

**UNIVERSIDADE FEDERAL DO PARANÁ**

**DANIEL VITOR RUIZ**

**PROJEÇÃO DE IMAGENS DIGITAIS SOBRE RESULTADOS DE  
ESCANEAMENTOS 3D**

**CURITIBA PR  
2017**

DANIEL VITOR RUIZ

**PROJEÇÃO DE IMAGENS DIGITAIS SOBRE RESULTADOS DE  
ESCANEAMENTOS 3D**

Trabalho apresentado como requisito parcial à conclusão do Curso de Bacharelado em Ciência da Computação, setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Ciência da Computação*.

Orientador: Luciano Silva.

CURITIBA PR  
2017

**Universidade Federal do Paraná**  
**Setor de Ciências Exatas**  
**Curso de Ciência da Computação**

**Ata de Apresentação de Trabalho de Graduação II**

**Título do Trabalho:** PROJEÇÃO DE IMAGENS DIGITAIS SOBRE RESULTADOS DE ESCANEAMENTOS 3D

**Autor(es):**

GRR 20131061 Nome: DANIEL VITOR RUIZ

GRR \_\_\_\_\_ Nome: \_\_\_\_\_

GRR \_\_\_\_\_ Nome: \_\_\_\_\_

Apresentação: Data: 15/12/2017 Hora: 14:00 Local: SALA VÍDEO CONF.

Orientador: LUCIANO SILVA

Membro 1: Edardo Tadz

Membro 2: ANDRÉ L. VIGNATTI

(nome)

Luciano Si

Edardo Tadz

(assinatura)

AVALIAÇÃO – Produto escrito	ORIENTADOR	MEMBRO 1	MEMBRO 2	MÉDIA
Conteúdo (00-40)				
Referência Bibliográfica (00-10)				
Formato (00-05)				
AVALIAÇÃO – Apresentação Oral				
Domínio do Assunto (00-15)				
Desenvolvimento do Assunto (00-05)				
Técnica de Apresentação (00-03)				
Uso do Tempo (00-02)				
AVALIAÇÃO – Desenvolvimento				
Nota do Orientador (00-20)		*****	*****	
<b>NOTA FINAL</b>	*****	*****	*****	<b>100</b>

Pesos indicados são sugestões.

Conforme decisão do colegiado do curso de Ciência da Computação, a entrega dos documentos comprobatório de trabalho de graduação 2 deve respeitar os seguintes procedimentos: Orientador deve abrir um processo no Sistema Eletrônico de Informações (SEI – UFPR); Selecionar o tipo: Graduação: Trabalho Conclusão de Curso; informar os interessados: nome do aluno e o nome do orientador; anexar esta ata escaneada e a versão final do pdf da monografia do aluno.; Tramita o processo para CCOMP (Coordenação Ciência da Computação).

*Dedico este trabalho a todos que contribuíram direta ou indiretamente em minha formação acadêmica.*

# **Agradecimentos**

Gostaria de agradecer o apoio da minha família, os ensinamentos dos meus professores e a oportunidade de aprendizado que a UFPR me propiciou durante a graduação.

# Resumo

O trabalho desenvolvido busca investigar possibilidades, algoritmos e dificuldades para gerar uma textura para um objeto 3D a partir de uma fotografia ou sequência de quadros de vídeo, afim de colorir uma malha poligonal resultante de um escaneamento 3D. Através de um levantamento bibliográfico e de um embasamento teórico das áreas de Visão computacional, Computação Gráfica e Processamento de Imagens, está sendo apresentada uma descrição da implementação de um protótipo funcional capaz de realizar experimentos. A abordagem usada consiste de uma tentativa de recriar a cena original em um ambiente 3D simulado para tal é necessário obter uma pose do objeto próxima à pose real, o equiparo das poses é medido com a avaliação de uma função que mede a distância entre a imagem 3D projetada e a imagem 2D de interesse, a pose então é otimizada afim de minimizar a distância entre as imagens e finalmente é feito o mapeamento para uma nova textura.

São descritos alguns dos experimentos realizados e os resultados obtidos, que apontam o comportamento do protótipo em diversos casos, entre eles: a influência de fontes de luz em diferentes posições; a influência da resolução das imagens; a influência de se usar retroalimentação no otimizador; o comportamento utilizando diferentes fotografias reais; qual o equiparo perfeito; qual a influência do ruído; qual o resultado obtido utilizando imagens sintéticas e como o protótipo se comporta quando é utilizada uma sequência de quadros de vídeo.

**Palavras-chave:** *normal-mapping*, re-texturização e digitalização-3D.

# Abstract

The work developed seeks to investigate possibilities, algorithms and difficulties to generate a texture for a 3D object, from a photo or a sequence of video frames, in order to colorize a polygon mesh resulting from a 3D scan. Through bibliographic survey, and theoretical background from the fields of Computer Vision, Computer Graphics and Image Processing, it's presented a description of the implementation of a functional prototype capable of running experiments. The approach used consists of an attempt to recreate the original scene in a simulated 3D environment for such it is necessary a pose of the object close enough to the actual pose, the match between poses is measured by the evaluation of a function which measures the distance between the projected 3D image and the 2D image of interest, the pose is then optimized in order to minimize the distance between the images and finally occurs the mapping to a new texture.

Some of the experiments performed and the results are described, which point out the behavior of the prototype in several cases, among them: the influence of light sources in different positions; the influence of image resolution; the influence of using feedback in the optimizer; the behavior using different real photographs; what is the perfect match; what is the influence of noise; the result obtained using synthetic images and how the prototype behaves when a sequence of video frames is used.

**Keywords:** normal-mapping, re-texturing and 3D-digitalization.

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Motivação . . . . .	1
1.2	Desafio . . . . .	2
1.3	Objetivo . . . . .	6
1.4	Contribuição . . . . .	6
1.5	Organização . . . . .	6
<b>2</b>	<b>Embasamento teórico</b>	<b>8</b>
2.1	Modelo tridimensional . . . . .	8
2.2	Mapeamento de cor . . . . .	9
2.3	Sombreamento Lambertiano e Blinn-Phong . . . . .	15
2.4	Estimativa de pose . . . . .	16
2.5	Distância de Mahalanobis . . . . .	19
2.6	Função de perda invariante à luz . . . . .	19
2.6.1	Formulação teórica . . . . .	19
2.6.2	Modelo de reflexão Lambertiano . . . . .	20
2.6.3	Medida de distância invariante à luz . . . . .	20
2.7	<i>Downhill-Simplex</i> . . . . .	21
2.8	Remoção de plano de fundo . . . . .	23
2.9	Conclusão do capítulo . . . . .	24
<b>3</b>	<b>Implementação</b>	<b>25</b>
3.1	Pré-processamento . . . . .	25
3.2	Renderização . . . . .	25
3.3	Matrizes de Transformação <i>Affine</i> . . . . .	26
3.4	Representação da Pose . . . . .	27
3.5	Estimativa de Pose Inicial . . . . .	27
3.6	Passos da execução . . . . .	28
3.7	Cálculo da distância . . . . .	28
3.8	Obtendo a nova textura . . . . .	30
3.9	Ilustração do processo . . . . .	32
3.10	Dificuldades e problemas . . . . .	37
3.11	Conclusão do capítulo . . . . .	38
<b>4</b>	<b>Experimentos</b>	<b>39</b>
4.1	Experimento 1 . . . . .	41
4.2	Experimento 2 . . . . .	44
4.3	Experimento 3 . . . . .	46
4.4	Experimento 4 . . . . .	51

4.5	Experimento 5 . . . . .	52
4.6	Experimento 6 . . . . .	55
4.7	Experimento 7 . . . . .	59
4.8	Experimento 8 . . . . .	61
4.9	Conclusões sobre os experimentos . . . . .	111
<b>5</b>	<b>Conclusão</b>	<b>112</b>
5.1	Vantagens da abordagem escolhida . . . . .	112
5.2	Desvantagens da abordagem escolhida . . . . .	112
5.3	Aplicação prática . . . . .	112
5.4	Trabalhos Futuros . . . . .	113
<b>Referências Bibliográficas</b>		<b>114</b>

# Listas de Figuras

1.1	Cena onde o posicionamento da câmera produz oclusão dos objetos . . . . .	2
1.2	Influência da luz na cena . . . . .	2
1.3	Objeto "cilíndrico" em diferentes poses . . . . .	3
1.4	Influência do ambiente em objeto transparente . . . . .	3
1.5	Comparação entre distâncias focais, fonte [Baker, 2012] . . . . .	4
1.6	Objeto com peças articuláveis . . . . .	4
1.7	Borramento de movimento . . . . .	5
1.8	Presença de ruído HSV em imagens . . . . .	6
2.1	Exemplo de malha, fonte:[Shirley e Marschner, 2009] . . . . .	8
2.2	<i>Texture Tiling</i> , fonte:[Shirley e Marschner, 2009] . . . . .	9
2.3	(a) representa uma imagem com pixels pretos e brancos, três estratégias de interpolação são apresentadas (b) nearest-neighbor, (c) bilinear e (d) hermite, fonte:[Shirley e Marschner, 2009] . . . . .	10
2.4	As sub-figuras (a), (b), (c) mostram diferentes malhas e suas respectivas coordenadas UV, já as sub-figuras (e), (f) mostram o resultado do mapeamento usando a textura de calibração vista em (d), fonte:[Shirley e Marschner, 2009] adaptado	10
2.5	Diferentes <i>meshes</i> e seus respectivos <i>texture atlas</i> resultantes do LSCM, fonte:[Lévy et al., 2002] . . . . .	12
2.6	Tabela de estatísticas e tempo relativas a figura 2.5, fonte:[Lévy et al., 2002] . .	12
2.7	Diferentes <i>meshes</i> e seus <i>texture atlas</i> resultantes do ABF++, as parametrizações 2D foram coloridas usando 3D <i>normal map</i> , fonte [Sheffer et al., 2005] . . . . .	13
2.8	Texturas e seus respectivos mapeamentos, gerados usando implementação provida por [ALICE, 2017] . . . . .	14
2.9	Geometria para o sombreamento lambertiano, fonte:[Shirley e Marschner, 2009]	15
2.10	Geometria para o sombreamento Blinn-Phong, fonte:[Shirley e Marschner, 2009]	16
2.11	Componentes do sombreamento Blinn-Phong . . . . .	16
2.12	influência do coeficiente <i>p</i> , <i>Phong exponent</i> , fonte:[Shirley e Marschner, 2009]	16
2.13	Organização da pesquisa,fonte:[Chin e Dyer, 1986] . . . . .	17
2.14	Ilustração dos quatro métodos dentro do <i>Downhill simplex</i> para definir novos pontos do simplex. (a) Reflexão,(b) Expansão,(c) Contração,(d) Compressão. Fonte [Fox, 2013] . . . . .	22
2.15	Fluxograma do Algoritmo <i>Downhill Simplex</i> , fonte [Fox, 2013] adaptado . . . . .	23
3.1	Sequência de espaços e transformações, fonte:[Shirley e Marschner, 2009] . . .	27
3.2	Imagens de entrada para a distância <i>loss</i> . . . . .	28
3.3	Imagens de entrada para a distância <i>loss</i> . . . . .	29
3.4	Textura original, obtida na conversão da cor de vértice para o mapeamento LSCM com 45 graus . . . . .	32

3.5	Exemplo de renderização resultante utilizando a textura exibida na figura 3.4 . . . . .	33
3.6	Textura gerada pela pose $\theta$ , descrita na coluna "inicial" da tabela 4.6 . . . . .	34
3.7	Exemplo de renderização resultante utilizando a textura exibida na figura 3.6 . . . . .	35
3.8	Textura gerada pela pose $\theta'$ , descrita na coluna "final" da tabela 4.6 . . . . .	36
3.9	Exemplo de renderização resultante utilizando a textura exibida na figura 3.8 . . . . .	37
4.1	Imagen 1 do experimento 1, convergência de (a) para (b) afim de demonstrar influência da luz ambiente . . . . .	41
4.2	Imagen 2 do experimento 1, convergência de (a) para (b) afim de demonstrar influência da luz ambiente . . . . .	42
4.3	Imagen 3 do experimento 1, convergência de (a) para (b) afim de demonstrar influência da luz ambiente . . . . .	43
4.4	Imagen 4 do experimento 1, convergência de (a) para (b) afim de demonstrar influência da luz ambiente . . . . .	44
4.5	Imagen 1 do experimento 2, convergência de (a) para (b) afim de demonstrar a influência da resolução da imagem, resolução testada 4896x3672 . . . . .	45
4.6	Imagen 2 do experimento 2, convergência de (a) para (b) afim de demonstrar a influência da resolução da imagem, resolução testada 612x459 . . . . .	46
4.7	Imagen 1 do experimento 3, convergência de (a) para (b) usando estimativa inicial . . . . .	47
4.8	Imagen 2 do experimento 3, convergência de (a) para (b), usando parâmetros finais da tabela 4.9 como estimativa inicial . . . . .	48
4.9	Imagen 3 do experimento 3, convergência de (a) para (b), usando parâmetros finais da tabela 4.10 como estimativa inicial . . . . .	49
4.10	Imagen 4 do experimento 3, convergência de (a) para (b), usando parâmetros finais da tabela 4.11 como estimativa inicial . . . . .	50
4.11	Imagen 1 do experimento 4, convergência de (a) para (b). A imagem tem sua escala de cinza invertida, mantendo o plano de fundo com valor 0, quando usada no calculo da distancia . . . . .	51
4.12	Imagen 2 do experimento 4, convergência de (a) para (b). . . . .	52
4.13	Imagen 1 do experimento 5, convergência de (a) para (b), afim de demonstrar a convergência para imagens sintéticas utilizando escalas de cinza. Dessaturação com média dos canais . . . . .	53
4.14	Imagen 2 do experimento 5, convergência de (a) para (b), afim de demonstrar a convergência para imagens sintéticas utilizando escalas de cinza. Dessaturação com média dos canais . . . . .	54
4.15	Imagen 3 do experimento 5, convergência de (a) para (b), afim de demonstrar a convergência para imagens sintéticas utilizando escalas de cinza. Dessaturação com média dos canais. A imagem tem sua escala de cinza invertida, mantendo o plano de fundo com valor 0, quando usada no calculo da distancia se aproximando da figura 4.13 . . . . .	55
4.16	Imagen 1 do experimento 6, convergência de (a) para (b), afim de demonstrar a influência de ruído HSV para imagens sintéticas. Teste sem ruído . . . . .	56
4.17	Imagen 2 do experimento 6, convergência de (a) para (b), afim de demonstrar a influência de ruído HSV para imagens sintéticas. Teste com ruído . . . . .	57
4.18	Imagen 3 do experimento 6, convergência de (a) para (b), afim de demonstrar a influência de ruído HSV para imagens sintéticas. Teste sem ruído. A imagem tem sua escala de cinza invertida, mantendo o plano de fundo com valor 0, quando usada no calculo da distancia. . . . .	58

4.19	Imagen 4 do experimento 6, convergência de (a) para (b), afim de demonstrar a influência de ruído HSV para imagens sintéticas. Teste com ruído. A imagem tem sua escala de cinza invertida, mantendo o plano de fundo com valor 0, quando usada no calculo da distancia. . . . .	59
4.20	Imagen 1 do experimento 7, convergência de (a) para (b), afim de demonstrar a convergência usando imagem sintética sem sombreamento apenas textura . . .	60
4.21	Imagen 1 do experimento 8, convergência de (a) para (b), do frame de vídeo 0	62
4.22	Imagen 2 do experimento 8, convergência de (a) para (b), do frame de vídeo 1	63
4.23	Imagen 3 do experimento 8, convergência de (a) para (b), do frame de vídeo 2	64
4.24	Imagen 4 do experimento 8, convergência de (a) para (b), do frame de vídeo 3	65
4.25	Imagen 5 do experimento 8, convergência de (a) para (b), do frame de vídeo 4	66
4.26	Imagen 6 do experimento 8, convergência de (a) para (b), do frame de vídeo 5	67
4.27	Imagen 7 do experimento 8, convergência de (a) para (b), do frame de vídeo 6	68
4.28	Imagen 8 do experimento 8, convergência de (a) para (b), do frame de vídeo 7	69
4.29	Imagen 9 do experimento 8, convergência de (a) para (b), do frame de vídeo 8	70
4.30	Imagen 10 do experimento 8, convergência de (a) para (b), do frame de vídeo 9	71
4.31	Imagen 11 do experimento 8, convergência de (a) para (b), do frame de vídeo 10	72
4.32	Imagen 12 do experimento 8, convergência de (a) para (b), do frame de vídeo 11	73
4.33	Imagen 13 do experimento 8, convergência de (a) para (b), do frame de vídeo 12	74
4.34	Imagen 14 do experimento 8, convergência de (a) para (b), do frame de vídeo 13	75
4.35	Imagen 15 do experimento 8, convergência de (a) para (b), do frame de vídeo 14	76
4.36	Imagen 16 do experimento 8, convergência de (a) para (b), do frame de vídeo 15	77
4.37	Imagen 17 do experimento 8, convergência de (a) para (b), do frame de vídeo 16	78
4.38	Imagen 18 do experimento 8, convergência de (a) para (b), do frame de vídeo 17	79
4.39	Imagen 19 do experimento 8, convergência de (a) para (b), do frame de vídeo 18	80
4.40	Imagen 20 do experimento 8, convergência de (a) para (b), do frame de vídeo 19	81
4.41	Imagen 21 do experimento 8, convergência de (a) para (b), do frame de vídeo 20	82
4.42	Imagen 22 do experimento 8, convergência de (a) para (b), do frame de vídeo 21	83
4.43	Imagen 23 do experimento 8, convergência de (a) para (b), do frame de vídeo 22	84
4.44	Imagen 24 do experimento 8, convergência de (a) para (b), do frame de vídeo 23	85
4.45	Imagen 25 do experimento 8, convergência de (a) para (b), do frame de vídeo 24	86
4.46	Imagen 26 do experimento 8, convergência de (a) para (b), do frame de vídeo 25	87
4.47	Imagen 27 do experimento 8, convergência de (a) para (b), do frame de vídeo 26	88
4.48	Imagen 28 do experimento 8, convergência de (a) para (b), do frame de vídeo 27	89
4.49	Imagen 29 do experimento 8, convergência de (a) para (b), do frame de vídeo 28	90
4.50	Imagen 30 do experimento 8, convergência de (a) para (b), do frame de vídeo 29	91
4.51	Imagen 31 do experimento 8, convergência de (a) para (b), do frame de vídeo 30	92
4.52	Imagen 32 do experimento 8, convergência de (a) para (b), do frame de vídeo 31	93
4.53	Imagen 33 do experimento 8, convergência de (a) para (b), do frame de vídeo 32	94
4.54	Imagen 34 do experimento 8, convergência de (a) para (b), do frame de vídeo 33	95
4.55	Imagen 35 do experimento 8, convergência de (a) para (b), do frame de vídeo 34	96
4.56	Imagen 36 do experimento 8, convergência de (a) para (b), do frame de vídeo 35	97
4.57	Imagen 37 do experimento 8, convergência de (a) para (b), do frame de vídeo 36	98
4.58	Imagen 38 do experimento 8, convergência de (a) para (b), do frame de vídeo 37	99
4.59	Imagen 39 do experimento 8, convergência de (a) para (b), do frame de vídeo 38	100
4.60	Imagen 40 do experimento 8, convergência de (a) para (b), do frame de vídeo 39	101
4.61	Imagen 41 do experimento 8, convergência de (a) para (b), do frame de vídeo 40	102
4.62	Imagen 42 do experimento 8, convergência de (a) para (b), do frame de vídeo 41	103

- 4.63 Imagem 43 do experimento 8, convergência de (a) para (b), do frame de vídeo 42 104
- 4.64 Imagem 44 do experimento 8, convergência de (a) para (b), do frame de vídeo 43 105
- 4.65 Imagem 45 do experimento 8, convergência de (a) para (b), do frame de vídeo 44 106
- 4.66 Imagem 46 do experimento 8, convergência de (a) para (b), do frame de vídeo 45 107
- 4.67 Imagem 47 do experimento 8, convergência de (a) para (b), do frame de vídeo 46 108
- 4.68 Imagem 48 do experimento 8, convergência de (a) para (b), do frame de vídeo 47 109
- 4.69 Imagem 49 do experimento 8, convergência de (a) para (b), do frame de vídeo 48 110

# **Lista de Tabelas**

4.1	Parâmetros iniciais e finais da figura 4.1 . . . . .	41
4.2	Parâmetros iniciais e finais da figura 4.2 . . . . .	42
4.3	Parâmetros iniciais e finais da figura 4.3 . . . . .	43
4.4	Parâmetros iniciais e finais da figura 4.4 . . . . .	44
4.5	Tempo de execução do protótipo para realizar a execução completa da instância 1 do experimento 2 . . . . .	45
4.6	Parâmetros iniciais e finais da figura 4.5 . . . . .	45
4.7	Tempo de execução do protótipo para realizar a execução completa da instância 2 do experimento 2 . . . . .	46
4.8	Parâmetros iniciais e finais da figura 4.6 . . . . .	46
4.9	Parâmetros iniciais e finais da figura 4.7 . . . . .	47
4.10	Parâmetros iniciais e finais da figura 4.8 . . . . .	48
4.11	Parâmetros iniciais e finais da figura 4.9 . . . . .	49
4.12	Parâmetros iniciais e finais da figura 4.10 . . . . .	50
4.13	Parâmetros iniciais e finais da figura 4.11 . . . . .	51
4.14	Parâmetros iniciais e finais da figura 4.12 . . . . .	52
4.15	Parâmetros iniciais e finais da figura 4.13 . . . . .	53
4.16	Parâmetros iniciais e finais da figura 4.15 . . . . .	54
4.17	Parâmetros iniciais e finais da figura 4.15 . . . . .	55
4.18	Parâmetros iniciais e finais da figura 4.16 . . . . .	56
4.19	Parâmetros iniciais e finais da figura 4.17 . . . . .	57
4.20	Parâmetros iniciais e finais da figura 4.18 . . . . .	58
4.21	Parâmetros iniciais e finais da figura 4.19 . . . . .	59
4.22	Parâmetros iniciais e finais da figura 4.20 . . . . .	60
4.23	Parâmetros iniciais e finais da figura 4.21 . . . . .	62
4.24	Parâmetros iniciais e finais da figura 4.22 . . . . .	63
4.25	Parâmetros iniciais e finais da figura 4.23 . . . . .	64
4.26	Parâmetros iniciais e finais da figura 4.24 . . . . .	65
4.27	Parâmetros iniciais e finais da figura 4.25 . . . . .	66
4.28	Parâmetros iniciais e finais da figura 4.26 . . . . .	67
4.29	Parâmetros iniciais e finais da figura 4.27 . . . . .	68
4.30	Parâmetros iniciais e finais da figura 4.28 . . . . .	69
4.31	Parâmetros iniciais e finais da figura 4.29 . . . . .	70
4.32	Parâmetros iniciais e finais da figura 4.30 . . . . .	71
4.33	Parâmetros iniciais e finais da figura 4.31 . . . . .	72
4.34	Parâmetros iniciais e finais da figura 4.32 . . . . .	73
4.35	Parâmetros iniciais e finais da figura 4.33 . . . . .	74
4.36	Parâmetros iniciais e finais da figura 4.34 . . . . .	75

4.37	Parâmetros iniciais e finais da figura 4.35	76
4.38	Parâmetros iniciais e finais da figura 4.36	77
4.39	Parâmetros iniciais e finais da figura 4.37	78
4.40	Parâmetros iniciais e finais da figura 4.38	79
4.41	Parâmetros iniciais e finais da figura 4.39	80
4.42	Parâmetros iniciais e finais da figura 4.40	81
4.43	Parâmetros iniciais e finais da figura 4.41	82
4.44	Parâmetros iniciais e finais da figura 4.42	83
4.45	Parâmetros iniciais e finais da figura 4.43	84
4.46	Parâmetros iniciais e finais da figura 4.44	85
4.47	Parâmetros iniciais e finais da figura 4.45	86
4.48	Parâmetros iniciais e finais da figura 4.46	87
4.49	Parâmetros iniciais e finais da figura 4.47	88
4.50	Parâmetros iniciais e finais da figura 4.48	89
4.51	Parâmetros iniciais e finais da figura 4.49	90
4.52	Parâmetros iniciais e finais da figura 4.50	91
4.53	Parâmetros iniciais e finais da figura 4.51	92
4.54	Parâmetros iniciais e finais da figura 4.52	93
4.55	Parâmetros iniciais e finais da figura 4.53	94
4.56	Parâmetros iniciais e finais da figura 4.54	95
4.57	Parâmetros iniciais e finais da figura 4.55	96
4.58	Parâmetros iniciais e finais da figura 4.56	97
4.59	Parâmetros iniciais e finais da figura 4.57	98
4.60	Parâmetros iniciais e finais da figura 4.58	99
4.61	Parâmetros iniciais e finais da figura 4.59	100
4.62	Parâmetros iniciais e finais da figura 4.60	101
4.63	Parâmetros iniciais e finais da figura 4.61	102
4.64	Parâmetros iniciais e finais da figura 4.62	103
4.65	Parâmetros iniciais e finais da figura 4.63	104
4.66	Parâmetros iniciais e finais da figura 4.64	105
4.67	Parâmetros iniciais e finais da figura 4.65	106
4.68	Parâmetros iniciais e finais da figura 4.66	107
4.69	Parâmetros iniciais e finais da figura 4.67	108
4.70	Parâmetros iniciais e finais da figura 4.68	109
4.71	Parâmetros iniciais e finais da figura 4.69	110

# List of Acronyms

DINF	Departamento de Informática
PPGINF	Programa de Pós-Graduação em Informática
UFPR	Universidade Federal do Paraná
LSCM	<i>Least Squares Conformal Maps</i>
ABF	<i>Angle based Flattening</i>
RGB	<i>Red,Green,Blue</i> , representação de cor em três canais vermelho, verde e azul
HSV	<i>Hue,Saturation,Value</i> , representação de cor em três canais matiz, saturação e valor
LOD	<i>Level of Detail</i> , Nível de detalhe refere à pirâmide LOD
PnP	<i>the Perspective-n-Point problem</i>
ICP	<i>Iterative closest point</i>
GPU	<i>graphics processing unit</i>
FPS	<i>frame per second</i>
API	<i>Application Programming Interface</i>
GLSL	<i>OpenGL Shading Language</i>
DM	Distância de Mahalanobis

# List of Symbols

- $\alpha$  Primeira letra do alfabeto grego, refere-se ao ângulo de incidência do raio luminoso
- $\beta$  Segunda letra do alfabeto grego, conjunto de entrada para o *vertex shader*
- $\Gamma$  Terceira letra do alfabeto grego na forma maiúscula, usada pra se referir à uma imagem sintética
- $\theta$  Oitava letra do alfabeto grego, usada para se referir à uma pose em específico
- $\iota$  Nona letra do alfabeto grego, refere-se a uma matriz de transformação
- $\mu$  Decima segunda letra do alfabeto grego, refere-se a média
- $\Omega$  Ultima letra do alfabeto grego na forma maiúscula, usada como transformação resultante

# Capítulo 1

## Introdução

Há muitos anos a humanidade tenta capturar a realidade. A passagem do tempo foi marcada por diversas técnicas criadas com esse objetivo, desde dos pictogramas pré-históricos a métodos mais recentes como as fotografias, e com o advento dos computadores se tornou viável construir modelos tridimensionais altamente precisos, afim de fornecer dados para tais modelos foram criadas máquinas, conhecidas como escâneres 3D, capazes de extrair informações de objetos reais sobre sua geometria, forma e topologia.

Diferentes tecnologias podem ser usadas para a fabricação dessas máquinas cada uma com sua própria vantagem, limitação e custo. As que operam utilizando princípios ópticos encontram dificuldade com objetos transparentes, brilhantes ou reflexivos, mas obtém a cor da superfície. Já as baseadas em tomografia computadorizada industrial, utilizam técnicas de radiografia, fazem uso de raios-x e não tem enfoque na cor da superfície do objeto e sim na geometria do interior do objeto, essa abordagem é largamente aplicável a qualquer material ou objeto no qual um raio de radiação penetrante possa passar e ser detectado, maiores detalhes em [ASTM, 1992].

A informação coletada por escâneres 3D tem aplicação em diversas áreas como design industrial, engenharia reversa, prototipagem, controle de qualidade, ortopedia, próteses, cinematografia, jogos eletrônicos, documentação histórica de artefatos. Então a criação de ferramentas que cooperam ou complementam tais máquinas tem relevância.

### 1.1 Motivação

As cores da superfície de um objeto são uma informação importante, e podem ser codificadas de diversas formas, dentre elas o uso de texturas, que consistem de imagens digitais onde existe um mapeamento de *pixels* para regiões da superfície.

Caso exista a necessidade de novas texturas para objetos reais, previamente escaneados em 3D, um artista é encarregado de produzir alterações nas texturas pré-existentes, diminuindo a consistência da peça com a realidade ou é necessário repetir o processo de escaneamento completo ou parcial do objeto, o que demanda tempo e uma certa logística de mover o objeto até o escâner ou o escâner até o objeto. Este trabalho propõe a possibilidade de atualizar tais texturas utilizando fotografias ou quadros de vídeos, garantindo o realismo, promovendo uma agilização no processo e ainda permitindo que imagens de maior resolução sejam utilizadas.

## 1.2 Desafio

Ao analisar imagens de objetos reais alguns problemas devem ser considerados: como determinar a orientação do objeto, qual a distância focal da câmera utilizada, existência ou não de borramento, existência ou não de oclusão, ocorrência de ruído, se existem outros objetos na cena, resolução da imagem, influência da iluminação ambiente. A seguir seguem ilustrações de alguns desses problemas.



(a) Oclusão quase completa do sapo pela caneca  
 (b) Oclusão parcial do sapo pela caneca  
 (c) Oclusão quase completa da caneca pelo sapo

Figura 1.1: Cena onde o posicionamento da câmera produz oclusão dos objetos

Na figura 1.1 demonstra o problema da oclusão em uma cena. Observe que a esfera embora estando em poses diferentes só apresenta diferença visual no sombreamento, não havendo distinção em forma ou na cor de sua superfície. Observe também o problema existente em cubos, aqui exemplificado por um cubo de Rubik, o cubo de Rubik em particular possui partes articuláveis, onde em uma pose onde apenas uma face do cubo é visível não é possível determinar pela forma qual das faces é a visível.



Figura 1.2: Influência da luz na cena

Na figura 1.2 quatro situações de iluminação diferentes são observadas para a mesma cena, (a) uma fonte de luz situada a direita do objeto, (b) uma fonte de luz situada a esquerda/acima do objeto, (c) uma fonte de luz situada a frente/acima do objeto, (d) e uma situação onde não há uma fonte de luz próxima do objeto.

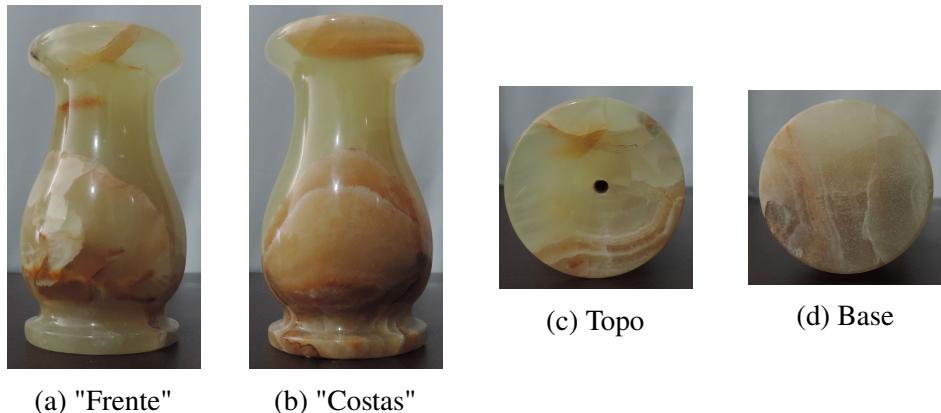


Figura 1.3: Objeto "cilíndrico" em diferentes poses

Observe que na figura 1.3 (a) e (b) são representadas diferentes poses o que só é perceptível visualmente devido à cor da superfície do objeto e não por distinção em sua forma. Já em (c) e (d) a diferença que se nota visualmente é um furo, o que também constitui uma diferença na forma, outro detalhe a ser observado é que enquanto a base é plana e o topo tem forma de funil.

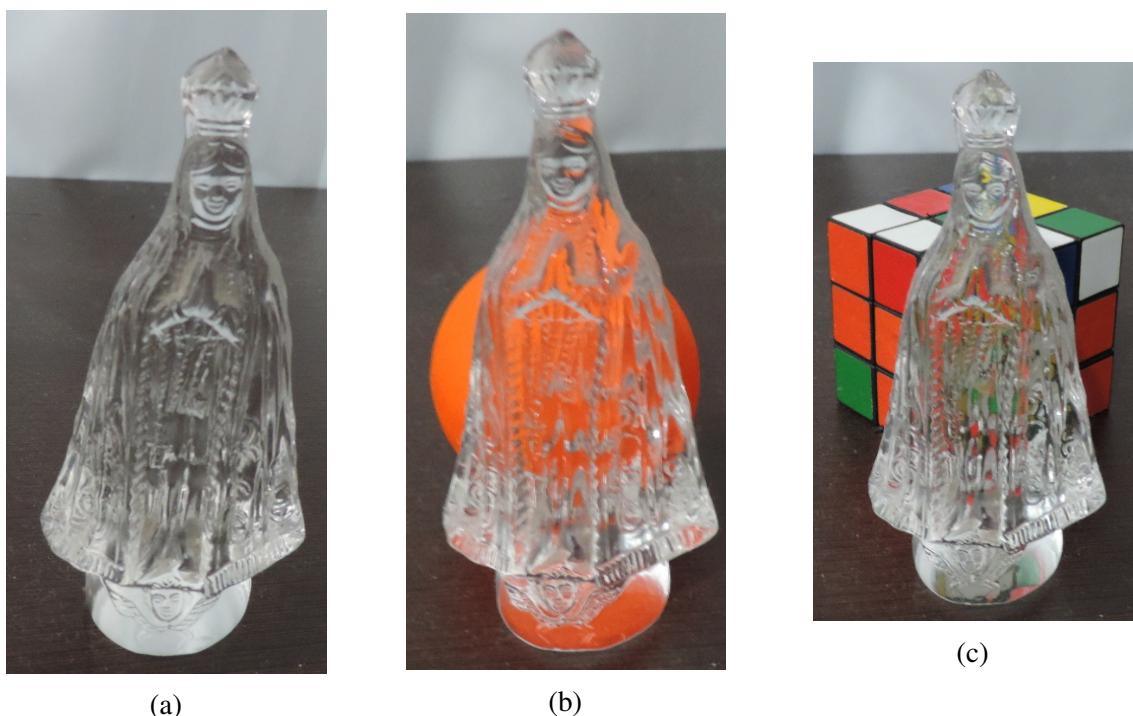


Figura 1.4: Influência do ambiente em objeto transparente

Observe na figura 1.4, como objetos atrás da estátua se misturam visualmente à própria estátua, onde em um objeto não ocorreria oclusão.

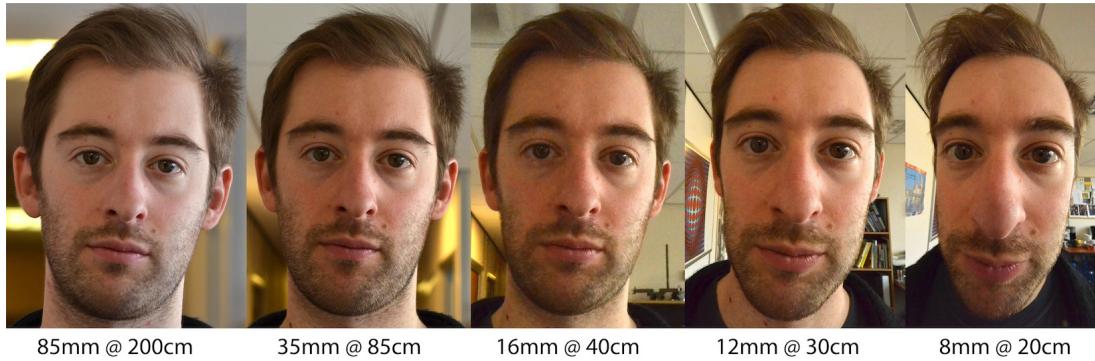


Figura 1.5: Comparação entre distâncias focais, fonte [Baker, 2012]

A figura 1.5 mostra a distorção causada pela distância focal, onde a medida em mm descreve o tipo de lente usada, e em cm a distância em que a câmera está em relação ao homem sendo fotografado.

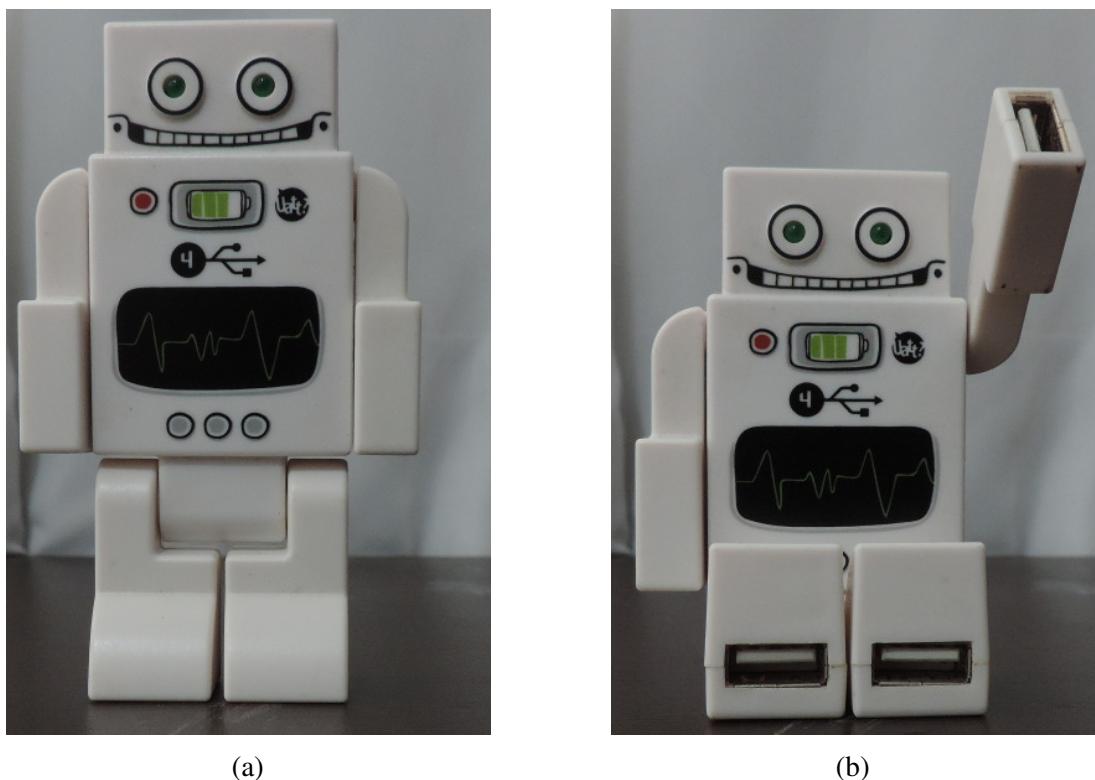


Figura 1.6: Objeto com peças articuláveis

Objetos com peças articuláveis como o da figura 1.6, tem sua forma alterada quando suas peças são articuladas.



(a) Variância do laplaciano = 45.64



(b) Variância do laplaciano = 67.16



(c) Variância do laplaciano = 92.89

Figura 1.7: Borramento de movimento

Na figura 1.7 é usada a métrica variância do resultado do filtro laplaciano para medir o borramento nas imagens.



Figura 1.8: Presença de ruído HSV em imagens

Observe na figura 1.8, a diferença entre (a), sem ruido e (b) com a presença de ruído gaussiano aplicado separadamente a cada um dos três canais de cor numa representação de cor em HSV.

Como os objetos não são limitados a uma única categoria, não é possível utilizar uma abordagem universal. Assim afim de evitar oclusão, a cena deve conter apenas o objeto de interesse. E não serão analisados por esse trabalho, objetos transparentes, objetos com peças articuláveis e objetos sem distinção ao terem sua orientação alterada, como esferas, ou que sofrem mudanças na sua forma (ex: objetos não sólidos).

### 1.3 Objetivo

Modelar o problema de se obter uma nova textura a partir de imagens digitais de um objeto real, afim de criar um algoritmo, tal que a entrada seja uma malha poligonal resultante de um escaneamento 3D e uma imagem digital desse mesmo objeto real, e a saída seja uma textura com mapeamento apropriado para a representação digital desse mesmo objeto. Visando atingir o objetivo principal, alguns objetivos específicos são requeridos, entre eles:

- Analisar o estado da arte relativo à estimativa de poses, remoção de plano de fundo, mapeamento de cor, segmentação de malhas, *UV unwrapping* e renderização.
- Desenvolvimento de um protótipo capaz de realizar o que o objetivo geral propõe.
- Discutir resultados obtidos.

### 1.4 Contribuição

Este trabalho realiza seu objetivo geral, sem informações prévias sobre a cena como ângulo entre o objeto e a câmera em relação ao plano, distância entre o objeto e a câmera, e ainda obtém uma textura parcial mesmo com um único ponto de vista.

### 1.5 Organização

Com o intuito de proporcionar um melhor entendimento, o presente trabalho está dividido em cinco capítulos, sendo eles:

O primeiro capítulo nomeado de Introdução apresenta a motivação do trabalho, os desafios de se lidar com objetos reais, qual o objetivo do trabalho, qual sua contribuição e a forma em que o texto está organizado. O segundo capítulo nomeado de Embasamento Teórico é apresentado alguns conceitos e algoritmos de forma breve afim de prover um embasamento teórico para os capítulos seguintes. O terceiro capítulo nomeado de Implementação será descrito como o protótipo desenvolvido funciona, como renderizar imagens de uma malha poligonal em uma certa pose, comparar a imagem projetada com uma fotografia, como obter uma textura 2D com mapeamento UV que pode ser usada com a malha original e é discutido alguns dos problemas e dificuldades encontrados durante o processo de implementação. O quarto capítulo nomeado de Experimentos descreve os experimentos realizados, os resultados obtidos, e conclusões relativas a cada experimento. Os experimentos aqui realizados buscam ilustrar o comportamento do protótipo em diferentes cenários. O quinto capítulo nomeado de Conclusão apresenta conclusões gerais sobre o trabalho, quais as vantagens da abordagem escolhida, quais as desvantagens da abordagem escolhida, qual a aplicação prática e é discutido trabalhos futuros.

# Capítulo 2

## Embasamento teórico

Neste capítulo serão apresentados alguns conceitos e algoritmos de forma breve afim de prover um embasamento teórico para os capítulos seguintes.

### 2.1 Modelo tridimensional

A seguir serão descritas diferentes representações de objetos tridimensionais. A representação em *polygon mesh* ou malha poligonal, define uma superfície, construída a partir de um conjunto de polígonos que compartilham arestas. Tal representação consiste de um grafo não direcionado que define a forma de um poliedro, onde cada vértice é um ponto no espaço  $\mathbb{R}^3$  e pode conter mais informações além da sua posição, como cor e coordenadas UV, as arestas do grafo, também conhecidas como *edges*, mostram a topologia entre os pontos. No contexto de computação gráfica, uma aresta não conecta um vértice a ele mesmo, um conjunto de arestas que formam um polígono é chamado de face. Uma face geralmente possui três ou quatro arestas e representa um polígono convexo. Uma malha composta unicamente de faces com três arestas é chamada malha triangular.

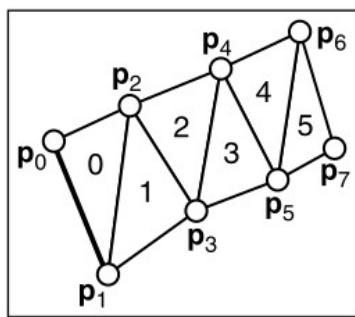


Figura 2.1: Exemplo de malha, fonte:[Shirley e Marschner, 2009]

A figura 2.1 mostra um exemplo de malha triangular onde  $p_0$  a  $p_7$  são vértices, e 0 a 5 são faces, onde 0 é formada pelos vértices  $p_0, p_1, p_2$  e as arestas  $p_0 - p_1, p_1 - p_2, p_0 - p_2$ .

A representação conhecida como nuvem de pontos, é um conjunto de pontos no espaço  $\mathbb{R}^3$ , nenhuma relação de topologia é descrita entre esses pontos. Existem diversos algoritmos para gerar uma malha a partir de uma nuvem de pontos, tais como Triangulação de Delaunay usada em [Su et al., 2016] e outras técnicas como em [Li et al., 2014], onde primeiro é reordenado os pontos usando *spatial curve order* e amostras sequenciais são geradas a partir dos dados

ordenados, em sequência é usada *witness complex theory* para reconstruir uma superfície *manifold* triangular a partir das amostragens usando como limitante os dados originais.

## 2.2 Mapeamento de cor

A seguir são descritas diferentes formas de atribuir cor à um objeto 3D. Na representação conhecida como Cor do vértice, ou *Vertex Color*, cada vértice tem uma informação adicional que é uma cor, tipicamente representada em RGB. Assim a cor dos *pixels* de cada face é obtida através da interpolação das cores de cada vértice que forma essa face.

Mapeamento utilizando texturas, para tal cada vértice da malha tem duas informações adicionais, as coordenadas u,v. As letras u e v são uma convenção para evitar confusão entre os eixos x,y,z tipicamente usados para representar o mundo<sup>1</sup>, com os eixos u,v usados para representar posições 2D na textura. As coordenadas u,v variam de [0, 1], onde u inicia-se na esquerda com 0 e termina na direita com 1 e v inicia-se abaixo com 0 e termina no topo com 1.

No caso de uma coordenada UV estar fora do intervalo [0, 1] ocorre um processo chamado *texture tiling* onde as imagens se repetem em ambos os eixos, por exemplo [3, 2] é equivalente a [1, 1] pois mapeiam o mesmo pixel.

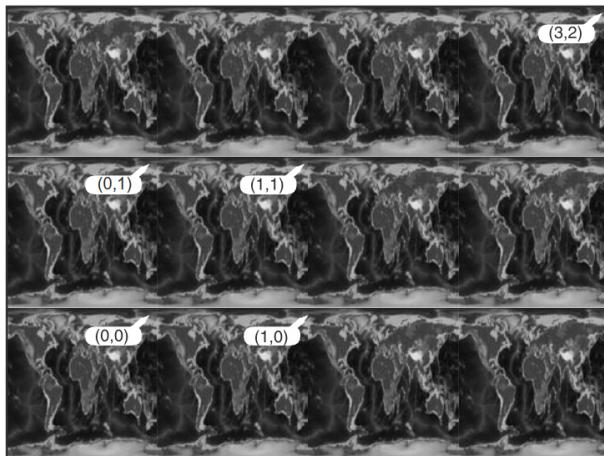


Figura 2.2: *Texture Tiling*, fonte:[Shirley e Marschner, 2009]

Pode existir a necessidade representar n texturas, sendo n um número natural, com mapeamentos UV distintos nesse caso é preciso que cada vértice v contenha as coordenadas  $\{u_0v_0, u_1v_1, \dots, u_nv_n\}$ .

No processo de renderização são utilizadas diferentes técnicas de interpolação, nesse caso são as coordenadas UV de cada vértice que são interpoladas e o pixel correspondente da textura é utilizado para colorir os *pixels* visíveis das faces.

---

<sup>1</sup>O mundo aqui citado refere-se ao mundo virtual onde o objeto 3D existe, também conhecido como *world space*.

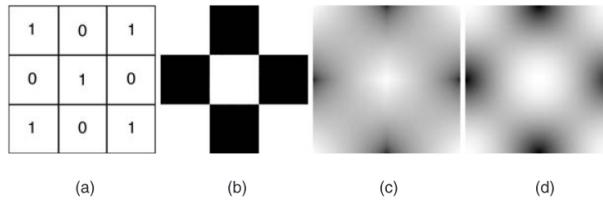


Figura 2.3: (a) representa uma imagem com pixels pretos e brancos, três estratégias de interpolação são apresentadas (b) nearest-neighbor, (c) bilinear e (d) hermite, fonte:[Shirley e Marschner, 2009]

A figura 2.3 ilustra como uma imagem (a) pode produzir diferentes texturas ao se utilizar diferentes estratégias de interpolação de valores.

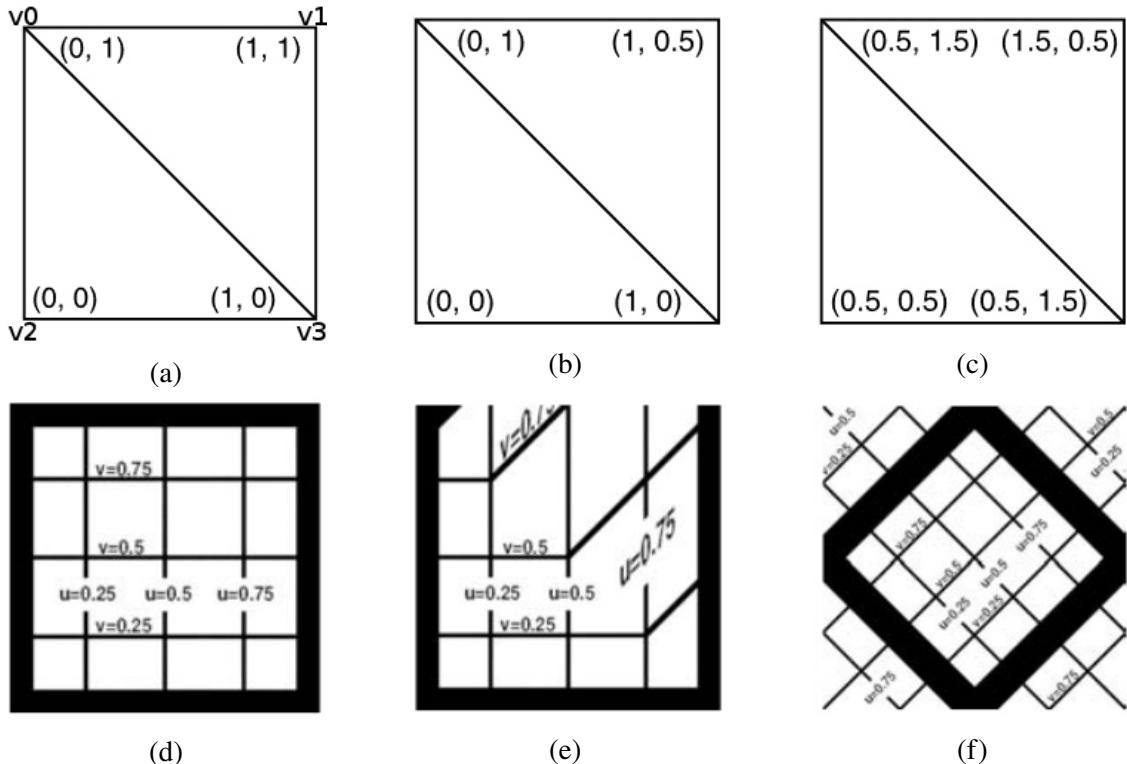


Figura 2.4: As sub-figuras (a), (b), (c) mostram diferentes malhas e suas respectivas coordenadas UV, já as sub-figuras (e), (f) mostram o resultado do mapeamento usando a textura de calibração vista em (d), fonte:[Shirley e Marschner, 2009] adaptado

Na figura 2.4 temos uma ilustração de como funciona o mapeamento UV, observe em (a) uma malha triangular com quatro vértices, onde o vértice  $v_0$  tem as coordenadas  $u=0, v=1$ , onde o vértice  $v_1$  tem as coordenadas  $u=1, v=1$ , onde o vértice  $v_2$  tem as coordenadas  $u=0, v=0$  e onde o vértice  $v_3$  tem as coordenadas  $u=1, v=0$ , e em (d) uma textura de calibração, imagine que a malha (a) é renderizada utilizando a textura (d), o que se observa como resultado é a própria textura (d), a mesma textura (d) é usada com a malha (b) e resulta em (e), também a textura (d) é usada com a malha (c) e resulta em (f).

Existem diferentes algoritmos conhecidos como *UV unwrapping* para obter-se coordenadas UV, dois em especial serão descritos a seguir. *Least Squares Conformal Maps* ou LSCM proposto por [Lévy et al., 2002], opera baseado em *Texture Atlas*, onde o modelo 3D é segmentado e cada segmento é mapeado para um *chart* homeomórfico a um disco, cada *chart* é

parametrizado, e os *charts* "desdobrados" são empacotados no espaço de textura. O LSCM utiliza um método de parametrização *quasi-conformal*, baseado na aproximação, utilizando mínimos quadrados, de equações de Cauchy-Riemann, tal abordagem tem como objetivo minimizar deformações de ângulo.

Um *texture atlas* quando usado para colorir um objeto 3D, deve seguir as seguintes premissas:

- As fronteiras do *chart* devem ser escolhidas afim de minimizar artefatos de textura.
- A amostragem do espaço de textura deve ser tão uniforme quanto possível.
- O Atlas deve fazer uso ótimo do espaço de textura.

A geração de um *texture atlas* pode ser decomposta nos seguintes passos:

- Segmentação: O modelo é particionado em um conjunto de *charts*.
- Parametrização: Cada *chart* é "desdoblado", ou seja, posto em correspondência com um subconjunto do  $\mathbb{R}^2$ .
- Empacotamento: Os *charts* são levados ao espaço de textura.

Implementações dos algoritmos de *UV Unwrapping* aceitam arestas com uma *flag* especial chamada de *seam* ou costura, que forçam com essas arestas sejam escolhidas como fronteiras no passo de segmentação da malha.

A figura 2.5 mostra as seguintes malhas *dinosaur*, *skull*, *bunny* e *horse*, onde na coluna à esquerda observa-se a imagem 3D projetada de cada malha, onde as linhas em vermelho são as arestas da malha sem *flag* e em preto são as arestas de costura, cada segmento da malha delimitado por arestas de costura produz um *chart* correspondente, na coluna à direita encontra-se o *texture atlas* correspondente de cada malha com todos os *charts* da malha, cada onde cada linha preta representa uma aresta da malha. A figura 2.6 mostra parâmetros sobre as malhas vistas na figura 2.5.

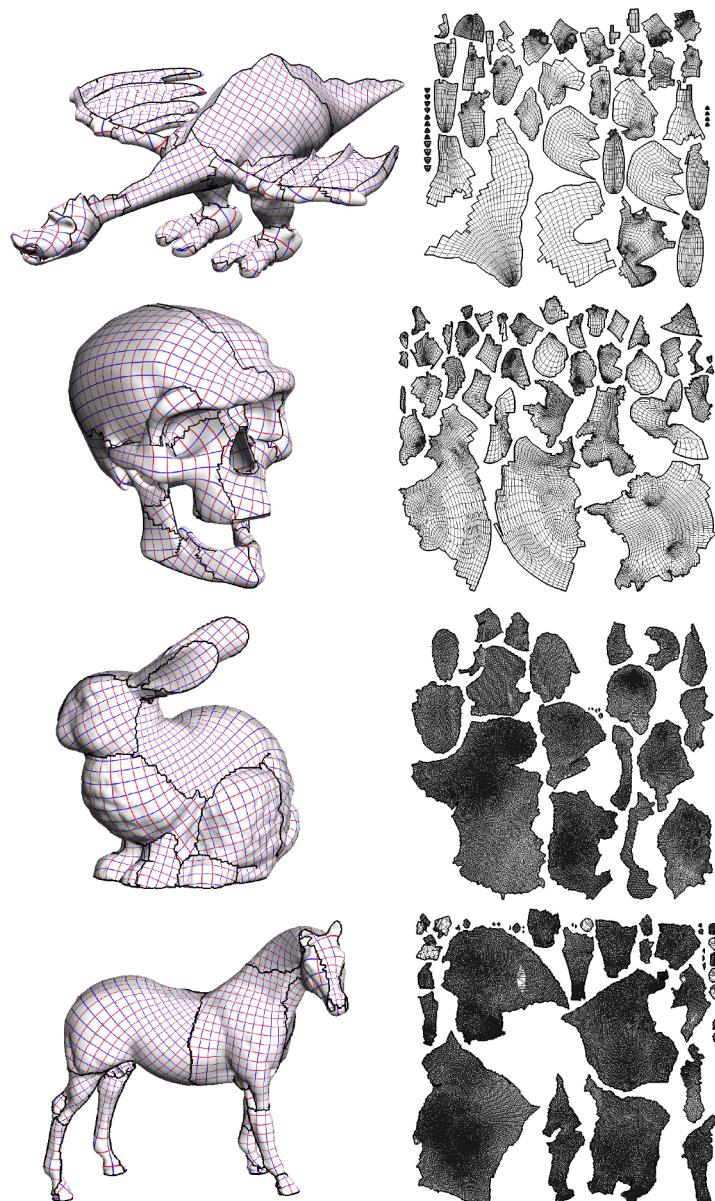


Figura 2.5: Diferentes *meshes* e seus respectivos *texture atlases* resultantes do LSCM, fonte:[Lévy et al., 2002]

	dinosaur	skull	bunny	horse
# vertices	14,669	16,949	34,834	48,485
# facets	14,384	15,124	69,451	96,968
# charts	43	40	23	44
segmentation time (s)	8	17	30	43
parameterization time (s)	10	23	95	190
packing ratio (rectangles)	0.48	0.51	0.43	0.37
packing ratio (our algo.)	0.55	0.55	0.6	0.58
stretch (before optim.)	2.9	2.5	1.16	1.14
stretch (after optim.)	1.26	1.55	1.14	1.12

Figura 2.6: Tabela de estatísticas e tempo relativas a figura 2.5, fonte:[Lévy et al., 2002]

Uma nova versão do LSCM foi proposta por [Ray e Levy, 2003] conhecido como *Hierarchical least squares conformal map* ou HLSCM.

ABF++ proposto por [Sheffer et al., 2005] um aprimoramento do *Angle Based Flattening* original [Sheffer e de Sturler, 2001], também opera gerando um *texture atlas*. O algoritmo é baseado na observação de que o conjunto de ângulos de uma triangulação 2D define unicamente a triangulação em relação a escalonamento global e transformações rígidas. Assim o ABF primeiramente computa a parametrização em espaço de ângulo e então a converte para coordenadas 2D. A formulação do espaço de ângulo faz com que essa técnica seja particularmente adequada para reduzir a distorção angular do mapeamento UV.

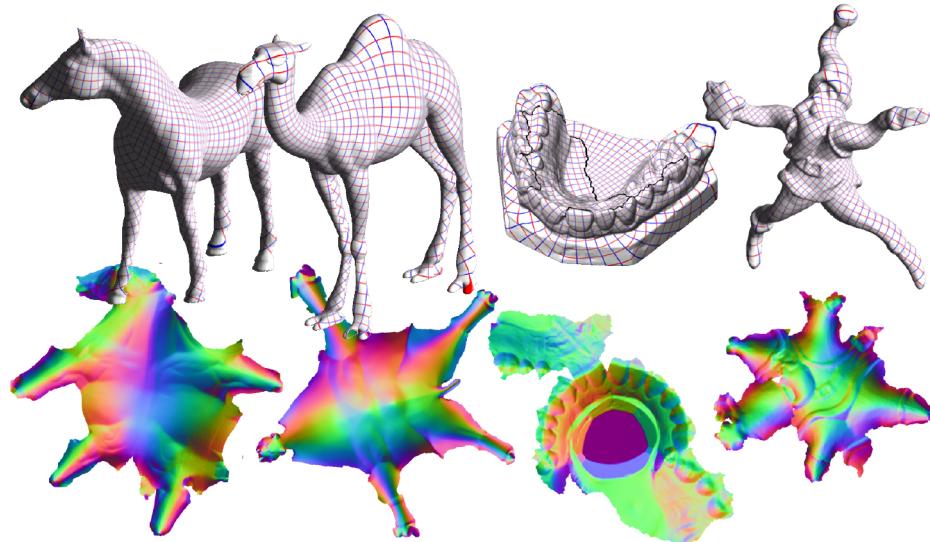


Figura 2.7: Diferentes *meshes* e seus *texture atlas* resultantes do ABF++, as parametrizações 2D foram coloridas usando 3D *normal map*, fonte [Sheffer et al., 2005]

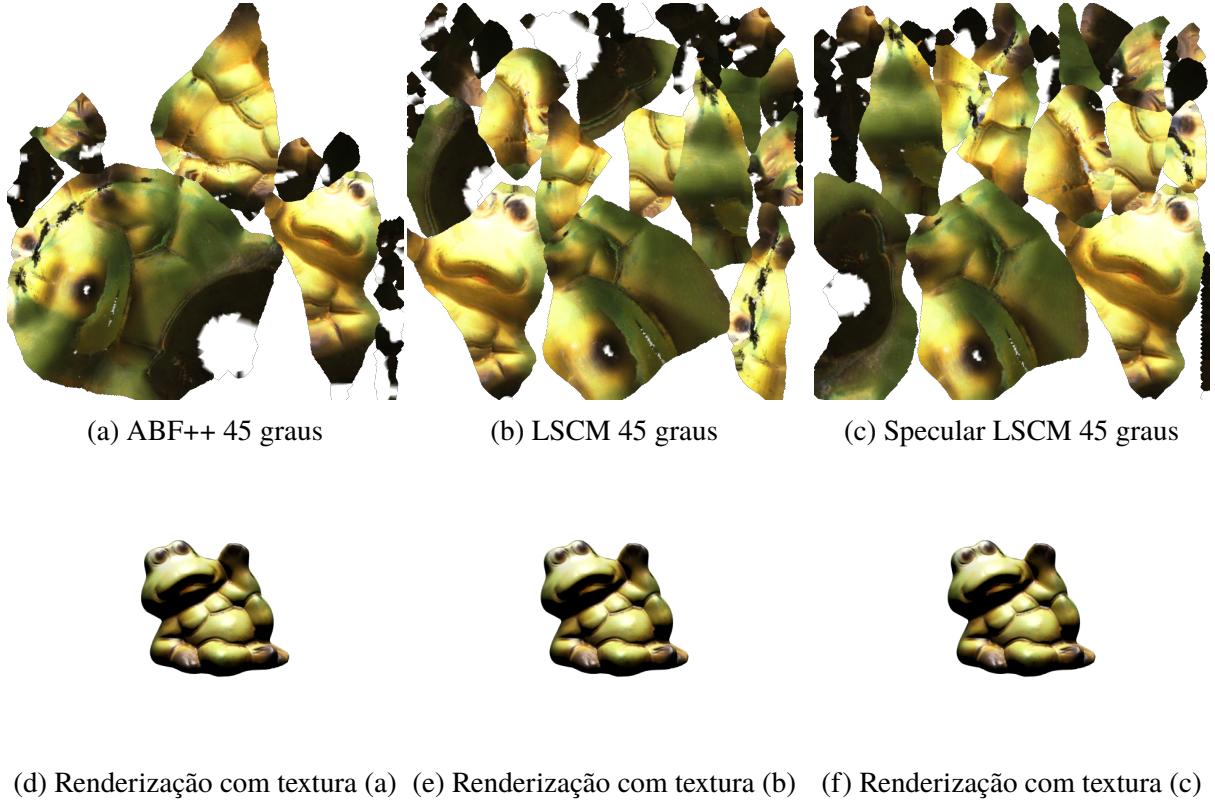


Figura 2.8: Texturas e seus respectivos mapeamentos, gerados usando implementação provida por [ALICE, 2017]

Vale aqui mencionar o trabalho de [Tarini, 2016] que propõe um Mapeamento UV codificado por volume, na qual cada ponto na superfície é mapeado para uma coordenada UV, em função unicamente da sua posição 3D, o que apresenta uma vantagem que é a reutilização do mesmo mapeamento para diferentes malhas da pirâmide LOD(onde a malha é simplificada em vários níveis em novas malhas mas que tentam preservar sua forma original). O trabalho de [Vallet e Lévy, 2009] que propõe um método semiautomático para a segmentação de malhas afim de obter *charts* mais próximos do que um artista *Modeler* escolheria ao marcar manualmente arestas como *seams*. E o trabalho recente de [Velho e Sossai Jr, 2017] que também trabalha com a construção de *Texture Atlas*.

Texturas podem simplesmente ser usadas para colorir superfícies com RGB, HSV, outros sistemas de cores ou ainda para incluir atributos especiais como no *Bump Mapping*, *Normal Mapping* e *Displacement Mapping*.

*Bump Mapping*, consiste numa técnica proposta por [Blinn, 1978] onde uma textura utiliza escalas de cinza como mapeamento de altura, essa altura é somada ao vetor normal a superfície, para falsificar o sombreamento do objeto e criar solavancos e rugas.

*Normal Mapping*, consiste numa técnica proposta por [Cohen et al., 1998] onde uma textura utiliza os canais RGB como coordenadas x,y,z do vetor  $\vec{n}$  normal a superfície, afim de falsificar o sombreamento do objeto ao ser renderizado, dando a ilusão de que o objeto possui o mesmo detalhamento que uma malha com mais vértices, arestas e faces.

*Displacement Mapping*, consiste numa técnica proposta por [Cook et al., 1987] onde a textura deforma a malha durante a etapa de renderização, criando um efeito parecido com o *Normal Mapping* com a diferença de que como a malha de fato é alterada assim é possível produzir deformações e calcular sombras mais realísticas.

## 2.3 Sombreamento Lambertiano e Blinn-Phong

O modelo mais simples de sombreamento é baseado na observação feita por Lambert no século dezoito: a quantidade de energia de uma fonte de luz que chega em uma área da superfície depende do ângulo da superfície com a luz. A superfície que esteja apontando diretamente em direção da luz, recebe iluminação máxima; a superfície tangente à direção da luz( ou apontando em direção contraria a luz) recebe nenhuma iluminação; e no intervalo entre esses a iluminação é proporcional ao cosseno do ângulo  $\alpha$  entre a normal à superfície e a fonte de luz (Figura 2.9). Isso leva ao modelo de sombreamento Lambertiano:

$$L = k_d I \max(0, n \cdot l) \quad (2.1)$$

Onde  $L$  é a cor do pixel;  $k_d$  é o coeficiente difuso, ou cor da superfície;  $I$  é a intensidade da fonte de luz. Por causa que  $n$  e  $l$  são vetores unitários,  $n \cdot l$  é equivalente a  $\cos\alpha$ .

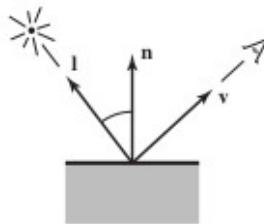


Figura 2.9: Geometria para o sombreamento lambertiano, fonte:[Shirley e Marschner, 2009]

O sombreamento Lambertiano é independente de visão: a cor da superfície não depende da direção a qual alguém olha. Muitas superfícies reais mostram algum grau de brilho, produzindo regiões de realce, ou reflexões especulares, as quais parecem se mover quando o ponto de vista muda. Como o sombreamento Lambertiano não produz nenhum realce, ele produz uma aparência *matte*, pálida, e acaba sendo incorporado como a parte difusa de outros modelos.

Um modelo bem simples e largamente utilizado para reflexão especular foi proposto por [Phong, 1975] e depois atualizado por [Blinn e Newell, 1976] para a forma que é comumente usada atualmente. A ideia é produzir reflexões onde o brilho é máximo quando  $v$  e  $l$  estão simetricamente posicionados através da superfície normal, que é quando a reflexão espelhada ocorre; a reflexão então diminui suavemente quando os vetores se afastam da configuração de espelhagem.

É possível dizer o quanto próximo se está da configuração de espelhagem comparando o *half vector*  $h$  (o bissetor do ângulo entre  $v$  e  $l$ ) com a superfície normal (Figura 2.10). Se o *half vector* está próximo da superfície normal, o componente especular deve ser claro, se está longe deve ser escuro. Esse resultado é obtido computando o produto escalar entre  $h$  e  $n$ , e então elevando o resultado a potencia  $p > 1$  para que o decrescimento ocorra mais rápido. O *half vector*  $h$  é descrito pela equação 2.2.

$$h = \frac{v + l}{\|v + l\|} \quad (2.2)$$

O modelo também comporta iluminação ambiente e múltiplas fontes de luz resultando em:

$$L = k_a I_a + \sum_{i=1}^N [k_d I_i \max(0, n \cdot l_i) + k_s I_i \max(0, n \cdot h_i)^p] \quad (2.3)$$

Onde  $k_a$  é o coeficiente ambiente,  $I_a$  a intensidade do iluminação ambiente,  $k_s$  é o coeficiente especular e  $I_i$  a intensidade da fonte de luz  $i$ , sendo  $N$  o número de fontes de luz na cena.

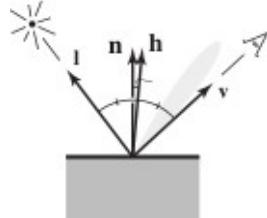


Figura 2.10: Geometria para o sombreamento Blinn-Phong, fonte:[Shirley e Marschner, 2009]

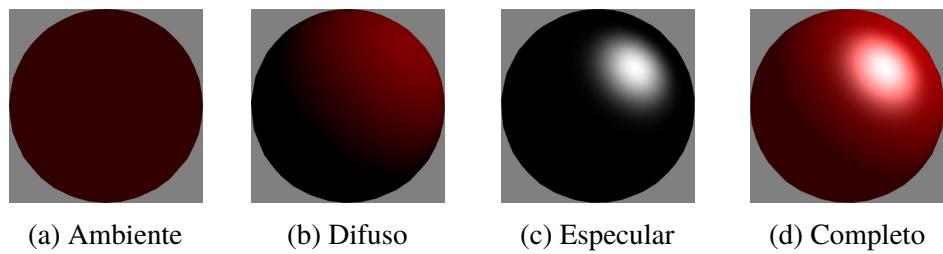


Figura 2.11: Componentes do sombreamento Blinn-Phong

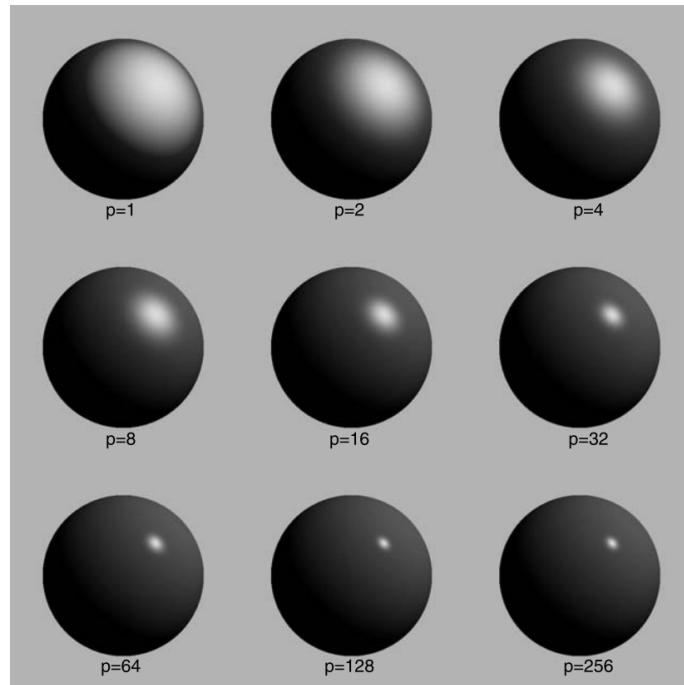


Figura 2.12: influência do coeficiente  $p$ , *Phong exponent*, fonte:[Shirley e Marschner, 2009]

## 2.4 Estimativa de pose

O problema de estimar a pose de um objeto é um problema clássico de visão computacional, e pode ser considerado um problema relacionado à detecção e identificação de objetos.

No trabalho de [Chin e Dyer, 1986], são descritas abordagens para o reconhecimento de objetos, onde é preciso comparar modelos afim de dizer se o objeto em questão é o objeto, desejado numa pose específica ou na verdade trata-se de outro objeto. É dito que as abordagens para tal são divisíveis em três categorias: 2D, 2.5D e 3D.

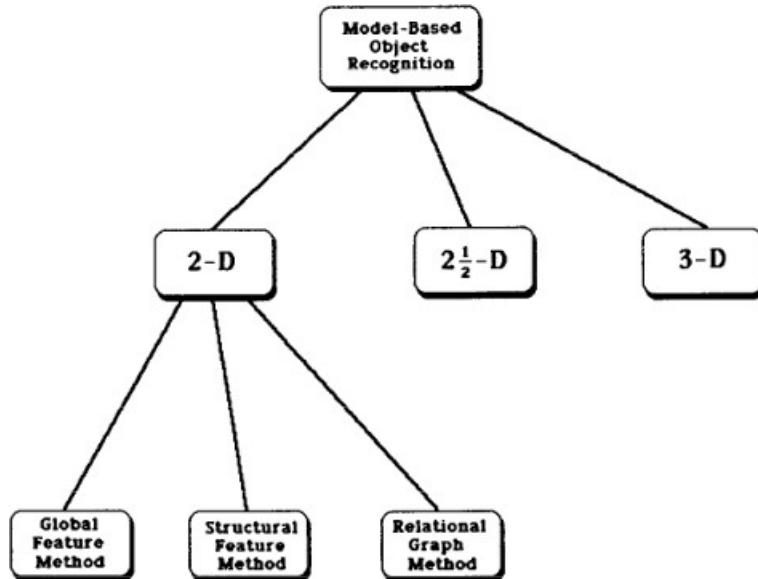


Figura 2.13: Organização da pesquisa, fonte:[Chin e Dyer, 1986]

Representações 2D, guardam a informação de uma visão 2D particular de um objeto (visão característica), como modelo e usam essa informação para identificar o objeto a partir de uma imagem 2D. Métodos de características globais, são usados por [Gleason et al., 1979] para identificar objetos como chaves de porca e porcas numa esteira. Tais métodos usam características como área, perímetro, número de buracos visíveis e outras características globais para modelar o objeto. Métodos de características estruturais, são usadas por [Perkins, 1978] na qual segmentos de fronteiras são utilizados para detectar partes de máquinas usando modelos 2D. Métodos utilizando grafos relacionais, são usados por [Yachida e Tsuji, 1977] para equiparar objetos com um modelo 2D usando técnicas de equiparação de grafos. Esses algoritmos baseados em representações 2D, requerem treinamento a priori do sistema, usando uma técnica baseada em exemplos.

Abordagens em 2.5D também são centradas em visões específicas, onde sabe-se que o objeto irá ocorrer numa visão particular. Eles diferem da abordagem 2D porque o modelo guarda informações adicionais como parâmetros intrínsecos da imagem e mapas de orientação-superfície. O trabalho feito por [POJE e DELP, 1982] explica o uso de parâmetros intrínsecos da cena na forma de *depth maps* e *needle maps* (orientação local de superfície). [Horn, 1989] obtém a forma do objeto a partir de informações de sombreamento e [Woodham, 1980] utiliza uma abordagem baseada em estéreo fotométrico.

Várias técnicas que se encaixam em representações 2D/2.5D são descritas por [Forsyth e Ponce, 2003], colocando o problema de reconhecimento de objetos como um problema de correspondência. Esses métodos obtêm a hipótese com base na correspondência de alguns pontos na imagem com o modelo. A hipótese é então validada contra os pontos conhecidos remanescentes.

Abordagens 3D: são utilizadas em situações onde o objeto de interesse pode aparecer na cena em múltiplas visões. Tipicamente abordagens 3D podem ser ou "representação exata" ou "*multi-view feature representation*". *Multi-view feature representation* usa um modelo de

composição consistindo de modelos 2D/2.5D para um conjunto limitado de visões. *Multi-view feature representation* é usada em conjunto do conceito de cilindros generalizados por [Rodney A. Brooks, 1981] para detectar diferentes tipos de motores industriais no sistema chamado ACRONYM. Os modelos usados no método de representação exata, de forma contraria, contêm uma representação exata do objeto 3D completo. Então a projeção 2D do objeto pode ser criada para a visão desejada. Esse tipo de método é geralmente considerado muito custoso em termos de processamento.

Vale aqui mencionar o problema de comparar uma malha com outra, o que é o enfoque do trabalho de [Johnson, 1997]. E o *Go-ICP* proposto por [Yang et al., 2016] que consiste numa variação do algoritmo ICP, que tem como objetivo minimizar a diferença entre duas nuvens de pontos. O trabalho de [Huttenlocher e Ullman, 1990] usa um modelo 3D que contém a localização de bordas. As bordas/contornos identificados na imagem 2D são equiparados com as bordas no modelo 3D para calcular a pose do objeto. Esse método tem sido implementado para objetos 3D simples. Entretanto não é claro se esse método irá funcionar bem com objetos com superfícies redondas e sem bordas claramente visíveis.

**Gradientes de Imagem.** Gradientes em escala de cinza são usados para estimar a pose 3D em um vídeo do trânsito a partir de uma câmera estacionaria por [Kollnig e Nagel, 1997]. O método compara gradientes da imagem invés de segmentos de borda, para melhor performance. Gradientes de imagem provenientes de modelos poliedrais projetados são comparados com gradientes de imagem vindo de quadros de vídeo. A pose é formulada usando três graus de liberdade; dois para a posição  $x$ ,  $y$  e um para a orientação angular. [Tan e Baker, 2000] usam gradientes de imagem e um algoritmo baseado em transformação de Hough, para estimar a posição de um veiculo em cenas do transito, também nesse caso é utilizado três graus de liberdade para representar a pose. A representação com apenas três graus de liberdade é útil quando a câmera é estacionaria, entretanto em situações mais generalizadas onde a câmera pode estar em uma posição desconhecida é insuficiente.

**Modelos de Forma Implícitos.** O trabalho de [Arie-Nachimson e Basri, 2009] usa "Modelos de Forma Implícitos" para reconhecer objetos 3D a partir de imagens 2D. O modelo consiste de: um conjunto de *features* aprendidas, a localização 3D e as visões onde o objeto é visível. O processo de aprendizado é refinado usando métodos de fatoração. A estimativa de pose consiste na avaliação de transformações de *features* que provem o melhor equiparo. Um modelo tipico necessita de aproximadamente 65 imagens para ser treinado.

**Métodos baseados em *features*.** Onde se tenta simultaneamente estimar a pose e resolver o problema de correspondência entre os pontos. [David et al., 2004] propõe uma variação do algoritmo *POSIT*, nomeada como *softPOSIT*. [Moreno-Noguer et al., 2008] comparação entre o *blind PnP* e o *softPOSIT*. [Shi e Ulrich, 2016] Adaptam o *softPOSIT* para estimativa de poses de espaço-naves usando câmera monocular. [Snavely et al., 2008] Propõe usar imagens da internet para recriar digitalmente locais famosos, numa abordagem parecida com o Google StreetView<sup>2</sup>. Tipicamente um problema com esses métodos é se existe regiões de alta refletância, o que causa falsos positivos, e se a superfície tem grandes regiões homogêneas com consequentemente *features* esparsas.

Métricas de distância podem ser usadas para representar a distância entre dois conjuntos de dados e assim dar uma medida de sua similaridade. Assim, métricas de distância podem ser usadas para medir a similaridade entre diferentes imagens 2D, assim como imagens 2D e a projeção 2D de um modelo 3D. Uma métrica de distância simples é a distância Euclidiana. Entretanto essa medida tem a desvantagem de ser dependente de escala. Assim uma medida como a distância de Mahalanobis [Mahalanobis, 1936] que é invariante à escala é mais apropriada para comparações,

---

<sup>2</sup>Google Maps, <http://maps.google.com>.

como é o caso em [Xing et al., 2003] que usa para *clustering* e [Deriche e Faugeras, 1990] que usa para equiparar segmentos de reta em uma sequência de tempo onde as imagens variam.

## 2.5 Distância de Mahalanobis

A distância de Mahalanobis (DM) proposta por [Mahalanobis, 1936], é a distância entre dois pontos em um espaço multivariado. Em um espaço Euclidiano regular, variáveis são representadas por eixos ortogonais entre si; caso esteja lidando com um plano euclideano a distância entre dois pontos pode ser medida com uma régua. Para variáveis não-correlacionadas, a distância Euclidiana é igual a DM. Entretanto, se duas variáveis são correlacionadas, os eixos não estão em ângulos retos, e medições se tornam impossíveis com uma régua, e caso seja necessário ter mais que três variáveis correlacionadas, não é possível plotá-las em um espaço 3D regular. A DM resolve esse problema de medida, pois ela mede as distâncias entre os pontos, mesmo pontos correlacionados por múltiplas variáveis.

A distância de Mahalanobis mede a distância relativa ao centroide, um ponto central que pode ser considerado como a média geral de um conjunto de dados multivariados. O centroide é o ponto em um espaço multivariado no qual as médias de todas as variáveis se intersectam. Assim quanto maior a DM, mais longe do centroide o ponto se encontra.

A distância de Mahalanobis de uma observação  $\vec{x} = (x_1, x_2, \dots, x_n)^\top$  de um conjunto de observações com média  $\vec{\mu} = (\mu_1, \mu_2, \dots, \mu_n)^\top$  e matriz de covariância  $C$ , pode ser definida como:

$$D_{\text{Mahalanobis}}(\vec{x}) = \sqrt{[(\vec{x} - \vec{\mu})^\top * C^{-1} * (\vec{x} - \vec{\mu})]} \quad (2.4)$$

A distância de Mahalanobis, também pode definida como a medida de dissimilaridade entre dois vetores aleatórios  $\underline{x}$ ,  $\underline{y}$  com a mesma distribuição e matriz de covariância  $C$ , pode ser definida como:

$$D_{\text{Mahalanobis}}(\vec{x}, \vec{y}) = \sqrt{[(\vec{x} - \vec{y})^\top * C^{-1} * (\vec{x} - \vec{y})]} \quad (2.5)$$

## 2.6 Função de perda invariante à luz

A *novel illumination-invariant loss function*, aqui traduzida como Função de perda invariante à luz, uma versão adaptada da distância de Mahalanobis, é descrita por [Jayawardena et al., 2013] e [Jayawardena, 2013] para a comparação de imagens 2D e a projeção 2D de um modelo 3D. Nas subseções seguintes são descritas informações acerca dessa função.

### 2.6.1 Formulação teórica

Seja  $P = \{1, \dots, n_x\} \times \{1, \dots, n_y\}$  o conjunto de coordenadas em pixéis com  $|P|$  (inteiros), e  $p = (x, y) \in P$  uma coordenada de pixel. Seja  $F : P \rightarrow \mathbb{R}^n$  uma fotografia com  $n$  atributos, e  $M_\theta : P \rightarrow \mathbb{R}^m$  a projeção 2D de um objeto 3D usando os parâmetros de pose  $\theta$  com  $m$  atributos. Possíveis atributos incluem cor, informações locais de textura ou normais à superfície. A seguir considere o caso de uma fotografia em escala de cinza ( $n = 1$ ), e considere como atributos: normais à superfície e brilho da superfície ( $m = 4$ ) de um modelo 3D (projeto).

## 2.6.2 Modelo de reflexão Lambertiano

O objetivo aqui é uma medida de distância entre  $F : P \rightarrow \mathbb{R}^n$  e  $M_\theta : P \rightarrow \mathbb{R}^m$ , a qual pode ser otimizada variando os parâmetros de pose para obter a pose 3D correta. Um modelo de reflexão Lambertiano descrito na seção 2.3 não é realístico o suficiente para resultar em uma distância zero utilizando fotografias reais, mesmo na pose correta. Mesmo assim é demonstrado por [Jayawardena et al., 2013] que uma adaptação do modelo Phong resulta em um mínimo local na pose correta, o que é suficiente para obter os parâmetros de pose com certa margem de erro.

Assim considere o Sombreamento Phong, previamente descrito na seção 2.3, sem reflexão especular. Seja  $I_a \in \mathbb{R}$  a intensidade da iluminação global de ambiente e  $I_d \in \mathbb{R}$  e a intensidade de luz difusa de uma cena 3D. Seja  $L \in \mathbb{R}^3$  o vetor apontando em direção da fonte de luz (ou a media ponderada dos vetores no caso de múltiplas fontes de luz). Para cada ponto de superfície  $p$ , seja  $k_a(p) \in \mathbb{R}$  e  $k_d(p) \in \mathbb{R}$  a constante de reflexão ambiente e a constante de reflexão difusa (brilho intrínseco da superfície) e  $\phi(p) \in \mathbb{R}^3$  seja a o vetor unitário(interpolado) da normal à superfície. Assim, a intensidade  $I$  do ponto  $p$  na projeção  $M_\theta(p)$  é

$$I(p) = k_a(p)I_a + k_d(p)(L^\top \phi(p))I_d + I_0 \equiv A \cdot M_\theta(p) + b \quad (2.6)$$

O que pode ser reescrito como

$$M_\theta(p) := (k_a(p), k_d(p)\phi(p))^\top \in \mathbb{R}^{4 \times 1} \quad (2.7)$$

onde nessa forma são representados os parâmetros de superfície (conhecidos) e,  $A := (I_a, I_d L^\top) \in \mathbb{R}^{1 \times 4}$  são as quatro constantes de iluminação global (desconhecidas), e  $b = I_0$ . Como aqui é aproximada a iluminação da cena com o modelo de sombreamento lambertiano, é incluso o termo  $I_0$  o qual não existe no modelo Phong original. O termo  $I_0$  age como uma correção parcial aos elementos de iluminação não-lambertianos presentes na fotografia real. Já que  $I(\cdot)$  é linear em  $A$  e  $M_\theta(\cdot)$ , assim qualquer renderização é uma função global linear simples de  $M_\theta(p)$ . Esse modelo se mantém verdadeiro mesmo com múltiplas fontes de luz, entretanto não acomoda auto oclusão.

## 2.6.3 Medida de distância invariante à luz

Então é medida a similaridade da projeção de modelo 3D  $M_\theta$  com a fotografia  $F$  usando a distância  $D(F, A\dot{M}_\theta + b)$ , sendo  $D$  uma medida de distância (como por exemplo: distância euclidiana, distância absoluta ou distância de Mahalanobis). Como é de interesse nesse caso não assumir qualquer conhecimento extra sobre as condições de iluminação  $A$  sobre a qual a foto foi tirada, o uso direto de  $D$  é descartado. Aqui é buscado a "distância" entre  $F$  e  $M$ , a qual é independente de  $A$  e é zero se e somente se existe uma condição de iluminação  $A$  tal que  $F$  e  $A\dot{M}_\theta + b$  coincidem.

De fato isso é possível, se (invés de definir  $M_\theta$  como um projeção de  $M$  dependente de  $A$ ) utilizar o brilho e normais à superfície independente de  $A$  (como definido na equação 2.7). Assim a distância é definida da seguinte forma. Seja

$$\bar{F} := \frac{1}{|P|} \sum_{p \in P} F(p) \in \mathbb{R} \quad \bar{M}_\theta := \frac{1}{|P|} \sum_{p \in P} M_\theta(p) \in \mathbb{R}^4 \quad (2.8)$$

a média dos atributos da fotografia e da projeção de modelo 3D, e

$$C_{FM_\theta} := \frac{1}{|P|} \sum_{p \in P} (F(p) - \bar{F})(M_\theta(p) - \bar{M}_\theta)^\top \in \mathbb{R}^{1 \times 4} \quad (2.9)$$

a matriz de covariância cruzada entre  $F$  e  $M_\theta$ . E  $C_{M_\theta F} = C_{F M_\theta}^\top \in \mathbb{R}^{4 \times 1}$ , as matrizes de covariância  $C_{FF} \in \mathbb{R}^{1 \times 1}$  e  $C_{M_\theta M_\theta} \in \mathbb{R}^{4 \times 4}$  de maneira similar. Na seção 4.3 de [Jayawardena, 2013] é descrito com detalhes como é derivada a Função de perda invariante à luz, na qual se chega na seguinte equação:

$$\min\{D(X, Y), D(Y, X)\} = \min\{n, m\} - \text{tr}[C_{XX} C_{YY}^{-1} C_{YX} C_{XX}^{-1}] \quad (2.10)$$

Então quando  $X = F, Y = M_\theta, n = 1$  e  $m = 4$ .

$$D(F, M_\theta) = D_I(\theta) = 1 - \text{tr}[C_{FM_\theta} C_{M_\theta M_\theta}^{-1} C_{M_\theta F} C_{FF}^{-1}] \quad (2.11)$$

Note que essa expressão não envolve  $A$ , que representa os parâmetros de iluminação. Na seção 4.3 de [Jayawardena, 2013] também é demonstrado que  $D_I(\theta)$  é invariante a transformações lineares regulares, dos valores de atributos de pixel de  $F$  e  $M_\theta$ , e zero se e somente existe uma transformação linear perfeita dos valores de atributos de pixel de  $M_\theta$  para  $F$ . Isso faz com que seja desnecessário saber as constantes exatas de reflexão da superfície ( $k_a(p) \in \mathbb{R}, k_d(p) \in \mathbb{R}$ ). Assim podemos obter <sup>3</sup>

$$D_I(\theta) = \min_{A,b} D_{\text{Mahalanobis}}(F, A \cdot M_\theta + b) \quad (2.12)$$

Isso implica que  $D_I(\theta)$  é zero se e somente se existe uma condição de iluminação  $A$  tal que  $F$  e  $M_\theta$  coincidem.

## 2.7 Downhill-Simplex

É constatado por [Jayawardena et al., 2013] que a Função de perda invariante à luz, não é facilmente diferenciável, uma vez que as transformações de escala, rotação, translação, e perspectiva quando combinadas para gerar a imagem 3D projetada produzem um comportamento descontínuo na função de perda invariante à luz, impedindo que métodos como o gradiente descendente que dependem da derivada para seu funcionamento sejam usados. Então é proposto a utilização do método *Downhill simplex*, uma vez que ele é baseado inteiramente em avaliações do valor de uma função, entretanto embora isso resolva o problema da descontinuidade também faz com que a confiabilidade da otimização seja prejudicada pela existência de mínimos locais. Felizmente, o método *Downhill-Simplex* possui uma propriedade útil. Na maioria dos casos, se o algoritmo é reinicializado com os parâmetros de pose do mínimo local e a otimização é realizada novamente, o simplex sai do mínimo local e eventualmente converge para o global mínimo. Lembrando que a parametrização adequada é importante para o otimizador dar bons resultados.

O método *Downhill simplex* proposto por [Nelder e Mead, 1965], também conhecido como "Algoritmo da Amoeba" ou "Nelder-Mead simplex" é baseado em um simplex, o volume mais simples no espaço N-dimensional, o qual é construído por N+1 pontos. Dado uma função continua  $y = f(x_1, \dots, x_n)$ . O objetivo é encontrar o mínimo local  $y_{\min}$  dessa função com o ponto  $X_{\min}$  pertinente. Então é construído um simplex de N+1 pontos com os vetores  $X_1, \dots, X_n, X_{n+1}$ .

O procedimento é como segue. Os vetores são avaliados pela função  $f$  e em seguida ordenados em ordem crescente usando o valor  $y$ , resultante de cada avaliação, como comparador. Sendo assim o melhor ponto  $(y_{\min}, X_{\min}) = (y_1, X_1)$ , o pior ponto  $(y_{\max}, X_{\max}) = (y_{n+1}, X_{n+1})$  e o

---

<sup>3</sup>a notação  $\text{tr}[]$  usada na Equação 2.11 refere-se à operação de álgebra linear traço

segundo pior ponto ( $y_n, X_n$ ) são determinados. Então é possível calcular o ponto de espelhamento

$$M = \frac{1}{N} \sum_{i=1}^n X_i \quad (2.13)$$

Usando todos os pontos exceto  $X_{n+1}$ . Em seguida um novo simplex é calculado usando a reflexão do pior ponto a partir do ponto de espelhamento

$$R = M + \alpha(M - X_{max}) \quad (2.14)$$

Existem três outros métodos para construir um novo simplex: a expansão para acelerar a redução do simplex para um simplex de volume menor,

$$E = M + \beta(R - M) \quad (2.15)$$

a contração para manter o simplex pequeno,

$$C = M + \gamma(X_{max} - M) \quad (2.16)$$

e a compressão para manter os pontos em torno do melhor ponto atual.

$$X_i = X_{min} + \sigma(X_i - X_{min}) \quad (2.17)$$

Os métodos são utilizados num laço até que o critério de parada seja atingido, como demonstrado na figura 2.15, onde é mostrado na forma de um fluxograma as comparações e atribuições feitas pelo algoritmo.

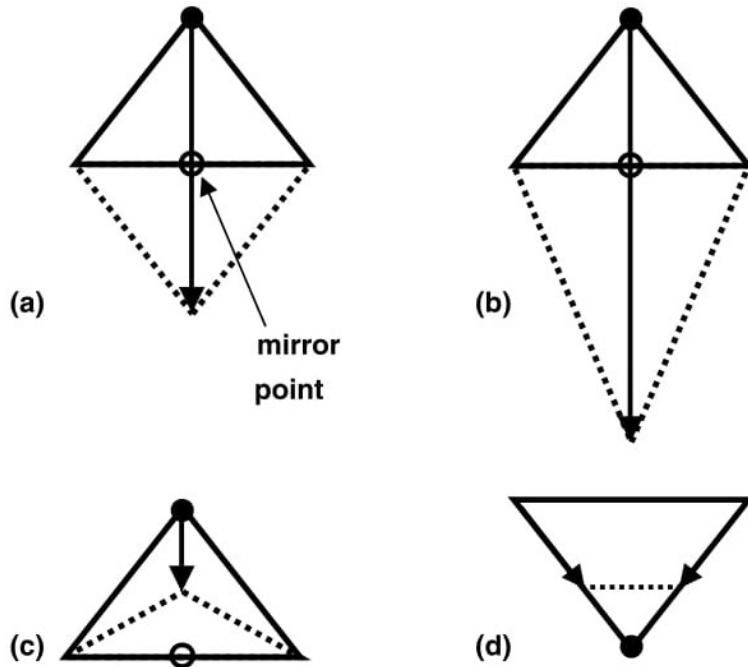


Figura 2.14: Ilustração dos quatro métodos dentro do *Downhill simplex* para definir novos pontos do simplex. (a) Reflexão,(b) Expansão,(c) Contração,(d) Compressão. Fonte [Fox, 2013]

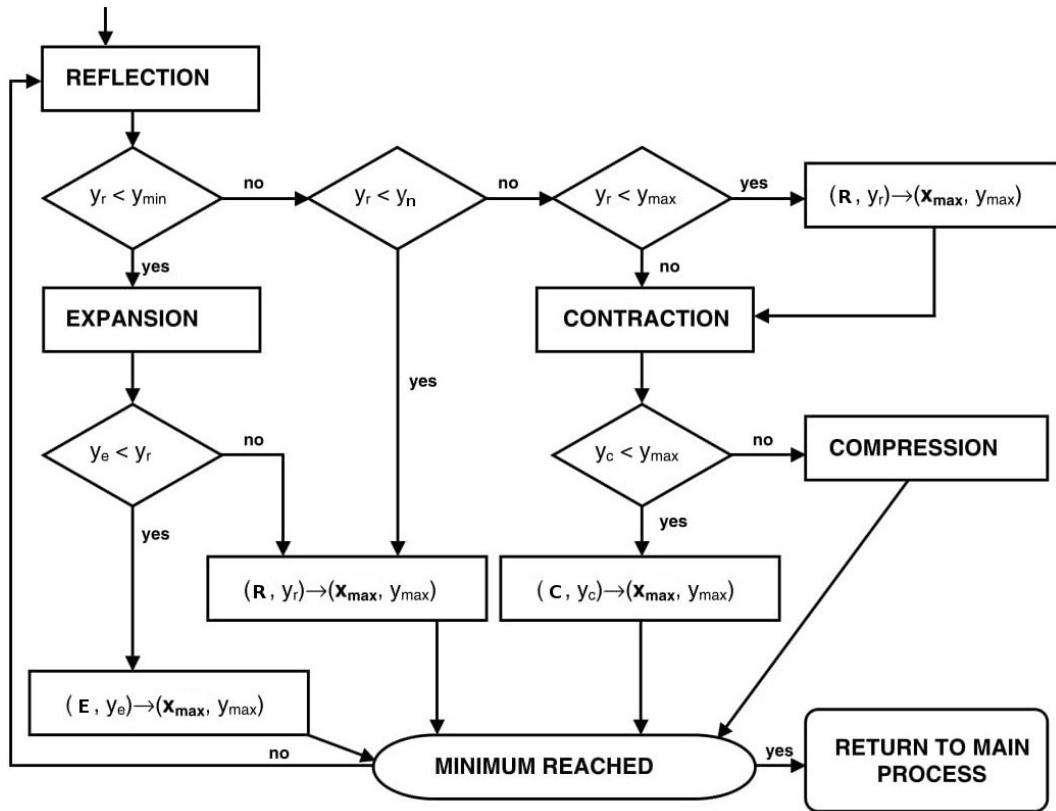


Figura 2.15: Fluxograma do Algoritmo *Downhill Simplex*, fonte [Fox, 2013] adaptado

Um dos critérios de parada propostos por [Nelder e Mead, 1965] é

$$I = 3.16(n + 1)^{2.11} \quad (2.18)$$

Onde  $I$  é o número de iterações do método. Vale ainda mencionar que o *Downhill-Simplex* é um método popular de busca direta para otimização multidimensional sem restrições, para otimização com restrições [Nelder e Mead, 1965] sugerem duas formas de integrar restrições à otimização. Um método é modificar a função de tal forma que valores indesejados da variável independente resultem em valores muito grandes e fiquem fora do intervalo de interesse. O outro método é transformar a variável independente (por exemplo: usar logaritmos para excluir valores negativos).

## 2.8 Remoção de plano de fundo

Embora existam diversos algoritmos para a remoção de plano de fundo, aqui só será descrito o *GrabCut*, uma vez que foi o algoritmo implementado no protótipo. Proposto por [Rother et al., 2004], O *GrabCut* opera de forma semiautomática recebendo um retângulo (quatro coordenadas x,y) ou uma máscara(matriz binaria) e uma imagem, o método itera  $i$  vezes e um resultado intermediário é obtido, o usuário então pode afirmar *pixels* como plano de fundo ou plano de fundo, o método então itera novamente e produz um novo resultado, esse processo ocorre até o usuário decidir que o resultado atual é o final.

## 2.9 Conclusão do capítulo

Nesse capítulo vimos o que é uma malha poligonal, nuvem de pontos, cor de vértice, mapeamento UV, os algoritmos LSCM e ABF para *UV Unwrapping*, Sombreamento Lambertiano e Blinn-Phong, diferentes técnicas para estimar pose, uma descrição sobre a função de perda invariante a luz, o método otimizador de função Downhill-Simplex e o algoritmo *GrabCut*.

Levando essas informações em consideração, é possível dizer que o objetivo de igualar a pose de um objeto real escaneado, do qual se tem a malha poligonal, com a pose do mesmo objeto em uma fotografia ou quadro de vídeo, para qualquer visão do objeto em que não exista oclusão e sem a necessidade de métodos de aprendizado, pode ser atingido usando uma abordagem 3D exata e uma métrica de distância, nesse caso a função de perda invariante a luz.

# Capítulo 3

## Implementação

Neste capítulo será descrito como renderizar imagens de uma malha poligonal em uma certa pose, comparar a imagem projetada com uma fotografia e finalmente obter uma textura 2D com mapeamento UV que pode ser usada com a malha original. A titulo de facilitação o valor resultante da chamada da função de perda invariante a luz será chamado de distância *loss*.

### 3.1 Pré-processamento

As Fotografias usadas devem sofrer um pré-processamento de segmentação, remoção de plano de fundo e centralização da imagem resultante, sombras e regiões de realce são preservadas como na imagem original. Vídeos devem ter seus *frames* extraídos e salvos como imagens digitais,e que sofrem o pré-processamento similar as fotografias.

Malhas precisam estar em um escala tal que os vértices estejam no intervalo  $[-1, 1]$ , com a profundidade no eixo z, o eixo y como a altura e o eixo x a largura, com a origem como o centro do *bounding box*, e com mapeamento UV.

### 3.2 Renderização

Um protótipo foi desenvolvido afim de realizar os experimentos, a implementação foi realizada na linguagem de programação C++, com programas auxiliares escritos em Python, a API de renderização utilizada foi o OpenGL [OpenGL, 2016].

O OpenGL opera utilizando *Shaders*, que consistem de um código em GLSL que é executado na GPU. tipicamente *Shaders* existem em pares *Vertex Shader* e *Fragment Shader*, onde o *Vertex Shader* é o primeiro a ser executado e opera sobre os vértices, com um entrada  $\beta$ , tipicamente a operação é multiplicar a matriz de transformação resultante com as posições do vértice. Em seguida os valores de saída do *vertex shader* são interpolados e enviados como entrada para o *Fragment Shader* que opera sobre os *pixels* do *buffer* que contem o resultado da renderização. A operação de *clipping* ocorre baseada nos valores *near* e *far* da matriz de projeção. As matrizes utilizadas pelo OpenGL são *column-major* invés do convencional *row-major* do C++, no texto atitulo de facilidade de compreensão será mantida a notação convencional. Mais detalhes sobre o funcionamento e implementação do OpenGL disponível em [OpenGL, 2016].

### 3.3 Matrizes de Transformação *Affine*

Matriz de Translação *Affine* T

$$T = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.1)$$

Matriz de Rotação *Affine* R

$$R = \begin{bmatrix} r_{1x} & r_{1y} & r_{1z} & 0 \\ r_{2x} & r_{2y} & r_{2z} & 0 \\ r_{3x} & r_{3y} & r_{3z} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.2)$$

Sendo  $r_1, r_2$  e  $r_3$  vetores ortonormais que formam uma base ortonormal.

Matriz de Escalonamento *Affine* S

$$S = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.3)$$

A matriz  $C = T \cdot R \cdot S$ , conhecida como *Model transformation Matrix*, representa as transformações do objeto em relação ao espaço o *World Space*, o espaço  $\mathbb{R}^3$  em que ocorre a cena

A Matriz V conhecida como *View space transform matrix* tem como objetivo posicionar a câmera em relação ao espaço, a câmera aqui segue um modelo de *Pin-hole camera*.

$$V = \begin{bmatrix} Right_x & Up_x & Forward_x & Pos_x \\ Right_y & Up_y & Forward_y & Pos_y \\ Right_z & Up_z & Forward_z & Pos_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.4)$$

Projeção Ortográfica

$$P' = \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & \frac{2}{n-f} & -\frac{n+f}{n-f} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.5)$$

Projeção Perspectiva

$$P = \begin{bmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{n+f}{n-f} & \frac{2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{bmatrix} \quad (3.6)$$

tal que  $n = near, f = far, r = right, l = left, t = top, b = bottom$   
 Transformação resultante  $\Omega = P \cdot V \cdot M$

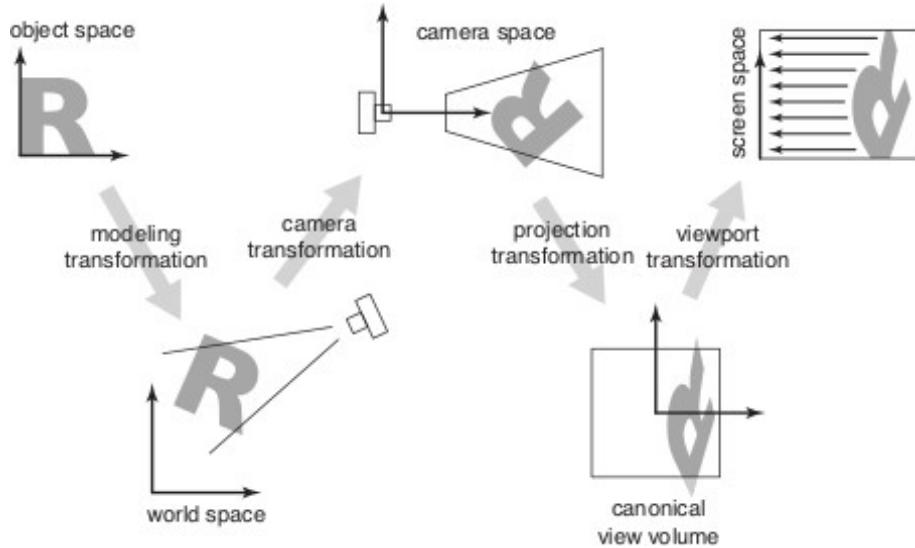


Figura 3.1: Sequência de espaços e transformações, fonte:[Shirley e Marschner, 2009]

### 3.4 Representação da Pose

Em [Ezquerra e Mullick, 1996] é descrito diferentes formas de representar uma pose 3D. A representação de pose que será usada é:

$$\theta = (r_{1x}, r_{1y}, r_{1z}, r_{2x}, r_{2y}, r_{2z}, \text{near}, \text{scale}, t_x, t_y) \quad (3.7)$$

Uma vez que  $r_3 = r_1 \times r_2$ , a operação de escala é igual nas três dimensões para preservar o modelo de distorções, não é utilizada translação no eixo z, uma vez que  $t_z$  é equivalente a operações com escala e variação do termo *near* e pode provocar *clipping* indesejado.

### 3.5 Estimativa de Pose Inicial

A estimativa inicial parte de uma heurística assim pode-se observar as abordagens de [Ozusyal et al., 2009], o qual utiliza *bounding box*. [Sun et al., 2010], utiliza Hough. E [Hutter e Brewer, 2009], tenta equiparar uma elipse 2D a uma elipse 3D projetada. Foram implementadas as seguintes abordagens:

**Cube-faces:** imagine que a malha é um cubo, assim existem 6 visões diferentes tal que a cada vez uma face das 6 é paralela ao plano da câmera. Cada face é testada com a distância *loss*. A com menor distância é escolhida como estimativa inicial. Essa abordagem não é efetiva, principalmente quando faces diferentes acabam sendo visualmente equivalentes devido a forma do objeto.

**Semiautomática:** Aqui é utilizado técnicas como escalar a malha para que o *bounding box* da imagem projetada se equipe com o *bounding box* da fotografia e então a fotografia é translada para se alinhar com a imagem projetada, a translação da fotografia invés da projeção é feita afim de evitar a influência da projeção perspectiva, a distância focal por sua vez deve ser modificada produzir apenas o efeito de deformação e não escalar o objeto nos três eixos, o usuário então prove refinamentos de translação, escala e rotação e quando satisfeita encerra o método e um arquivo de pose é gerado.

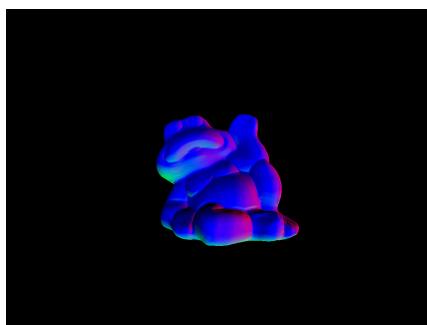
## 3.6 Passos da execução

A seguir será descrito brevemente os passos para a execução de um experimento.

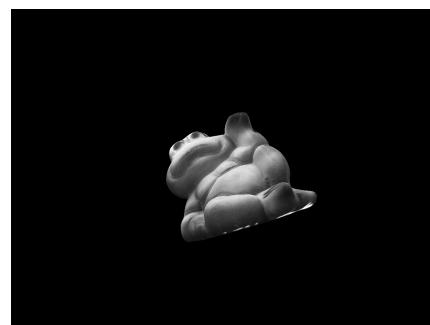
1. Pré-processar a imagem a ser utilizada.
2. Pré-processar a malha a ser utilizada.
3. Obter uma estimativa inicial de pose.
4. Carregar as malhas do disco para a memória.
5. Carregar as texturas do disco para memória.
6. Definir as matrizes de transformação, afim de se obter uma pose  $\theta$ .
7. Execução dos *Shaders*.
8. Cópia do *buffer* de renderização resultante  $M_\theta$  da memória da GPU para a cache e RAM.
9. Calcular a distância *loss* entre  $M_\theta$  e a imagem  $F$ .
10. Inicializar os vetores necessários para o Downhill-Simplex, com a chamada de heurísticas.
11. Iterar os passos do Downhill-Simplex, repetindo os passos de 6 a 9 com os novos parâmetros.
12. Gerar nova textura

## 3.7 Cálculo da distância

A função de perda invariante a luz descrita na seção 2.6, originalmente foi proposta para modelos de carros onde o brilho não é uniforme por toda a superfície. Na implementação do protótipo ela foi alterada no sentido que  $m$  o número de atributos de  $M$  que era 4 foi alterado para 3, o coeficiente de brilho da superfície foi removido, consequentemente assume-se aqui que a superfície tem o coeficiente de brilho igual em todos os pontos, mantendo apenas os valores da normal a superfície, a consequência dessa simplificação é o afastamento do modelo com a realidade, fazendo com a distância *loss* nunca chegue em 0 para imagens reais. A equação 2.11 descreve  $D(F, M_\theta)$ . Dada a malha "frog", com 155333 vértices e 310662 faces, e as imagens de  $M_\theta$  e  $F$  ambas com resolução 1224x918 pixels.



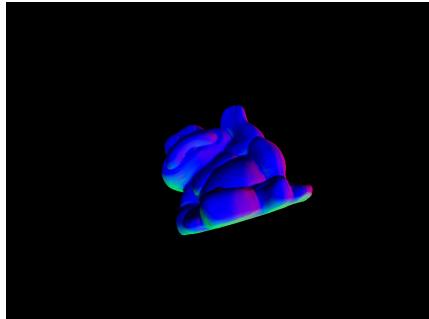
(a) Resultado do *render* com a pose inicial  $M_\theta$



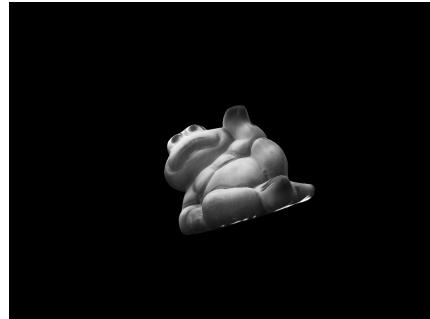
(b) Imagem de interesse em escala de cinza  $F$

Figura 3.2: Imagens de entrada para a distância *loss*

As imagens da figura 3.2 são as entradas da função, onde (a) da figura 3.2 é a mesma pose da figura (a) 4.20, onde  $\theta$  é descrito coluna "Inicial"da tabela 4.20, onde  $D(F, M_\theta) = 0.58641$



(a) Resultado do *render* com a pose final  $M'_{\theta'}$



(b) Imagem de interesse em escala de cinza  $F'$

Figura 3.3: Imagens de entrada para a distância *loss*

As imagens da figura 3.9 são as entradas da função, onde (a) da figura 3.9 é a mesma pose da figura (b) 4.20,onde  $\theta'$  é descrito coluna "Final"da tabela 4.20, onde  $D(F', M'_{\theta'}) = 0.161554$

## 3.8 Obtendo a nova textura

Suponha que uma pose  $\theta$  foi escolhida e a distância  $loss$  entre  $M_\theta$ , a matriz com o resultado do render, e  $L$ , a matriz com imagem a fotografia ou quadro de vídeo, representa o mínimo global. Então as matrizes de transformação já estão definidas corretamente, em seguida ocorre a execução dos *Shaders* mais uma vez. Seja  $\iota = (V \cdot C)^{-1}$ ,  $u_{\text{NormalMatrix}} = \iota^\top$ , entretanto  $u_{\text{NormalMatrix}}$  é reduzida a uma matriz  $3 \times 3$ , descartando a quarta linha e quarta coluna.

*Simple Vertex Shader*

```

1 layout (location = 0) in vec3 i_position;
2 layout (location = 1) in vec3 i_normal;
3 layout (location = 2) in vec2 i_texCoords;
4
5 // data for fragment shader
6 out vec3 normalInterp;
7 out vec2 o_TexCoords;
8
9 // Transformation Matrices
10 uniform mat3 u_NormalMatrix;
11 uniform mat4 u_PVM;
12
13 void main()
14 {
15     o_TexCoords = i_texCoords;
16     normalInterp = u_NormalMatrix * i_normal;
17     gl_Position = u_PVM * vec4(i_position, 1.0f);
18 }
```

*Simple Fragment Shader*

```

1 in vec2 o_TexCoords;
2 in vec3 normalInterp;
3
4 layout(location = 0) out vec3 resultingColor;
5 layout(location = 1) out vec3 newUVS;
6
7 void main()
8 {
9     vec3 normal = normalize(normalInterp);
10    resultingColor = normal;
11    newUVS = vec3(o_TexCoords.y, o_TexCoords.x, 0.0);
12 }
```

Assim a saída do *Simple Fragment Shader* são duas matrizes.  $M_\theta$ , onde cada índice  $p$  é um vetor (r,g,b), tendo um 1 byte por canal. E a matriz  $K$  onde cada índice  $p'$  é um vetor de *floats* (v,u,0). Seja B a nova textura a ser gerada, onde cada índice  $p''$  é um vetor (r,g,b,a), tendo um 1 byte por canal, tal que  $h$  é altura e  $w$  é largura de B, e  $p''$  é inicializado com o valor (0,0,0,0) em todos os pontos.

$$B = \begin{bmatrix} p''_{00} & \cdots & p''_{0w} \\ \vdots & \ddots & \vdots \\ p''_{h0} & \cdots & p''_{hw} \end{bmatrix} \quad (3.8)$$

O pseudocódigo a seguir descreve como é obtida a nova textura.

```

1: for  $i \leftarrow 0$  until  $h$  do
2:   for  $j \leftarrow 0$  until  $w$  do
3:      $v' \leftarrow \lfloor p'_{ij0} \cdot h \rfloor$ 
4:      $u' \leftarrow \lfloor p'_{ij1} \cdot w \rfloor$ 
5:      $p''_{v'u'} \leftarrow L_{ij}$ 
6:     canal alpha de  $p''_{v'u'} \leftarrow 255$ 
7:   end for
8: end for
```

### 3.9 Ilustração do processo

A título de ilustrar qual a influência de uma pose na criação da nova textura. É exibido a seguir a texturas e suas renderizações.

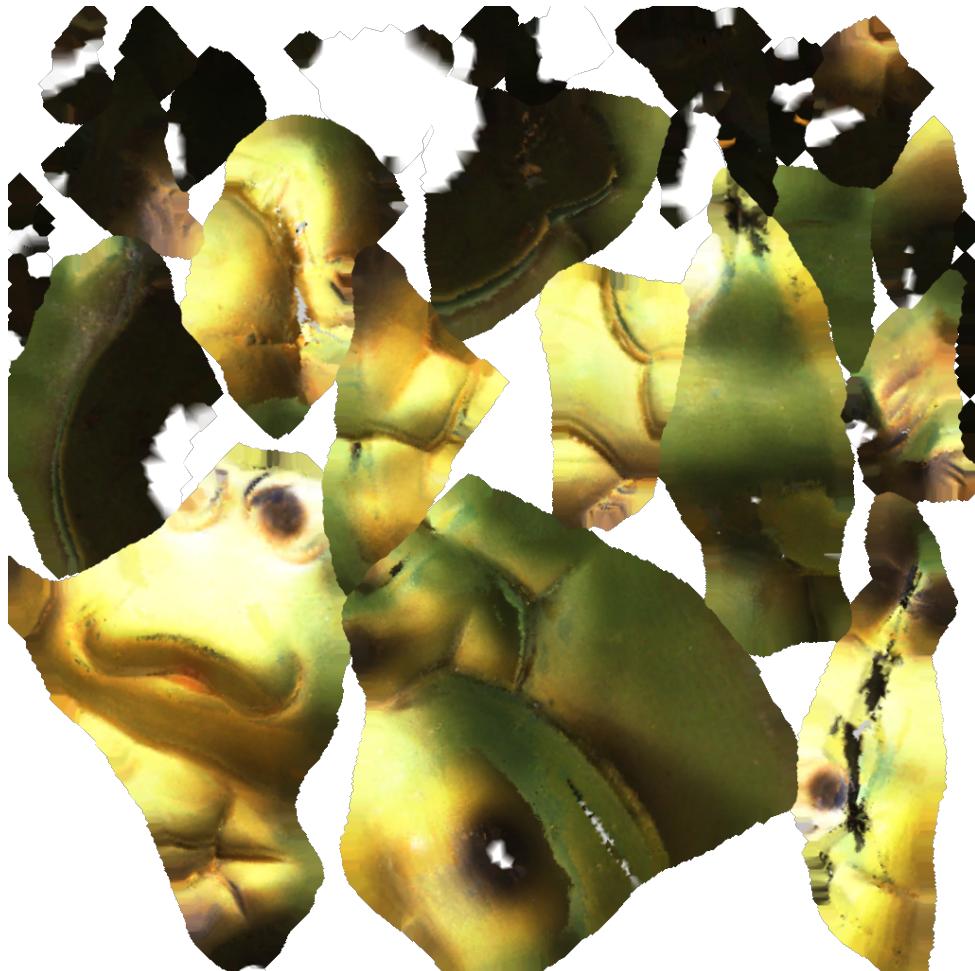


Figura 3.4: Textura original, obtida na conversão da cor de vértice para o mapeamento LSCM com 45 graus



(a) Render sem textura só sombreamento



(b) Render apenas com textura



(c) Render com a textura e sombreamento

Figura 3.5: Exemplo de renderização resultante utilizando a textura exibida na figura 3.4

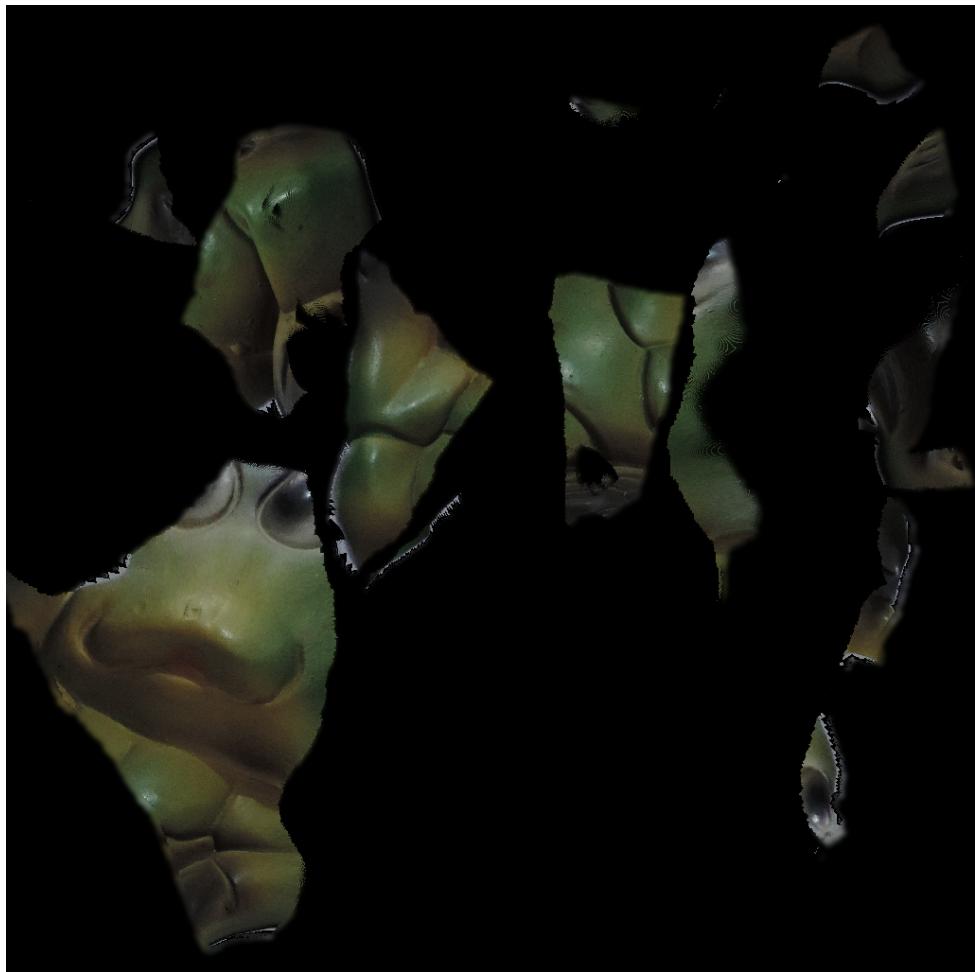


Figura 3.6: Textura gerada pela pose  $\theta$ , descrita na coluna "inicial" da tabela 4.6



(a) Render apenas com textura



(b) Render com a textura e sombreamento

Figura 3.7: Exemplo de renderização resultante utilizando a textura exibida na figura 3.6

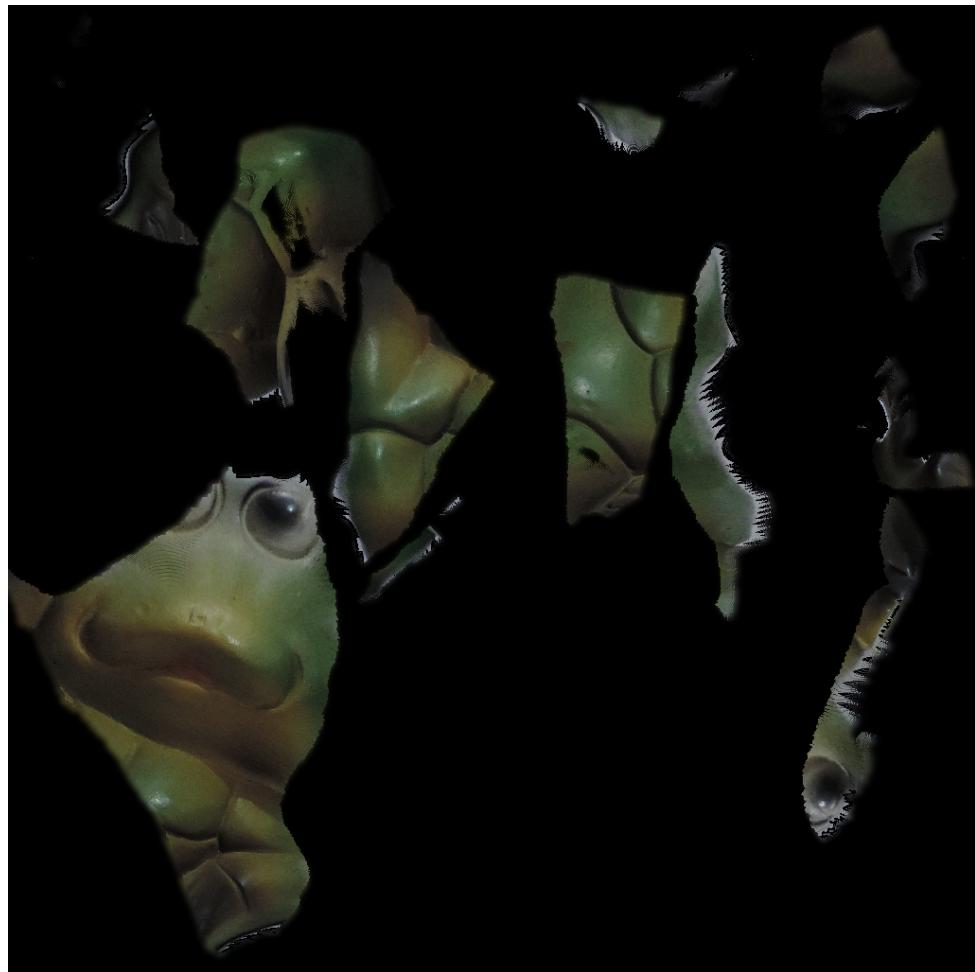


Figura 3.8: Textura gerada pela pose  $\theta'$ , descrita na coluna "final" da tabela 4.6



(a) Render apenas com textura



(b) Render com a textura e sombreamento

Figura 3.9: Exemplo de renderização resultante utilizando a textura exibida na figura 3.8

Observe a diferença de textura entre as figuras 3.4, 3.7 e 3.9, sendo tal diferença mais visível na região das mãos, pés e contorno.

## 3.10 Dificuldades e problemas

Aqui será discutido dificuldades e problemas que ocorreram durante a implementação. Boa parte dos problemas foi em relação a representação de pose, no sentido de que a representação deveria ter o menor número possível de graus de liberdade, ser utilizada pelo otimizador *Downhill-Simplex* e os parâmetros otimizados realmente produzam a pose desejada. Assim teoricamente todos os parâmetros deveriam estar numa escala de  $[-1, 1]$  entretanto essa conversão acabou ocasionando diversos problemas na etapa de renderização.

Idealmente o método poderia convergir simultaneamente a rotação, escala, translação e distância focal a cada passo. Na prática foi necessário fixar a translação e escala na estimativa inicial, sendo otimizado apenas rotação e distância focal, isso se deu por causa de um problema de escala dos parâmetros de pose, uma vez que é necessário colocar todos os parâmetros na mesma escala e depois retornar à sua escala original para aplicar as transformações à malha, uma vez que a parametrização do otimizador é de suma importância na convergência, a função de retorno implementada não retornava os valores otimizados a escala apropriada produzindo uma renderização deformada e isso fazia com que o otimizador divergisse invés de convergir. Se de fato possa ser otimizado esses parâmetros simultaneamente a pose final estará mais próxima do mínimo global do que atualmente o protótipo produz.

Definir a distância focal de tal forma em que o objeto apenas se deformasse, mas não sofresse um escalonamento nos três eixos.

Estimativa inicial foi outro grande problema, como não é assumido conhecimento prévio, as heurísticas totalmente automáticas propostas no capítulo 3 acabam guiando o otimizador a

um mínimo local que não fazia sentido visualmente (pose de frente, acabava de costas). Isso foi parcialmente contornado utilizando uma abordagem semiautomática, onde o usuário calibra a olho nu uma estimativa.

O mapeamento UV é discretizado no momento de gerar a nova textura isso causa o problema de que a imagem original necessite ter uma resolução superior a textura gerada, uma solução para tal seria interpolar valores ou ainda utilizar a media ponderada para colorir os pixels vizinhos, entretanto isso provoca a redução do realismo. Então existe um *trade-off* entre resolução e tempo de execução, onde maior resolução garante maior realismo mas significa um aumento no número de operações e consequentemente no tempo de execução, as tabelas 4.5 e 4.7 ilustram o tempo de execução com diferentes resoluções.

Foi escolhido usar transformações no objeto invés da câmera, afim de obter a pose. Uma vez que para rotacionar a câmera seria necessário orbitar o objeto(transladar a câmera) apontando em direção à ele, enquanto para rotacionar o objeto, a operação pode ser feita mudando a base ortonormal do espaço não sendo necessário uma operação adicional de translação.

### 3.11 Conclusão do capítulo

Nesse capítulo foi descrito qual o pré-processamento necessário das imagens digitais, uma visão geral de como o processo de renderização ocorre na API OpenGL, é descrita as matrizes de transformação usadas, como é formulado  $\theta$  afim de ser utilizado no otimizador Downhill-Simplex, como é obtida um estimativa de pose inicial, quais são os passos da execução do protótipo, como é calculada a distância *loss*, como é gerada a textura final quando se tem a pose com distância mínima, é apresentado um caso para ilustrar como textura final gerada influência em um render, e finalmente é discutido alguns dos problemas e dificuldades encontrados durante o processo de implementação.

# Capítulo 4

## Experimentos

Neste capítulo serão descritos os experimentos realizados, os resultados obtidos, e conclusões relativas a cada experimento. Os experimentos aqui realizados buscam ilustrar o comportamento do protótipo em diferentes cenários.

As figuras deste capítulo, contem a sobreposição do objeto 3D projetado com o filtro Laplaciano sobre a fotografia pré-processada e subsequentemente a tabela com os parâmetros que geraram aquela pose.

Duas malhas foram utilizadas para os experimentos: "Frog" o resultado de um escaneamento 3D de uma estatua de jardim decorativa de cerâmica, de tamanho (altura 12,3 cm, largura 14 cm e comprimento 13 cm), tendo a malha 155333 vértices e 310662 faces.

E a versão disponibilizada pelo museu digital do IMAGO[UFPR, 2016] o Profeta Joel. Estantua em Pedra sabão em dois blocos 234 cm. (1800 -1805). Produzida por Antônio Francisco Lisboa, Aleijadinho (1730/38 – 1814). Parte do Santuário de Bom Jesus de Matosinhos, Congonhas, MG. Tendo a malha 100002 vértices e 199998 faces.

A seguir é descrito brevemente o se busca observar com cada experimento, delimitado quais figuras e tabelas pertencem a qual experimento, sendo as conclusões sobre eles apresentadas na seção 4.9.

1. Este experimento busca observar qual a influência da iluminação no cálculo da distância e na convergência do otimizador, quatro situações de iluminação diferentes são observadas, uma fonte de luz situada a direita do objeto, uma fonte de luz situada a esquerda/acima do objeto, uma fonte de luz situada a frente/acima do objeto, e uma situação onde não há uma fonte de luz próxima do objeto. As imagens utilizadas aqui tem resolução de 1224x918.
2. Este experimento busca observar a influência da resolução no cálculo da distância, convergência do otimizador, e tempo de execução. Duas resoluções são testadas aqui, 4896x3672 e 612x459.
3. O experimento busca verificar se repetir todos os passos do protótipo, incluindo inicialização do vetor para a chamada do *downhill-simplex*, com a diferença em que a nova estimativa inicial seria a estimativa final da ultima execução do protótipo, produziria melhores resultados do que simplesmente aumentar o número de iterações do otimizador. As imagens utilizadas aqui tem resolução de 2000x3008.
4. Este é um experimento mais simples onde a estatua do Profeta Joel é vista completamente na figura em outro ponto de vista, ele compõe parte dos testes com fotografias reais. A imagem utilizada tem resolução de 1024x1024.

5. Este experimento em especial compara a imagem renderizada  $M_\theta$  com outra imagem  $M_{\theta'}$  previamente gerada (utilizando como coloração as normais a superfície) e dessaturada para escalas de cinza, utilizando a média dos canais, a diferença entre  $\theta$  e  $\theta'$  é a matriz de rotação sendo os parâmetros mantidos iguais. Este caso de teste busca demonstrar um cenário de equiparo perfeito de poses. As imagens utilizadas aqui tem resolução de 1224x918.
6. Este experimento busca observar a influência do ruído HSV no cálculo da distância e na convergência do otimizador, O teste consiste em comparar uma imagem  $M_\theta$  com a uma imagem sintética  $\Gamma$ , sendo  $\Gamma$  o resultado de *render* com parâmetros conhecidos onde a coloração se da unicamente pela textura (sem sombreamento). Outro detalhe importante é que todos os parâmetros exceto a matriz de rotação são os mesmo de  $\Gamma$ .As imagens utilizadas aqui tem resolução de 1024x1024.
7. Outro experimento de imagem sintética sem sombreamento apenas textura, desta vez com o sapo. A imagem utilizada tem resolução de 1224x918.
8. Este experimento consiste em um teste com vídeo, o vídeo foi filmado em 640x480 em 120 FPS, onde a câmera estava estacionaria e o objeto foi rotacionado até ocorrer um volta completa. A base rotatória em que o sapo foi colocado, não era ideal, fazendo com o sapo rotaciona-se em mais de um eixo e saísse de sua posição inicial, 49 frames foram selecionados e pré-processados, as figuras 4.15 até 4.63, onde uma estimativa inicial foi usada para o *frame* 0 e os *frames* subsequentes usam como entrada a pose resultado do *frame* anterior.

## 4.1 Experimento 1

Este experimento busca observar qual a influência da iluminação no cálculo da distância e na convergência do otimizador, quatro situações de iluminação diferentes são observadas, uma fonte de luz situada a direita do objeto, uma fonte de luz situada a esquerda/acima do objeto, uma fonte de luz situada a frente/acima do objeto, e uma situação onde não há uma fonte de luz próxima do objeto. As imagens utilizadas aqui tem resolução de 1224x918.

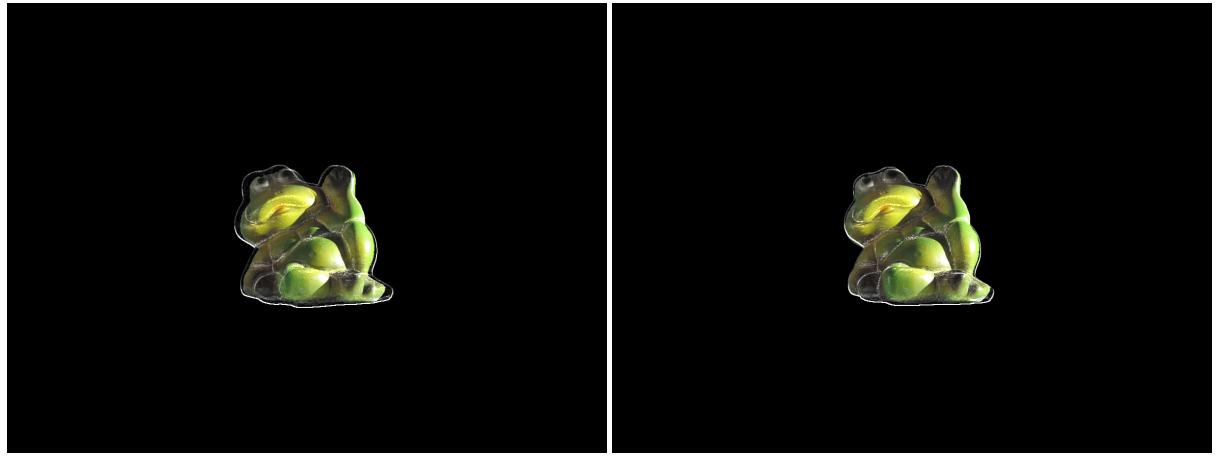


Figura 4.1: Imagem 1 do experimento 1, convergência de (a) para (b) afim de demonstrar influência da luz ambiente

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.989092 & 0.00745012 & 0.147109 & 0 \\ 0 & -0.999404 & -0.0345071 & 0 \\ -0.146764 & -0.0341307 & 0.988503 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 31.0937 & 0 & 0 & 0 \\ 0 & 31.0937 & 0 & 0 \\ 0 & 0 & 31.0937 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 31.0937 & 0 & 0 & 0 \\ 0 & 31.0937 & 0 & 0 \\ 0 & 0 & 31.0937 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 11$	$t_x = 11$
$t_y = 15$	$t_y = 15$

Tabela 4.1: Parâmetros iniciais e finais da figura 4.1

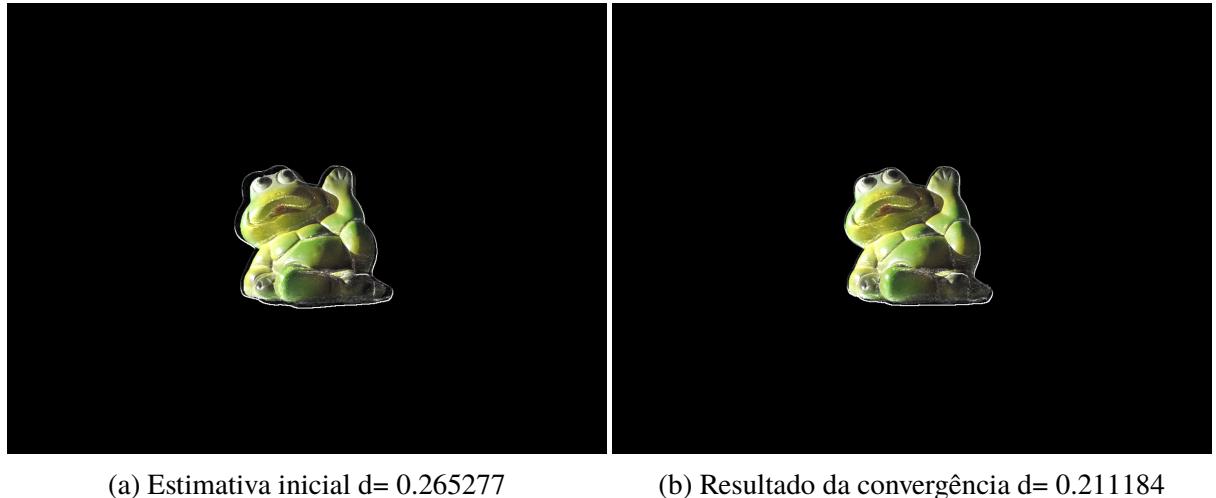
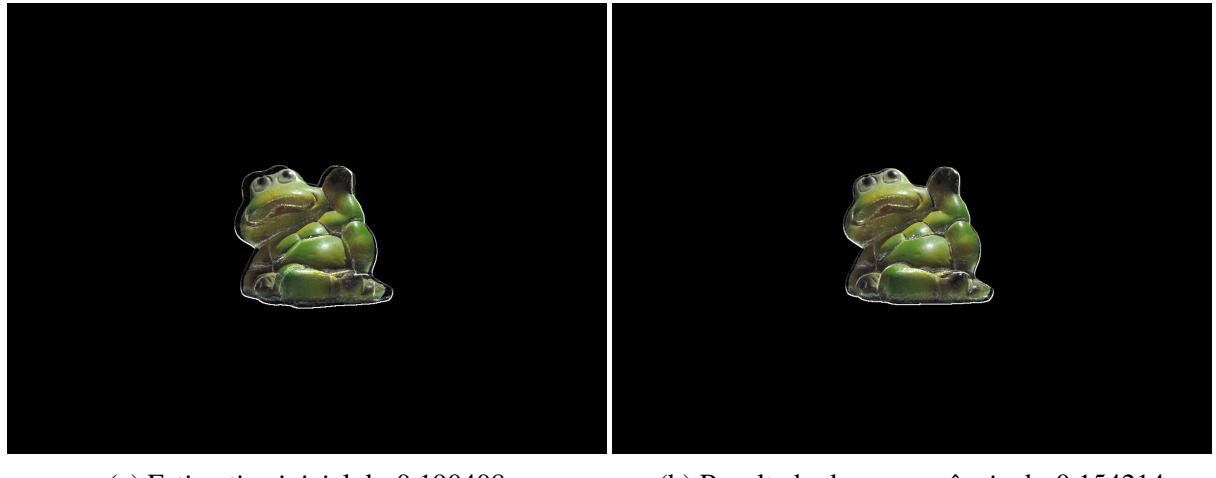


Figura 4.2: Imagem 2 do experimento 1, convergência de (a) para (b) afim de demonstrar influência da luz ambiente

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.988937 & 0.0146638 & 0.147613 & 0 \\ 0.0174431 & -0.998936 & -0.0426889 & 0 \\ -0.14683 & -0.0447915 & 0.98814 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 31.0937 & 0 & 0 & 0 \\ 0 & 31.0937 & 0 & 0 \\ 0 & 0 & 31.0937 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 31.0937 & 0 & 0 & 0 \\ 0 & 31.0937 & 0 & 0 \\ 0 & 0 & 31.0937 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 11$	$t_x = 11$
$t_y = 15$	$t_y = 15$

Tabela 4.2: Parâmetros iniciais e finais da figura 4.2



(a) Estimativa inicial d= 0.190408

(b) Resultado da convergência d= 0.154214

Figura 4.3: Imagem 3 do experimento 1, convergência de (a) para (b) afim de demonstrar influência da luz ambiente

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.993024 & 0.0125282 & 0.117242 & 0 \\ 0.0263238 & -0.997941 & -0.058484 & 0 \\ -0.116268 & -0.0611622 & 0.99131 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 31.0937 & 0 & 0 & 0 \\ 0 & 31.0937 & 0 & 0 \\ 0 & 0 & 31.0937 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 31.0937 & 0 & 0 & 0 \\ 0 & 31.0937 & 0 & 0 \\ 0 & 0 & 31.0937 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 11$	$t_x = 11$
$t_y = 15$	$t_y = 15$

Tabela 4.3: Parâmetros iniciais e finais da figura 4.3

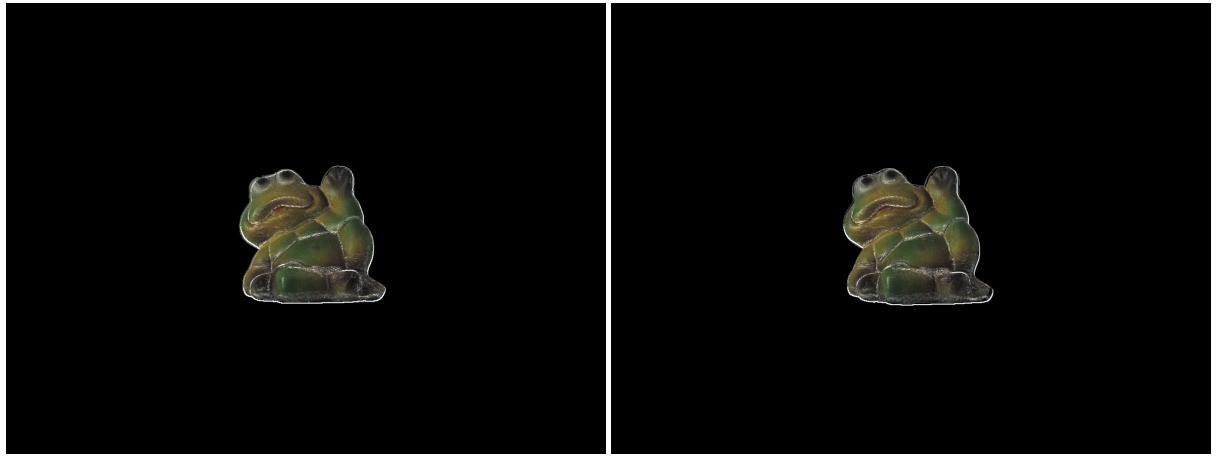
(a) Estimativa inicial  $d = 0.158979$ (b) Resultado da convergência  $d = 0.143537$ 

Figura 4.4: Imagem 4 do experimento 1, convergência de (a) para (b) afim de demonstrar influência da luz ambiente

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.975917 & -0.0174949 & 0.217441 & 0 \\ 0 & -0.996779 & -0.0801989 & 0 \\ -0.218143 & -0.0782674 & 0.972773 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.988061 & 0.00130169 & 0.154056 & 0 \\ -0.00555026 & -0.998448 & -0.0554225 & 0 \\ -0.153744 & -0.0539057 & 0.98652 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 31.0937 & 0 & 0 & 0 \\ 0 & 31.0937 & 0 & 0 \\ 0 & 0 & 31.0937 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 31.0937 & 0 & 0 & 0 \\ 0 & 31.0937 & 0 & 0 \\ 0 & 0 & 31.0937 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 11$	$t_x = 11$
$t_y = 15$	$t_y = 15$

Tabela 4.4: Parâmetros iniciais e finais da figura 4.4

## 4.2 Experimento 2

Este experimento busca observar a influência da resolução no cálculo da distância, convergência do otimizador, e tempo de execução. Duas resoluções são testadas aqui, 4896x3672 e 612x459.

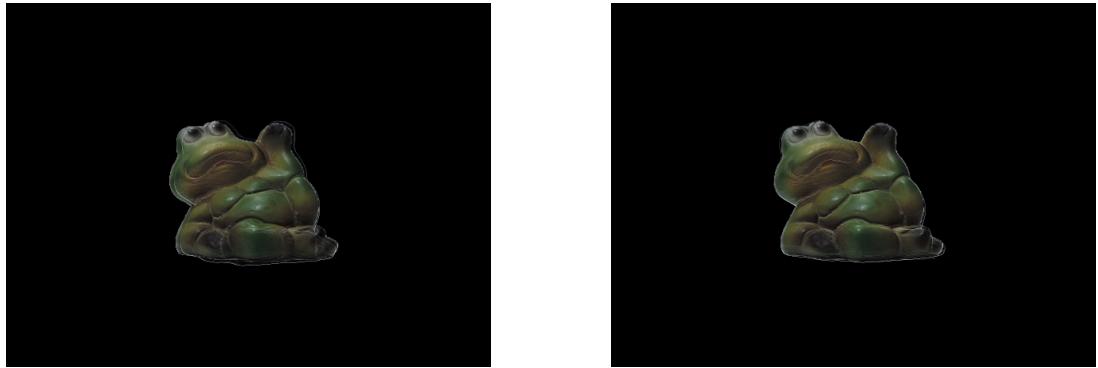
(a) Estimativa inicial  $d= 0.173699$ (b) Resultado da convergência  $d= 0.104499$ 

Figura 4.5: Imagem 1 do experimento 2, convergência de (a) para (b) afim de demonstrar a influência da resolução da imagem, resolução testada 4896x3672

real	1m27.419s
user	1m15.536s
sys	0m11.182s

Tabela 4.5: Tempo de execução do protótipo para realizar a execução completa da instancia 1 do experimento 2

Inicial	Final
$near = 1.6$	$near = 1.6$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.978466 & 0.027769 & -0.204531 & 0 \\ 0 & -0.998696 & -0.0510578 & 0 \\ 0.205682 & -0.0499584 & 0.97719 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 29.4707 & 0 & 0 & 0 \\ 0 & 29.4707 & 0 & 0 \\ 0 & 0 & 29.4707 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 29.4707 & 0 & 0 & 0 \\ 0 & 29.4707 & 0 & 0 \\ 0 & 0 & 29.4707 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 52$	$t_x = 52$
$t_y = 64$	$t_y = 64$

Tabela 4.6: Parâmetros iniciais e finais da figura 4.5



(a) Estimativa inicial d= 0.194672



(b) Resultado da convergência d= 0.128776

Figura 4.6: Imagem 2 do experimento 2, convergência de (a) para (b) afim de demonstrar a influência da resolução da imagem, resolução testada 612x459

real	0m5.455s
user	0m4.045s
sys	0m0.866s

Tabela 4.7: Tempo de execução do protótipo para realizar a execução completa da instancia 2 do experimento 2

Inicial	Final
$near = 1.6$	$near = 1.6$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.971172 & 0.0179822 & -0.2377 & 0 \\ 0 & -0.999472 & -0.032502 & 0 \\ 0.238159 & -0.031565 & 0.970659 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 29.4707 & 0 & 0 & 0 \\ 0 & 29.4707 & 0 & 0 \\ 0 & 0 & 29.4707 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 29.4707 & 0 & 0 & 0 \\ 0 & 29.4707 & 0 & 0 \\ 0 & 0 & 29.4707 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 6$	$t_x = 6$
$t_y = 8$	$t_y = 8$

Tabela 4.8: Parâmetros iniciais e finais da figura 4.6

### 4.3 Experimento 3

O experimento busca verificar se repetir todos os passos do protótipo, incluindo inicialização do vetor para a chamada do downhill-simplex , com a diferença em que a nova

estimativa inicial seria a estimativa final da ultima execução do protótipo, produziria melhores resultados do que simplesmente aumentar o número de iterações do otimizador. As imagens utilizadas aqui tem resolução de 2000x3008.



(a) Estimativa inicial d= 0.582753

(b) Resultado da convergência d= 0.409007

Figura 4.7: Imagem 1 do experimento 3, convergência de (a) para (b) usando estimativa inicial

Inicial	Final
$near = 4$	$near = 4$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.785705 & 0.436165 & -0.438665 & 0 \\ 0 & -0.899643 & -0.436627 & 0 \\ 0.585083 & -0.34306 & 0.706854 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 75.4338 & 0 & 0 & 0 \\ 0 & 75.4338 & 0 & 0 \\ 0 & 0 & 75.4338 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 75.4338 & 0 & 0 & 0 \\ 0 & 75.4338 & 0 & 0 \\ 0 & 0 & 75.4338 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = -24$	$t_x = -24$
$t_y = 64$	$t_y = 64$

Tabela 4.9: Parâmetros iniciais e finais da figura 4.7



(a) Estimativa inicial d= 0.409011

(b) Resultado da convergência d= 0.383769

Figura 4.8: Imagem 2 do experimento 3, convergência de (a) para (b), usando parâmetros finais da tabela 4.9 como estimativa inicial

Inicial	Final
$near = 4$	$near = 4$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.785705 & 0.436165 & -0.438665 & 0 \\ 0 & -0.899643 & -0.436627 & 0 \\ 0.585083 & -0.34306 & 0.706854 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.848527 & 0.407149 & -0.337981 & 0 \\ -0.0475085 & -0.897358 & -0.438738 & 0 \\ 0.481922 & -0.388338 & 0.74209 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 75.4338 & 0 & 0 & 0 \\ 0 & 75.4338 & 0 & 0 \\ 0 & 0 & 75.4338 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 75.4338 & 0 & 0 & 0 \\ 0 & 75.4338 & 0 & 0 \\ 0 & 0 & 75.4338 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = -24$	$t_x = -24$
$t_y = 64$	$t_y = 64$

Tabela 4.10: Parâmetros iniciais e finais da figura 4.8

(a) Estimativa inicial  $d = 0.383779$ (b) Resultado da convergência  $d = 0.383196$ 

Figura 4.9: Imagem 3 do experimento 3, convergência de (a) para (b), usando parâmetros finais da tabela 4.10 como estimativa inicial

Inicial	Final
$near = 4$	$near = 4$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.848527 & 0.407149 & -0.337981 & 0 \\ -0.0475085 & -0.897358 & -0.438738 & 0 \\ 0.481922 & -0.388338 & 0.74209 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.845629 & 0.410006 & -0.341771 & 0 \\ -0.0464927 & -0.892834 & -0.44798 & 0 \\ 0.488819 & -0.394714 & 0.735944 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 75.4338 & 0 & 0 & 0 \\ 0 & 75.4338 & 0 & 0 \\ 0 & 0 & 75.4338 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 75.4338 & 0 & 0 & 0 \\ 0 & 75.4338 & 0 & 0 \\ 0 & 0 & 75.4338 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = -24$	$t_x = -24$
$t_y = 64$	$t_y = 64$

Tabela 4.11: Parâmetros iniciais e finais da figura 4.9

(a) Estimativa inicial  $d = 0.383201$ (b) Resultado da convergência  $d = 0.383033$ 

Figura 4.10: Imagem 4 do experimento 3, convergência de (a) para (b), usando parâmetros finais da tabela 4.11 como estimativa inicial

Inicial	Final
$near = 4$	$near = 4$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.845629 & 0.410006 & -0.341771 & 0 \\ -0.0464927 & -0.892834 & -0.44798 & 0 \\ 0.488819 & -0.394714 & 0.735944 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.845646 & 0.408755 & -0.343223 & 0 \\ -0.0473551 & -0.892777 & -0.448002 & 0 \\ 0.489545 & -0.395105 & 0.735617 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 75.4338 & 0 & 0 & 0 \\ 0 & 75.4338 & 0 & 0 \\ 0 & 0 & 75.4338 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 75.4338 & 0 & 0 & 0 \\ 0 & 75.4338 & 0 & 0 \\ 0 & 0 & 75.4338 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = -24$	$t_x = -24$
$t_y = 64$	$t_y = 64$

Tabela 4.12: Parâmetros iniciais e finais da figura 4.10

## 4.4 Experimento 4

Este é um experimento mais simples onde a estatua do Profeta Joel é vista completamente na figura em outro ponto de vista, ele compõe parte dos testes com fotografias reais. A imagem utilizada tem resolução de 1024x1024.

(a) Estimativa inicial  $d = 0.276848$ (b) Resultado da convergência  $d = 0.249275$ 

Figura 4.11: Imagem 1 do experimento 4, convergência de (a) para (b). A imagem tem sua escala de cinza invertida, mantendo o plano de fundo com valor 0, quando usada no calculo da distancia

Inicial	Final
$near = 2.1$	$near = 2.1$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.999294 & 0.00395224 & 0.0373751 & 0 \\ 0 & -0.99602 & -0.0891257 & 0 \\ -0.0368741 & -0.0890627 & 0.995317 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 75.9897 & 0 & 0 & 0 \\ 0 & 75.9897 & 0 & 0 \\ 0 & 0 & 75.9897 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 75.9897 & 0 & 0 & 0 \\ 0 & 75.9897 & 0 & 0 \\ 0 & 0 & 75.9897 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 1$	$t_x = 1$
$t_y = 15$	$t_y = 15$

Tabela 4.13: Parâmetros iniciais e finais da figura 4.11

(a) Estimativa inicial  $d = 0.278929$ (b) Resultado da convergência  $d = 0.248914$ 

Figura 4.12: Imagem 2 do experimento 4, convergência de (a) para (b).

Inicial	Final
$near = 2.1$	$near = 2.1$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.994437 & -0.00209578 & 0.105311 & 0 \\ 0 & -0.998923 & -0.0463908 & 0 \\ -0.105295 & -0.0461327 & 0.993367 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 75.9897 & 0 & 0 & 0 \\ 0 & 75.9897 & 0 & 0 \\ 0 & 0 & 75.9897 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 75.9897 & 0 & 0 & 0 \\ 0 & 75.9897 & 0 & 0 \\ 0 & 0 & 75.9897 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 1$	$t_x = 1$
$t_y = 15$	$t_y = 15$

Tabela 4.14: Parâmetros iniciais e finais da figura 4.12

## 4.5 Experimento 5

Este experimento em especial compara a imagem renderizada  $M_\theta$  com outra imagem  $M_{\theta'}$  previamente gerada (utilizando como coloração as normais a superfície) e dessaturada para escalas de cinza, utilizando a média dos canais, a diferença entre  $\theta$  e  $\theta'$  é a matriz de rotação sendo

os parâmetros mantidos iguais. Este caso de teste busca demonstrar um cenário de equiparo perfeito de poses. As imagens utilizadas aqui tem resolução de 1224x918.

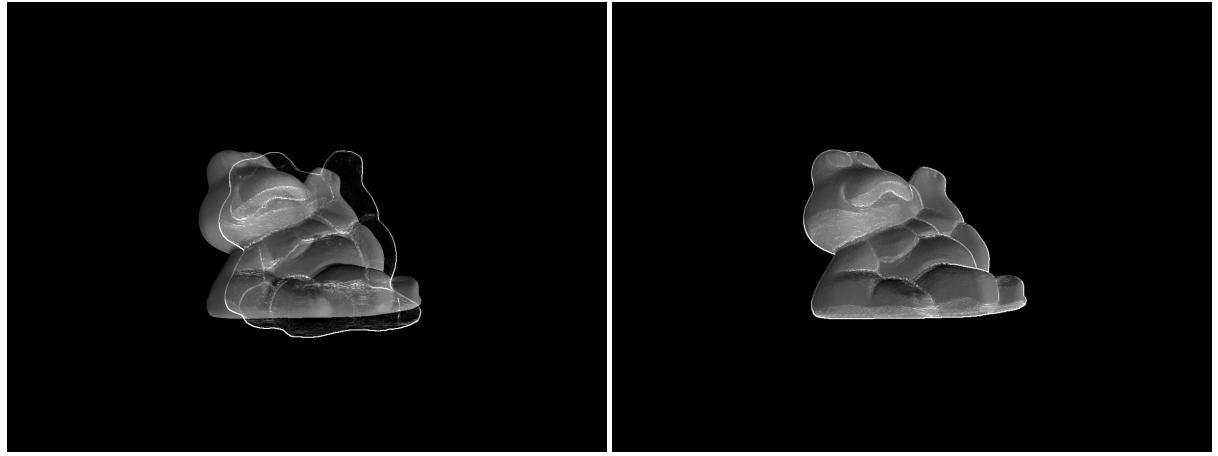
(a) Estimativa inicial  $d= 0.459172$ (b) Resultado da convergência  $d= 0.00928405$ 

Figura 4.13: Imagem 1 do experimento 5, convergência de (a) para (b), afim de demonstrar a convergência para imagens sintéticas utilizando escalas de cinza. Dessaturação com média dos canais

Inicial	Final
$near = 1$	$near = 1$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.76855 & 0.146171 & -0.622868 & 0 \\ 0.00121965 & -0.973331 & -0.229403 & 0 \\ 0.639789 & -0.175548 & 0.748232 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 46.2286 & 0 & 0 & 0 \\ 0 & 46.2286 & 0 & 0 \\ 0 & 0 & 46.2286 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 46.2286 & 0 & 0 & 0 \\ 0 & 46.2286 & 0 & 0 \\ 0 & 0 & 46.2286 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 0$	$t_x = 0$
$t_y = 0$	$t_y = 0$

Tabela 4.15: Parâmetros iniciais e finais da figura 4.13

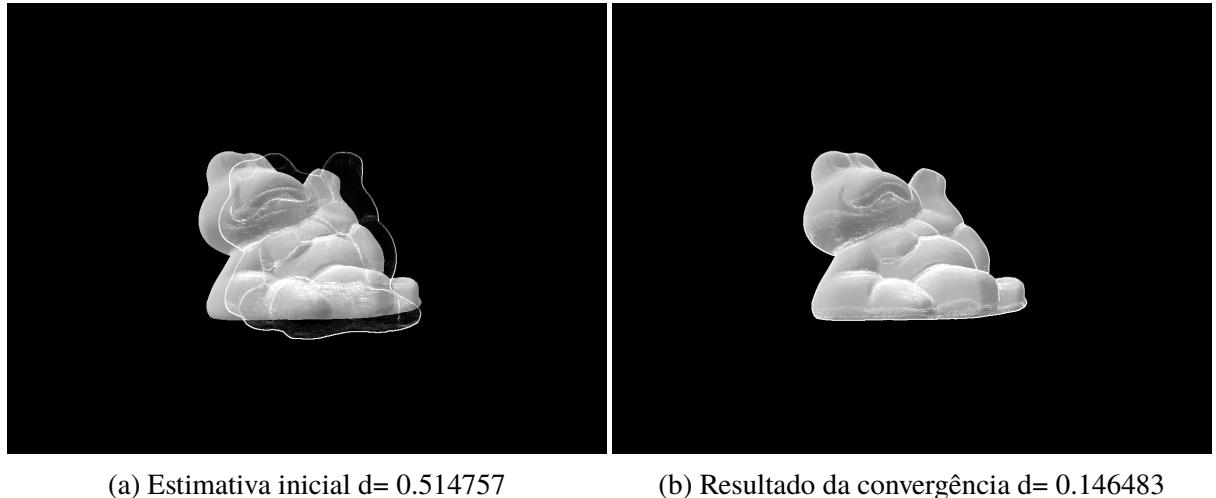


Figura 4.14: Imagem 2 do experimento 5, convergência de (a) para (b), afim de demonstrar a convergência para imagens sintéticas utilizando escalas de cinza. Dessaturação com média dos canais

Inicial	Final
$near = 1$	$near = 1$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.758225 & 0.13765 & -0.637296 & 0 \\ 0.00684817 & -0.974141 & -0.225837 & 0 \\ 0.651903 & -0.166871 & 0.739561 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 46.2286 & 0 & 0 & 0 \\ 0 & 46.2286 & 0 & 0 \\ 0 & 0 & 46.2286 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 46.2286 & 0 & 0 & 0 \\ 0 & 46.2286 & 0 & 0 \\ 0 & 0 & 46.2286 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 0$	$t_x = 0$
$t_y = 0$	$t_y = 0$

Tabela 4.16: Parâmetros iniciais e finais da figura 4.15

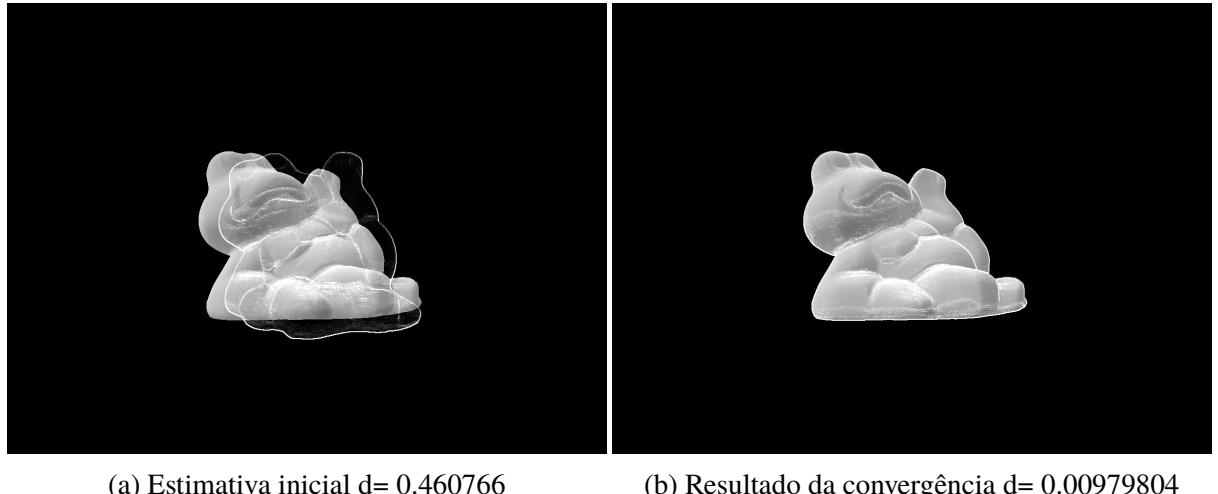
(a) Estimativa inicial  $d = 0.460766$       (b) Resultado da convergência  $d = 0.00979804$ 

Figura 4.15: Imagem 3 do experimento 5, convergência de (a) para (b), afim de demonstrar a convergência para imagens sintéticas utilizando escalas de cinza. Dessaturação com média dos canais. A imagem tem sua escala de cinza invertida, mantendo o plano de fundo com valor 0, quando usada no calculo da distancia se aproximando da figura 4.13

Inicial	Final
$near = 1$	$near = 1$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.769465 & 0.147733 & -0.621369 & 0 \\ -0.00280474 & -0.973229 & -0.229822 & 0 \\ 0.638686 & -0.178583 & 0.748451 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 46.2286 & 0 & 0 & 0 \\ 0 & 46.2286 & 0 & 0 \\ 0 & 0 & 46.2286 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 46.2286 & 0 & 0 & 0 \\ 0 & 46.2286 & 0 & 0 \\ 0 & 0 & 46.2286 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 0$	$t_x = 0$
$t_y = 0$	$t_y = 0$

Tabela 4.17: Parâmetros iniciais e finais da figura 4.15

## 4.6 Experimento 6

Este experimento busca observar a influência do ruído HSV no cálculo da distância e na convergência do otimizador. O teste consiste em comparar uma imagem  $M_\theta$  com a uma imagem sintética  $\Gamma$ , sendo  $\Gamma$  o resultado de *render* com parâmetros conhecidos onde a coloração se da unicamente pela textura (sem sombreamento). Outro detalhe importante é que todos

os parâmetros exceto a matriz de rotação são os mesmo de  $\Gamma$ . As imagens utilizadas aqui tem resolução de 1024x1024.



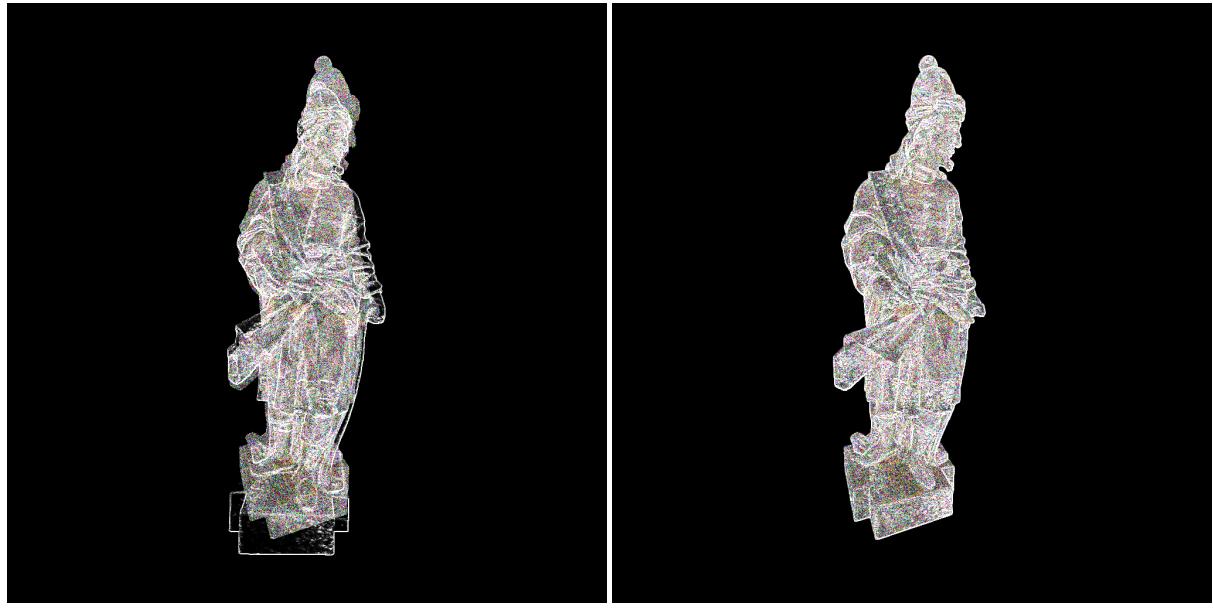
(a) Estimativa inicial  $d= 0.357445$

(b) Resultado da convergência  $d= 0.139384$

Figura 4.16: Imagem 1 do experimento 6, convergência de (a) para (b), afim de demonstrar a influência de ruído HSV para imagens sintéticas. Teste sem ruído

Inicial	Final
$near = 1$	$near = 1$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.936216 & -0.0717355 & -0.344026 & 0 \\ 0 & -0.978857 & 0.204545 & 0 \\ 0.351425 & 0.191498 & 0.916422 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 150.81 & 0 & 0 & 0 \\ 0 & 150.81 & 0 & 0 \\ 0 & 0 & 150.81 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 150.81 & 0 & 0 & 0 \\ 0 & 150.81 & 0 & 0 \\ 0 & 0 & 150.81 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 0$	$t_x = 0$
$t_y = 0$	$t_y = 0$

Tabela 4.18: Parâmetros iniciais e finais da figura 4.16



(a) Estimativa inicial d= 0.455598

(b) Resultado da convergência d= 0.274084

Figura 4.17: Imagem 2 do experimento 6, convergência de (a) para (b), afim de demonstrar a influência de ruído HSV para imagens sintéticas. Teste com ruído

Inicial	Final
$near = 1$	$near = 1$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.936226 & -0.0717375 & -0.343999 & 0 \\ 0 & -0.978864 & 0.204513 & 0 \\ 0.351399 & 0.19147 & 0.916438 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 150.81 & 0 & 0 & 0 \\ 0 & 150.81 & 0 & 0 \\ 0 & 0 & 150.81 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 150.81 & 0 & 0 & 0 \\ 0 & 150.81 & 0 & 0 \\ 0 & 0 & 150.81 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 0$	$t_x = 0$
$t_y = 0$	$t_y = 0$

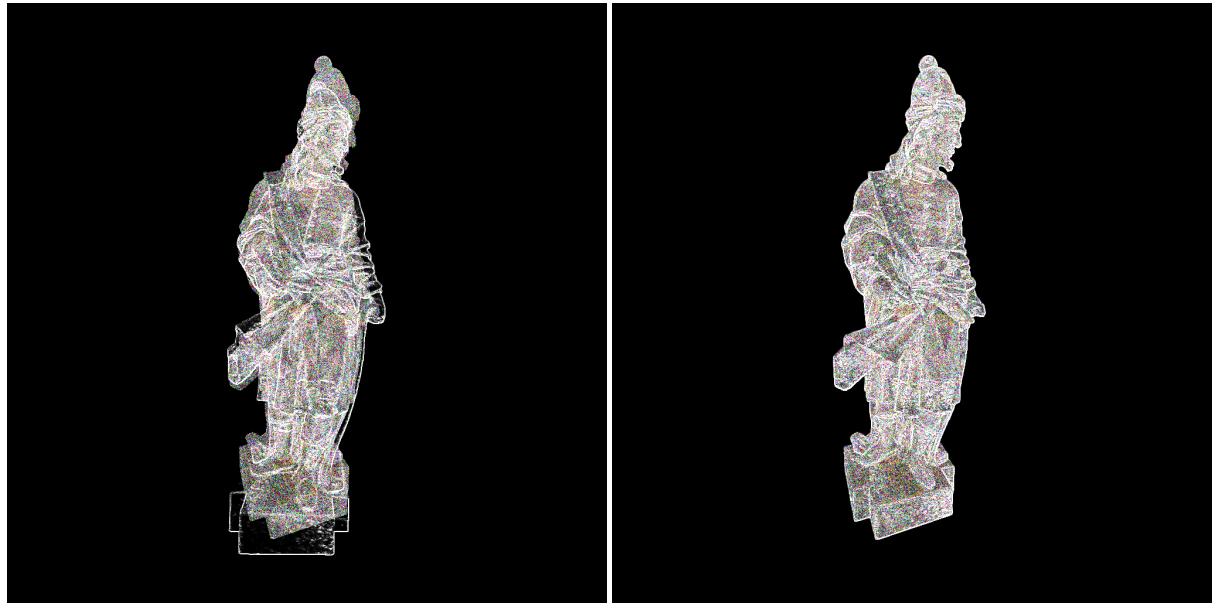
Tabela 4.19: Parâmetros iniciais e finais da figura 4.17

(a) Estimativa inicial  $d = 0.40253$ (b) Resultado da convergência  $d = 0.222539$ 

Figura 4.18: Imagem 3 do experimento 6, convergência de (a) para (b), afim de demonstrar a influência de ruído HSV para imagens sintéticas. Teste sem ruído. A imagem tem sua escala de cinza invertida, mantendo o plano de fundo com valor 0, quando usada no cálculo da distância.

Inicial	Final
$near = 1$	$near = 1$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.93618 & -0.071691 & -0.344134 & 0 \\ 0 & -0.97888 & 0.204435 & 0 \\ 0.351522 & 0.191388 & 0.916408 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 150.81 & 0 & 0 & 0 \\ 0 & 150.81 & 0 & 0 \\ 0 & 0 & 150.81 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 150.81 & 0 & 0 & 0 \\ 0 & 150.81 & 0 & 0 \\ 0 & 0 & 150.81 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x =$	$t_x = 0$
$t_y =$	$t_y = 0$

Tabela 4.20: Parâmetros iniciais e finais da figura 4.18



(a) Estimativa inicial d= 0.48996

(b) Resultado da convergência d= 0.33427

Figura 4.19: Imagem 4 do experimento 6, convergência de (a) para (b), afim de demonstrar a influência de ruído HSV para imagens sintéticas. Teste com ruído. A imagem tem sua escala de cinza invertida, mantendo o plano de fundo com valor 0, quando usada no calculo da distancia.

Inicial	Final
$near = 1$	$near = 1$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.936849 & -0.0670308 & -0.343252 & 0 \\ 0 & -0.978946 & 0.20412 & 0 \\ 0.349707 & 0.191229 & 0.917124 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 150.81 & 0 & 0 & 0 \\ 0 & 150.81 & 0 & 0 \\ 0 & 0 & 150.81 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 150.81 & 0 & 0 & 0 \\ 0 & 150.81 & 0 & 0 \\ 0 & 0 & 150.81 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x =$	$t_x = 0$
$t_y =$	$t_y = 0$

Tabela 4.21: Parâmetros iniciais e finais da figura 4.19

## 4.7 Experimento 7

Outro experimento de imagem sintética sem sombreamento apenas textura, desta vez com o sapo. A imagem utilizada tem resolução de 1224x918.

(a) Estimativa inicial  $d = 0.58641$ (b) Resultado da convergência  $d = 0.161554$ 

Figura 4.20: Imagem 1 do experimento 7, convergência de (a) para (b), afim de demonstrar a convergência usando imagem sintética sem sombreamento apenas textura

Inicial	Final
$near = 1$	$near = 1$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.944234 & -0.30345 & 0.127831 & 0 \\ -0.270802 & -0.936567 & -0.222507 & 0 \\ -0.187242 & -0.175482 & 0.966513 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 46.2286 & 0 & 0 & 0 \\ 0 & 46.2286 & 0 & 0 \\ 0 & 0 & 46.2286 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 46.2286 & 0 & 0 & 0 \\ 0 & 46.2286 & 0 & 0 \\ 0 & 0 & 46.2286 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 0$	$t_x = 0$
$t_y = 0$	$t_y = 0$

Tabela 4.22: Parâmetros iniciais e finais da figura 4.20

## 4.8 Experimento 8

Este experimento consiste em um teste com vídeo, o vídeo foi filmado em 640x480 em 120 FPS, onde a câmera estava estacionaria e o objeto foi rotacionado até ocorrer um volta completa. A base rotatória em que o sapo foi colocado, não era ideal, fazendo com o sapo rotaciona-se em mais de um eixo e saísse de sua posição inicial, 49 *frames* foram selecionados e pré-processados, as figuras 4.15 até 4.63, onde uma estimativa inicial foi usada para o *frame* 0 e os *frames* subsequentes usam como entrada a pose resultado do *frame* anterior.

A seguir temos o vídeo analisado e um vídeo mostrando as texturas resultantes de cada *frame*. Para poder visualizar os vídeos neste documento é necessário abrir este arquivo no *Adobe Acrobat Reader* (disponível em: <http://get.adobe.com/reader> )





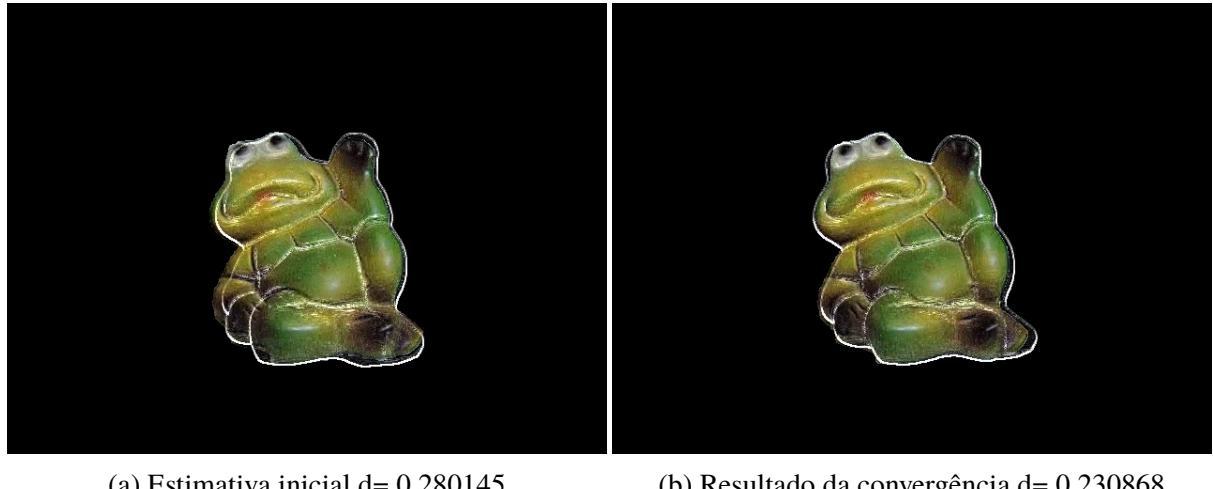
(a) Estimativa inicial d= 0.231983

(b) Resultado da convergência d= 0.225317

Figura 4.21: Imagem 1 do experimento 8, convergência de (a) para (b), do frame de vídeo 0

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.959805 & 0.0337184 & 0.278634 & 0 \\ 0 & -0.992757 & 0.120137 & 0 \\ -0.280667 & 0.115308 & 0.952854 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.968385 & 0.0380634 & 0.246542 & 0 \\ -0.00285935 & -0.991531 & 0.129838 & 0 \\ -0.249396 & 0.126438 & 0.960074 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.23: Parâmetros iniciais e finais da figura 4.21



(a) Estimativa inicial d= 0.280145

(b) Resultado da convergência d= 0.230868

Figura 4.22: Imagem 2 do experimento 8, convergência de (a) para (b), do frame de vídeo 1

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.968385 & 0.0380634 & 0.246542 & 0 \\ -0.00285935 & -0.991531 & 0.129838 & 0 \\ -0.249396 & 0.126438 & 0.960074 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.994441 & 0.0316038 & 0.100444 & 0 \\ 0.00409104 & -0.994669 & 0.103036 & 0 \\ -0.103164 & 0.102052 & 0.989269 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.24: Parâmetros iniciais e finais da figura 4.22

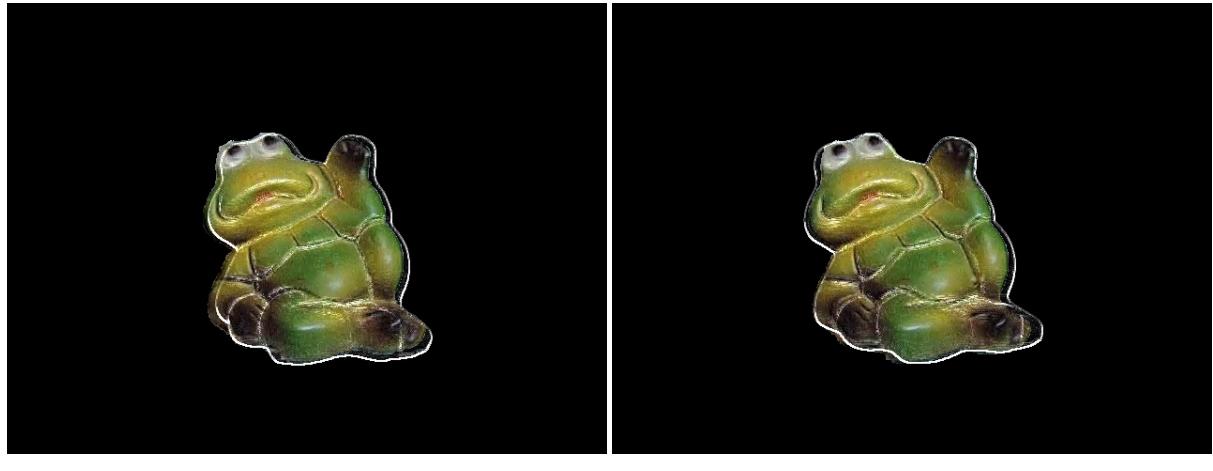


Figura 4.23: Imagem 3 do experimento 8, convergência de (a) para (b), do frame de vídeo 2

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.994441 & 0.0316038 & 0.100444 & 0 \\ 0.00409104 & -0.994669 & 0.103036 & 0 \\ -0.103164 & 0.102052 & 0.989269 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.995746 & 0.00808057 & -0.0917878 & 0 \\ 0.00660219 & -0.99686 & 0.0789138 & 0 \\ 0.0908619 & 0.0791841 & 0.992672 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.25: Parâmetros iniciais e finais da figura 4.23

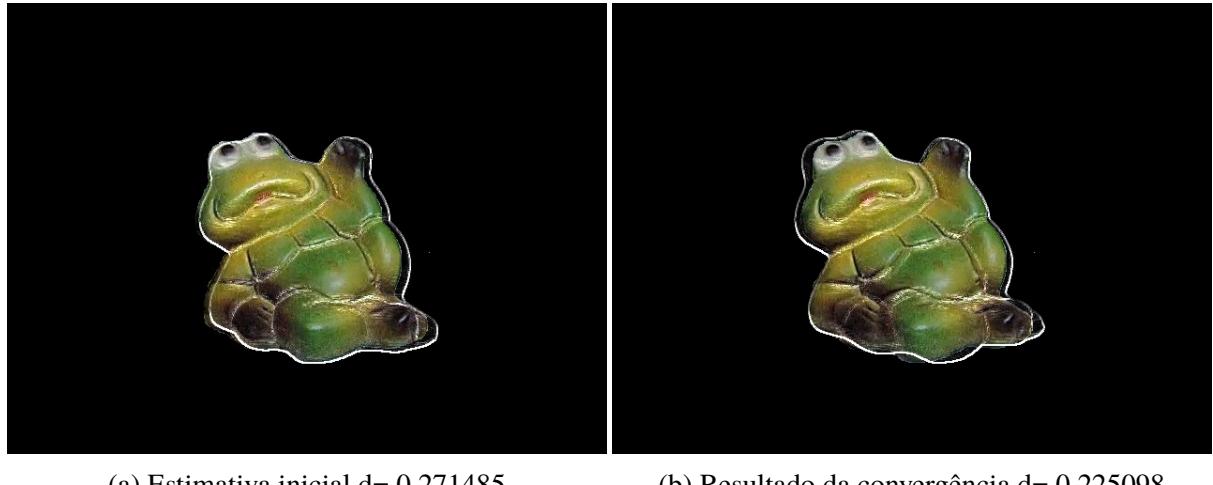
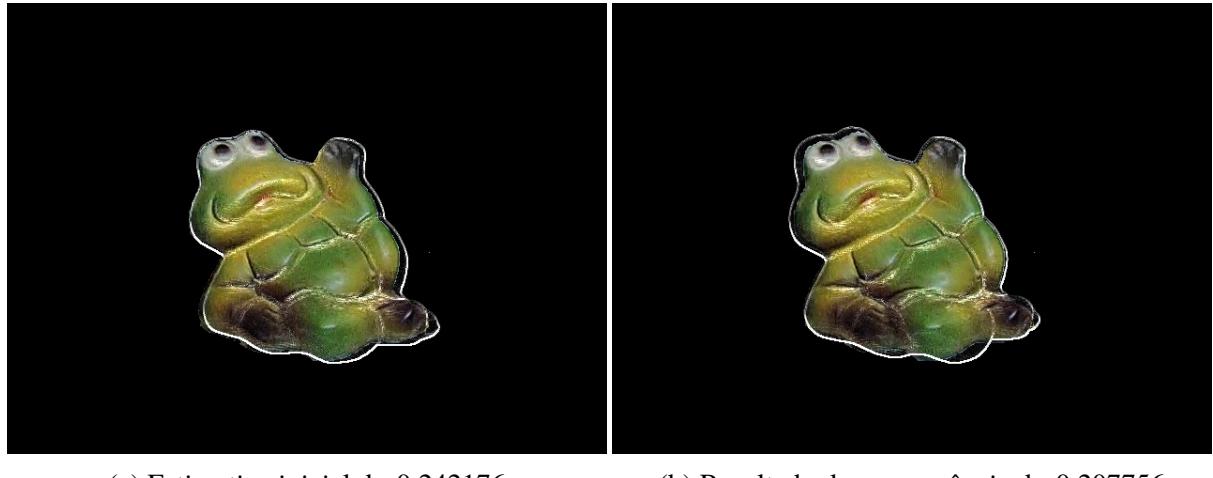
(a) Estimativa inicial  $d = 0.271485$ (b) Resultado da convergência  $d = 0.225098$ 

Figura 4.24: Imagem 4 do experimento 8, convergência de (a) para (b), do frame de vídeo 3

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.995746 & 0.00808057 & -0.0917878 & 0 \\ 0.00660219 & -0.99686 & 0.0789138 & 0 \\ 0.0908619 & 0.0791841 & 0.992672 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.952284 & -0.00820713 & -0.305105 & 0 \\ 0.00854367 & -0.994938 & 0.100123 & 0 \\ 0.304382 & 0.0979522 & 0.947393 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.26: Parâmetros iniciais e finais da figura 4.24



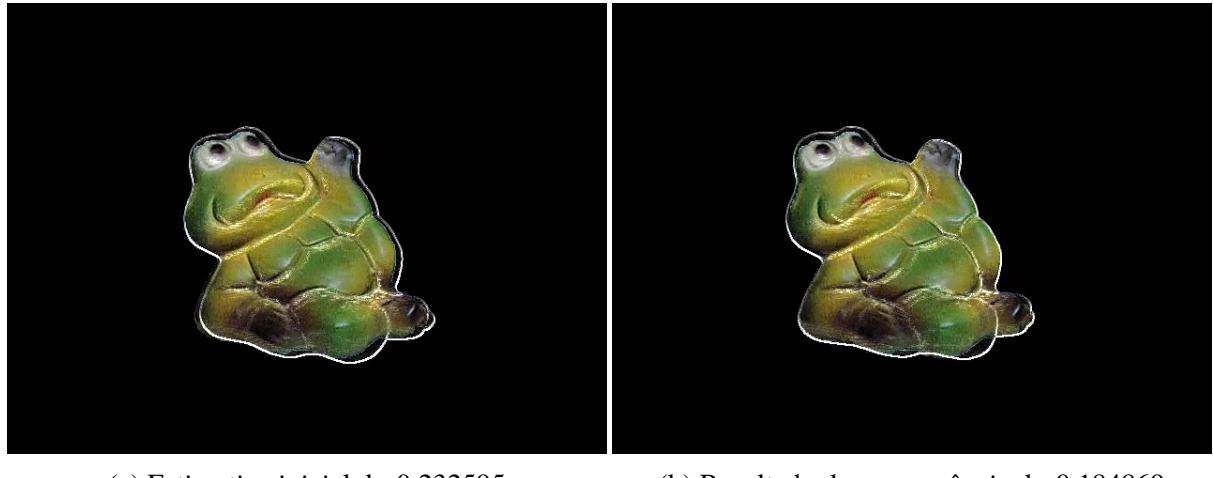
(a) Estimativa inicial d= 0.242176

(b) Resultado da convergência d= 0.207756

Figura 4.25: Imagem 5 do experimento 8, convergência de (a) para (b), do frame de vídeo 4

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.952284 & -0.00820713 & -0.305105 & 0 \\ 0.00854367 & -0.994938 & 0.100123 & 0 \\ 0.304382 & 0.0979522 & 0.947393 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.886735 & -0.0401813 & -0.460529 & 0 \\ 0.0211046 & -0.990046 & 0.139157 & 0 \\ 0.461536 & 0.133114 & 0.87706 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.27: Parâmetros iniciais e finais da figura 4.25



(a) Estimativa inicial d= 0.232595

(b) Resultado da convergência d= 0.184869

Figura 4.26: Imagem 6 do experimento 8, convergência de (a) para (b), do frame de vídeo 5

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.886735 & -0.0401813 & -0.460529 & 0 \\ 0.0211046 & -0.990046 & 0.139157 & 0 \\ 0.461536 & 0.133114 & 0.87706 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.814651 & -0.0163909 & -0.57972 & 0 \\ 0.060081 & -0.995187 & 0.0774123 & 0 \\ 0.578199 & 0.0978941 & 0.809745 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.28: Parâmetros iniciais e finais da figura 4.26

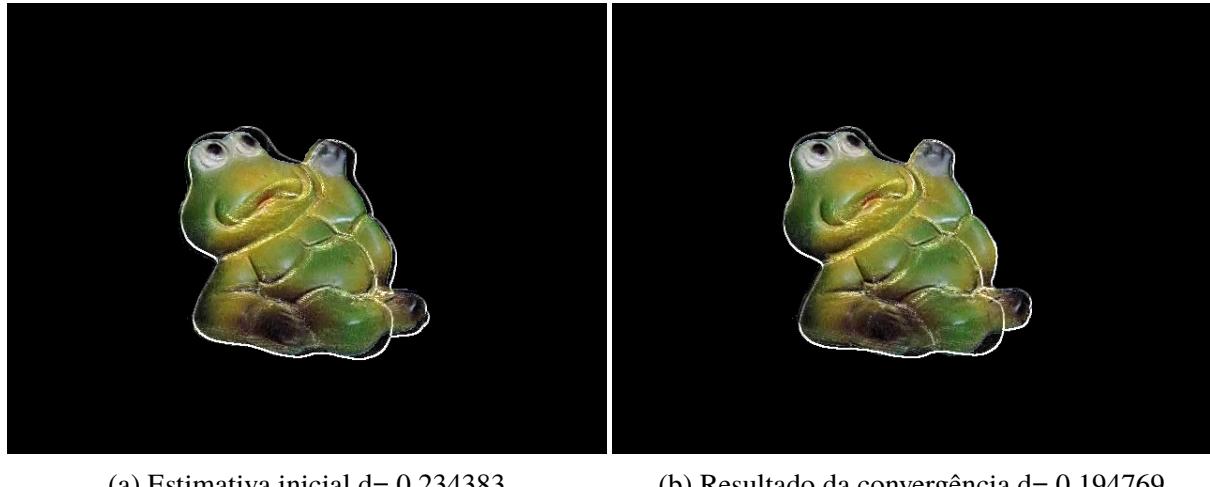
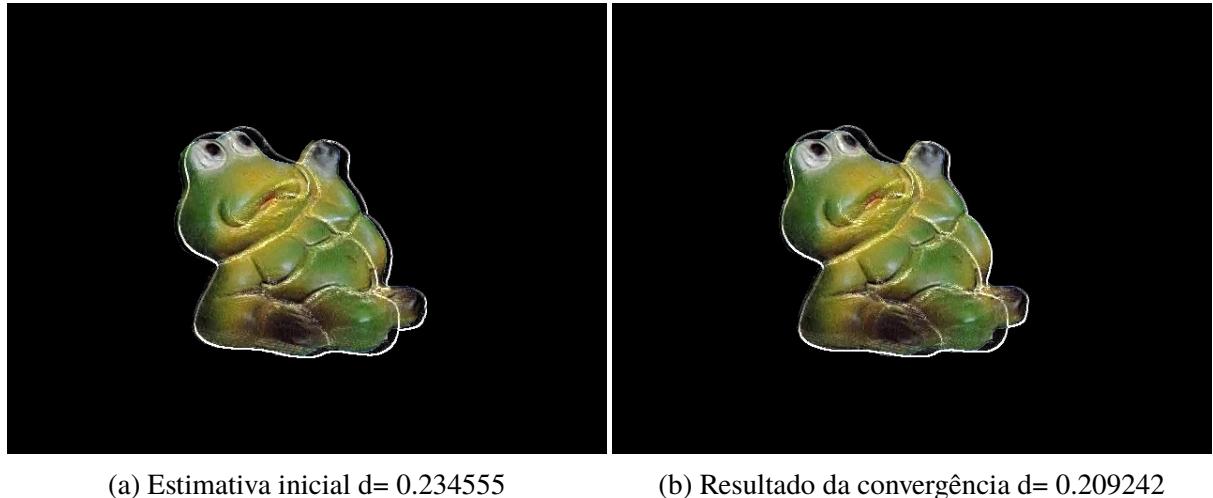


Figura 4.27: Imagem 7 do experimento 8, convergência de (a) para (b), do frame de vídeo 6

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.814651 & -0.0163909 & -0.57972 & 0 \\ 0.060081 & -0.995187 & 0.0774123 & 0 \\ 0.578199 & 0.0978941 & 0.809745 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.741317 & -0.0397226 & -0.669978 & 0 \\ 0.0425919 & -0.997198 & 0.0615052 & 0 \\ 0.670544 & 0.0741305 & 0.737548 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.29: Parâmetros iniciais e finais da figura 4.27



(a) Estimativa inicial d= 0.234555

(b) Resultado da convergência d= 0.209242

Figura 4.28: Imagem 8 do experimento 8, convergência de (a) para (b), do frame de vídeo 7

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.741317 & -0.0397226 & -0.669978 & 0 \\ 0.0425919 & -0.997198 & 0.0615052 & 0 \\ 0.670544 & 0.0741305 & 0.737548 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.669902 & -0.0312623 & -0.741791 & 0 \\ 0.0511523 & -0.998117 & 0.0338505 & 0 \\ 0.741453 & 0.0606209 & 0.667041 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.30: Parâmetros iniciais e finais da figura 4.28

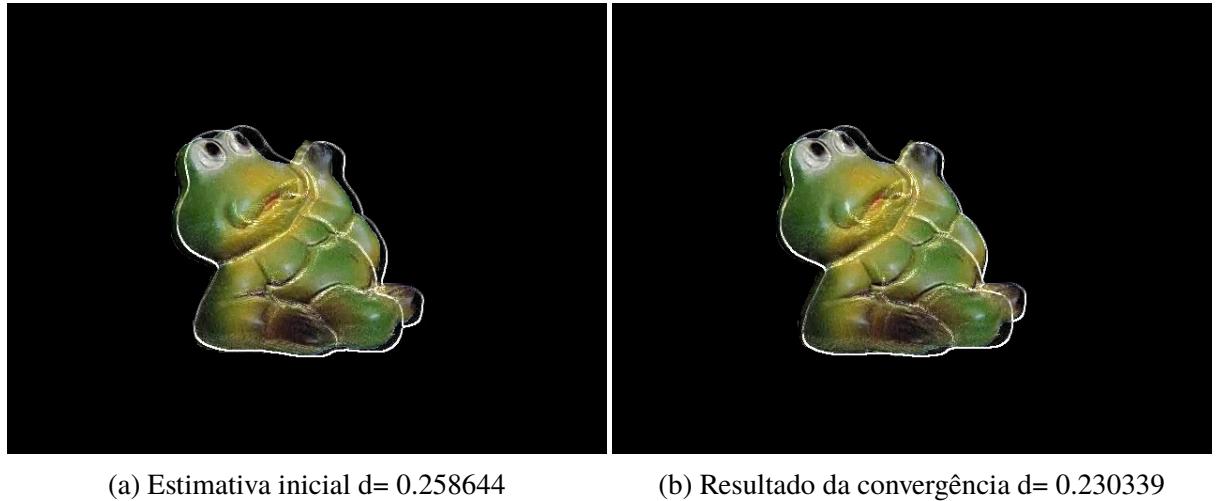
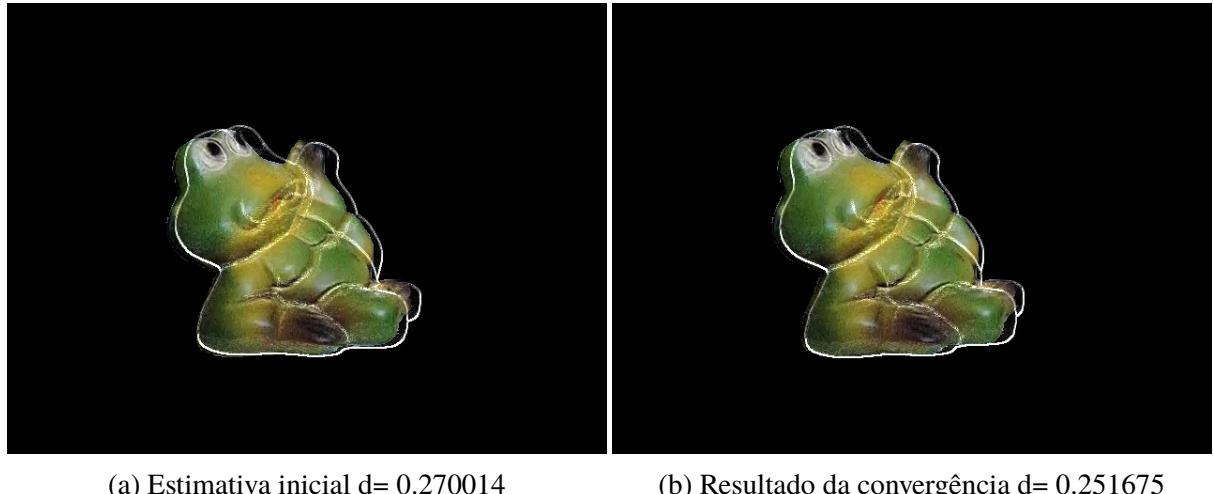


Figura 4.29: Imagem 9 do experimento 8, convergência de (a) para (b), do frame de vídeo 8

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.669902 & -0.0312623 & -0.741791 & 0 \\ 0.0511523 & -0.998117 & 0.0338505 & 0 \\ 0.741453 & 0.0606209 & 0.667041 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.586351 & -0.0372006 & -0.809203 & 0 \\ 0.0359958 & -0.999275 & 0.0123689 & 0 \\ 0.809076 & 0.0363805 & 0.584587 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.31: Parâmetros iniciais e finais da figura 4.29



(a) Estimativa inicial d= 0.270014

(b) Resultado da convergência d= 0.251675

Figura 4.30: Imagem 10 do experimento 8, convergência de (a) para (b), do frame de vídeo 9

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.586351 & -0.0372006 & -0.809203 & 0 \\ 0.0359958 & -0.999275 & 0.0123689 & 0 \\ 0.809076 & 0.0363805 & 0.584587 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.527548 & -0.0371635 & -0.848712 & 0 \\ 0.0244159 & -0.999577 & -0.0158056 & 0 \\ 0.847765 & 0.0123838 & 0.526418 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.32: Parâmetros iniciais e finais da figura 4.30

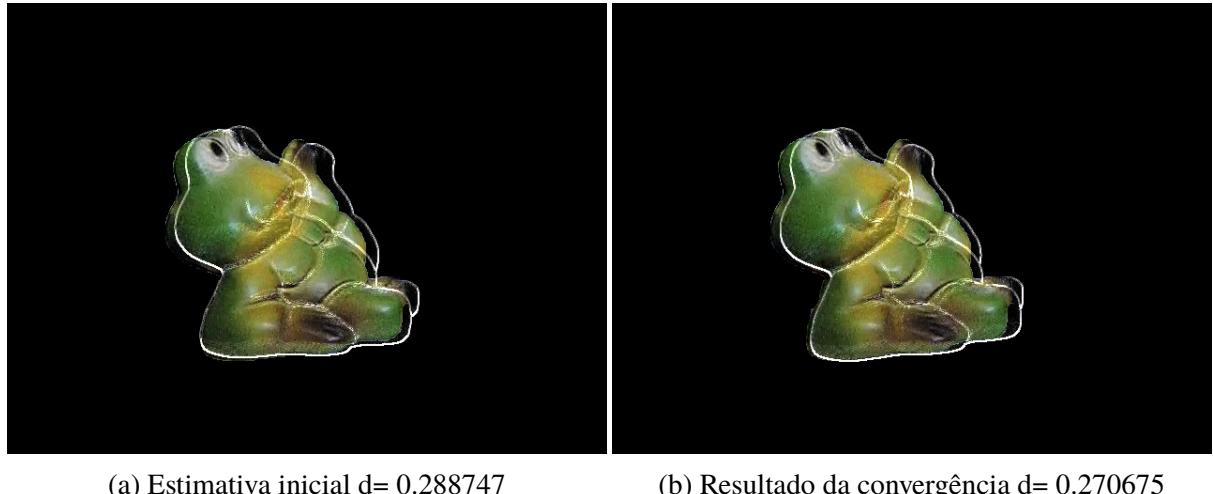
(a) Estimativa inicial  $d = 0.288747$ (b) Resultado da convergência  $d = 0.270675$ 

Figura 4.31: Imagem 11 do experimento 8, convergência de (a) para (b), do frame de vídeo 10

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.527548 & -0.0371635 & -0.848712 & 0 \\ 0.0244159 & -0.999577 & -0.0158056 & 0 \\ 0.847765 & 0.0123838 & 0.526418 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.469118 & -0.0393177 & -0.88226 & 0 \\ -0.00804784 & -0.999092 & -0.041832 & 0 \\ 0.879814 & -0.0267244 & 0.469008 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.33: Parâmetros iniciais e finais da figura 4.31

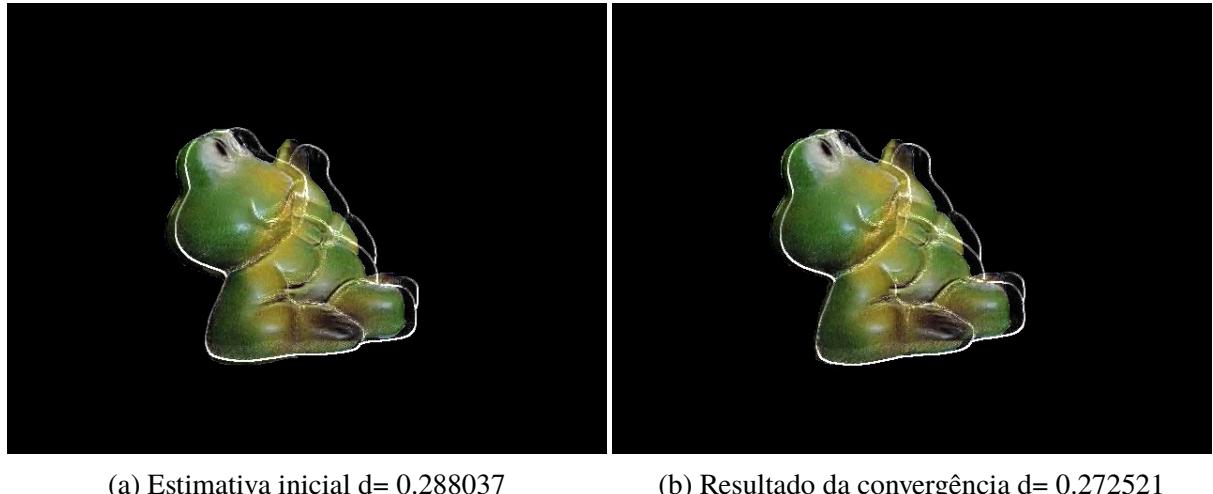
(a) Estimativa inicial  $d = 0.288037$ (b) Resultado da convergência  $d = 0.272521$ 

Figura 4.32: Imagem 12 do experimento 8, convergência de (a) para (b), do frame de vídeo 11

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.469118 & -0.0393177 & -0.88226 & 0 \\ -0.00804784 & -0.999092 & -0.041832 & 0 \\ 0.879814 & -0.0267244 & 0.469008 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.451488 & -0.0755348 & -0.889074 & 0 \\ -0.0476734 & -0.998674 & -0.0194433 & 0 \\ 0.886427 & -0.0511636 & 0.45449 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.34: Parâmetros iniciais e finais da figura 4.32

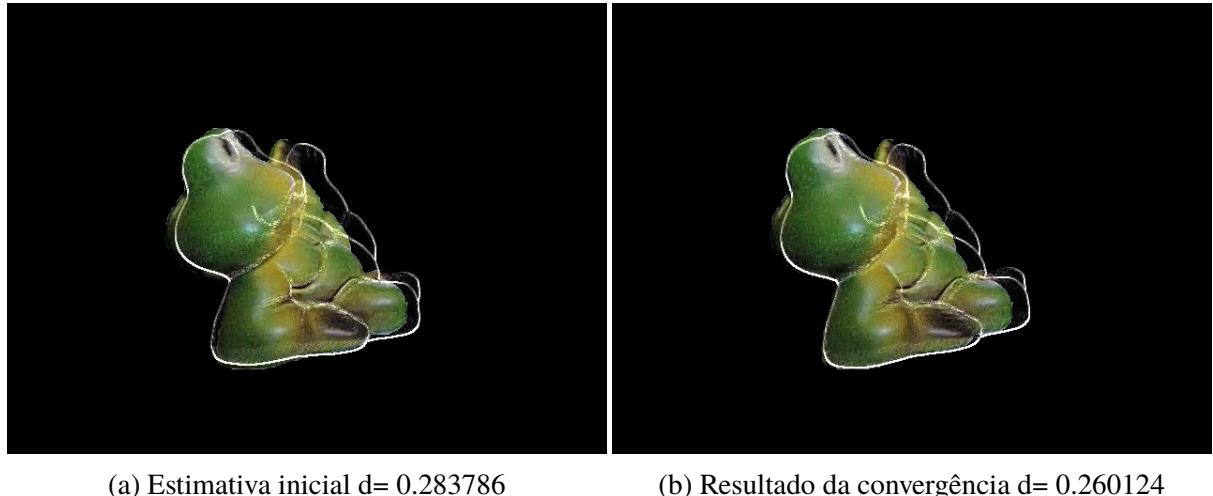
(a) Estimativa inicial  $d = 0.283786$ (b) Resultado da convergência  $d = 0.260124$ 

Figura 4.33: Imagem 13 do experimento 8, convergência de (a) para (b), do frame de vídeo 12

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.451488 & -0.0755348 & -0.889074 & 0 \\ -0.0476734 & -0.998674 & -0.0194433 & 0 \\ 0.886427 & -0.0511636 & 0.45449 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.412432 & -0.0884594 & -0.906683 & 0 \\ -0.0954776 & -0.995174 & -0.0226488 & 0 \\ 0.900304 & -0.095909 & 0.418888 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.35: Parâmetros iniciais e finais da figura 4.33

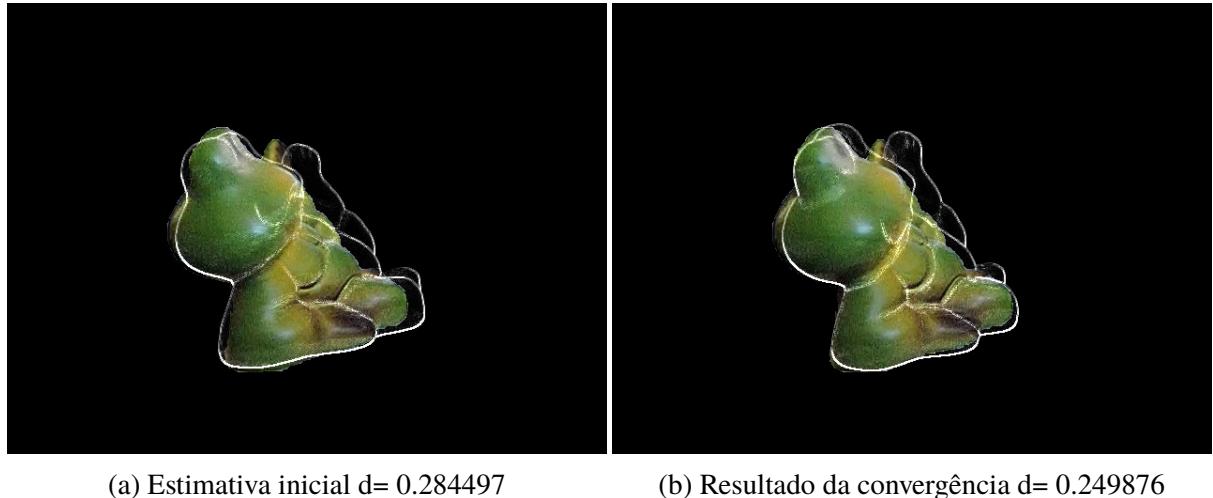
(a) Estimativa inicial  $d = 0.284497$ (b) Resultado da convergência  $d = 0.249876$ 

Figura 4.34: Imagem 14 do experimento 8, convergência de (a) para (b), do frame de vídeo 13

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.412432 & -0.0884594 & -0.906683 & 0 \\ -0.0954776 & -0.995174 & -0.0226488 & 0 \\ 0.900304 & -0.095909 & 0.418888 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.24607 & -0.242636 & -0.938391 & 0 \\ -0.00895643 & -0.97092 & 0.239238 & 0 \\ 0.96915 & 0.0504647 & 0.241088 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.36: Parâmetros iniciais e finais da figura 4.34

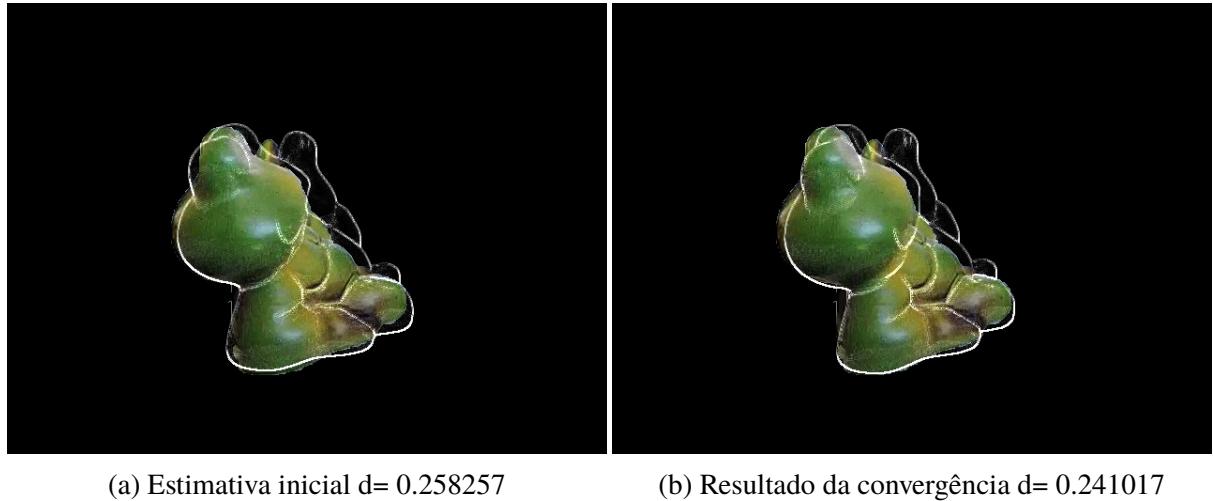


Figura 4.35: Imagem 15 do experimento 8, convergência de (a) para (b), do frame de vídeo 14

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.24607 & -0.242636 & -0.938391 & 0 \\ -0.00895643 & -0.97092 & 0.239238 & 0 \\ 0.96915 & 0.0504647 & 0.241088 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.152198 & -0.294817 & -0.943355 & 0 \\ 0.0161041 & -0.957561 & 0.28778 & 0 \\ 0.988163 & 0.0589913 & 0.140991 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.37: Parâmetros iniciais e finais da figura 4.35

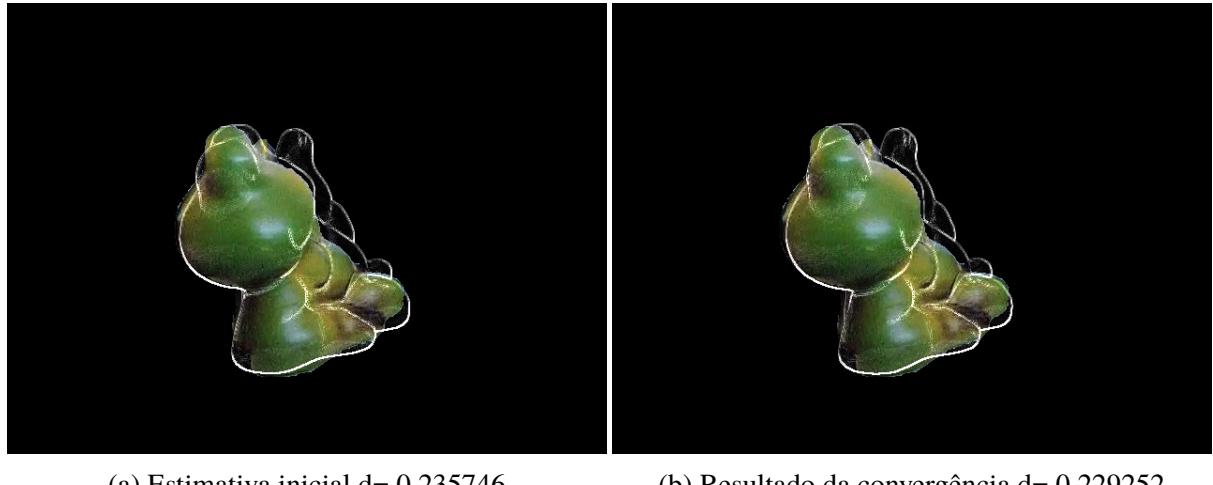
(a) Estimativa inicial  $d= 0.235746$ (b) Resultado da convergência  $d= 0.229252$ 

Figura 4.36: Imagem 16 do experimento 8, convergência de (a) para (b), do frame de vídeo 15

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.152198 & -0.294817 & -0.943355 & 0 \\ 0.0161041 & -0.957561 & 0.28778 & 0 \\ 0.988163 & 0.0589913 & 0.140991 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.0992479 & -0.306501 & -0.946682 & 0 \\ 0.0396475 & -0.95274 & 0.301187 & 0 \\ 0.994257 & 0.0674258 & 0.0824055 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.38: Parâmetros iniciais e finais da figura 4.36

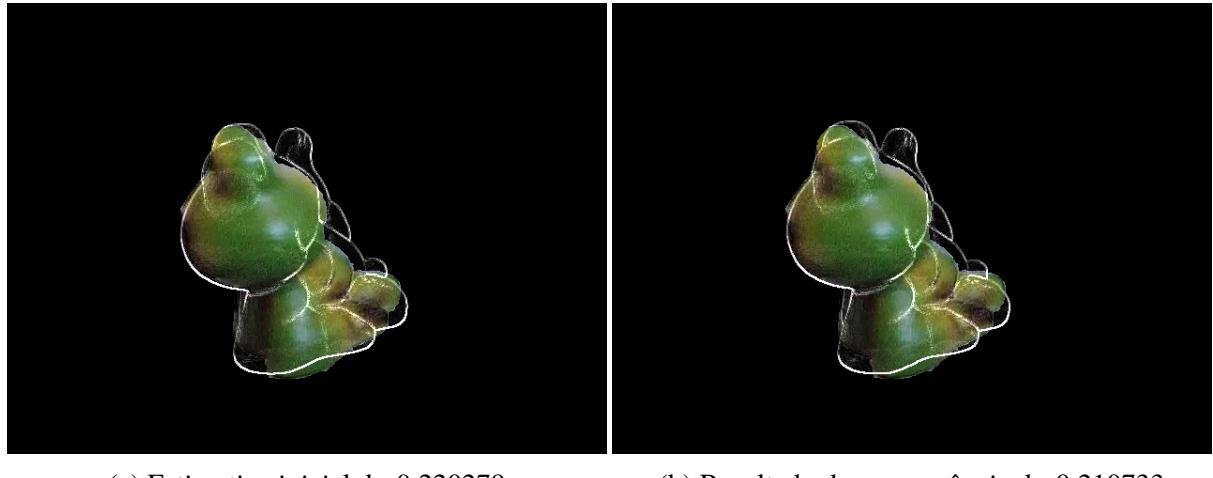
(a) Estimativa inicial  $d = 0.220279$ (b) Resultado da convergência  $d = 0.210733$ 

Figura 4.37: Imagem 17 do experimento 8, convergência de (a) para (b), do frame de vídeo 16

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.0992479 & -0.306501 & -0.946682 & 0 \\ 0.0396475 & -0.95274 & 0.301187 & 0 \\ 0.994257 & 0.0674258 & 0.0824055 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.0657453 & -0.295766 & -0.952995 & 0 \\ 0.0694289 & -0.954839 & 0.288897 & 0 \\ 0.995403 & 0.0851591 & 0.0422415 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.39: Parâmetros iniciais e finais da figura 4.37

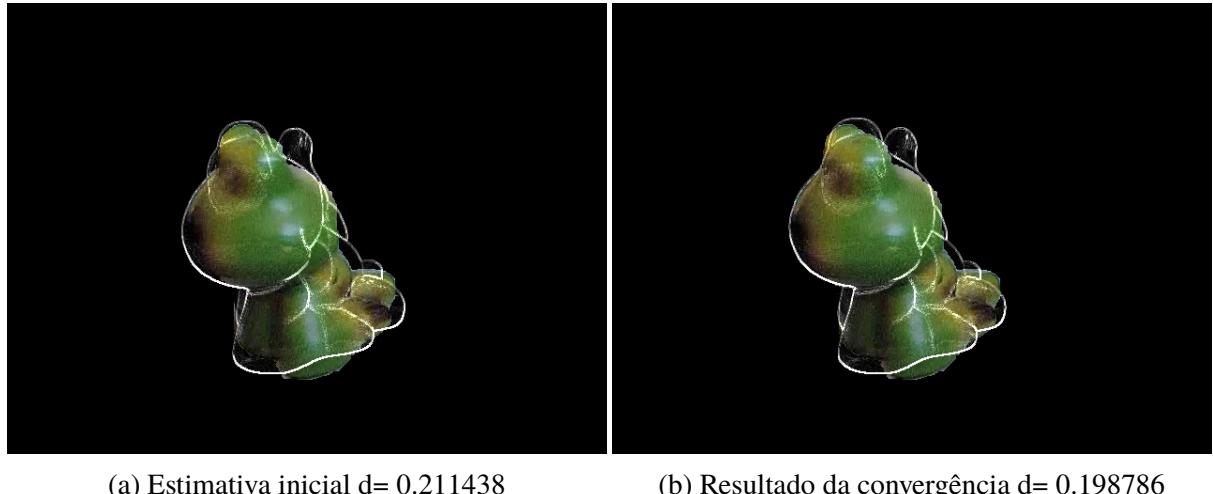
(a) Estimativa inicial  $d = 0.211438$ (b) Resultado da convergência  $d = 0.198786$ 

Figura 4.38: Imagem 18 do experimento 8, convergência de (a) para (b), do frame de vídeo 17

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.0657453 & -0.295766 & -0.952995 & 0 \\ 0.0694289 & -0.954839 & 0.288897 & 0 \\ 0.995403 & 0.0851591 & 0.0422415 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.00699103 & -0.290044 & -0.956988 & 0 \\ 0.111517 & -0.954111 & 0.277915 & 0 \\ 0.99368 & 0.108663 & -0.0256745 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.40: Parâmetros iniciais e finais da figura 4.38

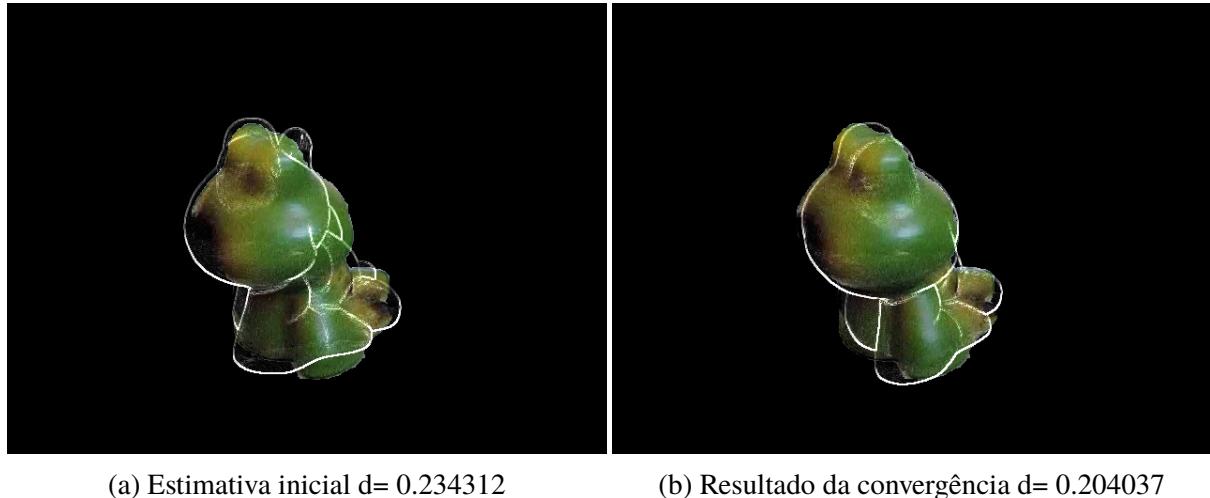


Figura 4.39: Imagem 19 do experimento 8, convergência de (a) para (b), do frame de vídeo 18

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.00699103 & -0.290044 & -0.956988 & 0 \\ 0.111517 & -0.954111 & 0.277915 & 0 \\ 0.99368 & 0.108663 & -0.0256745 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.31292 & -0.301489 & -0.900658 & 0 \\ 0.0498406 & -0.968516 & 0.243911 & 0 \\ 0.945839 & -0.0314354 & -0.318094 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.41: Parâmetros iniciais e finais da figura 4.39

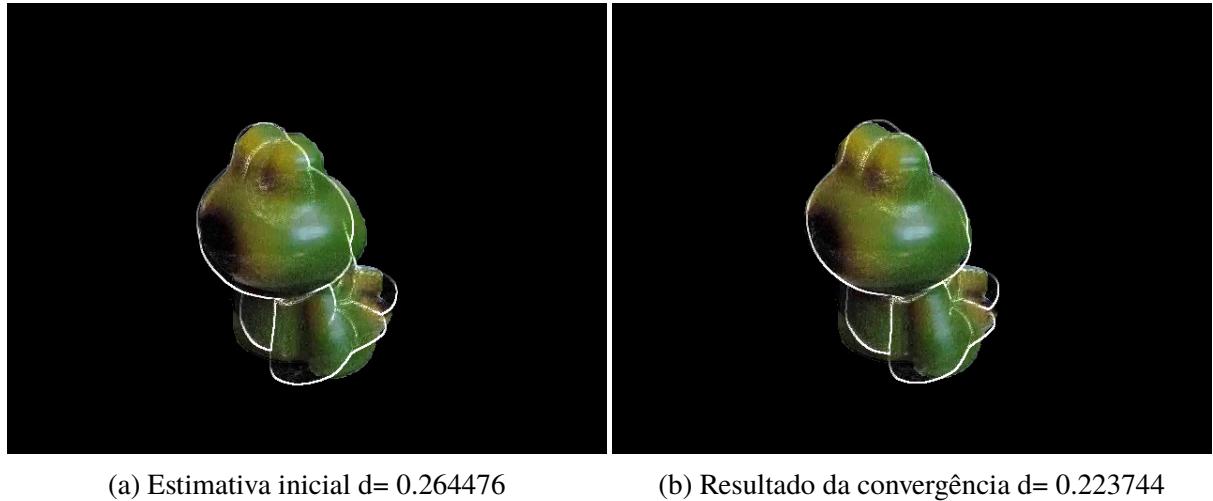


Figura 4.40: Imagem 20 do experimento 8, convergência de (a) para (b), do frame de vídeo 19

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.31292 & -0.301489 & -0.900658 & 0 \\ 0.0498406 & -0.968516 & 0.243911 & 0 \\ 0.945839 & -0.0314354 & -0.318094 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.429873 & -0.242001 & -0.869854 & 0 \\ 0.0269653 & -0.979951 & 0.197406 & 0 \\ 0.900186 & -0.0614036 & -0.42778 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.42: Parâmetros iniciais e finais da figura 4.40

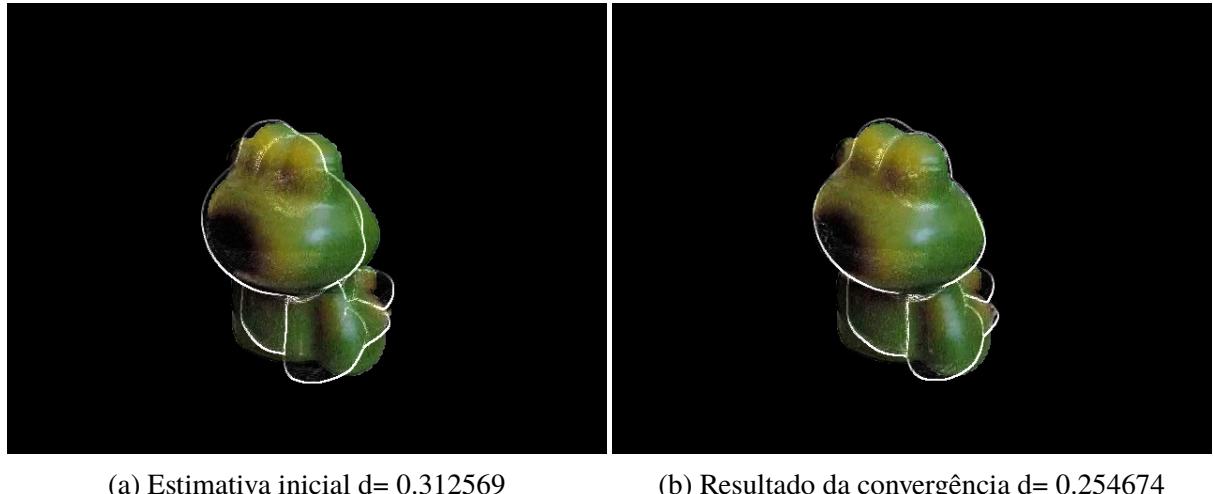
(a) Estimativa inicial  $d = 0.312569$ (b) Resultado da convergência  $d = 0.254674$ 

Figura 4.41: Imagem 21 do experimento 8, convergência de (a) para (b), do frame de vídeo 20

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.429873 & -0.242001 & -0.869854 & 0 \\ 0.0269653 & -0.979951 & 0.197406 & 0 \\ 0.900186 & -0.0614036 & -0.42778 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.568143 & -0.186919 & -0.80142 & 0 \\ -0.0093309 & -0.986197 & 0.165315 & 0 \\ 0.821259 & -0.101401 & -0.558557 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.43: Parâmetros iniciais e finais da figura 4.41

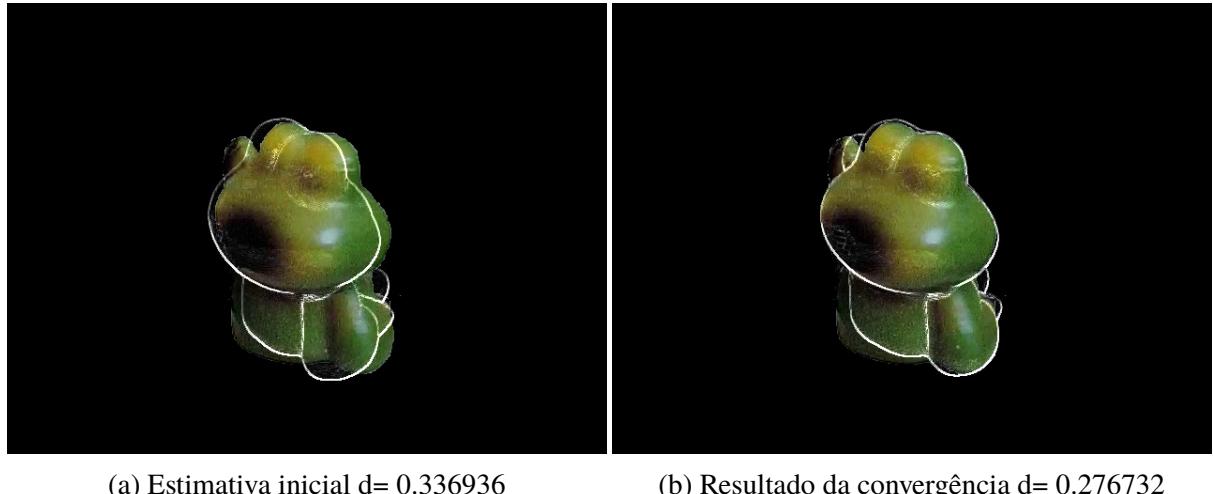
(a) Estimativa inicial  $d = 0.336936$ (b) Resultado da convergência  $d = 0.276732$ 

Figura 4.42: Imagem 22 do experimento 8, convergência de (a) para (b), do frame de vídeo 21

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.568143 & -0.186919 & -0.80142 & 0 \\ -0.0093309 & -0.986197 & 0.165315 & 0 \\ 0.821259 & -0.101401 & -0.558557 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.693999 & -0.131072 & -0.707944 & 0 \\ -0.0769236 & -0.983605 & 0.16311 & 0 \\ 0.717716 & -0.167656 & -0.672538 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.44: Parâmetros iniciais e finais da figura 4.42

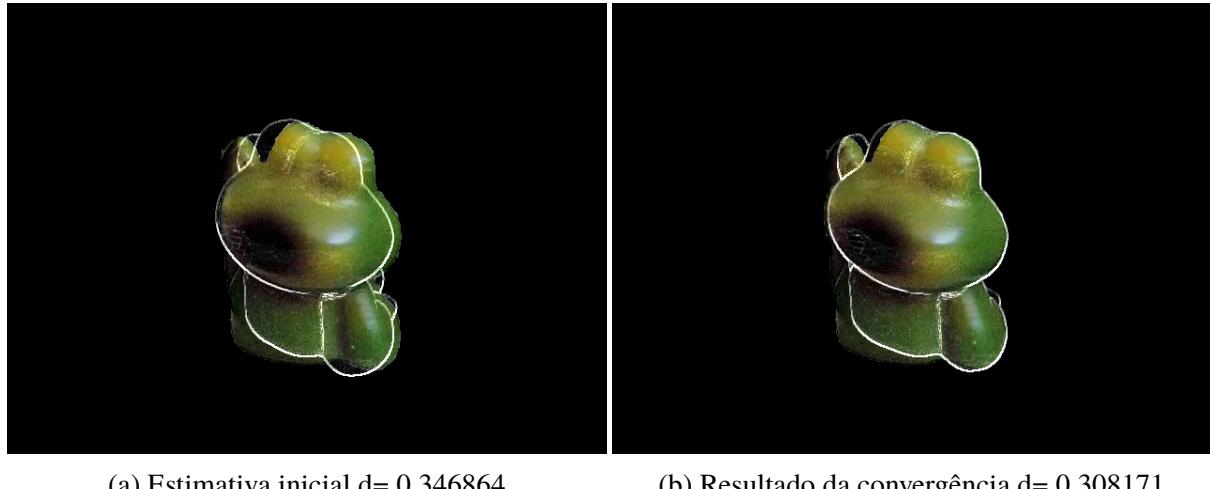
(a) Estimativa inicial  $d = 0.346864$ (b) Resultado da convergência  $d = 0.308171$ 

Figura 4.43: Imagem 23 do experimento 8, convergência de (a) para (b), do frame de vídeo 22

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.693999 & -0.131072 & -0.707944 & 0 \\ -0.0769236 & -0.983605 & 0.16311 & 0 \\ 0.717716 & -0.167656 & -0.672538 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.783403 & -0.10329 & -0.612871 & 0 \\ -0.10183 & -0.977669 & 0.183834 & 0 \\ 0.618173 & -0.206425 & -0.755391 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.45: Parâmetros iniciais e finais da figura 4.43

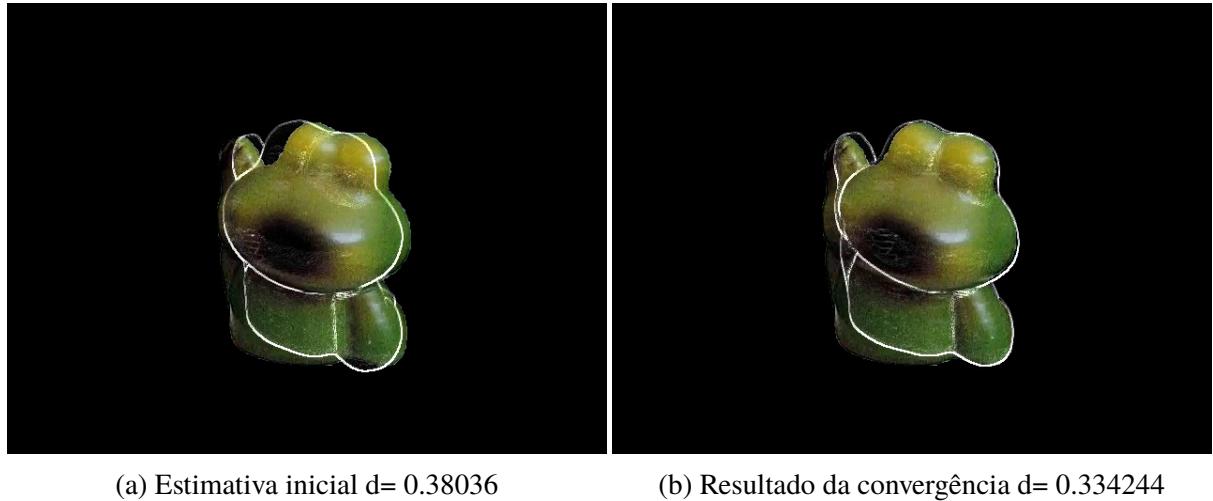
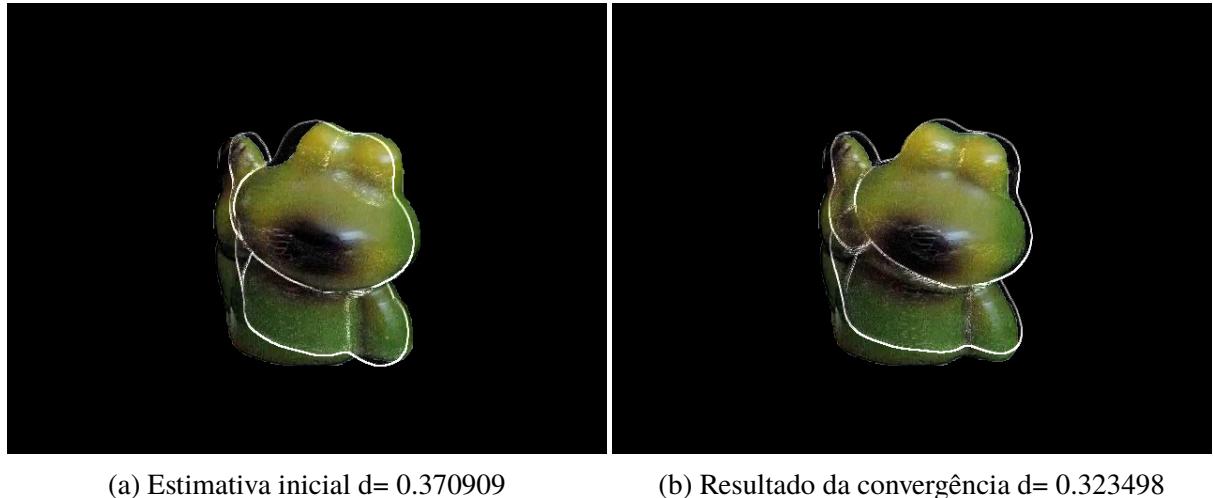
(a) Estimativa inicial  $d = 0.38036$ (b) Resultado da convergência  $d = 0.334244$ 

Figura 4.44: Imagem 24 do experimento 8, convergência de (a) para (b), do frame de vídeo 23

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.783403 & -0.10329 & -0.612871 & 0 \\ -0.10183 & -0.977669 & 0.183834 & 0 \\ 0.618173 & -0.206425 & -0.755391 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.883715 & -0.0893441 & -0.45942 & 0 \\ -0.098134 & -0.977749 & 0.185408 & 0 \\ 0.465762 & -0.208932 & -0.855284 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.46: Parâmetros iniciais e finais da figura 4.44



(a) Estimativa inicial d= 0.370909

(b) Resultado da convergência d= 0.323498

Figura 4.45: Imagem 25 do experimento 8, convergência de (a) para (b), do frame de vídeo 24

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.883715 & -0.0893441 & -0.45942 & 0 \\ -0.098134 & -0.977749 & 0.185408 & 0 \\ 0.465762 & -0.208932 & -0.855284 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.974409 & -0.0322327 & -0.222461 & 0 \\ -0.098134 & -0.977749 & 0.185408 & 0 \\ 0.223487 & -0.202494 & -0.949564 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.47: Parâmetros iniciais e finais da figura 4.45

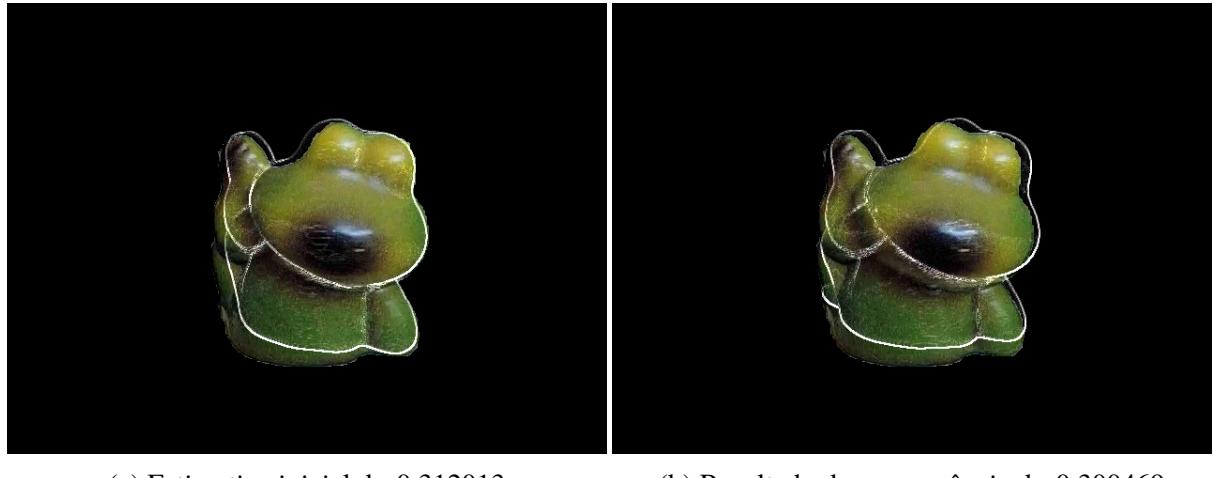
(a) Estimativa inicial  $d = 0.312013$ (b) Resultado da convergência  $d = 0.300469$ 

Figura 4.46: Imagem 26 do experimento 8, convergência de (a) para (b), do frame de vídeo 25

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.974409 & -0.0322327 & -0.222461 & 0 \\ -0.098134 & -0.977749 & 0.185408 & 0 \\ 0.223487 & -0.202494 & -0.949564 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.999296 & 0.0212802 & 0.030892 & 0 \\ -0.098134 & -0.977749 & 0.185408 & 0 \\ -0.0341502 & -0.182246 & -0.979149 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.48: Parâmetros iniciais e finais da figura 4.46

(a) Estimativa inicial  $d= 0.290178$ (b) Resultado da convergência  $d= 0.283201$ 

Figura 4.47: Imagem 27 do experimento 8, convergência de (a) para (b), do frame de vídeo 26

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.999296 & 0.0212802 & 0.030892 & 0 \\ -0.098134 & -0.977749 & 0.185408 & 0 \\ -0.0341502 & -0.182246 & -0.979149 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.956628 & 0.0677395 & 0.283327 & 0 \\ -0.098134 & -0.977749 & 0.185408 & 0 \\ -0.289582 & -0.149562 & -0.94199 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.49: Parâmetros iniciais e finais da figura 4.47

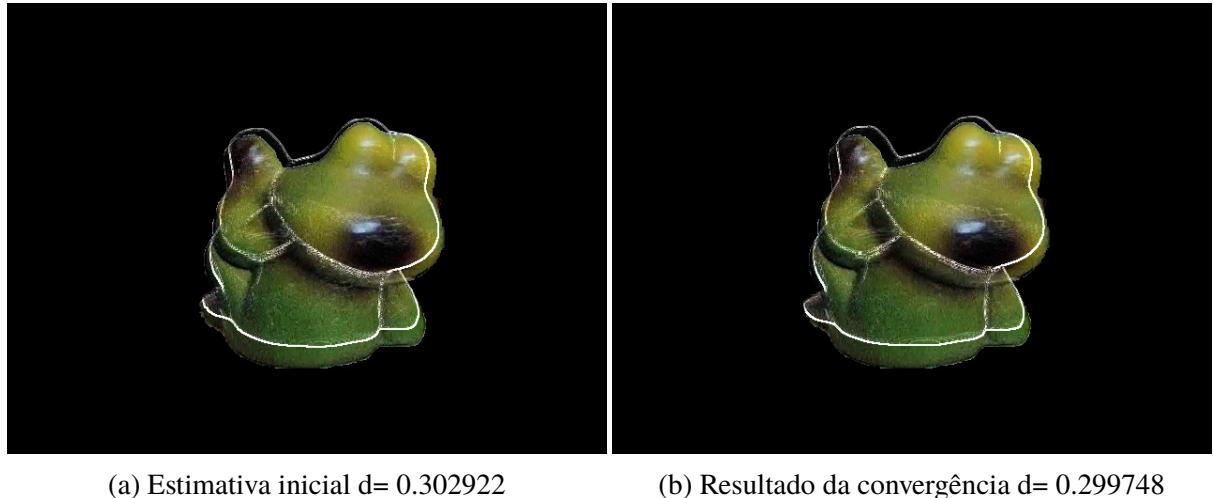


Figura 4.48: Imagem 28 do experimento 8, convergência de (a) para (b), do frame de vídeo 27

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.956628 & 0.0677395 & 0.283327 & 0 \\ -0.098134 & -0.977749 & 0.185408 & 0 \\ -0.289582 & -0.149562 & -0.94199 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.945507 & 0.0479416 & 0.322054 & 0 \\ -0.0887109 & -0.990047 & 0.109257 & 0 \\ -0.324087 & -0.0747339 & -0.940349 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

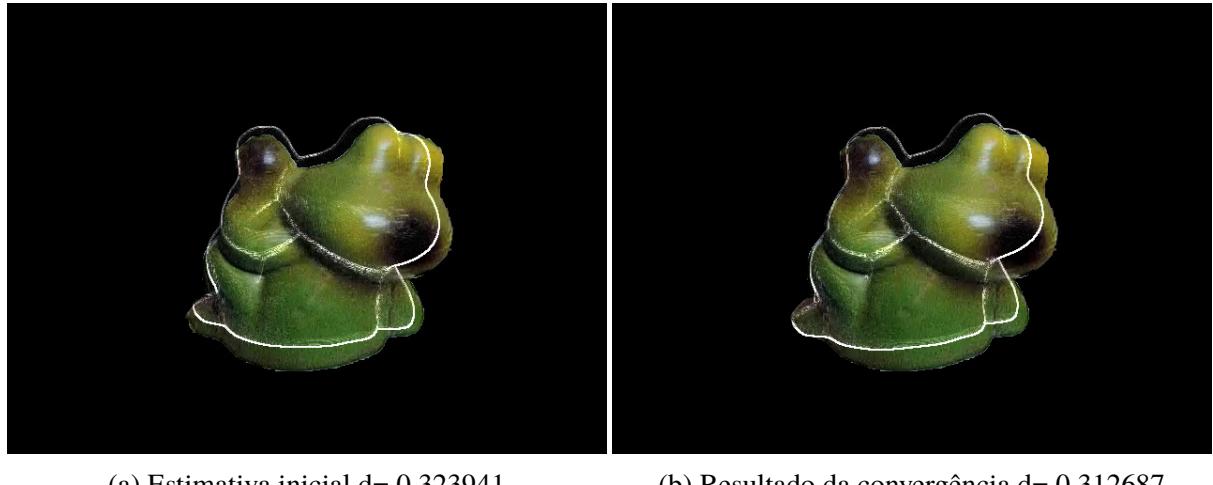
Tabela 4.50: Parâmetros iniciais e finais da figura 4.48



Figura 4.49: Imagem 29 do experimento 8, convergência de (a) para (b), do frame de vídeo 28

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.945507 & 0.0479416 & 0.322054 & 0 \\ -0.0887109 & -0.990047 & 0.109257 & 0 \\ -0.324087 & -0.0747339 & -0.940349 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.930874 & 0.0583041 & 0.360657 & 0 \\ -0.0440642 & -0.993375 & 0.106136 & 0 \\ -0.364456 & -0.082907 & -0.927276 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.51: Parâmetros iniciais e finais da figura 4.49



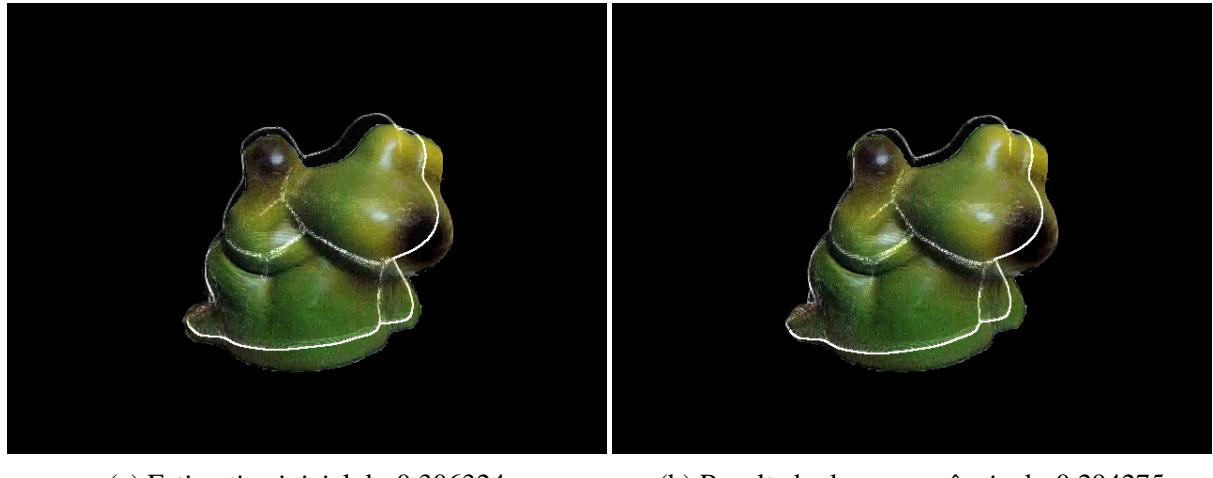
(a) Estimativa inicial d= 0.323941

(b) Resultado da convergência d= 0.312687

Figura 4.50: Imagem 30 do experimento 8, convergência de (a) para (b), do frame de vídeo 29

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.930874 & 0.0583041 & 0.360657 & 0 \\ -0.0440642 & -0.993375 & 0.106136 & 0 \\ -0.364456 & -0.082907 & -0.927276 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.885549 & 0.106299 & 0.45222 & 0 \\ -0.0416343 & -0.998744 & 0.0278926 & 0 \\ -0.454617 & -0.00587239 & -0.888862 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.52: Parâmetros iniciais e finais da figura 4.50



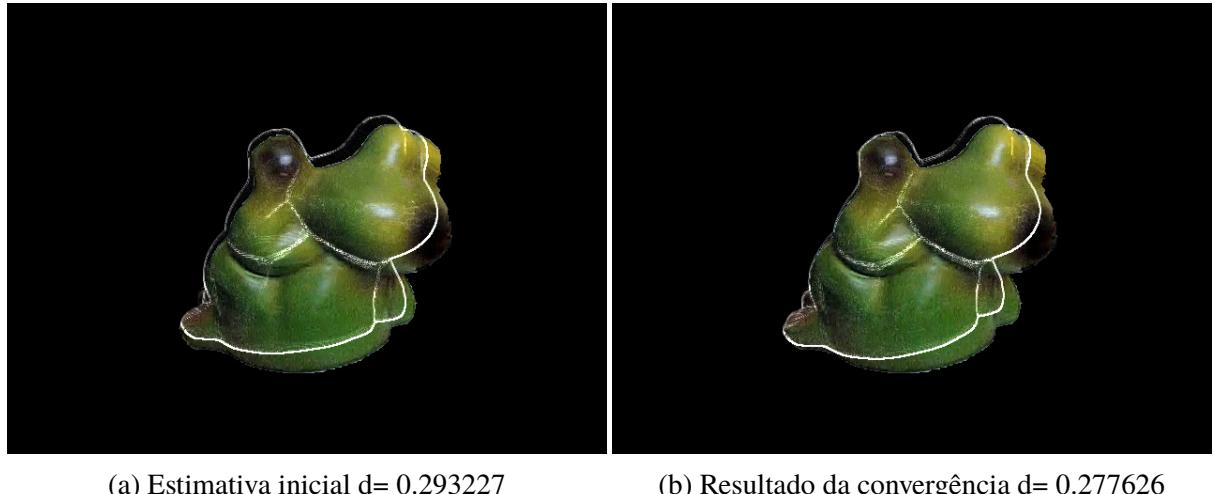
(a) Estimativa inicial d= 0.306324

(b) Resultado da convergência d= 0.294275

Figura 4.51: Imagem 31 do experimento 8, convergência de (a) para (b), do frame de vídeo 30

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.885549 & 0.106299 & 0.45222 & 0 \\ -0.0416343 & -0.998744 & 0.0278926 & 0 \\ -0.454617 & -0.00587239 & -0.888862 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.856061 & 0.137203 & 0.498332 & 0 \\ 0.0131529 & -0.998927 & 0.0444097 & 0 \\ -0.50389 & -0.0445719 & -0.853338 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.53: Parâmetros iniciais e finais da figura 4.51



(a) Estimativa inicial d= 0.293227

(b) Resultado da convergência d= 0.277626

Figura 4.52: Imagem 32 do experimento 8, convergência de (a) para (b), do frame de vídeo 31

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.856061 & 0.137203 & 0.498332 & 0 \\ 0.0131529 & -0.998927 & 0.0444097 & 0 \\ -0.50389 & -0.0445719 & -0.853338 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.792062 & 0.204938 & 0.575012 & 0 \\ 0.0342882 & -0.995628 & 0.0868848 & 0 \\ -0.590304 & -0.0885343 & -0.781572 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.54: Parâmetros iniciais e finais da figura 4.52

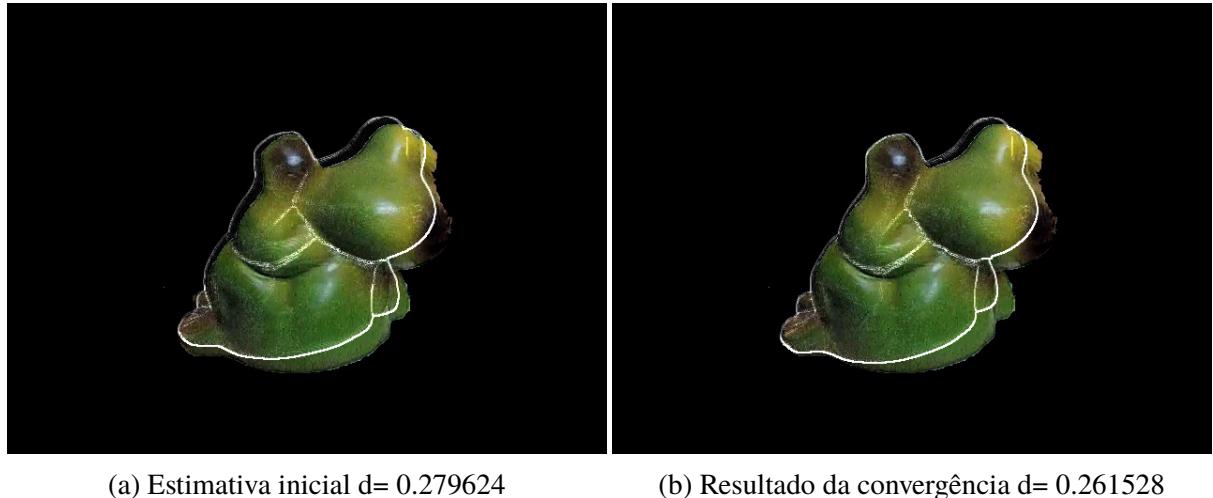


Figura 4.53: Imagem 33 do experimento 8, convergência de (a) para (b), do frame de vídeo 32

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.792062 & 0.204938 & 0.575012 & 0 \\ 0.0342882 & -0.995628 & 0.0868848 & 0 \\ -0.590304 & -0.0885343 & -0.781572 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.721598 & 0.249525 & 0.645781 & 0 \\ 0.0497818 & -0.993757 & 0.0998477 & 0 \\ -0.666664 & -0.104198 & -0.704671 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.55: Parâmetros iniciais e finais da figura 4.53

(a) Estimativa inicial  $d = 0.271396$ (b) Resultado da convergência  $d = 0.247144$ 

Figura 4.54: Imagem 34 do experimento 8, convergência de (a) para (b), do frame de vídeo 33

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.721598 & 0.249525 & 0.645781 & 0 \\ 0.0497818 & -0.993757 & 0.0998477 & 0 \\ -0.666664 & -0.104198 & -0.704671 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.596252 & 0.282407 & 0.751485 & 0 \\ 0.0438098 & -0.992858 & 0.110966 & 0 \\ -0.777456 & -0.0990863 & -0.579621 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.56: Parâmetros iniciais e finais da figura 4.54



(a) Estimativa inicial d= 0.256663

(b) Resultado da convergência d= 0.23449

Figura 4.55: Imagem 35 do experimento 8, convergência de (a) para (b), do frame de vídeo 34

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.596252 & 0.282407 & 0.751485 & 0 \\ 0.0438098 & -0.992858 & 0.110966 & 0 \\ -0.777456 & -0.0990863 & -0.579621 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.457679 & 0.317422 & 0.830526 & 0 \\ 0.0365704 & -0.989425 & 0.140357 & 0 \\ -0.866296 & -0.0946109 & -0.441231 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.57: Parâmetros iniciais e finais da figura 4.55

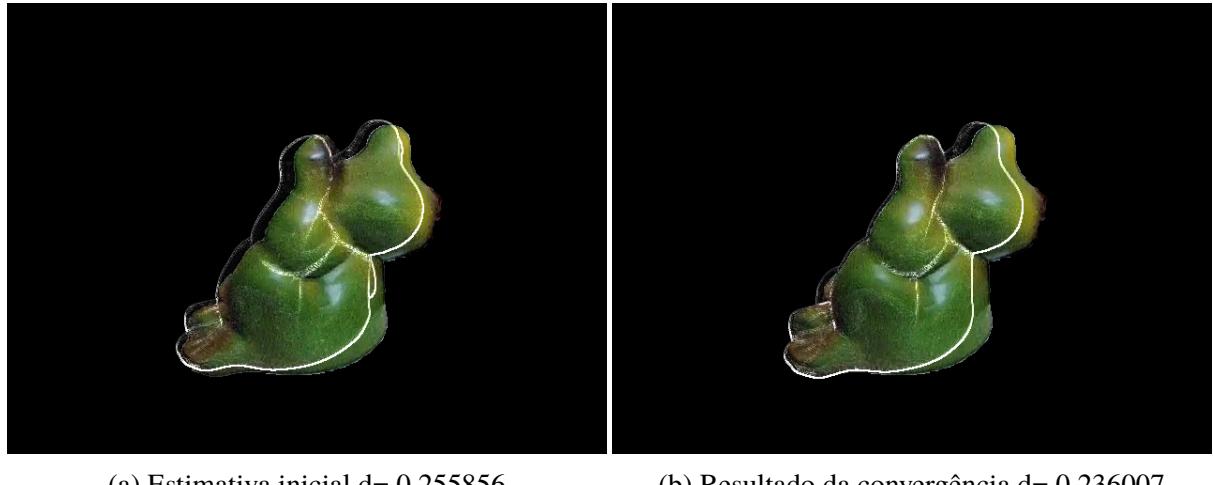
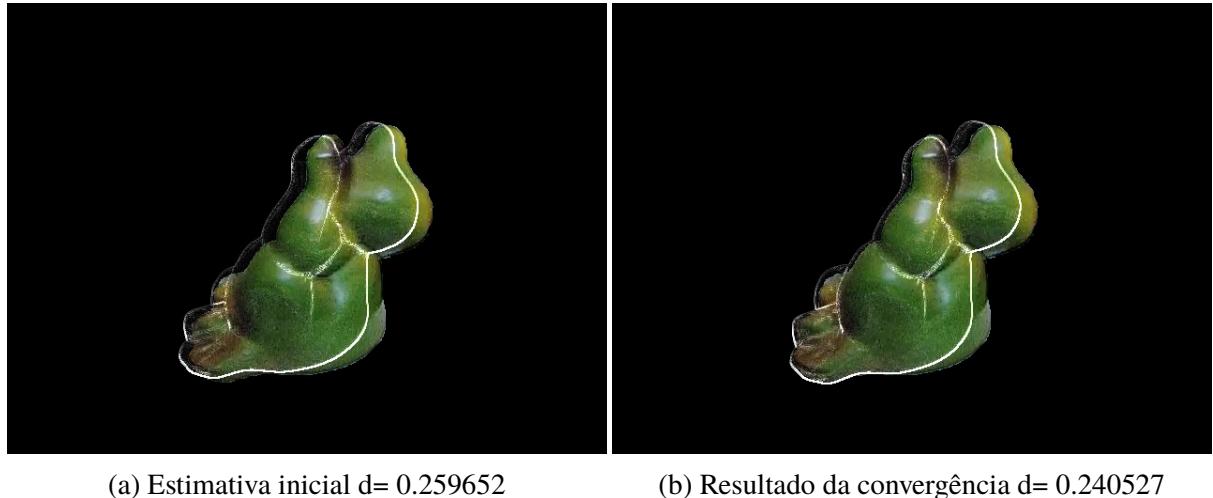
(a) Estimativa inicial  $d = 0.255856$ (b) Resultado da convergência  $d = 0.236007$ 

Figura 4.56: Imagem 36 do experimento 8, convergência de (a) para (b), do frame de vídeo 35

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.457679 & 0.317422 & 0.830526 & 0 \\ 0.0365704 & -0.989425 & 0.140357 & 0 \\ -0.866296 & -0.0946109 & -0.441231 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.313931 & 0.322127 & 0.89313 & 0 \\ 0.0424926 & -0.989313 & 0.139482 & 0 \\ -0.928516 & -0.081739 & -0.296888 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.58: Parâmetros iniciais e finais da figura 4.56



(a) Estimativa inicial d= 0.259652

(b) Resultado da convergência d= 0.240527

Figura 4.57: Imagem 37 do experimento 8, convergência de (a) para (b), do frame de vídeo 36

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.313931 & 0.322127 & 0.89313 & 0 \\ 0.0424926 & -0.989313 & 0.139482 & 0 \\ -0.928516 & -0.081739 & -0.296888 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.226276 & 0.35981 & 0.905172 & 0 \\ 0.0368311 & -0.983631 & 0.176391 & 0 \\ -0.953822 & -0.0732514 & -0.209319 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.59: Parâmetros iniciais e finais da figura 4.57

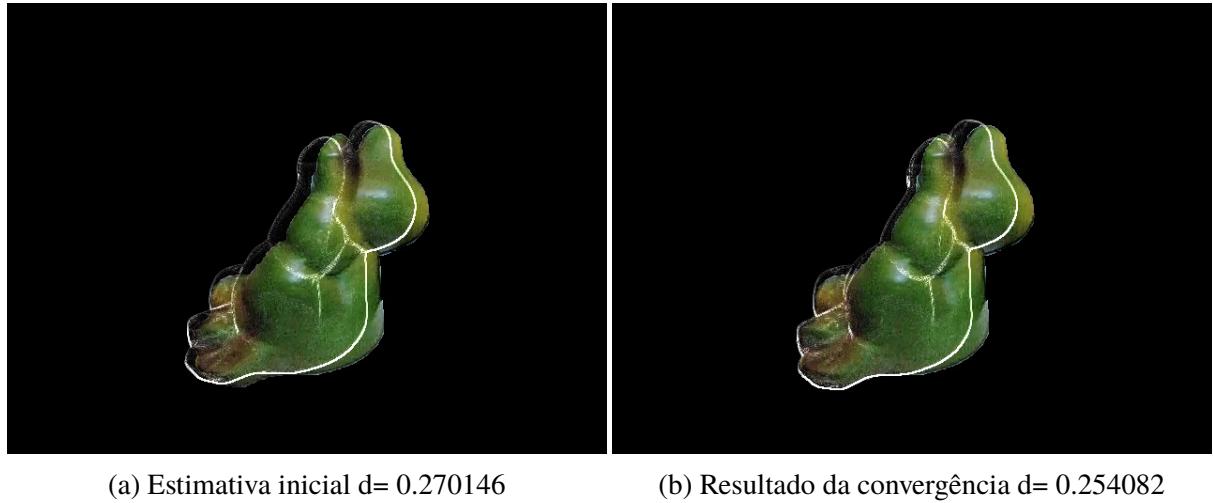


Figura 4.58: Imagem 38 do experimento 8, convergência de (a) para (b), do frame de vídeo 37

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.226276 & 0.35981 & 0.905172 & 0 \\ 0.0368311 & -0.983631 & 0.176391 & 0 \\ -0.953822 & -0.0732514 & -0.209319 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.127323 & 0.390921 & 0.911576 & 0 \\ 0.028454 & -0.978754 & 0.203053 & 0 \\ -0.971586 & -0.0517912 & -0.113494 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.60: Parâmetros iniciais e finais da figura 4.58

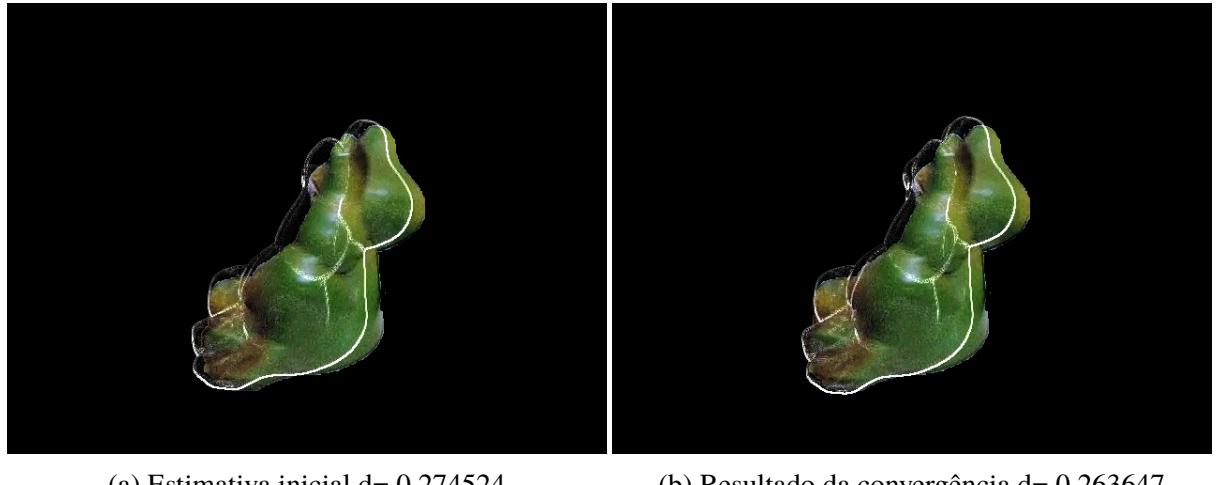
(a) Estimativa inicial  $d = 0.274524$ (b) Resultado da convergência  $d = 0.263647$ 

Figura 4.59: Imagem 39 do experimento 8, convergência de (a) para (b), do frame de vídeo 38

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.127323 & 0.390921 & 0.911576 & 0 \\ 0.028454 & -0.978754 & 0.203053 & 0 \\ -0.971586 & -0.0517912 & -0.113494 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} -0.074498 & 0.433939 & 0.897857 & 0 \\ 0.00908089 & -0.969136 & 0.246358 & 0 \\ -0.97705 & -0.0265065 & -0.0682582 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.61: Parâmetros iniciais e finais da figura 4.59

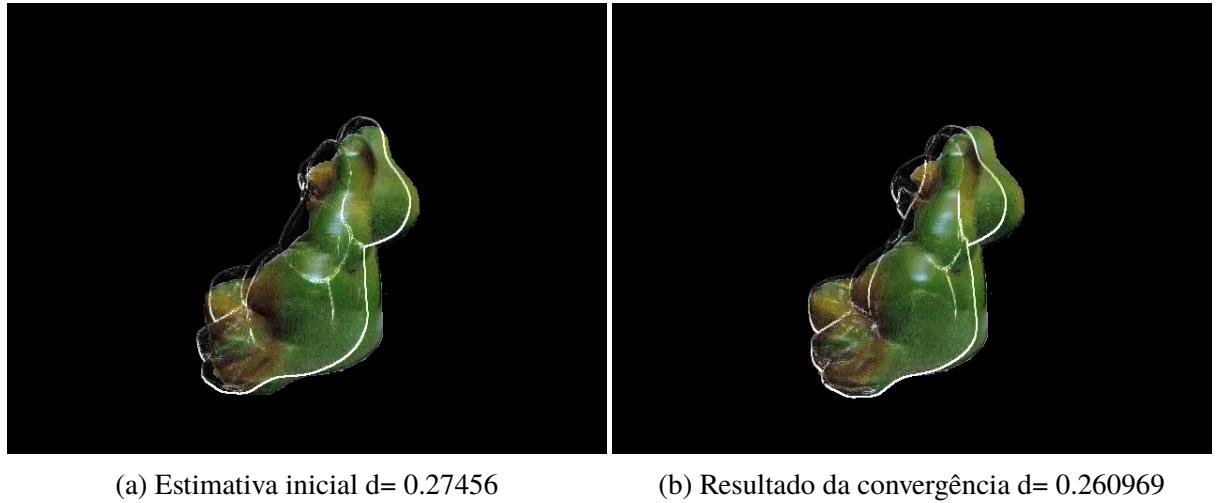


Figura 4.60: Imagem 40 do experimento 8, convergência de (a) para (b), do frame de vídeo 39

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} -0.074498 & 0.433939 & 0.897857 & 0 \\ 0.00908089 & -0.969136 & 0.246358 & 0 \\ -0.97705 & -0.0265065 & -0.0682582 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.132788 & 0.292759 & 0.946921 & 0 \\ -0.00716455 & -0.994066 & 0.108546 & 0 \\ -0.97308 & 0.0211979 & 0.129903 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.62: Parâmetros iniciais e finais da figura 4.60

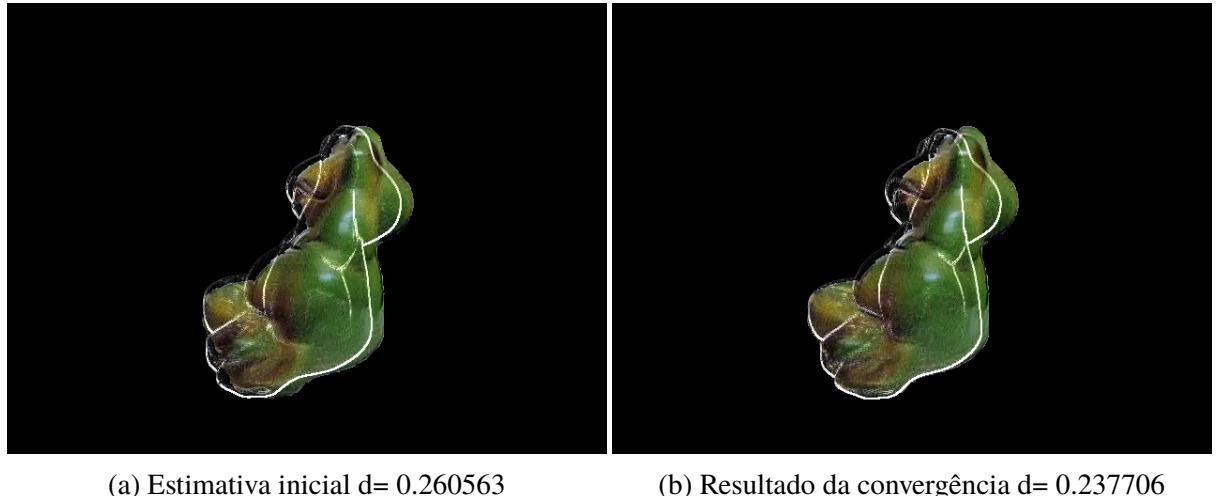
(a) Estimativa inicial  $d = 0.260563$ (b) Resultado da convergência  $d = 0.237706$ 

Figura 4.61: Imagem 41 do experimento 8, convergência de (a) para (b), do frame de vídeo 40

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.132788 & 0.292759 & 0.946921 & 0 \\ -0.00716455 & -0.994066 & 0.108546 & 0 \\ -0.97308 & 0.0211979 & 0.129903 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.226081 & 0.240948 & 0.943839 & 0 \\ -0.0392471 & -0.99805 & 0.0485403 & 0 \\ -0.953694 & 0.0480169 & 0.216183 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.63: Parâmetros iniciais e finais da figura 4.61

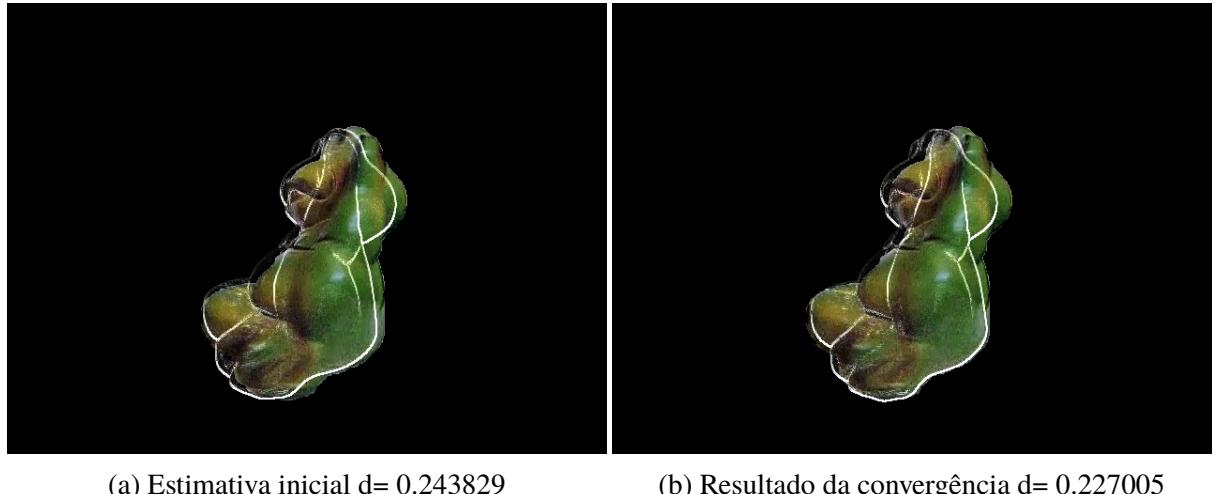
(a) Estimativa inicial  $d = 0.243829$ (b) Resultado da convergência  $d = 0.227005$ 

Figura 4.62: Imagem 42 do experimento 8, convergência de (a) para (b), do frame de vídeo 41

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.226081 & 0.240948 & 0.943839 & 0 \\ -0.0392471 & -0.99805 & 0.0485403 & 0 \\ -0.953694 & 0.0480169 & 0.216183 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.292782 & 0.219691 & 0.930599 & 0 \\ -0.0773728 & -0.996439 & 0.033516 & 0 \\ -0.934648 & 0.081816 & 0.274741 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.64: Parâmetros iniciais e finais da figura 4.62

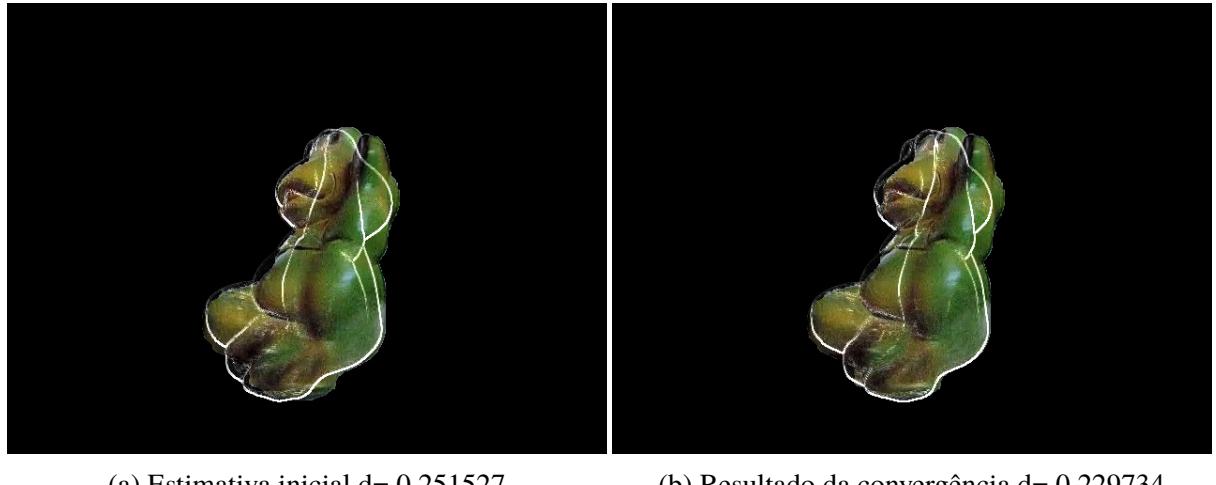
(a) Estimativa inicial  $d= 0.251527$ (b) Resultado da convergência  $d= 0.229734$ 

Figura 4.63: Imagem 43 do experimento 8, convergência de (a) para (b), do frame de vídeo 42

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.292782 & 0.219691 & 0.930599 & 0 \\ -0.0773728 & -0.996439 & 0.033516 & 0 \\ -0.934648 & 0.081816 & 0.274741 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.402575 & 0.215484 & 0.889663 & 0 \\ -0.0769799 & -0.996255 & 0.0393749 & 0 \\ -0.894816 & 0.0843375 & 0.384479 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.65: Parâmetros iniciais e finais da figura 4.63

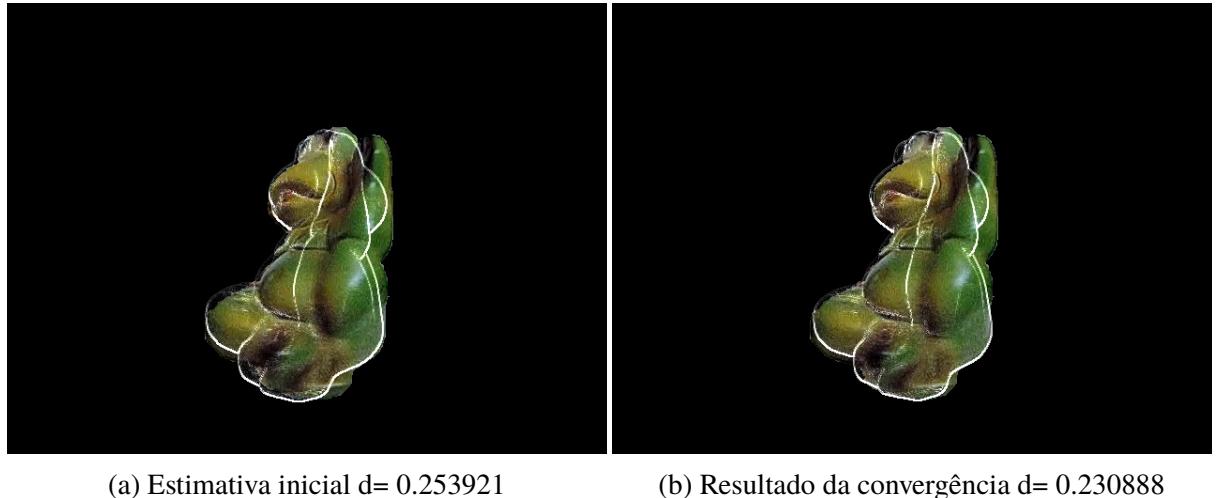


Figura 4.64: Imagem 44 do experimento 8, convergência de (a) para (b), do frame de vídeo 43

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.402575 & 0.215484 & 0.889663 & 0 \\ -0.0769799 & -0.996255 & 0.0393749 & 0 \\ -0.894816 & 0.0843375 & 0.384479 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.460234 & 0.187493 & 0.867774 & 0 \\ -0.104946 & -0.99414 & 0.0259122 & 0 \\ -0.867547 & 0.102995 & 0.43786 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.66: Parâmetros iniciais e finais da figura 4.64

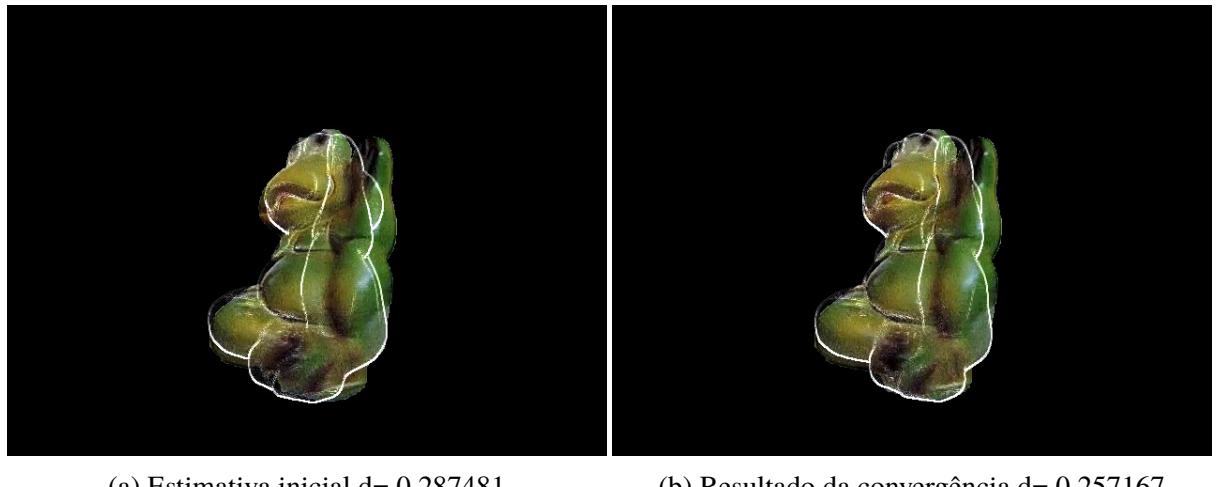
(a) Estimativa inicial  $d = 0.287481$ (b) Resultado da convergência  $d = 0.257167$ 

Figura 4.65: Imagem 45 do experimento 8, convergência de (a) para (b), do frame de vídeo 44

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.460234 & 0.187493 & 0.867774 & 0 \\ -0.104946 & -0.99414 & 0.0259122 & 0 \\ -0.867547 & 0.102995 & 0.43786 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.575936 & 0.194206 & 0.794092 & 0 \\ -0.0882905 & -0.995165 & 0.0430364 & 0 \\ -0.79861 & 0.094897 & 0.556004 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.67: Parâmetros iniciais e finais da figura 4.65

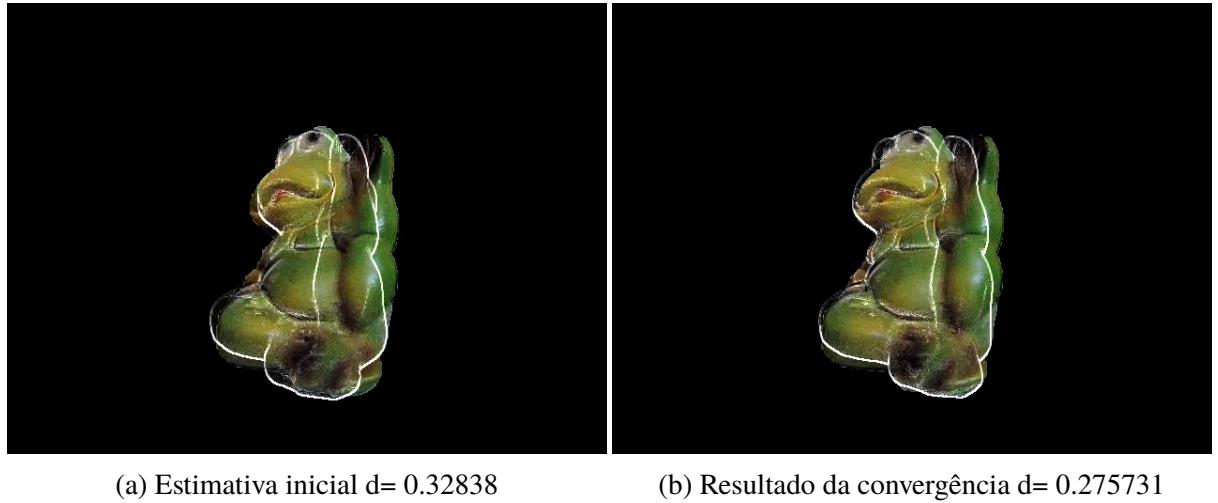


Figura 4.66: Imagem 46 do experimento 8, convergência de (a) para (b), do frame de vídeo 45

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.575936 & 0.194206 & 0.794092 & 0 \\ -0.0882905 & -0.995165 & 0.0430364 & 0 \\ -0.79861 & 0.094897 & 0.556004 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.697028 & 0.210032 & 0.685594 & 0 \\ -0.0578128 & -0.995939 & 0.0690091 & 0 \\ -0.697304 & 0.0877374 & 0.682055 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.68: Parâmetros iniciais e finais da figura 4.66

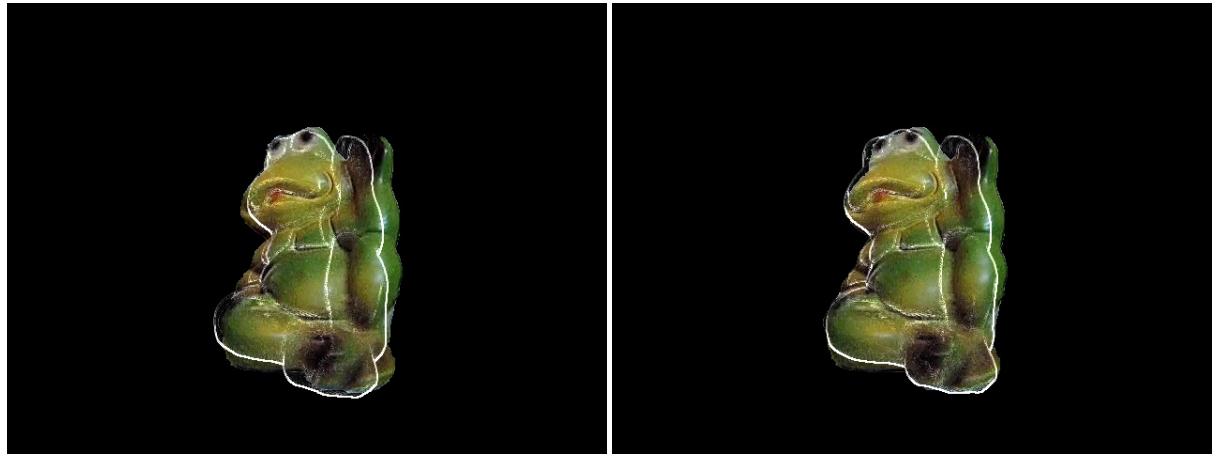
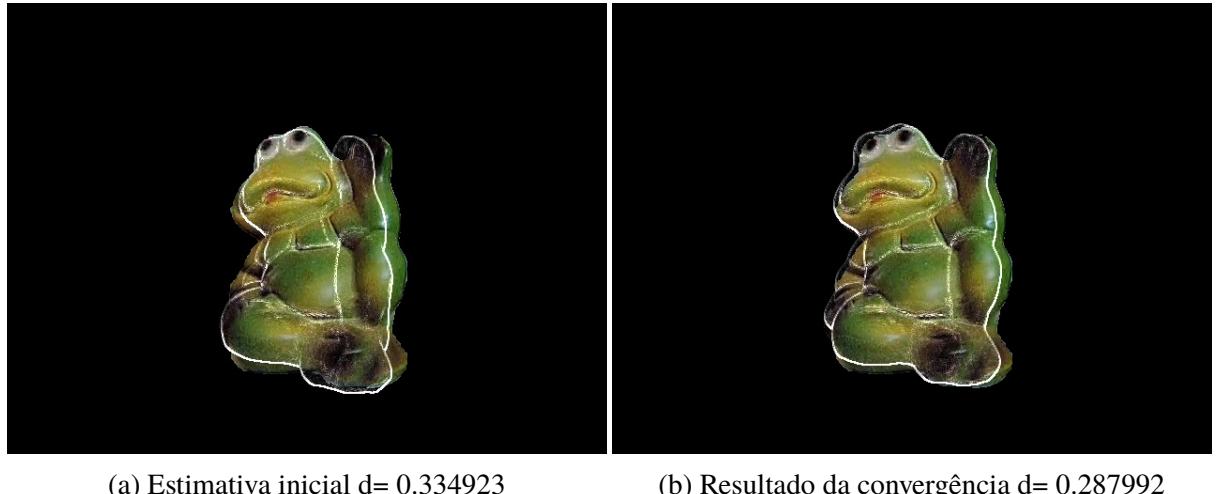


Figura 4.67: Imagem 47 do experimento 8, convergência de (a) para (b), do frame de vídeo 46

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.697028 & 0.210032 & 0.685594 & 0 \\ -0.0578128 & -0.995939 & 0.0690091 & 0 \\ -0.697304 & 0.0877374 & 0.682055 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.780653 & 0.195934 & 0.593457 & 0 \\ -0.0562467 & -0.99316 & 0.102323 & 0 \\ -0.609446 & 0.113259 & 0.764292 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.69: Parâmetros iniciais e finais da figura 4.67



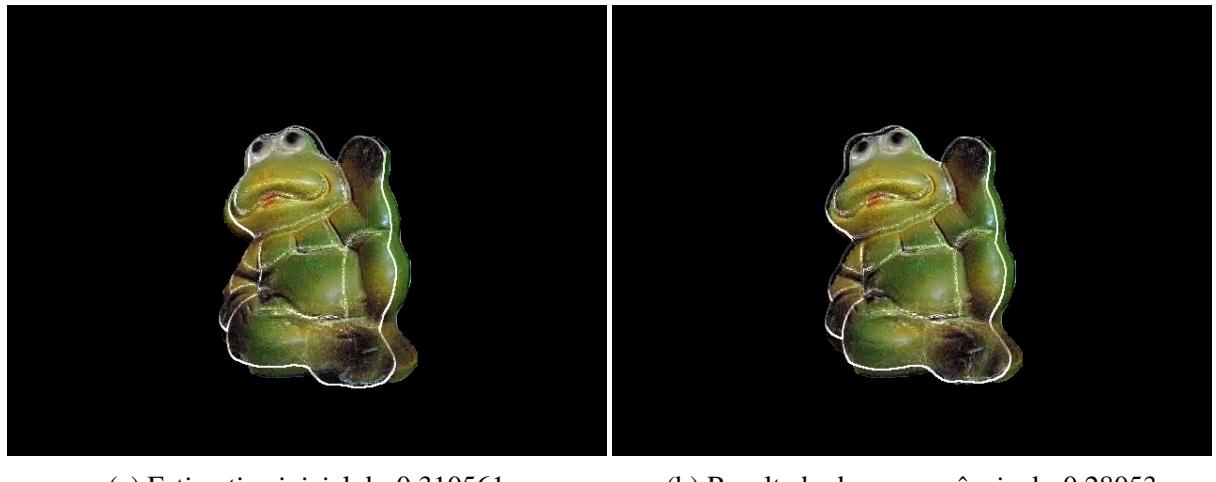
(a) Estimativa inicial d= 0.334923

(b) Resultado da convergência d= 0.287992

Figura 4.68: Imagem 48 do experimento 8, convergência de (a) para (b), do frame de vídeo 47

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.780653 & 0.195934 & 0.593457 & 0 \\ -0.0562467 & -0.99316 & 0.102323 & 0 \\ -0.609446 & 0.113259 & 0.764292 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.87885 & 0.183839 & 0.440257 & 0 \\ 0.0438839 & -0.966757 & 0.251901 & 0 \\ -0.471931 & 0.202063 & 0.857702 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.70: Parâmetros iniciais e finais da figura 4.68



(a) Estimativa inicial d= 0.310561

(b) Resultado da convergência d= 0.28053

Figura 4.69: Imagem 49 do experimento 8, convergência de (a) para (b), do frame de vídeo 48

Inicial	Final
$near = 1.2$	$near = 1.2$
$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$R = \begin{bmatrix} 0.87885 & 0.183839 & 0.440257 & 0 \\ 0.0438839 & -0.966757 & 0.251901 & 0 \\ -0.471931 & 0.202063 & 0.857702 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$R = \begin{bmatrix} 0.935267 & 0.16114 & 0.315135 & 0 \\ 0.0563865 & -0.96713 & 0.247951 & 0 \\ -0.344731 & 0.214131 & 0.913611 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$S = \begin{bmatrix} 45.0694 & 0 & 0 & 0 \\ 0 & 45.0694 & 0 & 0 \\ 0 & 0 & 45.0694 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
$t_x = 10$	$t_x = 10$
$t_y = 21$	$t_y = 21$

Tabela 4.71: Parâmetros iniciais e finais da figura 4.69

## 4.9 Conclusões sobre os experimentos

Aqui serão descritas conclusões baseadas na observação dos resultados obtidos.

Experimento 1: É possível observar uma diferença na rotação final a partir da segunda casa decimal, mostrando que a implementação da função de perda invariante a luz para o protótipo na verdade sofre sim variância com a luz entretanto é uma diferença pequena mostrando que a influência é menor do que seria usando uma distância euclidiana. Também é observável que a imagem 4 do experimento 1, figura 4.4, é a com menor distância, uma vez que tem menor contraste de luz sobre a superfície.

Experimento 2: É possível observar que a resolução da imagem influência no cálculo da distância, o que é condizente uma vez que a distância é calculada usando valores da matrizes de covariância e covariância cruzada, assim se a amostragem é maior o valor dessas matrizes são influenciados. Também é condizente que resoluções maiores exijam maior tempo de execução do que menores.

Experimento 3: É possível verificar que em poucas iterações a vantagem de repetir todos os passos se mostra insignificante. Observe também que a região sem mapeamento devido a segmentação da fotografia influência no valor da distância, impossibilitando um equiparó perfeito.

Experimento 4: É possível observar uma melhora em relação a estimativa inicial utilizando o otimizador *Downhill-Simplex*, neste caso onde é utilizado uma fotografia real sem oclusão da estatua.

Experimento 5: É possível observar uma particularidade do protótipo, como a malha é posicionada de ponta cabeça e posteriormente invertida no eixo Y, isso faz com a imagem renderizada  $M_\theta$  tenha o eixo Y invertido produzindo imagens com o canal G invertido também isso causa uma influência no cálculo da distância como constato pelo experimento. A inversão da escala de cinza, preservando o plano de fundo em 0, foi repetida em outros experimentos e apresentou comportamento diferente em cada caso. Assim é aconselhável rodar um caso de teste com e sem a inversão afim de observar mais dados.

Experimento 6: É possível observar a influência do ruído HSV, onde, na figura 4.16 sem ruído é obtido  $d_{final} - d_{inicial} = 0.218061$ , e, na figura 4.17 com ruído  $d_{final} - d_{inicial} = 0.181514$ . Também observamos que nesse caso inverter a escala de cinza, preservando o plano de fundo em 0, piorou a convergência.

Experimento 7: O experimento 7 demonstra um comportamento similar ao observado no experimento 6, onde apenas a matriz de rotação é desconhecida, o método converge para uma pose visualmente adequada, mesmo quando usando texturas sintéticas com nenhuma relação direta a normais da superfície, e com imagens e malhas diferentes da do experimento anterior.

Experimento 8: É perceptível o problema previamente discutido na seção 3.10, onde o protótipo só converge a rotação mantendo a escala e translação inicial. Isso ocorre pois a revolução do sapo vista no vídeo não ocorre em somente um eixo, fazendo com que em alguns frames ele esteja mais próximo à câmera e outros mais distantes, uma vez que o protótipo não corrige esses variações de translação e escala(devido a perspectiva), vários frames do experimento tem uma pose com alinhamento parcial. Mesmo assim é possível produzir uma textura única combinando a textura de cada frame. As regiões de realce são suavizadas pela combinação de texturas. Vale lembrar que o borramento de movimento também influencia a cor da textura final.

# Capítulo 5

## Conclusão

Neste trabalho foi discutido como obter texturas de objetos 3D, a partir de fotografias e quadros de vídeo em diferentes resoluções. O escopo dos objetos foi limitado a objetos não transparentes, não articuláveis com baixa refletância. Foi possível criar um protótipo que de fato realiza o objetivo proposto na introdução, gerar uma textura com mapeamento adequado ainda que com uma certa margem de erro, a partir de uma fotografia ou quadro de vídeo, e assim realizar diversos experimentos. Futuramente será disponibilizado um repositório com código do protótipo e a base necessária para replicar os experimentos realizados no capítulo 4.

### 5.1 Vantagens da abordagem escolhida

A comparação entre imagens 2D, permite fazer com o custo de utilizar malhas com alta complexidade (por exemplo com mais de 100 mil vértices) se limite à etapa de renderização, e o custo de calcular a distância entre as imagens seja em função meramente da resolução delas. O que é uma vantagem sobre algoritmos como o *softPOSIT* em que o número vértices da malha influem diretamente no custo da convergência.

### 5.2 Desvantagens da abordagem escolhida

Não assumir conhecimento prévio sobre a cena analisada, como foi o caso neste trabalho, dificulta na elaboração de heurísticas para estimar uma pose inicial, consequentemente inicializando o otimizador com valores "ruins" que comprometerão a convergência do mesmo.

### 5.3 Aplicação prática

Uma aplicação prática do trabalho aqui desenvolvido é a criação de uma capa de cobertura para o objeto original, previamente escaneado, onde a coloração dessa cobertura é a textura gerada pelo algoritmo que teve como base uma fotografia ou quadro de vídeo desse mesmo objeto. Para tal é preciso futuramente melhorar a precisão das poses e o mapeamento UV afim de que os *charts* possam ser recortados e costurados facilmente, e o máximo possível de costuras coincidam com bordas na forma do objeto assim sejam pouco visíveis.

## 5.4 Trabalhos Futuros

O problema aqui abordado ainda deixa em aberto diversas oportunidades para trabalhos futuros. Alguns deles são: explorar outras funções de distância e observar a convergência produzida; Criação de diferentes técnicas para tratar as classes de objetos não abordados por esse trabalho; Implementação da função de distância para executar na GPU; Incorporar elementos de aprendizado de máquina para obter *features* do objeto a partir da malha, para assim propor uma estimativa inicial mais próxima do mínimo global; Utilizar outros métodos otimizadores em conjunto ou em substituição do DownHill-Simplex; Propor uma melhor representação de pose, se possível eliminando graus de liberdade.

# Referências Bibliográficas

- [ALICE, 2017] ALICE (2017). Graphite. [http://alice.loria.fr/index.php?option=com\\_content&view=article&id=22](http://alice.loria.fr/index.php?option=com_content&view=article&id=22). Acessado em 5/5/2017.
- [Arie-Nachimson e Basri, 2009] Arie-Nachimson, M. e Basri, R. (2009). Constructing implicit 3d shape models for pose estimation. Em *2009 IEEE 12th International Conference on Computer Vision*, páginas 1341–1348.
- [ASTM, 1992] ASTM (1992). Standard guide for computed tomography (ct) imaging, astm designation e 1441 - 92a. Em *1992 Annual Book of ASTM Standards*, páginas 690–713.
- [Baker, 2012] Baker, D. (2012). Face distortion is not due to lens distortion. <https://bakerdh.wordpress.com/2012/05/05/face-distortion-is-not-due-to-lens-distortion/>. Acessado em 5/12/2017.
- [Blinn, 1978] Blinn, J. (1978). Simulation of wrinkled surfaces. Association for Computing Machinery, Inc.
- [Blinn e Newell, 1976] Blinn, J. F. e Newell, M. E. (1976). Texture and reflection in computer generated images. *Commun. ACM*, 19(10):542–547.
- [Chin e Dyer, 1986] Chin, R. T. e Dyer, C. R. (1986). Model-based recognition in robot vision. *ACM Comput. Surv.*, 18(1):67–108.
- [Cohen et al., 1998] Cohen, J., Olano, M. e Manocha, D. (1998). Appearance-preserving simplification. Em *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '98, páginas 115–122, New York, NY, USA. ACM.
- [Cook et al., 1987] Cook, R. L., Carpenter, L. e Catmull, E. (1987). The reyes image rendering architecture. Em *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '87, páginas 95–102, New York, NY, USA. ACM.
- [David et al., 2004] David, P., DeMenthon, D., Duraiswami, R. e Samet, H. (2004). Softposit: Simultaneous pose and correspondence determination. *International Journal of Computer Vision*, 59(3):259–284.
- [Deriche e Faugeras, 1990] Deriche, R. e Faugeras, O. (1990). Tracking line segments. *Image Vision Comput.*, 8(4):261–270.
- [Ezquerra e Mullick, 1996] Ezquerra, N. e Mullick, R. (1996). An approach to 3d pose determination. 15:99–120.
- [Forsyth e Ponce, 2003] Forsyth, D. A. e Ponce, J. (2003). *Computer Vision A Modern Approach*. Prentice Hall.

- [Fox, 2013] Fox, M. (2013). A practical guide to optical metrology for thin films, by michael quinten. *Contemporary Physics*, 54(5):255–255.
- [Gleason et al., 1979] Gleason, G., Agin, G., of Manufacturing Engineers, S., Group, S. R. I. A. I., International, S. e Institute, S. R. (1979). *A Modular Vision System for Sensor-controlled Manipulation and Inspection*. Creative manufacturing engineering program. Stanford Research Inst.
- [Horn, 1989] Horn, B. K. P. (1989). Shape from shading. capítulo: Obtaining Shape from Shading Information, páginas 123–171. MIT Press, Cambridge, MA, USA.
- [Huttenlocher e Ullman, 1990] Huttenlocher, D. P. e Ullman, S. (1990). Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, 5(2):195–212.
- [Hutter e Brewer, 2009] Hutter, M. e Brewer, N. (2009). Matching 2-d ellipses to 3-d circles with application to vehicle pose identification. Em *2009 24th International Conference Image and Vision Computing New Zealand*, páginas 153–158.
- [Jayawardena, 2013] Jayawardena, S. (2013). *Image based automatic vehicle damage detection*. Tese de doutorado, College of Engineering and Computer Science (CECS).
- [Jayawardena et al., 2013] Jayawardena, S., Hutter, M. e Brewer, N. (2013). A novel illumination-invariant loss for monocular 3d pose estimation. *CoRR*, abs/1311.7186.
- [Johnson, 1997] Johnson, A. E. (1997). *Spin-images: a representation for 3-D surface matching*. Tese de doutorado, Citeseer.
- [Kollnig e Nagel, 1997] Kollnig, H. e Nagel, H.-H. (1997). 3d pose estimation by directly matching polyhedral models to gray value gradients. *Int. J. Comput. Vision*, 23(3):283–302.
- [Lévy et al., 2002] Lévy, B., Petitjean, S., Ray, N. e Maillot, J. (2002). Least squares conformal maps for automatic texture atlas generation. *ACM Trans. Graph.*, 21(3):362–371.
- [Li et al., 2014] Li, E., Zhang, X. e Chen, Y. (2014). Sampling and surface reconstruction of large scale point cloud. Em *Proceedings of the 13th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, VRCAI ’14, páginas 35–41, New York, NY, USA. ACM.
- [Mahalanobis, 1936] Mahalanobis, P. C. (1936). On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta)*, 2:49–55.
- [Moreno-Noguer et al., 2008] Moreno-Noguer, F., Lepetit, V. e Fua, P. (2008). *Pose Priors for Simultaneously Solving Alignment and Correspondence*, páginas 405–418. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [Nelder e Mead, 1965] Nelder, J. A. e Mead, R. (1965). A Simplex Method for Function Minimization. *Comput. J.*, 7:308–313.
- [OpenGL, 2016] OpenGL (2016). Opengl. <https://www.opengl.org/>. Acessado em 30/10/2016.
- [Ozuyosal et al., 2009] Ozuyosal, M., Lepetit, V. e Fua, P. (2009). Pose estimation for category specific multiview object localization. Em *2009 IEEE Conference on Computer Vision and Pattern Recognition*, páginas 778–785.

- [Perkins, 1978] Perkins, W. A. (1978). A model-based vision system for industrial parts. *IEEE Transactions on Computers*, C-27(2):126–143.
- [Phong, 1975] Phong, B. T. (1975). Illumination for computer generated pictures. *Commun. ACM*, 18(6):311–317.
- [POJE e DELP, 1982] POJE, J. F. e DELP, E. J. (1982). A review of techniques for obtaining depth information with applications to machine vision. Relatório técnico, Center for Robotics and Integrated Manufacturing, Univ. of Michigan, Ann Arbor.
- [Ray e Levy, 2003] Ray, N. e Levy, B. (2003). Hierarchical least squares conformal map. Em *11th Pacific Conference on Computer Graphics and Applications, 2003. Proceedings.*, páginas 263–270.
- [Rodney A. Brooks, 1981] Rodney A. Brooks, T. O. B. (1981). Geometric modeling in vision for manufacturing.
- [Rother et al., 2004] Rother, C., Kolmogorov, V. e Blake, A. (2004). "grabcut": Interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314.
- [Sheffer e de Sturler, 2001] Sheffer, A. e de Sturler, E. (2001). Parameterization of faceted surfaces for meshing using angle-based flattening. *Engineering with Computers*, 17(3):326–337.
- [Sheffer et al., 2005] Sheffer, A., Lévy, B., Mogilnitsky, M. e Bogomyakov, A. (2005). Abf++: Fast and robust angle based flattening. *ACM Trans. Graph.*, 24(2):311–330.
- [Shi e Ulrich, 2016] Shi, J. F. e Ulrich, S. (2016). Softposit enhancements for monocular camera spacecraft pose estimation. Em *2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR)*, páginas 30–35.
- [Shirley e Marschner, 2009] Shirley, P. e Marschner, S. (2009). *Fundamentals of Computer Graphics*. A. K. Peters, Ltd., Natick, MA, USA, 3rd edition.
- [Snavely et al., 2008] Snavely, N., Seitz, S. M. e Szeliski, R. (2008). Modeling the world from internet photo collections. *Int. J. Comput. Vision*, 80(2):189–210.
- [Su et al., 2016] Su, T., Wang, W., Lv, Z., Wu, W. e Li, X. (2016). Rapid delaunay triangulation for randomly distributed point cloud data using adaptive hilbert curve. *Comput. Graph.*, 54(C):65–74.
- [Sun et al., 2010] Sun, M., Bradski, G., Xu, B.-X. e Savarese, S. (2010). Depth-encoded hough voting for joint object detection and shape recovery. Em *Proceedings of the 11th European Conference on Computer Vision: Part V*, ECCV'10, páginas 658–671, Berlin, Heidelberg. Springer-Verlag.
- [Tan e Baker, 2000] Tan, T. N. e Baker, K. D. (2000). Efficient image gradient based vehicle localization. *IEEE Transactions on Image Processing*, 9(8):1343–1356.
- [Tarini, 2016] Tarini, M. (2016). Volume-encoded uv-maps. *ACM Trans. Graph.*, 35(4):107:1–107:13.
- [UFPR, 2016] UFPR, I. (2016). Imago ufpr. <http://www.imago.ufpr.br/3dexhibition/>. Acessado em 30/10/2016.

- [Vallet e Lévy, 2009] Vallet, B. e Lévy, B. (2009). What you seam is what you get. Relatório técnico, INRIA - ALICE Project Team.
- [Velho e Sossai Jr, 2017] Velho, L. e Sossai Jr, J. (2017). Projective texture atlas construction and applications.
- [Woodham, 1980] Woodham, R. (1980). Photometric method for determining surface orientation from multiple images. 19:139–144.
- [Xing et al., 2003] Xing, E. P., Jordan, M. I., Russell, S. J. e Ng, A. Y. (2003). Distance metric learning with application to clustering with side-information. Em Becker, S., Thrun, S. e Obermayer, K., editores, *Advances in Neural Information Processing Systems 15*, páginas 521–528. MIT Press.
- [Yachida e Tsuji, 1977] Yachida, M. e Tsuji, S. (1977). A versatile machine vision system for complex industrial parts. *IEEE Transactions on Computers*, C-26(9):882–894.
- [Yang et al., 2016] Yang, J., Li, H., Campbell, D. e Jia, Y. (2016). Go-icp: A globally optimal solution to 3d icp point-set registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11):2241–2254.