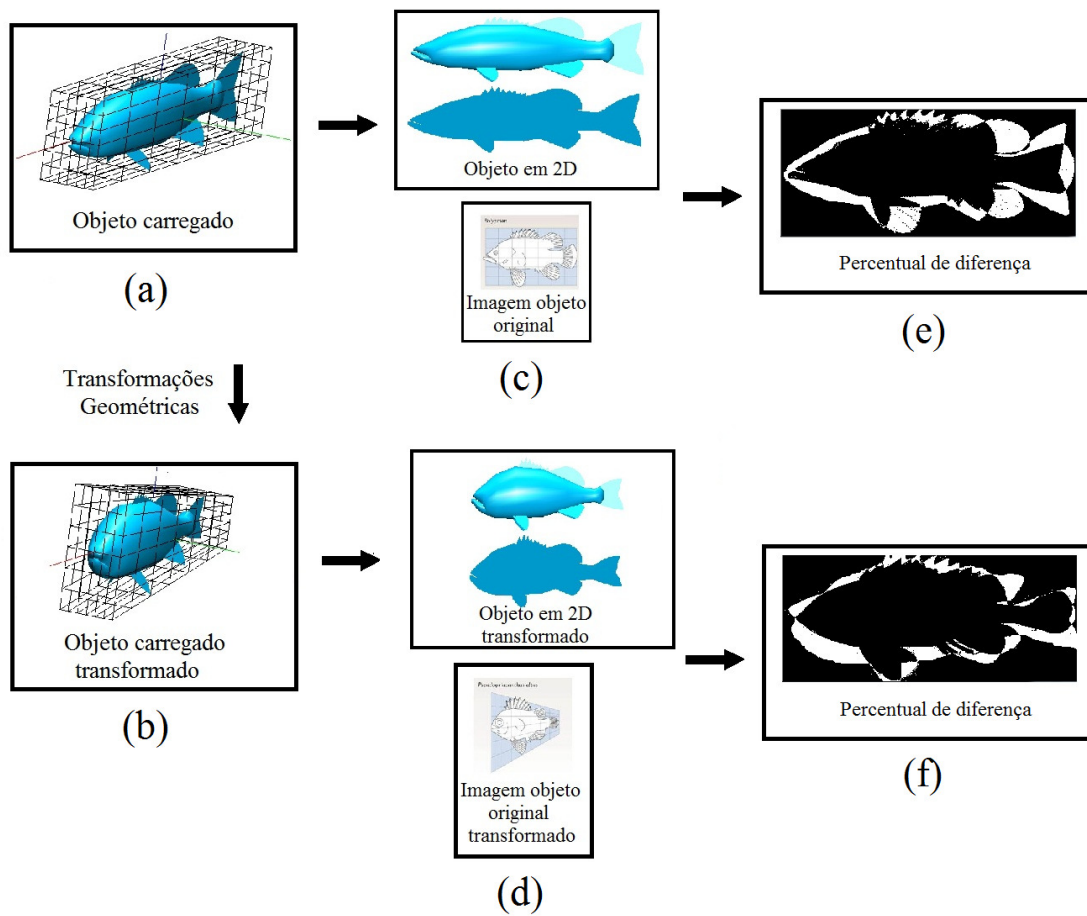


Figura 1.2: Diagrama de visualização do processo. (a) Modelo inicial; (b) Modelo inicial transformado; (c) Visão 2D lateral de (a) e do peixe original equivalente de Thompson; (d) Visão 2D lateral de (b) e do peixe original equivalente de Thompson; (e) Percentual de diferença entre as figuras de (c); (f) Percentual de diferença entre as figuras de (d).

2 REVISÃO BIBLIOGRÁFICA

A ideia original de Thompson aborda diversos aspectos de trabalhos já desenvolvidos e que se relacionam com o assunto geral. O contexto geral é de modelagem geométrica e especificamente de transformações entre formas. Abaixo apresentamos uma revisão bibliográfica sobre esses assuntos no contexto do trabalho.

2.1 Modelagem Geométrica

De acordo com (MORTENSEN, 1985), modelagem geométrica é uma coleção de métodos matemáticos interrelacionados usados para descrever a forma de um objeto. Seu uso é fundamental para que testes e análises sejam feitos primeiramente em um modelo, sem precisar aplicar os processos desejados diretamente em um objeto real. Dessa forma é possível evitar que eventuais problemas que possam danificar o objeto verdadeiro ocorram, além de não ser necessária a construção ou a manutenção do objeto físico.

Três categorias fundamentais de aplicação são apresentadas, sendo que todas estão relacionadas entre si. A primeira consiste na representação de um objeto físico, havendo a necessidade de criar o modelo uma única vez. A segunda envolve a criação de um objeto com o propósito de estudá-lo até que o objeto físico seja criado, seja pelo tempo, dificuldade ou até mesmo recursos requeridos para seu desenvolvimento. O último requisito apontado é o rendering, precisando de uma cópia do modelo para avaliá-lo.

Uma técnica ilustrativa do uso da modelagem geométrica é a *computer-aided manufacturing*, que consiste no controle e planejamento das produções de manufaturas utilizando um sistema computacional, lançando mão de um modelo. Outro exemplo de sua abrangência é em realidade aumentada, por meio do uso de modelos sobrepostos no mundo real a fim de acrescentar informações a ele.

A utilização de modelos em Computação Gráfica destaca-se como um dos principais tópicos na área. Para o desenvolvimento das mais variadas aplicações gráficas, a necessidade de bons modelos 3D surge como fator inicial fundamental. Dessa forma, a flexibilidade da variação das espécies obtida por meio de transformações exploradas por Thompson relaciona-se à busca de diferentes métodos para a alteração de formas a fim de gerar objetos visualmente semelhantes mas ao mesmo tempo diferentes.

2.2 Métodos para Representação da Forma em Computação Gráfica

Uma forma é concretizada por um modelo ou representação. Portanto podemos falar sobre as primitivas que, em última instância, representam uma determinada forma. Para objetos definidos no espaço tridimensional, podemos distinguir entre primitivas baseadas

em superfície e volume. A definição da primitiva adequada depende da aplicação. Para esse trabalho os objetos que usaremos serão convenientemente descritos por malhas poligonais. Numa malha poligonal a forma é representada por uma coleção de polígonos. Um polígono é descrito por uma coleção fechada de arestas e cada aresta é compartilhada por exatamente dois polígonos. Arestas são segmentos de linha reta que conectam dois vértices. Um vértice é um ponto no espaço. Um polígono conveniente para a maioria das aplicações é o triângulo, uma vez que qualquer polígono pode ser expresso como uma coleção de triângulos e a maioria dos hardwares gráficos tem sistemas dedicados para processamento de triângulos. A principal desvantagem dos triângulos é o grande número de triângulos necessários para representar formas intrincadas ou mais complexas. Além disso, o número necessário de polígonos pode ser arbitrariamente grande dependendo da exatidão necessária da aproximação geométrica da forma.

2.3 Morphing 2D

O termo *morphing* refere-se ao processo de transição entre um objeto inicial e um final. Derivado a partir do termo metamorfose, no contexto 2D representa uma sequência de imagens que representam estados intermediários de transformação entre duas imagens fornecidas. O objetivo da operação é apresentar uma mudança suave entre as figuras, fazendo com que a transição flua de forma natural. Sua utilização ocorre principalmente na indústria cinematográfica, englobando cenas de filmes que requerem efeitos especiais para a transformação entre objetos.

Uma possível abordagem para a solução do problema seria modificar manualmente cada quadro a partir da imagem inicial em direção ao resultado final e vice-versa, observando quais detalhes deveriam ser modificados gradualmente na imagem até que o objetivo final fosse atingido. Essa técnica necessita ser muito bem aplicada a fim de produzir bons resultados além de consumir muito tempo. Cada caso deve ser tratado isoladamente, além de requerer um artista responsável que crie as imagens intermediárias e determine se o resultado realmente é condizente.

A primeira abordagem computacional ao problema de mudança suave entre formas foi apresentada no trabalho (BEIER; NEELY, 1992). Os autores utilizaram uma nova técnica chamada de *field morphing*, que consiste em mapear pixel a pixel a imagem final para a correspondente posição da imagem inicial. Para determinar o local correto, foi sugerida inicialmente a transformação com um par de linhas, onde um vetor deve ser especificado para cada imagem, gerando assim um mapeamento dos pixels entre elas. Dessa forma foi possível realizar operações de rotação, translação e escala em um eixo, conforme mostrado na figura 2.1(a), onde uma letra “F” sofre as transformações descritas usando apenas um vetor. Transformações que exigem mudanças em ambas as direções, como a operação de deslizamento (*shear*), não são possíveis uma vez que requerem o uso de outro vetor a fim de controlar ambos os eixos. No mesmo trabalho, uma nova transformação foi apresentada, porém com múltiplos pares de linhas, adicionando essa flexibilidade necessitada, dado que agora os pixels serão mais influenciados quanto mais próximos estiverem dos vetores, afetando globalmente a imagem, ainda que com menos intensidade, modificando a letra “F” agora usando dois vetores, conforme visto na figura 2.1(b).

Essa técnica foi também aplicada para vídeos, bastando selecionar *frames* importantes dentre a sequência de imagens e aplicar o algoritmo, produzindo resultados interessantes. Uma aplicação comercial desta técnica foi no *clip* musical *Black or White* de Michael

Jackson, com ótimas finalizações vistas na figura 2.2. Neste caso, a partir de duas imagens distintas (fig. 2.2(a) e fig. 2.2(b)), desejava-se obter diferentes estados de transições durante a metamorfose entre as imagens (fig. 2.2(c) e fig. 2.2(d)), até o ponto intermediário entre elas (fig. 2.2(e)). A técnica aumentou a expressividade em comparação com outros trabalhos pois permitiu que o artista controlasse exatamente quais pontos eram importantes de serem mapeados. Por outro lado, a técnica exige muito tempo do usuário, que precisa selecionar as linhas manualmente em cada imagem além da falta de controle gerada pela interpolação dos pixels entre as linhas definidas.

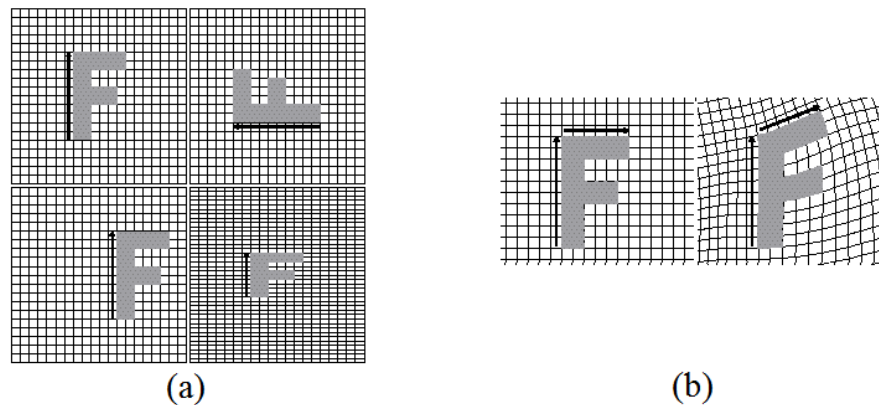


Figura 2.1: (a) Transformação com um par de linhas; (b) Transformação com múltiplos pares de linhas. (Figuras extraídas do trabalho (BEIER; NEELY, 1992))

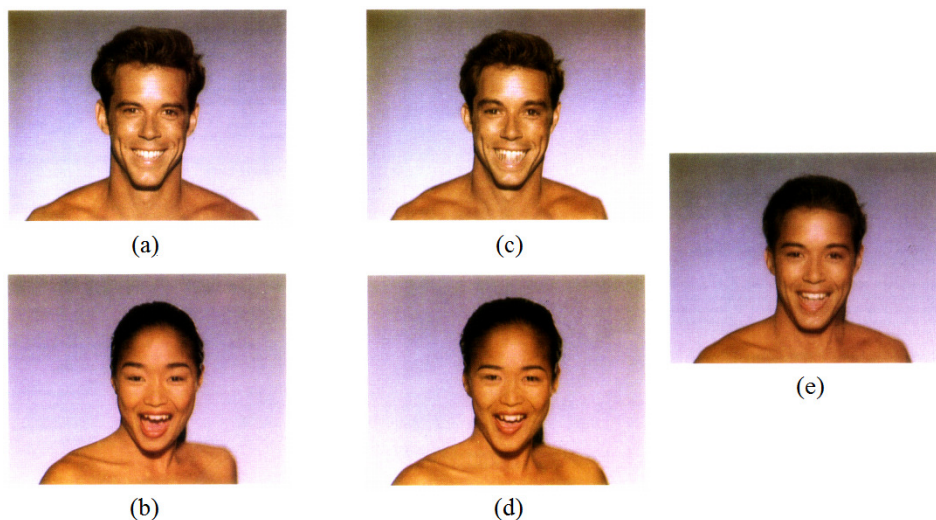


Figura 2.2: Imagens do clip musical de vídeo do Michael Jackson. (a) Primeira imagem a ser transformada na segunda; (b) Segunda imagem a ser transformada na primeira; (c) Transformação intermediária da primeira para a segunda; (d) Transformação intermediária da segunda para a primeira; (e) Resultado intermediário entre as duas imagens. (Figura extraída do trabalho de (BEIER; NEELY, 1992))

Mais recentemente a técnica apresentada em (ALEXA; COHEN-OR; LEVIN, 2000) altera a forma do objeto através de mudanças também no interior em vez de alterações apenas nos limites externos do objeto. Isso se torna possível por meio da divisão do

objeto em triângulos (2D) e tetraedros (3D) dados os objetos com suas fronteiras já destacadas, visando portanto a obtenção das imagens intermediárias de transformação entre os objetos. A partir daí é utilizado um método para possibilitar a interpolação local de cada um dos vértices correspondentes entre ambas as imagens. Os resultados obtidos quando comparados com interpolação linear apresentaram uma melhora considerável, já que a forma dos objetos também é levada em consideração em vez de pontos apenas. Um dos resultados pode ser visto na figura 2.3, que compara interpolação linear (linha superior) e o método novo criado (linha inferior), onde a tromba do elefante transforma-se no pescoço da girafa, sem necessidade de especificação explícita pelo usuário.

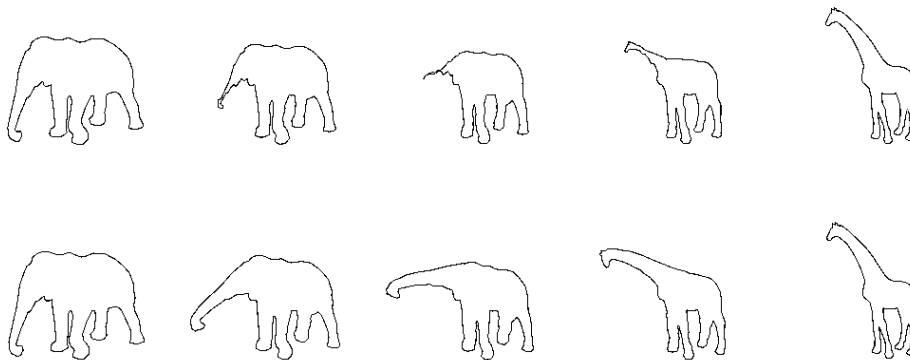


Figura 2.3: Representação de morphing de um elefante para uma girafa usando interpolação linear (primeira linha) e o método As-Rigid-As-Possible (segunda linha). (Figura extraída do trabalho de (ALEXA; COHEN-OR; LEVIN, 2000))

2.4 Deformação de Superfícies e Morphing 3D

Uma abordagem para transformação entre formas 3D são as *Free Form Deformations* introduzidas em (SEDERBERG; PARRY, 1986). A ideia básica consiste em modificar o objeto pela modificação do espaço onde ele está definido. O algoritmo consiste em englobar o objeto dividindo-o por meio de uma ou mais figuras geométricas, de acordo com a deformação requerida. As deformações podem ser aplicadas tanto global quanto localmente. Um exemplo de seu trabalho pode ser visto na figura 2.4. O método original não permite muito controle sobre as transformações pois elas são indiretamente controladas pelo usuário através da manipulação direta do *grid* 3D no qual o objeto está definido. Algumas abordagens tentaram reduzir a complexidade do controle, (CHANG; ROCKWOOD, 1994) e (HSU; HUGHES; KAUFMAN, 1992), mas essas soluções ainda não conseguem lidar com formas com muitas características complexas.

O morphing 3D objetiva a transição suave entre formas definidas tridimensionalmente, como os objetos utilizados usualmente em computação gráfica. No trabalho de (KENT; CARLSON; PARENT, 1992) uma nova forma de transformação de objetos tridimensionais é apresentada. Nessa técnica, dado um par de modelos poliédricos, o algoritmo funde as estruturas topológicas em uma rede comum de dados, possibilitando, por meio da interpolação dos vértices, as transformações entre os objetos. O algoritmo é dividido em duas etapas. Na primeira é realizado um mapeamento entre as superfícies dos objetos, ponto a ponto sobre uma esfera. A segunda consiste na criação de modelos intermediários usando interpolação entre os pontos mapeados. Um resultado encontra-se na figura 2.5.

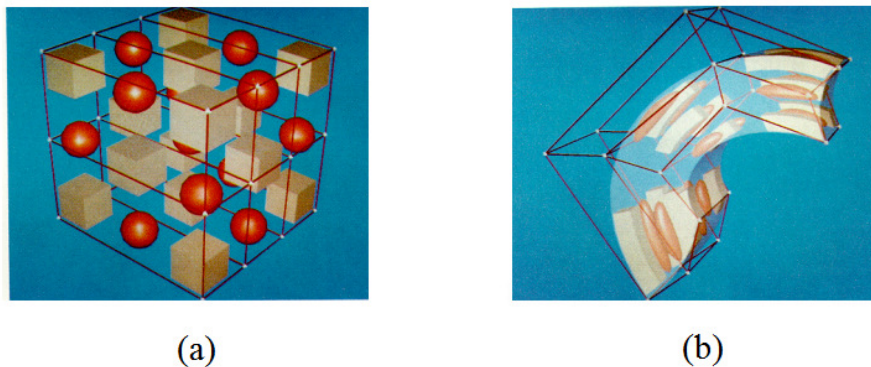


Figura 2.4: (a) Representação do objeto a ser transformado já limitado por um cubo; (b) Objeto (a) deformado. (Figuras extraídas do trabalho (SEDERBERG; PARRY, 1986))

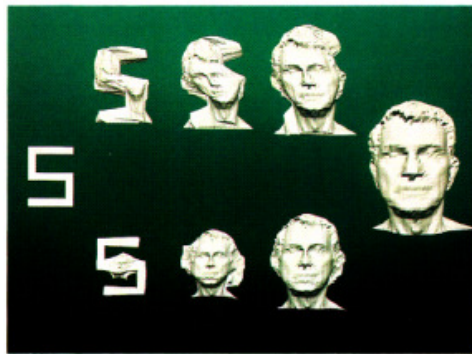


Figura 2.5: Transformação apresentando um par de objetos nas extremidades e suas etapas intermediárias. (Figura extraída do trabalho de (KENT; CARLSON; PARENT, 1992))

2.5 Comparação entre Formas

Um dos problemas ainda hoje recorrentes é a comparação entre formas. Esta técnica permite obter um resultado que indique quão semelhantes são duas formas e pode ser usada tanto com imagens quanto com modelos tridimensionais.

A comparação entre formas utilizando a distância *Hamming* assume que, dado dois polígonos sobrepostos, pode ser calculada a área de diferença entre eles, que consiste na área dentro de um dos polígonos que não se encontra no outro. A distância 0 indica que os polígonos são iguais, e a medida que o número aumenta indica que os polígonos são diferentes. Essa medida é muito utilizada para verificar a semelhança entre polígonos em estruturas computacionais, já que essa área pode ser medida em pontos discretos. A grande dificuldade, entretanto, é fazer com que as imagens tenham o mesmo alinhamento. A heurística produz um resultado muito bom considerando a simplicidade de implementação e o custo do algoritmo. No trabalho de (LANDRÉ; TRUCHETET, 2007), foi utilizada a distância *Hamming* para comparação entre imagens coloridas, colocando as características das imagens na forma binária, utilizando um amplo banco de dados para busca. Foram produzidos resultados interessantes tanto na qualidade encontrada quanto no tempo de execução. Um exemplo de resultado pode ser visto na figura 2.6.

Outro modo usado em (HUTTENLOCHER; KLANDERMAN; RUCKLIDGE, 1993) é a distância de *Hausdorff*, que identifica o ponto em um subconjunto que tem a maior

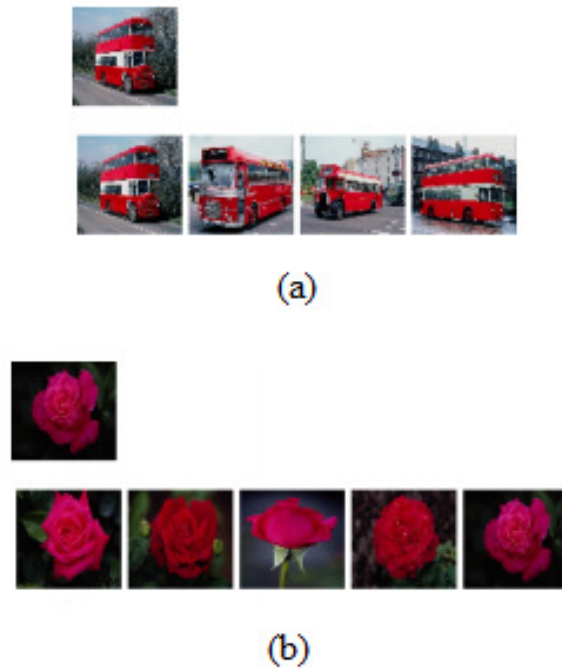


Figura 2.6: Resultados para comparação de um ônibus (a) e de uma flor (b) usando a distância *Hamming*. (Figura extraída do trabalho de (LANDRÉ; TRUCHETET, 2007))

distância entre todos os pontos de um segundo subconjunto, retornando essa distância. Assim, dados dois objetos sobrepostos um ao outro, foi possível ampliar o conceito da distância de *Hausdorff* para uma comparação entre uma imagem binária e um modelo. A translação e rotação foi um diferencial importante trabalhado no projeto, obtendo resultados interessantes de conhecimento de padrões conforme a figura 2.7. Essa distância não é comutativa.

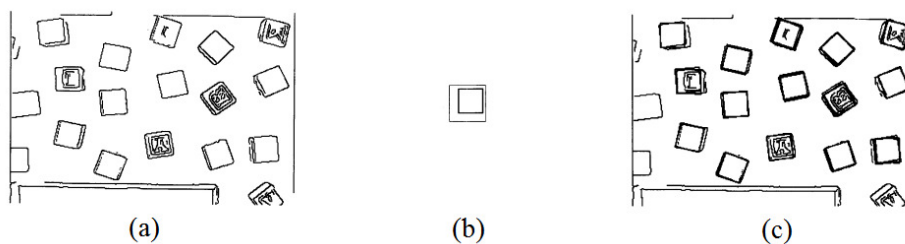


Figura 2.7: (a) Figura base para reconhecimento de estruturas semelhantes; (b) Modelo de comparação para a figura (a); (c) Figura (a) com os modelos similares a (b) destacados. (Figura extraída do trabalho de (HUTTENLOCHER; KLANDERMAN; RUCKLIDGE, 1993))

2.6 Discussão

As técnicas de *morphing* apresentadas se relacionam com o assunto principal deste trabalho, entretanto, ao contrário das técnicas de *morphing*, o nosso objetivo é a transfor-

mação em si e o resultado final, e menos a transição suave entre as formas. A abordagem para solução se inspira na transformação do espaço proposta por Sederberg e colegas (SEDERBERG; PARRY, 1986), porém com objetivos diferentes, já que visamos uma transformação global e não um conjunto de transformações locais, sem controle direto do usuário.

Entre as técnicas de comparação entre formas vistas, será utilizada a distância *Hamming* devido aos bons resultados alcançados para comparação entre imagens de mesmo tamanho alinhadas.

3 IMPLEMENTAÇÃO

Com o objetivo de validar a aplicação da teoria de D'Arcy Thompson, foi desenvolvido um sistema protótipo para testes e avaliações. Neste capítulo é apresentado o processo de desenvolvimento desta aplicação, dividida em dois módulos.

A primeira seção relaciona-se ao desenvolvimento de um software que recebe como entrada um modelo tridimensional no formato OBJ e permite a manipulação em tempo real do objeto carregado a fim de alterar suas coordenadas com equações matemáticas, gerando portanto um novo modelo 3D. A segunda seção consiste em uma aplicação que recebe duas imagens de mesmo tamanho já destacadas do fundo e devolve outras imagens e dados relacionados às figuras a fim de gerar um critério de validação na comparação de duas imagens.

3.1 Módulo de Modelagem

Nesta seção são apresentados os elementos do módulo de modelagem, que consiste na deformação de um modelo 3D a fim de obter novas formas. Para o escopo deste trabalho, são estudadas principalmente deformações entre animais, semelhantes às apresentadas no trabalho original, a fim de obter uma mudança similar à realizada por Thompson. A partir do executável, é necessário ter um formato OBJ simples, sem informações adicionais sobre texturas e outros arquivos complementares, que é carregado como parâmetro ao iniciar o aplicativo. A figura 3.1 apresenta uma síntese do processo, em que dado um modelo inicial (fig. 3.1(a)), é gerado um novo objeto (fig. 3.1(b)) por meio de transformações geométricas.

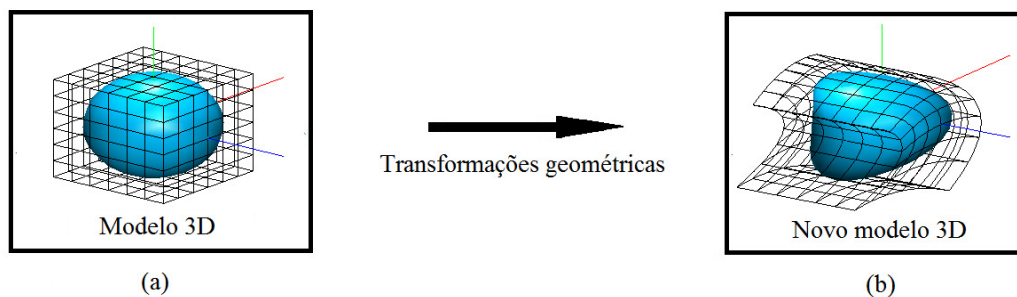


Figura 3.1: Diagrama do módulo de modelagem. (a) Objeto inicial carregado pelo programa; (b) Exemplo de transformação de (a), gerando um novo objeto.

3.1.1 Funcionalidades

Ao iniciar o programa observa-se o modelo geométrico carregado ocupando o lado direito da janela e o menu de interação do lado esquerdo, conforme ilustrado na figura 3.2. O objeto é exibido no lado direito com projeção ortogonal, a fim de preservar a forma do modelo, além das funções de iluminação básicas providas pela OpenGL e a cor num tom de azul uma vez que texturas não são carregadas. Além disso, o objeto sofre uma translação para a origem e mudança de escala de acordo com a função *glmUtilize* da biblioteca GLM.

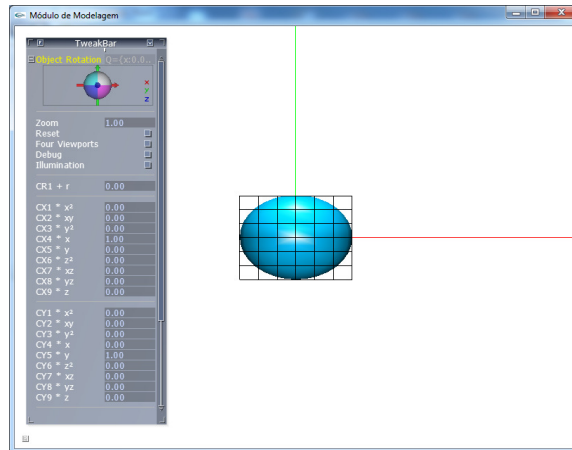


Figura 3.2: Tela inicial do aplicativo ao carregar o modelo.

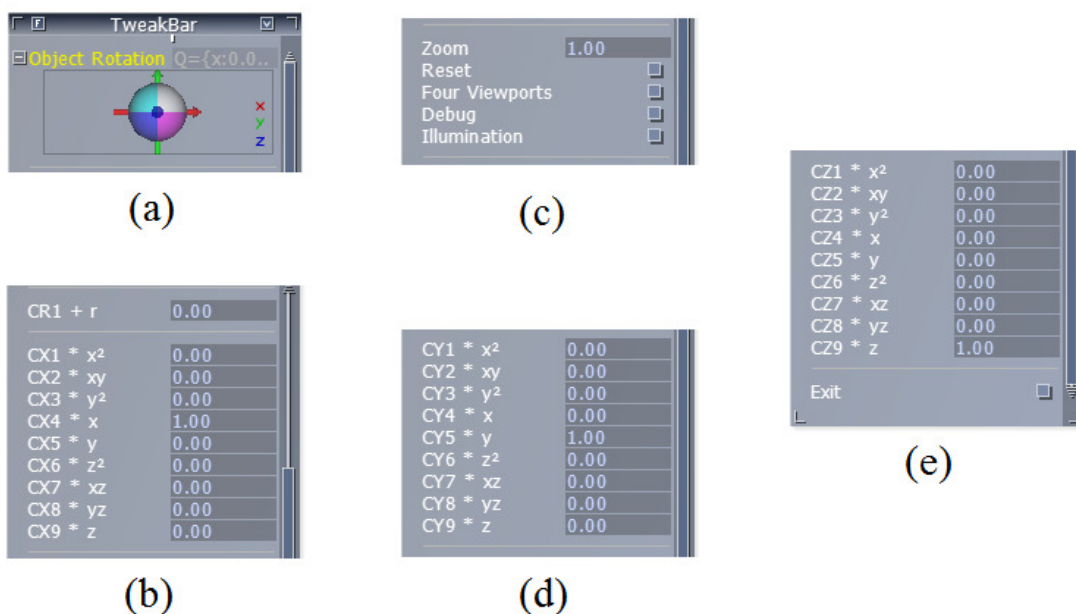


Figura 3.3: Menu de interação detalhado. (a) Trackball; (b) Coeficiente r correspondente ao raio das coordenadas polares juntamente com os coeficientes do polinômio de segundo grau para o eixo x ; (c) Funcionalidades *Zoom*, *Reset*, *Four Viewports*, *Debug* e *Illumination*; (d) Coeficientes do polinômio de segundo grau para o eixo y ; (e) Coeficientes do polinômio de segundo grau para o eixo z juntamente com o botão *Exit*.

O menu é o único meio de interação do aplicativo e está representado detalhadamente

na figura 3.3. No topo da interface encontra-se a *trackball* (CHEN; MOUNTFORD; SELLEN, 1988) (fig. 3.3(a)), a qual permite rotacionar o objeto como se estivesse englobado em uma esfera por meio do mouse. Logo abaixo é disponibilizado um botão para controle de afastamento ou aproximação da câmera (*Zoom*). Em seguida há a opção para reinicialização do objeto, conforme primeiramente carregado (*Reset*), que altera todos os parâmetros para o valor inicial.

Logo abaixo dá-se continuidade às demais funcionalidades. A primeira divide a câmera em quatro ângulos distintos (*Four Viewports*) para possibilitar uma melhor observação do objeto de diferentes perspectivas. Todas as quatro câmeras apresentam visão ortogonal e as coordenadas da câmera original são mantidas na janela do canto inferior direito, no qual o objeto mantém a cor padrão, diferenciando-se das demais.

Há também um botão que, quando ativado, inclui ferramentas de apoio para a melhor visualização do objeto (*Debug*). Essas ferramentas envolvem uma grade em torno do objeto e quatro vetores representando os eixos x , y e z com cores distintas, correspondentes às oferecidas pela *trackball*. A primeira ferramenta adapta-se diretamente sobre qualquer formato e tamanho de objeto e foi criada levando em conta as oito coordenadas mínimas e máximas de cada eixo, formando portanto um paralelepípedo justo ao modelo. Como se não bastasse, ainda possibilita controlar o número de linhas que o *grid* terá e o número de pontos de cada linha, uma vez que são constituídas por pequenas linhas que ligam cada par de pontos dentre os fornecidos.

Mais à frente observa-se o botão *Illumination*, que trata de retirar ou colocar a iluminação padrão do OpenGL, tornando o objeto da cor original inicialmente ajustada seguido do botão *Exit*, que finaliza a aplicação.

Por fim são apresentados os botões para alteração dos parâmetros do polinômio do segundo grau (cx_i , cy_i e cz_i , com i variando de 1 a 9) e de coordenadas polares (cr_1). As subseções seguintes detalham a função e o uso dos parâmetros.

3.1.2 Transformações do Espaço

A ideia principal apresentada por D'Arcy Thompson em seu trabalho original consiste na obtenção de novas formas de seres vivos a partir da modificação do sistema de coordenadas de um organismo original. D'Arcy Thompson não apresentou em seu trabalho equações precisas que detalhassem as transformações envolvidas. O livro apresenta apenas as figuras das transformações, além de algumas noções gerais da matemática utilizada. Para esse trabalho exploramos duas possibilidades: polinômios de segundo grau e coordenadas polares, explicadas a seguir.

3.1.3 Polinômio do Segundo Grau com Três Variáveis

Dentre as diversas possibilidades de alteração do espaço, a utilização de um polinômio de segundo grau garante simplicidade pelo baixo grau e transformações mais interessantes do que as lineares. Esse tipo de modificação foi também utilizado em um trabalho¹ para transformações entre formas 2D. Para o desenvolvimento do trabalho corrente foi utilizado o mesmo polinômio, porém estendido ao espaço tridimensional. Dessa forma são utilizados três polinômios de segundo grau com três variáveis, um para cada dimensão x , y e z . A ideia aproveita-se da flexibilidade gerada pelos diferentes parâmetros e simplicidade da adaptação do polinômio às coordenadas do modelo. Os coeficientes podem ser alterados diretamente pela interface, conforme mencionado acima.

¹<http://www-groups.dcs.st-and.ac.uk/history/Miscellaneous/darcy.html>

O polinômio padrão com duas variáveis para o espaço 2D pode ser definido por

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + F$$

onde A, B, C, D, E e F são os coeficientes e x e y as variáveis.

Estendendo o polinômio para o espaço tridimensional, obtém-se

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + Fz^2 + Gxz + Hyz + Iz + J$$

onde $A, B, C, D, E, F, G, H, I$ e J são os coeficientes e x, y e z as variáveis.

Este polinômio é aplicado para cada uma das dimensões em que o objeto é definido. Observa-se que os termos independentes F e J , na primeira e segunda equação respectivamente, podem ser desprezados dado que promovem apenas a translação do objeto no eixo respectivo, mostrando-se desnecessários para os objetivos do trabalho. Desta forma reduz-se o número de coeficientes para nove. Na implementação a notação dos coeficientes foi alterada para cx_i, cy_i e cz_i , com i variando de 1 a 9, para facilitar a correspondência entre cada coeficiente e sua respectiva dimensão. Nos exemplos, para as dimensões x, y e z , os coeficientes D, E e I , respectivamente, começam com valor um. Todos os demais, se não modificados, possuem valor zero para todas as dimensões. Alguns exemplos das variações de acordo com os coeficientes do polinômio podem ser vistos nas figuras 3.4 e 3.5.

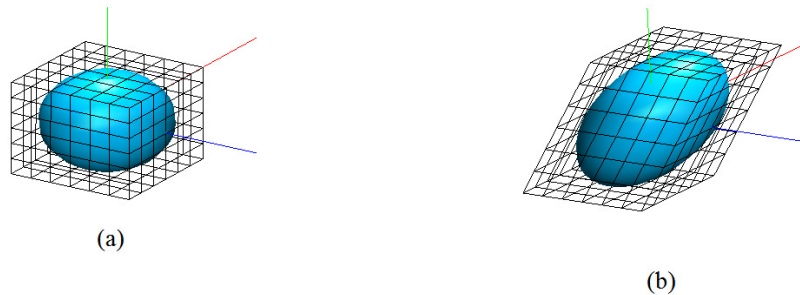


Figura 3.4: (a) Modelo inicial; (b) Transformação a partir de (a) na dimensão x com coeficiente $E = 1.0$.

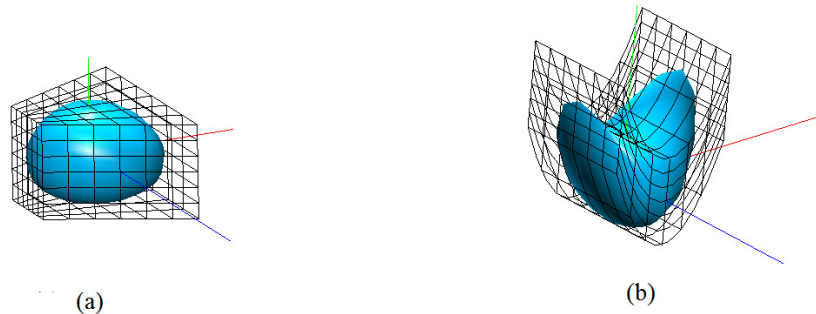


Figura 3.5: (a) Transformação a partir da figura 3.4(a) na dimensão z com coeficiente $G = 0.5$; (b) Transformação a partir da figura 3.4(a) na dimensão y com coeficiente $A = 2.0$.

Após o carregamento do objeto num *array vertices* de n posições (onde n é o número total de vértices), temos acesso a todos os vértices que definem a malha poligonal. Cada

sequência de três em três valores armazenados no *array* corresponde a um vértice, formado pelos eixos x , y e z , respectivamente. Os valores de x , y e z de cada vértice podem ser obtidos a cada laço de repetição de $vertices[i + j]$, com i variando de 0 a $(n - 1)$ e j variando de 0 a 2. Assim, cada valor dos três possíveis para a variável j representa, respectivamente, os eixos x , y e z de um vértice, representado pela variável i .

A fim de incluir o polinômio no cálculo de cada ponto, logo após obter o *array* *vertices* original, é criada uma cópia chamada *old_vertices*, de mesmo tamanho que contém um backup dos pontos do modelo. Podemos agora alterar diretamente a variável *vertices*, modelam-se as coordenadas em função do polinômio por meio de

$$\begin{aligned}
vertices[i + 0] = & cx_1 * negX * (old_vertices[i + 0])^2 \\
& + cx_2 * (old_vertices[i + 0]) * (old_vertices[i + 2]) \\
& + cx_3 * (old_vertices[i + 1])^2 \\
& + cx_4 * (old_vertices[i + 0]) \\
& + cx_5 * (old_vertices[i + 1]) \\
& + cx_6 * (old_vertices[i + 2])^2 \\
& + cx_7 * (old_vertices[i + 0]) * (old_vertices[i + 2]) \\
& + cx_8 * (old_vertices[i + 1]) * (old_vertices[i + 2]) \\
& + cx_9 * (old_vertices[i + 2])
\end{aligned}$$

$$\begin{aligned}
vertices[i + 1] = & cy_1 * (old_vertices[i + 0])^2 \\
& + cy_2 * (old_vertices[i + 0]) * (old_vertices[i + 2]) \\
& + cy_3 * negY * (old_vertices[i + 1])^2 \\
& + cy_4 * (old_vertices[i + 0]) \\
& + cy_5 * (old_vertices[i + 1]) \\
& + cy_6 * (old_vertices[i + 2])^2 \\
& + cy_7 * (old_vertices[i + 0]) * (old_vertices[i + 2]) \\
& + cy_8 * (old_vertices[i + 1]) * (old_vertices[i + 2]) \\
& + cy_9 * (old_vertices[i + 2])
\end{aligned}$$

$$\begin{aligned}
vertices[i + 2] = & cz_1 * (old_vertices[i + 0])^2 \\
& + cz_2 * (old_vertices[i + 0]) * (old_vertices[i + 2]) \\
& + cz_3 * (old_vertices[i + 1])^2 \\
& + cz_4 * (old_vertices[i + 0]) \\
& + cz_5 * (old_vertices[i + 1]) \\
& + cz_6 * negZ * (old_vertices[i + 2])^2 \\
& + cz_7 * (old_vertices[i + 0]) * (old_vertices[i + 2]) \\
& + cz_8 * (old_vertices[i + 1]) * (old_vertices[i + 2]) \\
& + cz_9 * (old_vertices[i + 2])
\end{aligned}$$

com i variando de 0 a n e incrementando 3 unidades a cada laço de repetição.

Após testes iniciais, foi observado que muitos dos parâmetros não eram aproveitados corretamente devido ao objeto estar localizado na origem, englobando, portanto, todos os oito octantes possíveis no espaço 3D, sofrendo variações relacionadas aos eixos negativos de cada dimensão. Foram acrescentadas então novas variáveis $negX$, $negY$ e $negZ$ que podem assumir valor -1, caso o termo quadrático da respectiva dimensão seja negativo ou valor 1 caso contrário, multiplicando o termo quadrático do respectivo polinômio. Por exemplo, ao realizar uma transformação com o termo independente de x^2 sobre o eixo x (fig. 3.6(a)), a parte negativa do objeto torna-se positiva, ampliando quadraticamente a parte que era positiva corretamente, porém espelhando para o lado positivo a parte que até então era negativa, produzindo um resultado que nada acrescenta à deformação do objeto.

Como correção, é alterado o valor final que sempre é positivo para negativo, caso apenas um dos termos seja inferior a zero, permitindo que o objeto varie quadraticamente em relação ao eixo desejado (fig. 3.6(b)).

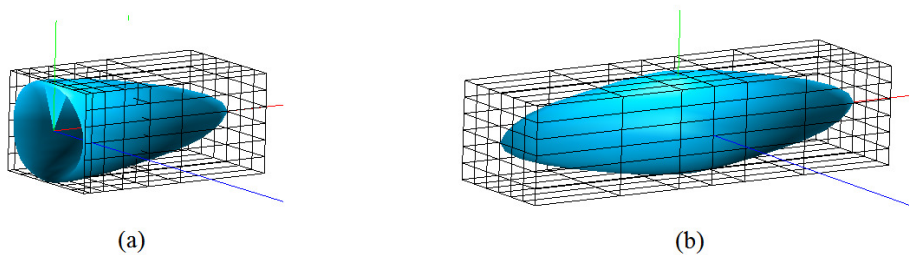


Figura 3.6: (a) Transformação a partir da figura 3.4(a) na dimensão x com coeficiente $A = 2.0$ e variáveis neg desativadas; (b) Transformação a partir da figura 3.4(a) na dimensão x com coeficiente $A = 2.0$ e variáveis neg ativadas.

3.1.4 Coordenadas Polares

Uma outra modificação 2D, mencionada como possível por D'Arcy Thompson, consiste na mudança das coordenadas cartesianas originais para coordenadas polares seguida de transformações nas próprias coordenadas polares. O trabalho original cita e até mesmo esboça alguns resultados produzidos utilizando coordenadas polares, porém com ênfase

em modelos da flora e animais que apresentavam algum padrão evidente de simetria. Assim, para o desenvolvimento do trabalho, foi considerado o uso de coordenadas polares de modo a expandir as possibilidades de deformações nos modelos, porém foi aplicado apenas para o modo 2D, de forma a evitar a utilização de coordenadas esféricas. O coeficiente pode ser alterado diretamente pela interface.

A conversão de coordenadas cartesianas para polares pode ser efetuada por meio de:

$$r = \sqrt{x^2 + y^2}$$

$$\theta = \begin{cases} \arctan(y/x) & \text{se } x > 0 \\ \arctan(y/x) + \pi & \text{se } x < 0 \text{ e } y \geq 0 \\ \arctan(y/x) - \pi & \text{se } x < 0 \text{ e } y < 0 \\ \pi/2 & \text{se } x = 0 \text{ e } y > 0 \\ -\pi/2 & \text{se } x = 0 \text{ e } y < 0 \end{cases}$$

igualmente podemos converter de coordenadas polares para coordenadas cartesianas com:

$$x = r \cos(\theta)$$

$$y = r \sin(\theta)$$

A transformação do modelo situa-se no mesmo laço iterativo do processo descrito anteriormente, isto é, utilizando o *array vertices* e a variável inteira *i* com os respectivos parâmetros do comando. Entretanto, o plano *z* não faz parte da transformação.

O trecho abaixo realiza a conversão de cada vértice para seu correspondente valor em coordenadas polares, soma o valor da variável *r* a uma constante representada por *cr1* e logo em seguida converte novamente para coordenadas cartesianas

$$r = \sqrt{(\text{vertices}[i + 0])^2 + (\text{vertices}[i + 1])^2}$$

$$\text{theta} = \text{atan2}(\text{vertices}[i + 0], \text{vertices}[i + 1])$$

$$r = r + \text{cr1}$$

$$\text{vertices}[i + 0] = r * \cos(\text{theta})$$

$$\text{vertices}[i + 1] = r * \sin(\text{theta})$$

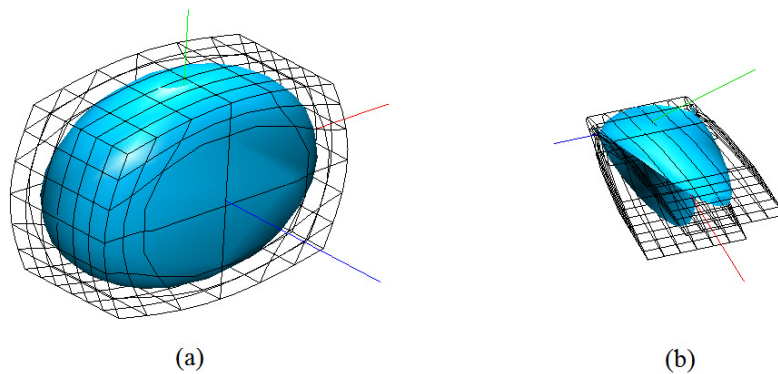


Figura 3.7: (a) Transformação a partir da figura 3.4(a) com coeficiente $cr1 = 1.2$; (b) Transformação a partir da figura 3.4(a) nas dimensões x , x e y com coeficientes $C = 2.0$, $D = 0.5$ e $E = 0.5$, respectivamente, e $cr1 = -0.15$.

onde a variável r representa o raio e $theta$ o ângulo θ . Além disso foi utilizada a função $atan2$ da biblioteca $math.h$, que recebe x e y como parâmetros, convertendo de acordo com a equação e verificando as restrições. Não foi implementada uma modificação quanto à variável θ , uma vez que seu uso não foi fundamental para a obtenção de resultados positivos. Um exemplo de transformação com o coeficiente $cr1$ está na figura 3.7.

3.2 Módulo de Validação

Esta seção trata sobre o módulo de validação, que consiste no desenvolvimento de um critério de validação para relacionar as imagens extraídas do trabalho de D'Arcy Thompson com os modelos tridimensionais obtidos. O processo pode ser resumido pela figura 3.8, onde dadas duas imagens iniciais (fig. 3.8(a)), é realizada uma segmentação objetivando destacar o objeto desejado do plano de fundo para ambas as imagens iniciais, gerando duas novas imagens (fig. 3.8(b)) e comparando-as por meio da distância *Hamming*, produzindo um resultado (fig. 3.8(c)) juntamente com um valor, chamado de percentual de diferença.

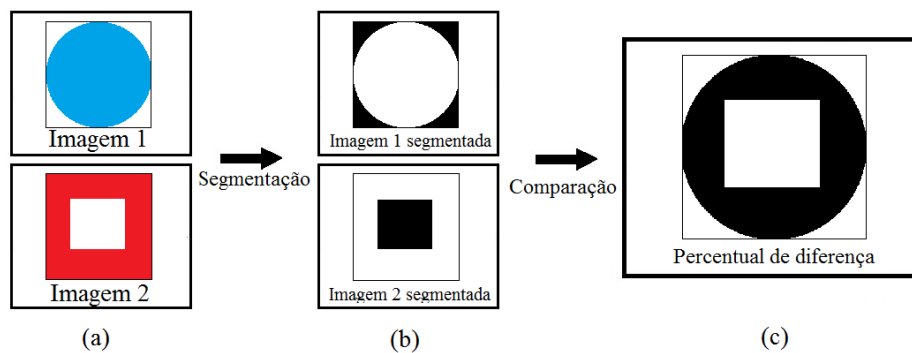


Figura 3.8: Diagrama geral de exemplo do módulo de validação.

O critério de validação será utilizado para comparar as formas obtidas com as formas apresentadas no trabalho original. Além disso, considerando que não foi possível encontrar modelos 3D exatamente das mesmas espécies do trabalho original, utilizaremos modelos aproximados aos originais. Ou seja, o ponto de partida das transformações já começa com uma forma que não é a mesma proposta por D'Arcy Thompson, com erro inicial, pode-se dizer assim, associado.

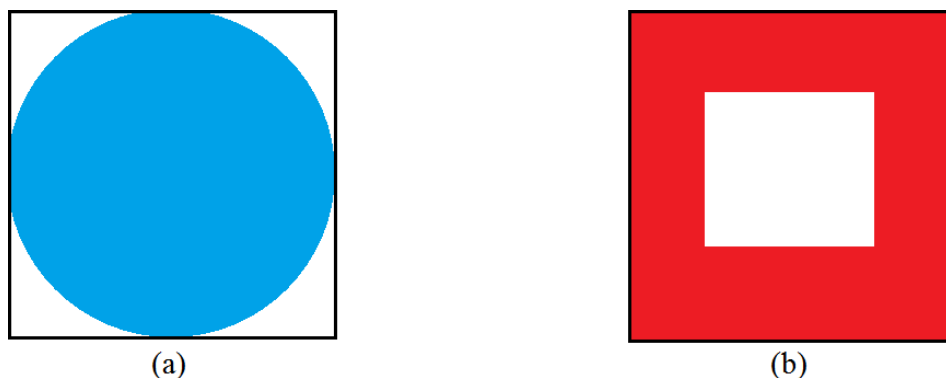


Figura 3.9: (a) Imagem de entrada 1; (b) Imagem de entrada 2.

Sobre o programa, considera-se que *img1* e *img2* são dois *arrays* que já contêm os valores dos pixels das imagens carregadas, assumindo que cada pixel dessas imagens possui três canais no formato RGB e ambas as imagens apresentam mesma resolução, três canais, fundo claro e a figura destacada em cor escura. As figuras ilustradas nessa seção baseiam-se nas duas imagens de entrada observadas na figura 3.9. Os valores de *x*, *y* e *z* de cada vértice podem ser obtidos a cada laço de repetição de *img[i + j]*, com *i* variando de 0 a (*n* - 1) e *j* variando de 0 a 2. Assume-se *img* como qualquer uma das imagens acima descritas, *i* a variável responsável por percorrer todos os pixels e *j* cada um dos canais RGB.

Lançando mão do mesmo laço para acesso dos pixels, cada um deles é ajustado para um novo valor (branco ou preto) de acordo com o antigo, respeitando um *limite* estipulado, conforme visto na figura 3.10. Essa técnica é chamada de *limiarização* em processamento de imagens.

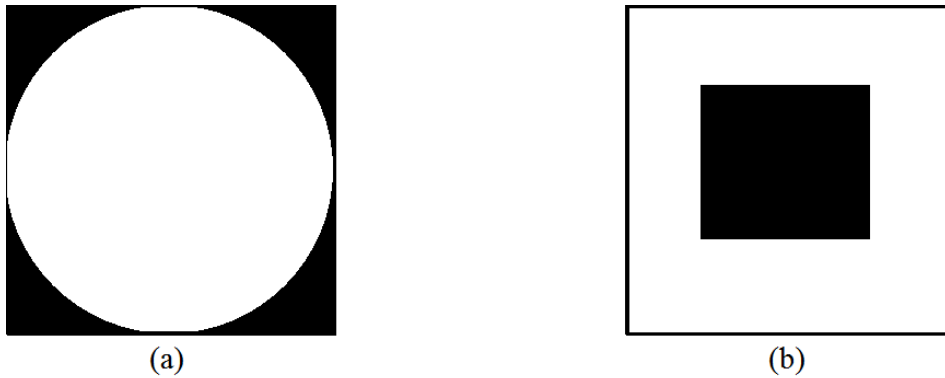


Figura 3.10: (a) Imagem 1 segmentada por limite definido; (b) Imagem 2 segmentada por limite definido.

$$seg_img[i + j] = \begin{cases} 0 & \text{se } img[i + j] > limite \\ 255 & \text{se } img[i + j] \leq limite \end{cases}$$

Desse modo, todos os pixels que forem menores que o valor imposto por *limite* serão brancos e os demais pretos, invertendo os tons para facilitar a visualização e unificando os grupos de pixels com um mesmo valor, tornando possível eliminar eventuais variações ou ruídos presentes nas imagens dos arquivos originais. Nota-se também que a todos os canais são atribuídos o mesmo valor.

Ao realizar a operação acima mencionada com *img1* e *img2*, são geradas duas novas imagens binárias *seg_img1* e *seg_img2*, ou seja, ambas segmentadas com apenas duas cores. É possível visualizar então todos os objetos a serem salientados em branco, e o fundo em preto.

Em seguida, são calculadas novas imagens que contêm a união e a intersecção dos pixels destacados (em branco) para ambas as imagens já segmentadas (fig. 3.11), chamadas de *uniao_img* e *intersec_img*, novamente com o laço iterativo apresentado para *i* e *j*.

$$uniao_img[i + j] = \begin{cases} 255 & \text{se } seg_img1[i + j] = 255 \text{ ou } seg_img2[i + j] = 255 \\ 0 & \text{caso contrário} \end{cases}$$

$$intersec_img[i + j] = \begin{cases} 255 & \text{se } seg_img1[i + j] = 255 \text{ e } seg_img2[i + j] = 255 \\ 0 & \text{caso contrário} \end{cases}$$

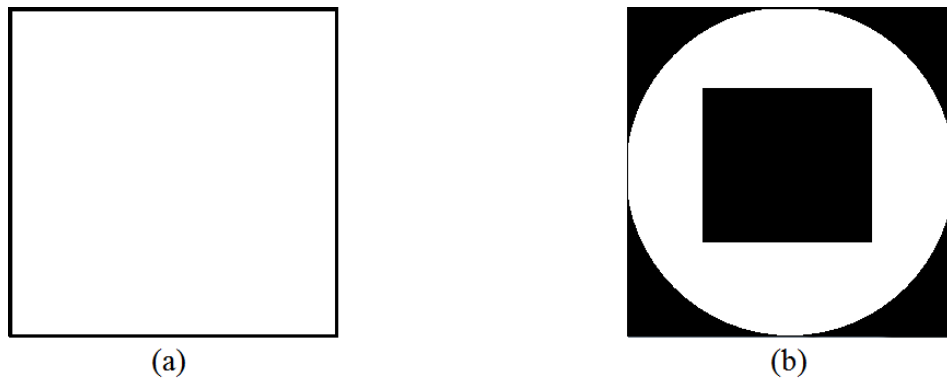


Figura 3.11: (a) União da imagem 1 com imagem 2; (b) Intersecção da imagem 1 com imagem 2.

O método de validação proposto consiste em obter uma nova imagem $un-in_img$ com o conteúdo destacado que está dentro da união, porém fora da intersecção, obtida subtraindo-se a primeira da segunda, conforme proposto pela distância Hamming. O trecho de código também está junto do mesmo laço. Um exemplo dessa imagem pode ser visto na figura 3.12.

$$un-in_img[i+j] = \begin{cases} 255 & \text{se } uniao_img[i+j] = 255 \text{ e } intersec_img2[i+j] = 0 \\ 0 & \text{caso contrário} \end{cases}$$

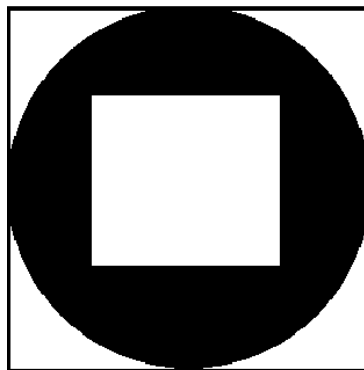


Figura 3.12: União da imagem 1 e imagem 2 menos intersecção da imagem 1 e imagem 2; O percentual de diferença encontrado foi de aproximadamente 0.45.

Com a imagem pronta, para obter a métrica de comparação basta dividir a quantidade de pixels destacados do resultado obtido de $un-in_img$ pelos pixels destacados de união, gerando um número que indica que quanto mais próximo de 0, maior é a correspondência entre os pixels destacados e quanto mais próximo de 1 menor é a correspondência.

$$\text{percentual de diferença} = \frac{(tam_img1 \cup tam_img2) - (tam_img1 \cap tam_img2)}{(tam_img1 \cup tam_img2)}$$

As operações de união e intersecção acima seguem a notação de conjuntos matemática, onde tam_img1 e tam_img2 referem-se aos tamanhos dos conjuntos, ou seja, ao número de pixels destacados em cada uma das imagens.

Uma das medidas que foi comparada à proposta apresentada foi a de criar uma sexta imagem $i2-i1_img$, que consiste simplesmente em subtrair a primeira imagem da segunda, executada sobre o laço descrito anteriormente:

$$i2-i1_img[i + j] = \begin{cases} 255 & \text{se } seg_img2[i + j] = 255 \text{ e } seg_img1[i + j] = 0 \\ 0 & \text{caso contrário} \end{cases}$$

Embora também fornecesse um resultado válido para comparação, a diferença de resultado com a simples inversão da ordem de entrada das imagens geraria resultados diferentes. Um exemplo pode ser visto na figura 3.13, em que a operação da primeira imagem sobre a segunda (fig. 3.13(a)) produz um resultado diferente da segunda sobre a primeira (fig. 3.13(b)).

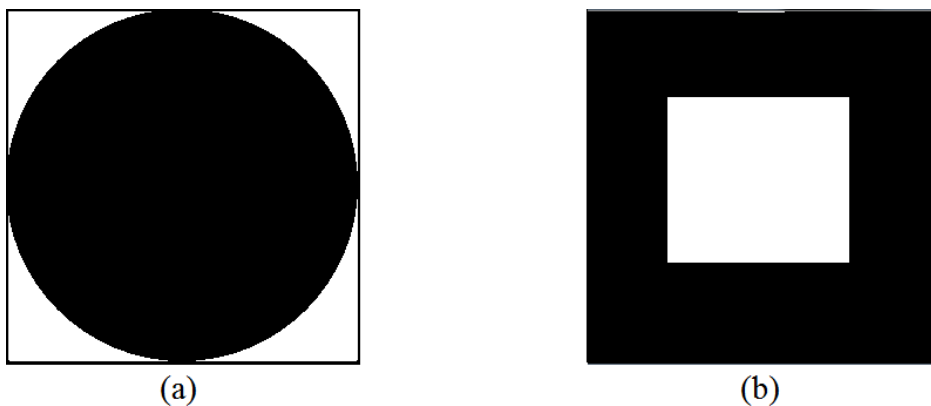


Figura 3.13: (a) Imagem 1 menos Imagem 2; (b) Imagem 2 menos Imagem 1; Os percentuais de diferença encontradas foram de aproximadamente 0.27 e 0.31, respectivamente.

3.3 Ambiente de Desenvolvimento

Para o desenvolvimento da aplicação foi utilizada a linguagem C/C++ juntamente com a IDE CodeBlocks² multiplataforma e de código aberto. O compilador usado foi o MinGW/GCC além do sistema operacional Windows 7 32 bits.

O módulo de Modelagem utiliza a API OpenGL³, as bibliotecas GLUT⁴ e GLM⁵, que apresentam funcionalidades para OpenGL. Para a interface gráfica foi utilizada a biblioteca AntTweakBar⁶. A segunda fase lança mão da biblioteca OpenCV⁷ para manipulação de imagens.

²<http://www.codeblocks.org>

³<http://www.opengl.org>

⁴<http://www.opengl.org/resources/libraries/glut>

⁵<http://www.altuma.ro/opengl/glm.zip>

⁶<http://www.antisphere.com/Wiki/tools:anttweakbar>

⁷<http://www.opencv.org>

4 RESULTADOS

Este capítulo apresenta os resultados obtidos para diferentes amostras, dividindo-se em duas partes. A primeira mostra a aplicação do trabalho sobre os exemplos utilizados por Thompson enquanto a segunda apresenta uma discussão sobre os resultados da seção anterior.

4.1 Transformações do Trabalho de Thompson

Essa seção contém diversos resultados usando ambos os módulos implementados. A primeira imagem de cada par contém o modelo tridimensional carregado juntamente com a transformação aplicada, além de seus respectivos coeficientes e as duas imagens correspondentes dos objetos usadas por Thompson. Os coeficientes D , E e I para as dimensões x , y e z , respectivamente, se não mencionados possuem valor um, enquanto os demais valor zero para todas as dimensões. Os coeficientes das transformações foram estimados por experimentação. A segunda imagem do par contém imagens 2D dos modelos iniciais e finais sem iluminação, bem como máscaras de imagens reais das espécies usadas por Thompson e calcula o percentual de diferença para diversos pares que possuem alguma relação. Todas as imagens foram padronizadas nas dimensões de 350x150 *pixels*.

O percentual de diferença é calculado para todos os tipos de imagens que podem ser comparadas, como as duas geradas a partir dos modelos 3D, as duas dos peixes reais, a imagem do modelo 3D inicial com a imagem real do peixe inicial e a imagem do modelo 3D final com a imagem real do peixe final. As formas usadas para comparação com as imagens de perfil do arquivo OBJ são fotos reais das espécies retratadas por D'Arcy Thompson ao invés dos desenhos feitos por ele. As figuras 4.1 e 4.2, 4.3 e 4.4, 4.5 e 4.6, 4.7 e 4.8, 4.9 e 4.10, 4.11 e 4.12 apresentam, aos pares, os resultados obtidos baseando-se nas transformações do trabalho de Thompson.

4.2 Discussão

Os peixes usados como modelos 3D foram os mais parecidos dentre uma baixa disponibilidade de arquivos OBJ encontrados e que fossem gratuitos. As espécies retratadas são extremamente específicas, tornando a busca ainda mais difícil. Por esse motivo foi sugerido o uso de um critério de validação, ou seja, uma medida que fosse padrão para todos os pares de imagens que pudessem ser comparados. Os resultados visualmente corresponderam às características alcançadas nos experimentos, que podem ser observados por meio da grade nas figuras. O objeto muitas vezes não se tornou tão equivalente quanto o esperado, porque apresentava diferenças entre a imagem do objeto real, também

diferente do desenho utilizado por Thompson. Assim é possível ter um resultado aproximado das alterações sofridas pelos objetos. O percentual de diferença apresenta melhores resultados quanto mais próximo de zero.

O experimento representado nas figuras 4.1 e 4.2 partiu do uso de um modelo tridimensional do próprio peixe *polyprion*, apresentando um percentual de diferença de 0.2328 sobre o item (h), que compara os peixes transformados. Dentre todos os experimentos realizados, esse foi o menor percentual encontrado para esse item. Além disso, o mesmo exemplo encontra o menor percentual de diferença (0.2219) entre o próprio modelo inicial e final (item c), mostrando que as modificações sobre o mesmo peixe usado no trabalho original tem melhor efeito do que com peixes semelhantes.

Outro resultado que obteve todos os percentuais de forma geral baixos foi o apresentado nas figuras 4.5 e 4.6, já que, mesmo revelando um erro inicial mais alto do que o caso do *polyprion*, consistia apenas de uma operação de deslizamento, não alterando tanto as imagens entre si e apresentando uma mudança bem semelhante ao trabalho original se observado pelo *grid*. Outro resultado que produziu uma mudança no *grid* muito semelhante à original e apresentou um percentual de diferença muito baixo, principalmente devido à escolha da imagem inicial foi o experimento representado nas figuras 4.11 e 4.12.

Outro ponto importante foi o uso das coordenadas polares para o caso mostrado nas figuras 4.3 e 4.4, dado que possibilitou uma mudança que gerasse o peixe *Antigonia capros*, alterando o *grid* de forma similar ao trabalho original (fig. 4.3(d)). Uma outra maneira de produzir um resultado parecido, entretanto com uma mudança diferente no *grid*, pode ser visto na figuras 4.7 e 4.8. Nesse exemplo, os pontos foram alterados de forma a aumentar a parte da frente do peixe e diminuir a parte de trás.

Um experimento que não produziu um resultado tão fiel ao original pode ser visto nas figuras 4.9 e 4.10, embora tenha apresentado bons percentuais de diferença devido a escolha de um modelo inicial parecido com o *Diodon*. Ao comparar o *grid* entre as figuras 4.9(b) e 4.9(d), nota-se que o primeiro não apresentou linhas curvas tanto superiormente quanto inferiormente conforme o segundo. Como o algoritmo é global, não havia como tornar ambos os lados curvos com direções contrárias.

Em geral a maioria dos resultados apresentou um percentual baixo de diferença, mostrando uma similaridade alta. Algumas escolhas para a busca de fotos de peixes reais não foram as melhores se comparadas com os modelos desenhados no trabalho original. Porém vale lembrar que os modelos propostos por Thompson eram desenhos, apresentando um erro inicial entre espécies originais.

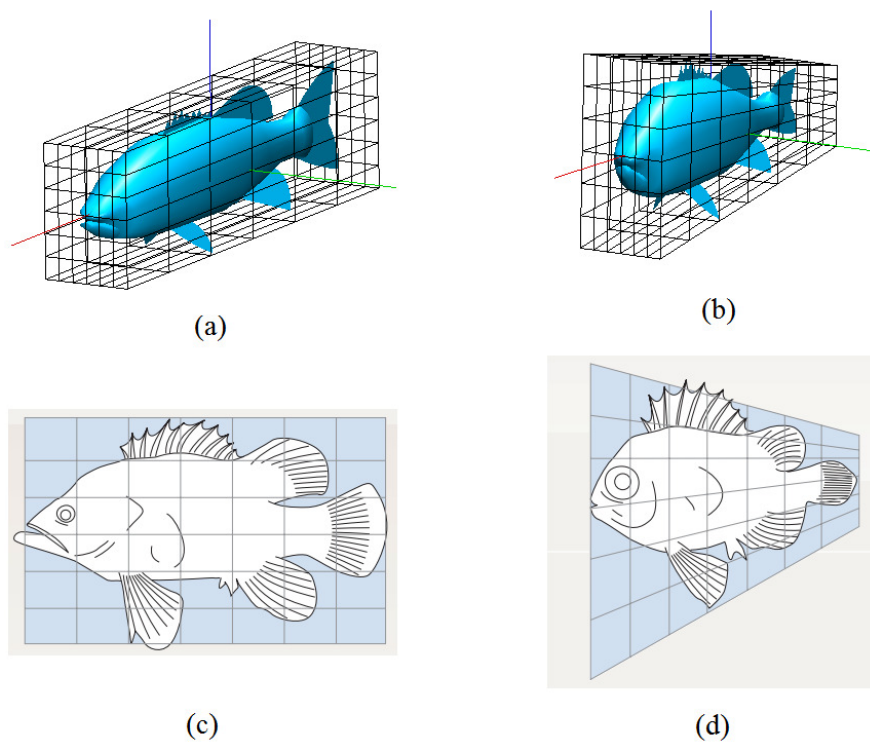


Figura 4.1: (a) Modelo inicial; (b) Transformação de (a) nas dimensões x e z , com coeficientes $D = 0.65$ e $H = 2.0$, respectivamente; (c) Imagem original do peixe *Polyprion* de Thompson; (d) Imagem original do peixe *Pseudopriacanthus altus* de Thompson.

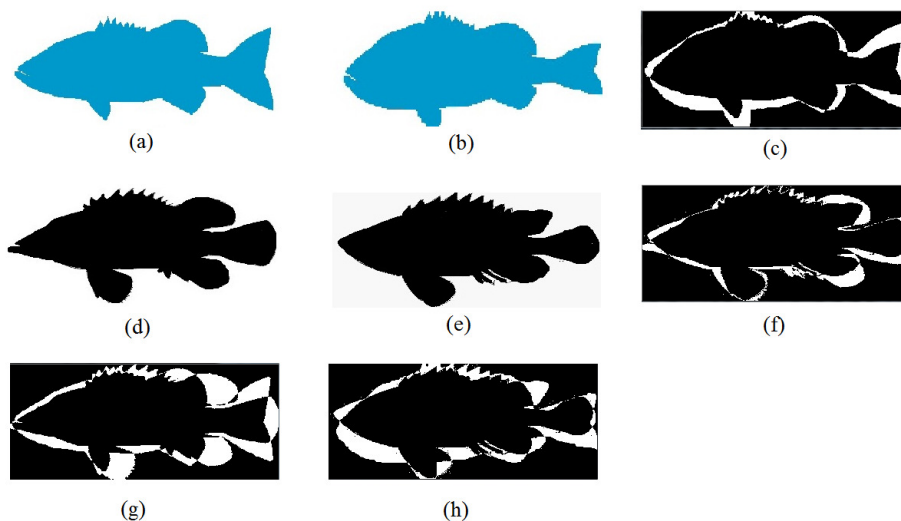


Figura 4.2: (a) Imagem do modelo (a) de 4.1; (b) Imagem do modelo (b) de 4.1; (c) Percentual de diferença de (a) e (b): 0.2219; (d) Imagem do peixe *Polyprion*; (e) Imagem do peixe *Pseudopriacanthus altus*; (f) Percentual de diferença de (d) e (e): 0.1708; (g) Percentual de diferença de (a) e (d): 0.2992; (h) Percentual de diferença de (b) e (e): 0.2328.

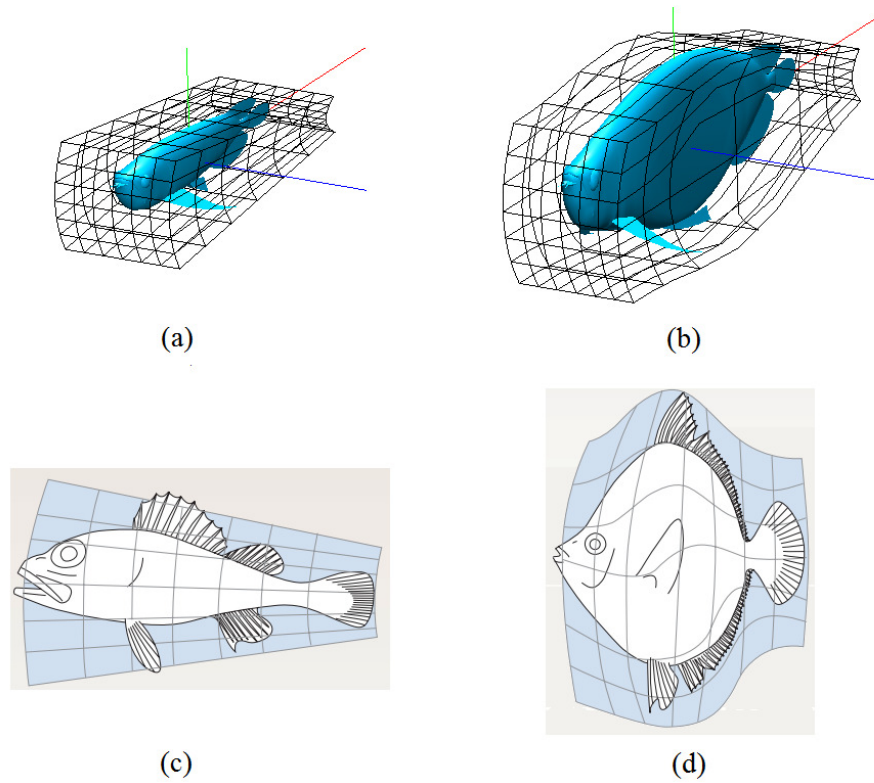


Figura 4.3: (a) Modelo inicial; (b) Transformação de (a) nas dimensões x e y com coeficientes $C = 1.0$ e $B = -0.5$, respectivamente, e $cr1 = 0.5$; (c) Imagem do peixe *Scorpaena* de Thompson; (d) Imagem do peixe *Antigonía capros* de Thompson.

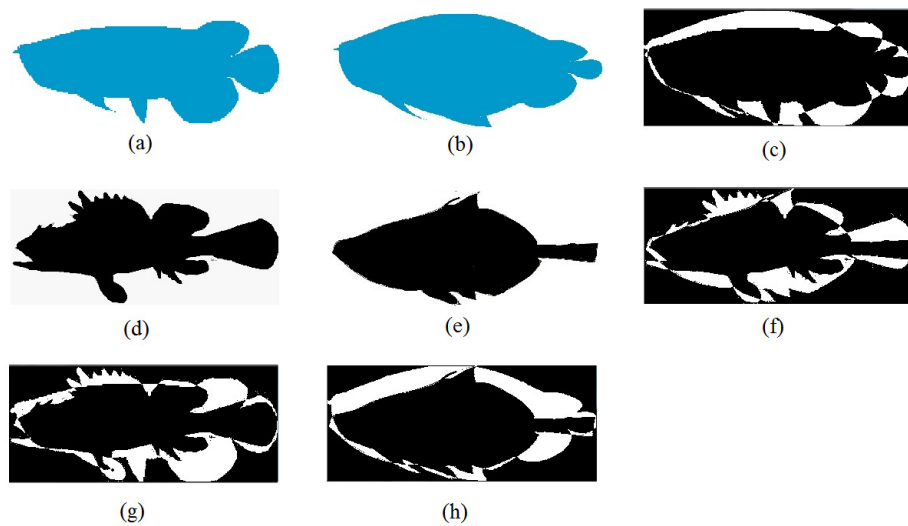


Figura 4.4: (a) Imagem do modelo (a) de 4.3; (b) Imagem do modelo (b) de 4.3; (c) Percentual de diferença de (a) e (b): 0.2679; (d) Imagem do peixe *Scorpaena*; (e) Imagem do peixe *Antigonía capros*; (f) Percentual de diferença de (d) e (e): 0.3373; (g) Percentual de diferença de (a) e (d): 0.3460; (h) Percentual de diferença de (b) e (e): 0.3186.

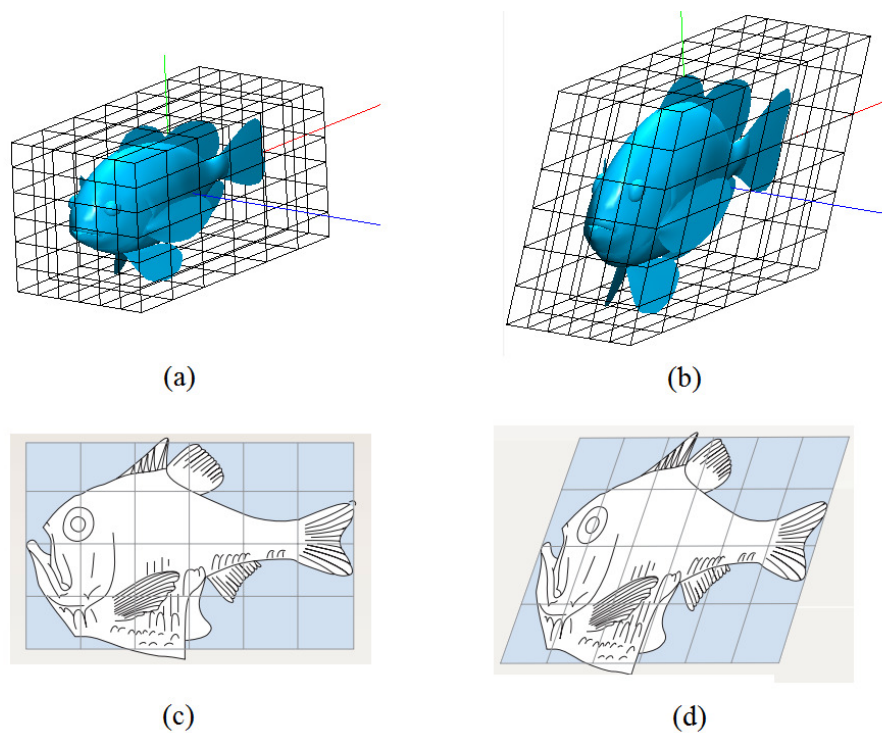


Figura 4.5: (a) Modelo inicial; (b) Transformação de (a) nas dimensões x e y com coeficientes $E = 0.5$, $E = -1.4$, respectivamente; (c) Imagem do peixe *Argyropelecus olfersi* de Thompson; (d) Imagem do peixe *Sternoptyx diaphana* de Thompson.

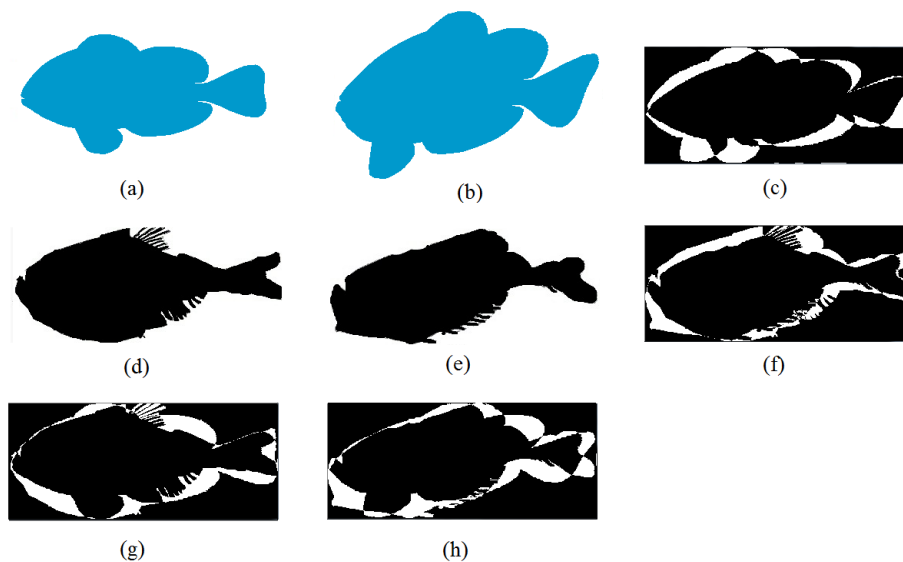


Figura 4.6: (a) Imagem do modelo (a) de 4.5; (b) Imagem do modelo (b) de 4.5; (c) Percentual de diferença de (a) e (b): 0.2556; (d) Imagem do peixe *Argyropelecus olfersi*; (e) Imagem do peixe *Sternoptyx diaphana*; (f) Percentual de diferença de (d) e (e): 0.2235; (g) Percentual de diferença de (a) e (d): 0.2210; (h) Percentual de diferença de (b) e (e): 0.2588.

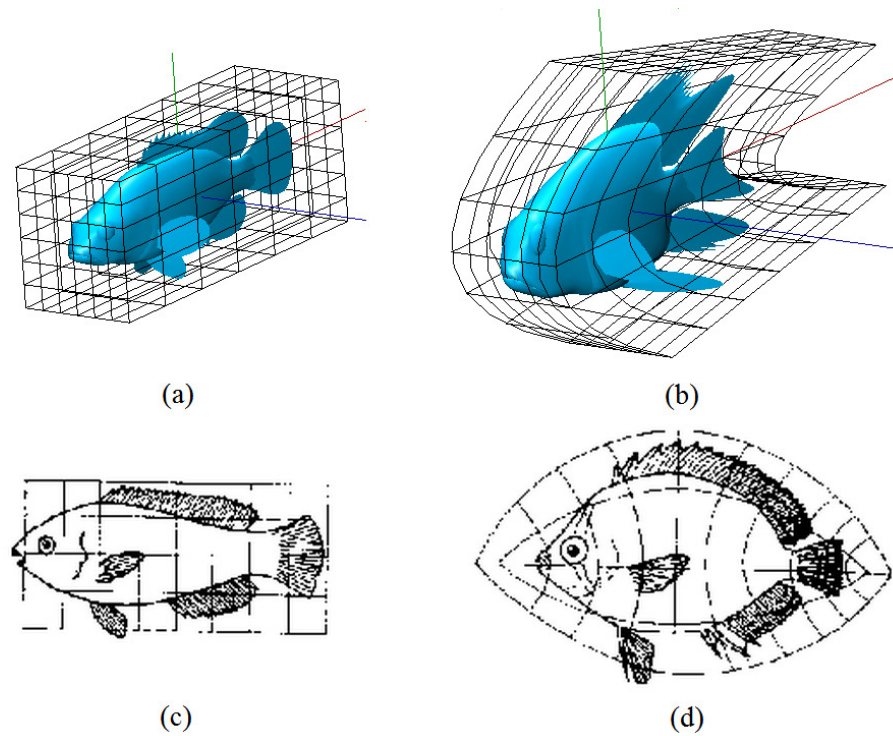


Figura 4.7: (a) Modelo inicial; (b) Transformação de (a) nas dimensões x , y e y com coeficientes $C = 4.0$, $B = -0.5$ e $E = 2.0$, respectivamente; (c) Imagem do peixe *Scarus sp.* de Thompson; (d) Imagem do peixe *Pomacanthus* de Thompson.

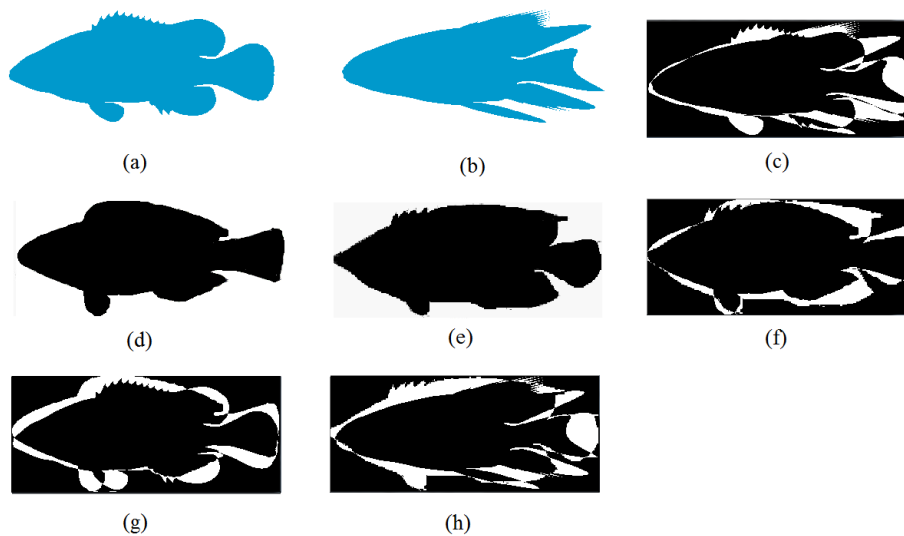


Figura 4.8: (a) Imagem do modelo (a) de 4.7; (b) Imagem do modelo (b) de 4.7; (c) Percentual de diferença de (a) e (b): 0.2796; (d) Imagem do peixe *Scarus sp.*; (e) Imagem do peixe *Pomacanthus*; (f) Percentual de diferença de (d) e (e): 0.1934; (g) Percentual de diferença de (a) e (d): 0.2466; (h) Percentual de diferença de (b) e (e): 0.2757.

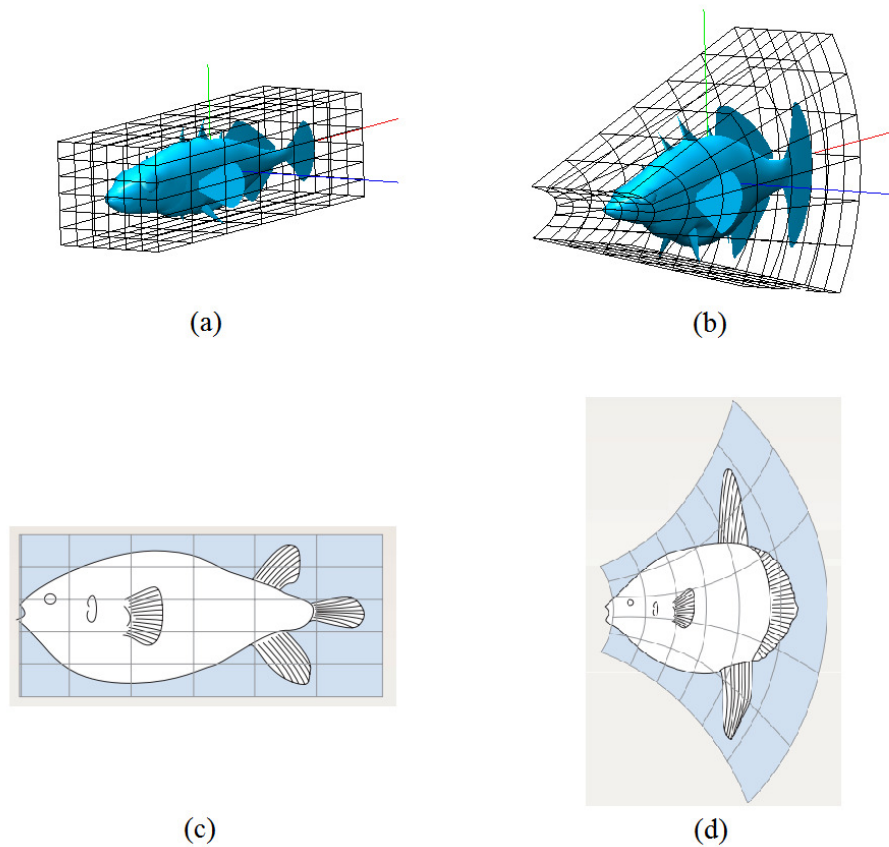


Figura 4.9: (a) Modelo inicial; (b) Transformação de (a) nas dimensões x , y e y com coeficientes $C = -2.0$, $B = 1.0$ e $E = 1.5$, respectivamente; (c) Imagem do peixe *Diodon* de Thompson; (d) Imagem do peixe *Orthogoriscus* de Thompson.

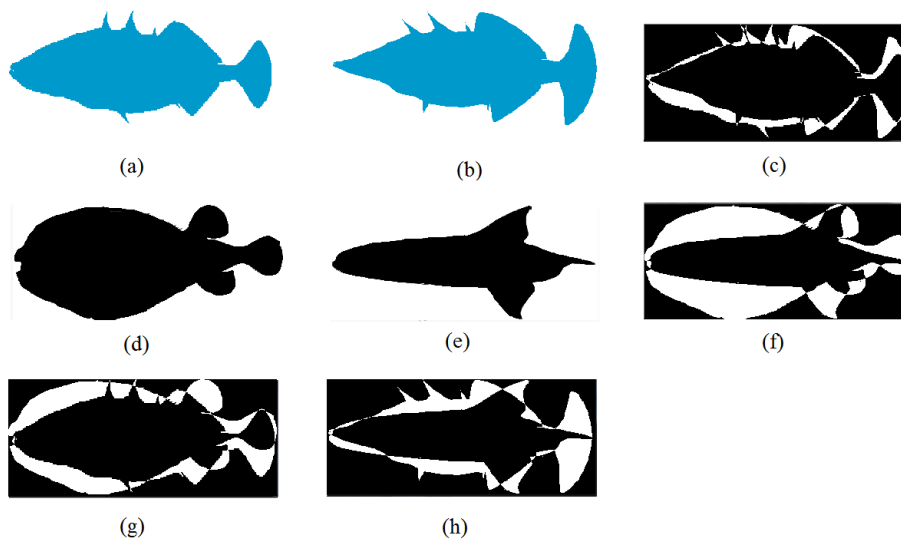


Figura 4.10: (a) Imagem do modelo (a) de 4.9; (b) Imagem do modelo (b) de 4.9; (c) Percentual de diferença de (a) e (b): 0.2388; (d) Imagem do peixe *Diodon*; (e) Imagem do peixe *Orthogoriscus*; (f) Percentual de diferença de (d) e (e): 0.5114; (g) Percentual de diferença de (a) e (d): 0.3470; (h) Percentual de diferença de (b) e (e): 0.3821.

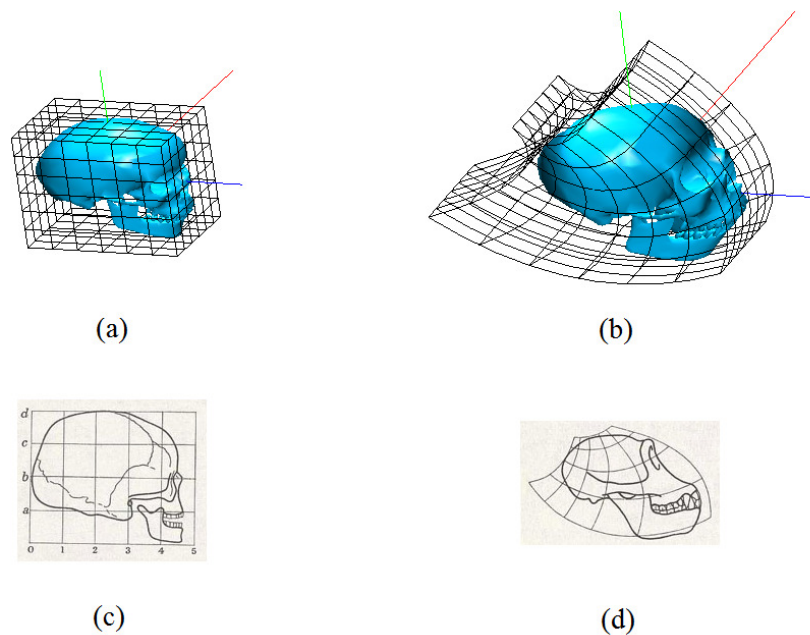


Figura 4.11: (a) Modelo inicial; (b) Transformação de (a) nas dimensões y , y , z e z com coeficientes $F = 0.5$, $I = 0.3$, $C = -0.5$ e $I = -0.5$, respectivamente; (c) Imagem de *Human Skull* de Thompson; (d) Imagem de *Cimpanzee Skull* de Thompson.

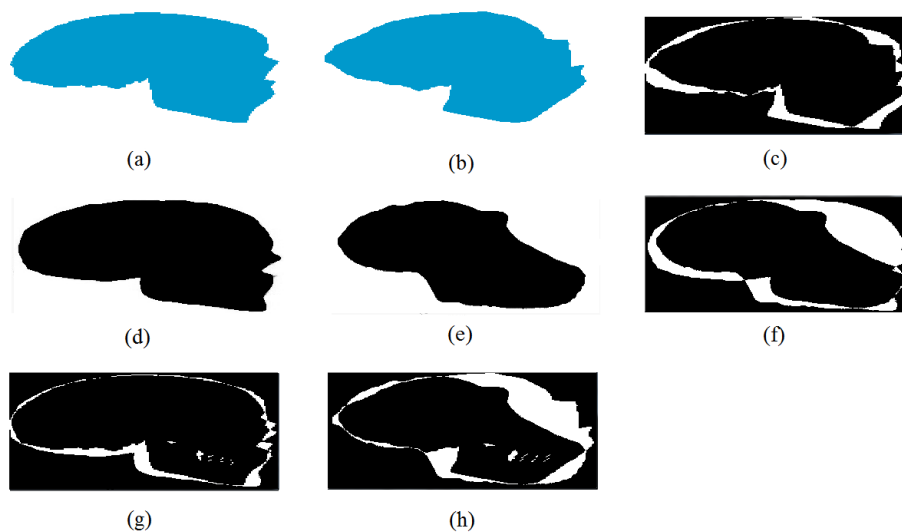


Figura 4.12: (a) Imagem do modelo (a) de 4.11; (b) Imagem do modelo (b) de 4.11; (c) Percentual de diferença de (a) e (b): 0.1574; (d) Imagem do crânio humano; (e) Imagem do crânio do chimpanzé; (f) Percentual de diferença de (d) e (e): 0.2742; (g) Percentual de diferença de (a) e (d): 0.1595; (h) Percentual de diferença de (b) e (e): 0.2998.

5 CONCLUSÃO

Este trabalho teve por objetivo explorar a teoria das transformações de D'Arcy Thompson e sua aplicabilidade em computação gráfica. Através dele foi possível um estudo não apenas específico na área das transformações entre as espécies de Thompson, como também na de modelagem geométrica de forma geral. A utilização de um polinômio de segundo grau em três variáveis, onde cada uma representa uma dimensão do espaço cartesiano, mostrou-se uma ferramenta flexível para a transformação entre objetos, gerando diversos resultados interessantes, usando um algoritmo global, onde todos os pontos do objeto variam simplesmente com a alteração dos coeficientes do polinômio. O uso da grade em torno do objeto, aliado com a visualização do resultado dinamicamente, permitiu deduzir mais facilmente os coeficientes que produziam os melhores resultados sobre o trabalho de D'Arcy Thompson.

Resultados como os ilustrados nas figuras 4.1 e 4.3, por serem de operações simplificadas, alteraram o objeto da mesma forma que o trabalho original. Além disso, resultados aparentemente difíceis de serem reproduzidos com uma equação global, obtiveram uma semelhança muito boa com as espécies retratadas. A flexibilidade proporcionada pela formulação polinomial foi considerada satisfatória, possibilitando o uso de um algoritmo simples para alcançar resultados satisfatórios quanto à deformação de objetos.

Apenas alguns casos, como na figura 4.9, não foram corretamente reproduzidos pois globalmente não foi possível alterar as curvas das linhas horizontais em diferentes concavidades. Entretanto, um caso parecido, ilustrado na figura 4.7, reproduziu um resultado semelhante ao original, porém com uma alteração bem diferente na grade. Para reproduzir da mesma forma que na imagem original, seria necessário um algoritmo local que variasse conforme a região em que os coeficientes fossem alterados, um ponto não abordado nesse trabalho.

A falta de dados das espécies de animais usados por Thompson, tanto no material original como em pesquisas online, mostrou-se uma dificuldade para a exploração das transformações incluindo a terceira dimensão z , que não foi abordada no trabalho original. Por causa disso, os coeficientes dos resultados das transformações restringem-se a dois eixos, variando dependendo da posição inicial do objeto. Além disso, a métrica usada para comparação entre as formas necessitaria de aperfeiçoamento, já que as máscaras das espécies usadas foram baseadas em fotos reais, quando na verdade em seu trabalho foram utilizados esboços das mesmas.

Para trabalhos futuros pode-se incluir transformações também no eixo z . Uma possibilidade seria o uso de relações de proporções entre as partes como, por exemplo, espessura versus comprimento. A criação de modelos tridimensionais baseados nas medidas padrões das espécies de animais estudadas por Thompson ajudaria a reduzir drasticamente o erro inicial encontrado. Em relação ao aspecto visual, o uso de texturas para as es-

pécies tratadas também poderá ser introduzido, expandindo, até mesmo para um estudo sobre um padrão de alteração das texturas entre as formas. O critério de validação poderá ser substituído por outro mais preciso e bastará ser comparado com as figuras usadas por Thompson. Novas transformações poderão ser incorporadas para produzir diferentes resultados, expandindo o uso dos polinômios para tentar reproduzir ainda melhor as transformações do trabalho original. Um exemplo dessa idéia pode ser visto na figura 5.1, onde uma maçã (fig. 5.1(a)) transforma-se em uma pêra (fig. 5.1(b)). O uso de uma interface que possibilite trabalhar com pontos do *grid* localmente baseado nos coeficientes também aumentaria bastante a flexibilidade do programa.

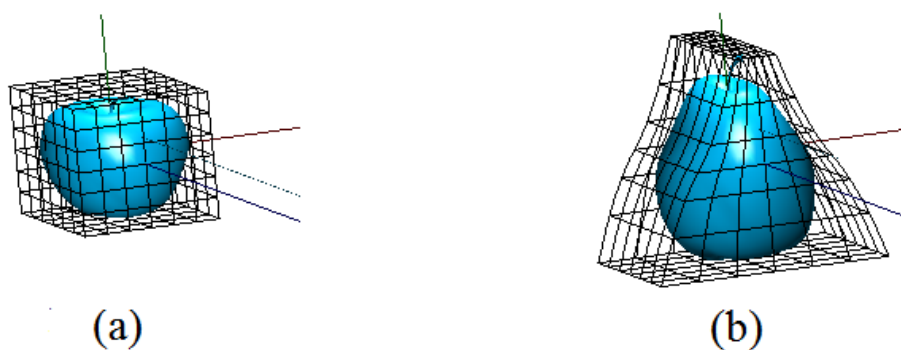


Figura 5.1: (a) Modelo inicial; (b) Transformação de (a) nas dimensões x e y com coeficientes $B = -1.0$ e $C = 1.0$, respectivamente.

REFERÊNCIAS

ALEXA, M.; COHEN-OR, D.; LEVIN, D. As-rigid-as-possible shape interpolation. In: COMPUTER GRAPHICS AND INTERACTIVE TECHNIQUES, 27. **Proceedings...** [S.l.: s.n.], 2000. p.157–164.

ARTHUR, W. D’Arcy Thompson and the theory of transformations. **Nature Reviews Genetics**, [S.l.], v.7, n.5, p.401–406, 2006.

BEIER, T.; NEELY, S. Feature-based image metamorphosis. In: ACM SIGGRAPH COMPUTER GRAPHICS. **Anais...** [S.l.: s.n.], 1992. v.26, n.2, p.35–42.

CHANG, Y.; ROCKWOOD, A. A generalized de Casteljau approach to 3D free-form deformation. In: COMPUTER GRAPHICS AND INTERACTIVE TECHNIQUES, 21. **Proceedings...** [S.l.: s.n.], 1994. p.257–260.

CHEN, M.; MOUNTFORD, S. J.; SELLEN, A. A Study in Interactive 3-D Rotation Using 2-D Control Devices. In: COMPUTER GRAPHICS (PROCEEDINGS OF SIGGRAPH 88). **Anais...** [S.l.: s.n.], 1988. p.121–129.

D’ARCY, W. **On growth and form**. [S.l.]: Cambridge Univ Press, 1963.

HSU, W.; HUGHES, J.; KAUFMAN, H. Direct manipulation of free-form deformations. In: ACM SIGGRAPH COMPUTER GRAPHICS. **Anais...** [S.l.: s.n.], 1992. v.26, n.2, p.177–184.

HUTTENLOCHER, D.; KLANDERMAN, G.; RUCKLIDGE, W. Comparing images using the Hausdorff distance. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, [S.l.], v.15, n.9, p.850–863, 1993.

KENT, J.; CARLSON, W.; PARENT, R. Shape transformation for polyhedral objects. In: ACM SIGGRAPH COMPUTER GRAPHICS. **Anais...** [S.l.: s.n.], 1992. v.26, n.2, p.47–54.

LANDRÉ, J.; TRUCHETET, F. Image retrieval with binary hamming distance. **Proc. 2nd VISAPP**, [S.l.], 2007.

MORTENSEN, M. **Geometric modeling**. [S.l.]: John Wiley & Sons, Inc., 1985.

SEDERBERG, T.; PARRY, S. Free-form deformation of solid geometric models. In: ACM SIGGRAPH COMPUTER GRAPHICS. **Anais...** [S.l.: s.n.], 1986. v.20, n.4, p.151–160.