



Pós-Graduação em Ciência da Computação

**“Realidade Aumentada sem Marcadores a partir
de Rastreamento Baseado em Textura – Uma
abordagem Baseada em Pontos de Interesse e
Filtro de Partículas”**

Por

Francisco Paulo Magalhães Simões

Dissertação de Mestrado



Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

RECIFE, MARÇO/2011



UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO



FRANCISCO PAULO MAGALHÃES SIMÕES

FPMS@CIN.UFPE.BR

**REALIDADE AUMENTADA SEM MARCADORES A PARTIR DE
RASTREAMENTO BASEADO EM TEXTURA – UMA ABORDAGEM BASEADA
EM PONTOS DE INTERESSE E FILTRO DE PARTÍCULAS**

DISSERTAÇÃO SUBMETIDA AO CENTRO DE INFORMÁTICA DA
UNIVERSIDADE FEDERAL DE PERNAMBUCO, COMO REQUISITO
PARCIAL PARA OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIA DA
COMPUTAÇÃO.

ORIENTADORA: VERONICA TEICHRIEB (VT@CIN.UFPE.BR)

RECIFE, MARÇO DE 2011

Agradecimentos

Definitivamente agradecer não é uma tarefa simples. Não pela dificuldade de reconhecer aqueles que nos ajudam ao longo do caminho, mas pela falta de expressividade das palavras para representar sentimentos tão fortes.

Primeiramente muito obrigado a Deus por me dar forças para seguir meu caminho a cada dificuldade surgida. Obrigado a minha família por ser um alicerce forte tanto nos momentos fáceis quanto nos momentos difíceis. Sem eles o caminho seria 1000 vezes mais árduo e incerto. Obrigado aos meus amigos e minha namorada pelo apoio e por deixar meus dias mais divertidos. Aos companheiros de trabalho, os quais muitos se tornaram verdadeiros amigos e exemplos, pelo suporte, companheirismo e brilhantismo no dia-a-dia. Um obrigado especial a minha orientadora (Prof. Veronica, vulgo vt) pela confiança e ajuda ao longo de todo este trabalho. Além de uma brilhante orientadora acabou se tornando uma grande amiga.

A todos que de alguma forma passaram pelo caminho, seja com uma mão amiga ou com uma palavra de conforto muito obrigado.

Quem distribui sorrisos colhe carinho por onde passa.

Sorria, distribua palavras de carinho e aproveite cada dia para tornar o mundo um lugar melhor!

Francisco Paulo Magalhães Simões

Abstract

Augmented Reality techniques rely on recovery of virtual camera information in each frame of an image sequence to correctly associate 3D information to real scenes, a process usually known as 3D tracking. In this context, this dissertation introduces a texture based 3D tracking technique from the classical Interest Point Based (IPB) technique in association to a Particle Filter approach. It was developed to enable a robust 3D tracking of textured objects. The algorithm has a promising performance although it still needs some optimizations to achieve real time constraints. Three reference techniques were implemented and analyzed (Spatial IPB, Temporal IPB and Spatial Temporal IPB), and performance and robustness analysis will show the next steps to do. Tests with synthetic and real data were done, and the IPB variations as well as the Particle Filter approach have achieved good results.

Keywords: 3D tracking, particle filters, marker-less augmented reality, texture based, interest point based tracking, computer vision.

Resumo

Técnicas de realidade aumentada dependem da recuperação de informações da câmera virtual a cada quadro de uma sequência de imagens para corretamente associar informações 3D às cenas reais, comumente conhecida como rastreamento 3D. Neste contexto, esta dissertação introduz uma técnica de rastreamento 3D baseado em textura através de uma variação da técnica clássica *Interest Point Based* (IPB) associada à abordagem de filtro de partículas. Ela foi desenvolvida para possibilitar o rastreamento robusto de objetos texturizados. O algoritmo possui desempenho promissor apesar de ainda precisar de otimizações para atingir restrições de tempo-real podendo ser usado em diferentes cenários de realidade aumentada. Três implementações de referência da técnica IPB foram implementadas e analisadas (Spatial IPB, Temporal IPB e Spatial *Temporal* IPB), e comparações de desempenho e precisão apontaram os caminhos a seguir. Foram realizados testes com casos sintéticos e reais, e as variações do IPB bem como a abordagem usando Filtro de Partículas obtiveram boa precisão.

Palavras-chave: rastreamento 3D baseado em textura, filtros de partículas, realidade aumentada sem marcadores, rastreamento baseado em pontos de interesse, visão computacional.

Conteúdo

1 INTRODUÇÃO 13

1.1	DEFINIÇÃO DO PROBLEMA.....	14
1.2	OBJETIVOS DA PESQUISA	14
1.3	ORGANIZAÇÃO DA DISSERTAÇÃO	15

2 CONCEITOS MATEMÁTICOS E ESTATÍSTICOS 16

2.1	REPRESENTAÇÃO DA CÂMERA	16
2.2	CÁLCULO DE POSE <i>BOTTOM-UP</i>	19
2.2.1	PERSPECTIVE- <i>n</i> -POINT	19
2.2.2	MINIMIZAÇÃO DO ERRO DE REPROJEÇÃO	21
2.2.3	CÁLCULO ROBUSTO.....	21
2.3	CÁLCULO DE POSE <i>Top-Down</i>	25
2.3.1	FILTRO DE PARTÍCULAS	25

3 REALIDADE AUMENTADA SEM MARCADORES 27

3.1	TAXONOMIA	27
3.1.1	RASTREAMENTO BASEADO EM MODELOS.....	28
3.1.2	ANÁLISE DAS TÉCNICAS BASEADAS EM MODELOS	30
3.1.3	RASTREAMENTO BASEADO EM RECONSTRUÇÃO 3D	33

4 RASTREAMENTO BASEADO EM TEXTURA 37

4.1	INTEREST POINT BASED	37
4.2	SPATIAL IPB	38
4.2.1	GERAÇÃO DE <i>KEYFRAMES</i>	39
4.2.2	ESCOLHA DE <i>KEYFRAME</i> MAIS PRÓXIMO	42
4.2.3	CÁLCULO DE <i>KEYFRAME</i> INTERMEDIÁRIO	44
4.2.4	EXTRAÇÃO DE CARACTERÍSTICAS.....	45
4.2.5	CASAMENTO DE CARACTERÍSTICAS.....	46

4.2.6	CÁLCULO DE POSE	47
4.2.7	ATUALIZAÇÃO DO QUADRO ATUAL	48
4.3	TEMPORAL IPB.....	49
4.4	SPATIAL TEMPORAL IPB	50
4.4.1	CÁLCULO DE POSE	52
4.5	IPB + FILTRO DE PARTÍCULAS.....	53
4.5.1	PARTÍCULAS.....	55
4.5.2	EXPANSÃO DAS PARTÍCULAS	55
4.5.3	ANÁLISE DAS PARTÍCULAS (CÁLCULO DE VEROSIMILHANÇA)	56
4.5.4	OTIMIZAÇÃO DAS MELHORES PARTÍCULAS	58
4.5.5	NORMALIZAÇÃO DO CONJUNTO DE PARTÍCULAS (ATUALIZAÇÃO DOS PESOS).....	58
4.5.6	CÁLCULO DE POSE	59
4.5.7	REAMOSTRAGEM.....	60

5 RESULTADOS 61

5.1	FERRAMENTAS E MÉTODOS	61
5.2	RASTREAMENTO UTILIZANDO SPATIAL IPB.....	64
5.3	RASTREAMENTO UTILIZANDO TEMPORAL IPB	67
5.4	RASTREAMENTO UTILIZANDO SPATIAL TEMPORAL IPB.....	69
5.4.1	RASTREAMENTO UTILIZANDO SPATIAL TEMPORAL IPB + RANSAC	72
5.5	RASTREAMENTO UTILIZANDO IPB + FILTRO DE PARTÍCULAS.....	74
5.5.1	FERRAMENTA PARA ANÁLISE DE PARTÍCULAS.....	77

6 CONCLUSÕES 80

6.1	CONTRIBUIÇÕES GERAIS	81
6.2	CONTRIBUIÇÕES PARA A COMUNIDADE CIENTÍFICA (PUBLICAÇÕES)	81
6.2.1	CAPÍTULOS DE LIVROS	81
6.2.2	ARTIGO COMPLETO (CONFERÊNCIA)	82
6.2.3	ARTIGO COMPLETO (PERIÓDICO)	82
6.2.4	MINI-CURSOS	82
6.3	TRABALHOS FUTUROS	82

Lista de Figuras

FIGURA 1-1 APLICAÇÃO DE RA À TRANSMISSÃO ESPORTIVA (POSIÇÃO, NOME E VELOCIDADE DOS PILOTOS).....	13
FIGURA 2-1 O OBJETO 3D, SUA PROJEÇÃO NO PLANO DE IMAGEM E A RELAÇÃO ENTRE OS SISTEMAS DE COORDENADAS DE MUNDO (WX, WY, WZ) E DE CÂMERA (CX, CY, CZ).	17
FIGURA 2-2 PROJEÇÃO EM PERSPECTIVA DE PONTOS 3D EM PONTOS DE IMAGEM.....	18
FIGURA 2-3 PASSO-A-PASSO DO ALGORITMO RANSAC: (A) AMOSTRAGEM ALEATÓRIA DAS CORRESPONDÊNCIAS E CÁLCULO DE POSE; (B) AVALIAÇÃO DA POSE; (C) CLASSIFICAÇÃO DAS CORRESPONDÊNCIAS EM <i>INLIERS</i> E <i>OUTLIERS</i> ; (D) CÁLCULO DA PORCENTAGEM DE <i>INLIERS</i>	23
2-4 FUNÇÕES DO M-ESTIMADOR DE HUBER COM $C = 1$ (ESQUERDA) E DO M-ESTIMADOR DE TUKEY COM $C = 4$ (DIREITA),	25
FIGURA 3-1 TAXONOMIA DE TÉCNICAS DE RASTREAMENTO BASEADO EM MODELOS PARA MAR.	28
FIGURA 3-2 EXEMPLOS DE RASTREAMENTO RECURSIVO: AMOSTRAGEM DE PONTOS (ACIMA E À ESQUERDA), EXTRAÇÃO EXPLÍCITA DAS ARESTAS (ACIMA E À DIREITA), CASAMENTO DE PADRÕES (ABAIXO E À ESQUERDA), PONTOS DE INTERESSE (ABAIXO E AO CENTRO) E TÉCNICA HÍBRIDA (ABAIXO E À DIREITA).	29
FIGURA 3-3 EXEMPLOS DE RASTREAMENTO POR DETECÇÃO: VISTAS (ESQUERDA) E PONTOS-CHAVE (DIREITA).	30
FIGURA 3-4 RASTREAMENTO BASEADO EM RECONSTRUÇÃO 3D UTILIZANDO SFM DE TEMPO REAL (DAVID NISTÉR, 2003). OBJETOS RECONSTRUÍDOS NO CENTRO, CÂMERAS VISUALIZADAS COMO QUADRILÁTEROS (PLANO DE VISÃO) TRANSPASSADOS PELA LINHA QUE LIGA O CENTRO DA CÂMERA AO PLANO DE VISÃO EM TORNO DOS OBJETOS.	33
FIGURA 3-5 APLICAÇÃO DE RA COM SIMULAÇÃO FÍSICA A PARTIR DE RASTREAMENTO UTILIZANDO SFM DE TEMPO REAL (NEWCOMBE & A. J. DAVISON, 2010).	34
FIGURA 3-6 INICIALIZAÇÃO PARA RASTREAMENTO UTILIZANDO SLAM ATRAVÉS DE OBJETO CONHECIDO (OBJETO VERDE À ESQUERDA) E CENA AUMENTADA COM QUATRO OBJETOS VIRTUAIS AO LONGO DO RASTREAMENTO (DIREITA).	35
FIGURA 3-7 REALIDADE AUMENTADA EM DISPOSITIVOS MÓVEIS UTILIZANDO O <i>PARALLEL TRACKING AND MAPPING SYSTEM</i> (PTAM).	36
FIGURA 4-1 FLUXO DE EXECUÇÃO DA TÉCNICA SPATIAL IPB. O FUNCIONAMENTO SE BASEIA NA AQUISIÇÃO DE INFORMAÇÕES OFFLINE SOB A FORMA DE KEYFRAMES. NO INÍCIO DO RASTREAMENTO É REALIZADA A EXTRAÇÃO DE CARACTERÍSTICAS DO QUADRO ATUAL (PASSO 1). EM SEGUIDA, É FEITA A ESCOLHA DO KEYFRAME MAIS PRÓXIMO DO QUADRO ANTERIOR (PASSO 2), TAMBÉM CONHECIDO NO INÍCIO DO RASTREAMENTO, E EM SEGUIDA É FEITO O CÁLCULO DO KEYFRAME INTERMEDIÁRIO (PASSO 3), APROXIMANDO O KEYFRAME DO QUADRO ANTERIOR. APÓS REALIZADOS OS PASSOS ANTERIORES (PODEM SER FEITOS EM PARALELO, 1 EM PARALELO COM 2 E 3), É REALIZADO O CASAMENTO DAS CARACTERÍSTICAS EXTRAÍDAS DO QUADRO ATUAL E DO KEYFRAME INTERMEDIÁRIO (PASSO 4). EM SEGUIDA, É FEITO O CÁLCULO DA POSE ATRAVÉS DA MINIMIZAÇÃO DO ERRO DE REPROJEÇÃO (PASSO 5). APÓS O CÁLCULO DA POSE O RASTREAMENTO É CONCLUÍDO AUMENTANDO A CENA COM A INFORMAÇÃO DESEJADA (PASSO 6) E É FEITA A ATUALIZAÇÃO DOS DADOS DO QUADRO ATUAL (PASSO 7) PARA UTILIZAÇÃO NO PRÓXIMO QUADRO POR SE TRATAR DE UMA TÉCNICA RECURSIVA.	39
FIGURA 4-2 KEYFRAME GERADO OFFLINE CONTENDO A IMAGEM DO QUADRO CAPTURADO (I), AS CARACTERÍSTICAS EXTRAÍDAS SOB A FORMA DE PONTOS 2D DA IMAGEM (II), OS PONTOS 3D DO MODELO CORRESPONDENTES (III) E AS INFORMAÇÕES DA CÂMERA (IV).	40

FIGURA 4-3 ALGORITMO FACET-ID: CADA FACETA DO OBJETO POSSUI UMA COR ÚNICA E, A PARTIR DA AMOSTRAGEM DO PONTO 2D (VERMELHO), PODE-SE ENCONTRAR O PONTO 3D CORRESPONDENTE (VERMELHO) ATRAVÉS DO RAIO GERADO A PARTIR DA CÂMERA ($K[R T]$) E SUA INTERSEÇÃO COM A FACETA CORRESPONDENTE.....	41
FIGURA 4-4 APENAS ALGUNS KEYFRAMES SÃO NECESSÁRIOS PARA REALIZAR O RASTREAMENTO. FIGURA EXTRAÍDA DE (VACCHETTI, V. LEPESTIT, & P. FUA, 2004B).....	42
FIGURA 4-5 HISTOGRAMAS GERADOS A PARTIR DE DIFERENTES POSES DO MESMO OBJETO. É IMPORTANTE NOTAR A SEMELHANÇA ENTRE OS HISTOGRAMAS DE POSES MAIS PRÓXIMAS (I E II) E SUA DIFERENÇA PARA O HISTOGRAMA (III). FIGURA EXTRAÍDA DE (VACCHETTI, V. LEPESTIT, & P. FUA, 2003).....	43
FIGURA 4-6 KEYFRAME INTERMEDIÁRIO (DIREITA) GERADO A PARTIR DA HOMOGRAFIA H APLICADA A PATCHES DO KEYFRAME MAIS PRÓXIMO (ESQUERDA). FIGURA EXTRAÍDA DE (VACCHETTI, V. LEPESTIT, & P. FUA, 2004B).....	45
FIGURA 4-7 PONTOS DE INTERESSE EXTRAÍDOS (DIREITA) DE QUADRO A SER RASTREADO (ESQUERDA) ATRAVÉS DA UTILIZAÇÃO DO DETECTOR DE CANTOS DE HARRIS.....	46
FIGURA 4-8 CASAMENTO DE CARACTERÍSTICAS. PONTOS CASADOS SÃO CONSIDERADOS VÁLIDOS QUANDO MESMO CASAMENTO OCORRE TANTO NO SENTIDO $I_2 \rightarrow I_1$ QUANTO NO SENTIDO $I_1 \rightarrow I_2$. NO EXEMPLO, O ÚNICO CASAMENTO VÁLIDO É O PAR a, α .47	47
FIGURA 4-9 CÁLCULO DE POSE DO QUADRO ATUAL (P_t) REALIZADO A PARTIR DO CASAMENTO DE CARACTERÍSTICAS DO QUADRO ATUAL COM O KEYFRAME INTERMEDIÁRIO.	48
FIGURA 4-10 RASTREAMENTO UTILIZANDO TEMPORAL IPB. PRIMEIRAMENTE SÃO EXTRAÍDAS AS CARACTERÍSTICAS DO QUADRO ATUAL (PASSO 1) PARA SEREM CASADAS COM AS CARACTERÍSTICAS DO QUADRO ANTERIOR (PASSO 2, JÁ EXTRAÍDAS NO PROCESSAMENTO DO QUADRO ANTERIOR). EM SEGUIDA É FEITO O CASAMENTO DAS CARACTERÍSTICAS DO QUADRO ANTERIOR COM O QUADRO ATUAL (PASSO 3) PARA SER UTILIZADO NO CÁLCULO DE POSE (PASSO 5). APÓS A POSE CALCULADA É FEITA A EXIBIÇÃO DA APLICAÇÃO DE RA PARA O USUÁRIO (PASSO 6) E EM SEGUIDA É FEITA A ATUALIZAÇÃO DO QUADRO ATUAL PARA UTILIZAÇÃO NO PRÓXIMO QUADRO COMO QUADRO ANTERIOR (PASSO 7).	50
FIGURA 4-11 PIPELINE DA TÉCNICA SPATIAL TEMPORAL IPB. INICIALMENTE SÃO EXTRAÍDAS AS CARACTERÍSTICAS TANTO DO QUADRO ATUAL (PASSO 1) QUANTO DO QUADRO ANTERIOR (PASSO 2 PRÉ-COMPUTADO). EM SEGUIDA, É REALIZADA A ESCOLHA DO KEYFRAME MAIS PRÓXIMO (PASSO 3) PARA UTILIZAÇÃO NA GERAÇÃO DO KEYFRAME INTERMEDIÁRIO (PASSO 4). TERMINADAS AS EXTRAÇÕES DE CARACTERÍSTICAS E A GERAÇÃO DO KEYFRAME INTERMEDIÁRIO, É FEITO O CÁLCULO DAS CORRESPONDÊNCIAS DO QUADRO ATUAL COM O QUADRO ANTERIOR (PASSO 5) E COM O KEYFRAME INTERMEDIÁRIO (PASSO 6). EM SEGUIDA, É FEITO O CÁLCULO DE POSE (PASSO 7), MAIS COMPLEXO E DEPENDENTE DE UMA SÉRIE DE PROCESSAMENTOS ADICIONAIS SE COMPARADO COM AS VARIAÇÕES PRELIMINARES DA TÉCNICA. FINALMENTE, É REALIZADO O AUMENTO DA CENA (PASSO 8) E A ATUALIZAÇÃO DAS INFORMAÇÕES DO QUADRO ATUAL (PASSO 9) PARA UTILIZAÇÃO NO PRÓXIMO QUADRO DA CENA.	51
FIGURA 4-12 RASTREAMENTO UTILIZANDO IPB + FILTRO DE PARTÍCULAS. O CONJUNTO INICIAL DE PARTÍCULAS $x_{k-1}, i, 1Ni = 0..N$ É EXPANDIDO PARA GERAÇÃO DO CONJUNTO $x'ki, , 1Ni = 0..N$ (PASSO 1). EM SEGUIDA É FEITA A EXTRAÇÃO DAS CARACTERÍSTICAS E GERAÇÃO DO MAPA DE DISTÂNCIAS DO QUADRO ATUAL (PASSO 2). UTILIZA-SE O MAPA DE DISTÂNCIAS PARA CALCULAR A VEROSIMILHANÇA DAS PARTÍCULAS E ORDENAR O CONJUNTO (PASSO 3). O SUBCONJUNTO DAS MELHORES PARTÍCULAS É UTILIZADO PARA SOFRER OTIMIZAÇÃO UTILIZANDO O TEMPORAL IPB (PASSO 4). A VEROSIMILHANÇA DAS NOVAS PARTÍCULAS $xki * , 1N + N * i = 0..N *$ É CALCULADA E OS CONJUNTOS SÃO COMBINADOS (PASSO 5). É FEITA A NORMALIZAÇÃO DAS PARTÍCULAS (PASSO 6) JÁ QUE A VEROSIMILHANÇA DOS DOIS CONJUNTOS OBEDECEM A FUNÇÕES DIFERENTES (AS NOVAS PARTÍCULAS	

OBEDECEM A UMA FUNÇÃO f E O CONJUNTO TOTAL OBEDECE A FUNÇÃO g). EM SEGUIDA É FEITO O CÁLCULO DE POSE PONDERADO PELOS PESOS $\pi k_1, \pi k_2 \dots \pi k_n + n * i = 0..N + N *$ A PARTIR DAS PARTÍCULAS $xk_1, xk_2 \dots xk_n + n * i = 0..N + N *$. FINALMENTE É REALIZADA A REAMOSTRAGEM PARA MANUTENÇÃO DAS MELHORES PARTÍCULAS.....	54
FIGURA 4-13 PARTÍCULAS GERADAS ALEATORIAMENTE (TONS DE AMARELO) SOBREPOSTAS À POSE CORRETA DO PRIMEIRO QUADRO DA CENA (CIANO) DURANTE A INICIALIZAÇÃO DO SISTEMA	55
FIGURA 4-14 ESQUEMA DE CÁLCULO DO MAPA DE DISTÂNCIAS DE VALORES INTEIROS (DIREITA) PARA UMA TABELA SINTÉTICA (ESQUERDA). CADA PIXEL DO MAPA DE DISTÂNCIAS RECEBE O VALOR DE SUA DISTÂNCIA AO PONTO DE INTERESSE MAIS PRÓXIMO....	57
FIGURA 4-15 MAPA DE DISTÂNCIAS COM PRECISÃO DE PONTO FLUTUANTE (DIREITA) GERADO A PARTIR DE CARACTERÍSTICAS EXTRAÍDAS (CENTRO) DO QUADRO ATUAL (ESQUERDA). NO MAPA DE DISTÂNCIAS (DIREITA) PONTOS MAIS ESCUROS CORRESPONDENTES A PONTOS MAIS PRÓXIMOS DE PONTOS DE INTERESSE E PONTOS MAIS CLAROS CORRESPONDENTES A PONTOS MAIS AFASTADOS.....	57
FIGURA 4-16 ANTES (A) E DEPOIS (B) DA REAMOSTRAGEM. APÓS A REAMOSTRAGEM, O CONJUNTO DE PARTÍCULAS POSSUI MAIOR REPRESENTATIVIDADE DAS MELHORES PARTÍCULAS.....	60
FIGURA 5-1 IMAGENS DE 25 QUADROS AMOSTRADOS DA SEQUÊNCIA DE 250 QUADROS SINTÉTICOS UTILIZADOS PARA O RASTREAMENTO (RESOLUÇÃO 320x240 PIXELS).....	63
FIGURA 5-2 IMAGENS DE 25 QUADROS AMOSTRADOS DA SEQUÊNCIA DE 400 QUADROS REAIS UTILIZADOS PARA O RASTREAMENTO (RESOLUÇÃO 320x240 PIXELS).....	63
FIGURA 5-3 RESULTADOS DE RASTREAMENTO DO SPATIAL IPB PARA UMA SEQUÊNCIA SINTÉTICA (LINHA SUPERIOR) E UMA SEQUÊNCIA REAL (LINHA INFERIOR).....	64
FIGURA 5-4 TEMPO TOTAL DE COMPUTAÇÃO DO SPATIAL IPB PARA CADA UM DOS PRIMEIROS 250 QUADROS DAS SEQUÊNCIAS SINTÉTICA E REAL.....	65
FIGURA 5-5 PRECISÃO DA ESTIMAÇÃO DO SPATIAL IPB RELATIVA À SEQUÊNCIA DO CUBO SINTÉTICO.....	66
FIGURA 5-6 RESULTADOS DO RASTREAMENTO DO EIXO X DA CÂMERA PELO SPATIAL IPB RELATIVO À SEQUÊNCIA REAL DA CAIXA DE CAFÉ.....	66
FIGURA 5-7 RESULTADOS DE RASTREAMENTO DO TEMPORAL IPB PARA UMA SEQUÊNCIA SINTÉTICA (LINHA SUPERIOR) E UMA SEQUÊNCIA REAL (LINHA INFERIOR). AMBAS AS SEQUÊNCIAS NÃO CONSEGUIRAM SER RASTREADAS ATÉ O FIM DEVIDO AO ACÚMULO DE ERROS DA TÉCNICA.....	67
FIGURA 5-8 TEMPO TOTAL DE COMPUTAÇÃO DO TEMPORAL IPB PARA CADA UM DOS PRIMEIROS 250 QUADROS DAS SEQUÊNCIAS SINTÉTICA E REAL.....	68
FIGURA 5-9 PRECISÃO DA ESTIMAÇÃO DO TEMPORAL IPB RELATIVA À SEQUÊNCIA DO CUBO SINTÉTICO.....	68
FIGURA 5-10 RESULTADOS DO RASTREAMENTO DO EIXO X DA CÂMERA PELO TEMPORAL IPB RELATIVO À SEQUÊNCIA REAL DA CAIXA DE CAFÉ.....	69
FIGURA 5-11 RESULTADOS DE RASTREAMENTO DO SPATIAL TEMPORAL IPB PARA UMA SEQUÊNCIA SINTÉTICA (LINHA SUPERIOR) E UMA SEQUÊNCIA REAL (LINHA INFERIOR).....	70
FIGURA 5-12 TEMPO TOTAL DE COMPUTAÇÃO DO SPATIAL TEMPORAL IPB PARA CADA UM DOS PRIMEIROS 250 QUADROS DAS SEQUÊNCIAS SINTÉTICA E REAL.....	71
FIGURA 5-13 PRECISÃO DA ESTIMAÇÃO DO SPATIAL TEMPORAL IPB RELATIVA À SEQUÊNCIA DO CUBO SINTÉTICO.....	71
FIGURA 5-14 RESULTADOS DO RASTREAMENTO DO EIXO X DA CÂMERA PELO SPATIAL TEMPORAL IPB RELATIVO À SEQUÊNCIA REAL DA	

CAIXA DE CAFÉ.....	72
FIGURA 5-15 RASTREAMENTO UTILIZANDO 30 SUBCONJUNTOS COM 20 AMOSTRAS CADA. ERRO MÉDIO DE 3,86 MM.....	73
FIGURA 5-16 RASTREAMENTO UTILIZANDO 20 SUBCONJUNTOS COM 20 AMOSTRAS CADA. ERRO MÉDIO DE 4,11MM.....	73
FIGURA 5-17 RASTREAMENTO UTILIZANDO 10 SUBCONJUNTOS COM 20 AMOSTRAS CADA. ERRO MÉDIO DE 4,78MM.....	73
FIGURA 5-18 RASTREAMENTO UTILIZANDO 1 SUBCONJUNTO COM 20 AMOSTRAS. ERRO MÉDIO DE 5,0 MM.....	74
FIGURA 5-19 RESULTADOS DE RASTREAMENTO UTILIZANDO O FP COM 300 PARTÍCULAS PARA UMA SEQUÊNCIA SINTÉTICA.	75
FIGURA 5-20 TEMPO TOTAL DE COMPUTAÇÃO DO FP PARA CADA UM DOS 80 QUADROS RASTREADOS DA SEQUÊNCIA DO CUBO COM 300 PARTÍCULAS.	76
FIGURA 5-21 PRECISÃO DA ESTIMAÇÃO DO FP RELATIVA À SEQUÊNCIA DO CUBO SINTÉTICO UTILIZANDO 100 PARTÍCULAS. É VÁLIDO OBSERVAR A FALHA DE RASTREAMENTO NO QUADRO 45.....	76
FIGURA 5-22 PRECISÃO DA ESTIMAÇÃO DO FP RELATIVA À SEQUÊNCIA DO CUBO SINTÉTICO UTILIZANDO 300 PARTÍCULAS. É VÁLIDO OBSERVAR A FALHA DE RASTREAMENTO NO QUADRO 91.....	77
FIGURA 5-23 FERRAMENTA PARA ANÁLISE DE PARTÍCULAS. A LEGENDA CONTEMPLA A VISUALIZAÇÃO DA VEROSSIMILHANÇA DAS PARTÍCULAS ORDENADAS (DECREScente) COM COLORAÇÃO PROPORCIONAL À VEROSSIMILHANÇA (TOM CIANO INDICA POSE CORRETA E TONS DE AMARELO PARA PARTÍCULAS). NA REGIÃO CENTRAL OBSERVAM-SE AS PARTÍCULAS SOBREPOSTAS À POSE CORRETA.	78
FIGURA 5-24 ANÁLISE DA FUNÇÃO DE VEROSSIMILHANÇA. VARIAÇÕES DE TRANSLAÇÃO EM X E Y EM TORNO DA POSE CORRETA DENOTANDO CORRETA FUNÇÃO DE VEROSSIMILHANÇA DEVIDO À CURVATURA APONTANDO PARA A POSE CORRETA, SEM PROFUNDIDADE MUITO ACENTUADA E SUAVIDADE NA VARIAÇÃO DA QUALIDADE DA VEROSSIMILHANÇA.....	79

Lista de Tabelas

TABELA 3-1 COMPARAÇÃO DAS TÉCNICAS DE RASTREAMENTO BASEADO EM MODELOS	31
TABELA 5-1 COMPARAÇÃO DOS TEMPOS E NÚMEROS DE CASAMENTOS RELATIVOS AO ALGORITMO DE RASTREAMENTO SPATIAL IPB.	65
TABELA 5-2 COMPARAÇÃO DOS TEMPOS E NÚMEROS DE CASAMENTOS RELATIVOS AO ALGORITMO DE RASTREAMENTO TEMPORAL IPB.....	67
TABELA 5-3 COMPARAÇÃO DOS TEMPOS E NÚMEROS DE CASAMENTOS RELATIVOS AO ALGORITMO DE RASTREAMENTO SPATIAL TEMPORAL IPB.....	70
TABELA 5-4 COMPARAÇÃO DOS TEMPOS E NÚMEROS DE PARTÍCULAS RELATIVOS AO ALGORITMO DE RASTREAMENTO UTILIZANDO IPB + FP.....	75

Lista de Abreviações e Acrônimos

CGAL	<i>Computational Geometry Algorithms Library</i>
EKF	<i>Extended Kalman Filter – Filtro de Kalman Extendido</i>
fdp	Função de Densidade de Probabilidade
GPGPU	<i>General Purpose Graphical Processing Unit</i>
GPU	<i>Graphical Processing Unit</i>
IPB	<i>Interest Point Based – Baseado em Pontos de Interesse</i>
KF	<i>Kalman Filter – Filtro de Kalman</i>
MAR	<i>Markerless Augmented Reality - Realidade Aumentada sem Marcadores</i>
OpenCV	<i>Open Computer Vision Library</i>
OpenGL	<i>Open Graphics Library</i>
PF	<i>Particle Filter – Filtro de Partículas</i>
PTAM	<i>Parallel Tracking And Mapping</i>
RA	Realidade Aumentada
RanSaC	<i>Random Sample Concensus</i>
SLAM	<i>Simultaneous Localization and Mapping</i>
SVD	<i>Singular Value Decomposition - Decomposição em Valores Singulares</i>
Visp	<i>Visual Servoing Platform</i>
VXL	<i>Vision “Something” Library</i>

1 Introdução

"O gênio é um por cento de inspiração e noventa e nove por cento de transpiração." – Thomas Edison.

Atualmente, a Realidade Aumentada (RA) mostra-se como um ramo de pesquisa na vanguarda da ciência da computação. Associando conhecimentos de áreas afins como computação gráfica, processamento de imagens e visão computacional, a RA se propõe a realizar a inserção de elementos virtuais em cenas reais em tempo real, como forma de expandir as possibilidades e experiências dos usuários (Duh & Billinghurst, 2008). Um exemplo de aplicação pode ser visto na Figura 1-1.



Figura 1-1 Aplicação de RA à transmissão esportiva (posição, nome e velocidade dos pilotos).

A RA sem marcadores (do inglês *Markerless Augmented Reality*, ou MAR) se caracteriza por executar o rastreamento (*tracking*) da câmera, processo que permite a correta inserção dos elementos virtuais na cena real, utilizando informações naturalmente presentes como arestas (Simões, Lima, Teichrieb, Kelner, & Santos, 2008), texturas (Vacchetti, V. Lepetit, & P. Fua, 2004b) ou a própria estrutura da cena (Pollefeys et al., 2007) sem a inserção de

elementos intrusivos (chamados marcadores) ao ambiente.

Devido as suas perspectivas de aplicação nas mais diversas áreas, tais como construção civil, aviação e engenharias (Lima, Simões, Figueiredo, Teichrieb, et al., 2009), as técnicas de MAR têm sido muito estudadas e discutidas tanto pela academia quanto pela indústria.

Neste contexto, esta dissertação se propõe a fazer um levantamento abrangente acerca das técnicas de MAR monoculares (utilizando apenas uma câmera), principalmente as baseadas em textura para, a partir destas observações, realizar uma análise detalhada das técnicas baseadas em pontos de interesse e propor uma nova abordagem de utilização de pontos de interesse associados a um filtro de partículas. Este objetivo visa o aumento de precisão do rastreador, que permitirá o desenvolvimento de soluções de RA mais eficazes.

1.1 Definição do Problema

Rastreamento 3D baseado em pontos de interesse comprehende técnicas responsáveis por corretamente posicionar objetos virtuais numa cena real utilizando para isso informações provenientes da textura do objeto rastreado. Com o intuito de tratar objetos texturizados de forma eficiente e robusta, técnicas de rastreamento tendem a utilizar o máximo de informação possível para reduzir problemas como *drift*, *jitter*, ocorrência de falhas e perda de precisão. Desta forma, o principal problema abordado neste trabalho é construir uma técnica de rastreamento baseada em textura que seja mais robusta aos problemas supracitados e que mantenha uma boa eficiência, funcionando preferencialmente em tempo real.

1.2 Objetivos da Pesquisa

O objetivo principal desta dissertação é propor uma nova técnica de rastreamento baseado em pontos de interesse e filtro de partículas. De forma a atingir este objetivo final, vários objetivos específicos precisam ser atingidos:

- Pesquisa e experimentação na área de Realidade Aumentada sem Marcadores e Filtro de Partículas;
- Estudo e análise comparativa de diversas técnicas de rastreamento para Realidade Aumentada sem Marcadores;
- Análise de desempenho e robustez de técnicas de rastreamento baseadas em pontos

- de interesse;
- Desenvolvimento e implementação de uma nova técnica de rastreamento a partir de informação de textura e filtro de partículas.

1.3 Organização da Dissertação

Esta dissertação foi organizada de forma a introduzir os conceitos de realidade aumentada sem marcadores, com ênfase nas técnicas de rastreamento, responsáveis pela definição da área em questão. Também é um objetivo auxiliar o leitor no entendimento das técnicas baseadas em modelos e as possíveis variações de rastreamento baseado em textura, possibilitando assim uma análise crítica das técnicas implementadas e propostas neste trabalho. A dissertação está organizada como segue.

No segundo capítulo são discutidos os conceitos estatísticos e matemáticos necessários ao entendimento dos algoritmos e técnicas presentes nesta dissertação. É através dele que o leitor entende os conceitos de representação de câmera, rastreamento (cálculo de pose), RanSaC, m-Estimadores e filtro de partículas, necessários para esclarecer os objetivos, desenvolvimento e resultados deste trabalho.

O terceiro capítulo define e categoriza as técnicas de rastreamento sem marcadores, através da construção de uma taxonomia para facilitar o entendimento e a análise crítica do leitor acerca dos resultados encontrados na literatura. Este capítulo também permite ao leitor um macro-entendimento da área de rastreamento sem marcadores.

O quarto capítulo é responsável por definir as técnicas e algoritmos propostos para realizar rastreamento baseado em informação de textura. Foram implementadas variações de técnicas baseadas em pontos de interesse presentes na literatura, bem como variações propostas nesta dissertação através do uso de filtro de partículas.

O quinto capítulo discute os estudos de caso utilizados para avaliar qualitativamente as técnicas implementadas acerca de robustez e desempenho.

O sexto capítulo foca nas conclusões desta dissertação, explora suas contribuições e direciona futuros trabalhos na área.

2 Conceitos Matemáticos e Estatísticos

“O assunto mais importante do mundo pode ser simplificado até ao ponto em que todos possam apreciá-lo e compreendê-lo. Isso é - ou deveria ser - a mais elevada forma de arte.” – Charles Chaplin.

Técnicas de rastreamento 3D se apóiam diretamente em diversos conceitos matemáticos e estatísticos para realizar a recuperação de informações sobre a cena.

Com foco na informação a ser recuperada, este capítulo descreve a representação da câmera virtual e o cálculo de pose da câmera através de algoritmos e propriedades de visão computacional. O cálculo de pose será discutido tanto por algoritmos *bottom-up*, que estimam a nova pose a partir de correspondências 2D-3D, quanto algoritmos *top-down*, que geram várias possíveis poses e utilizam as correspondências para mensurar a qualidade das possíveis poses e combiná-las na formação da pose estimada. Também serão discutidos os conceitos estatísticos necessários a obtenção da pose de forma robusta, como RanSaC e M-estimadores.

2.1 Representação da Câmera

O rastreamento da câmera, que é um aspecto fundamental da fase de registro do *pipeline* de RA, consiste em recuperar a informação que descreve corretamente a câmera virtual usada para posicionar objetos virtuais na cena real e renderizar estes objetos na imagem. Existem vários modelos para projetar objetos 3D em imagens 2D, variando de modelos simples de câmera de orifício (perspectiva) a modelos de lentes complexos que simulam a visão humana (Forsyth & Ponce, 2002). Neste trabalho, foi considerado o modelo de câmera de orifício sem fatores de distorção (lentes), que é um modelo bastante conhecido que aproxima corretamente uma câmera virtual em termos de geometria.

Em todos os modelos de câmera os objetos virtuais são definidos em um sistema de coordenadas genérico, também chamado de sistema de coordenadas de mundo (w_x, w_y, w_z), de forma que possuam uma descrição genérica que não dependa do sistema de coordenadas

de câmera usado (c_x, c_y, c_z). O sistema de coordenadas de câmera corresponde ao sistema de coordenadas de mundo após a aplicação de uma rotação e uma translação. Devido a isso, é necessário converter as coordenadas dos objetos do sistema de coordenadas de mundo para o sistema de coordenadas de câmera antes de projetá-los no plano de imagem (ver Figura 2-1). Esta transformação afim é descrita pela composição das matrizes de rotação R_{3x3} e translação t_{3x1} , resultando em uma matriz $[R|t]_{3x4}$. Quando multiplicada pelas coordenadas homogêneas do ponto 3D, a matriz composta leva ao mesmo ponto 3D no sistema de coordenadas de câmera. Esta matriz é chamada de matriz de pose ou matriz de parâmetros extrínsecos, devido à sua relação com o movimento do modelo de câmera virtual.

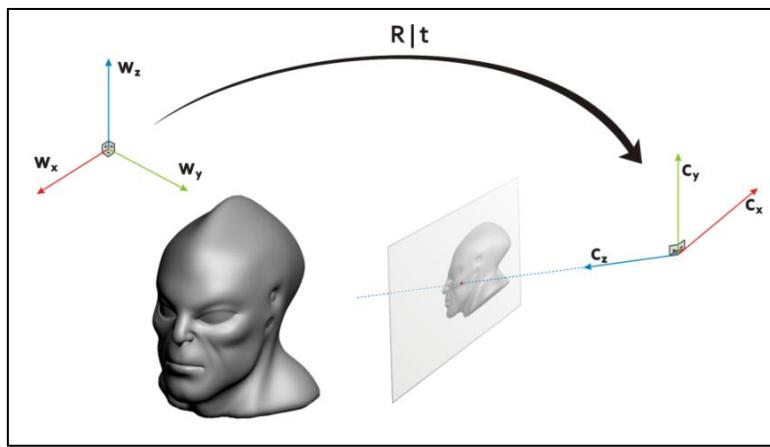


Figura 2-1 O objeto 3D, sua projeção no plano de imagem e a relação entre os sistemas de coordenadas de mundo (w_x, w_y, w_z) e de câmera (c_x, c_y, c_z).

Também é importante notar que, para outros propósitos tais como estimativa de pose, existem várias maneiras de representar a rotação. Uma delas é a representação por vetor de rotação, que corresponde a um vetor representando um eixo de rotação fixo (w_x, w_y, w_z)^t, com sua magnitude se referindo a um ângulo de rotação θ . Esta representação possui uma correspondência unívoca com a forma R_{3x3} ao se usar as fórmulas de Rodrigues e de Rodrigues inversa (Brockett, 1984).

No modelo de câmera de orifício, um ponto no plano de imagem $m = [u, v, f]^t$ é obtido ao se projetar um ponto 3D $M = [x, y, z]^t$ escrito em coordenadas de câmera, respeitando-se as condições da projeção em perspectiva (ver Figura 2-2). Por semelhança de triângulos,

$$u = \frac{x}{z}f + u_0, \quad v = \frac{y}{z}f + v_0. \quad (1)$$

Entretanto, o plano de imagem é dividido em unidades de pixels que correspondem a

áreas bem definidas do mundo real medidas em milímetros. Elas são, por padrão, chamadas de largura do pixel (p_w) e altura do pixel (p_h). Considerando isto, as dimensões de u e v não são escritas em milímetros, mas sim em pixels, e desta forma a equação (1) é reescrita como

$$u = \frac{x}{z p_w} + u_0, v = \frac{y}{z p_h} + v_0 \quad (2)$$

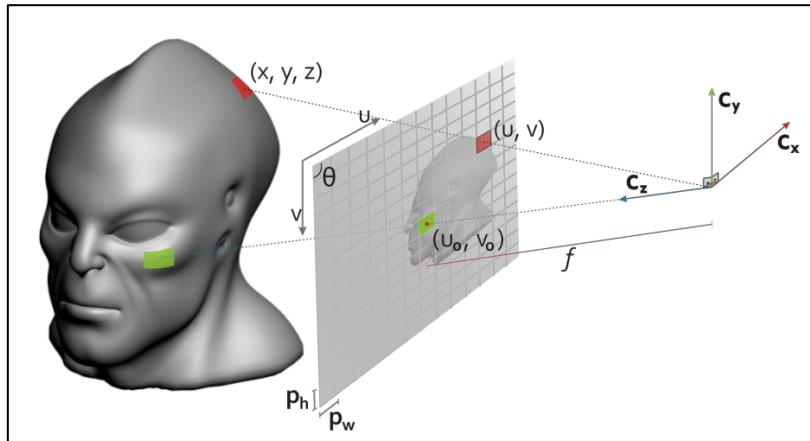


Figura 2-2 Projeção em perspectiva de pontos 3D em pontos de imagem.

Ao se olhar novamente para o problema de projetar um ponto 3D, surge uma primeira versão da matriz de transformação k , que toma um ponto 3D em coordenadas de câmera e retorna a sua representação na imagem 2D em coordenadas homogêneas:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f/p_w & 0 & u_0 \\ 0 & f/p_h & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x/z \\ y/z \\ 1 \end{bmatrix}. \quad (3)$$

Se os *pixels* da câmera não são quadrados, é adicionado à equação um novo parâmetro, também chamado fator de cisalhamento¹, que relaciona o ângulo θ entre as dimensões u e v com o ponto 3D correspondente, tornando a transformação afim (3) em:

¹ Como θ é geralmente próximo de 90° , o fator de cisalhamento é comumente referenciado apenas como $-\cot(\theta)$ e a influência do termo $\cosec(\theta)$ é descartada.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f/p_w & -\cotg(\theta)/p_w & u_0 \\ 0 & f * \cossec(\theta)/p_h & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x/z \\ y/z \\ 1 \end{bmatrix}. \quad (4)$$

A matriz final k apresentada na equação é chamada de matriz de calibração ou matriz de parâmetros intrínsecos, devido à sua dependência da câmera real usada para exibir a cena. Ao combinar as matrizes de parâmetros intrínsecos e extrínsecos, obtém-se uma matriz de projeção P que é responsável por tomar pontos 3D no sistema de coordenadas de mundo e projetá-los no plano de imagem da câmera²:

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \underbrace{\begin{bmatrix} f/p_w & -\cotg(\theta)/p_w & u_0 \\ 0 & f * \cossec(\theta)/p_h & v_0 \\ 0 & 0 & 1 \end{bmatrix}}_k * \underbrace{\begin{bmatrix} R_{11} & R_{12} & R_{13} & T_1 \\ R_{21} & R_{22} & R_{23} & T_2 \\ R_{31} & R_{32} & R_{33} & T_3 \end{bmatrix}}_{R|t} * \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (5)$$

2.2 Cálculo de Pose *Bottom-Up*

Para estimar os parâmetros extrínsecos da câmera para um dado quadro de imagem através de uma técnica *bottom-up*, são necessárias algumas correspondências entre pontos 2D da imagem e pontos 3D do modelo. Nas próximas subseções, duas classes de métodos para estimativa de pose são descritas: Perspective-*n*-Point (PnP) e minimização do erro de reprojeção. Também será descrito o cálculo de pose robusto, através da aplicação de conceitos estatísticos.

2.2.1 Perspective- *n*-Point

PnP consiste basicamente no problema de estimar a pose da câmera $[R|t]$ dadas n correspondências 2D-3D.

Uma primeira abordagem para resolver esse problema é aplicar a equação $PM_i = sm_i$ para cada correspondência i e então resolver um sistema linear. Este método é chamado

² Para finalizar a transformação de pontos 3D para pontos 2D, é também necessário normalizar a resposta utilizando o fator de escala s : $[su, sv, s]^t$ para $[u, v, 1]^t$.

Direct Linear Transformation (DLT) (Faugeras, 1993) e pode estimar todos os parâmetros de P (mesmo se os parâmetros intrínsecos não são conhecidos). Entretanto, ao usar o DLT para calcular P , na maioria dos casos o número de correspondências precisa ser maior que 15, o que é mais que o necessário ao se aplicar outros métodos e para algumas técnicas não é um número aceitável. Além disso, o método DLT minimiza um erro algébrico, enquanto que para o problema de estimação de pose é preferível minimizar um erro geométrico.

Em muitas aplicações de RA os parâmetros intrínsecos não mudam ao longo da seqüência de quadros, visto que a mesma configuração de câmera é usada durante a execução do sistema. Sendo assim, é preferível obtê-los separadamente, reduzindo de uma maneira considerável o número de correspondências necessárias para estimar a pose atual e provavelmente também o erro de estimação. Encorajado por este contexto, o problema PnP usa explicitamente os parâmetros intrínsecos, que precisam ser previamente obtidos, e estima apenas os parâmetros extrínsecos.

Desta forma, ao se tentar resolver o problema P3P, quatro soluções são encontradas. Isto significa que não é possível encontrar uma única solução tendo apenas 3 correspondências. Uma abordagem para encontrar a pose correta é adicionar uma correspondência e resolver o problema P3P para cada subconjunto de 3 correspondências; então, uma pose comum emergirá a partir dos resultados. Ao se resolver os problemas P4P e P5P, frequentemente uma única solução é encontrada, a não ser que as correspondências estejam alinhadas. Para um $n \geq 6$, a solução é quase sempre única.

Várias soluções foram propostas para o problema PnP pelas comunidades de visão computacional e RA. Em geral, elas tentam representar os n pontos 3D em coordenadas de câmera encontrando suas profundidades (que consiste na distância entre o centro óptico da câmera C e o ponto M_i). Na maioria dos casos isso é feito usando restrições dadas pelos triângulos formados pelos pontos 3D e C . Então $[R|t]$ é recuperada a partir do movimento Euclídeo (que é uma transformação afim cuja parte linear é uma transformação ortogonal) que alinha as coordenadas.

Em (Lu, Hager, & Mjolsness, 2000) foi proposta uma solução iterativa, precisa e rápida que minimiza o erro baseado na colinearidade no espaço de objeto. Posteriormente, foi criado o Efficient PnP (EPnP) (Moreno-Noguer, Vincent Lepetit, & Pascal Fua, 2007), que consiste em

um método de complexidade $O(n)$ para o problema PnP com $n \geq 4$. Ele representa todos os pontos como uma soma ponderada de quatro pontos de controle virtuais. Então o problema se reduz a estimar estes pontos de controle no sistema de coordenadas de câmera.

2.2.2 Minimização do Erro de Reprojeção

Apesar de serem capazes de estimar a pose baseados apenas nas correspondências 2D-3D, os métodos PnP são sensíveis a ruídos nas medidas, resultando em perda de precisão. Neste cenário, uma abordagem mais adequada para calcular a pose é a de minimização do erro de reprojeção. Ela consiste em uma minimização por mínimos quadrados não-linear definida pela equação a seguir:

$$[R|t] = \underset{[R|t]}{\operatorname{argmin}} \sum_{i=0}^n \text{dist}^2(\theta(P, M_i), m_i) \quad (6)$$

onde: M_i e m_i são os pontos 3D e 2D correspondentes em coordenadas homogêneas, respectivamente; θ é a função de projeção, que toma como argumentos a matriz de projeção P e o ponto 3D M_i e retorna o ponto 2D projetado; dist é a função de distância Euclideana entre os pontos 2D, que é chamada de residual; e $[R|t]$ são os parâmetros extrínsecos a serem estimados.

Devido ao fato de que a função θ é não-linear, não há uma solução de forma fechada para a equação (6). Neste caso, um método de otimização deve ser usado, tais como o Gauss-Newton (GN) ou o Levenberg-Marquardt (LM) (Triggs, McLauchlan, Hartley, & Fitzgibbon, 2000). Estes métodos refinam iterativamente uma estimativa da pose até ser obtido o melhor resultado possível de acordo com as restrições do próprio método de busca empregado. O incremento de pose entre iterações consecutivas é calculado usando a matriz Jacobiana de θ . Esta matriz pode ser calculada analiticamente ou usando diferenciação. Um requisito de tal método iterativo é uma boa estimativa inicial. Como a diferença entre poses consecutivas é comumente pequena, a pose calculada para o quadro anterior pode ser usada como estimativa para o quadro atual.

2.2.3 Cálculo Robusto

Ao calcular a pose, poucas correspondências 2D-3D espúrias (chamadas de *outliers*) podem arruinar a estimativa mesmo quando existem muitas correspondências corretas (chamadas de

inliers). Existem dois métodos comuns para diminuir a influência destes *outliers* (V. Lepetit & P. Fua, 2005): RANdom SAmple Concensus (RanSaC) e M-estimadores. Eles são descritos na sequência.

2.2.3.1 Random Sample Consensus

O método RanSaC é um algoritmo iterativo que tenta obter a melhor pose usando uma sequência de pequenas amostras aleatórias de correspondências 2D-3D. A ideia é que a probabilidade de ter um *outlier* em uma amostra pequena é muito menor que quando o conjunto inteiro de correspondências é considerado. Embora seja possível utilizar diferentes métricas e funções de custo para avaliar uma pose, a formulação clássica do RanSaC abordada neste trabalho utiliza o erro de reprojeção e a contagem de *inliers* e *outliers* gerados por uma dada hipótese.

O algoritmo recebe basicamente quatro entradas:

1. Um conjunto D de correspondências 2D-3D;
2. Um tamanho de amostra n , que possui um valor pequeno (6, por exemplo);
3. Um limiar t , usado para classificar as correspondências como *inliers* ou *outliers*. Ele consiste em um valor máximo permitido ao retorno da função $dist$ da equação (6). Um valor comumente usado para t é 2.0;
4. Uma probabilidade Γ de encontrar um conjunto que gera uma boa pose. Esta probabilidade é utilizada para calcular o número de iterações do algoritmo. Valores usualmente utilizados são 95% ou 99%.

A partir destas entradas, o RanSaC funciona da seguinte maneira: inicialmente, é determinado um número m de iterações a serem executadas pelo algoritmo (por exemplo, 500). O número de iterações pode ser diminuído durante a execução do algoritmo, dependendo de quão boa é a hipótese de pose até o momento.

Depois disso, a execução do algoritmo começa. Do conjunto D provido, n correspondências são escolhidas aleatoriamente, como mostra a Figura 2-3a, onde os pontos laranja consistem nos pontos 2D das correspondências. A partir desta amostra, uma hipótese

de pose é calculada usando qualquer dos métodos apresentados na Seção 2.2. A seguir, as outras correspondências que não foram incluídas na amostra são utilizadas para verificar o quanto boa é a hipótese encontrada. A Figura 2-3b ilustra esta etapa, onde os pontos cinza são as projeções dos pontos 3D correspondentes. Como mostra a Figura 2-3c, a função $dist$ da equação (6) é aplicada às correspondências e caso a distância (linhas verdes) seja menor que o limiar t , a correspondência é um *inlier* (pontos azuis), caso contrário é um *outlier* (pontos vermelhos). Após todas as correspondências serem testadas, é verificada a porcentagem w de correspondências em D que foram classificadas como *inliers* (Figura 2-3d). Se o valor atual de w for maior que qualquer porcentagem previamente obtida, a hipótese de pose calculada é armazenada, visto que é a mais refinada até o momento.

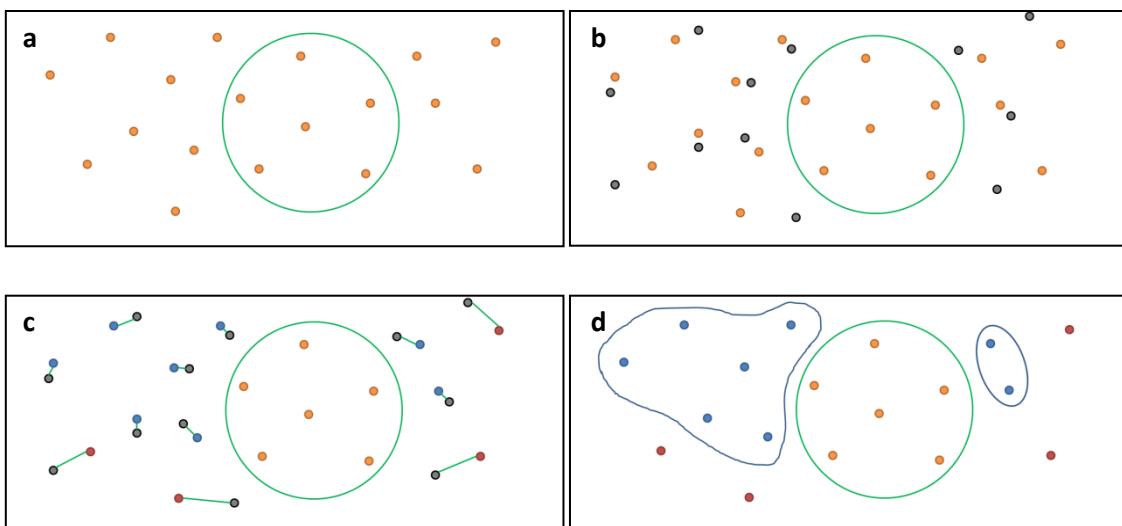


Figura 2-3 Passo-a-passo do algoritmo RanSaC: (a) amostragem aleatória das correspondências e cálculo de pose; (b) avaliação da pose; (c) classificação das correspondências em *inliers* e *outliers*; (d) cálculo da porcentagem de *inliers*.

Quando uma pose refinada é encontrada, o algoritmo tenta diminuir o número de iterações m necessárias. A ideia por trás deste cálculo é bem intuitiva. Como as n correspondências são amostradas independentemente, a probabilidade de que todas as n correspondências sejam *inliers* é w^n . Então, a probabilidade de que exista algum *outlier* é $1 - w^n$. A probabilidade de que todas as m amostras contenham um *outlier* é $(1 - w^n)^m$ e ela deve ser igual a $1 - \Gamma$, resultando em:

$$1 - \Gamma = (1 - w^n)^m. \quad (7)$$

Após calcular o logaritmo em ambos os lados, a seguinte equação é obtida:

$$m = \frac{\log(1-\Gamma)}{\log(1-w^n)}. \quad (8)$$

2.2.3.2 M-Estimadores

Este método é frequentemente usado junto com a minimização do erro de reprojeção de forma a diminuir a influência de *outliers*. Os M-estimadores aplicam uma função aos resíduos que possui um comportamento Gaussiano para valores pequenos e um comportamento linear ou constante para valores maiores. Desta forma, apenas os resíduos que são menores que um limiar c terão um impacto na minimização. Uma versão modificada da equação (6) é então usada:

$$[R|t] = \underset{[R|t]}{\operatorname{argmin}} \sum_{i=0}^n \rho(\operatorname{dist}(\theta(P, M_i), m_i)), \quad (9)$$

onde ρ é a função do M-estimador. Dois dos M-estimadores mais usados são o de Huber e o de Tukey (V. Lepetit & P. Fua, 2005). O M-estimador de Huber é definido por:

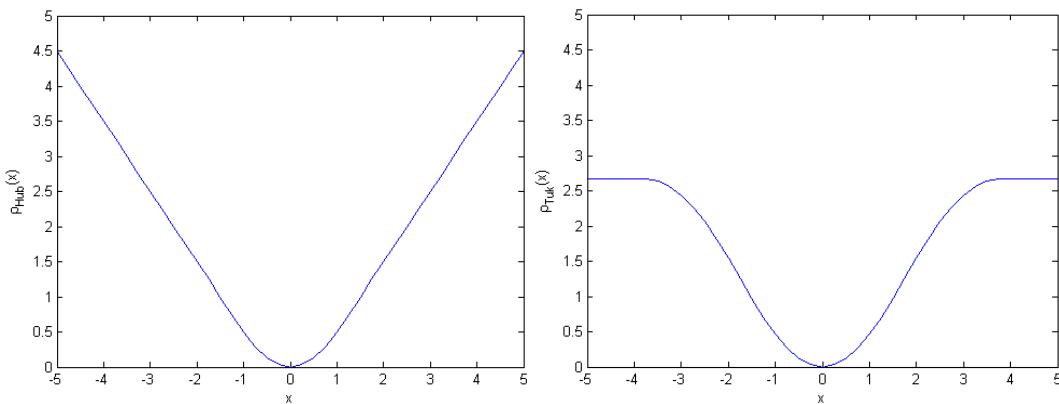
$$\rho_{Hub}(x) = \begin{cases} \frac{x^2}{2}, & |x| \leq c \\ c\left(|x| - \frac{c}{2}\right), & |x| > c \end{cases} \quad (10)$$

onde c é um limiar que depende do desvio padrão do erro de estimação.

O M-estimador de Tukey pode ser computado usando a seguinte função:

$$\rho_{tuk}(x) = \begin{cases} \frac{c^2}{6} \left[1 - \left(1 - \left(\frac{x}{c} \right)^2 \right)^3 \right], & |x| \leq c \\ \frac{c^2}{6}, & |x| > c \end{cases} \quad (11)$$

Os gráficos das funções dos M-estimadores de Huber e Tukey, que podem ser vistos na **Erro! Fonte de referência não encontrada.**, destacam como os resíduos são ponderados de acordo com sua magnitude.



2-4 Funções do M-estimador de Huber com $c = 1$ (esquerda) e do M-estimador de Tukey com $c = 4$ (direita).

2.3 Cálculo de Pose *Top-Down*

Ao contrário das técnicas *bottom-up* que utilizam correspondências entre pontos 2D e 3D para estimar uma pose, as técnicas *top-down* geram diversas possíveis poses e as analisam (validam) utilizando as correspondências 2D-3D de forma a gerar uma única pose final. Nesta subseção será descrita a técnica de Filtros de Partículas (FP), comumente utilizada para tratar problemas não-lineares e não-gaussianos, como é o caso do cálculo de pose.

2.3.1 Filtro de Partículas

O FP, ou método de Monte Carlo Sequencial, é uma técnica bayesiana de rastreamento que busca estimar o novo estado de um sistema a partir de medidas com ruído do próprio sistema (Arulampalam, Maskell, Gordon, & Clapp, 2002). Tais técnicas se baseiam na representação de densidade de probabilidade por nuvens de pontos, também chamados de partículas, para realizar a estimativa do novo estado, sendo muito utilizadas não apenas no cálculo de pose, mas em qualquer sistema descrito por um modelo bayesiano, como em econometria.

Quando aplicado ao cálculo de pose, FP também é conhecido como algoritmo *Condensation* (Isard & Blake, 1998). A ideia principal do algoritmo é representar a função de densidade de probabilidade (*fdp*) *a posteriori* $p(\mathbf{x}_k, \mathbf{z}_{1:k})$ do estado \mathbf{x}_k no quadro atual (estado atual k) a partir de um conjunto finito $\{\mathbf{x}_k^i, \pi_k^i\}_{i=0..N}$ de N amostras (partículas) em que cada partícula \mathbf{x}_k^i está diretamente associada a um peso π_k^i . Cada partícula \mathbf{x}_k^i representa uma possível pose e o conjunto $\mathbf{z}_{1:k}$ representa as observações até antes do quadro k . Para cada

novo quadro, ou seja, a cada iteração, o conjunto de partículas $\{x_{k-1}^i, \frac{1}{N}\}_{i=0..N}$ do quadro anterior é expandido aplicando-se um ruído em cada partícula a partir de um modelo estatístico dinâmico (pode ser aplicado ruído gaussiano (Klein & D. Murray, 2006) ou ruído uniforme (Pupilli & Calway, 2005)) gerando um novo conjunto $\{x_k^i, \frac{1}{N^*}\}_{i=0..N^*}$.

Uma vez propagadas as partículas, a verossimilhança de cada partícula é calculada de forma a mensurar sua qualidade. Para tanto, pode-se considerar que o erro de reprojeção é uma boa métrica para os fatores residuais, e considerar a função de densidade de probabilidade (fdp) *a posteriori* de cada partícula $p(\mathbf{z}_{1:k}, \mathbf{x} = \mathbf{x}_k)$ como sendo equivalente a verossimilhança (Teuliere, Eric Marchand, & Eck, 2010). Desta forma, aproxima-se a verossimilhança pela razão entre os pontos (v) que possuem um erro residual menor que determinado limiar (distância em *pixels*) em relação ao total de pontos projetados (n):

$$p(\mathbf{z}_{1:k}, \mathbf{x} = \mathbf{x}_k) = \lambda \frac{v}{n} \quad (12)$$

na qual λ determina uma constante de proporcionalidade. De forma a otimizar o cálculo, é possível aproximar o cálculo da verossimilhança utilizando um mapa de distâncias do quadro atual (imagem de tamanho igual a do quadro atual que guarda para cada *pixel* a distância ao ponto de interesse mais próximo).

Através disto, o erro residual é aproximado pela distância do ponto extraído mais próximo ao projetado, não necessariamente uma correspondência válida, mas que permite selecionar as melhores partículas para os passos seguintes.

Após o cálculo da verossimilhança, são gerados os pesos π_k^i através do uso da própria verossimilhança. É importante destacar a importância de uma normalização dos pesos π_k^i de forma que $\sum_{i=1}^{N^*} \pi_k^i = 1$. Uma vez gerados os pesos obtém-se o conjunto $\{x_k^i, \pi_k^i\}_{i=0..N^*}$ que será utilizado para estimativa da pose. Tal estimativa é feita utilizando-se a média ponderada pelos pesos π_k^i do conjunto de partículas x_k^i . Uma vez estimada a pose, é executado o algoritmo de reamostragem para reduzir o conjunto de partículas e pesos $\{x_k^i, \pi_k^i\}_{i=0..N^*}$ ao novo conjunto $\{x_k^i, \frac{1}{N}\}_{i=0..N}$ para utilização na próxima iteração. Um detalhamento acerca da utilização de FP é dado no trabalho publicado por (Arulampalam, Maskell, Gordon, & Clapp, 2002).

3 Realidade Aumentada Sem Marcadores

"Ler sem pensar é uma ocupação inútil." – Confúcio.

Realidade Aumentada sem Marcadores, do inglês *Markerless Augmented Reality* (MAR), têm por objetivo realizar a correta inserção de elementos virtuais em cenas reais em tempo real, de modo a permitir uma melhor interação entre o usuário e o mundo real (Azuma *et al.*, 2001). Esta melhora ocorre principalmente pela possibilidade que o usuário tem de perceber informações adicionais sobre o ambiente real fornecidas pela interface de RA e aplicá-las na realização das tarefas. Para obter tais resultados são necessários três pilares (Duh & Billinghurst, 2008): rastreamento e registro, que são os processos e técnicas pelas quais o objeto virtual é corretamente posicionado sobre a cena real; interação, a qual define como o usuário irá interagir com o meio no qual está inserido; e renderização, que trata de como exibir o objeto virtual de forma a possibilitar, em um mundo ideal, a completa homogeneização visual entre o real e o virtual. Devido ao atual estágio de desenvolvimento desta tecnologia, a maior parte dos esforços despendidos se concentram na busca por técnicas de rastreamento e registro cada vez mais eficientes e robustas, de forma a possibilitar o correto relacionamento entre o real e o virtual. Desta forma, este capítulo aborda uma taxonomia das técnicas de rastreamento monocular, definida pelo autor, com enfoque principal nas técnicas baseadas em modelos recursivas.

3.1 Taxonomia

Técnicas de rastreamento para MAR monocular em tempo real podem ser classificadas em dois tipos principais: baseadas em modelos (V. Lepetit & P. Fua, 2005) e baseadas em reconstrução 3D (Lima *et al.*, 2007). Nas técnicas baseadas em modelos, o conhecimento sobre o mundo real é obtido antes do início do rastreamento e armazenado em um modelo 3D utilizado para recuperar a pose da câmera. Nas técnicas baseadas em reconstrução 3D, o

movimento da câmera a cada quadro é estimado sem nenhum conhecimento prévio da cena, sendo adquirido ao longo do rastreamento juntamente com a estrutura 3D do ambiente.

3.1.1 Rastreamento Baseado em Modelos

Considerando a natureza do rastreamento, técnicas baseadas em modelos podem ser classificadas em duas categorias (Lima, Simões, Figueiredo, Teichrieb, et al., 2009): técnicas de rastreamento recursivas, onde a pose da câmera no quadro anterior é utilizada como estimativa para o cálculo da pose atual (Simões, Lima, Teichrieb, Kelner, & Santos, 2008) (Teuliere, Eric Marchand, & Eck, 2010) (Drummond & Cipolla, 2002) (Jurie & Dhome, 2002) (Vacchetti, V. Lepetit, & P. Fua, 2004b) (Pressigout & E. Marchand, 2006); e rastreamento por detecção, onde é possível calcular a pose atual sem o auxílio da pose anterior, permitindo inicialização automática e recuperação de falhas (Skrypnyk & Lowe, 2004) (Hinterstoisser, Kutter, Navab, P Fua, & V. Lepetit, 2009) (Ladikos, Benhimane, & Navab, 2007).

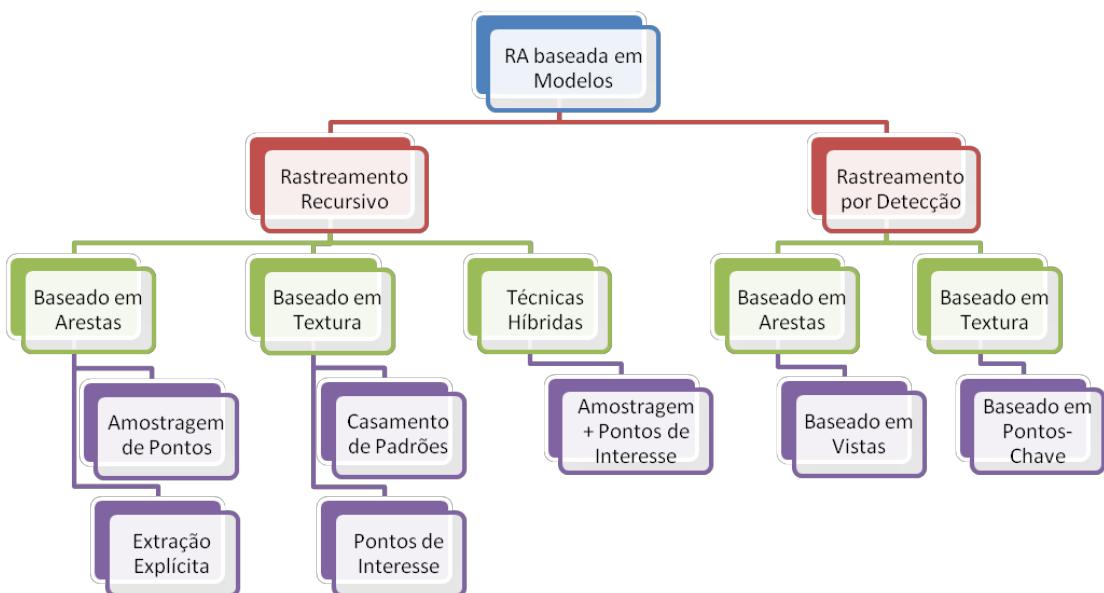


Figura 3-1 Taxonomia de técnicas de rastreamento baseado em modelos para MAR.

Levando em consideração o tipo de característica utilizada para o rastreamento, as técnicas baseadas em modelos (Lima, Simões, Figueiredo, Teichrieb, & Kelner, 2010) podem ser classificadas em três categorias: baseadas em aresta, nas quais a pose da câmera é estimada através do casamento do modelo 3D aramado de um objeto com as informações de arestas extraídas da imagem (Ababsa & Mallem, 2008) (Drummond & Cipolla, 2002); baseadas em textura, que levam em consideração a informação de textura da imagem para realizar o

rastreamento (Vacchetti, V. Lepetit, & P. Fua, 2004b) (Xiao, Moriyama, Kanade, & Cohn, 2003); e técnicas híbridas, que utilizam as informações de aresta e de textura de forma complementar (Vacchetti, V. Lepetit, & P. Fua, 2004a) (Pressigout & E. Marchand, 2006).

Técnicas recursivas baseadas em aresta podem ser divididas em duas subcategorias: baseadas em amostragem de pontos (Figura 3-2, acima e à esquerda), são métodos que compararam pontos de controle amostrados nas arestas do modelo 3D rastreado com pontos de forte gradiente extraídos da imagem (Simões, Lima, Teichrieb, Kelner, & Santos, 2008); e baseadas na extração explícita das arestas (Figura 3-2, acima e à direita), que realizam o rastreamento a partir de arestas extraídas da imagem e as casam com a projeção do modelo (Koller, Daniilidis, & Nagel, 1993). Técnicas recursivas baseadas em textura são classificadas em duas subcategorias: casamento de padrões (Figura 3-2, abaixo e à esquerda), que aplica um modelo de distorção a uma imagem de referência para recuperar o movimento de objetos rígidos (Jurie & Dhome, 2002); e baseadas em pontos de interesse (Figura 3-2, abaixo e ao centro), que levam em consideração características localizadas na estimativa da pose da câmera (Vacchetti, V. Lepetit, & P. Fua, 2004b). Técnicas recursivas híbridas (Figura 3-2, abaixo e à direita) podem ser classificadas em amostragem de pontos com pontos de interesse, na qual realizam o rastreamento baseado tanto nas informações oriundas das arestas quanto nas informações oriundas da textura do objeto rastreado (Pressigout & E. Marchand, 2006).

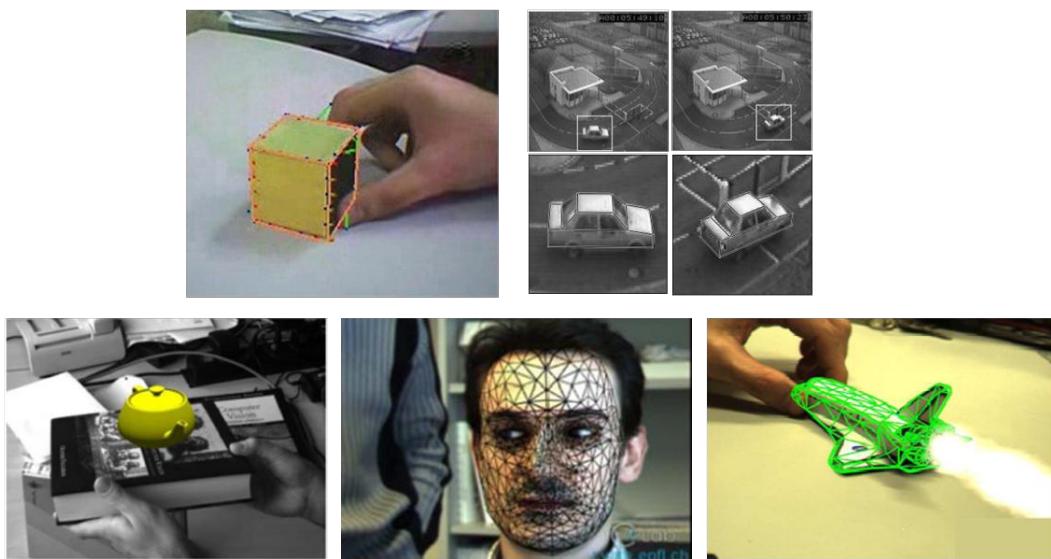


Figura 3-2 Exemplos de rastreamento recursivo: amostragem de pontos (acima e à esquerda), extração explícita das arestas (acima e à direita), casamento de padrões (abaixo e à esquerda), pontos de interesse (abaixo e ao centro) e técnica híbrida (abaixo e à direita).

As técnicas de rastreamento por detecção baseadas em aresta são chamadas de baseadas em vistas (Figura 3-3, direita), visto que o quadro atual é casado com visões 2D do objeto previamente obtidas de diferentes posições e orientações (WIEDEMANN, ULRICH, & STEGER, 2008). Os métodos de rastreamento por detecção baseados em textura são conhecidos como baseados em pontos-chave (Figura 3-3, direita) (Skrypnyk & Lowe, 2004). Pontos-chave são invariantes a escala, ponto de vista e mudanças de iluminação. Eles são extraídos da imagem do objeto a cada quadro, provendo as correspondências 2D-3D necessárias para a estimativa da pose.

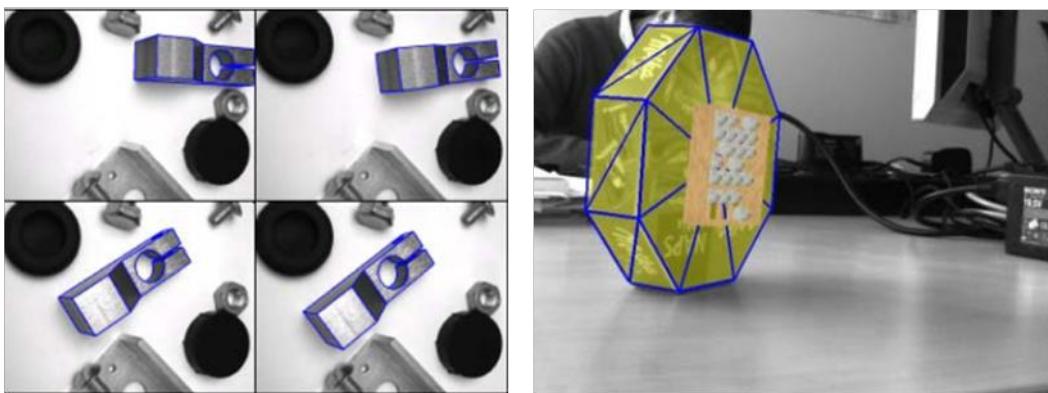


Figura 3-3 Exemplos de rastreamento por detecção: vistas (esquerda) e pontos-chave (direita).

3.1.2 Análise das Técnicas Baseadas em Modelos

As abordagens de rastreamento baseado em modelos apresentadas para MAR (Lima, Simões, Figueiredo, Teichrieb, et al., 2009) podem ser analisadas através de algumas métricas relevantes para tal tarefa. Uma das mais importantes métricas é a presença de detecção automática, onde a intervenção do usuário não é necessária para determinar a pose inicial da câmera. Outra métrica importante é o tempo de processamento para realizar o rastreamento. Caso o tempo requerido para calcular a pose da câmera seja pequeno, significa que o tempo restante pode ser gasto em outras tarefas como inserção de conteúdo 3D na cena real ou animações, por exemplo. Precisão e robustez são as duas últimas métricas utilizadas. Enquanto precisão corresponde à medida de quão correto é o rastreamento realizado ao longo dos quadros da sequência, robustez mede quão sensível a ruídos é o rastreador. A tabela Tabela 3-1 compara as técnicas de rastreamento baseadas em modelos com respeito aos critérios citados. As comparações levam em consideração a natureza do rastreamento que é comum a

maioria das técnicas de cada categoria.

Tabela 3-1 Comparação das técnicas de rastreamento baseado em modelos

Categoria	Técnica	Detecção	Processamento	Precisão	Robustez
Rastreamento Recursivo Baseado em Aresta	Amostragem de Pontos	Não	Baixo	<i>Jitter</i>	Sensível a: - Movimentos rápidos - Fundo poluído
	Extração Explícita	Não	Baixo		Sensível a: - Movimentos rápidos - Mudanças de iluminação
Rastreamento Recursivo Baseado em Textura	Casamento de Padrões	Não	Baixo	Muito Preciso	Sensível a: - Movimentos rápidos - Mudanças de iluminação - Oclusão
	Pontos de Interesse	Não	Baixo	Preciso	Sensível a: - Movimentos rápidos
Rastreamento Recursivo Híbrido	Amostragem de Pontos + Pontos de Interesse	Não	Baixo	Muito Preciso	Sensível a: - Movimentos rápidos
Rastreamento por Detecção Baseado em Aresta	Baseado em Vistas	Sim	Alto	Preciso	Restrições nas poses rastreadas
Rastreamento por Detecção Baseado em Textura	Baseado em Pontos-Chave	Sim	Alto	<i>Jitter e Drift</i>	Robusto

Técnicas de MAR também podem ser classificadas quanto ao cenário de aplicação.

Técnicas baseadas em aresta costumam ser empregadas em cenários nos quais os objetos são poligonais, especulares ou possuem contornos bem definidos. Além disso, técnicas baseadas em aresta costumam ter um bom desempenho, sendo indicadas para utilização em dispositivos móveis (Lima, Teichrieb, Kelner, & Lindeman, 2009). Quando o cenário possui objetos texturizados, técnicas baseadas em textura são a melhor opção. Em casos de objetos texturizados em ambientes sem mudanças de iluminação, as técnicas baseadas em casamento de padrão costumam ser a melhor escolha tendo ótimo desempenho com objetos planares. Quando casamento de padrões não é uma opção, as técnicas baseadas em pontos de interesse devem ser utilizadas. Quando a precisão é um aspecto crítico, técnicas híbridas aparecem como melhor escolha devido à possibilidade de tratar diferentes características do objeto rastreado como arestas e texturas. É válido observar a ineficiência de combinar técnicas de rastreamento de forma direta (utilizando o resultado de uma como entrada para a outra), pois esta abordagem tende a degenerar o resultado devido a falhas no casamento das características (Vacchetti, V. Lepetit, & P. Fua, 2003). Para obter resultados satisfatórios, faz-se importante combinar as informações oriundas de ambas as fontes no cálculo da pose (Vacchetti, V. Lepetit, & P. Fua, 2004a). Técnicas baseadas em pontos-chave podem sofrer de *jitter* (instabilidade no rastreamento devido à descontinuidade temporal do cálculo de pose) uma vez que estimam a pose atual baseado apenas na informação do quadro atual. Deste modo, a utilização de informação temporal pode ser levada em consideração para reduzir o problema. Apesar disto, técnicas baseadas em pontos-chave costumam ser menos precisas que as técnicas recursivas baseadas em textura devido a falta de precisão no casamento das características sendo geralmente suscetíveis a *drift* (erro no cálculo da pose). Já as técnicas baseadas em vistas são extremamente precisas, mas só cobrem um espectro restrito de rotações e escalas do objeto rastreado atingindo baixas taxas de detecção.

É válido observar também a utilização de técnicas estatísticas para aumentar a robustez das técnicas de rastreamento. Abordagens como RanSaC e M-estimadores são bastante comuns para minimizar a influência de casamentos espúrios (*outliers*), como discutido na seção 2.2.3. Já o *framework* estatístico FP é comumente utilizado para prever novas poses a partir do conhecimento acumulado, levando a técnicas computacionalmente mais custosas, mas com uma precisão muitas vezes superior (Arulampalam, Maskell, Gordon, & Clapp, 2002).

3.1.3 Rastreamento Baseado em Reconstrução 3D

As técnicas de rastreamento baseadas em reconstrução 3D (Teixeira, Simões, Roberto, & Teichrieb, 2010) são divididas em duas categorias distintas: técnicas baseadas em *Structure from Motion* (SfM), onde o rastreamento e a recuperação da estrutura da cena ocorrem a partir de informações oriundas da geometria epipolar sem auxílio de nenhum elemento previamente conhecido (Hartley & Zisserman, 2004)(David Nistér, 2003)(Pollefeys et al., 2007); e técnicas baseadas em *Simultaneous Localization and Mapping* (SLAM), a qual se baseia na recuperação de marcos em um *framework* estatístico para realizar a reconstrução e o rastreamento em tempo real (a J. Davison & D. W. Murray, 2002)(A. J. Davison, Reid, Molton, & Stasse, 2007) (Newcombe & A. J. Davison, 2010).

SfM é uma técnica clássica em visão computacional que permite tanto a aquisição da estrutura quanto o rastreamento de uma cena (ver Figura 3-4). Devido a estas características, SfM é amplamente utilizado para realizar reconstrução 3D a partir de imagens (Pollefeys et al., 2007). Em sua implementação tradicional, o *pipeline* de SfM não possui restrições para funcionar em tempo real e um conhecimento aprofundado de seus algoritmos e técnicas é requisito básico para realizar as otimizações necessárias para funcionamento em tempo real.



Figura 3-4 Rastreamento baseado em reconstrução 3D utilizando SfM de tempo real (David Nistér, 2003). Objetos reconstruídos no centro, câmeras visualizadas como quadriláteros (plano de visão) transpassados pela linha que liga o centro da câmera ao plano de visão em torno dos objetos.

Usualmente, o *pipeline* de SfM é constituído das seguintes fases: extração de características, responsável por extrair e relacionar características de várias imagens; geração de poses, etapa correspondente ao cálculo das poses a partir de conceitos de geometria epipolar; auto-calibração, fase na qual os parâmetros intrínsecos da câmera são descobertos; triangulação, etapa na qual os pontos 3D são recuperados; e texturização, etapa final que

aplica informação de cor aos pontos reconstruídos de forma a obter uma reconstrução fiel.

De forma a otimizar o *pipeline* para realizar o rastreamento em tempo real, as etapas de auto-calibração e texturização são geralmente eliminadas (a câmera é considerada calibrada e como o objetivo é apenas rastrear, não há necessidade de realizar a reconstrução completa). A fase de triangulação é mantida de forma a realizar a correção e validação da câmera calculada na etapa anterior. Diversos passos das etapas são simplificados e também implementados em *Graphical Processing Unit* (GPU). Dessa forma, o maior gargalo das técnicas baseadas em SfM para rastreamento em tempo real é a manutenção da robustez apesar das muitas simplificações e otimizações necessárias ao requisito de tempo de processamento (Pollefeys et al., 2007). A grande vantagem de técnicas baseadas em SfM é a completa liberdade com relação ao cenário de aplicação, pois uma vez que reconstroem a cena enquanto realizam o rastreamento, tais técnicas são aplicáveis a objetos genéricos e ambientes diversos (Hartley & Zisserman, 2004).

Em (Newcombe & A. J. Davison, 2010) é realizada a reconstrução densa (reconstrução na qual a quantidade de pontos reconstruídos é da mesma ordem da quantidade de pixels da imagem) em tempo real. Através de diversas otimizações no *pipeline* de reconstrução 3D e utilizando técnicas avançadas de visão computacional, como *plane sweep* e fusão de mapas de profundidade associadas ao SfM, esta técnica atinge uma alta precisão com baixo tempo de processamento possibilitando a criação de aplicações de RA com simulação física (ver Figura 3-5).

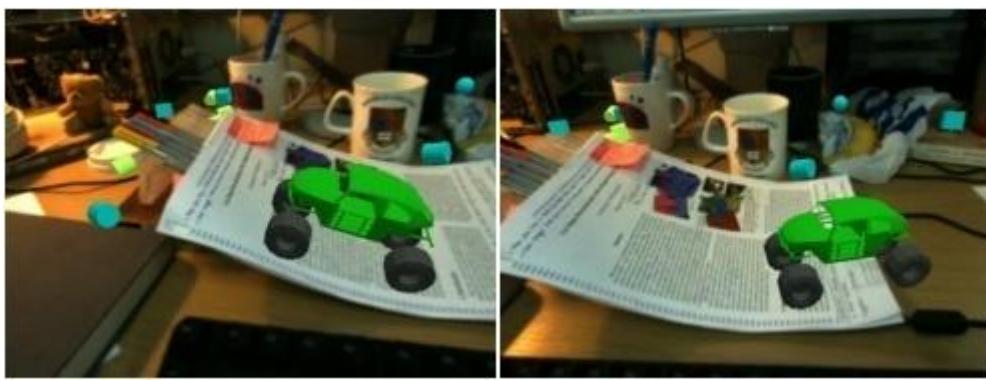


Figura 3-5 Aplicação de RA com simulação física a partir de rastreamento utilizando SfM de tempo real (Newcombe & A. J. Davison, 2010).

Técnicas de rastreamento baseadas em SLAM são bem definidas e consolidadas na comunidade de robótica, sendo a sua principal utilização o controle para navegação autônoma

de robôs. Para realizar tal navegação, técnicas de SLAM realizam a reconstrução esparsa (quantidade de pontos inferior à quantidade de pixels das imagens) da cena em tempo real de forma a possibilitar o rastreamento e evitar obstáculos, com enfoque principal na localização e não na reconstrução.

Para realizar o rastreamento, as técnicas de SLAM utilizam no primeiro quadro da cena um objeto conhecido a uma distância previamente determinada para inicialização do sistema, conforme ilustra a Figura 3-6. Isto se dá devido à utilização de um *framework* de inferência bayesiana sequencial, geralmente através de filtro de Kalman (KF) ou filtro de Kalman estendido (EKF). Através deste *framework*, utilizando propriedades de previsibilidade, é possível realizar a reconstrução esparsa de marcos (cantos com assinaturas únicas e mais bem definidas que cantos padrão) e rastrear a câmera a uma taxa de 30 quadros por segundo, suficiente para considerar uma aplicação de tempo real.



Figura 3-6 Inicialização para rastreamento utilizando SLAM através de objeto conhecido (objeto verde à esquerda) e cena aumentada com quatro objetos virtuais ao longo do rastreamento (direita).

Técnicas mais recentes como o *Parallel Tracking and Mapping System* (PTAM) (Klein & D. Murray, 2009) utilizam uma variação do SLAM de forma a realizar o rastreamento com base em centenas de marcos e não de apenas alguns como ocorre nas técnicas tradicionais de SLAM (A. J. Davison, Reid, Molton, & Stasse, 2007). O PTAM consegue atingir um ótimo desempenho possibilitando a criação de aplicações de realidade aumentada em dispositivos móveis como pode ser visto na Figura 3-7.

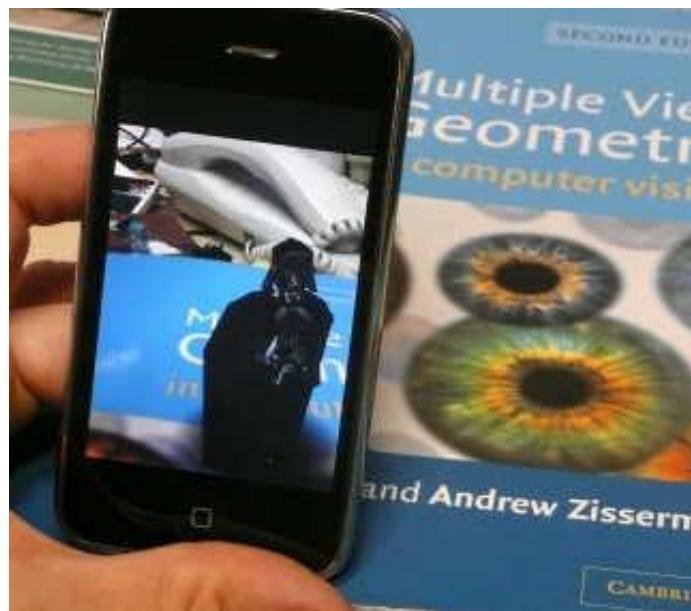


Figura 3-7 Realidade aumentada em dispositivos móveis utilizando o *Parallel Tracking and Mapping System* (PTAM).

4 Rastreamento Baseado em Textura

“Qualquer tecnologia suficientemente avançada é indistinta de magia.” – Arthur C. Clarke.

Este capítulo discute o rastreamento a partir de informação de textura. Foi implementada e analisada uma técnica a partir de pontos de interesse, doravante chamada Interest Point Based (IPB), através da utilização de informação espacial (Spatial IPB), de informação temporal (Temporal IPB) e da complementação de ambas informações (Spatial Temporal IPB) (Vacchetti, V. Lepetit, & P. Fua, 2004b). Tais implementações possibilitam, além do rastreamento, a observação de diversas características diferentes de rastreamento utilizadas em realidade aumentada sem marcadores. Também será discutida uma variação do IPB, proposta nesta dissertação, através da utilização do *framework* estatístico de filtro de partículas como forma de melhorar a robustez da técnica. Como produto direto da proposição do uso do filtro de partículas associado ao IPB, este trabalho também gerou uma nova função de verossimilhança a partir da informação de textura.

4.1 Interest Point Based

Uma das mais aclamadas e desenvolvidas técnicas de rastreamento baseado em textura, a técnica baseada em pontos de interesse (IPB) recebeu bastante atenção nos últimos anos devido à utilização de características locais que podem tratar oclusões do objeto e mudanças de iluminação na cena real a ser rastreada de forma efetiva. Uma vez que realiza o casamento de apenas um subconjunto das características da textura rastreada, a técnica reduz sua complexidade computacional, aumentando sua eficiência consideravelmente. Nos últimos anos, diversos melhoramentos foram propostos relacionados a precisão, robustez e eficiência da técnica.

Existem diversas características do IPB que buscam se complementar para atingir melhores resultados (exploradas nesta dissertação de mestrado como técnicas básicas para a construção do conhecimento sobre a técnica IPB). A utilização de informações espaciais (na

técnica básica Spatial IPB) é responsável por eliminar o problema de *drift* (erro acumulado entre os quadros da cena que leva a problemas de precisão) através do uso de informações *offline* sob a forma de *keyframes*. Já a técnica baseada apenas em informação temporal (chamada Temporal IPB) é utilizada para eliminar o problema de *jitter*, aquele responsável pela sensação de falta de suavidade no movimento entre os quadros rastreados, devido à utilização apenas de informação *offline* como na variação Spatial IPB. Uma combinação de ambas as abordagens propõe não sofrer de problemas de *jitter* e *drift* sob a forma da técnica Spatial Temporal IPB (Vacchetti, V. Lepetit, & P. Fua, 2004b).

4.2 Spatial IPB

Na variação da técnica IPB chamada Spatial IPB, a utilização de informação *offline* se apresenta como carro chefe da técnica, levando a um rastreamento preciso, porém com *jitter*, ou sensação de falta de suavidade na mudança do ponto de vista da câmera. Seu *pipeline* utiliza quadros previamente computados para realizar o casamento das características da cena e o posterior cálculo da pose. O *pipeline* da técnica Spatial IPB pode ser resumido através do diagrama da Figura 4-1 e suas etapas serão detalhadas nas próximas subseções: geração de *keyframes*, escolha de *keyframe* mais próximo, geração de *keyframe* intermediário, extração de características, casamento de características, cálculo de pose e atualização do quadro atual.

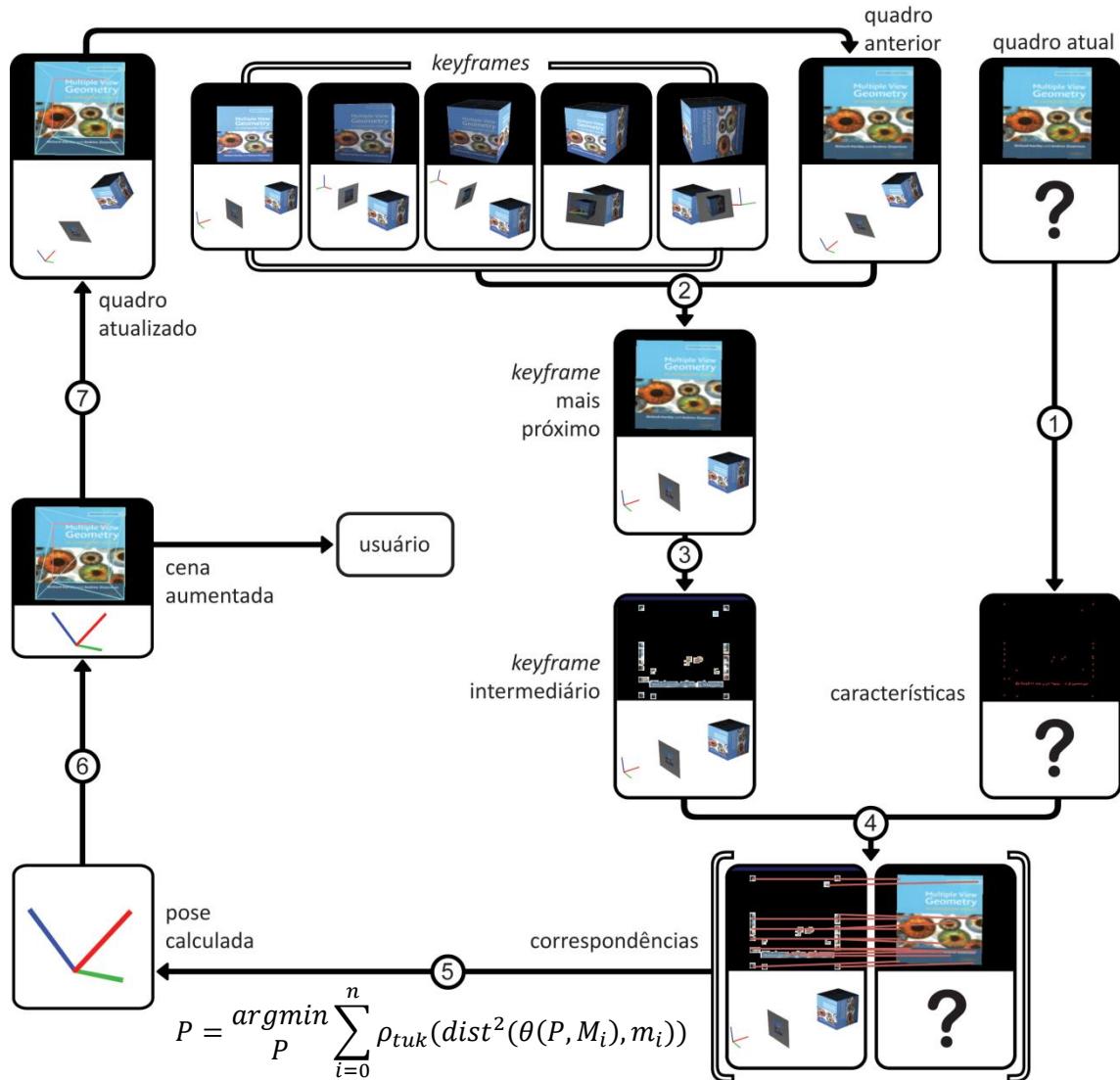


Figura 4-1 Fluxo de execução da técnica Spatial IPB. O funcionamento se baseia na aquisição de informações *offline* sob a forma de *keyframes*. No início do rastreamento é realizada a extração de características do quadro atual (passo 1). Em seguida, é feita a escolha do *keyframe* mais próximo do quadro anterior (passo 2), também conhecido no início do rastreamento, e em seguida é feito o cálculo do *keyframe* intermediário (passo 3), aproximando o *keyframe* do quadro anterior. Após realizados os passos anteriores (podem ser feitos em paralelo, 1 em paralelo com 2 e 3), é realizado o casamento das características extraídas do quadro atual e do *keyframe* intermediário (passo 4). Em seguida, é feito o cálculo da pose através da minimização do erro de reprojeção (passo 5). Após o cálculo da pose o rastreamento é concluído aumentando a cena com a informação desejada (passo 6) e é feita a atualização dos dados do quadro atual (passo 7) para utilização no próximo quadro por se tratar de uma técnica recursiva.

4.2.1 Geração de Keyframes

A estrutura básica da técnica de rastreamento Spatial IPB é o *keyframe*. É através desta

estrutura adquirida previamente, fato que denota um conhecimento prévio da cena, que se torna possível um relacionamento com características extraídas da imagem do quadro atual de forma a realizar a correta recuperação da pose da câmera. Cada *keyframe* guarda informações relativas ao quadro capturado de forma a evitar a ocorrência de *drift*, acúmulo de erros ocasionado pela não utilização de informação *offline*. Estas informações (ver Figura 4-2) correspondem à imagem do *keyframe* (pode ser definida como uma imagem RGB ou qualquer formato conveniente), a pose da câmera que gerou a imagem (pose definida pela rotação e translação), os parâmetros intrínsecos da câmera (matriz de 9 elementos, ver seção 2.1), os pontos 2D pertencentes a imagem (extraídos através de um extrator de cantos como o detector de cantos de Harris, técnica descrita na seção 4.2.4) e os pontos 3D do objeto correspondentes a cada ponto 2D extraído (reprojeção do ponto 2D).

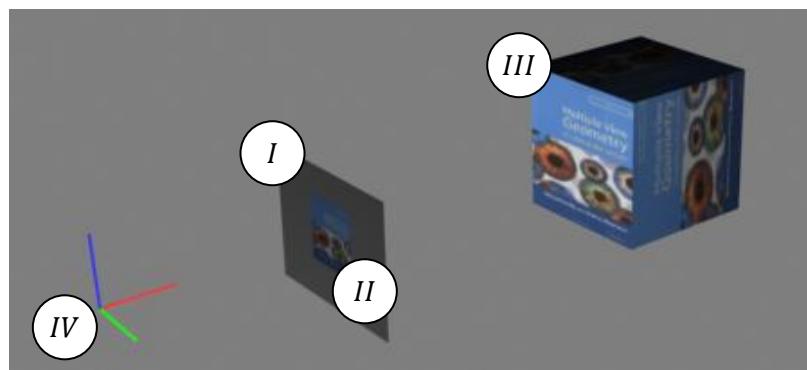


Figura 4-2 *Keyframe* gerado *offline* contendo a imagem do quadro capturado (I), as características extraídas sob a forma de pontos 2D da imagem (II), os pontos 3D do modelo correspondentes (III) e as informações da câmera (IV).

Para gerar o *keyframe*, se faz necessária a descoberta da pose da câmera e a reprojeção dos pontos 2D extraídos pelo detector de cantos de Harris no objeto 3D a ser rastreado (modelo), podendo ser utilizado para esta finalidade algum *software* comercial como o RealViz MatchMover ou o Autodesk Image Modeler 2009, no caso de cenas reais. Caso se deseje tratar cenas sintéticas, pode-se usar o algoritmo de Facet-ID (Vacchetti, V. Lepetit, & P. Fua, 2004b) para calcular a reprojeção.

O algoritmo Facet-ID é definido como a geração de uma identificação única para cada faceta do objeto. A partir disto, a descoberta de cada ponto 3D correspondente a cada ponto 2D ocorre segundo o *pipeline*: gera-se uma cor única (identificador) para cada faceta do objeto; projeta-se o objeto utilizando a pose conhecida; amostra-se o ponto 2D na imagem

gerada; traça-se um raio a partir da câmera através do ponto em questão e encontra-se a interseção com a faceta correspondente do modelo. O diagrama esquemático do algoritmo Facet-ID pode ser visto na Figura 4-3.

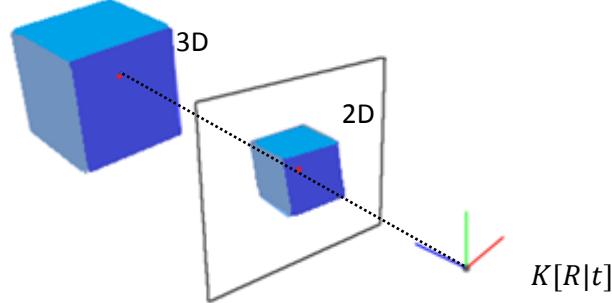


Figura 4-3 Algoritmo Facet-ID: cada faceta do objeto possui uma cor única e, a partir da amostragem do ponto 2D (vermelho), pode-se encontrar o ponto 3D correspondente (vermelho) através do raio gerado a partir da câmera ($K[R|t]$) e sua interseção com a faceta correspondente.

Considerando as facetas como sendo triângulos, o raio pode ser calculado facilmente através do centro da câmera $C = K[R|t]$ e do ponto 2D $m = (u, v, 1)$ pois o centro da câmera C_{opt} (origem do raio r) é dado por

$$C_{opt} = -(R^t t), \quad (13)$$

e a direção do raio r é dada por

$$\vec{r} = (KR)^{-1}m. \quad (14)$$

Para realizar o rastreamento através da informação espacial, apenas alguns *Keyframes* precisam ser extraídos da cena. Segundo (Vacchetti, V. Lepetit, & P. Fua, 2003) aproximadamente 5 *keyframes* são necessários para rastrear um objeto poligonal num fundo não muito poluído (ver Figura 4-4).

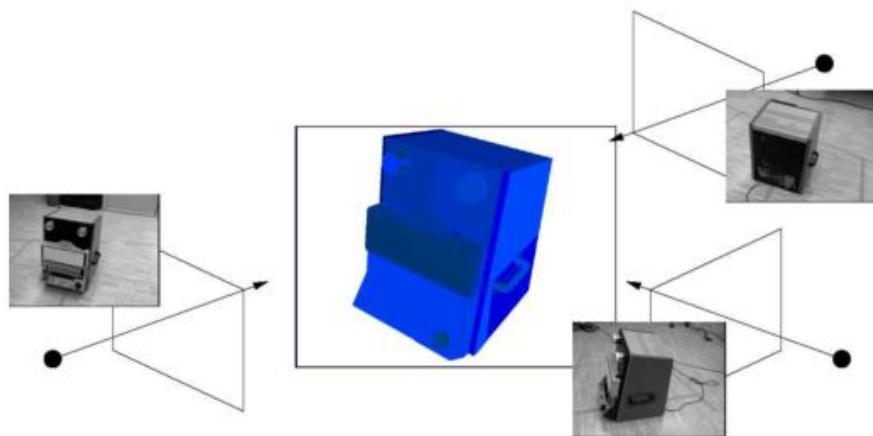


Figura 4-4 Apenas alguns *keyframes* são necessários para realizar o rastreamento. Figura extraída de (Vacchetti, V. Lepetit, & P. Fua, 2004b).

4.2.2 Escolha de *Keyframe* mais Próximo

Para realizar o correto rastreamento da cena na técnica Spatial IPB é necessário selecionar o *keyframe* que está mais próximo do quadro atual (quadro a ser rastreado). Esta descoberta se faz fundamental devido à necessidade de relacionar informações do *keyframe* com o quadro atual para recuperação da pose. Uma vez que a pose do quadro atual ainda não foi estimada, é possível utilizar o *keyframe* que mais se aproxima do quadro anterior da sequência (como o movimento é suave, o quadro anterior funciona como uma boa estimativa do quadro atual, uma característica inerente às técnicas recursivas).

Para realizar a escolha do *keyframe* mais próximo, duas são as alternativas: a utilização da análise do histograma ou a utilização da distância de Mahalanobis entre as poses. Ambas as técnicas serão detalhadas a seguir.

4.2.2.1 *Histograma*

A utilização do histograma para selecionar o *keyframe* mais próximo se baseia no algoritmo do Facet-ID. Esta abordagem, definida em (Vacchetti, V. Lepetit, & P. Fua, 2003) compara o histograma gerado pela projeção de um modelo após a coloração do mesmo pelo Facet-ID. O que ocorre neste caso é a construção de um histograma que conta quantos *pixels* projetados são de cada faceta do objeto, já que cada faceta possui uma cor única. Através de uma comparação simples entre os histogramas, é possível definir o *keyframe* que mais se aproxima

do quadro anterior (boa estimativa do quadro atual), já que a comparação é feita comparando o quadro anterior com todos os *keyframes*.

É importante observar a coerência do método do histograma para escolha do *keyframe*. Uma vez que cada faceta possui sua própria cor única, a contagem dos *pixels* revela quanto de cada faceta do objeto está visível em determinada pose. Já que poses mais próximas tendem a revelar e esconder as mesmas facetas em ambas as cenas, a contagem nada mais é do que uma análise do quanto parecida é a posição da câmera em cada cena através de informação geométrica (ver Figura 4-5).

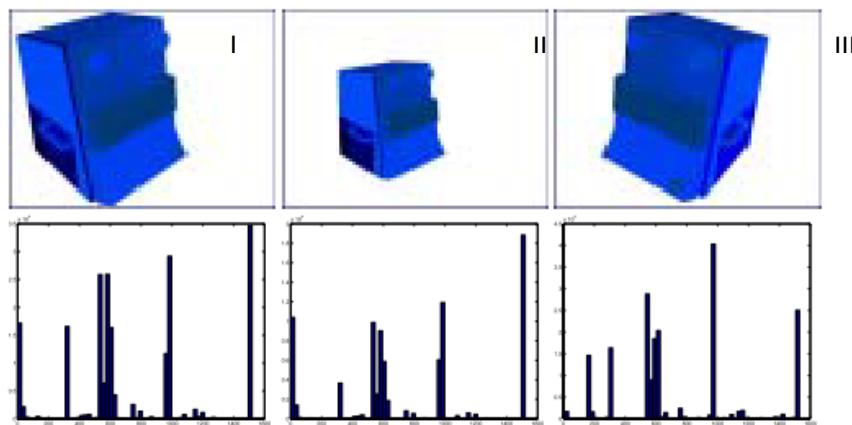


Figura 4-5 Histogramas gerados a partir de diferentes poses do mesmo objeto. É importante notar a semelhança entre os histogramas de poses mais próximas (I e II) e sua diferença para o histograma (III). Figura extraída de (Vacchetti, V. Lepetit, & P. Fua, 2003).

4.2.2.2 Distância de Mahalanobis

Já que para escolher o *keyframe* mais próximo é utilizada a pose do quadro anterior, uma possível abordagem é a utilização da distância de Mahalanobis entre as poses. A distância de Mahalanobis é um método de comparação estatística bastante utilizado na literatura para medir grandezas vetoriais ponderadas. Tal abordagem se prevalece da covariância das dimensões dos vetores para realizar uma comparação embasada na diferença entre cada dimensão, ponderada pela variância do conjunto amostrado. A distância de Mahalanobis entre dois vetores (neste caso dois vetores de pose com 6 dimensões, conforme seção 2.1) é dada por:

$$dist_{mah}(P_{ant}, P_i) = (P_{ant} - P_i)^t M (P_{ant} - P_i), \quad (15)$$

onde P_{ant} e P_i são vetores que representam uma pose cada (P_i representa um *Keyframe* e P_{ant} representa o quadro anterior) e M é a matriz de covariância de dimensão 6×6 dada por:

$$M = \frac{1}{n-1} \sum_{i=1}^n (P_i - \bar{P})(P_i - \bar{P})^t, \quad (16)$$

para o conjunto $P_Key = \{P_1, P_2, \dots, P_n\}$ das poses dos *Keyframes* com média \bar{P} .

Para realizar a escolha, basta utilizar a distância de Mahalanobis entre a pose do quadro anterior e todos os *keyframes* do conjunto P_Key , escolhendo o *keyframe* com a menor distância para o quadro anterior.

Utilizando esta abordagem é possível realizar a escolha do *keyframe* de forma bastante rápida já que computacionalmente o custo depende apenas do cálculo de uma matriz de covariância única para todos os *keyframes* e algumas multiplicações de matrizes para cada *keyframe*. Apesar de mais rápida que a abordagem baseada em histograma e de ser invariante a escala, a utilização da distância de Mahalanobis não leva em consideração o modelo (aspecto geométrico), utilizando apenas um quesito algébrico na escolha do *keyframe* mais próximo, podendo levar a pequenas incoerências na escolha em quadros intermediários a pares de *keyframes*.

4.2.3 Cálculo de *Keyframe* Intermediário

As poses do *keyframe* escolhido e do quadro atual podem não estar próximas o suficiente para que a etapa de casamento dos pontos de interesse (seção 4.2.1) possa ser realizada. Graças a isto, um *keyframe* sintético (chamado *keyframe* intermediário), com uma pose mais próxima do quadro atual, é gerado a partir do *keyframe* escolhido. Para realizar tal transformação, um *patch* em torno de cada ponto de interesse do *keyframe* é transferido para o *keyframe* intermediário através da aplicação de uma homografia (Hartley & Zisserman, 2004). Dada uma estimativa da matriz de projeção atual $K(R_{ant}|t_{ant})$ (dada pelo quadro anterior), um *keyframe* com matriz de projeção conhecida $K(R_k|t_k)$ (dada pelo *keyframe* escolhido) e um plano π aproximado pelo *patch* pertencente à superfície do objeto com normal \vec{n} e distância à origem d conhecidos, a homografia H é escrita como

$$H = K(\delta R - \delta t \vec{n}'^t / d') K^{-1}, \quad (17)$$

com

$$\delta R = R_{ant}R_k^t; \delta t = R_{ant}R_k^t t_k + t_{ant}; \vec{n}' = R_k \vec{n}; d' = d - t_k^t(R_k \vec{n}). \quad (18)$$

A homografia calculada H é então utilizada para transformar os pontos m pertencentes ao *patch* em torno de m_0 (ponto do *keyframe*) para os pontos m' do *keyframe* intermediário de forma que $m' = Hm$, como pode ser visto na Figura 4-6.

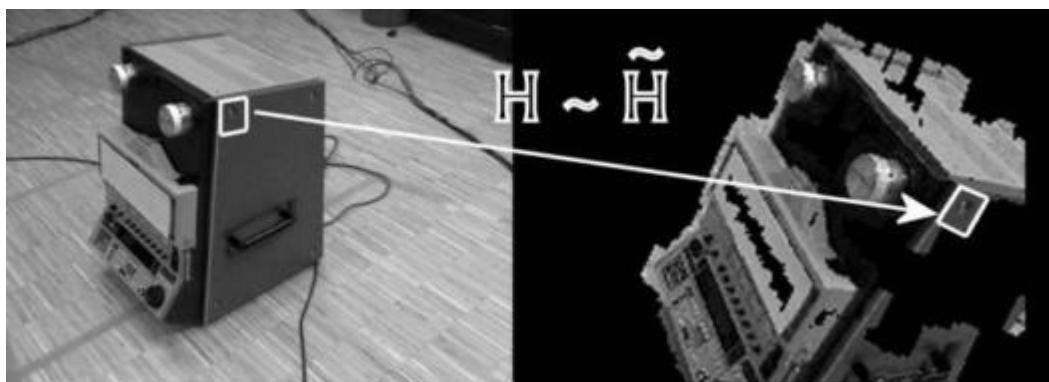


Figura 4-6 *Keyframe* intermediário (direita) gerado a partir da homografia H aplicada a *patches* do *keyframe* mais próximo (esquerda). Figura extraída de (Vacchetti, V. Lepetit, & P. Fua, 2004b)

4.2.4 Extração de Características

As características utilizadas para realizar o rastreamento são geralmente responsáveis por determinar o nome da técnica de rastreamento. No caso do rastreamento Interest Point Based (baseado em pontos de interesse), a informação extraída e utilizada são os pontos de interesse, informações provenientes da textura da imagem, caracterizadas nesta dissertação através do detector de cantos de Harris (Harris & Stephens, 1988).

O detector de cantos de Harris extrai a informação de canto considerando características importantes a um detector que vai ser utilizado como extrator de pontos de interesse (V. Lepetit & P. Fua, 2005): utilizar pontos que diferem de seus vizinhos imediatos, a área em torno do ponto deve ser texturizada para facilitar o casamento e é importante evitar pontos de padrões repetitivos para minimizar casamentos errôneos. Para implementar estas características, o detector de cantos de Harris utiliza uma abordagem baseada em auto-correlação, definida pelos próprios autores a partir da aplicação de uma matriz de autovalores da imagem.

A extração de características do algoritmo retorna uma lista de pontos 2D correspondentes aos pontos extraídos da imagem (ver Figura 4-7). Tais pontos são utilizados

na etapa seguinte do *pipeline* sendo relacionados com pontos 2D extraídos da imagem do *keyframe* intermediário (automaticamente obtidos a partir da aplicação da homografia aos pontos do *keyframe* mais próximo).

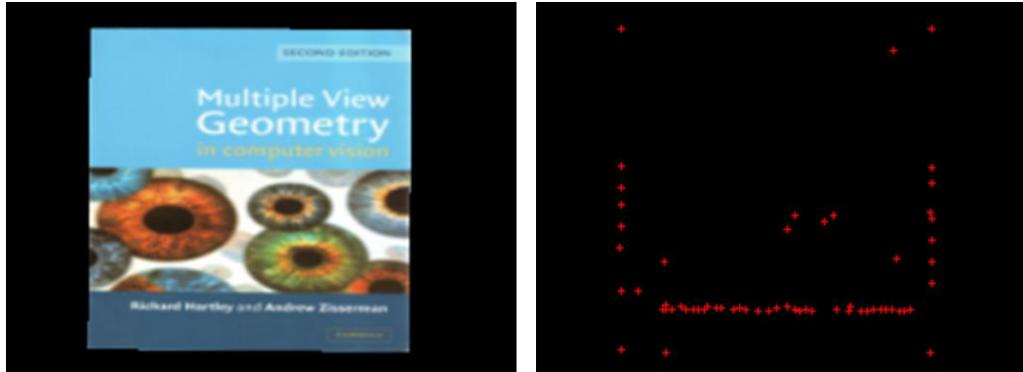


Figura 4-7 Pontos de interesse extraídos (direita) de quadro a ser rastreado (esquerda) através da utilização do detector de cantos de Harris.

4.2.5 Casamento de Características

Uma vez extraídas as características do quadro atual (quadro a ser rastreado) e do *keyframe* intermediário (realizado já na criação do *keyframe* intermediário), faz-se necessário realizar o casamento das características para permitir o correto cálculo da pose da câmera. Para realizar o casamento, é possível utilizar a técnica proposta por (Zhang, Deriche, Faugeras, & Luong, 1995). Nesta técnica, o nível de similaridade de um ponto de interesse $m_1(x_1, y_1)$ da imagem I_1 e um ponto de interesse $m_2(x_2, y_2)$ da imagem I_2 é determinado por seu NCC (*normalized cross correlation*), que é dado por

$$NCC(m_1, m_2) = \frac{\sum_{i=-n}^n \sum_{j=-n}^n I_1(x_1 + i, y_1 + j) \cdot I_2(x_2 + i, y_2 + j)}{\sqrt{\sum_{i=-n}^n \sum_{j=-n}^n I_1(x_1 + i, y_1 + j)^2 \cdot I_2(x_2 + i, y_2 + j)^2}} \quad (19)$$

De forma esquemática, pode-se observar o NCC como uma técnica que analisa a área (*patch*) na qual o pixel se encontra (de tamanho $n \times n$) com relação a área da imagem correspondente candidata para decidir se o casamento é válido.

Supondo k o conjunto de pixels extraídos pelo detector de Harris na imagem I_1 e l o conjunto de pixels extraídos na imagem I_2 , o algoritmo busca para um pixel $a \in k$ qual o pixel α do conjunto l que possui menor NCC com relação ao pixel a . Isto é feito para todos os pixels da imagem I_1 (busca no sentido $I_1 \rightarrow I_2$) dando origem aos pares

$\psi_1 = \{(a, \alpha), (b, \gamma), (c, \beta) \dots\}$. Em seguida, a mesma busca é feita no sentido oposto ($I_2 \rightarrow I_1$) gerando o conjunto $\psi_2 = \{(\alpha, a), (\gamma, c), (\beta, b) \dots\}$. Após geração de ambos os conjuntos (possíveis casamentos), é feita uma análise para saber que pixels são pares válidos (pares são considerados válidos quando o par da busca no sentido $I_1 \rightarrow I_2$ é o mesmo do sentido $I_2 \rightarrow I_1$). No exemplo dado, o par (a, α) pode ser considerado um casamento válido, pois $(a, \alpha) \in \psi_1$ e $(\alpha, a) \in \psi_2$. A análise esquemática pode ser vista na Figura 4-8.

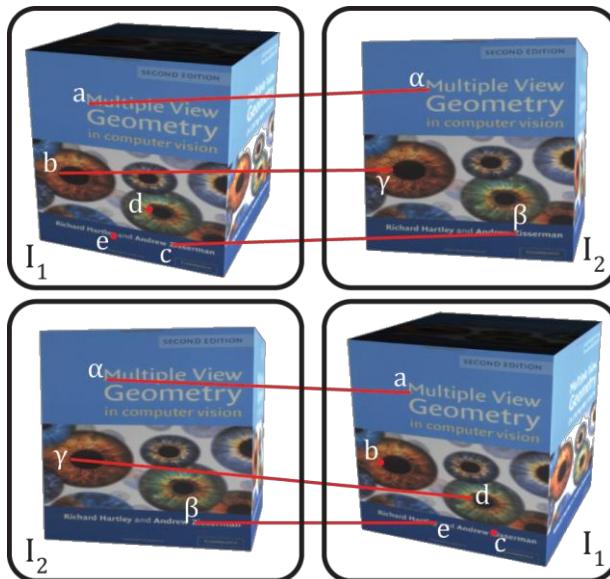


Figura 4-8 Casamento de características. Pontos casados são considerados válidos quando mesmo casamento ocorre tanto no sentido $I_2 \rightarrow I_1$ quanto no sentido $I_1 \rightarrow I_2$. No exemplo, o único casamento válido é o par (a, α) .

4.2.6 Cálculo de Pose

Uma vez encontradas as correspondências entre os pontos extraídos do quadro atual e do *keyframe* intermediário (correspondências 2D) é possível realizar o efetivo cálculo da pose do quadro atual a partir da minimização do erro de reprojeção (ver seção 2.2.2.). Como os pontos 3D correspondentes aos pontos 2D extraídos do *keyframe* intermediário são conhecidos, após relacioná-los com os pontos 2D do quadro atual, pode-se utilizar os pontos 3D do *keyframe* intermediário para achar a pose que melhor casa os pontos 3D do *keyframe* intermediário com os pontos 2D do quadro atual. Um diagrama da solução utilizada pode ser visto na Figura 4-9.

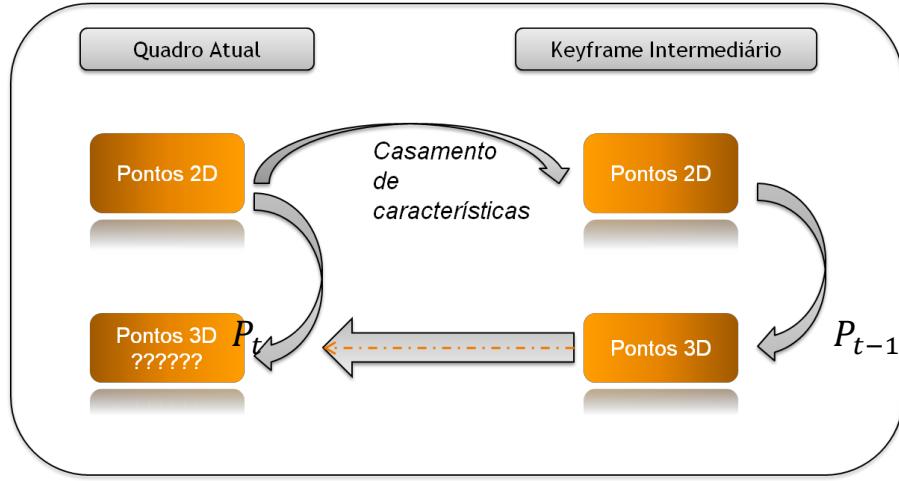


Figura 4-9 Cálculo de pose do quadro atual (P_t) realizado a partir do casamento de características do quadro atual com o *Keyframe* intermediário.

Além de se utilizar do algoritmo não-linear Levemberg-Marquardt para realizar a minimização, a técnica *Spatial IPB* realiza a minimização com o auxilio do M-estimador de Tukey, como descrito na seção 2.2.3, de forma a diminuir a influência de correspondências espúrias (*outliers*), levando a equação:

$$P = [R|t] = \underset{[R|t]}{\operatorname{argmin}} \sum_{i=0}^n \rho_{tuk}(\operatorname{dist}^2(\theta(P, M_i), m_i)), \quad (20)$$

onde os pontos 3D do *keyframe* intermediário (M_i) são relacionados com os pontos 2D do quadro atual (m_i) através da pose a ser calculada ($P = [R|t]$) sendo ponderados pela função de Tukey. Como se trata de uma técnica que precisa de uma estimativa inicial, pode-se utilizar a pose do quadro anterior e iterativamente refiná-la para descobrir a pose atual.

4.2.7 Atualização do Quadro Atual

Após finalizado o cálculo de pose, é possível realizar o aumento da cena em questão para exibir o resultado ao usuário através de técnicas de renderização (um dos pilares de RA não abordados neste trabalho – observar (Duh & Billinghurst, 2008)). Apesar disto, ainda se faz necessária a atualização das informações do quadro atual a partir da pose calculada para utilização como entrada do próximo quadro a ser rastreado.

A atualização do quadro atual utiliza a abordagem do facet-ID, já discutida na seção 4.2.1, através da aplicação da pose conhecida aos pontos 2D extraídos como forma de obter os

pontos 3D do modelo correspondentes. Desta forma, o *pipeline* se encerra para o quadro em questão deixando o sistema pronto para o novo quadro a ser rastreado.

4.3 Temporal IPB

O Temporal IPB pode ser visto como uma simplificação do *pipeline* do Spatial IPB através do descarte e não utilização da informação *offline* oriunda dos *keyframes*. Em contrapartida a informação temporal, proveniente do quadro anterior é utilizada integralmente. Tal abordagem torna a técnica Temporal IPB menos intensa computacionalmente e ao contrário da técnica Spatial IPB que sofre de *jitter* e evita *drift*, a técnica Temporal IPB sofre de *drift* graças a não utilização dos *keyframes* e evita o *jitter* graças a utilização do quadro anterior. O *pipeline* da técnica Temporal IPB pode ser visto na Figura 4-10.

Uma vez observado o *pipeline* da variação Temporal IPB é importante destacar as diferenças básicas com relação à Spatial IPB. A etapa *offline* de geração de *keyframes* deixa de existir, simplificando a inicialização do sistema mas não impactando no desempenho por fazer parte da inicialização. Já as etapas de escolha do *keyframe* mais próximo e geração do *keyframe* intermediário passam a ser substituídas pela extração de características do quadro anterior que já foi previamente efetuada já que o quadro anterior era o quadro atual na iteração anterior. Já a extração de características do quadro anterior continua existindo.

Uma etapa importante que sofre leves mudanças é o cálculo da pose. Apesar de utilizar a mesma base discutida na seção 2.2.2, sendo realizada a partir da minimização do erro de reprojeção, a entrada da etapa é modificada, substituindo a informação *offline* pela *online*. Desta forma as correspondências utilizadas passam a ser as do quadro anterior e não do *keyframe* mais próximo. Esta substituição define tanto o nome desta variação da técnica IPB quanto a ocorrência de *drift*.

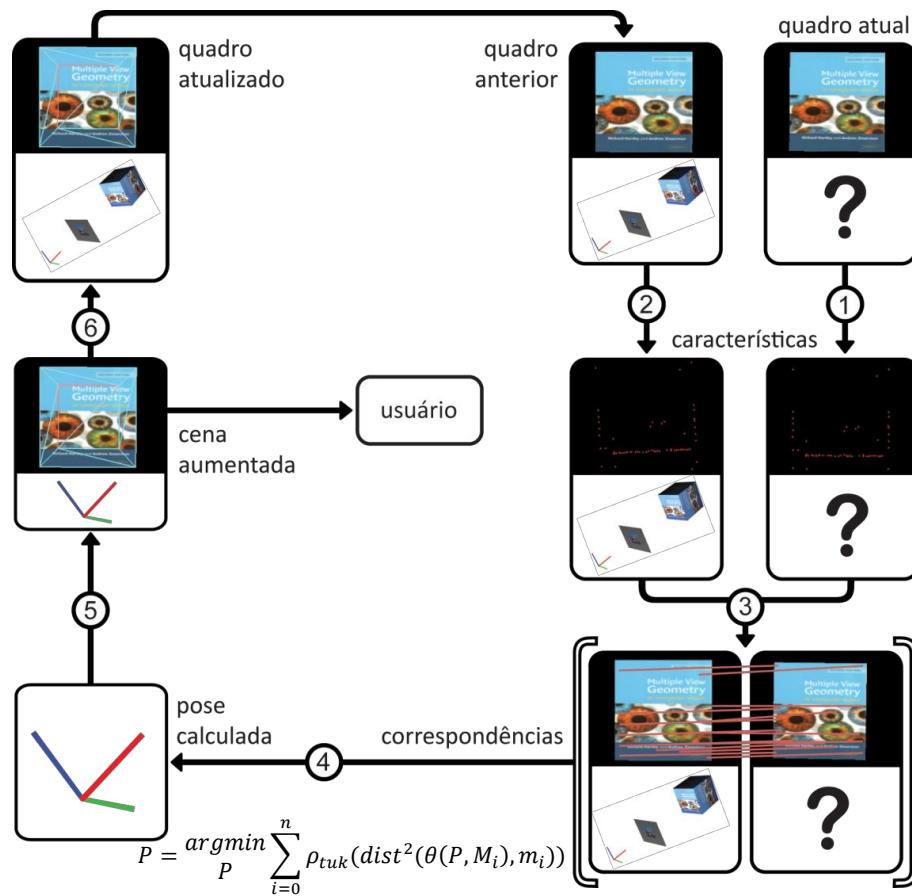


Figura 4-10 Rastreamento utilizando *Temporal IPB*. Primeiramente são extraídas as características do quadro atual (passo 1) para serem casadas com as características do quadro anterior (passo 2, já extraídas no processamento do quadro anterior). Em seguida é feito o casamento das características do quadro anterior com o quadro atual (passo 3) para ser utilizado no cálculo de pose (passo 5). Após a pose calculada é feita a exibição da aplicação de RA para o usuário (passo 6) e em seguida é feita a atualização do quadro atual para utilização no próximo quadro como quadro anterior (passo 7).

Assim como a Spatial IPB, é necessária uma atualização das informações do quadro atual mesmo após o rastreamento realizado, para que o quadro atual possa ser utilizado como quadro anterior na próxima iteração e todo o ciclo seja repetido, através da utilização do Facet-ID para descoberta das correspondências entre os pontos 2D extraídos pelo detector de Harris e os pontos 3D do modelo.

4.4 Spatial Temporal IPB

Com o objetivo de evitar tanto *jitter* quanto *drift*, é proposto em (Vacchetti, V. Lepetit, & P. Fua, 2004b) uma abordagem mista, através da utilização tanto da informação do quadro anterior quanto dos *keyframes*. Tal abordagem, chamada Spatial Temporal IPB evita o *jitter* e o

drift combinando as informações obtidas *online* e *offline* no cálculo de pose.

O *pipeline* da variação Spatial Temporal IPB é descrito na Figura 4-11. É importante notar a complementação entre as informações oriundas do quadro anterior e dos *keyframes* através da criação das duas listas de correspondências utilizadas como entrada para o cálculo da pose. Ao contrário do que se imagina num primeiro momento, as informações temporal (*online*) e espacial (*offline*) não são trivialmente complementares, mas podem ser utilizadas de forma combinada com eficiência, conforme descrito na seção seguinte.

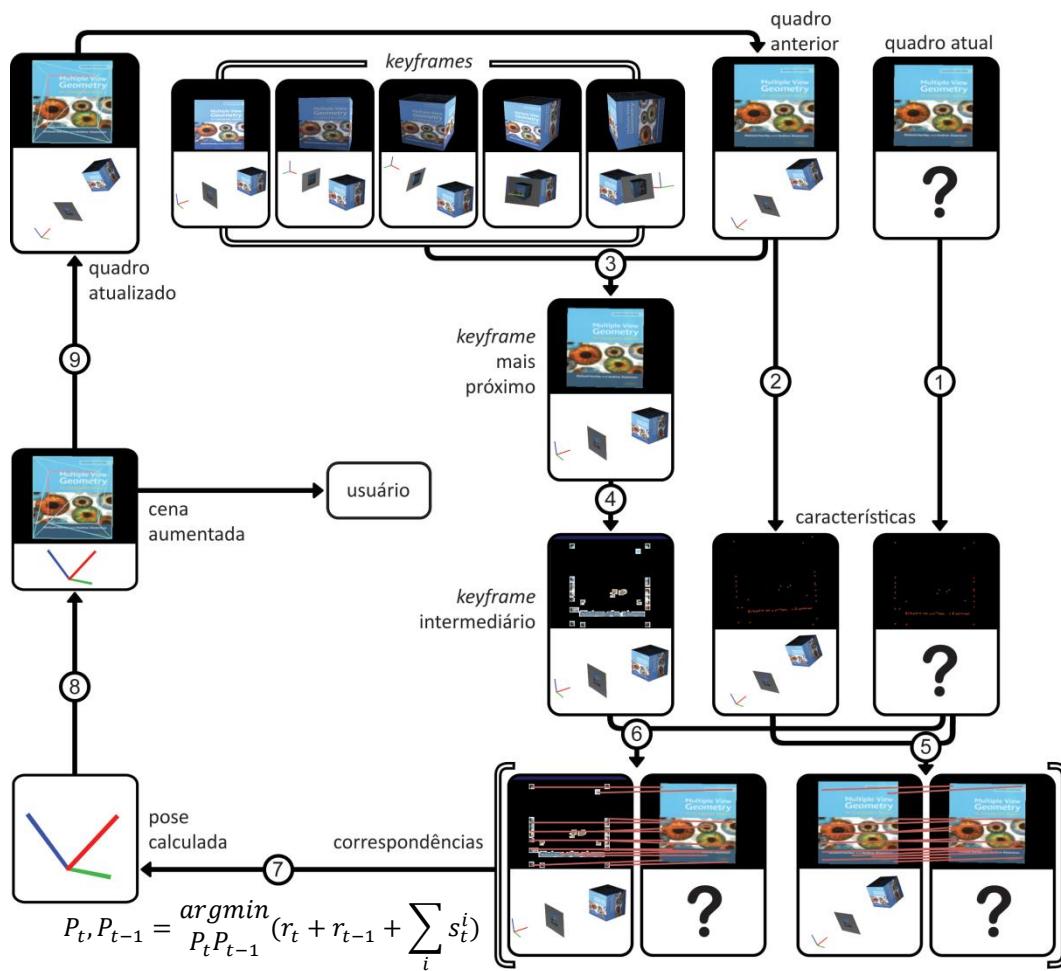


Figura 4-11 *Pipeline* da técnica Spatial Temporal IPB. Inicialmente são extraídas as características tanto do quadro atual (passo 1) quanto do quadro anterior (passo 2 pré-computado). Em seguida, é realizada a escolha do *keyframe* mais próximo (passo 3) para utilização na geração do *keyframe* intermediário (passo 4). Terminadas as extrações de características e a geração do *keyframe* intermediário, é feito o cálculo das correspondências do quadro atual com o quadro anterior (passo 5) e com o *keyframe* intermediário (passo 6). Em seguida, é feito o cálculo de pose (passo 7), mais complexo e dependente de uma série de processamentos adicionais se comparado com as variações preliminares da técnica. Finalmente, é realizado o aumento da cena (passo 8) e a atualização das informações do

quadro atual (passo 9) para utilização no próximo quadro da cena.

4.4.1 Cálculo de Pose

Na maioria dos casos, somar duas fontes de características leva a uma degradação do resultado do rastreamento ao invés de melhorá-lo, segundo (Vacchetti, V. Lepetit, & P. Fua, 2004a). Desta forma, para utilização de múltiplas fontes (*keyframe* intermediário e quadro anterior), se faz necessária uma combinação das fontes como proposto por (Vacchetti, V. Lepetit, & P. Fua, 2004b).

A combinação das informações de textura extraídas do quadro anterior e do *keyframe* intermediário se dá também pela observação de pontos em comum a ambos os casamentos. Pontos extraídos do *keyframe* intermediário que podem ser correspondidos com o quadro anterior complementam a formação de uma base de dados mais confiável para o cálculo de pose.

O cálculo de pose utilizando as informações do quadro anterior e do *keyframe* intermediário se dá minimizando a equação:

$$P_t, P_{t-1} = \underset{P_t P_{t-1}}{\operatorname{argmin}} (r_t + r_{t-1} + \sum_i s_t^i), \quad (21)$$

onde:

$$r_{t-1} = \underset{P_{t-1}}{\operatorname{argmin}} \sum_{i=0}^n \rho_{tuk}(\operatorname{dist}^2(\theta(P_{t-1}, M_i), m'_i)), \quad (22)$$

$$s_i = \rho_{tuk} \left(\operatorname{dist}^2(\psi(P_{t-1}, P_t, m_i), m'_i) + \operatorname{dist}^2(\psi'(P_t, P_{t-1}, m'_i), m_i) \right). \quad (23)$$

As equações r_t e r_{t-1} originam, respectivamente, *keyframe* intermediário e quadro anterior e os elementos s_i são representantes dos pontos em comum do quadro anterior e do *keyframe* intermediário numa análise dependente da homografia ψ induzida pela faceta do modelo que contém o ponto i , ver (Vacchetti, V. Lepetit, & P. Fua, 2004b).

4.4.1.1 IPB + RANSAC

Para tentar minimizar a influência dos *outliers* e melhorar a eficiência da técnica, pode-se utilizar a abordagem de RANSAC, como descrito na seção 2.2.3.1, gerando várias poses através da aplicação do cálculo a vários subconjuntos (amostras) do conjunto principal de resíduais

(diferença entre pontos 3D reprojetados e pontos 2D extraídos). Tais subconjuntos, gerados de forma aleatória (amostragem), são responsáveis por realizar o cálculo da pose de forma independente e o melhor subconjunto tem sua pose escolhida como pose correta. A utilização de subconjuntos, apesar de acrescentar computações, acaba sugerindo melhor desempenho devido a rápida convergência em decorrência do baixo número de residuais.

4.5 IPB + Filtro de Partículas

Abordagens estatísticas são comumente utilizadas em associação com técnicas de rastreamento para incrementar sua robustez e facilitar a recuperação de falhas. A abordagem de filtro de partículas, assim como descrito na seção 2.3.1, se baseia em representar a função de densidade de probabilidade (fdp) posteriori $p(\mathbf{x}_k, \mathbf{z}_{1:k})$ do estado \mathbf{x}_k no quadro atual (estado atual k) a partir de um conjunto finito $\{\mathbf{x}_k^i, \pi_k^i\}_{i=0..N}$ de N amostras (partículas) em que cada partícula \mathbf{x}_k^i está diretamente associada a um peso π_k^i . De forma resumida, o filtro de partículas se propõe a encontrar a pose do novo quadro do sistema (quadro atual) a partir de medidas ruidosas (partículas) oriundas de quadros anteriores e seu *pipeline* pode ser observado na Figura 4-12.

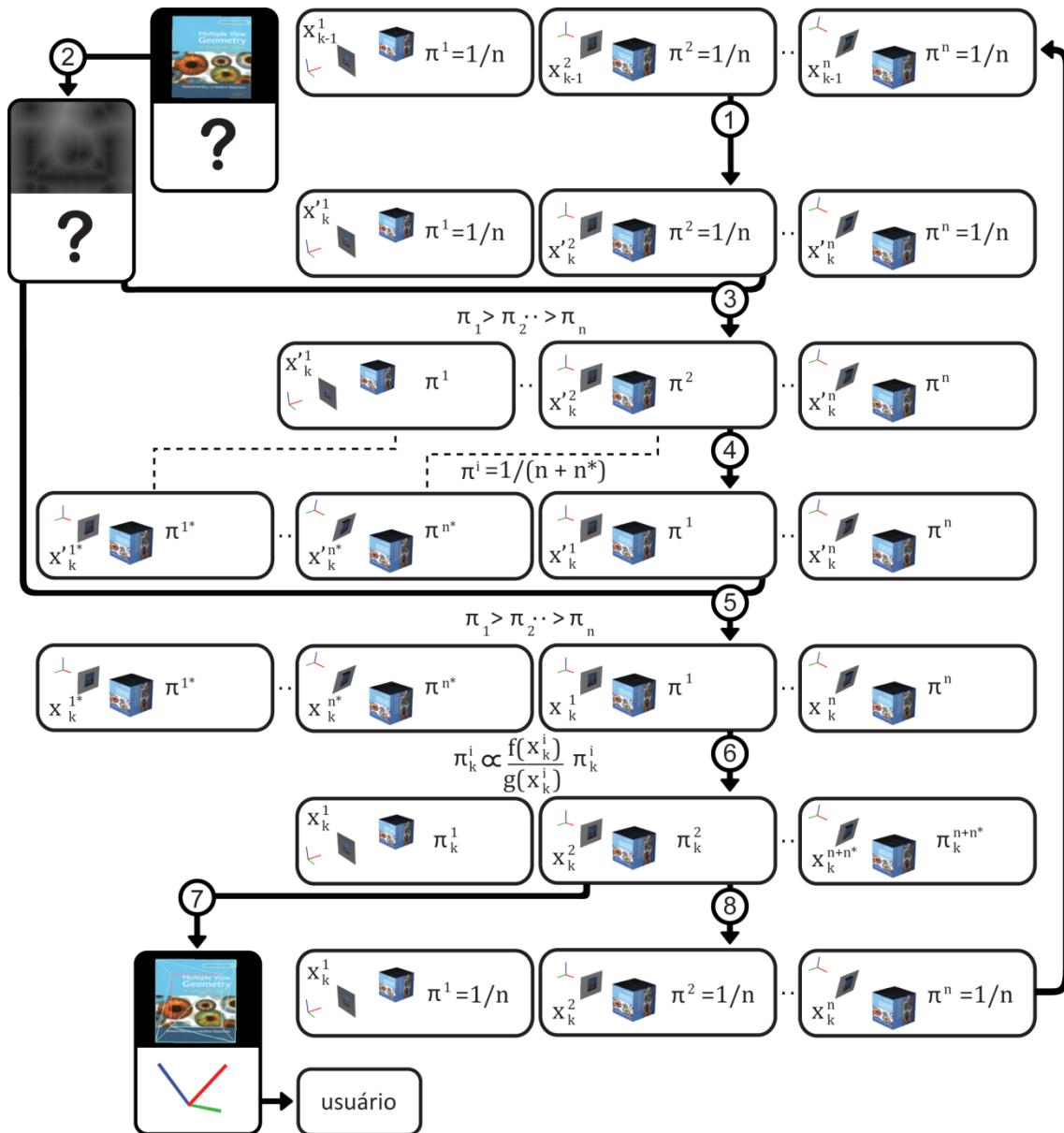


Figura 4-12 Rastreamento utilizando IPB + Filtro de Partículas. O conjunto inicial de partículas $\{x_{k-1}^i, \frac{1}{N}\}_{i=0..N}$ é expandido para geração do conjunto $\{x'_k^i, \frac{1}{N}\}_{i=0..N}$ (passo 1). Em seguida é feita a extração das características e geração do mapa de distâncias do quadro atual (passo 2). Utiliza-se o mapa de distâncias para calcular a verossimilhança das partículas e ordenar o conjunto (passo 3). O subconjunto das melhores partículas é utilizado para sofrer otimização utilizando o *Temporal IPB* (passo 4). A verossimilhança das novas partículas $\{x_k^{i*}, \frac{1}{N+N^*}\}_{i=0..N^*}$ é calculada e os conjuntos são combinados (passo 5). É feita a normalização das partículas (passo 6) já que a verossimilhança dos dois conjuntos obedecem a funções diferentes (as novas partículas obedecem a uma função f e o conjunto total obedece a função g). Em seguida é feito o cálculo de pose ponderado pelos pesos $\{\pi_k^1, \pi_k^2 \dots \pi_k^{N+N^*}\}_{i=0..N+N^*}$ a partir das partículas $\{x_k^1, x_k^2 \dots x_k^{N+N^*}\}_{i=0..N+N^*}$. Finalmente é realizada a reamostragem para manutenção das melhores partículas.

4.5.1 Partículas

A unidade básica para a utilização do *framework* estatístico de FP, como discutido na seção 2.3.1, é a definição do que são as partículas do sistema a ser tratado. Uma vez que a técnica de rastreamento foca na recuperação das informações de câmera, é natural que as partículas sejam formadas por tais informações. Para utilização integrada ao IPB, é proposta a representação da partícula pela pose da câmera, definida na seção 2.1 por um vetor de seis dimensões, no qual as três primeiras correspondem à rotação na representação de Rodrigues e as três últimas correspondem à translação.

Por se tratar de uma técnica recursiva, a pose do primeiro quadro é conhecida. Desta forma, é a partir dele que o conjunto de partículas inicial é gerado aplicando-se a mesma abordagem utilizada na fase de expansão das partículas, discutida na seção 4.5.2, dando origem ao conjunto inicial $\{x_{k-1}^i, \pi_k^i\}_{i=0..N}$ no qual $\pi_k^i = 1/N$ para todas as partículas.

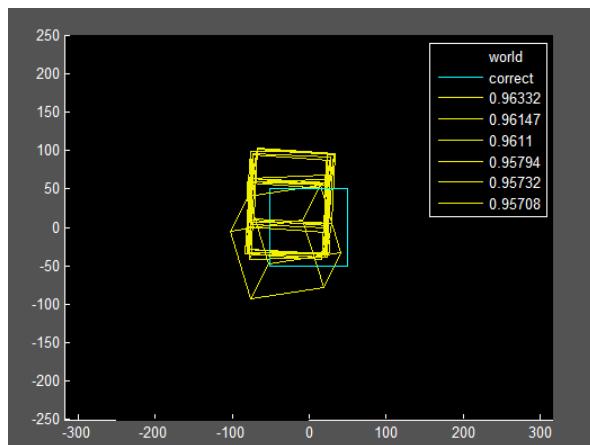


Figura 4-13 Partículas geradas aleatoriamente (tons de amarelo) sobrepostas à pose correta do primeiro quadro da cena (ciano) durante a inicialização do sistema.

4.5.2 Expansão das Partículas

Como descrito na seção 2.3.1, as técnicas que utilizam o *framework* estatístico de FP, necessitam de um modelo estatístico dinâmico, de forma a realizar a expansão das partículas do sistema $\{x_{k-1}^i, \frac{1}{N}\}_{i=0..N}$ provenientes do quadro anterior rastreado representadas como a tupla (partícula, peso) em que cada partícula nesta etapa possui o mesmo peso $\frac{1}{N}$. O modelo estatístico utilizado nesta dissertação é a expansão a partir de ruído gaussiano, aplicado

diretamente em cada dimensão das partículas anteriores (a pose é um vetor de seis dimensões) gerando o novo conjunto de partículas $\{x'_k, \frac{1}{N}\}_{i=0..N}$. O ruído gaussiano é aplicado através da representação matricial pela equação:

$$x'_k = M * x_{k-1}^i, \quad (24)$$

onde M é uma matrix 6×6 anti-simétrica da forma:

$$M = \exp\left(\sum_{i=1}^6 \alpha_i G_i\right), \alpha \sim \mathcal{N}(0, \sigma^2 I_6), \quad (25)$$

na qual G_i é uma base de coordenadas exponenciais de $se(3)$.

Também é válido observar que por se tratar de uma transformação formada a partir de coordenadas exponenciais, ela só é capaz de representar fielmente rotações de poucos graus, levando à conclusão de que a expansão das partículas leva o conjunto de partículas a um conjunto de partículas próximas das originais (em torno das partículas originais).

4.5.3 Análise das Partículas (Cálculo de Verossimilhança)

Após obtido o conjunto ampliado de partículas, é preciso mensurar o quanto boa é cada partícula do conjunto $\{x'_k, \frac{1}{N}\}_{i=0..N}$ através do cálculo de verossimilhança. É neste passo do processo que o conjunto de partículas é reduzido para que o desempenho atinja níveis aceitáveis, já que a quantidade de partículas é fator determinante desta métrica (apesar do tratamento das partículas ser facilmente paralelizável) e para que sejam utilizadas no restante do *pipeline* as melhores partículas (organizando o conjunto da maior verossimilhança para a menor).

A partir do cálculo de verossimilhança proposto por (Teuliere, Eric Marchand, & Eck, 2010), foi proposta nesta dissertação uma adaptação para sua utilização com informações (características) provenientes de textura sob a forma de pontos de interesse. Enquanto em (Teuliere, Eric Marchand, & Eck, 2010) a verossimilhança é oriunda da distância dos pontos projetados à informação de aresta, a técnica proposta nesta dissertação se utiliza da distância de pontos projetados do modelo aos pontos extraídos do detector de cantos de Harris (informação de textura).

Inicialmente, após a extração de características do quadro atual, é feita a geração de um mapa de distâncias. Este mapa de distâncias corresponde a uma imagem de mesmo

tamanho da imagem do quadro atual na qual cada *pixel* possui como valor a menor distância, a partir do *pixel* em questão, para qualquer ponto de interesse extraído pelo detector de Harris. Um esquema da geração do mapa de distâncias pode ser observado na Figura 4-14 e um mapa de distâncias gerado a partir de uma imagem rastreada pode ser observado na Figura 4-15.

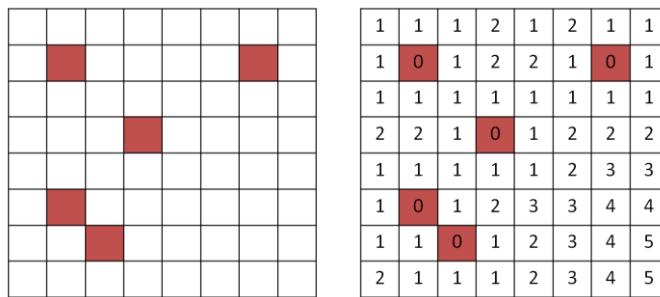


Figura 4-14 Esquema de cálculo do mapa de distâncias de valores inteiros (direita) para uma tabela sintética (esquerda). Cada *pixel* do mapa de distâncias recebe o valor de sua distância ao ponto de interesse mais próximo.

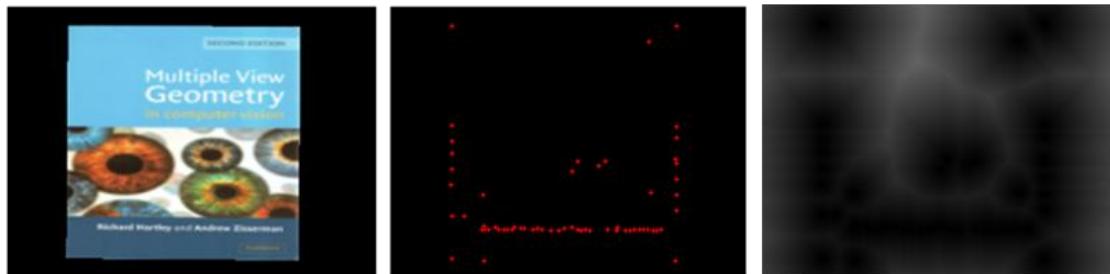


Figura 4-15 Mapa de distâncias com precisão de ponto flutuante (direita) gerado a partir de características extraídas (centro) do quadro atual (esquerda). No mapa de distâncias (direita) pontos mais escuros correspondem a pontos mais próximos de pontos de interesse e pontos mais claros correspondem a pontos mais afastados.

Uma vez criado o mapa de distâncias do quadro atual pode-se utilizar esta informação para calcular a verossimilhança de todas as partículas com a vantagem de que o mapa de distâncias só é computado uma vez a cada quadro rastreado deixando a computação da verossimilhança muito rápida. Para calcular a verossimilhança de uma partícula, é utilizada uma abordagem baseada no somatório das distâncias das projeções dos pontos 3D extraídos do modelo em cada partícula às características extraídas do quadro atual, utilizando para isto o mapa de distâncias. A distância média dos pontos relacionados é dada por:

$$d(s) = \frac{1}{N} \sum_{i=1}^N d_i , \quad (26)$$

onde d_i é a distância do ponto $i = (x, y)$ ao ponto extraído mais próximo calculado através da

amostragem no mapa de distâncias da posição (x, y).

Após o cálculo das distâncias de todos os pontos, a verossimilhança da partícula é dada por:

$$p(\mathbf{z}_{1:k}, \mathbf{x} = \mathbf{x}_k) = \begin{cases} e^{-\lambda(d(s)-d_{min})} & \text{se } d_{max} \neq d_{min} \\ 1 & \text{outros casos.} \end{cases} \quad (27)$$

Quanto maior a verossimilhança melhor a partícula será, ou seja, mais parecida com a pose correta do quadro atual esta partícula é, e devido a isto a lista de partículas é ordenada de forma decrescente com respeito à verossimilhança para simplificar a utilização na próxima etapa.

4.5.4 Otimização das Melhores Partículas

Após a obtenção do conjunto de partículas ordenado por verossimilhança (qualidade das partículas), é realizada a otimização das melhores partículas (só é aplicada a otimização a um subconjunto de partículas devido ao custo computacional) de modo a aperfeiçoar as partículas antes de sua utilização no cálculo de pose. Para tanto, é utilizada a técnica de rastreamento associada ao FP e, neste caso, poderia ser utilizado qualquer rastreador como o Temporal IPB, Spatial IPB ou mesmo Spatial Temporal IPB.

Nesta dissertação optou-se pela utilização da técnica Temporal IPB para associação com o FP devido a seu tempo de processamento menor que as outras variações do IPB e sua simplicidade. Já que a computação da otimização é realizada para várias partículas, é de extrema importância acelerar sua computação ao máximo para permitir técnicas de tempo-real, requisito essencial para MAR. Outro ponto importante é a possível ocorrência de *jitter* devido a utilização do *Temporal IPB* que pode ser minimizado tanto pela informação espacial oriunda dos *keyframes* quanto pela informação espacial oriunda de outras fontes de características como as arestas (Teuliere, Eric Marchand, & Eck, 2010), levando a possibilidade de se trabalhar com técnicas híbridas (Vacchetti, V. Lepetit, & P. Fua, 2004a).

4.5.5 Normalização do Conjunto de Partículas (Atualização dos Pesos)

Uma vez otimizadas as melhores partículas, é novamente realizada a análise das partículas através do cálculo de verossimilhança das novas partículas (otimizadas). Em seguida, a etapa de normalização é iniciada, uma vez que o conjunto de partículas otimizadas ($\{\mathbf{x}_k^{*i}\}_{i=0..N^*}$) não

obedece a mesma distribuição de probabilidades que o conjunto inicial $\{x_k^i\}_{i=0..N}$ segundo a *importance sampling theory* (Arulampalam, Maskell, Gordon, & Clapp, 2002). O novo conjunto total $\{x_k^i\}_{i=0..N+N^*}$ formado pela junção do conjunto inicial com o conjunto das partículas otimizadas é corrigido a partir de um termo de normalização aplicado aos pesos das partículas da forma f/g onde f é originado do conjunto inicial (função definida pelo conjunto original de partículas) e g é originado do conjunto total (função definida pelo conjunto total de partículas).

Como definido na teoria de FP, as funções $f_k(x_k)$ e $g_k(x_k)$ são dadas por misturas de gaussianas da forma:

$$f_k(x_k) = \frac{1}{N} \sum_i^N \mathcal{N}(x_k^i, \Sigma)(x_k), \quad (28)$$

$$g_k(x_k) = \frac{1}{N + N^*} \left(\frac{1}{N} \sum_i^N \mathcal{N}(x_k^i, \Sigma)(x_k) + \frac{1}{N^*} \sum_i^{N^*} \mathcal{N}(x_k^{*i}, \Sigma)(x_k) \right), \quad (29)$$

onde $x_k^i \in \{x_k^i\}_{i=0..N}$, $x_k^{*i} \in \{x_k^{*i}\}_{i=0..N^*}$ e x_k é qualquer partícula analisada pertencente ao conjunto $\{x_k^i\}_{i=0..N}$ ou ao conjunto $\{x_k^{*i}\}_{i=0..N^*}$.

4.5.6 Cálculo de Pose

Uma vez encontrados os novos pesos $\left(\frac{f_k(x_k)}{g_k(x_k)} p(\mathbf{z}_{1:k}, \mathbf{x} = \mathbf{x}_k) \right)$ das partículas do sistema, o cálculo de pose se origina da média ponderada das partículas do sistema.

Já que a operação de adição no conjunto de rotações $SO(3)$ não é uma operação binária, calcular a média de uma série de rotações não é tão simples quanto calcular a média das translações (média simples dada por uma média aritmética: $T_m = \bar{T} = \frac{1}{N} \sum_{i=1}^N T_i$). Para tanto, é necessário utilizar uma aproximação para calcular a rotação média do conjunto R_m a partir da média aritmética das rotações \bar{R} .

Dado um conjunto de partículas $\{x_k^i, \pi_k^i\}_{i=0..N}$, onde cada partícula possui uma pose x_k^i e é formada por uma rotação R_i e uma translação T_i , a média aritmética das rotações \bar{R} pode ser calculada como $\bar{R} = \frac{1}{N} \sum_{i=1}^N R_i$. O problema reside no fato de que a nova rotação calculada (\bar{R}) não é necessariamente uma rotação. Para garantir que a matriz resultante seja de fato

uma rotação, é feita a decomposição em valores singulares (SVD) da média aritmética calculada \bar{R} e é extraído o termo Σ , correspondente aos autovalores do sistema representado pela rotação \bar{R} . Observando em detalhes isto significa que a decomposição de \bar{R} resulta em $U\Sigma V$, que é utilizada na composição da rotação média R_m da forma:

$$R_m = \begin{cases} VU^t & \text{se } \det(\bar{R}) > 0 \\ VHU^t & \text{caso contrário,} \end{cases} \quad (30)$$

onde $H = \text{diag}[1, 1, -1]$.

4.5.7 Reamostragem

Com o objetivo de minimizar a ocorrência de degeneração do FP (partículas de baixo peso serem acumuladas e usadas ao longo dos quadros rastreados) é realizada a etapa de reamostragem. Uma das formas mais comuns de realizar a reamostragem é utilizando a redistribuição das partículas de acordo com seu peso (Arulampalam, Maskell, Gordon, & Clapp, 2002), aumentando a importância de partículas de maior peso e diminuindo das partículas de menor peso, como pode ser visto na Figura 4-16.

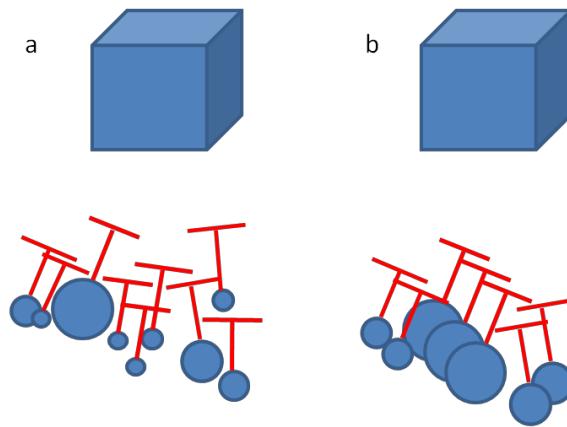


Figura 4-16 Antes (a) e depois (b) da reamostragem. Após a reamostragem, o conjunto de partículas possui maior representatividade das melhores partículas.

5 Resultados

“Você nunca sabe que resultados virão da sua ação. Mas se você não fizer nada, não existirão resultados.” – Mohandas Karamchand Gandhi.

Este capítulo descreve os resultados obtidos nas implementações realizadas nesta dissertação. Foram realizadas implementações das técnicas Temporal IPB, Spatial IPB, Spatial Temporal IPB, IPB + Filtro de Partículas e uma ferramenta para visualização de partículas. Comparações de precisão e desempenho foram realizadas através de análises visuais, gráficas e numéricas das técnicas de rastreamento para indicar o caminho a seguir e uma avaliação acerca da função de verossimilhança a partir de informações de textura também foi realizada para sua validação. Cabe ressaltar que uma análise comparativa das técnicas através da avaliação de seus pontos fortes e fracos pode ser observada no capítulo seguinte (Conclusões).

5.1 Ferramentas e Métodos

As técnicas implementadas nesta dissertação foram avaliadas com respeito à taxas de execução e métricas de precisão. O tempo de execução é de extrema importância na avaliação de técnicas de RA porque se a fatia de tempo utilizada para realizar a estimativa da pose for pequena, o tempo restante pode ser utilizado para outras tarefas como exibição ou interação com usuário. Assim como a taxa de execução, a precisão também é uma métrica indispensável já que é responsável por avaliar a qualidade do rastreamento da pose ao longo da sequência, impactando diretamente em quão bom o aumento da cena será.

O computador utilizado para realizar os testes possui um processador Intel Core Duo 2.39 GHz, 2 GB de RAM, uma placa gráfica NVIDIA GeForce 7900 GTX com 512 MB de memória e resolução da tela de 1024x1024. A *webcam* utilizada na cena real foi a A4Tech ViewCam PK-635, com resolução de 320x240 pixels e uma taxa de 30 quadros por segundo. O sistema operacional presente no computador utilizado nos testes foi o Microsoft Windows 7 Professional Edition. A interface de desenvolvimento (IDE) utilizada foi o Microsoft Visual

Studio .NET 2010 Professional Edition.

Para realizar as implementações foi utilizada a linguagem de programação C++ e a ferramenta Matlab. A decisão pela utilização de C++ se deve à diversidade de bibliotecas de visão computacional e computação gráfica disponíveis (VxL, OpenCV, Visp, CGAL, OpenGL, Direct 3D, entre outras) e, principalmente, as restrições de desempenho existentes no desenvolvimento de aplicações de MAR. Já a ferramenta Matlab foi utilizada principalmente para verificações de resultados, análises estatísticas e visuais devido a simplicidade de trabalho com elementos matriciais, vetoriais e gráficos qualitativos.

As técnicas Spatial IPB, Temporal IPB, Spatial Temporal IPB e IPB + Filtro de Partículas foram implementadas utilizando a VxL, biblioteca de visão computacional que permite de forma simples a utilização de estruturas matriciais, vetores, operações matemáticas, geometria e estatística. Algoritmos como SVD, Levenberg-Marquardt, aplicação de M-Estimadores, cálculo de distribuição normal, entre outros, também possuem implementação na VxL. Detalhes acerca da organização e uso da VxL podem ser encontrados em (“VxL,” 2010).

Para visualização dos resultados em tempo real e acelerar diversos algoritmos como a geração do mapa de distâncias ou o Facet-ID, foi utilizada a biblioteca gráfica OpenGL em conjunto com a VxL. Na visualização, foram sobrepostos modelos aramados dos modelos rastreados para validar o rastreamento em todas as técnicas implementadas nesta dissertação.

Foram duas as sequências geradas para os testes de rastreamento: uma sequência sintética de um cubo texturizado composta de 250 quadros, conforme a Figura 5-1; e uma sequência real de uma caixa de café com 400 quadros, exibida na Figura 5-2. A importância da sequência sintética para os testes é a existência de um movimento precisamente conhecido que facilita a comparação de resultados, enquanto que na sequência real são observados os problemas provenientes do movimento e das imagens ruidosas capturadas pela câmera utilizada.

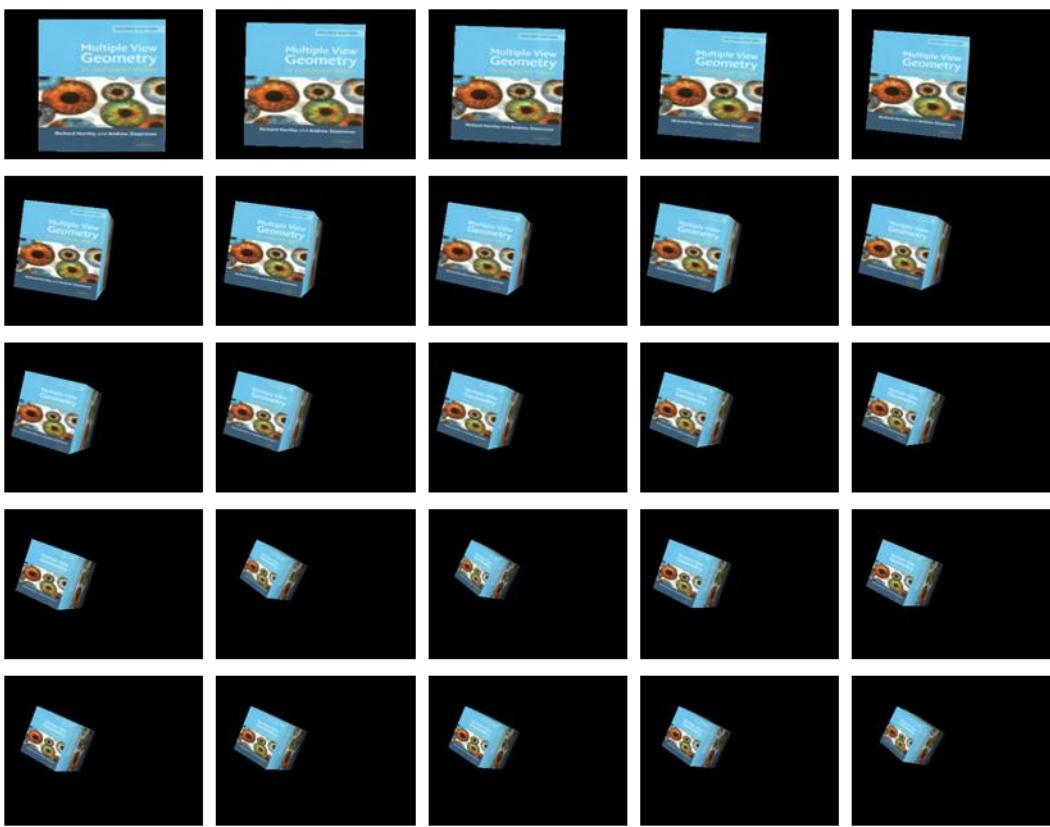


Figura 5-1 Imagens de 25 quadros amostrados da sequência de 250 quadros sintéticos utilizados para o rastreamento (resolução 320x240 pixels).



Figura 5-2 Imagens de 25 quadros amostrados da sequência de 400 quadros reais utilizados para o rastreamento (resolução 320x240 pixels).

5.2 Rastreamento Utilizando *Spatial IPB*

A técnica Spatial IPB foi avaliada usando dados sintéticos e reais. A Figura 5-3 mostra alguns resultados de estimativa de pose para a sequência sintética “cubo” e a sequência real “caixa de café”. Na sequência do cubo, 11 quadros-chave foram usados, enquanto que na sequência da caixa de café foram utilizados 8. Em ambos os casos, os objetos foram aumentados com seu modelo aramado, de forma a mostrar se os resultados do rastreamento são visualmente aceitáveis.

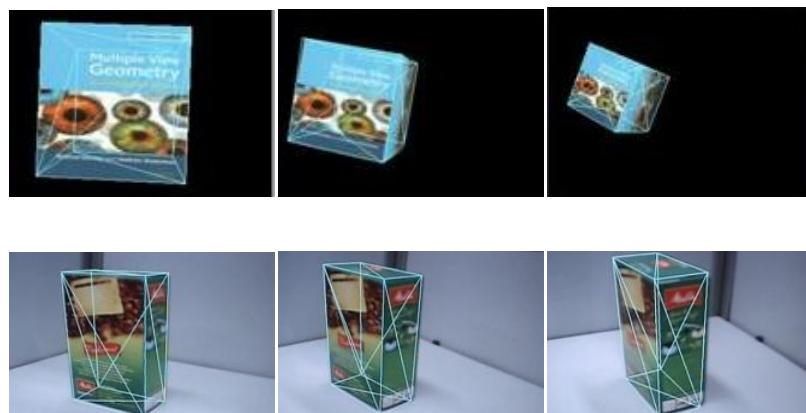


Figura 5-3 Resultados de rastreamento do Spatial IPB para uma sequência sintética (linha superior) e uma sequência real (linha inferior).

A Tabela 5-1 apresenta o tempo médio requerido em cada etapa do algoritmo de rastreamento usando ambas as sequências mencionadas acima como entrada. O método de escolha de quadro-chave Mahalanobis foi mais rápido que o de histograma. Considerando o pior caso para a escolha do quadro-chave (histograma), os tempos médios gastos para rastrear um quadro foram 50 ms para a sequência do cubo (resultando em uma taxa de 20 fps) e 90 ms para a sequência da caixa de café (resultando em uma taxa de 11 fps). A Figura 5-4 mostra os tempos totais gastos para rastrear cada um dos primeiros 250 quadros de ambas as sequências. A taxa de quadros obtida é adequada para aplicações de RA. Entretanto, algumas otimizações podem ainda ser feitas, especialmente relativas à extração e ao casamento de características, que é a etapa que se mostrou como o gargalo da técnica.

Tabela 5-1 Comparação dos tempos e números de casamentos relativos ao algoritmo de rastreamento Spatial IPB.

		Cubo	Caixa de café
Tempo (ms)	Escolha do keyframe	0.46	0.46
	Histograma	2.11	1.66
	Geração do keyframe intermediário	2.60	4.46
	Extração e casamento de características	43.18	79.54
	Cálculo de pose	2.20	3.45
	Número de casamentos	52	96

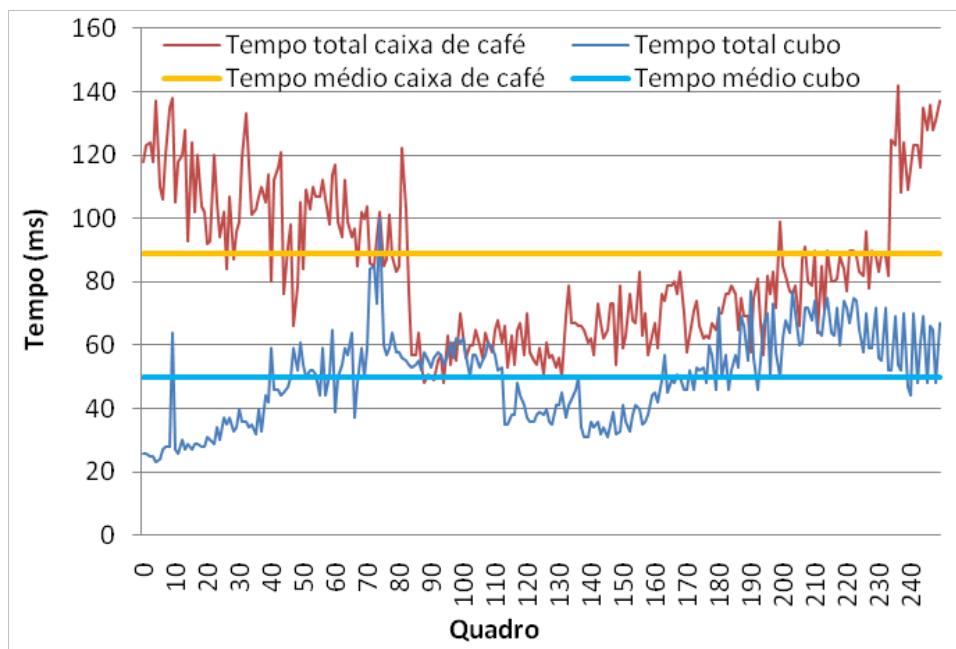


Figura 5-4 Tempo total de computação do Spatial IPB para cada um dos primeiros 250 quadros das sequências sintética e real.

O erro de rastreamento relativo à sequência sintética do cubo é apresentado na Figura 5-5. A distância entre o objeto rastreado e a câmera variou entre 200 e 600 mm. O lado do cubo mede 100 mm. Diferentes métodos de escolha do quadro-chave foram usados. Os erros médios foram 2.80 mm ao usar Mahalanobis e 3.44 mm ao usar histograma. O erro de rastreamento do histograma apresentou alguns picos que influenciaram o erro médio total, mas se mostrou mais estável que o Mahalanobis ao longo da sequência.

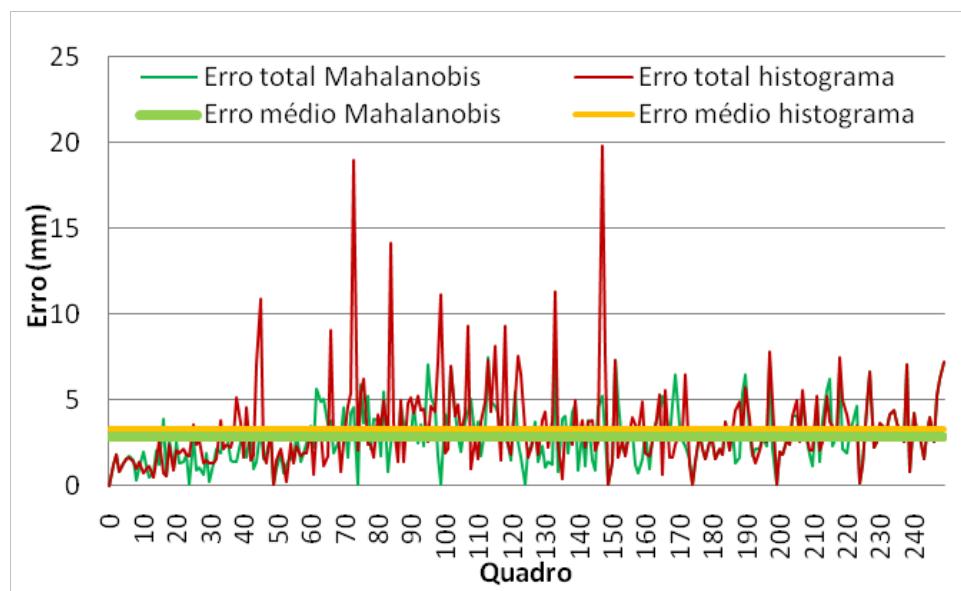


Figura 5-5 Precisão da estimativa do Spatial IPB relativa à sequência do cubo sintético.

A Figura 5-6 mostra resultados da precisão de rastreamento relativa à sequência real da caixa de café considerando ambos os métodos de escolha do quadro-chave. As posições da câmera calculadas pelo algoritmo de rastreamento no eixo x são comparadas com a verdade absoluta provida pelos quadros-chave. Ao usar o método de histograma, as poses calculadas pelo rastreador seguiram os quadros-chave ao longo da sequência, enquanto que usar o método de Mahalanobis resultou em uma falha de rastreamento perto do quadro 70 devido a instabilidade das escolhas.

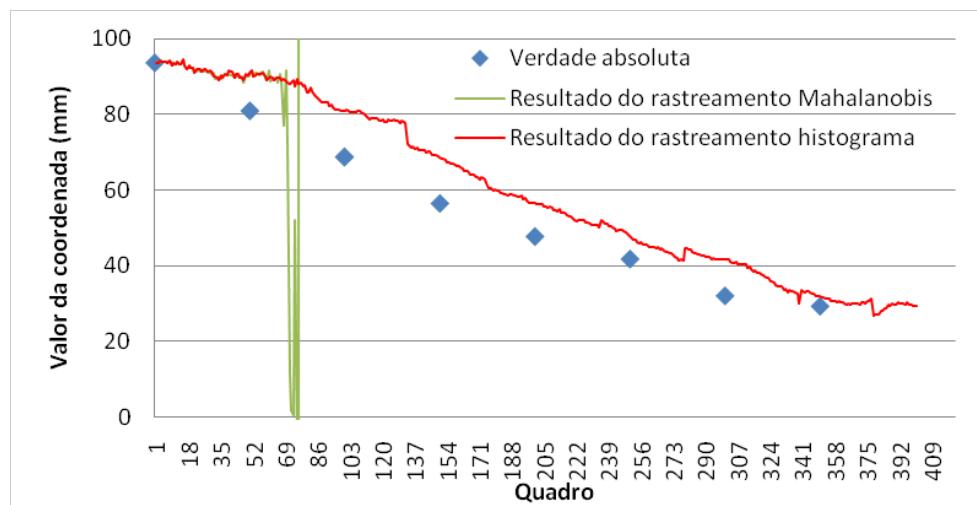


Figura 5-6 Resultados do rastreamento do eixo x da câmera pelo Spatial IPB relativo à sequência real da caixa de café.

5.3 Rastreamento Utilizando *Temporal IPB*

A técnica Spatial IPB também foi avaliada usando dados sintéticos e reais. A Figura 5-7 mostra alguns resultados de estimativa de pose para a sequência sintética “cubo” e a sequência real “caixa de café”. Em ambos os casos, os objetos foram aumentados com seu modelo aramado, de forma a mostrar se os resultados do rastreamento são visualmente aceitáveis e, como pode ser observado, o acúmulo de erros inerente às técnicas que utilizam informação temporal ocasionou falhas de rastreamento.

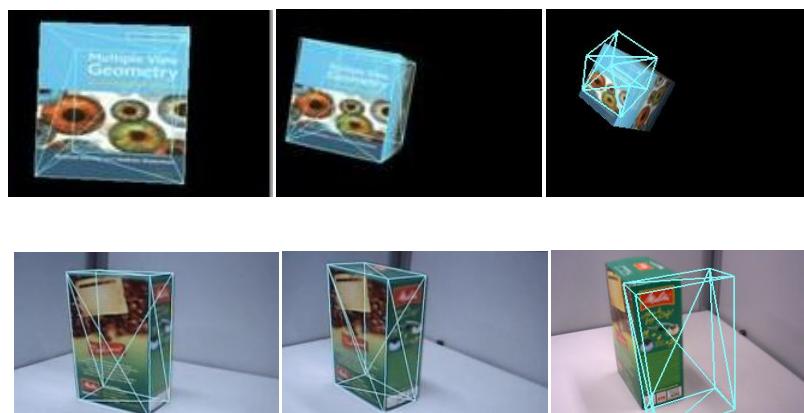


Figura 5-7 Resultados de rastreamento do Temporal IPB para uma sequência sintética (linha superior) e uma sequência real (linha inferior). Ambas as sequências não conseguiram ser rastreadas até o fim devido ao acúmulo de erros da técnica.

A Tabela 5-2 apresenta o tempo médio requerido em cada etapa do algoritmo de rastreamento usando ambas as sequências mencionadas acima como entrada. Os tempos médios gastos para rastrear um quadro foram 21 ms para a sequência do cubo (resultando em uma taxa de 47 fps) e 37 ms para a sequência da caixa de café (resultando em uma taxa de 30 fps). A Figura 5-8 mostra os tempos totais gastos para rastrear cada um dos primeiros 250 quadros de ambas as sequências. A taxa de quadros obtida é adequada para aplicações de RA.

Tabela 5-2 Comparação dos tempos e números de casamentos relativos ao algoritmo de rastreamento Temporal IPB.

		Cubo	Caixa de café
Tempo (ms)	Extração e casamento de características	18.2	27.276
	Cálculo de pose	2.20	5.45
Número de casamentos		60	110

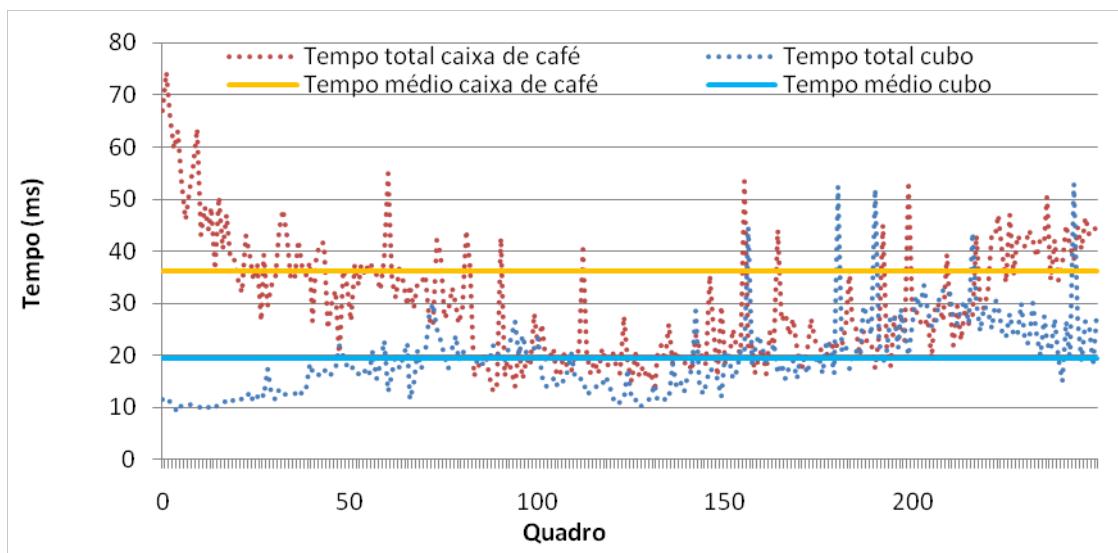


Figura 5-8 Tempo total de computação do Temporal IPB para cada um dos primeiros 250 quadros das sequências sintética e real.

O erro de rastreamento relativo à sequência sintética do cubo é apresentado na Figura 5-9. A distância entre o objeto rastreado e a câmera variou entre 200 e 600 mm. O lado do cubo mede 100 mm. O erro médio mostrou um crescimento com relação ao rastreamento utilizando informação espacial devido ao *jitter*, com falhas graves ocorrendo a partir do quadro 90.

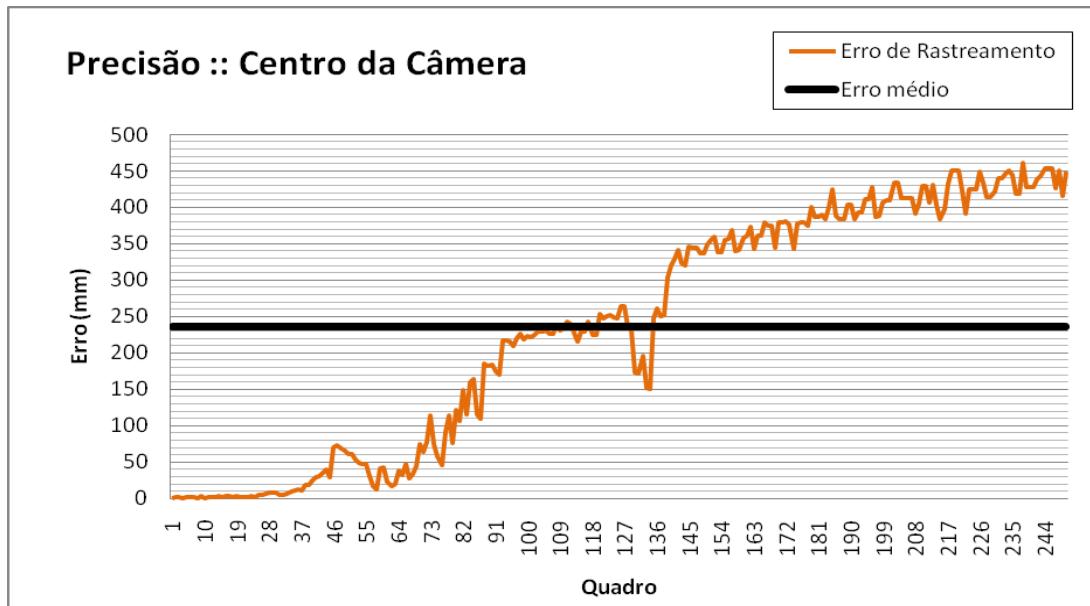


Figura 5-9 Precisão da estimativa do Temporal IPB relativa à sequência do cubo sintético.

A Figura 5-10 mostra os resultados da precisão de rastreamento relativa à sequência real da caixa de café. As posições da câmera calculadas pelo algoritmo de rastreamento no eixo x são comparadas com a verdade absoluta provida pelos quadros-chave (11 quadros calculados previamente para fornecer uma verdade absoluta). O método resultou em uma falha de rastreamento perto do quadro 335.

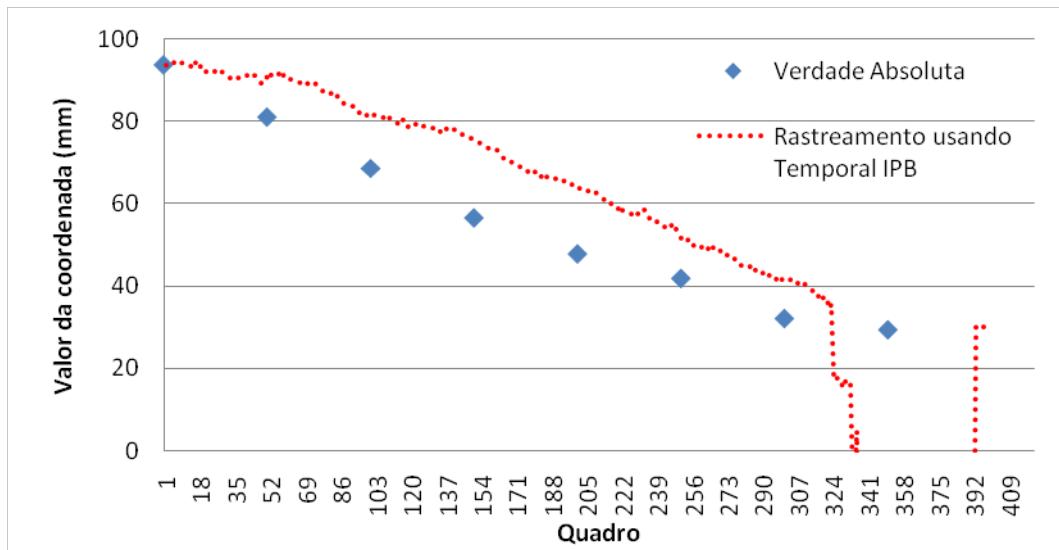


Figura 5-10 Resultados do rastreamento do eixo x da câmera pelo Temporal IPB relativo à sequência real da caixa de café.

5.4 Rastreamento Utilizando *Spatial Temporal IPB*

A técnica Spatial Temporal IPB demonstrou bom desempenho apesar do *overhead* imposto pelo uso das informações temporal e espacial de forma complementar. A Figura 5-11 mostra alguns resultados de estimativa de pose para a sequência sintética “cubo” e a sequência real “caixa de café”. Assim como as demais variações do IPB, os objetos foram aumentados com seu modelo aramado, de forma a mostrar se os resultados do rastreamento são visualmente aceitáveis.

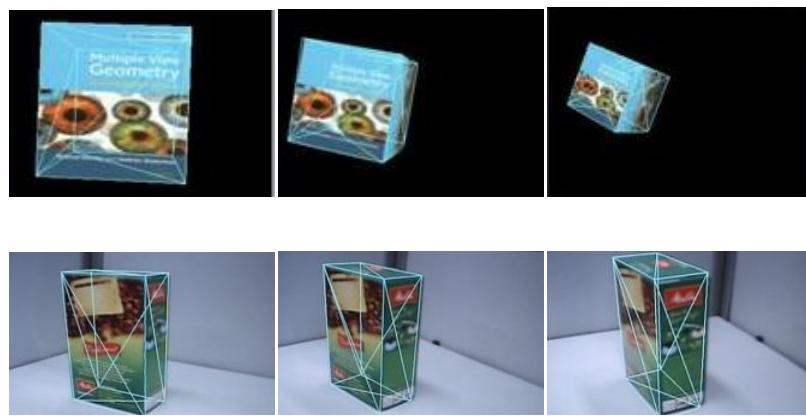


Figura 5-11 Resultados de rastreamento do Spatial Temporal IPB para uma sequência sintética (linha superior) e uma sequência real (linha inferior).

A Tabela 5-3 apresenta o tempo médio requerido por cada etapa do algoritmo de rastreamento usando ambas as sequências mencionadas acima como entrada. Os tempos médios gastos para rastrear um quadro (utilizando como seletor de *keyframes* o histograma) foram 62 ms para a sequência do cubo (resultando em uma taxa de 16 fps) e 110 ms para a sequência da caixa de café (resultando em uma taxa de 10 fps). A Figura 5-12 mostra os tempos totais gastos para rastrear cada um dos primeiros 250 quadros de ambas as sequências. A taxa de quadros obtida se mostra um pouco aquém do adequado para aplicações de RA.

Tabela 5-3 Comparação dos tempos e números de casamentos relativos ao algoritmo de rastreamento Spatial Temporal IPB.

		Cubo	Caixa de café	
Tempo (ms)	Escolha do <i>keyframe</i>	Mahalanobis	0.46	0.50
	Histograma	2.13	1.392	
	Geração do <i>keyframe</i> intermediário	4.60	8.39	
	Extração e casamento de características	50.13	95.35	
	Cálculo de pose	4.144	3.45	
	Número de casamentos (<i>keyframe</i> + anterior)	100	150	

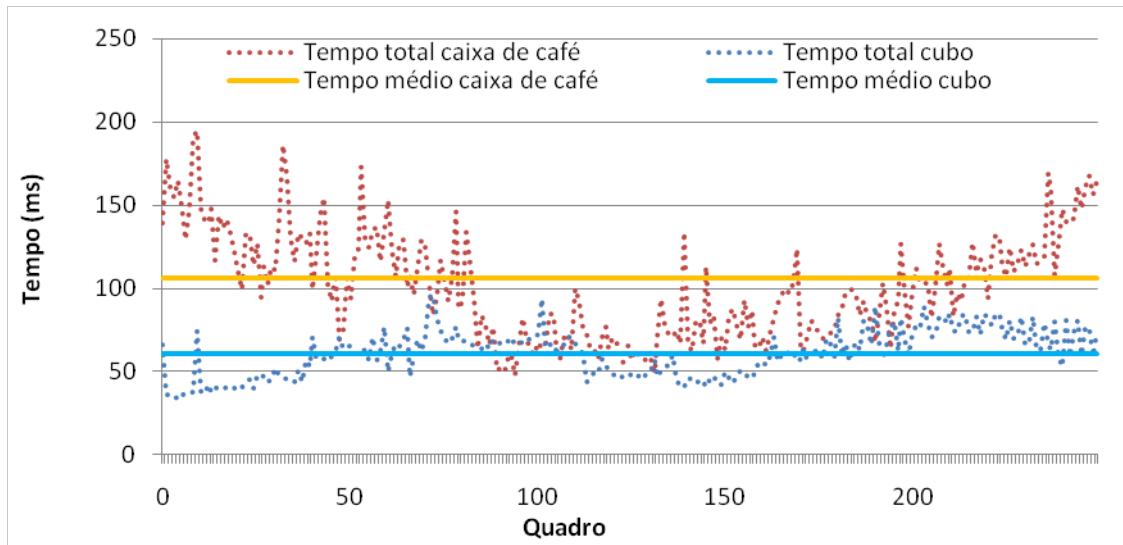


Figura 5-12 Tempo total de computação do Spatial Temporal IPB para cada um dos primeiros 250 quadros das sequências sintética e real.

O erro de rastreamento relativo à sequência sintética do cubo utilizando o histograma para escolher o melhor *keyframe* é apresentado na Figura 5-13. A distância entre o objeto rastreado e a câmera variou entre 200 e 600 mm. O lado do cubo mede 100 mm. O erro médio mostrou valores baixos, em torno de 4 mm, sendo suficientes para uma aplicação de RA.

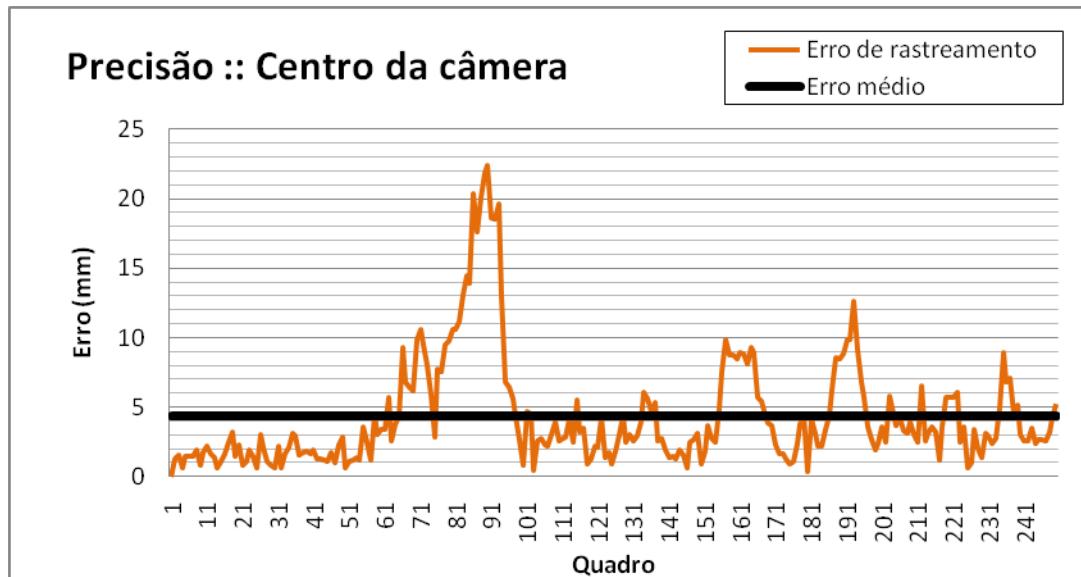


Figura 5-13 Precisão da estimativa do Spatial Temporal IPB relativa à sequência do cubo sintético.

A Figura 5-14 mostra resultados da precisão de rastreamento relativa à sequência real

da caixa de café. As posições da câmera calculadas pelo algoritmo de rastreamento no eixo x são comparadas com a verdade absoluta provida pelos quadros-chave (11 quadros calculados previamente para proporcionar uma verdade absoluta). O método resultou em uma falha de rastreamento perto do quadro 335.

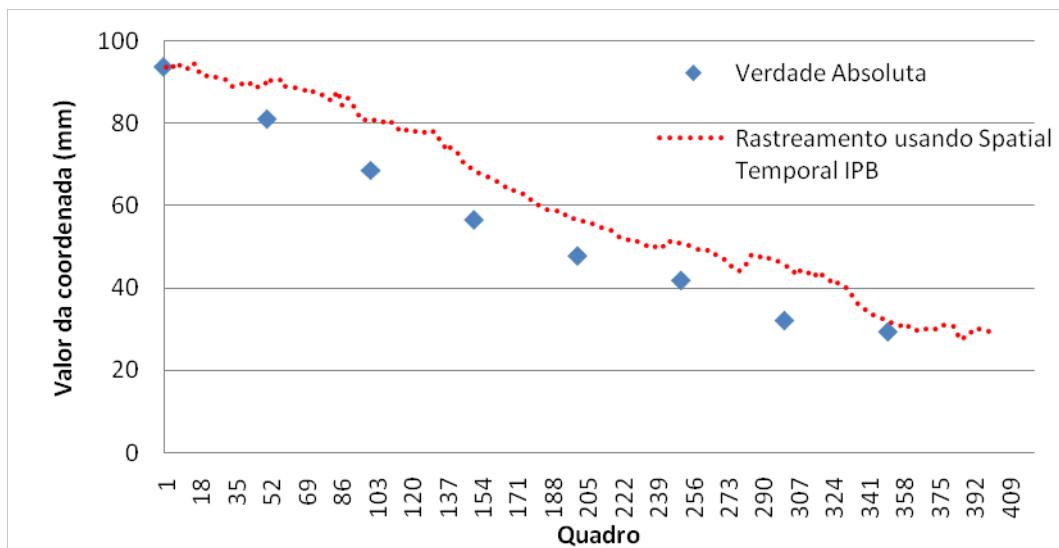


Figura 5-14 Resultados do rastreamento do eixo x da câmera pelo Spatial Temporal IPB relativo à sequência real da caixa de café.

5.4.1 Rastreamento Utilizando Spatial Temporal IPB + RanSaC

Para realizar uma análise de robustez acerca da técnica RanSaC, utilizada para minimizar a influência de *outliers* no processo de rastreamento, foram realizados testes com a aplicação do RanSaC em conjunto com a técnica Spatial Temporal IPB. Como discutido na seção 2.2.3.1, a utilização do RanSaC se dá através da seleção de subconjuntos de amostras a serem testadas e, o subconjunto que apresenta o melhor resultado é escolhido como solução do problema.

Nas Figuras Figura 5-15, Figura 5-16, Figura 5-17 e Figura 5-18 podem ser observados os resultados de precisão da aplicação do RanSaC com subconjuntos de tamanho 20 (cada minimização leva em conta 20 casamentos) e utilização de 30, 20, 10 e 1 subconjuntos gerados, respectivamente. A sequência rastreada foi a do “cubo” sintético. É possível notar o aumento do erro médio do rastreamento durante o decaimento da quantidade de subconjuntos utilizados, fator que indica um ganho real quando do aumento da quantidade de subconjuntos.

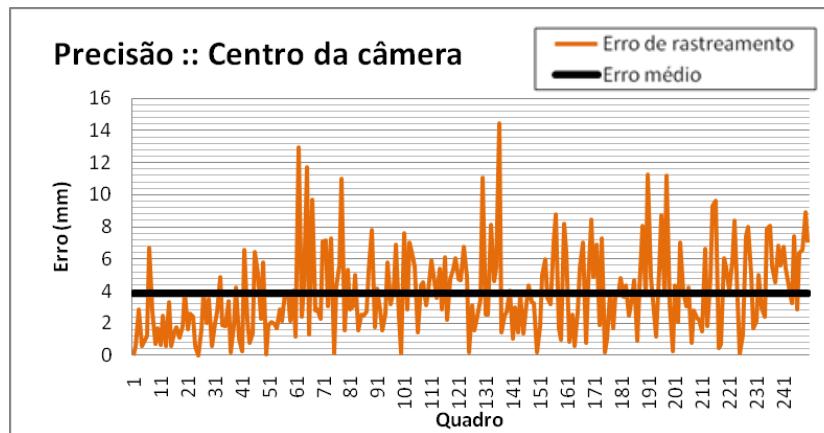


Figura 5-15 Rastreamento utilizando 30 subconjuntos com 20 amostras cada. Erro médio de 3,86 mm.

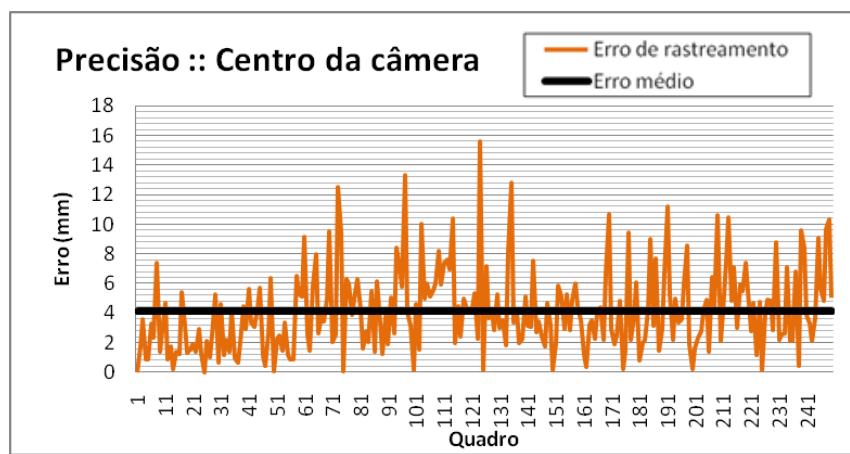


Figura 5-16 Rastreamento utilizando 20 subconjuntos com 20 amostras cada. Erro médio de 4,11mm.

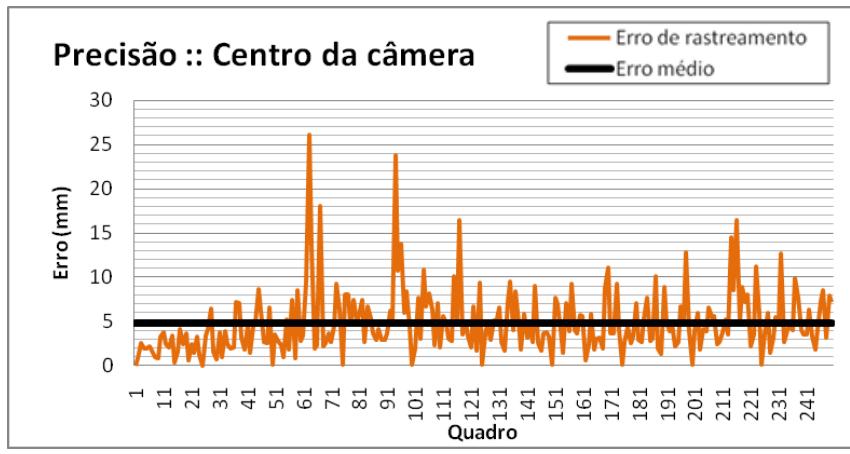


Figura 5-17 Rastreamento utilizando 10 subconjuntos com 20 amostras cada. Erro médio de 4,78mm.

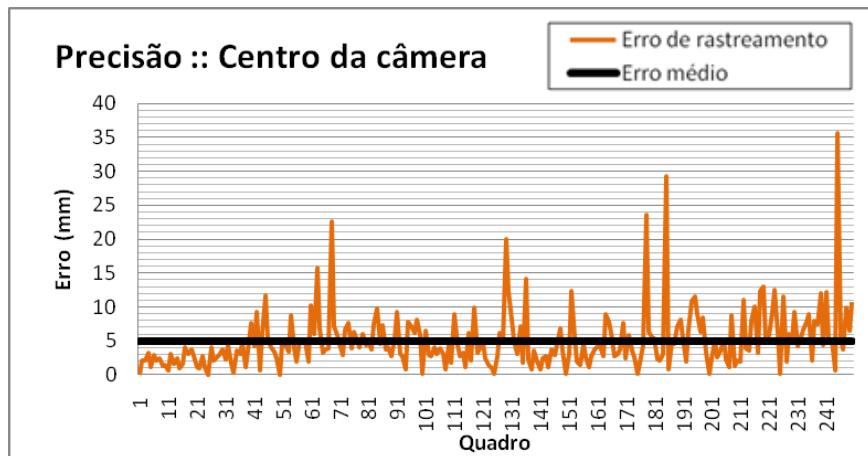


Figura 5-18 Rastreamento utilizando 1 subconjunto com 20 amostras. Erro médio de 5,0 mm.

5.5 Rastreamento utilizando IPB + Filtro de Partículas

A técnica proposta nesta dissertação para utilização de informação de textura juntamente com um FP se mostrou promissora. Apesar de ainda não conseguir uma taxa adequada a aplicações de RA devido à necessidade de otimizações de desempenho, a técnica validou a função de verossimilhança de forma a permitir seu uso em conjunto com outras abordagens de rastreamento (baseado em arestas, por exemplo).

Devido à necessidade de desempenho para testar parâmetros do FP, como quantidade de partículas ou variância do ruído gaussiano utilizado na propagação, utilizou-se na fase de otimização o Temporal IPB. A Figura 5-19 mostra alguns resultados de estimativa de pose para a sequência sintética “cubo”. A sequência real da “caixa de café” não pode ser rastreada utilizando o FP. Existem indícios de que o problema tenha sido a definição de parâmetros do filtro e estudos futuros devem ser realizados de forma a configurar os parâmetros de forma adequada.

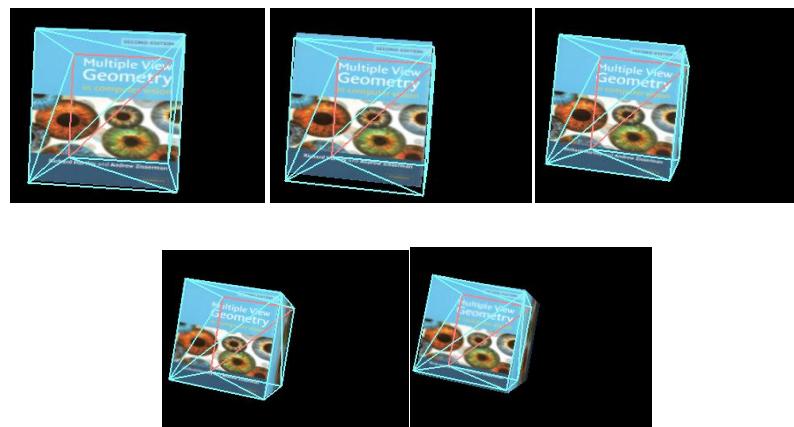


Figura 5-19 Resultados de rastreamento utilizando o FP com 300 partículas para uma sequência sintética.

A Tabela 5-4 apresenta o tempo médio requerido por cada etapa do algoritmo de rastreamento. O tempo médio gasto para rastrear um quadro foi de 1475 ms para a sequência do cubo (resultando em uma taxa de 0,67 fps). A Figura 5-20 mostra os tempos totais gastos para rastrear cada um dos 80 quadros da sequência antes do erro de rastreamento. A taxa de quadros obtida se mostra um aquém do adequado para aplicações de RA devido à ausência de algumas otimizações na implementação (implementação do FP em GPGPU por exemplo).

Tabela 5-4 Comparaçao dos tempos e números de partículas relativos ao algoritmo de rastreamento utilizando IPB + FP.

	Cubo
Tempo (ms)	
	Extração de características e geração do mapa de distâncias
	Propagação
	Análise das melhores partículas
	Otimização (Temporal IPB)
	Análise das partículas otimizadas
	Cálculo de pesos
	Cálculo de pose
	Reamostragem
	Quantidade de Partículas na Propagação

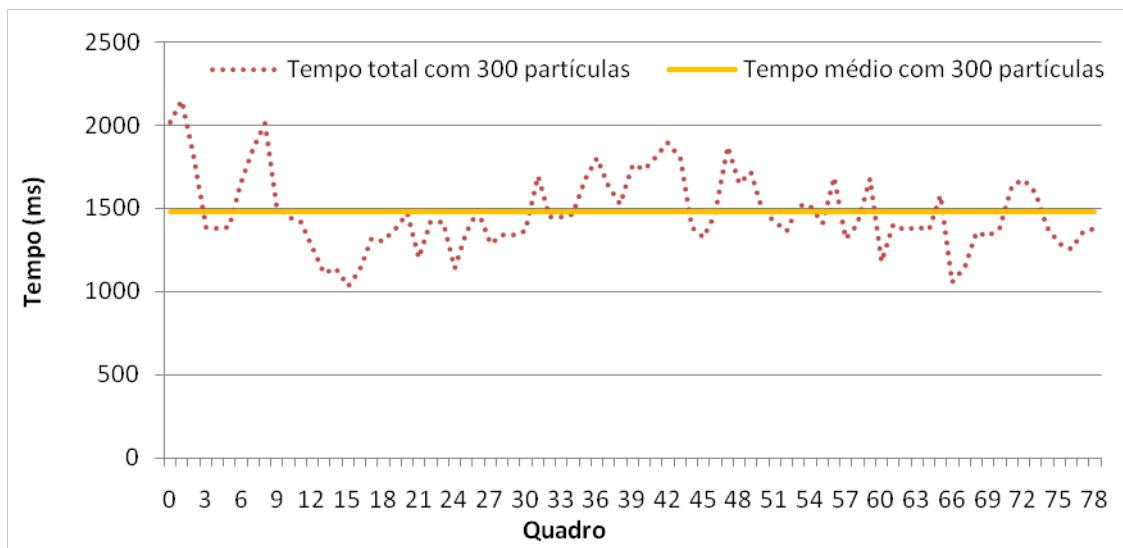


Figura 5-20 Tempo total de computação do FP para cada um dos 80 quadros rastreados da sequência do cubo com 300 partículas.

O erro de rastreamento relativo à sequência sintética do cubo utilizando 100 partículas é apresentado na Figura 5-21. É válido destacar a falha de rastreamento ocorrida no quadro 45 enquanto a sequência a qual erro é apresentado na Figura 5-22, utilizando 300 partículas só ocorre no quadro 80. Ambos os testes foram realizados tomando como parâmetros para propagação uma distribuição normal centrada em 0.0 e com variância de 0.001. Também foram criadas no início do rastreamento 100 e 300 partículas respectivamente. Os resultados apresentados demonstram uma técnica promissora apesar da necessidade de estudos mais elaborados acerca dos parâmetros utilizados em cada caso, prováveis causadores das falhas.

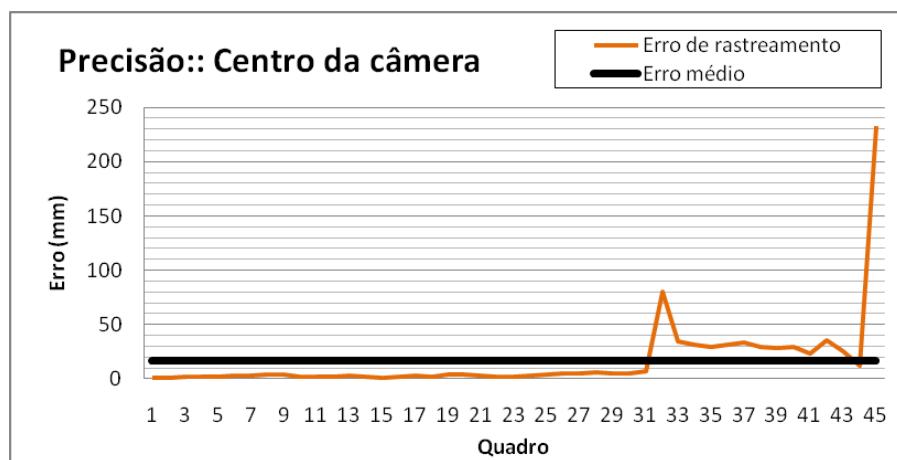


Figura 5-21 Precisão da estimativa do FP relativa à sequência do cubo sintético utilizando 100 partículas. É válido observar a falha de rastreamento no quadro 45.

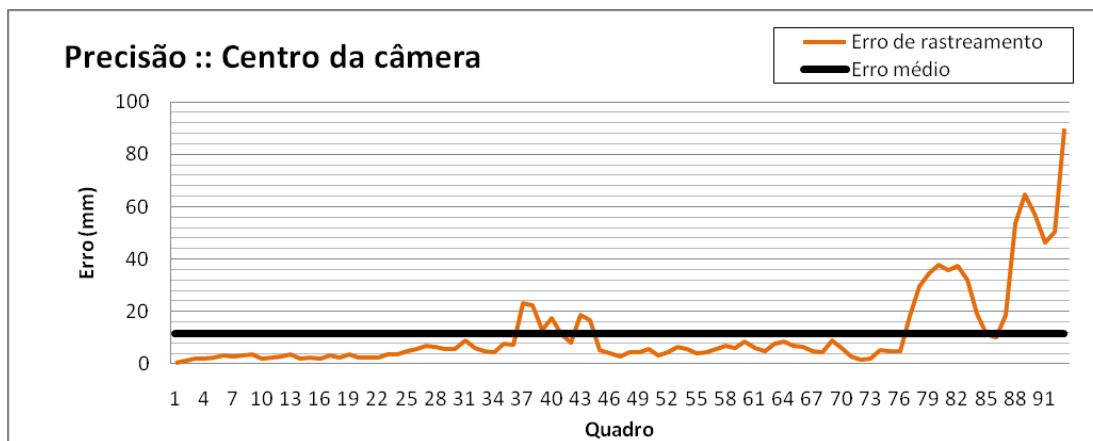


Figura 5-22 Precisão da estimação do FP relativa à sequência do cubo sintético utilizando 300 partículas. É válido observar a falha de rastreamento no quadro 91.

5.5.1 Ferramenta para Análise de Partículas

A ferramenta para análise visual e qualitativa de partículas foi desenvolvida utilizando o Matlab. Tal ferramenta se mostrou importante para auxiliar a validação visual do cálculo de verossimilhança, proposto nesta dissertação, através da observação das partículas em conjunto com sua verossimilhança sobre a pose correta do quadro em questão, como pode ser observado na Figura 5-23.

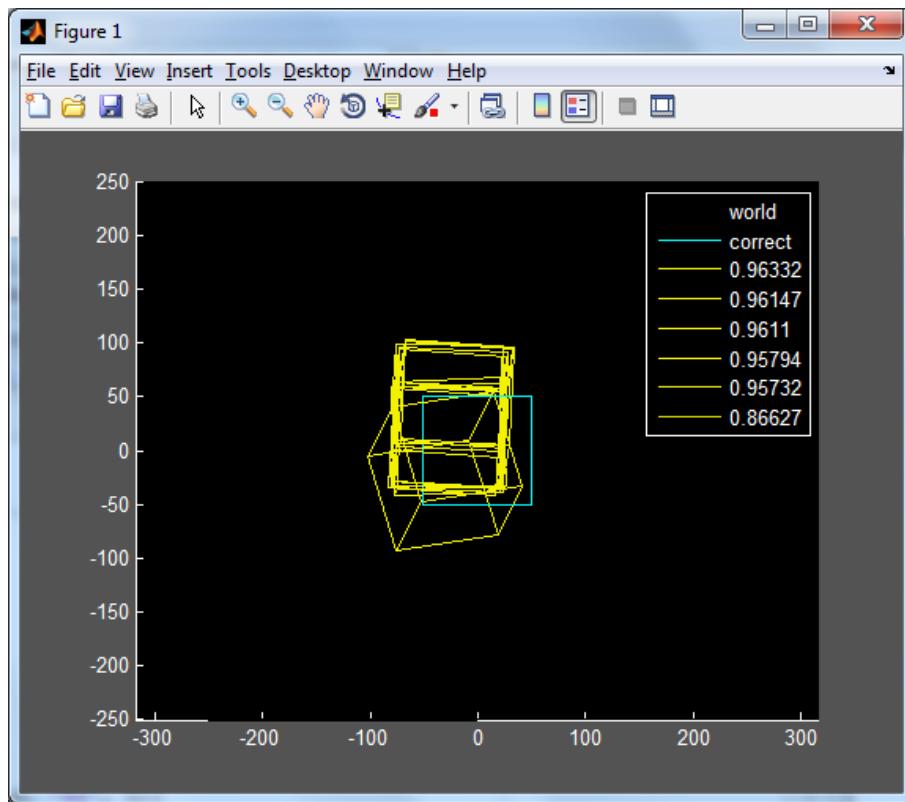


Figura 5-23 Ferramenta para análise de partículas. A legenda contempla a visualização da verossimilhança das partículas ordenadas (decrescente) com coloração proporcional à verossimilhança (tom ciano indica pose correta e tons de amarelo para partículas). Na região central observam-se as partículas sobrepostas à pose correta.

Para trabalhar com tal ferramenta, se faz necessário o arquivamento das informações das partículas na técnica implementada para que os dados sejam carregados no Matlab. Foram utilizados dois arquivos de texto, um contendo as poses das partículas e outro contendo as verossimilhanças calculadas no passo de cálculo da verossimilhança. A ferramenta criada carrega então estes arquivos e realiza a visualização aplicando as transformações definidas pelas matrizes de pose a um cubo, de forma a retratar as diferentes partículas do sistema. Para o correto funcionamento da composição a matriz deve ser aplicada às coordenadas homogêneas do cubo como uma matriz 3x4.

Através de diversas visualizações foi possível observar a coerência do cálculo de verossimilhança, pois como pode ser observado na Figura 5-23, as partículas de menor verossimilhança foram realmente aquelas mais afastadas da posição correta do cubo (observar partícula mais escura destoando do grupo de partículas principais).

Outro fator que valida a função de verossimilhança definida é o gráfico de variação da

verossimilhança ao longo dos eixos X e Y em torno da posição correta, ou seja, são geradas diversas poses em torno da pose correta aplicando variações nas coordenadas X e Y para que todas as poses geradas em torno da pose correta sejam avaliadas.

Para determinar uma correta verossimilhança, a função descrita precisa apontar a direção correta na análise da mesma. Outro ponto que caracteriza a função como boa para análise é sua convexidade, sem uma inclinação muito forte, denotando uma suavidade na mudança da verossimilhança, como observado na Figura 5-24.

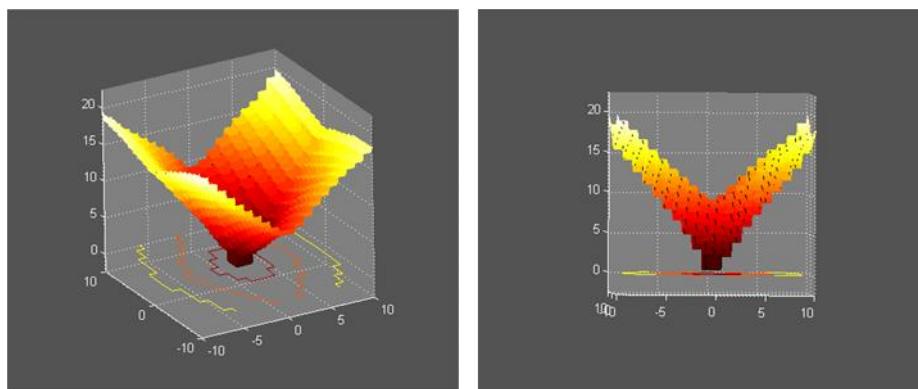


Figura 5-24 Análise da função de verossimilhança. Variações de translação em x e y em torno da pose correta denotando correta função de verossimilhança devido à curvatura apontando para a pose correta, sem profundidade muito acentuada e suavidade na variação da qualidade da verossimilhança.

6 Conclusões

“Sonhar é a nobre arte de acreditar no amanhã.” – Francisco Simões.

O trabalho apresentado nesta dissertação apresentou uma nova técnica de rastreamento baseado em textura e filtro de partículas. Para tanto, foi definida uma nova função de verossimilhança para tratar pontos de interesse de forma otimizada e realizadas implementações de variações da técnica básica.

As variações da técnica IPB tiveram sua robustez analisada através da utilização de um caso sintético e um caso real. Como previsto na literatura a variação Spatial IPB sofreu com o problema de *drift*, enquanto a variação *Temporal* IPB teve seu rastreamento muito prejudicado por *jitter*. A técnica Spatial Temporal IPB conseguiu reduzir a ocorrência de *drift* e *jitter* visualmente, mas em termos de precisão do rastreamento não conseguiu superar a técnica baseada em Spatial IPB. Isto ocorreu devido ao caso de teste escolhido que utiliza como *keyframes* quadros originalmente presentes na sequência. Em aplicações reais o *drift* torna a aplicação do Spatial IPB suscetível a falhas, da mesma forma que a técnica Temporal IPB é suscetível ao *jitter*.

Todas as variações do IPB implementadas podem sofrer otimizações de desempenho, principalmente na etapa de extração e casamento de características, através da utilização de GPGPU na computação das mesmas. Uma versão preliminar em CUDA já foi desenvolvida denotando ganhos de 30 vezes no tempo de computação.

A abordagem proposta através do uso de Filtro de Partículas em conjunto com o IPB se mostra bastante promissora, tendo sido validada a função de verossimilhança que utiliza informações de textura tanto pela análise do rastreamento obtido quanto pela ferramenta de análise de partículas. Através do uso de otimizações de desempenho como no casamento de características ou a paralelização do processamento das partículas (ambas podendo ser feitas em GPGPU), podem ser atingidos resultados de tempo real. Já a robustez do método pode ser incrementada através da utilização de mais de uma fonte de informação, como informações de

aresta (Simões, Lima, Teichrieb, Kelner, & Santos, 2008).

6.1 Contribuições Gerais

As principais contribuições deste trabalho foram:

- Pesquisa e experimentação na área de Realidade Aumentada sem Marcadores, que é uma área de vanguarda da computação;
- Estudo e análise de diversas técnicas de rastreamento para Realidade Aumentada sem Marcadores, com elaboração de uma taxonomia e uma análise comparativa;
- Análise de desempenho e robustez de técnicas de rastreamento baseadas em pontos de interesse;
- Desenvolvimento de um algoritmo para cálculo de verossimilhança a partir de informação de textura;
- Desenvolvimento e implementação de uma nova técnica de rastreamento a partir de informação de textura e filtro de partículas;
- Publicações em periódicos nacionais, conferências nacionais e livros.

6.2 Contribuições para a Comunidade Científica (Publicações)

As principais contribuições deste trabalho para a comunidade científica, sob a forma de publicações, foram:

6.2.1 Capítulos de Livros

Lima, João, Francisco Simões, Lucas Figueiredo, Veronica Teichrieb, and Judith Kelner. 2009. Online Monocular Markerless 3D Tracking for Augmented Reality. In *Abordagens Práticas de Realidade Virtual e Aumentada*, 81d. Fátima Nunes, Liliane Machado, Márcio Pinho, and Claudio Kirner, 1-30. 1st 81d. Porto Alegre: Pontifícia Universidade Católica – PUCRS.

Teixeira, João, Francisco Simões, Rafael Roberto, Veronica Teichrieb, and Judith Kelner. 2010. Reconstrução 3D usando luzes estruturadas. In *Tendências e Técnicas em Realidade Virtual e Aumentada*, SBC.

6.2.2 Artigo Completo (Conferência)

Simões, Francisco, João Lima, Veronica Teichrieb, and Judith Kelner. 2008. Realidade Aumentada sem Marcadores Baseada na Amostragem de pontos em Areias. In: *5º Workshop de Realidade Virtual e Aumentada (WRVA2008)*, Bauru/SP.

Lima, João, Francisco Simões, Lucas Figueiredo, Veronica Teichrieb, Judith Kelner, and Ismael Santos. 2009. Model Based 3D Tracking Techniques for Markerless Augmented Reality. In *XI Symposium on Virtual and Augmented Reality*. Porto Alegre/RS.

6.2.3 Artigo Completo (Periódico)

Lima, João, Francisco Simões, Lucas Figueiredo, Veronica Teichrieb and Judith Kelner. 2010. Model Based Markerless 3D Tracking Applied to Augmented Reality. *SBC Journal on 3D Interactive Systems*, no. 1.

6.2.4 Mini-cursos

Lima, João, Francisco Simões, Daliton Silva, Veronica Teichrieb, and Judith Kelner. 2008. Reconhecimento de padrões em tempo real utilizando a biblioteca OpenCV. In: *5º Workshop de Realidade Virtual e Aumentada (WRVA2008)*, Bauru/SP.

Lima, João, Francisco Simões, Lucas Figueiredo, Veronica Teichrieb, and Judith Kelner. 2009. Online Monocular Markerless 3D Tracking for Augmented Reality. In *XI Symposium on Virtual and Augmented Reality*, Porto Alegre: Pontifícia Universidade Católica – PUCRS.

Teixeira, João, Francisco Simões, Rafael Roberto, Veronica Teichrieb, and Judith Kelner. 2010. Reconstrução 3D usando luzes estruturadas. In *XII Symposium on Virtual and Augmented Reality*, Natal/RN.

6.3 Trabalhos Futuros

Algumas direções de melhoramentos e sugestões de possíveis investigações são:

- Implementar em GPGPU o filtro de partículas para otimizar o desempenho (atingir tempo real) e facilitar a realização de testes e a configuração de parâmetros;
- Escolher parâmetros de forma automática;

- Realizar uma análise mais detalhada da função de verossimilhança proposta e dos parâmetros do FP;
- Implementar uma aplicação de Realidade Aumentada sem Marcadores para validação da técnica proposta;
- Associar informação de textura e de aresta para melhorar a robustez do rastreamento;
- Analisar a utilização de RanSaC no passo de otimização das melhores partículas;
- Realizar a inicialização automática do rastreador;
- Incrementar a recuperação de falhas de rastreamento através de variações no filtro de partículas.

Outros pontos como rastreamento de objetos deformáveis (através de adaptações na técnica proposta), rastreamento sem modelos e rastreamento por detecção (técnicas promissoras) merecem um estudo mais aprofundado.

Referências

- Ababsa, F., & Mallem, M. (2008). Robust camera pose estimation combining 2D/3D points and lines tracking. *2008 IEEE International Symposium on Industrial Electronics*, 774-779. Ieee. doi: 10.1109/ISIE.2008.4676964.
- Arulampalam, M. S., Maskell, S., Gordon, N., & Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2), 174-188. doi: 10.1109/78.978374.
- Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., & MacIntyre, B. (2001). Recent Advances in Augmented Reality. *IEEE Computer Graphics and Applications*, 45(8), 506-508.
- Brockett, R. W. (1984). Robotic Manipulators and the Product of Exponentials Formula. *Mathematical Theory of Networks and Systems* (pp. 120-129).
- Davison, a J., & Murray, D. W. (2002). Simultaneous localization and map-building using active vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7), 865-880. doi: 10.1109/TPAMI.2002.1017615.
- Davison, A. J., Reid, I. D., Molton, N. D., & Stasse, O. (2007). MonoSLAM: real-time single camera SLAM. *IEEE transactions on pattern analysis and machine intelligence*, 29(6), 1052-67. doi: 10.1109/TPAMI.2007.1049.
- Drummond, T., & Cipolla, R. (2002). Real-time visual tracking of complex structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7), 932-946. doi: 10.1109/TPAMI.2002.1017620.
- Duh, H. B.-L., & Billinghurst, M. (2008). Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR. *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, 193-202. Ieee. doi: 10.1109/ISMAR.2008.4637362.
- Faugeras, O. (1993). *Three-Dimensional Computer Vision*. MIT Press. Retrieved from http://www.it.lut.fi/international_studies/study/Tuomo_Kauranne_files/3d_computer_visio_n.pdf.
- Forsyth, D. A., & Ponce, J. (2002). *Computer Vision: A Modern Approach. Notes* (Vol. 54, p. 693). Prentice Hall. doi: 10.1016/j.cbi.2010.05.017.
- Harris, C., & Stephens, M. (1988). A combined corner and edge detector. In M. M. Mathews (Ed.), *Alvey vision conference* (Vol. 15, p. 50). Manchester, UK. Retrieved from <http://www.cis.rit.edu/~cns pci/references/dip/harris1988.pdf>.
- Hartley, R., & Zisserman, A. (2004). *Multiple View Geometry in Computer Vision*.

- Hinterstoisser, S., Kutter, O., Navab, N., Fua, P., & Lepetit, V. (2009). Real-time learning of accurate patch rectification. *IEEE Conference on Computer Vision and Pattern Recognition* (2009), 2945-2952. Ieee. doi: 10.1109/CVPR.2009.5206794.
- Isard, M., & Blake, A. (1998). CONDENSATION - conditional density propagation for visual tracking. *International Journal of Computer Vision*. Retrieved September 26, 2010, from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.36.8357>.
- Jurie, F., & Dhome, M. (2002). Real Time Robust Template Matching. *BMVC* (pp. 123-132).
- Klein, G., & Murray, D. (2006). Full-3d edge tracking with a particle filter. *Proc. of BMVC 2006*, 1-10. Retrieved September 26, 2010, from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.62.6718&rep=rep1&t=ype=pdf>.
- Klein, G., & Murray, D. (2009). Parallel Tracking and Mapping on a camera phone. *2009 8th IEEE International Symposium on Mixed and Augmented Reality* (pp. 83-86). Ieee. doi: 10.1109/ISMAR.2009.5336495.
- Koller, D., Daniilidis, K., & Nagel, H. (1993). Model-Based Object Tracking in Monocular Image Sequences of Road Traffic Scenes. *International Journal of Computer Vision*, 3, 257-281.
- Ladikos, A., Benhimane, S., & Navab, N. (2007). A real-time tracking system combining template-based and feature-based approaches. *International Conference on Computer Vision Theory and Applications*. Citeseer. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.88.4914&rep=rep1&t=ype=pdf>.
- Lepetit, V., & Fua, P. (2005). Monocular Model-Based 3D Tracking of Rigid Objects: A Survey. *Foundations and Trends in Computer Graphics and Vision*. Retrieved September 26, 2010, from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.84.8498>.
- Lima, J., Apolinário, E., Farias, T., Silva, M., Teichrieb, V., & Kelner, J. (2007). A Survey of Online Monocular Markerless Augmented Reality. *International Journal of Modeling and Simulation for the Petroleum Industry*, 1, 1-8.
- Lima, J., Simões, F., Figueiredo, L., Teichrieb, V., & Kelner, J. (2009). Online Monocular Markerless 3D Tracking for Augmented Reality. In F. Nunes, L. Machado, M. Pinho, & C. Kirner (Eds.), *Abordagens Práticas de Realidade Virtual e Aumentada* (1st ed., pp. 1-30). Porto Alegre: Pontifícia Universidade Católica - PUCRS.
- Lima, J., Simões, F., Figueiredo, L., Teichrieb, V., & Kelner, J. (2010). Model Based Markerless 3D Tracking applied to Augmented Reality. *Journal on 3D Interactive Systems*, 1.
- Lima, J., Simões, F., Figueiredo, L., Teichrieb, V., Kelner, J., & Santos, I. (2009). Model Based 3D Tracking Techniques for Markerless Augmented Reality. *XI Symposium on Virtual and Augmented Reality*. Porto Alegre.

- Lima, J., Teichrieb, V., Kelner, J., & Lindeman, R. (2009). Standalone edge-based markerless tracking of fully 3-dimensional objects for handheld augmented reality. *Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology - VRST '09*, 139. New York, New York, USA: ACM Press. doi: 10.1145/1643928.1643960.
- Lu, C. P., Hager, G. D., & Mjolsness, E. (2000). Fast and globally convergent pose estimation from video images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(6), 610-622. IEEE. doi: 10.1109/34.862199.
- Moreno-Noguer, F., Lepetit, Vincent, & Fua, Pascal. (2007). Accurate Non-Iterative O(n) Solution to the PnP Problem. *IEEE 11th International Conference on Computer Vision (2007)*, 1-8. Ieee. doi: 10.1109/ICCV.2007.4409116.
- Newcombe, R. A., & Davison, A. J. (2010). Live Dense Reconstruction with a Single Moving Camera. *Computer Vision and Pattern Recognition*.
- Nistér, David. (2003). Preemptive RANSAC for live structure and motion estimation. *Proceedings Ninth IEEE International Conference on Computer Vision*, (Iccv), 199-206 vol.1. Ieee. doi: 10.1109/ICCV.2003.1238341.
- Pollefeys, M., Nistér, D., Frahm, J. M., Akbarzadeh, A., Mordohai, P., Clipp, B., et al. (2007). Detailed Real-Time Urban 3D Reconstruction from Video. *International Journal of Computer Vision*, 78(2-3), 143-167. Springer. doi: 10.1007/s11263-007-0086-4.
- Pressigout, M., & Marchand, E. (2006). Real-time 3D model-based tracking: combining edge and texture information. *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, (May), 2726-2731. Ieee. doi: 10.1109/ROBOT.2006.1642113.
- Pupilli, M., & Calway, A. (2005). Real-Time Camera Tracking Using a Particle Filter. In Proc. British Machine Vision Conference. Retrieved September 26, 2010, from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.60.1137>.
- Simões, F., Lima, J., Teichrieb, V., Kelner, J., & Santos, I. (2008). Realidade Aumentada sem Marcadores Baseada na Amostragem de Pontos em Arestas. *Workshop em Realidade Virtual e Aumentada*. Bauru.
- Skrypnyk, I., & Lowe, D. G. (2004). Scene Modelling, Recognition and Tracking with Invariant Image Features. *Third IEEE and ACM International Symposium on Mixed and Augmented Reality*, (Ismar), 110-119. Ieee. doi: 10.1109/ISMAR.2004.53.
- Teixeira, J. M., Simões, F., Roberto, R., & Teichrieb, V. (2010). Reconstrução 3D usando luzes estruturadas (pp. 1-33).
- Teulière, C., Marchand, Eric, & Eck, L. (2010). Using multiple hypothesis in model-based tracking. *Robotics and Automation (ICRA), 2010*. Retrieved September 26, 2010, from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.154.267&rep=rep1&t=ype=pdf>.

- Triggs, B., McLauchlan, P., Hartley, R., & Fitzgibbon, A. (2000). Bundle Adjustment — A Modern Synthesis 1 Introduction. *System, 34*099, 1-71.
- Vacchetti, L., Lepetit, V., & Fua, P. (2003). Fusing online and offline information for stable 3D tracking in real-time. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. (pp. II-241-8). IEEE Comput. Soc. doi: 10.1109/CVPR.2003.1211476.
- Vacchetti, L., Lepetit, V., & Fua, P. (2004a). Combining Edge and Texture Information for Real-Time Accurate 3D Camera Tracking. *Third IEEE and ACM International Symposium on Mixed and Augmented Reality*, 48-57. Ieee. doi: 10.1109/ISMAR.2004.24.
- Vacchetti, L., Lepetit, V., & Fua, P. (2004b). Stable real-time 3D tracking using online and offline information. *IEEE transactions on pattern analysis and machine intelligence*, 26(10), 1385-91. doi: 10.1109/TPAMI.2004.92.
- VxL. (2010). . Retrieved from <http://vxl.sourceforge.net/>.
- Wiedemann, C., Ulrich, M., & Steger, C. (2008). Learning, tracking and recognition of 3D objects. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems IROS94* (Vol. 1, pp. 132-141). Springer-Verlag. Retrieved from <http://www.springerlink.com/index/10.1007/978-3-540-69321-5>.
- Xiao, J., Moriyama, T., Kanade, T., & Cohn, J. F. (2003). Robust full-motion recovery of head by dynamic templates and re-registration techniques. *International Journal of Imaging Systems and Technology*, 13(1), 85-94. doi: 10.1002/ima.10048.
- Zhang, Z., Deriche, R., Faugeras, O., & Luong, Q. T. (1995). A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar. Retrieved from <http://linkinghub.elsevier.com/retrieve/pii/0004370295000224>.