

GUILLERMO ANGEL PÉREZ LÓPEZ

**Casamento de modelos baseado em projeções radiais e circulares
invariante a pontos de vista**



São Paulo

2015

GUILLERMO ANGEL PÉREZ LÓPEZ

**Casamento de modelos baseado em projeções radiais e circulares
invariante a pontos de vista**

Tese apresentada à Escola Politécnica da
Universidade de São Paulo para obtenção
do título de Doutor em Ciências

Área de concentração: Sistemas
Eletrônicos

Orientador: Prof. Dr. Hae Yong Kim

São Paulo

2015

Na memória do meu pai, minha mãe e meu irmão, falecidos durante meu curso de doutorado.

Foi fácil tomar a decisão de estudar em outro país, mas é difícil lidar com a ideia de que aquela decisão me separou de vocês para sempre.

AGRADECIMENTOS

Grato a todas as pessoas que contribuíram direta ou indiretamente para a elaboração e aprimoramento desta tese, quer tivessem consciência disso ou não.

Obrigado meu Deus, tua força invisível sempre esteve ao meu lado. Que seria da minha vida sem a presença do Senhor? Meus logros são parte da tua obra divina.

Expresso minha felicidade por ter a honra de concluir a pós-graduação na Universidade de São Paulo (USP). Agradeço imensamente à USP por ter me dado a possibilidade e facilitado os meios para conseguir meus objetivos.

Agradecimento especial ao Prof. Dr. Hae Yong Kim, meu orientador, pelas suas importantes orientações e pela paciência que teve comigo durante este período.

Agradeço a assessoria, sugestões e contribuições dos professores da Escola Politécnica, de maneira especial a: Dra. Maria Cristina Vidal Borba, Dr. Ademar Ferreira, Dr. Marcelo Knorich Zuffo, Dr. Jose Alberto Quintanilha, Dr. Emilio Del Moral Hernandez, e Dr. Ana Maria.

Agradeço as importantes sugestões dos professores: Dr. Nina Sumiko Tomita Hirata, do IME-USP. Dr. Sidnei Alves Araujo, da Universidade Uninove, São Paulo. Dr. Celso S. Kurashima, da UFABC São Paulo. E Dr. Guoshen Yu, da Ecole Polytechnique Paliseau, França.

Agradeço as duas instituições que me deram o apoio financeiro para minha manutenção. Obrigado à Fundação Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e ao Conjunto Residencial da Universidade de São Paulo (CRUSP).

Muitos amigos e colegas contribuíram bastante para a qualidade deste trabalho. Agradeço a todos por sua ajuda e suas críticas construtivas. Agradeço de maneira especial ao meu amigo Ricardo Hitoshi e sua família. A minha querida Leyla Varela pela convivência enriquecedora. Aos meus colegas de laboratório e colegas de apartamento. Aos meus amigos do grupo de futsal, o grupo de voleibol e o grupo de basquetebol pelos momentos compartilhados.

Finalmente agradeço o amor, a paciência e o incentivo de meus familiares. Querida sobrinha Laura Marcela muito obrigado pelo carinho.

RESUMO

Este trabalho aborda o problema de casamento entre duas imagens. Casamento de imagens pode ser do tipo casamento de modelos (template matching) ou casamento de pontos-chaves (keypoint matching). Estes algoritmos localizam uma região da primeira imagem numa segunda imagem. Nosso grupo desenvolveu dois algoritmos de casamento de modelos invariante por rotação, escala e translação denominados Ciratefi (*Circula, radial and template matchings filter*) e Forapro (*Fourier coefficients of radial and circular projection*). As características positivas destes algoritmos são a invariância a mudanças de brilho/contraste e robustez a padrões repetitivos. Na primeira parte desta tese, tornamos Ciratefi invariante a transformações afins, obtendo Aciratefi (*Affine-ciratefi*). Construimos um banco de imagens para comparar este algoritmo com Asift (*Affine-scale invariant feature transform*) e Aforapro (*Affine-forapro*). Asift é considerado atualmente o melhor algoritmo de casamento de imagens invariante afim, e Aforapro foi proposto em nossa dissertação de mestrado. Nossos resultados sugerem que Aciratefi supera Asift na presença combinada de padrões repetitivos, mudanças de brilho/contraste e mudanças de pontos de vista. Na segunda parte desta tese, construimos um algoritmo para filtrar casamentos de pontos-chaves, baseado num conceito que denominamos de coerência geométrica. Aplicamos esta filtragem no bem-conhecido algoritmo Sift (scale invariant feature transform), base do Asift. Avaliamos a nossa proposta no banco de imagens de Mikolajczyk. As taxas de erro obtidas são significativamente menores que as do Sift original.

Palavras chaves: Casamento de modelos. Transformação afim. Simulação de pontos de vista. Padrões repetitivos. Mudança de iluminação. Invariância a escala. Invariância afim. Forapro. Ciratefi. Sift. Asift. Pontos-chaves.

ABSTRACT

This work deals with image matching. Image matchings can be modeled as template matching or keypoints matching. These algorithms search a region of the first image in a second image. Our group has developed two template matching algorithms invariant by rotation, scale and translation called Ciratefi (circular, radial and template matching filter) and Forapro (Fourier coefficients of radial and circular projection). The positive characteristics of Ciratefi and Forapro are: the invariance to brightness/contrast changes and robustness to repetitive patterns. In the first part of this work, we make Ciratefi invariant to affine transformations, getting Aciratefi (Affine-ciratefi). We have built a dataset to compare Aciratefi with Asift (Affine-scale invariant feature transform) and Aforapro (Affine-forapro). Asift is currently considered the best affine invariant image matching algorithm, and Aforapro was proposed in our master's thesis. Our results suggest that Aciratefi overcome Asift in the combined presence of repetitive patterns, brightness/contrast and viewpoints changes. In the second part of this work, we filter keypoints matchings based on a concept that we call geometric coherence. We apply this filtering in the well-known algorithm Sift (scale invariant feature transform), the basis of Asift. We evaluate our proposal in the Mikolajczyk images database. The error rates obtained are significantly lower than those of the original Sift.

Keywords: Template matching. Viewpoints simulation. Repetitive patterns. Illumination changes. Scale invariant. Affine invariant. Keypoints. Forapro. Ciratefi. Sift. Asift.

LISTA DE FIGURAS

Fig. 2-1. Exemplo do casamento de pontos-chaves	19
Fig. 2-2. Exemplo do casamento de modelos.....	20
Fig. 2-3. Conversão de casamento de pontos-chaves em casamento de modelos...21	
Fig. 2-4. Transformações geométricas no plano 2D.....	24
Fig. 2-5. Mudança visual causada pela variação do ponto de vista.	25
Fig. 2-6. Variação do brilho/contraste.	27
Fig. 2-7. Submodelo ou padrão de uma imagem.....	28
Fig. 2-8. Exemplos de padrões repetitivos	28
Fig. 3-1. Primeira etapa do Forapro.....	30
Fig. 3-2. Funcionamento do Forapro básico.	31
Fig. 3-3. Forapro robusto a escala.....	32
Fig. 3-4. Exemplo do funcionamento de Ciratefi, imagens fonte.....	33
Fig. 3-5. Espaço de escala da imagem Q construído por Cifi.....	34
Fig. 3-6. Imagem A processada por Cifi.....	34
Fig. 3-7. Imagem resultante da filtragem por Cifi.	35
Fig. 3 -8. Filtragem baseado em projeções radiais (Rafi).....	36
Fig. 3-9. Casamento de modelos por Ciratefi, etapa final (Tefi).	37
Fig. 3 -10. Transformação em perspectiva induzida por mudança do ponto de vista.	40
Fig. 3 -11. Processo de normalização de regiões.....	41
Fig. 3 -12. Interpretação geométrica de um ponto de vista fotográfico.....	44
Fig. 3 -13. Exemplo da técnica de simulação de pontos de vista	46
Fig. 3-14. Exemplo de um espaço de escala usando núcleo Gaussiano.....	50
Fig. 4-1. Imagens que apresentam mudanças de ponto de vista.	56
Fig. 4-2. Imagens simuladas por transformações afins.....	57
Fig. 4-3. Casamento de modelos invariante a transformações afins.....	58
Fig. 4-4. Aciratefi invariante a mudanças de brilho/contraste.	58
Fig. 4-5. Aciratefi invariante a padrões que se repetem.....	59
Fig. 4-6. Representação de dois pares de pontos-chaves.	62

Fig. 5-1. Imagens de busca <i>Q</i> utilizadas no Conjunto-1.....	69
Fig. 5-2. Imagens de análise pertencentes ao grupo B1 do Conjunto-1.....	69
Fig. 5-3. Imagens modelo <i>Q</i> utilizadas no Conjunto-2.....	70
Fig. 5-4. Imagens de análise <i>A</i> pertencentes ao grupo D2 do Conjunto-2.	71
Fig. 5-5. Imagens <i>Q</i> utilizadas no Conjunto-3.....	71
Fig. 5-6. Imagens <i>A</i> pertencentes ao grupo C3 do Conjunto-3.....	72
Fig. 5-7. Exemplo dos resultados da invariância a pontos de vista.....	73
Fig. 5-8. Limitações de Aciratefi em invariância a pontos de vista.	74
Fig. 5-9. Exemplo dos resultados da invariância a mudanças de brilho/contraste..	75
Fig. 5-10. Exemplo dos resultados da invariância a padrões repetitivos.	76
Fig. 5-11. Quadro comparativo da região de casamento do Asift, Aforapro e Aciratefi.	78
Fig. 5-12. Imagens de referência <i>I1</i> utilizadas nos testes de filtragem de casamentos pela coerência geométrica.....	83
Fig. 5-13. Sequência de imagens “Boat” com variações de escala e rotação.	83

LISTA DE GRÁFICOS

Gráfico 5- 1. Taxa de acertos de Aciratefi, Aforapro e Asift.....	78
Gráfico 5- 2. Tempo de execução, Aciratefi e Aforapro vs. Asift.	80
Gráfico 5- 3. Taxa de acertos entre Boat 1 e Boat 4.....	85
Gráfico 5- 4. Taxa de acertos entre Bikes 1 e Bikes 6.	87
Gráfico 5- 5. Taxa de acertos entre UBC 1 e UBC 6.....	89
Gráfico 5- 6. Taxa de acertos para variações de iluminação.....	91
Gráfico 5- 7. Taxa de acertos entre Graf 1 e Graf 4.....	93

LISTA DE TABELAS

Tab.4- 1. Diferenças entre Ciratefi e Sift.....	54
Tab. 5-1. Taxa de acertos e tempo de processamento no Conjunto-1.....	73
Tab. 5-2. Taxa de acertos e tempo de processamento no Conjunto-2.....	75
Tab. 5-3. Taxa de acertos e tempo de processamento no Conjunto-3.....	77
Tab. 5-4. Tempos médios de execução.	80
Tab. 5-5. Comparação de características: Aciratefi, Aforapro e Asift.	81
Tab. 5-6. Taxa de erros para imagens com variações de escala e rotação, aceitando casamentos com $R < 0,8$	84
Tab. 5-7. Taxa de erros para imagens com borrão de foco, aceitando casamentos com $R < 0,8$	86
Tab. 5-8. Taxa de erros para imagens com diferentes níveis de compressão JPEG, aceitando casamentos com $R < 0,8$	88
Tab. 5-9. Taxa de erros para imagens com variações de iluminação, aceitando casamentos com $R < 0,8$	90
Tab. 5-10. Taxa de erros para imagens com variações do ponto de vista, aceitando casamentos com $R < 0,8$	92

SUMÁRIO

RESUMO.....	5
ABSTRACT	6
LISTA DE FIGURAS	7
LISTA DE GRÁFICOS.....	9
LISTA DE TABELAS	10
SUMÁRIO.....	11
1. INTRODUÇÃO.....	14
1.1. Preliminares	14
1.2. Objetivos	16
1.2.1. Objetivos gerais.....	16
1.2.2. Objetivos específicos.....	16
1.3. Publicações associadas a este trabalho	16
1.4. Contribuições	17
1.5. Organização.....	17
2. CONCEITOS UTILIZADOS.....	18
2.1. Imagem digital.....	18
2.2. Casamento de imagens	19
2.2.1. Casamento de pontos-chaves (<i>keypoint matching</i>).....	19
2.2.2. Casamento de modelos (<i>template matching</i>)	20
2.2.3. Conversão de casamento de pontos-chaves em casamento de modelos 20	
2.2.4. Dificuldades de casamento de imagens	21
2.3. Transformações geométricas no plano	22
2.3.1. Translação	22
2.3.2. Rotação	22
2.3.3. Escala.....	23
2.3.4. Transformação afim.....	23
2.3.5. Transformação em perspectiva	24
2.4. Três definições usadas nesta tese.....	25
2.4.1. Definição 1: invariância a pontos de vista.....	25
2.4.2. Definição 2: invariância a brilho/contraste	26

2.4.3.	Definição 3: robustez a padrões repetitivos	27
3.	ENFOQUE DO TEMA	29
3.1.	Algoritmo Forapro.....	29
3.1.1.	Forapro, etapa 1: fundamental	29
3.1.2.	Forapro, etapa 2: básica	31
3.1.3.	Forapro, etapa 3: robustez a escala	32
3.2.	Algoritmo Ciratefi.....	33
3.2.1.	Filtragem baseada em projeções circulares (Cifi).....	33
3.2.2.	Filtragem baseada em projeções radiais (Rafi)	35
3.2.3.	Filtragem baseada em casamento de modelos (Tefi).....	36
3.3.	Características invariantes	37
3.4.	Invariância a pontos de vista vs. invariância afim	39
3.4.1.	Normalização de regiões	40
3.4.2.	Simulação de pontos de vista	44
3.5.	Algoritmos Sift e Asift	46
3.5.1.	Algoritmo Sift	46
3.5.2.	Algoritmo Asift.....	48
3.6.	Invariância a escala.....	49
3.7.	Robustez a padrões repetitivos.....	51
3.8.	Invariância a mudanças de brilho/contraste	53
4.	ABORDAGEM DA PROPOSTA	54
4.1.	Aciratefi	54
4.1.1.	Aciratefi invariante a pontos de vista	56
4.1.2.	Aciratefi invariante a padrões repetitivos e mudanças de brilho/contraste 58	
4.1.3.	Funcionamento do Aciratefi	59
4.2.	Melhoramento do casamento de pontos-chaves pela coerência geométrica 61	
4.2.1.	Coerência geométrica entre dois casamentos.....	62
4.2.2.	Testes para verificar se dois casamentos são coerentes	64
4.2.3.	Descarte de casamentos sem coerência.....	65
4.2.4.	‘Modo n’ de ordenação de casamentos	65
4.2.5.	‘Modo f’ de ordenação de casamentos	66
4.2.6.	Teste de filtragem por coerência geométrica.....	66

5. RESULTADOS.....	68
5.1. Testes de avaliação do algoritmo Aciratefi.....	68
5.1.1. Descrição da base de imagens utilizada	68
5.1.2. Testes de invariância a pontos de vista.....	72
5.1.3. Testes de invariância a mudanças de brilho/contraste	74
5.1.4. Testes de invariância a padrões repetitivos.....	76
5.1.5. Análise dos resultados.....	77
5.2. Testes de avaliação da etapa de Filtragem de casamentos	82
5.2.1. Descrição da base de imagens utilizada	82
5.2.2. Testes com variações de escala e rotação.....	84
5.2.3. Testes com borrão de foco	85
5.2.4. Testes com compressão JPEG	87
5.2.5. Testes variações de iluminação.....	89
5.2.6. Testes com variações do ponto de vista.....	91
5.2.7. Análise da avaliação da etapa de filtragem de casamentos	93
6. CONCLUSÕES	95
6.1. Conclusões gerais.....	95
6.2. Comentários finais.....	95
6.2.1. Comentários sobre Aciratefi	95
6.2.2. Comentários sobre a filtragem geométrica	95
6.3. Trabalhos Futuros	96
REFERÊNCIAS.....	97

1. INTRODUÇÃO

1.1. Preliminares

Nesta tese, abordamos o tema de casamento de imagens, interpretado como casamento de modelos (template matching) ou casamento de pontos-chaves (keypoint matching). São algoritmos que procuram regiões correspondentes entre duas imagens. No casamento de modelos procura-se uma imagem modelo numa outra imagem analisada. No casamento de pontos-chaves procura-se correspondências de pontos-chaves entre duas imagens.

O casamento de imagens é útil em tarefas como reconhecimento de objetos, navegação robótica, reconstrução de imagens, registro de imagens, processamento de imagens médicas, entre outras. A captura das imagens pode gerar fotografias com diferentes condições fotométricas e/ou de posicionamento da câmera, dificultando o casamento das imagens. Em consequência, dadas duas imagens I_1 e I_2 de uma mesma cena, as regiões da imagem I_1 podem estar transformadas em relação às regiões correspondentes da imagem I_2 . As transformações fotométricas são induzidas por variações na iluminação, efeitos de sombras e/ou variações do fundo da imagem. As variações no ponto de vista da captura podem induzir transformações de escala, rotação e translação. Outra dificuldade no processo de casamento são os padrões repetitivos, isto é, subpartes da imagem I_1 que apresentam similaridades com subpartes de I_2 fora das regiões procuradas.

As variações de posição e/ou direção da câmera produzem transformações em perspectiva que dificultam o casamento. Alguns trabalhos como Harris-affine (MIKOLAJCZYK; SCHMID, 2004) e Hessian-affine, (MIKOLAJCZYK; SCHMID, 2002) aproximam a transformação em perspectiva para uma transformação afim. Depois, utilizam a técnica denominada normalização de região para obter a invariância afim. A normalização de região visa eliminar a transformação em perspectiva sofrida. Segundo Morel e Yu (2009), nenhum dos algoritmos baseados em normalização de região é estritamente invariante às transformações afins, já que a transformação afim e o borrão de foco introduzido pela câmera normalmente não são comutáveis.

Morel e Yu (2009) propõem outra solução para obter a invariância afim, simulando várias vistas. Esta proposta consiste em gerar uma série de imagens com diferentes transformações afins das duas imagens a serem casadas, visando achar a correspondência entre duas simulações que compensem as transformações sofridas pelas imagens. Morel e Yu propõem o casamento de pontos-chaves invariante afim chamado Asift (*Affine-invariant Sift*), que utiliza a simulação de vistas e o algoritmo Sift (*scale invariant feature transform*) de Lowe (2004). Asift é considerado atualmente o melhor algoritmo invariante a pontos de vista (YU; MOREL, 2011; TONG, et al., 2013; WANG, et. al., 2015). No entanto, Asift apresenta limitações na presença de grandes mudanças de brilho/contraste e padrões repetitivos (MOREL; YU, 2009).

O nosso grupo desenvolveu previamente dois algoritmos de casamento de padrões invariante por rotação, escala e translação: Forapro (*Fourier Coefficients of Radial and Circular Projection*) (KIM, HAE 2010) e Ciratefi (*Circular and Radial Template Matching Filter*) (KIM; ARAUJO, 2007; ARAUJO, 2009). O nosso primeiro desafio foi tornar estes algoritmos invariantes a transformações afins, para obter Aciratefi (*Affine-ciratefi*) e Aforapro (*Affine-forapro*). No nosso trabalho de mestrado, foi testada uma versão preliminar do algoritmo Aforapro (PEREZ, 2011; KIM; PEREZ, 2010) que demonstrou superar as limitações mencionadas do Asift (grandes mudanças de brilho/contraste e padrões repetitivos). Nesta tese, abordamos novamente o algoritmo Aforapro e propomos o novo algoritmo Aciratefi. Realizamos testes comparativos das nossas propostas com Asift.

O nosso segundo desafio foi melhorar o algoritmo Sift (LOWE, 2004), base do Asift. Construímos uma nova ordenação dos casamentos de pontos-chaves para aumentar a taxa de acertos no Sift, baseado no conceito “coerência geométrica” que introduzimos. Esta filtragem geométrica foi avaliada utilizando a base de imagens de Mikolajczyk (MIKOLAJCZYK, 2015), frequentemente utilizada para avaliar estes tipos de algoritmos. Os resultados sugerem que a filtragem geométrica consegue aumentar significativamente a taxa de acertos do Sift.

1.2. Objetivos

1.2.1. Objetivos gerais

- Propor uma técnica para o casamento de modelos invariante a transformações afins, mudanças de brilho/contraste e robustez a padrões repetitivos.
- Propor uma “filtragem” que diminua as taxas de erros de algoritmos de casamento de pontos-chaves, como Sift.

1.2.2. Objetivos específicos

- Tornar Ciratefi invariante a transformações afins, para obter Aciratefi.
- Construir uma base de imagens apropriadas para avaliar Aciratefi nos itens: invariância a transformações afins, mudanças de brilho/contraste e robustez a padrões repetitivos.
- Comparar os desempenhos de Aciratefi, Aforapro e Asift.
- Procurar uma base de imagens apropriada para realizar os testes da filtragem geométrica.
- Implementar e validar a proposta da filtragem geométrica.
- Descrever as principais abordagens relatadas na literatura sobre casamento de imagens invariante a transformações afins.

1.3. Publicações associadas a este trabalho

- No artigo PEREZ e KIM (2011), apresentamos resultados que demonstraram as características do Aforapro: invariância a transformações afins, robustez a padrões repetitivos e invariância às mudanças de iluminação. Também foram apresentados resultados comparativos que sugerem que Aforapro supera algumas das dificuldades de Asift. Este artigo foi escolhido entre os três melhores do Simpósio Brasileiro de Telecomunicações, Curitiba. SBrT 2011.

- No artigo PEREZ e KIM (2014), apresentamos Aciratefi e o comparamos com Aforapro e Asift. Avaliamos invariância a pontos de vista, padrões repetitivos e robustez a mudanças de brilho/contraste. Descrevemos e analisamos as vantagens e desvantagens dos três algoritmos.

1.4. Contribuições

- Nesta tese foi proposto um método de casamento de modelos invariante afim denominado Aciratefi. Este método é também invariante a mudanças de brilho/contraste e robusto a padrões repetitivos.

- Foi realizada uma avaliação comparativa entre os algoritmos Aciratefi, Aforapro e Asift.

- Foi proposta uma ferramenta que filtra casamentos falsos de pontos-chaves. Esta ferramenta foi testada no algoritmo Sift.

1.5. Organização

O restante da tese está organizado como segue. No Capítulo 2, apresentamos os conceitos principais utilizados neste trabalho. No Capítulo 3 descrevemos os algoritmos Ciratefi, Forapro, Sift e Asift. No Capítulo 4 apresentamos as nossas propostas: Aciratefi e a filtragem geométrica. No Capítulo 5 apresentamos resultados e análises dos experimentos realizados. No Capítulo 6 apresentamos as nossas conclusões.

2. CONCEITOS UTILIZADOS

Neste capítulo apresentamos os principais conceitos utilizados no trabalho.

2.1. Imagem digital

Uma imagem digital é uma função $f(x, y)$ que foi discretizada em coordenadas espaciais e níveis de intensidade. Os índices de linhas e colunas (x, y) identificam a posição do pixel, e o valor do pixel $f(\cdot)$ identifica a sua intensidade (MOESLUND, 2012). Uma imagem pode ser representada por uma matriz ou uma série de matrizes $2D$ de números inteiros. As imagens coloridas são representadas por três ou quatro matrizes dependendo do sistema de cores empregado, uma matriz para cada banda de cor. Uma imagem com uma única matriz de intensidades é chamada de imagem em níveis de cinza. Normalmente, os níveis de intensidade pertencem ao intervalo de valores $[0, 255]$. Este intervalo pode ser menor, como no caso das imagens binárias onde o intervalo é $[0, 1]$ (GONZALES; WOODS, 2002; PETROU; PETROU, 2010). Ou pode ser maior em imagens chamadas “deep color”, onde são alocados mais de 8 bits por canal.

Os algoritmos de casamento de imagens normalmente processam imagens em níveis de cinza, já que as cores de uma imagem colorida costumam variar de forma imprevisível com a mudança da iluminação. Funt et al. (1998) investigaram vários métodos utilizados no reconhecimento de objetos baseado em cores e concluíram que os métodos testados são insuficientes para lidar com o problema de constância de cores. Schaefer (2004) investigou a utilidade de características invariantes extraídas de cores na recuperação de imagens (image retrieval), e concluiu que as invariantes de cores nem sempre são úteis. Em casos específicos, a cor pode ser uma propriedade útil (MOESLUND, 2012; NIXON; AGUADO, 2012; TAM, 2011).

Geralmente, considera-se que uma imagem digital está constituída por um fundo (*background*) e um ou vários objetos que ficam sobre ele (*foreground*). Os objetos são definidos como regiões constitutivas que se diferenciam do fundo (e.g. uma letra ou caractere, uma forma ou silhueta, um automóvel, um prédio, etc.) (THEODORIDIS; KOUTROUMBAS, 2009; NIXON; AGUADO, 2012).

2.2. Casamento de imagens

Neste item, abordaremos dois tipos de casamento de imagens: casamento de pontos-chaves e casamento de modelos.

2.2.1. Casamento de pontos-chaves (*keypoint matching*)

O princípio implícito do casamento de pontos-chaves é a não utilização da imagem inteira para a procura, mas a utilização apenas de pontos “importantes” denominados pontos-chaves, e armazenar a informação sobre a vizinhança destes pontos numa sequência de características (vetor descritor) (BRAHMBHATT, 2014). Os pontos-chaves são pixels que apresentam algum tipo de característica discriminante, que permite localizá-los novamente mesmo após terem sofrido transformações como rotação, mudança de escala e mudança de brilho/contraste. Assim, a principal característica de um ponto chave é sua repetibilidade, que expressa a capacidade do detector de achar o mesmo ponto sob diferentes condições visuais (FISHER, et. al., 2014; GRAUMAN; LEIBE, 2011). Uma forma usual de representar um casamento de pontos-chaves é utilizar linhas para unir os pontos casados, como indicado na Fig. 2-1.



Fig. 2-1. Exemplo do casamento de pontos-chaves (a) Imagem I_1 , (b) Imagem I_2 , (c) Casamento, linhas indicando os pixels casados. Alguns destes casamentos apresentam falsas correspondências, devido talvez à variação da iluminação e/ou do ponto-de-vista.

O casamento de pontos-chaves pode ser dividido em três principais passos (BAY, et. al., 2008; SZELISKI, 2011): (a) detecção, na qual os pontos-chaves são selecionados; (b) descrição, na qual as particularidades da vizinhança de cada ponto chave são extraídas e armazenadas num vetor de características; (c) casamento, no qual a similaridade entre pontos-chaves é avaliada, e os pontos correspondentes são agrupados em duplas.

2.2.2. Casamento de modelos (*template matching*)

O casamento de modelos é uma técnica utilizada para procurar um modelo Q (imagem de consulta ou imagem de busca) numa imagem de análise A (imagem teste). O objetivo é localizar uma instância de Q em A (BRAHMBHATT, 2014; BRUNELLI, 2009). Normalmente, um indicador é traçado no local da instância achada, para advertir a localização do casamento, como ilustrado na Fig. 2-2.



Fig. 2-2. Exemplo do casamento de modelos.

(a) Imagem modelo Q , (b) Imagem de análise A , (c) Casamento, indicado por meio de um rótulo branco sobre instância do modelo achado.

Costuma-se usar a imagem Q de tamanho menor do que a imagem A . O modelo Q pode aparecer uma ou mais vezes na imagem de análise A . Alguns algoritmos como Ciratefi podem ser configurados para casar mais de uma instância do modelo Q numa única imagem A .

2.2.3. Conversão de casamento de pontos-chaves em casamento de modelos

Neste item, descrevemos como transformar um algoritmo de casamento de pontos-chaves num algoritmo de casamento de modelos, usando a transformada generalizada de Hough (DI RUBERTO, 2012; CYGANNEK, 2013; LOWE, 1999). Os casamentos verdadeiros podem ser filtrados e agrupados em conjuntos de pontos-chaves de um mesmo objeto, utilizando sua localização, escala e orientação. Na Fig. 2-3 apresentamos um exemplo do casamento de modelos utilizando Sift e transformada de Hough.

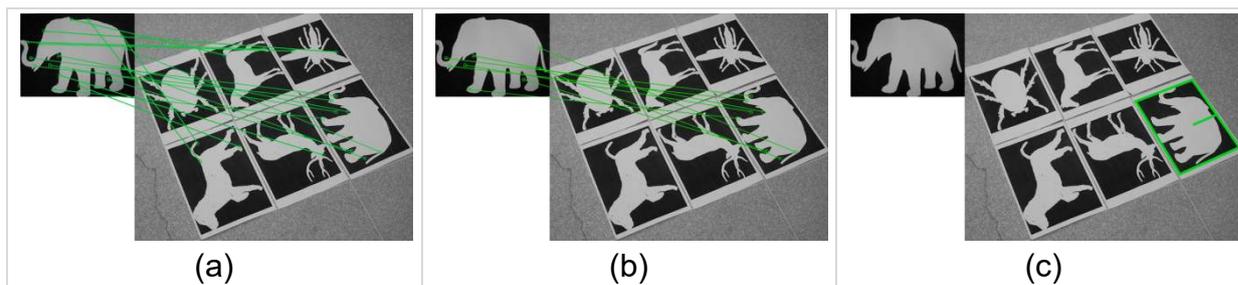


Fig. 2-3. Conversão de casamento de pontos-chaves em casamento de modelos. (a) casamento de pontos-chaves utilizando Sift. (b) agrupamento de casamentos por regiões, utilizando transformada generalizada de Hough. (c) casamento de modelos utilizando Sift e transformada de Hough. Note que a imagem (a) apresenta uma elevada taxa de casamentos falsos e que todos eles são eliminados na filtragem da imagem (b).

A transformada de Hough foi proposta originalmente por Hough (1962) e generalizada por Ballard (1981). Na proposta original, a finalidade foi detecção de formas definidas por funções analíticas, como linhas, círculos, elipses, etc. A transformada generalizada de Hough estende o campo de ação para detectar formas arbitrárias (HASSANEIN, 2015).

A transformada de Hough tem duas ideias principais: (a) acumular votos para cada possível forma paramétrica constituída por pixels da imagem; (b) selecionar as formas paramétricas com maior número de votos (AGUADO, et. al., 2002). Por exemplo, Sift seguida pela transformada generalizada de Hough para obter um casamento de modelos: (a) cada ponto-chave de Sift inclui a sua localização (x,y) , escala s e orientação θ , e um casamento consiste de dois pontos-chaves nas duas imagens Q e A ; (b) usando estas informações, cada casamento vota uma vez no espaço de Hough no centro do objeto; (c) se um ponto no espaço de Hough apresentar suficientes votos, então considera-se que detectou uma ocorrência de Q em A . No trabalho de Cao (2008) apresenta-se um exemplo detalhado sobre o casamento de modelos utilizando Sift e a transformada generalizada de Hough.

2.2.4. Dificuldades de casamento de imagens

O nível de dificuldade num casamento de imagens aumenta à medida em que as imagens apresentem variabilidades que podem ser causadas por (TREIBER, 2010):

- Efeitos fotométricos: incluem as posições e distribuição das fontes de luz na cena, o efeito de iluminação mútua entre objetos, a distribuição de sombras e

reflexão nas superfícies dos objetos físicos. Este tipo de efeito tende a mudar drasticamente a distribuição de brilho/contraste nas imagens.

- Ponto de vista: as imagens podem ser capturadas a partir de diferentes pontos de vista - direções e distâncias -, este efeito pode induzir transformações em perspectivas que deformam, transladam, rotacionam e/ou mudam a escala dos objetos na imagem.
- Disposição dos objetos: em cenas naturais os objetos raramente ficam isolados, eles se encontram parcialmente oclusos sobre algum fundo ou com outros objetos.
- Mudança de forma: muitos objetos físicos podem manter sua identidade enquanto mudam de forma - propriedades elásticas.

Os modelos matemáticos podem fornecer alguns tipos de soluções para superar estes problemas como, por exemplo, utilização dos invariantes. Este tema será abordado no item 3.3.

2.3. Transformações geométricas no plano

A transformação de uma imagem $I_1(x_1, y_1)$ é um mapeamento que leva essa imagem a outra imagem $I_2(x_2, y_2)$ (PERWASS, 2009).

Algumas transformações geométricas, relacionadas a matriz homográfica, são descritas a seguir.

2.3.1. Translação

Para transladar ou mover um objeto dentro duma imagem deve-se adicionar uma quantidade (T_x, T_y) nas suas coordenadas:

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \end{bmatrix}$$

Eq.2. 1

2.3.2. Rotação

Um objeto numa imagem pode ser rotado ou girado por um ângulo θ :

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

Eq.2. 2

Numa imagem, tipicamente o valor das coordenadas no eixo y aumenta no sentido inverso ao tradicional: y aumenta na medida em que descemos na imagem. Assim, se $\theta > 0$ a orientação do giro será no sentido horário e para $\theta < 0$ anti-horário.

2.3.3. Escala

Mudar a escala de um objeto significa multiplicar suas coordenadas por um fator de escala S_i .

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$$

Eq.2. 3

Quando muda de escala, um objeto pode manter ou não suas proporções de tamanho. A mudança de escala é uniforme se $S_x = S_y$ e é não uniforme caso contrário (MAJI; PAUL, 2014). O valor dos fatores determina o tipo de redimensionamento: (a) $S_i > 1$ aumenta o tamanho, (b) $0 < S_i < 1$ diminui o tamanho, (c) $S_i = 1$ mantém o tamanho original.

2.3.4. Transformação afim

A transformação afim de uma imagem $I(x, y)$ é uma transformação linear com seis graus de liberdade e pode ser escrita como (BERGOUNIOUX, 2011):

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} 1 & k \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$$

Eq.2. 4

onde S_x e S_y são fatores de escala, k é um fator de não ortogonalidade entre os eixos, ϕ é o ângulo de rotação sobre o plano, e os fatores e e f representam a translação. Desenvolvendo esta equação obtemos:

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} S_x(\cos \phi + k \sin \phi) & S_x(-\sin \phi + k \cos \phi) \\ S_y \sin \phi & S_y \cos \phi \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$$

Eq.2. 5

Que também pode ser expressa assim:

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$$

Eq.2. 6

Uma transformação afim no espaço euclidiano possui as seguintes propriedades (TREIBER, 2010):

- Preserva a relação de colinearidade entre pontos, isto é, três pontos colineares continuam sendo colineares após a transformação. Como consequência desta propriedade, a transformação afim preserva o paralelismo das retas.
- Preserva a razão das distâncias ao longo de uma linha, isto é, para pontos colineares distintos (p_1, p_2, p_3) a razão $|p_2 - p_1|/|p_3 - p_2|$ é preservada.

As transformações afins incluem translação, rotação, mudança de escala uniforme e não uniforme, assim como suas combinações. A Fig. 2-4 apresenta alguns exemplos de transformações afins.

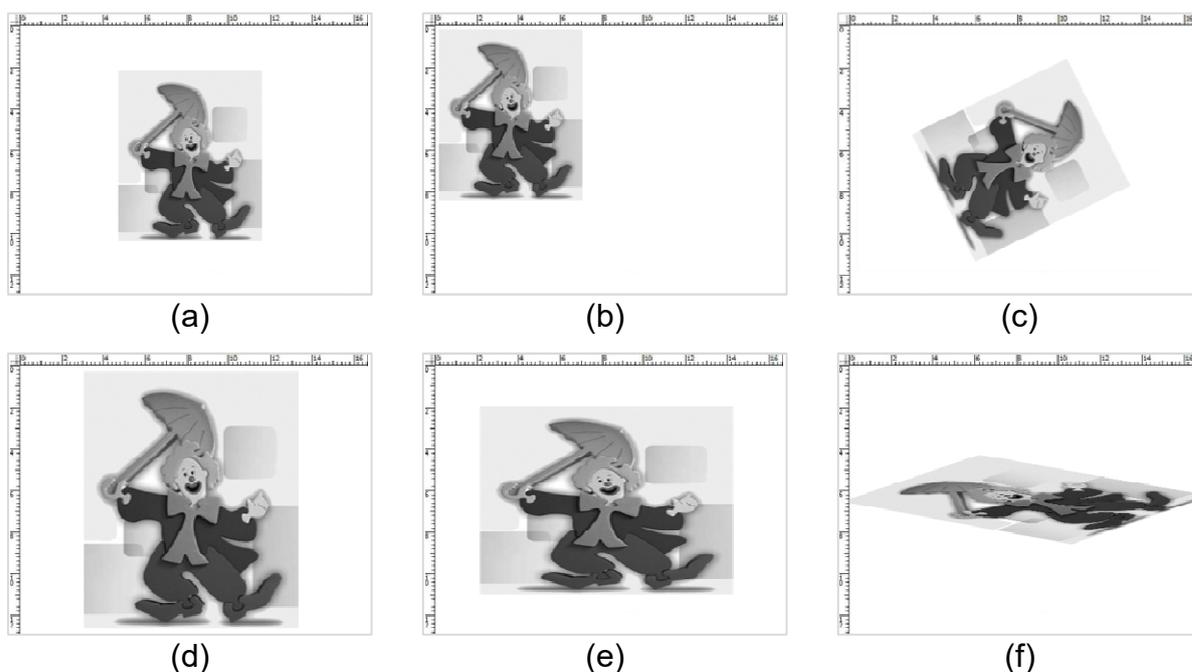


Fig. 2-4. Transformações geométricas no plano 2D. (a) imagem original, (b) translação, (c) rotação, (d) mudança uniforme de escala, (e) mudança não uniforme de escala, (f) transformação afim.

2.3.5. Transformação em perspectiva

Diferente das transformações geométricas já mencionadas, a transformação em perspectiva (ou projetiva) é uma transformação não-linear. A extração das imagens do mundo real 3D para o plano da imagem 2D executada por uma câmera ideal “buraco de agulha” (pin-hole) é uma transformação em perspectiva. A projeção das imagens realizadas por uma câmera fotográfica real se aproxima de uma

transformação em perspectiva, desprezando as distorções da lente e dos sensores. Neste processo, os pontos são projetados no plano da imagem, dividindo as coordenadas x e y pela terceira coordenada z (BERGOUNIOUX, 2011):

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = P_z(x_1, y_1, z) = \begin{bmatrix} x_1/z \\ y_1/z \\ 1 \end{bmatrix}$$

Eq.2. 7

onde P_z é a projeção na imagem do ponto (x_1, y_1, z) no mundo 3D. Depois da projeção, a informação de profundidade pertencente a terceira coordenada, não pode ser recuperada. Uma transformação em perspectiva pode causar uma inclinação (*tilt*) no plano da imagem relativo ao plano do objeto (TREIBER, 2010). Uma perspectiva pode ser aproximada por uma transformação afim em alguns casos. Falaremos deste tema no próximo capítulo.

2.4. Três definições usadas nesta tese

Nesta seção apresentamos as definições das três características invariantes utilizadas como parâmetros de avaliação em nossos testes experimentais.

2.4.1. Definição 1: invariância a pontos de vista

O ponto de vista está relacionado com a posição espacial da câmera em relação à cena fotografada, na hora de capturar a imagem (PRATT, 2007). A Fig. 2-5 apresenta dois pontos de vista do mesmo terreno (vista frontal e vista lateral). Note como o panorama visual muda de um ponto de vista para o outro.

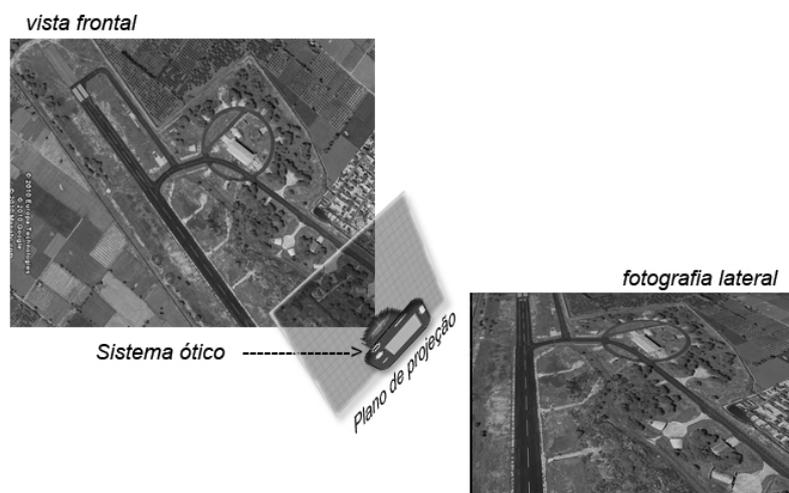


Fig. 2-5. Mudança visual causada pela variação do ponto de vista.

A mudança do ponto de vista causa uma transformação em perspectiva que aumenta a dificuldade do casamento de imagens.

Definição 1:

Sejam duas imagens, imagem modelo Q e imagem de análise A . Um algoritmo de casamento de imagens é invariante a pontos de vista se consegue achar instâncias de Q que aparecem sob diferentes pontos de vista na imagem A .

2.4.2. Definição 2: invariância a brilho/contraste

Dizemos que duas imagens A e Q são equivalentes sob variações de iluminação se existem fatores de correção de contraste $\beta > 0$ e de brilho γ tais que (ARAÚJO; KIM, 2011):

$$Q = \beta A + \gamma \mathbf{1} \quad \text{Eq.2. 8}$$

onde $\mathbf{1}$ é uma matriz de 1's.

O brilho de uma imagem pode ser medido calculando o nível de cinza médio da imagem. O contraste de uma imagem pode ser medido pelo desvio-padrão da imagem:

$$\sigma = \left[\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2 \right]^{1/2} \quad \text{Eq.2. 9}$$

onde x_i é o valor normalizado ($0 \leq x_i \leq 1$) do nível de cinza, \bar{x} é a média normalizada de todos os níveis de cinza na imagem, e n é o número de pixels da imagem. Na Fig. 2-6 são ilustrados os conceitos das mudanças de brilho/contraste.

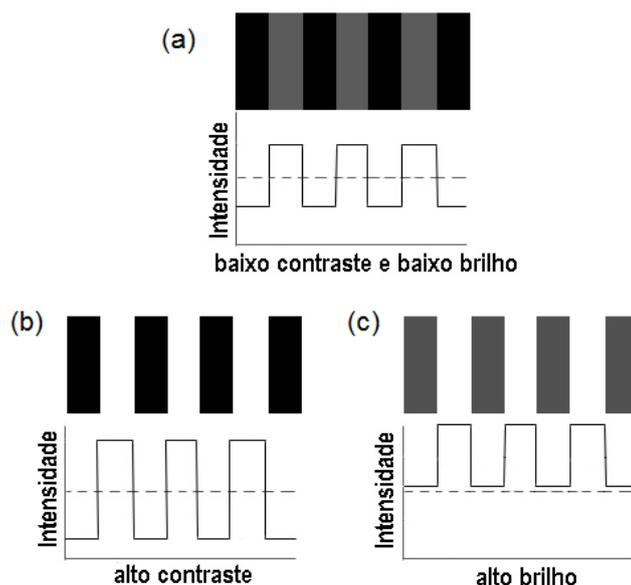


Fig. 2-6. Variação do brilho/contraste.

A imagem (b) é obtida aumentando o contraste da imagem (a). A imagem (c) é obtida aumentando o brilho da imagem (a).

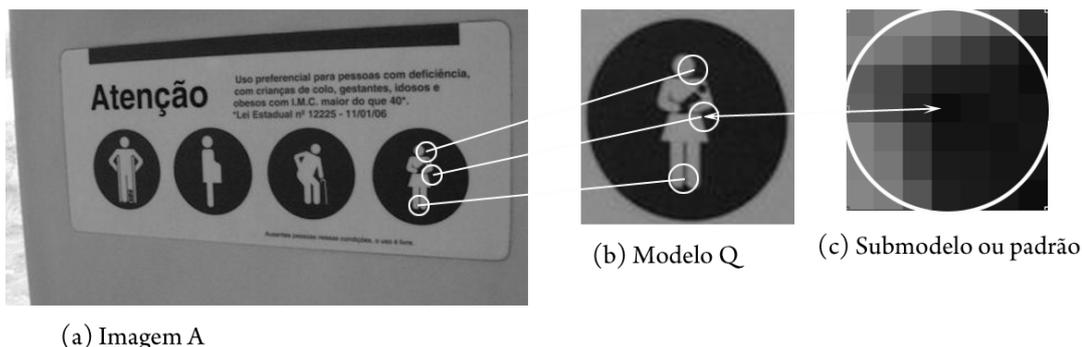
A variação do brilho/contraste nas imagens normalmente acontece por fatores tais como mudança de iluminação e efeitos de sombras. Dada a impossibilidade de se manter um controle constante sobre estes fatores, a invariância a mudanças de brilho/contraste adquire importância no casamento de imagens.

Definição 2 (PEREZ; KIM, 2011):

Sejam duas imagens, imagem modelo Q e imagem de análise A . Definimos que um algoritmo de casamento de modelos é invariante a mudanças de brilho/contraste se consegue achar instâncias de Q em A independentemente das mudanças de brilho/contraste presentes na instância.

2.4.3. Definição 3: robustez a padrões repetitivos

Um modelo pode ser dividido em submodelos ou padrões. Os padrões são pequenas regiões a partir das quais é possível extrair algum tipo de característica (BRADSKI, 2008). Na Fig. 2-7 são apresentados o modelo e os padrões de um casamento.

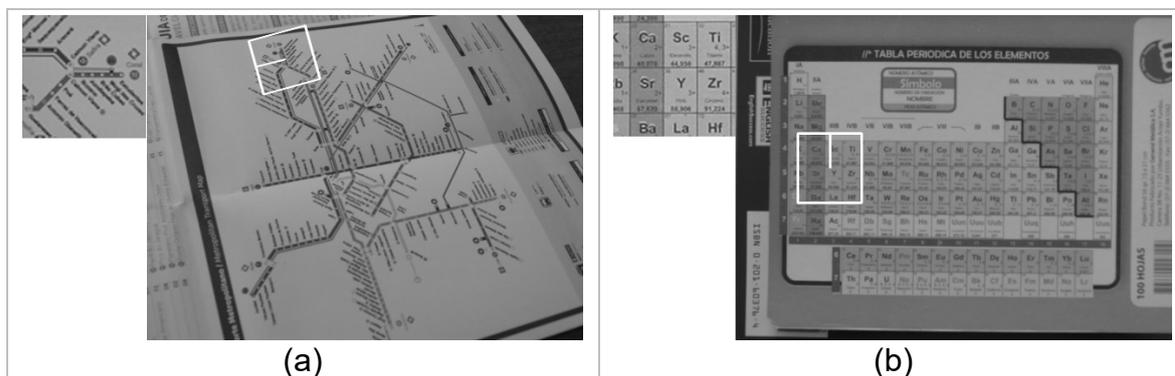


(a) Imagem A

Fig. 2-7. Submodelo ou padrão de uma imagem.

As imagens apresentam o casamento de um modelo Q utilizando três submodelos ou padrões. As linhas brancas entre a imagem A e os padrões de Q indicam as correspondências. Na imagem (c) um padrão é ampliado.

As instâncias de um modelo podem conter padrões que aparecem repetidamente, no fundo ou em outros objetos da imagem, como apresentado na Fig. 2-8. Os padrões repetitivos são um problema no casamento de imagens porque, dado que eles são indistinguíveis entre si, aumentam o número de casamentos errados (DOUBEK, et. al., 2010; WENDEL; DONOSER; BISCHOF, 2010).



(a)

(b)

Fig. 2-8. Exemplos de padrões repetitivos

(a) Mapa das linhas do trem, o modelo procurado é um cruzamento de linhas, (b) Tabela periódica dos elementos químicos, o modelo procurado é um grupo de elementos. Note que os modelos apresentam padrões que aparecem repetidamente em outras partes da imagem além de aparecer na instância correspondente. Estes casamentos foram realizados com o algoritmo Aforaprop.

Definição 3 (PEREZ; KIM, 2011):

Sejam duas imagens, imagem modelo Q e imagem a analisar A , onde n instâncias de Q aparecem em A . Dizemos que A possui padrões de Q repetitivos se algumas subpartes de Q aparecerem em A fora das n instâncias. Consideramos que um algoritmo de reconhecimento de objetos é robusto a padrões repetitivos se consegue achar as n instâncias de Q em A , mesmo que A possua padrões de Q repetitivos.

3. ENFOQUE DO TEMA

Neste capítulo, descreveremos os algoritmos e as características invariantes utilizados na tese.

3.1. Algoritmo Forapro

Forapro (*Fourier coefficient of radial and circular projection*) (KIM, 2010) foi desenvolvido em nosso grupo de pesquisa. Este é um algoritmo de casamento de modelos. Forapro apresenta invariância à rotação, translação, mudanças de brilho/contraste, e robustez a mudanças de escala e a padrões repetitivos. Para facilitar a nossa análise, separamos Forapro em três etapas: fundamental, básica e Forapro robusto a escala.

3.1.1. Forapro, etapa 1: fundamental

Forapro fundamental calcula, para todos os pixels p da imagem A , características invariantes por rotação e brilho/contraste numa área circular em torno de p . O cálculo deste campo denso de características é acelerado utilizando FFT (transformada rápida de Fourier) e núcleos circulares/radiais. Além das características, Forapro também calcula um ângulo “canônico” para cada pixel. Para fazer casamento de modelos, Forapro calcula as características na região circular em torno do pixel central de Q . Calculando as distâncias entre as características em torno de cada pixel de A e de Q , resulta a imagem P (Fig. 3-1). Quanto mais escuro for um pixel de P , maior é a chance de uma instância de Q aparecer em A . Limiarizando P e sobrepondo à imagem A , obtemos L , que localizou corretamente a ocorrência de elefante em A (Fig. 3-1). Para maiores detalhes de como as características são calculadas, veja Kim (2010).

Como em qualquer casamento baseado em características, Forapro pode gerar falsos positivos, já que uma região com muitos pixels é mapeada num vetor de características de dimensão menor que o número de pixels original. Para eliminar este e outros problemas, Forapro utiliza etapas complementares.

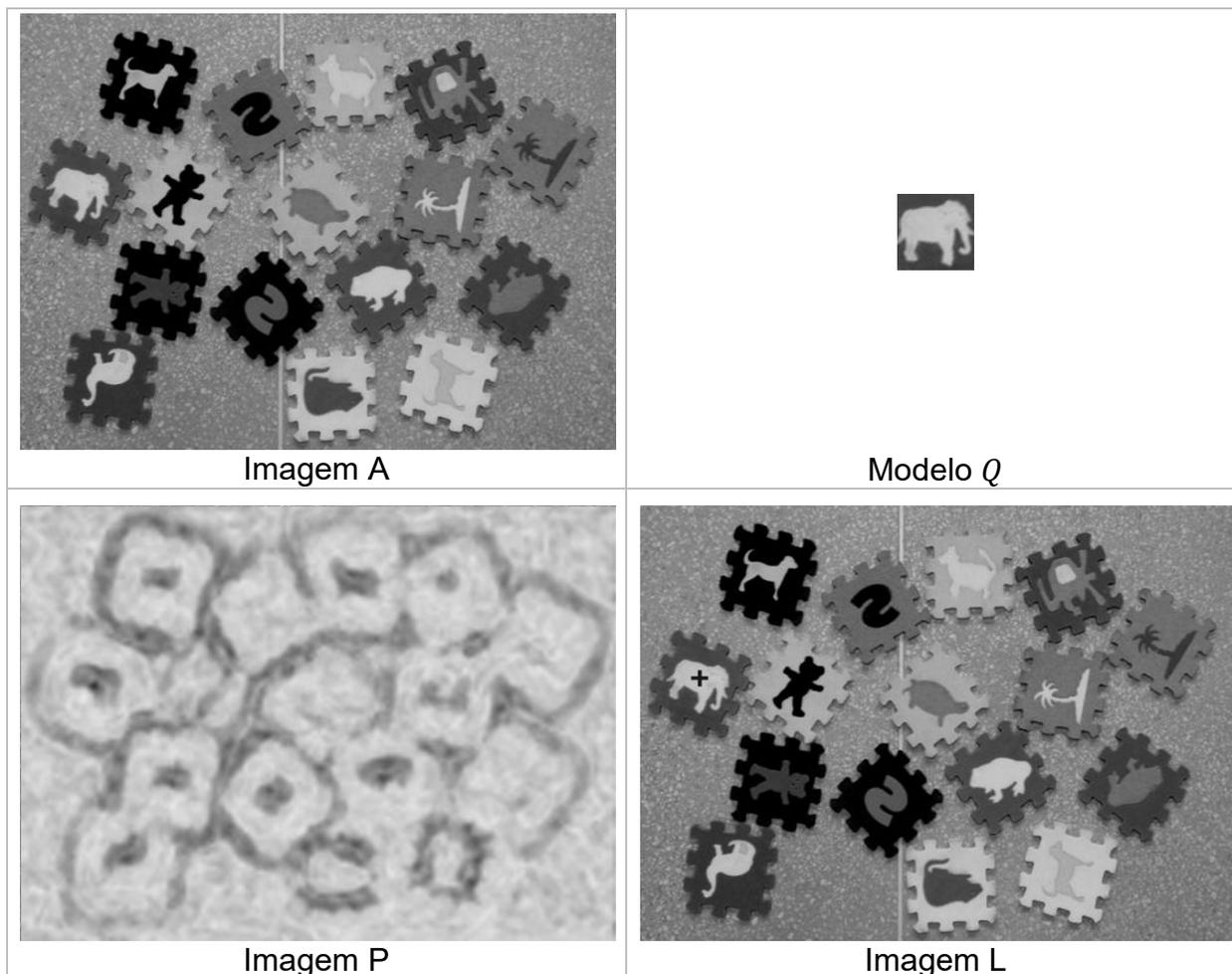


Fig. 3-1. Primeira etapa do Forapro.

Forapro fundamental calcula as características invariantes por rotação e brilho/contraste para cada pixel de A e para o pixel central de Q . Somente os pixels dentro do círculo inscrito em Q são levados em consideração. Calculando a distância entre essas características, resulta a imagem P . Limiarizando P e sobrepondo na imagem A , obtemos a imagem L , onde aparece a marca (x) sobre a instância achada.

Forapro fundamental busca o modelo Q na imagem A comparando diretamente o vetor de características de Q com o vetor de características de cada pixel de A . Assim, o número de comparações corresponde ao produto entre a dimensão dos vetores de características e o número total de pixels da imagem A . Isto normalmente será um número muito grande. Em consequência, o processo de busca pode levar um tempo de processamento considerável.

Por outro lado, para buscar um número grande de modelos Q_i em A , o campo vetorial de características da imagem A precisa ser calculado uma única vez. Isto

acelera o processamento, dado que o cálculo das características (que é a parte com mais alto custo computacional) é executado uma única vez.

3.1.2. Forapro, etapa 2: básica

Nesta etapa, utiliza-se Forapro fundamental para atingir dois objetivos: (a) eliminar falsos positivos; (b) obter casamento de modelos robusto à oclusão parcial. A Fig. 3-2 ilustra o funcionamento do Forapro básico. O número N de submodelos e o raio r dos submodelos são parâmetros dados pelo usuário. Dados N e r , o algoritmo escolhe submodelos $T_1, T_2, \dots, T_N \subset Q$ de raio r . Para cada submodelo, executa-se o Forapro fundamental e calculam-se os pixels onde o modelo Q provavelmente ocorre. Estes pixels candidatos podem ser verdadeiros positivos ou falsos positivos.

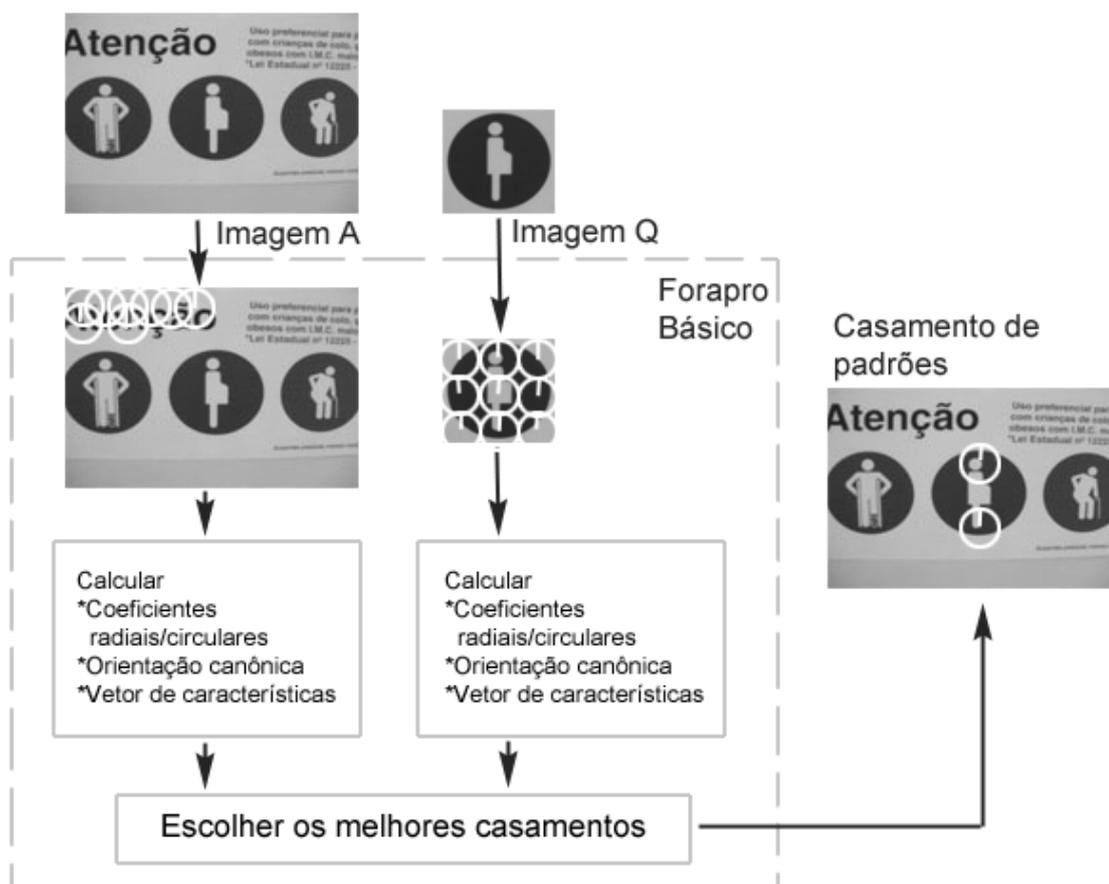


Fig. 3-2. Funcionamento do Forapro básico.

Neste exemplo, o modelo Q foi dividido em nove submodelos de raio r . Cada pixel da imagem A foi avaliado levando a conta sua vizinhança numa área circular de raio r .

Forapro básico usa a transformada de Hough para eliminar os falsos positivos. Ele considera como casamentos válidos somente posições com grandes concentrações de casamentos de submodelos. Uma vantagem de se utilizar a transformada de Hough é que o algoritmo resultante torna-se robusto a oclusões parciais.

3.1.3. Forapro, etapa 3: robustez a escala

Forapro torna-se robusto à escala usando a seguinte estratégia: (a) redimensiona a imagem Q por um conjunto de fatores de escala, obtendo Q_0, Q_1, \dots, Q_n onde n é um parâmetro fornecido pelo usuário; (b) procura cada imagem redimensionada Q_i na imagem de análise A . Na Fig.3-3 ilustra-se o funcionamento do Forapro robusto a escala.

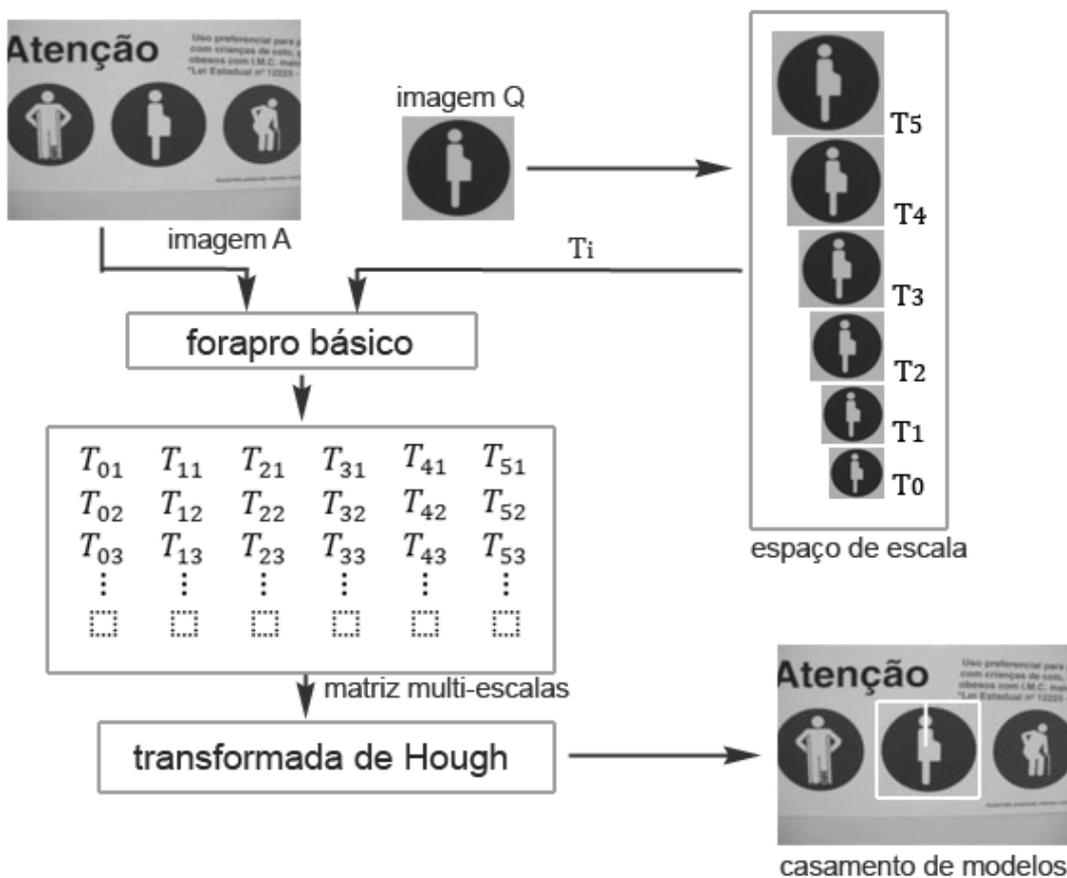


Fig. 3-3. Forapro robusto a escala.

O modelo Q é redimensionado em diferentes escalas, gerando modelos redimensionados Q_i . De cada modelo redimensionado Q_i são extraídos vários submodelos T_{ij} . Os submodelos são procurados na imagem A . O resultado final é obtido através da transformada generalizada de Hough.

No exemplo da Fig.3-3, a escala das instâncias de Q contidas em A pode variar de 0.3 a 1.2 vezes do tamanho de Q . Forapro extrai, de cada modelo redimensionado Q_i , os submodelos T_{ij} . Cada submodelo T_{ij} é procurado na imagem A . O casamento final é escolhido usando transformada generalizada de Hough.

3.2. Algoritmo Ciratefi

Ciratefi (*Circular and radial template matching filter*) (KIM; ARAUJO, 2007; ARAUJO, 2009) é um algoritmo de casamento de modelos invariante por rotação, translação, escala, brilho e contraste desenvolvido em nosso grupo de pesquisa. Ciratefi é robusto a padrões repetitivos.

Ciratefi procura o modelo Q utilizando três etapas consecutivas de filtragem. A seguir, descreveremos cada etapa seguindo um exemplo passo a passo.

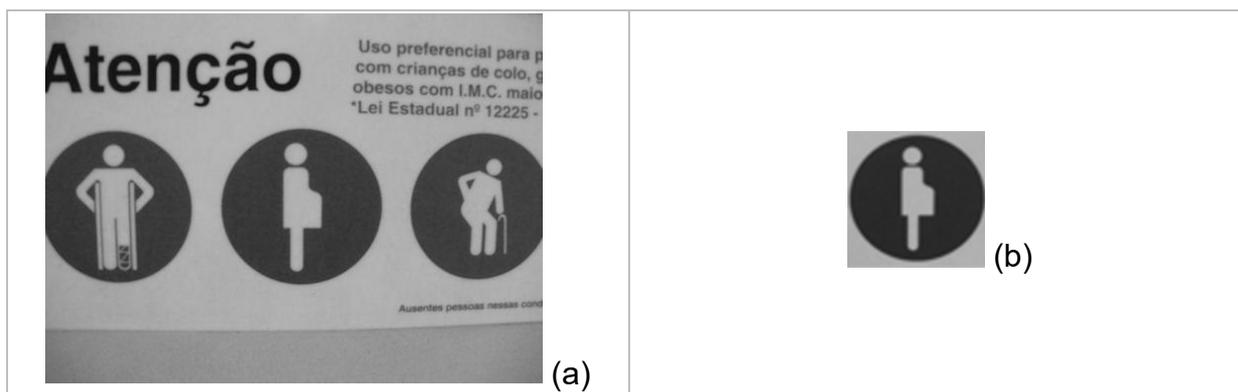


Fig. 3-4. Exemplo do funcionamento de Ciratefi, imagens fonte. (a) Imagem de análise A . (b) Imagem modelo Q . Tamanho original das imagens é 375x500 e 140x140 pixels, respectivamente. Foram utilizadas as mesmas imagens usadas com Forapro para comparar os dois algoritmos.

3.2.1. Filtragem baseada em projeções circulares (Cifi)

Cifi calcula pixels de A que têm alguma chance de casar com Q procurado e associa uma provável escala para cada candidato. Os pixels que não são candidatos serão descartados e não serão considerados nas etapas subsequentes. Denominaremos de “pixels candidatos de primeiro grau” ao conjunto de pixels escolhidos por Cifi.

A amostragem circular $Cis(x, y, k)$ (Eq.3.1) é a média dos níveis de cinza da imagem I , situados a uma distancia k do pixel central (x, y) .

$$Cis(x, y, k) = \frac{1}{2\pi k} \int_{\theta=0}^{2\pi} I(x + k \cos \theta, y + k \sin \theta)$$

Eq.3. 1

Na prática, uma somatória deve substituir o integral. Usando a imagem Q e a amostragem circular, Cifi constrói uma matriz multi-escala invariante a rotação, representada na Fig. 3-5.



Fig. 3-5. Espaço de escala da imagem Q construído por Cifi. Cifi gera uma sequência de imagens redimensionadas. As projeções circulares são aplicadas em cada imagem redimensionada. Neste exemplo foram utilizadas 6 escalas variando de 0,3 a 1,2.

Ciratefi original é invariante a escala num intervalo definido pelo usuário. No trabalho de Araújo e Kim (2011) foi apresentada uma melhora de Ciratefi onde o modelo Q é redimensionado para cinco valores usando uma progressão geométrica $\{0.5, 0.57, 0.66, 0.76, 0.87\}$, e é construída uma pirâmide multi-escala de A com valores de escalas $\{0.5, 0.25, 0.125, \dots\}$. Esta estratégia torna Ciratefi invariante à escala no intervalo $[0.5, \infty]$, e elimina a introdução manual dos parâmetros de escala. Esta versão melhorada do Ciratefi foi utilizada nos nossos testes.

Aplicando a Eq.3.1 a cada pixel da imagem A , Cifi constrói uma imagem 3D e guarda-a em formato de vídeo. Na Fig. 3-6 são apresentadas algumas imagens da sequência gerada.

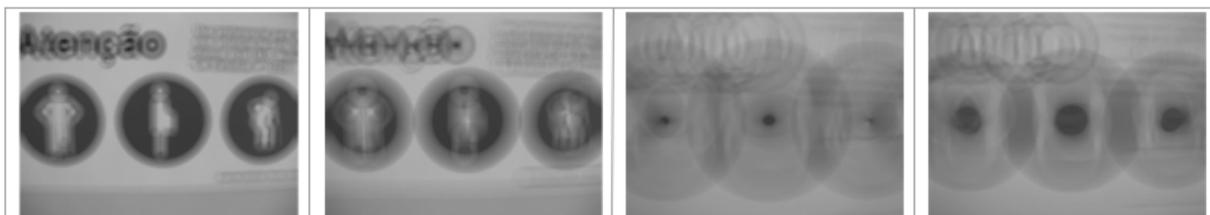


Fig. 3-6. Imagem A processada por Cifi. A terceira dimensão corresponde a valores predefinidos de raio k . Nesta figura apresentamos uma parte da sequência.

Cifi usa a matriz multi-escala obtida de Q e a imagem 3D obtida de A (Figs. 3-5 e 3-6) para calcular a correlação normalizada entre projeções circulares de Q e as

projeções circulares de cada pixel de A . Cifi procura, para cada pixel, a escala de maior correlação. Se a maior correlação for maior que certo limiar, o pixel é considerado um candidato de primeiro grau e a escala provável é a escala que deu a maior correlação. Em vez de usar um limiar de correlação, Ciratefi melhorado define que uma porcentagem (por exemplo, 75%) de total de pixels de A com as maiores correlações são candidatos de primeiro grau. A Fig. 3-7 apresenta em branco os candidatos de primeiro grau escolhidos por Cifi.



Fig. 3-7. Imagem resultante da filtragem por Cifi.

Cifi escolhe um grupo de pixels candidatos ao casamento, pixels marcados de branco, que representam os pixels candidatos de primeiro grau. Só estes pixels serão processados nas próximas etapas.

3.2.2. Filtragem baseada em projeções radiais (Rafi)

Rafi reavalia os pixels candidatos de primeiro grau (Fig.3-7). Alguns destes pixels são promovidos para o segundo grau. Os pixels não promovidos são descartados. Além disso, nesta segunda fase é associado um ângulo provável de rotação a cada pixel de segundo grau.

A amostragem radial Ras (Eq.3.2) é a média dos níveis de cinza dos pixels de uma imagem $I(x, y)$ numa linha radial.

$$Ras_I^\lambda(x, y, \alpha) = \frac{1}{N} \sum_{t=0}^{\lambda} I(x + t \cos \alpha, y + t \sin \alpha)$$

Eq.3. 2

onde N é o número de pixels ao longo da linha radial com inclinação α . E λ é o comprimento da linha radial, isto é, o raio do círculo maior utilizado por Cifi. Usando a Eq.3.2, Rafi constrói um vetor baseado na amostragem radial de Q (Fig.3.8a) e

uma matriz com os valores das projeções radiais para cada pixel de A candidato de primeiro grau na sua escala provável. Posteriormente, o vetor e a matriz gerados são correlacionados para obter os pixels candidatos de segundo grau (Fig.3-8b). Aqueles pixels que tiverem correlação acima de um limiar são considerados candidatos de segundo grau. No Ciratefi melhorado, em vez de usar um limiar, escolhe uma porcentagem de pixels com as maiores correlações. Também se associa um ângulo provável de rotação para cada pixel candidato de segundo grau. No final desta etapa, ainda pode haver pixels que geram casamentos errados. Na Fig.3-8c apresentamos a saída do Rafi.

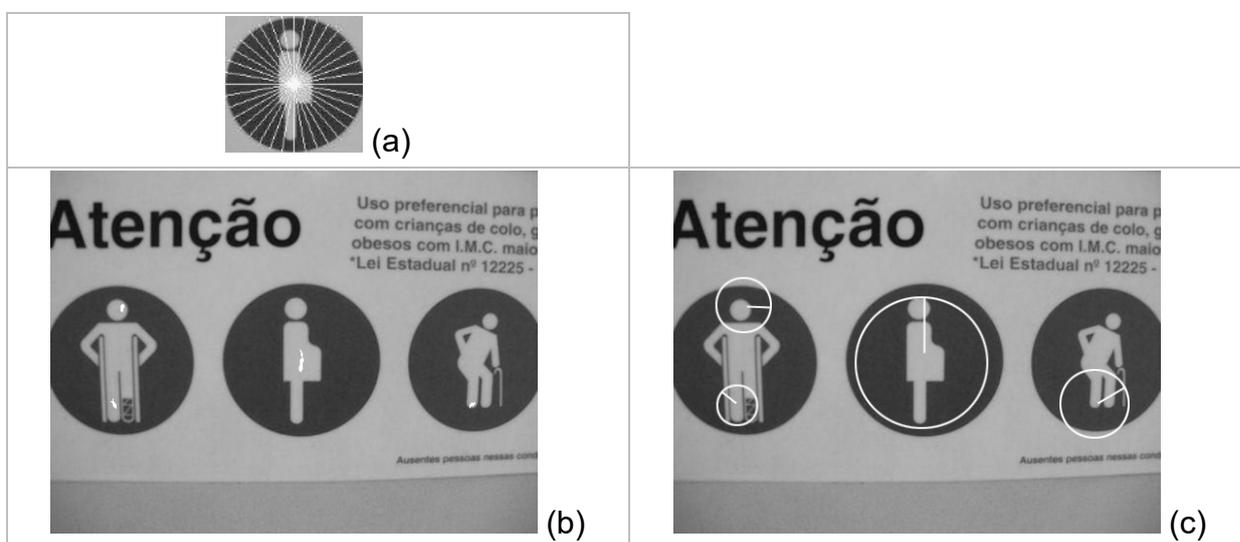


Fig. 3 -8. Filtragem baseado em projeções radiais (Rafi).
 (a) Projeções radiais na imagem Q ; (b) pixels candidatos de segundo grau, marcados em branco;
 (c) resultado do casamento de modelos, caso o processo termine nesta etapa. Neste exemplo temos três casamentos errados e um casamento certo.

Na Fig.3-8c, temos quatro casamentos, dos quais três são casamentos errados. Normalmente teremos pixels candidatos de segundo grau que geram casamentos errados, portanto, precisamos de uma terceira etapa de filtragem.

3.2.3. Filtragem baseada em casamento de modelos (Tefi)

Tefi reavalia os pixels candidatos de segundo grau. Calcula o coeficiente de correlação convencional para finalizar o processo de casamento. Nesta terceira etapa, restam poucos pixels candidatos de segundo grau a serem avaliados. Cada pixel candidato tem associado sua provável escala (s_i) e seu provável ângulo de inclinação (α_i). Isto simplifica o processo de casamento de padrão convencional

utilizado por Tefi, onde para cada pixel candidato $A(x, y)$ extrai-se uma sub-imagem em torno a sua vizinhança e compara-se com a imagem Q na escala s_i e o ângulo α_i . Se o valor absoluto da correlação for acima de certo limiar predefinido (t_3), então se considera que uma instância de Q foi detectada. No Ciratefi melhorado, em vez de usar limiar, detecta-se um número pré-definido de pixels. Na Fig.3-9 apresentamos o resultado final do casamento de modelos.



Fig. 3-9. Casamento de modelos por Ciratefi, etapa final (Tefi). O traço branco indica a instância da imagem Q achada. Neste caso, a posição de ponteiro indica rotação nula. O tamanho do círculo indica a escala aproximada na qual a instância foi achada.

3.3. Características invariantes

As características invariantes são parâmetros que se extraem de uma região da imagem, com o propósito de simplificar e robustecer a técnica de casamento de imagens. Idealmente estes invariantes devem conservar o seu valor mesmo que a imagem sofra determinados efeitos ou transformações. A robustez do processo de casamento de imagens depende da qualidade das características invariantes (CHEN, 2015).

Algumas das primeiras propostas para explorar as características invariantes a rotação, iluminação e escala, baseavam-se na técnica denominada de memórias associativas. A ideia básica consistia em armazenar um número suficiente de vistas associadas com a imagem, para depois comparar a imagem da vista atual com todas as vistas armazenadas na memória. Estas propostas, normalmente baseadas em redes neurais, poderiam armazenar grande quantidade de vistas $(I_1^1, I_2^1, \dots, I_n^1)$, para depois realizar comparações diretas da vista I_i^1 percorrendo a imagem I^2 (TREIBER, 2010). No entanto, a simples comparação direta entre duas vistas da

imagem não teve sucesso por duas razões: (a) a criação de uma base de imagens com todas as vistas que pode adotar um objeto é algo complexo; (b) é possível que a imagem a ser reconhecida não possua suficiente similaridade com as vistas armazenadas. No estado-da-arte esta ideia é novamente explorada, reduzindo estrategicamente o número de vistas associadas, por exemplo, na construção de características invariantes a transformações afins (MOREL; YU, 2009).

Duas propostas clássicas que exploram as características invariantes à rotação são os momentos centrais propostos por Hu (1962) e os momentos de Zernike (TEAGUE, 1980). Eles apresentam teorias baseadas em momentos aplicáveis a imagens binárias. Apesar de estas técnicas serem somente invariantes a rotação, elas são amplamente reconhecidas pela sua contribuição no desenvolvimento de outras técnicas como: (a) Os descritores baseados em momentos invariantes (VAN GOOL, et. al. 1996), onde os momentos são calculados pelas derivadas da intensidade da imagem; (b) Algumas propostas onde os autores demonstram que o valor de uma combinação específica de momentos invariantes em imagens binárias (FLUSSER; SUK, 1993) e imagens em níveis de cinza (WANG; et al., 2013) pode ficar constante mesmo que a região, da qual este foi derivado, tenha sofrido uma transformação afim. Isto evidencia que o uso dos momentos é uma alternativa para obter descritores invariantes a transformações afins. Porém, a quantidade de características invariantes afins que se consegue desta forma é muito reduzida.

A construção de invariantes utilizando a extração de características locais representa um panorama amplamente acolhido no estado da arte. As técnicas baseadas na extração de características locais utilizam dois passos estratégicos para descrever o conteúdo da imagem (MAJI; PAUL, 2014): (a) detecção de “pontos de interesse”, que são aqueles pixels facilmente localizáveis no plano espacial, no entanto, a imagem tenha experimentado uma transformação geométrica e/ou fotométrica (e.g. LOWE, 2004; MIKOLAJCZYK; SCHMID, 2004; BAY, 2008; RABIN, et. al. 2009; SHIN; TJAHHADI, 2010); (b) derivação do vetor de características, também chamado “descritor de regiões”. Cada vetor de características representa a informação da imagem extraída da vizinhança ao redor do “ponto de interesse” (e.g. MATAS, et. al. 2004; KIM; ARAÚJO, 2007; KIM, 2010; SHIN; TJAHHADI, 2010). O casamento de imagens pode ser executado, comparando os descritores das regiões

e suas localizações de configuração espacial. No estado da arte as propostas mais destacadas na construção de invariantes a transformações de rotação, translação, escala e transformações afins, baseiam-se na extração de características locais.

Algumas das vantagens das técnicas baseadas na extração de características locais são (TAM, PHUONG, 2011):

- A representação local está baseada na aparência, por isso não precisa da extração de primitivas geométricas (e.g. linhas, eixos, círculos, retângulos).
- Não requer segmentação prévia.
- Esta técnica apresenta robustez a oclusões parciais e a mudança de fundos, já que o casamento final não precisa da região toda.
- As variações complexas na aparência dos objetos causadas pela variação de pontos de vista e pelas condições na iluminação, podem se aproximar, usando a modelagem por meio de transformações a uma escala local.

3.4. Invariância a pontos de vista vs. invariância afim

O Postulado 3.4, amplamente aceito no estado da arte, relaciona a invariância a pontos de vista com a invariância a transformações afins.

Postulado 3.4 A aparente deformação da vista plana de um objeto, induzida pela mudança de ponto de vista, é uma transformação em perspectiva que pode ser localmente modelada por uma transformação afim (TUYTELAARS; GOOL, 2000; HSIAO, et. al., 2007; WU; YANG, 2008; MOREL; YU, 2009).

A Fig.3-10 apresenta uma deformação em perspectiva, induzida pela mudança de ponto de vista. Os retângulos sofrem uma transformação em perspectiva e tornam-se trapezoides. O retângulo externo tornou-se claramente um trapezoide que não pode ser aproximado por um paralelogramo. Os retângulos internos também se tornaram trapezoides, mas são aproximadamente paralelogramos. Isto permite aproximar localmente uma transformação em perspectiva (retângulo é mapeado em trapezoide) por uma transformação afim (retângulo é mapeado em paralelogramo). Esta aproximação não pode ser feita para a imagem toda.

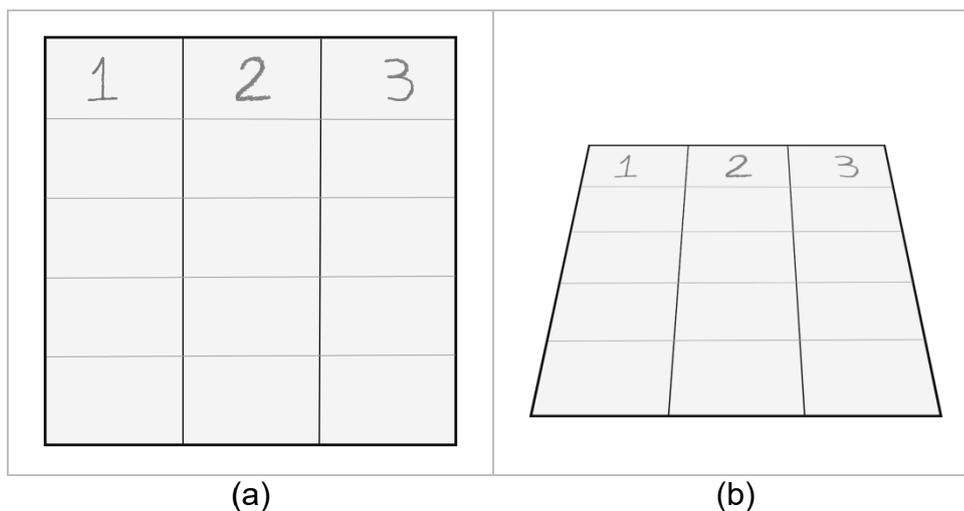


Fig. 3 -10. Transformação em perspectiva induzida por mudança do ponto de vista. (a) Ponto de vista frontal. (b) Mudança do ponto de vista, fotografada com 60 graus de inclinação. Na imagem b, cada retângulo interno deformado pode ser aproximadamente um paralelogramo, apesar de que a deformação em conjunto é claramente trapezoidal.

O Postulado 3.4, aplicado ao casamento de modelos, possibilita a obtenção de algoritmos invariantes a mudanças de pontos de vista. Assim, a instância da imagem Q que aparece com deformações em perspectiva na imagem A , pode ser procurada modelando a imagem Q com transformações afins. No trabalho apresentado por Alvarez, et al. (2015) é analisado o conceito de invariância a transformações afins e relacionado aos conceitos de distância entre pixels de uma região da imagem. Na literatura são duas as técnicas importantes na procura da invariância a pontos de vista (ou invariância a transformações afins): a normalização de regiões e a técnica de simulação de pontos de vista.

3.4.1. Normalização de regiões

A normalização de regiões padroniza de alguma forma os pixels numa região da imagem, procurando que os parâmetros extraídos daquela região sejam aproximadamente invariantes a uma transformação geométrica específica (TREIBER, 2010).

A normalização dos parâmetros da translação e da rotação é relativamente simples. O processo se reduz a transladar o centro do objeto ao centro do sistema de coordenadas e rotacioná-lo, levando em conta o ângulo de rotação do objeto determinado.

A normalização dos parâmetros da escala ou da inclinação é muito mais complexa, no entanto, existem alguns métodos que exploram este tipo de normalização. Na Fig.3-11 apresenta-se um exemplo da técnica de normalização utilizada para obter a invariância afim, pelo método do Hessian-affine (MIKOLAJCZYK; SCHMID, 2002).

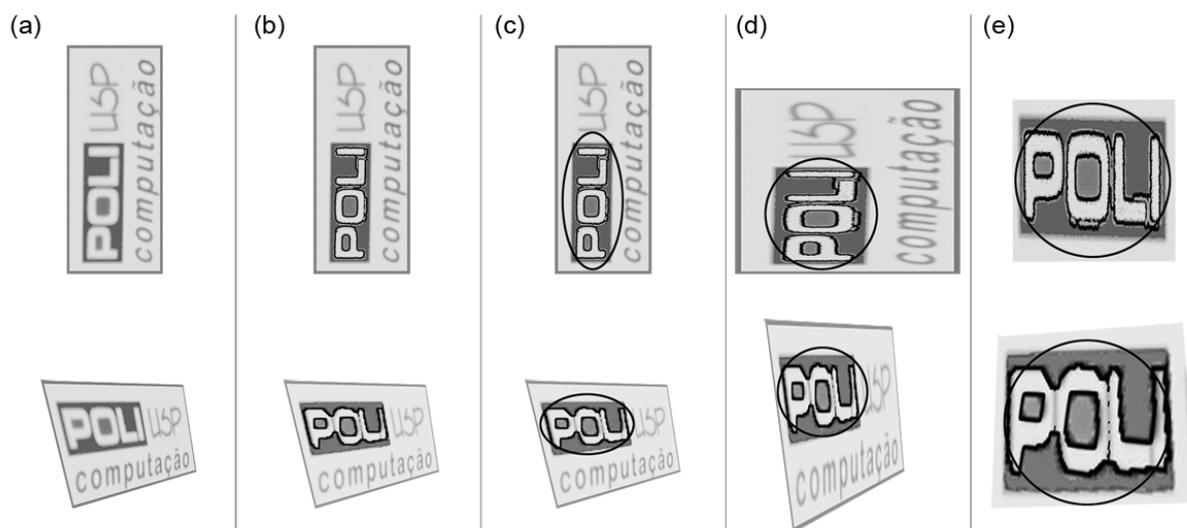


Fig. 3 -11. Processo de normalização de regiões. Duas instâncias do mesmo objeto, diferenciadas pela mudança do ponto de vista: a) imagens fonte, b) detecção da região do objeto, c) traço elíptico na área detectada, d) normalização geométrica para o círculo, e) normalização fotométrica e ajuste da rotação.

Alguns algoritmos que usam a técnica da normalização de parâmetros são:

Uma análise multi-escala para extrair cantos, proposto em (ALVAREZ; MORALES, 1997). Os autores deste trabalho assumem que a continuidade de um canto é dada por um trecho de linha formado por pontos e o declive desta linha identifica o ângulo do canto. Deste modo, é possível recuperar a localização inicial e orientação do canto para modelar a invariância afim. Porém, os pontos de interesse, em imagens de cenas reais, não podem ser aproximados por cantos perfeitos.

Um método para achar características afins em pequenas regiões de interesse foi proposto por Lindeberg e Garding (1997). Nele os pontos de interesse são extraídos usando valores máximos de intensidade numa representação espaço-escala, utilizando o detector de Harris (HARRIS; STEPHENS, 1988). A proposta de Lindeberg e Garding foi complementada por Baumberg (2000) que utilizou um detector de regiões de forma elíptica para melhorar a normalização. No entanto, os

pontos de interesse e suas regiões associadas não conseguem a invariância na presença de transformações afins com ângulo de inclinação maior que 60 graus.

Um algoritmo baseado em pontos de interesse invariantes que utiliza o detector de Harris-Laplace foi proposto por Schaffalitzky e Zisserman (2001). O foco desta proposta é uma normalização da transformada afim procurando correspondências em pequenas regiões previamente segmentadas.

Um algoritmo baseado em momentos invariantes obtidos de imagens binárias foi proposto por Flusser e Suk (1993). Uma falha presente neste tipo de proposta é a alta sensibilidade ao ruído. As duas propostas de Mikolajczyk e Schmid (2002; 2004), também baseadas em momentos invariantes, aproximam a região característica utilizando uma elipse que posteriormente é normalizada para um círculo. Nestas propostas, utiliza-se o detector Harris-affine e o Hessian-affine respectivamente.

Um algoritmo que, além da invariância afim, também apresenta invariância a oclusões parciais, baseado na transformada de Hough e complementado com a extração de contornos usando informação local, foi apresentado por Ecabert e Thiran (2004). Esta proposta demanda pouca capacidade de memória e tempo de execução, além de considerável robustez a deformações locais.

Algumas propostas para classificação de textura são baseadas em regiões invariantes afins (e.g. RUBNER; TOMASI, 1999; SCHAFFALITZKY; ZISSERMAN, 2001). Primeiro, utilizando um detector multi-escala, extrai-se uma pequena região da imagem. A seguir, a região é normalizada procurando um descritor invariante afim. Neste processo, utiliza-se um histograma de distribuição de intensidades. Na sequência, os descritores são distribuídos na forma de uma assinatura contendo um grupo central e valores relativos. Por fim, as assinaturas são comparadas e classificadas. Estes métodos são altamente sensíveis às mudanças de iluminação. Eles são ótimos para a restauração e classificação de imagens, mas não para processos como reconhecimento de objetos.

Um detector de regiões invariantes baseado em linhas de nível Mser (*Maximally stable extremal region*) foi apresentado por MATAS, et. al. (2004). O Mser baseia-se numa sequência de linearizações de intensidade, as regiões extremas são componentes conexos de um conjunto de máximo ou mínimo nível de intensidade. O Mser procura a invariância a pontos de vista normalizando os parâmetros da

transformada afim. Mser está definido pelos valores extremos de uma função de intensidade, numa região e suas regiões próximas.

O algoritmo “Clique descritor” (SHIN; TJAHJADI, 2010) estende o conceito de regiões invariantes do Mser. Nele, as características são agrupadas em pequenos conjuntos de regiões e correlacionadas por um mínimo número de ocorrências. O Clique descritor utiliza o Mser como detector de formas para combiná-lo com o descritor do Sift (LOWE, 2004). Ele tenta fazer melhor uso da influência da vizinhança entre regiões invariantes. Uma das vantagens desta fusão é a invariância a oclusões parciais. Esta fusão do Mser com o Sift é utilizada no Mm-Sift (CHEN, et al., 2013). O princípio básico do Mm-Sift é a utilização de áreas elípticas na extração de características, que logo são normalizadas para círculos (como na Fig. 3-11), e com o auxílio do Sift é construído o descritor. Os autores declaram que, Mm-Sift obtém menos casamentos do que Asift, no entanto, que o Mm-Sift é muito mais rápido.

O Mser foi avaliado no trabalho de MIKOLAJCZYK (2005) como o melhor detector, em comparação a outros detectores baseados na extração de pontos-chaves e invariantes a transformações afins. Porém, a técnica do Mser utiliza componentes de alto contraste na construção das regiões invariantes, o que constitui uma limitação, porque, as imagens de cenas naturais contêm poucas componentes de alto contraste.

Uma proposta para o registro de imagens invariante afim que utiliza a normalização de regiões, recentemente proposto por Zhang, et al. (2015), combina a normalização com o algoritmo Mser (Maximally Stable Extremal Regions) (MATAS; CHUM, 2004). Nesta proposta extrai-se informação prévia de cada imagem utilizando Mser. Utiliza-se a informação extraída para construir a matriz de transformação entre as duas imagens. Utilizando os centroides, as regiões detectadas são normalizadas para regiões circulares, como na Fig.3-11. Das regiões normalizadas são extraídos alguns pontos de interesse invariantes a escala e transformações afins, utilizando o filtro de Gabor (XU, et al., 2011). Finalmente, os pontos de interesse são filtrados e agrupados para construir uma matriz de transformação entre os conjuntos formados.

Uma limitação dos métodos baseados em normalização é a não-comutação entre o borramento do conjunto ótico da câmera e a transformação afim (MOREL;

YU, 2009). Assim, o processo de normalização não elimina exatamente a deformação, este fato é demonstrado matematicamente por Hartley e Zisserman (2004). Esta demonstração baseia-se no fato de que: quando as deformações incluem grandes escalamentos e/ou grandes inclinações, a base da compensação executada pela normalização aumenta sua complexidade.

3.4.2. Simulação de pontos de vista

A ideia da simulação de pontos de vista aparece no trabalho de Molton, et. al. (2004). Posteriormente é otimizada e apresentada com sucesso por Morel e Yu (2009), no trabalho onde apresenta Asift. Esta técnica baseia-se na interpretação geométrica do sistema formado pelo plano do objeto e a câmera fotográfica, como ilustrado na Fig. 3-12.

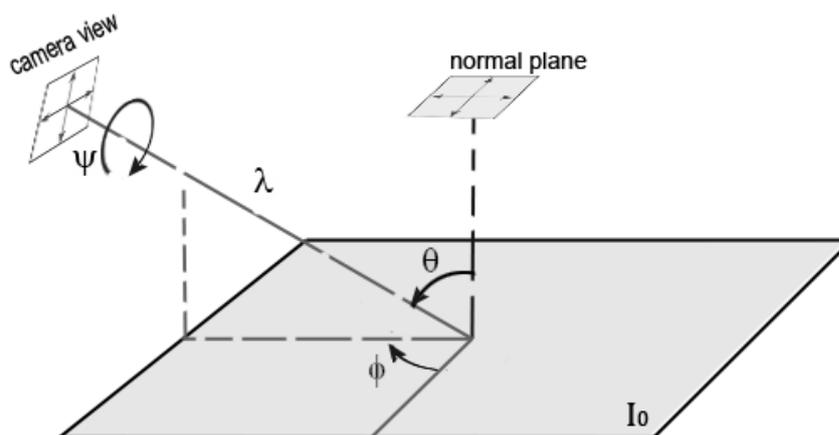


Fig. 3 -12. Interpretação geométrica de um ponto de vista fotográfico.

O plano I_0 representa a vista frontal do objeto físico; ψ é o ângulo de rotação da câmera sobre seu eixo óptico; λ é o parâmetro de escala e representa a distância entre a câmera e o ponto central do plano; ϕ é o ângulo de rotação da câmera sobre o plano da imagem; e θ é o ângulo de inclinação calculado entre a normal do plano da imagem e o eixo óptico da câmera. Assume-se que a vista frontal está determinada pelos valores; $\lambda = 1$ e $\psi = \phi = \theta = 0$.

Este modelo configura o núcleo da matriz de transformação ou matriz homográfica M segundo a Eq.3.3:

$$M = H_\lambda R_1(\psi) T_t R_2(\phi) = \lambda \begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix} \begin{bmatrix} t & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix}$$

Eq.3.3

onde H_λ é a matriz de escala, com $\lambda > 0$; as R_i são as rotações; $\phi \in [0, \pi)$ intervalo suficiente para ter uma boa aproximação das transformações; e T_t é a mudança de inclinação com $t = 1/\cos \theta$. O parâmetro t (tilt), que representa o grau de inclinação de uma vista para outra, é utilizado com duas definições: tilt absoluto, que é a diferença de inclinação de uma imagem para sua vista frontal; e tilt de transição, que é a medida de inclinação entre dois pontos de vista de uma imagem. Em nosso trabalho, consideramos somente o tilt absoluto, dado que é razoável supor que no casamento de modelos a imagem Q sempre apresenta a vista aproximadamente frontal.

A simulação de pontos de vista utiliza os valores de θ e ϕ para simular todas as possíveis distorções, causadas pela mudança de orientação do eixo ótico da câmera. A escolha correta do intervalo de valores destes dois parâmetros garante que um pequeno número de imagens simuladas seja suficiente para ter correspondência com todas as distorções presentes.

O algoritmo Asift combina a técnica de simulação de pontos de vista com o algoritmo Sift. O algoritmo Sift é invariante a 4 parâmetros da transformada afim (2 parâmetros da translação, 1 da rotação e 1 de escala). Enquanto isso, a técnica de simulação de pontos de vista é invariante a 2 parâmetros (inclinação e rotação da câmera). Juntando estas duas técnicas, Asift consegue a invariância aos 6 parâmetros de uma transformada afim.

Um exemplo da técnica de simulação de pontos de vista é apresentado na Fig. 3-13. Neste exemplo, foram simulados 27 pontos de vista.

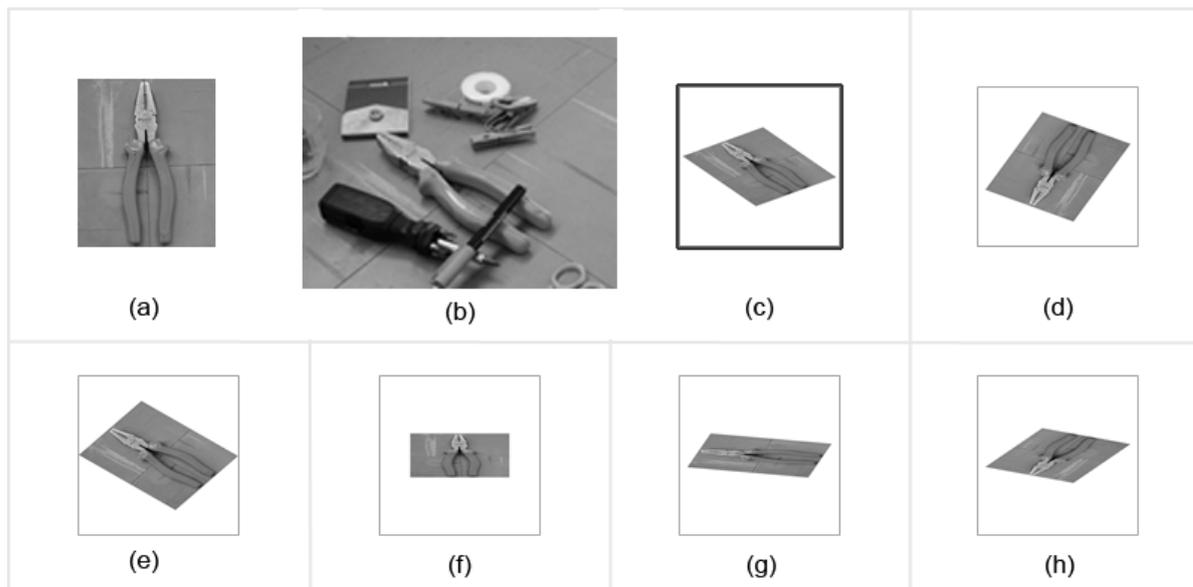


Fig. 3 -13. Exemplo da técnica de simulação de pontos de vista (a) Imagem Q ; (b) Imagem A ; (c) Q_i selecionado como melhor aproximação; (d-h) outros Q_i gerados na simulação.

A técnica de simulação de pontos de vista pode ser integrada a outras técnicas para estender suas áreas de aplicação: (a) Em nosso caso, os algoritmos Aciratefi e Aforapro adicionam a invariância afim aos algoritmos Ciratefi e Forapro que são invariantes por rotação, escala e translação; (b) Zuou, et. al. (2012) e Pang, et. al. (2012) integram a técnica de simulação de pontos de vista com o Surf para ganhar velocidade de processamento, já que Surf é mais rápido que Sift; (c) Gao, et al. (2010) combinam a simulação de pontos de vista com uma técnica de mineração de dados para reduzir o número de simulações e aumentar velocidade de processamento; (d) Ma, et al., (2015) utilizam a simulação de pontos de vista para obter invariância afim num aplicativo de telefonia celular.

3.5. Algoritmos Sift e Asift

Nesta seção, descrevemos brevemente os algoritmos Sift e Asift.

3.5.1. Algoritmo Sift

Sift (Scale Invariant feature transform) apresentado por Lowe (1999; 2004) é um algoritmo para o casamento de pontos-chaves. Sift pode ser sumarizado em quatro estágios: (a) criação do espaço de escala; (b) detecção de pontos-chaves no espaço de escala; (c) seleção da direção-principal ou direção do gradiente para cada ponto-chave; (d) construção dos vetores descritores (REY-OTERO; DELBRACIO, 2014).

- Criação do espaço de escala: O espaço de escala de uma imagem 2D $I(x, y)$ é uma imagem 3D $L(x, y, \sigma)$, adicionando o fator de escala σ :

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$$

Eq.3. 4

onde a função Gaussiana $G(x, y, \sigma)$ é usada como núcleo da convolução:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

Eq.3. 5

- Detecção de pontos-chaves no espaço de escala: os pontos-chaves são detectados procurando extremos locais na imagem 3D criada pela diferença de Gaussianas. As diferenças de Gaussianas são obtidas pela subtração de duas imagens de escalas próximas no espaço de escala. Os extremos locais determinam a localização e escala dos pontos-chaves. Ao mesmo tempo, Sift exclui extremos locais com baixo contraste ou instáveis, aumentando a estabilidade dos casamentos e melhorando a imunidade ao ruído.

- Seleção da direção do gradiente: Sift atribui uma orientação para cada ponto-chave, utilizando histograma de gradiente-orientado da região ao seu redor.

- Construção dos vetores descritores: Sift utiliza o histograma de gradiente-orientado em 4x4 sub-regiões em torno do ponto-chave. Calcula o histograma da direção do gradiente de oito direções em cada sub-região. Obtém assim vetores descritores com dimensões $4 \times 4 \times 8 = 128$. Finalmente, Sift normaliza o comprimento do vetor para eliminar o impacto das mudanças de iluminação.

Sift pode ser visto como um caso especial de uma classe geral de descritores de imagens baseados em histogramas de campo receptivo (receptive field histograms) (LINDBERG, 2013; 2015). Entre os pioneiros nesta área destacam-se Swain e Ballard (1991). Eles construíram um descritor de imagens baseado em histogramas, para demonstrar que o processo do reconhecimento de objetos poderia realizar-se por comparação dos histogramas dos objetos de uma imagem. Shiele e Crowley (2000) estenderam esta ideia, para histogramas de primeira-ordem de derivadas parciais ou combinações de magnitude de gradiente, e histogramas de cálculo da resposta Laplaciana nas múltiplas escalas. Linde e Lindeberg (2004; 2013; 2015)

generalizaram estas propostas introduzindo uma eficiente forma de aproveitamento dos histogramas de alta-dimensionalidade, e um conjunto de histogramas de marcas complexas que melhoram o desempenho dos histogramas de campo receptivo. Seguindo esta linha, surgem várias propostas destacando entre elas: Sift (LOWE, 2004; 1999), Surf (Speeded up robust features) (BAY, et al., 2006; 2008), GLOH (Gradient location and orientation histogram) (MIKOLAJCZYK; SCHMID, 2005), HOG (Histograms of oriented gradients) (DALAL; TRIGGS, 2005).

Numa avaliação experimental da robustez a deformações de imagens, incluindo diferentes descritores, realizada por Mikolajczyk e Schmid (2005), o descritor Sift foi o mais robusto quando comparado a outros algoritmos baseados em: momentos invariantes, filtragem complexa e correlação de diferentes tipos de pontos-característicos. O grande sucesso alcançado pelo Sift pode ser evidenciado pelas aplicações nas diferentes áreas, por exemplo: na detecção de pontos para imagens SAR (synthetic aperture radar) (DELLINGER, et al., 2015; SCHWIND, et al., 2010); detecção de objetos em aeroportos (TAO, et al., 2011); registro de imagens de sensoriamento remoto (LI, et al., 2009; HUO, et al., 2012; FAN, et al., 2013); reconstrução de imagens (YANG; NEWSAM, 2013); reconhecimento de objetos (JIANG; YU, 2009). Um estudo do Sift e alguns dos algoritmos que projetam suas variantes é apresentado por Wu, et al. (2013).

3.5.2. Algoritmo Asift

A proposta de Morel e Yu (2009) denominada Asift (Affine-Sift) utiliza o algoritmo Sift (LOWE, 2004) para procurar casamentos em dois conjuntos de imagens simuladas, alcançando a invariância a mudanças de pontos de vista. Asift constrói dois conjuntos de imagens simuladas, aplicando transformações afins sobre as imagens I_1 e I_2 . Estas transformações são aplicadas usando os parâmetros de inclinação (tilt) e rotação da imagem. Cada valor do tilt é combinado com uma série de rotações, para garantir a invariância às mudanças de pontos de vista. Esta técnica se denomina simulação de pontos de vista e está descrita no item 3.4.2.

Uma vez que as imagens são transformadas, estas são usadas pelo Sift para gerar os pontos-chaves. Os pontos-chaves gerados pelas imagens transformadas de I_1 são procurados no conjunto de pontos-chaves das imagens transformadas de I_2 .

Asift apresenta superioridade na invariância a pontos de vista quando comparado para outros métodos (YU; MOREL, 2011; MOREL; YU, 2009).

Algumas propostas derivadas de Asift são: (a) PODBREZNIK e POTOČNIK (2010; 2013), no seu trabalho denominado de Asift-sh (*affine scale invariant feature transform – segmentation and local homography*), utilizam o Asift para seguir objetos no monitoramento, aplicado a sistemas de câmeras móveis. (b) NOURY, et. al. (2010) apresentam uma técnica para tornar o Asift invariante a padrões repetitivos presentes em imagens de prédios (*man-made*). (c) LIU, et. al. (2013) exploram Asift, realizando duas variações no Sift de LOWE (2004). Para extrair características locais utilizam uma janela elíptica, em vez da janela circular original, e eles utilizam o segundo momento da matriz hessiana para conseguir que a região elíptica extraída seja invariante afim, como no trabalho de LINDBERG e GARDING (1997). Com isso, conseguem melhorar a invariância afim e construir uma ferramenta útil no registro de imagens de SAR (*synthetic aperture radar*).

Algumas das aplicações que usam Asift são: (a) registro de imagens de sensoriamento remoto (CAO, et al., 2013); (b) detecção e reconhecimento de objetos (REZA, 2012); (c) sistema de serviços de localização espacial (HSIANG, et al., 2013); (d) aplicativo de telefonia móvel para identificação de objetos (YANG; WANG, 2013).

3.6. Invariância a escala

A importância de analisar imagens em várias escalas procede da própria natureza das imagens. Cenas do mundo real são formadas por objetos de múltiplos tamanhos, e estes objetos por sua vez são estruturados por partes em diversos tamanhos. Adicionalmente, a distância entre observador/objeto altera o tamanho de objetos na imagem. Como resultado, as análises realizadas numa única escala podem perder informação. Uma solução é analisar em todas as escalas (ADELSON, et al., 1984; LINDBERG, 2015).

A teoria dos espaços de escala aparece pela necessidade de incorporar a noção de escala de uma forma explícita na representação da imagem, gerando uma família de imagens derivadas (BRADSKI; KAEHLER, 2008). A ideia principal é aumentar a dimensão da imagem $I(x,y)$ para aproveitar as estruturas através de escalas distintas. Assim, o espaço de escala de uma imagem bidimensional é um espaço de

escala tridimensional onde a terceira dimensão é a escala $I(x, y, s)$ (BRAHMBHATT, 2014). Os espaços de escala fornecem um meio de organizar, comparar e analisar todas as escalas de um objeto numa única estrutura. A passagem de uma escala fina para uma grossa simplifica o processo de análise, porque elimina informação desnecessária (LINDEBERG; BRETZNER, 2003). A função mais utilizada para representar a imagem em diferentes escalas é a convolução com o núcleo Gaussiano $G(x, y, s)$. A reprodução de uma imagem $L(x, y, s)$ numa escala específica s pode ser calculada pela convolução da imagem original $I(x, y)$ com o núcleo $G(x, y, s)$, como na Eq.3.4 e Eq.3.5 (RASMUSSEN; WILLIAMS, 2006). Na Fig.3-14 apresentamos um exemplo de espaço de escala utilizando um filtro gaussiano.

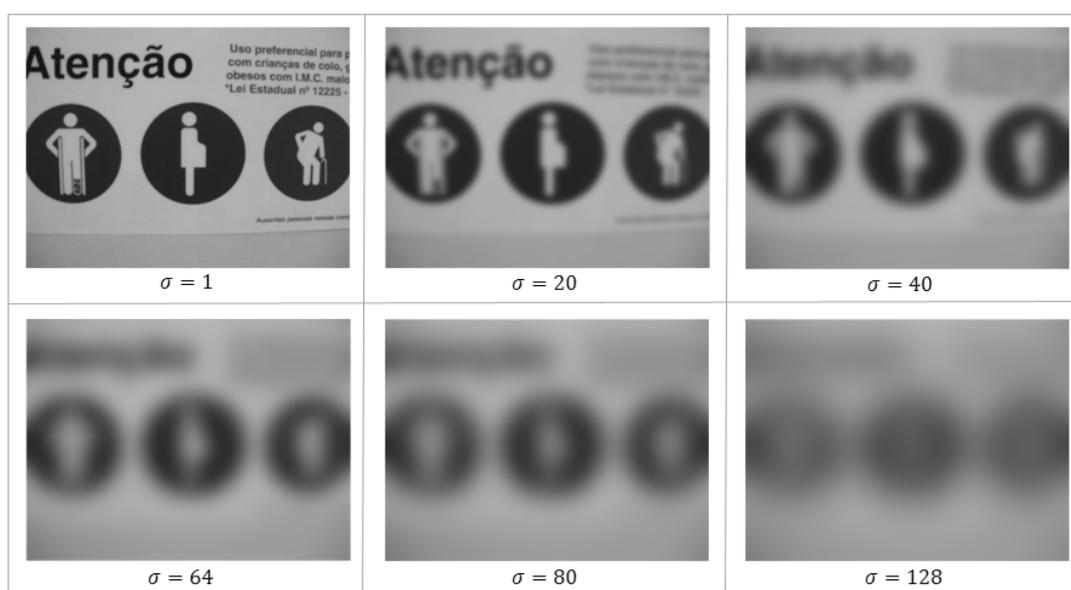


Fig. 3-14. Exemplo de um espaço de escala usando núcleo Gaussiano. Note que a simplificação da imagem é proporcional ao parâmetro σ .

Algumas técnicas propostas para explorar o espaço de escala e extrair a informação são: (a) Mikolajczyk e Schmid (2002), e Baumberg (2000) nestas propostas utiliza-se o detector de Harris em diferentes níveis do espaço de escala discretizado, para detectar pontos-chaves; (b) em Lowe (2004) são procurados máximos e mínimos locais num espaço de escala construído com diferenças de gaussianas usando repetidamente um núcleo gaussiano de tamanho fixo; (c) em Lindeberg e Bretzner (2003) é proposto um método para seleção automática de escalas, baseado na detecção de extremos sobre escalas em combinações não lineares de derivadas normalizadas. Seu espaço de escala é construído por

laplaciano de Gaussiana utilizando núcleos de diferentes tamanhos. Estas duas últimas técnicas são muito semelhantes, mas as diferenças de gaussianas podem ser calculadas muito mais rapidamente devido à separabilidade de núcleo Gaussiano.

Outras técnicas que procuram a invariância à escala, mas sem utilizar o espaço de escala, são: (a) Klinger (1971) utiliza uma pirâmide multi-resolução, organizada de forma que o número de pixels decresce por um fator constante de um andar para outro da pirâmide. Alguns trabalhos relevantes se fundamentam nesta técnica, como é o caso das transformadas Wavelets (MALLAT, 1989). A principal diferença entre multi-escala e multi-resolução é a variação sistemática do número de pixels. Na representação multi-resolução, o número de pixels varia e a resolução da imagem decresce gradualmente, por um fator que normalmente é dois ou quatro (RASMUSSEN; WILLIAMS, 2006). Na representação multi-escala o número de pixels é mantido constante ao longo das escalas. Há alguns métodos relevantes que adotam a técnica Wavelets, por exemplo; o algoritmo Surf (*Speeded up robust features*) (BAY, et. al. 2008) que utiliza a Haar Wavelet, e o método proposto por Ju e Kai-Kuang (2007) que utilizam a Gabor Wavelets.

No caso dos algoritmos Forapro e Ciratefi, o espaço de escala é construído a partir da imagem Q , redimensionando-o por um conjunto de fatores de escala, obtendo as imagens redimensionadas Q_1, Q_2, \dots, Q_t . O casamento é realizado procurando cada imagem Q_i na imagem A .

3.7. Robustez a padrões repetitivos

Os padrões repetitivos constituem um problema no casamento de imagens, porque sua presença normalmente aumenta os falsos casamentos.

As ideias básicas das propostas pioneiras, que evoluíram as pesquisas na problemática de padrões repetitivos, fundamentavam-se em dois estágios. Primeiro, detectar um conjunto de características de baixo-nível: são características básicas que podem ser extraídas automaticamente sem precisar informação alguma sobre seu relacionamento espacial (e.g. eixos, linhas, contornos, etc). Segundo, agrupar os conjuntos utilizando uma transformação em perspectiva geométrica ou deformação global. Algumas destas propostas são: (a) uma janela de casamento

móvel baseada em crescimento de regiões (LEUNG; MALIK, 1996); (b) agrupamento de padrões repetidos baseada na detecção de homografias (SCHAFFALITZKY; ZISSERMAN, 1999); (c) análise de texturas utilizando adição de estruturas por similaridade (HAYS, et al., 2006); (d) detecção de padrões por estruturas baseada em crescimento de regiões (LIN; LIU, 2007); (e) classificação de texturas próximas utilizando modelos gráficos (PARK, et al., 2009). No entanto, estas propostas estão condicionadas a dois fatores: o nível de distorção geométrica dos padrões que dificulta a representação das características e ao fato de que as estruturas de padrões globais nem sempre possibilitam o agrupamento.

Outras técnicas que procuram a robustez a padrões repetitivos são: (a) Shechtman e Irani (2007) apresentam uma proposta para medição e casamento de regiões similares baseado em correlação. Eles correlacionam um pixel centrado no ponto de interesse com uma grande região ao seu redor, e usam os valores de máxima correlação com um log-polar binário como descritor. O ponto negativo desta proposta é o fato de usar um limitado descritor binário; (b) Tell e Carlsson (2002) apresentam uma proposta para casamento de imagens baseado num filtro de janela móvel que se desloca ao longo duma linha ou perfil de intensidade. A ideia básica é extrair perfis de intensidade entre pontos característicos e guardar a informação numa matriz de acumulação. Os casamentos com mais votos na matriz de acumulação são selecionados e propostos no final do processo. (c) Pobeznik e Potocnik (2010) propõem uma combinação do ASIFT com uma etapa de segmentação para extrair regiões invariantes afins que apresentam robustez a padrões repetitivos visando a implementação de um algoritmo para reconhecer objetos em imagens de estruturas e construções cíveis (e.g. janelas, paredes e fachadas).

No Aciratefi e Aforapro, a robustez a padrões repetitivos é conseguida por dois fatores: (a) a região de casamento nos dois algoritmos é relativamente grande, este fato aumenta a robustez a padrões repetitivos porque os padrões ficam imersos na região de casamento; (b) a utilização da transformada de Hough para filtrar casamentos falsos e agrupar as estruturas de submodelos.

Nem sempre os padrões repetitivos representam problemas, alguns trabalhos os utilizam como uma ferramenta para facilitar a detecção e construção de descritores, por exemplo, Shechtman e Irani (2007) apresentam uma proposta para o casamento de dados visualmente complexos. Eles correlacionam uma área central com o ponto

de interesse que envolve uma ampla região, e usam valores de máxima correlação centrados numa função de logaritmo-polar binária como descritores. Demonstrando que a forma é um fator importante no casamento de padrões.

3.8. Invariância a mudanças de brilho/contraste

A mudança de brilho/contraste é um dos fatores mais comuns nas cenas do mundo real. Representa um problema no casamento de imagens porque diminui o número de casamentos verdadeiros. Duas regiões que diferem apenas por mudança de brilho/contraste devem ser consideradas regiões iguais sob diferentes condições de iluminação.

Alguns métodos que procuram a invariância a brilho/contraste são: (a) Sift, atuando numa região de gradiente local, e Surf integrando os valores do pixel sobre uma região local, reforçam sua técnica utilizando um pré-processamento, que consiste na utilização de um filtro Gaussiano para estabilizar os níveis altos de intensidade; (b) Ouyang, et al. (2014) apresentam uma proposta para procurar invariância a mudanças extremas de iluminação, para pontos característicos no espaço afim. Esta proposta está baseada na construção de um histograma de padrões binários; (c) Ramesh e Priya (2013) propõem um algoritmo que procura a invariância a pontos de vista e mudanças de iluminação. Eles utilizam numa janela móvel que estima a posição relativa e a relação de iluminação entre as regiões características. A informação é extraída utilizando um detector de características locais. Os autores estimam que este método melhora Asift, aumentando o número de casamentos verdadeiros em condições de mudanças extremas de iluminação. No entanto, os autores comentam que, este método pode falhar quando a estimação inicial falha; (d) Ethan, et al. (2011) na sua proposta denominada ORB utilizam uma correção básica da iluminação e uma filtragem por um limiar calculado. A correção básica da iluminação e a filtragem podem ser práticas na melhora prévia da imagem para calcular a gradiente. Mas em geral, a literatura não reporta resultados sobre algum método desta classe em específico (SCOTT KRIG, 2014).

Aciratefi e Aforapro conseguem a invariância a brilho/contraste devido a dois fatores: (a) a invariância ao brilho é obtida pela subtração do nível de cinza médio; (b) a invariância ao contraste é obtida usando correlação cruzada normalizada ou calculando o versor do vetor de características.

4. ABORDAGEM DA PROPOSTA

Este trabalho está dividido em duas partes: na primeira, propomos o algoritmo Aciratefi como um algoritmo de casamento de modelos invariantes a mudanças de ponto de vista; na segunda, propomos uma etapa de filtragem de casamentos de pontos-chaves e aplicamos esta filtragem no algoritmo Sift, com o objetivo de diminuir a sua taxa de erros.

4.1. Aciratefi

Nesta proposta, nosso desafio é construir um algoritmo de casamento de modelos invariante a pontos de vista com robustez a padrões repetitivos e invariante a mudanças de brilho/contraste. Com este propósito combinamos duas técnicas: (a) O algoritmo Ciratefi (*Circular and radial template matching filter*), proposto por KIM e ARAUJO (2007), e ARAUJO e KIM (2011); (b) A simulação de pontos de vista, proposta por MOREL e YU (2009).

A técnica de simulação de pontos de vista, descrita no Capítulo 3, foi apresentada como complemento do Sift (LOWE, 2004). Da mesma forma, esta técnica pode ser utilizada em combinação com outras técnicas de casamento de imagens, procurando superar limitações da versão original.

As diferenças entre Ciratefi e Sift são apresentadas na Tab.4-1.

Tab.4- 1. Diferenças entre Ciratefi e Sift.

Característica	Sift	Ciratefi
Especialidade	Casamento de pontos-chaves (keypoint matching)	Casamento de modelos (template matching)
Invariância a escala	Ótimo	Bom
Invariância a mudanças de brilho/contraste	Apresenta dificuldades	Ótimo
Invariância a padrões repetitivos	Apresenta dificuldades	Ótimo
Robustez a oclusões parciais	Ótimo	Apresenta dificuldades
Região de casamento	Pontos-chaves	Modelos (grande conjunto pontos)

- Especialidade: Sift foi projetado para o casamento de pontos-chaves, e não procura modelos. Para Sift procurar um modelo Q numa imagem A , precisa adicionalmente da transformada de Hough (BALLARD, 1981). A transformada generalizada de Hough permite converter casamento de pontos-chaves em casamento de modelos (abordado no Cap.2).

- Invariância à escala: Sift é reconhecido como o melhor algoritmo invariante à escala (HASSNET, et. al., 2012; MOREL; YU, 2010; MIKOLAJCZYK; SCHMID, 2005). Ciratefi é invariante à escala no intervalo restrito $[0.5, \infty]$ (ARAUJO; KIM, 2011).

- Invariância a mudanças de brilho/contraste: Experimentos mostram que Sift não é completamente invariante a mudanças de brilho/contraste. Ciratefi apresenta invariância a mudanças de brilho/contraste porque o seu processo de comparação de característica é feita usando coeficiente de correlação, que é intrinsecamente invariante a mudanças de brilho/contraste. A estratégia de Ciratefi envolve mais pixels da vizinhança e calcula valor médio. Isto faz Ciratefi mais confiável que Sift quando as imagens apresentam: borrão de foco, grandes áreas com níveis de cinza constantes, efeitos de compressão JPEG, pouca textura, ou mudanças extremas de brilho/contraste (ARAUJO; KIM, 2011). Em teoria, qualquer variação de iluminação uniforme poderia ser compensada pela subtração do valor médio seguida de correlação cruzada normalizada. Somente as variações parciais ou não uniformes de iluminação afetariam o processo. Entende-se por variações parciais as mudanças de iluminação que atingem somente uma parte da região dos pixels envolvidos.

- Região de casamento: A região de casamento do Sift é uma pequena área em torno do ponto-chave (keypoint). A região de casamento do Ciratefi é a imagem Q inteira. Projeções circulares e radiais são traçadas sobre a extensão da imagem Q para extrair as características.

- Invariância a padrões repetitivos: o padrão é uma subparte da imagem, menor do que o modelo Q (ver detalhes no item 2.4.3). Ciratefi apresenta robustez a padrões repetitivos, porque um padrão qualquer fica imerso na sua região de casamento. Enquanto isso, um ponto-chave do Sift não considera as regiões vizinhas ao padrão.

- Robustez a oclusões parciais: Sift complementado com a transformada de Hough é robusto a oclusões parciais, já que não precisa da imagem do modelo toda

para achar o casamento. Ou seja, em teoria Sift só precisa de pelo menos três pontos-chaves visíveis (não oclusos) para achar um modelo. Ciratefi precisa que toda a instância do modelo esteja visível para conseguir o casamento. Neste item o menor tamanho da região de casamento da vantagem a Sift sobre Ciratefi.

4.1.1. Aciratefi invariante a pontos de vista

Aciratefi obtém esta característica pela técnica de simulação de pontos de vista (MOREL e YU, 2009): (a) A imagem modelo Q é transformada gerando as imagens $\{Q_1, Q_2, \dots, Q_n\}$ que simulam todas as possíveis distorções lineares causadas pelas mudanças de orientação dos eixos da câmera. Os parâmetros usados são $t = \sqrt{2}^m$ como valores de $m \in \{0, 1, \dots, 5\}$ obtendo assim o intervalo $t \in \{\sqrt{2}^0, \sqrt{2}^1, \sqrt{2}^2, \sqrt{2}^3, \sqrt{2}^4, \sqrt{2}^5\}$ para atingir valores de inclinações $\theta \approx 0^\circ, 45^\circ, 60^\circ, 69^\circ, 75^\circ, 80^\circ$ consecutivamente, e rotações $\phi \in [0, \pi)$, como sugerido pelos autores (ver detalhes no item 3.4.2). (b) Utiliza-se Ciratefi para procurar cada simulação Q_i na imagem A . As Fig.4-1, Fig.4-2 e Fig.4-3 mostram um exemplo do casamento de duas imagens que apresentam mudanças de pontos de vista.

A Fig.4-1 apresenta as imagens Q e A que serão utilizadas no processo de casamento, o objetivo é achar na imagem A uma instância do modelo Q .



Fig. 4-1. Imagens que apresentam mudanças de ponto de vista. (a) imagem modelo Q , tamanho real 100x100 pixels. (b) imagem de análises A , tamanho real 500x375 pixels. Fonte da imagem Morel e Yu (2014).

Note que a imagem modelo Q apresenta um ponto de vista frontal, e que o ponto de vista da imagem A , com relação à Q , tem uma inclinação $\theta \approx 80^\circ$. Neste caso o

parâmetro de inclinação utilizado para as simulações será $m = 5 \rightarrow t = \sqrt{2}^5$. A Fig.4-2 apresenta a serie de imagens (Q_i) simuladas.

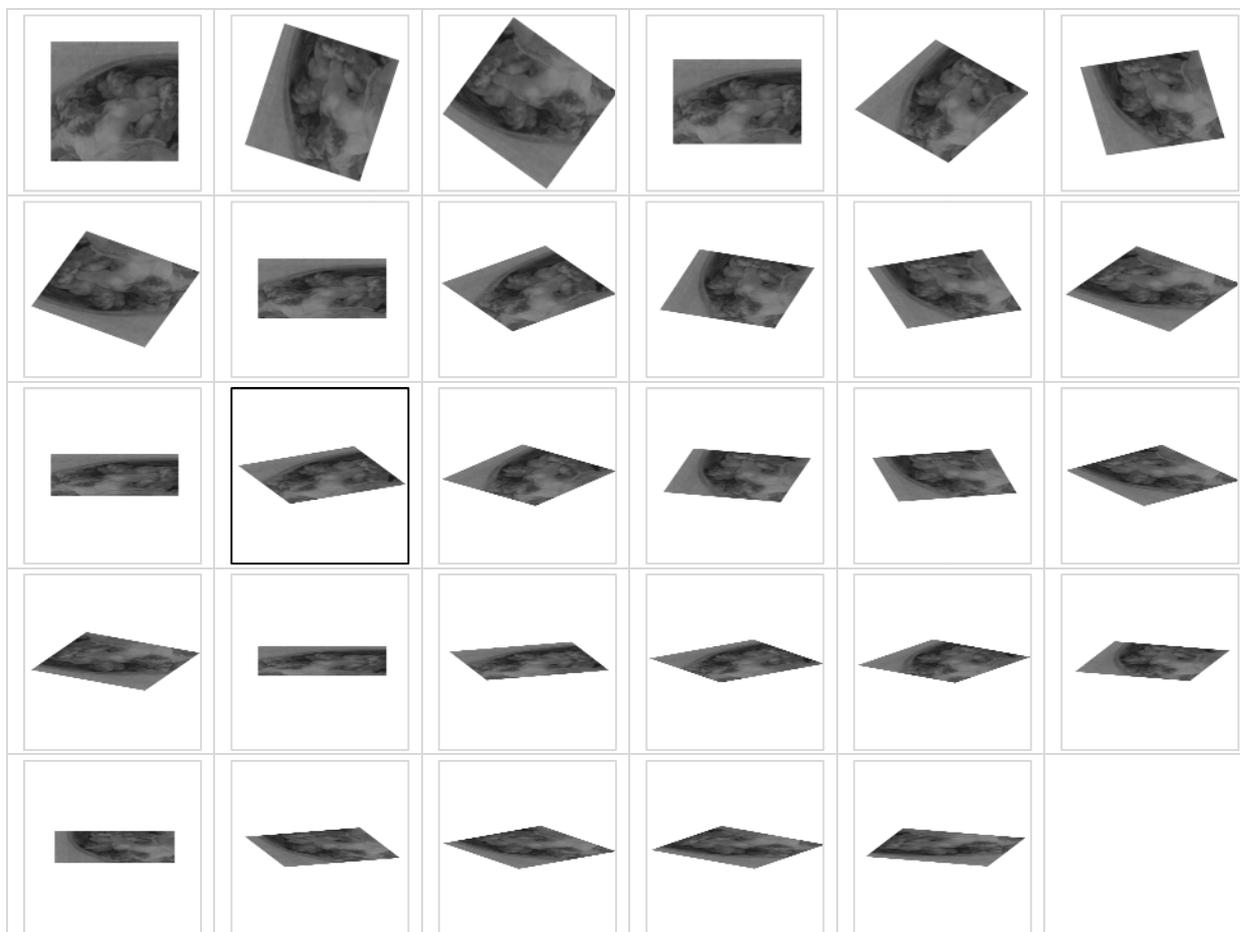


Fig. 4-2. Imagens simuladas por transformações afins.

A imagem modelo Q é distorcida tentando simular a transformação sofrida pela instância presente na imagem A .

Cada imagem Q_i será procurada na imagem A para achar a melhor correspondência. Este processo é realizado pelo Ciratefi, utilizando suas características invariantes por escala, rotação e brilho/contraste. Neste exemplo, Ciratefi selecionou a imagem com a borda preta (Fig.4-2). Na Fig.4-3 são apresentadas as imagens resultantes do casamento.



Fig. 4-3. Casamento de modelos invariante a transformações afins. (a) Imagem simulada Q_i , selecionada por Ciratefi. (b) Resultado do casamento, note que a linha verde no círculo indica a diferença de rotação entre as imagens.

No exemplo anterior, Aciratefi acertou corretamente a instância do modelo procurado, demonstrando sua invariância a mudanças no ponto de vista.

4.1.2. Aciratefi invariante a padrões repetitivos e mudanças de brilho/contraste

Aciratefi obtém estas características herdadas do algoritmo Ciratefi (KIM; ARAUJO, 2007; ARAUJO, 2009). Ciratefi utiliza o coeficiente de correlação para ter invariância a mudanças de brilho/contraste. Ciratefi possibilita o casamento de instâncias negativas (com contraste invertido) pelo uso do valor absoluto do coeficiente de correlação. Na Fig. 4-4 apresentamos um exemplo do Aciratefi invariante a mudanças de brilho/contraste.

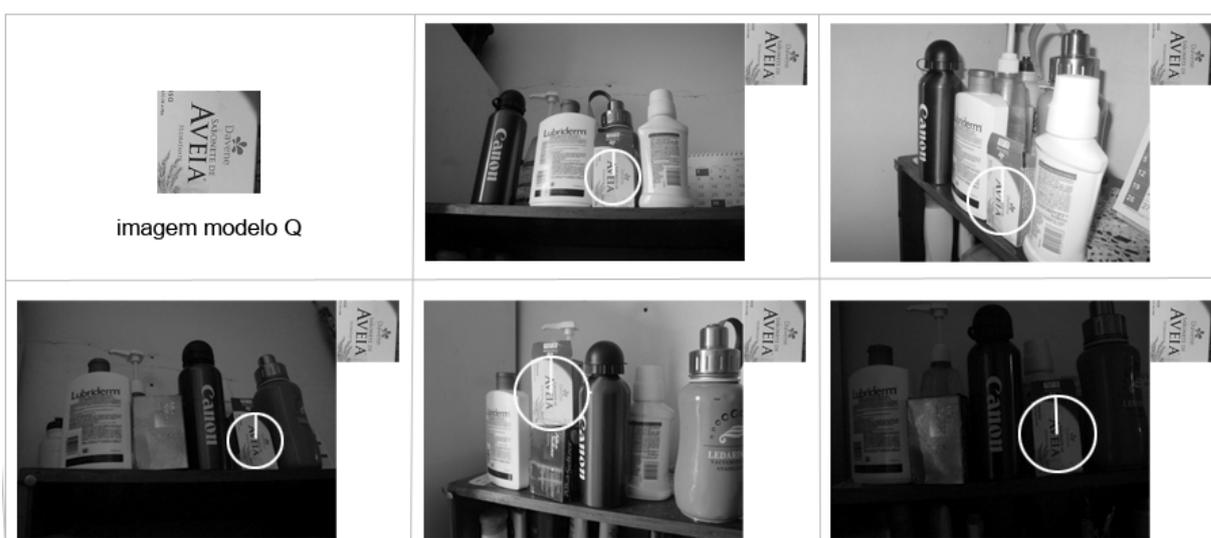


Fig. 4-4. Aciratefi invariante a mudanças de brilho/contraste. O tamanho da imagem Q foi alterado desproporcionalmente para esta ilustração. O tamanho real das imagens é 80x80 e 500x375 pixels.

A estratégia de Aciratefi na extração de características, utilizando projeções radiais e circulares, reforça a invariância a brilho/contraste e o torna invariante a padrões repetitivos. Aciratefi procura cada modelo simulado Q_i de forma integral na imagem A , o que o torna robusto a padrões repetitivos. Na Fig. 4-5 apresentamos um exemplo de robustez do Aciratefi a padrões repetitivos.

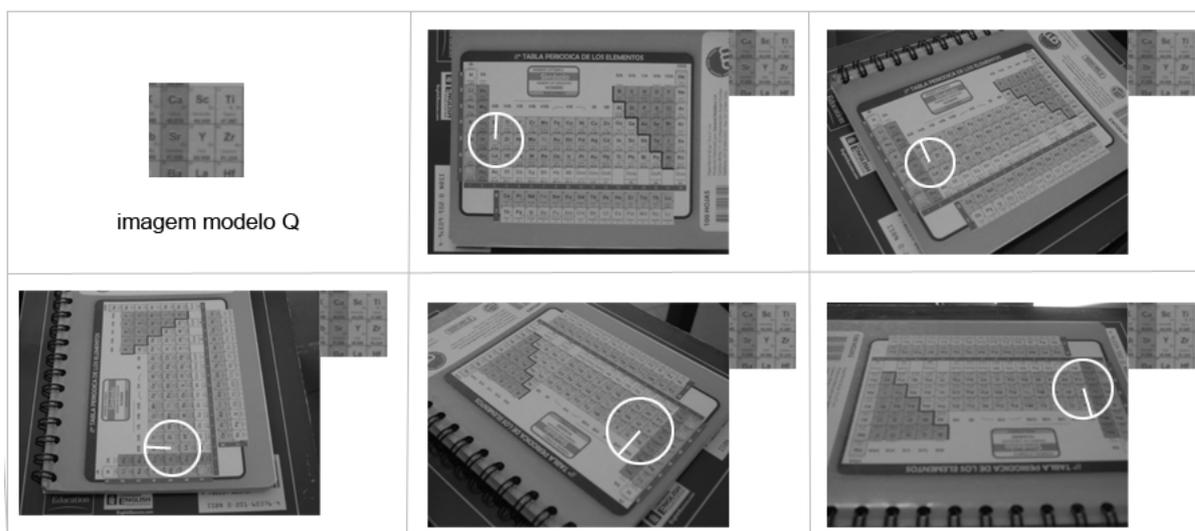


Fig. 4-5. Aciratefi invariante a padrões que se repetem.

Note que o tamanho da imagem Q foi alterado desproporcionalmente para esta ilustração. O tamanho real das imagens é 80x80 e 500x375 pixels.

4.1.3. Funcionamento do Aciratefi

Dadas duas imagens Q e A , assumiremos que a imagem Q apresenta a sua vista frontal ($\lambda = 1$ e $\psi = \phi = \theta = 0$) e a imagem A tem uma instância de Q fotografada desde um ponto de vista aleatório. Diferentemente do algoritmo Asift, não consideramos a “inclinação de transição” onde as duas imagens (Q e A) podem adotar diferentes inclinações. Em outras palavras, no nosso caso a imagem Q deve estar fotografada de frente, enquanto que na “inclinação de transição” a imagem Q pode estar fotografada desde um ponto de vista qualquer. No casamento de modelos, esta imposição não é muito restritiva, porque normalmente o usuário tem controle sobre a aquisição da imagem Q . A descrição do Aciratefi é apresentada no algoritmo I.

ALGORITMO I: ACIRATEFI

Entrada: imagens Q e A , e m (valor que determina o número de imagens simuladas).

Saída: Melhor casamento Q_i , junto com sua posição, ângulo, escala e parâmetros afins na imagem A .

1. Usar transformações afins para simular as vistas de Q desde diferentes pontos de vista, gerando $\{Q_1, Q_2, \dots, Q_i, \dots, Q_n\}$.
 2. Procurar cada imagem simulada Q_i em A usando Ciratefi.
 - (a) Calcular os pixels candidatos de primeiro grau (G_1) com a sua provável escala (s_i) usando projeções circulares.
 - (b) Promover alguns pixels candidatos de primeiro grau para segundo grau (G_2) com sua provável escala (s_i) e seu provável ângulo de rotação (ϕ_i) usando projeções radiais.
 - (c) Executar o casamento de modelos (template matching) utilizando somente os pixels candidatos de segundo grau G_2 na sua provável escala e provável rotação.
 3. Selecionar Q_i com o maior coeficiente de correlação normalizado como o casamento final entre as imagens Q e A .
-

O número de simulações deve englobar todos os possíveis pontos de vista que a câmera possa adotar. O número de simulações está relacionado com as variações dos parâmetros de rotação e inclinação dos eixos da câmera. Adotando as indicações de Morel e Yu (2009) os valores adotados são $\Delta t = t_{k+1}/t_k = \sqrt{2}$ e $\Delta\theta = 72^\circ/t$.

Assim, com $m = 5 \rightarrow t = \sqrt{2}^5$ são geradas 29 imagens simuladas (Q_i), alcançando inclinações até $\theta \approx 80^\circ$, como no exemplo apresentado nas figuras Fig. 4-1, Fig. 4-2 e Fig. 4-3. O valor de m pode ser adaptado para outros valores, por exemplo, se $m = 6 \rightarrow t = \sqrt{2}^6$ que alcança inclinações até $\theta \approx 83^\circ$ e o número de imagens simuladas incrementa para $n = 41$.

4.2. Melhoria do casamento de pontos-chaves pela coerência geométrica

No casamento de pontos-chaves, frequentemente acontecem casamentos falsos. Isto é, os pontos-chaves da imagem I_1 podem ser associados a pontos-chaves não correspondentes na imagem I_2 . O objetivo desta proposta é diminuir o número de casamentos falsos de Sift, baseada num conceito que denominamos de coerência geométrica. A mesma ideia pode ser utilizada em outros métodos de casamentos de pontos-chaves, como Surf.

Lowe (2004) sugere no seu trabalho sobre Sift o seguinte método para fazer casamentos entre pontos-chaves: para cada ponto chave A_1 da imagem I_1 , procuram-se dois pontos-chaves A_{2p} (primeiro ponto) e A_{2s} (segundo ponto) da imagem I_2 que apresentem as características mais similares às de A_1 . A semelhança entre as características de dois pontos-chaves é calculada usando a distância Euclidiana. Depois, calcula-se a razão (R) entre as distâncias:

$$R(A_1, A_{2p}) = d(A_1, A_{2p}) / d(A_1, A_{2s})$$

Eq.4.1

onde $d(A_1, A_{2p})$ é a distância Euclidiana entre as características dos pontos A_1 e A_{2p} . Se $R(A_1, A_{2p}) < \text{limiar}$, então (A_1, A_{2p}) é aceito como um possível casamento-verdadeiro. Lowe sugere usar $\text{limiar} = 0,8$. Evidentemente, este valor pode ser alterado para otimizar o número de falsos positivos ou falsos negativos. Resumindo, Lowe ordena os possíveis casamentos (A_1, A_{2p}) pela razão $R(A_1, A_{2p})$. Quanto menor esta razão, maior a chance do casamento ser verdadeiro.

Nosso desafio foi obter outras ordenações dos casamentos que, para um limiar dado, aumente a taxa de casamentos verdadeiros.

4.2.1. Coerência geométrica entre dois casamentos

Cada ponto-chave (K) do descritor Sift consiste do conjunto de componentes $[K_x, K_y, K_r, K_s]$, onde (x, y) representam as coordenadas do ponto, r representa o ângulo de rotação, e s representa a escala. Além disso, o ponto-chave K tem (normalmente) 128 características derivadas de histograma de gradiente orientado da região em torno do ponto-chave.

Vamos supor que um objeto O tenha dois pontos-chaves A e B . O objeto O aparece nas imagens I_1 e I_2 em diferentes posições, ângulos e escalas, como representado na Fig.4-6.

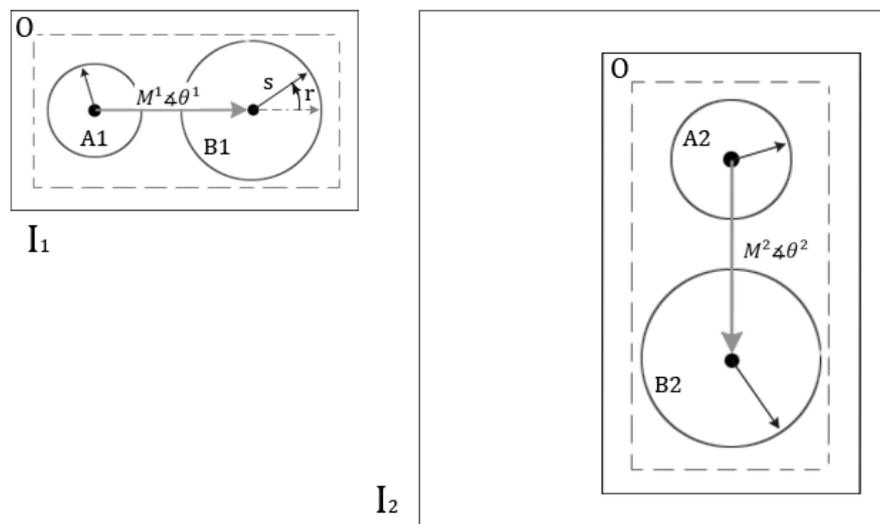


Fig. 4-6. Representação de dois pares de pontos-chaves. I_1 e I_2 são as duas imagens. A^* e B^* são os pontos-chaves. M^* é a magnitude e θ^* é o ângulo do vetor-diferença.

Vamos escrever explicitamente os parâmetros dos quatro pontos-chaves:

$$\begin{aligned} A^1 &= [A_x^1, A_y^1, A_r^1, A_s^1] & B^1 &= [B_x^1, B_y^1, B_r^1, B_s^1] & \text{na imagem } I_1 \\ A^2 &= [A_x^2, A_y^2, A_r^2, A_s^2] & B^2 &= [B_x^2, B_y^2, B_r^2, B_s^2] & \text{na imagem } I_2 \end{aligned}$$

Eq.4. 2

Vamos supor que um casamento A foi estabelecido entre pontos-chaves A^1 e A^2 , e outro casamento B entre pontos-chaves B^1 e B^2 . Diremos que dois casamentos A e B possuem coerência geométrica ou simplesmente são coerentes, se sua relação satisfaz as quatro igualdades seguintes:

- (a) A relação entre as escalas dos pontos-chaves A e B nas duas imagens deve ser similar.

$$A_s^2/A_s^1 = B_s^2/B_s^1 = s_r$$

Eq.4. 3

- (b) A diferença entre os ângulos de rotação dos pontos-chaves A e B deve ser similar.

$$A_r^2 \ominus A_r^1 = B_r^2 \ominus B_r^1 = r_d$$

Eq.4. 4

onde \ominus é subtração entre ângulos, e o resultado deve estar no intervalo $[0, 2\pi]$. Em linguagem C, a função $a \ominus b$ pode ser implementada:

```
double subtraiDeg(double a, double b) {
    double d1=a-b;
    while (d1<0) d1+=360;
    while (d1>360) d1-=360;
    return d1;
}
```

Os vetores-diferenças entre os centros dos dois pontos-chaves podem ser expressas como:

$$\begin{cases} [M^1 \angle \theta^1] = [B_x^1, B_y^1] - [A_x^1, A_y^1] \\ [M^2 \angle \theta^2] = [B_x^2, B_y^2] - [A_x^2, A_y^2] \end{cases}$$

Eq.4. 5

onde M é a magnitude e θ é o ângulo do vetor-diferença. Então as seguintes igualdades também devem ser satisfeitas para que os dois casamentos sejam coerentes:

- (c) A relação entre as magnitudes do vetor-diferença (M) deve satisfazer:

$$M^2/M^1 = s_r$$

Eq.4. 6

onde s_r é a relação de escalas da Eq.4.3.

(d) A relação entre os ângulos do vetor-diferença (θ) deve satisfazer:

$$\theta^2 \ominus \theta^1 = r_d$$

Eq.4. 7

onde r_d é a diferença entre ângulos de rotação da Eq.4.4.

Diremos que dois casamentos possuem escalas coerentes se satisfazem equações (a) e (c) e que possuem ângulos coerentes se satisfazem equações (b) e (d).

4.2.2. Testes para verificar se dois casamentos são coerentes

Na prática, utilizamos os dois testes abaixo para verificar se os casamentos A e B são coerentes ou não:

(a) Teste de escala: São calculados quatro valores abaixo:

$a_s = \frac{A_s^2}{A_s^1}$	$b_s = \frac{B_s^2}{B_s^1}$	$m_s = \frac{M_s^2}{M_s^1}$	$m_q = \sqrt{a_s b_s}$
-----------------------------	-----------------------------	-----------------------------	------------------------

Eq.4. 8

Os dois casamentos A e B possuem escalas coerentes se:

$$div_{maior}(a_s, b_s) > lim_{esc} \quad e \quad div_{maior}(m_q, m_s) > lim_{esc}$$

onde $div_{maior}(a_s, b_s)$ é uma função que divide menor argumento pelo maior. Isto é:

$$div_{maior}(a, b) = \begin{cases} \frac{a}{b} & \text{se } b > a \\ \frac{b}{a} & \text{se } b \leq a \end{cases}$$

$lim_{esc} \in [0, 1]$ é o limiar de tolerância de escala. Quanto maior for este valor, menos tolerante o algoritmo se torna às diferenças de escala. Em nossos testes, utilizamos os valores $lim_{esc} = 0,39$ e $lim_{esc} = 0,50$.

(b) Teste de ângulo de rotação: São calculados quatro valores abaixo:

$a_r = A_r^2 \ominus A_r^1$	$b_r = B_r^2 \ominus B_r^1$	$\theta_r = \theta^2 \ominus \theta^1$	$m_r = \frac{a_r + b_r}{2}$
-----------------------------	-----------------------------	--	-----------------------------

Eq.4. 9

Dois casamentos A e B possuem ângulos de rotação coerentes se:

$$a_r \ominus b_r < \lim_{ang} \quad e \quad \theta_r \ominus m_r < \lim_{ang}$$

onde $a_r \ominus b_r$ é subtração de ângulos definida na seção anterior e \lim_{ang} é o limiar de tolerância do diferença de ângulos. Utilizamos valores $\lim_{ang} = 13.5$ e $\lim_{ang} = 20.0$ graus.

Dois casamentos A e B são coerentes se possuem escalas e ângulos coerentes.

4.2.3. Descarte de casamentos sem coerência

A primeira ideia, para melhorar Sift usando coerência geométrica, foi testar se um casamento A possui ou não pelo menos um outro casamento coerente. Se o casamento A não possuir nenhum casamento coerente, então ele será descartado como um casamento falso. Em outras palavras, casamentos sem nenhum outro casamento coerente são colocados no final da lista ordenada, onde os piores casamentos aparecem no final da lista. Os resultados obtidos com esta primeira ideia foram piores do que a técnica original do Lowe e por isso esta ideia foi descartada.

4.2.4. 'Modo n' de ordenação de casamentos

Nesta segunda ideia, que denominamos de 'modo n', para cada casamento (A^1, A^2) é calculado o número de casamentos coerentes $n_{coe}(A^1, A^2)$. Este número é utilizado para ordenar os casamentos: quanto maior o número $n_{coe}(A^1, A^2)$, melhor será o casamento avaliado e aparecerá mais no começo da lista ordenada. Na etapa final, são selecionados os melhores N casamentos, onde N é o mesmo do número de casamentos aceitos pelo critério de Lowe.

4.2.5. 'Modo f' de ordenação de casamentos

Nesta terceira ideia, usamos a combinação entre o número de casamentos coerentes $n_{coe}(A^1, A^2)$ e a razão $R(A^1, A^2)$ da Eq.4.1. O 'modo f' ordena os casamentos pela função $f(A^1, A^2)$, como na Eq.4.10:

$$f(A^1, A^2) = n_{coe}(A^1, A^2) * (1 - R(A^1, A^2))$$

Eq.4. 10

Note que o valor $(1 - R(A^1, A^2)) \in [0, 1]$ será maior se o casamento tiver melhor qualidade. Na etapa final serão selecionados os melhores N casamentos, onde N é o mesmo do número de casamentos aceitos pelo critério de Lowe.

4.2.6. Teste de filtragem por coerência geométrica

O programa usado para testar a filtragem por coerência geométrica lê as imagens I_1 e I_2 , e a matriz homográfica M_{hom} . A matriz M_{hom} para cada combinação de imagens (I_1 e I_2) é fornecida pela base de imagens de Mikolajczyk (2015) utilizada.

Utilizando Sift, são extraídos os pontos-chaves (key_*) e os descritores (des_*) de cada imagem, Cada ponto-chave da imagem I_1 é comparado com todos os pontos-chaves da imagem I_2 . O objetivo é achar uma combinação com a menor distância Euclidiana $d(des_1, des_2)$ e casar aqueles pontos para construir o vetor de casamentos (Cas_1).

A partir do vetor Cas_1 realizam-se ordenações dos casamentos utilizando as três estratégias descritas nos itens anteriores: Cas_{sift} (a ordenação original do Sift), $Cas_{modo n}$, e $Cas_{modo f}$.

Utiliza-se o vetor Cas_{sift} para calcular o número de casamentos N a serem aceitos, isto é, o número de casamentos que passam pelo limiar sugerido por Lowe ($R(A^1, A^2) < 0,8$). Utilizando o número N , conseguimos selecionar aqueles casamentos aceitos por cada um dos três critérios de ordenação. O programa classifica cada casamento do vetor Cas_1 como verdadeiro ou falso utilizando a M_{hom} . Assim, é possível verificar o número de casamentos verdadeiros entre os casamentos aceitos pelos três critérios de ordenação. Para traçar os gráficos de

taxa de sucesso vs. taxa de aceitação, variamos N de zero até o número total de casamentos. Para cada N , calculamos o número de casamentos corretos.

Sumarizando o processo:

ALGORITMO II: TESTE DE SIFT FILTRADO PELA COERÊNCIA GEOMÉTRICA

Entrada: imagens I_1, I_2 , e M_{hom}

Saída: Taxa de erros e taxa de acertos das ordenações: Sift, ‘modo n’ e ‘modo f’.

- 1) Extrair: $I_1 \rightarrow \begin{cases} key_1 \\ des_1 \end{cases}$ e $I_2 \rightarrow \begin{cases} key_2 \\ des_2 \end{cases}$
 - 2) Construir o vetor de casamento básico:
 Para cada des_{1i} fazer
 Para cada des_{2j} calcular
 Casar (des_{1i}, des_{2j}) tal que $d(des_{1i}, des_{2j})$ seja a menor.
 Construir o vetor de casamento (Cas_1)
 - 3) Ordenar o vetor Cas_1 três vezes usando os critérios de ordenação:

$$\text{Ordenar os casamentos} \begin{cases} Cas_1 \xrightarrow[\text{critério Sift}]{} Cas_{Sift} \\ Cas_1 \xrightarrow[\text{critério modo n}]{} Cas_{modo n} \\ Cas_1 \xrightarrow[\text{critério modo f}]{} Cas_{modo f} \end{cases}$$
 - 4) Calcular o número N de casamentos a serem considerados, isto é, quantidade de casamentos que satisfazem $R < 0,8$.
 - 5) Classificar os vetores com os N melhores casamentos:

$$\text{Classificar os vetores} \begin{cases} Cas_{Sift} \xrightarrow[N \text{ melhores casamentos}]{} NCas_{Sift} \\ Cas_{modo n} \xrightarrow[N \text{ melhores casamentos}]{} NCas_{modo n} \\ Cas_{modo f} \xrightarrow[N \text{ melhores casamentos}]{} NCas_{modo f} \end{cases}$$
 - 6) Verificar cada casamento como verdadeiro/falso usando M_{hom}
 - 7) Calcular a taxa de acertos das três ordenações.
-

5. RESULTADOS

Na execução destes testes foi utilizado um computador com sistema operacional Windows 7 de 32 bits, processador Intel core™ 2 Duo (2,10 Ghz), e 3.00 GB de RAM. Os nossos resultados foram organizados em duas seções: testes do algoritmo Aciratefi e testes da filtragem de casamentos de pontos-chaves.

5.1. Testes de avaliação do algoritmo Aciratefi

O objetivo desta avaliação é analisar as características invariantes do algoritmo Aciratefi. Com este propósito, comparamos Aciratefi com Aforapro e Asift. Três itens foram avaliados: invariância a mudanças de brilho/contraste, invariância a pontos de vista e robustez a padrões repetitivos. A versão de Aciratefi e Aforapro utilizadas foram implementadas usando bibliotecas de programação Cekeikon (KIM HAE, 2015b), Proeikon (KIM HAE, 2015a) e OpenCV (INTEL, 2014). A versão de Asift utilizada foi obtida do site do autor (MOREL; YU, 2014).

Utilizamos dois critérios de avaliação diferentes, já que Asift indica os casamentos utilizando linhas de correspondência entre os pontos-chaves enquanto Aciratefi e Aforapro indicam o casamento traçando um círculo e um retângulo respectivamente. Assim, consideramos que um casamento é correto se: (a) para Asift, se existirem pelo menos três linhas indicando corretamente a correspondência entre a imagem Q e sua instância achada na imagem A . (b) para Aciratefi e Aforapro, se o círculo ou retângulo intersecta pelo menos 60% da área da instância procurada.

5.1.1. Descrição da base de imagens utilizada

Para realizar os testes, construímos uma base com 247 imagens, divididas em três conjuntos. Cada conjunto foi subdividido em grupos. Um grupo consiste de uma imagem modelo Q e várias imagens de análise A . Em cada imagem A aparece uma instância de Q fotografada desde algum ponto de vista. A seguir, descrevermos os três conjuntos:

5.1.1.1. Conjunto 1: Imagens com variação do ponto de vista

Este conjunto consiste de 47 imagens clássicas, frequentemente usadas para analisar invariância a pontos de vista em algoritmos de casamento de pontos-

chaves. Algumas destas imagens foram usadas nos testes do Asift. Este conjunto está dividido em 3 grupos, contendo 17, 8 e 22 imagens de análise (A). O tamanho das imagens de busca Q é 70x70 pixels e o tamanho das imagens de análise A é 500x375 pixels. Estas imagens foram obtidas da data-base de Morel e Yu (2014) e a data-base de Mikolajczyk (2014). Na Fig.5-1 são apresentadas as imagens modelos (Q) usadas neste conjunto.



Fig. 5-1. Imagens de busca Q utilizadas no Conjunto-1.

Algumas imagens de análise A , pertencentes a grupo B1 do Conjunto-1, são apresentadas na Fig.5-2. Este grupo tem 23 imagens, das quais apresentamos nove.

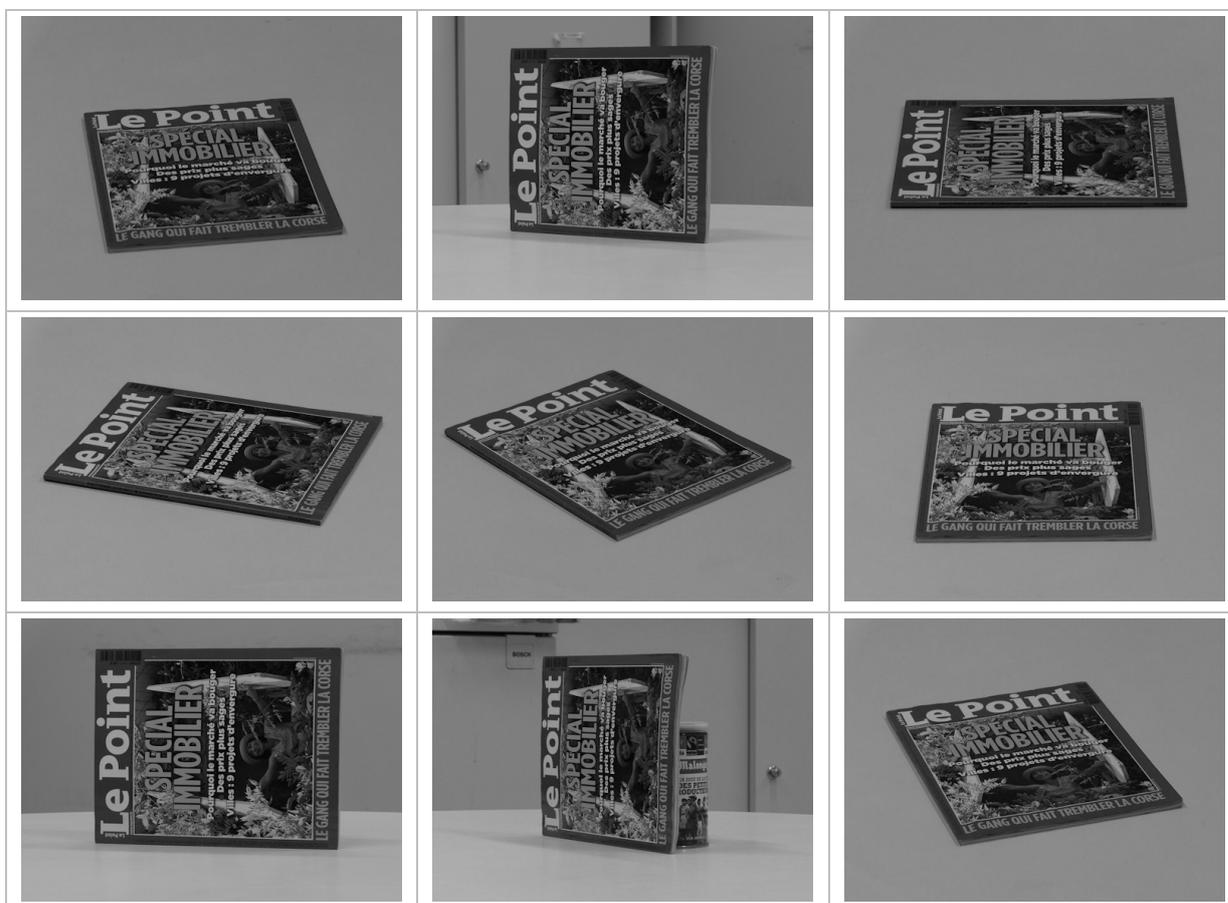


Fig. 5-2. Imagens de análise pertencentes ao grupo B1 do Conjunto-1.

5.1.1.2. Conjunto 2: Imagens que incluem variações de brilho/contraste

Este conjunto consiste de 100 imagens, divididas em 10 grupos. Cada grupo contém uma imagem modelo (Q) e nove imagens a serem analisadas (A). Em cada imagem A aparece uma instância de Q fotografada desde um ponto de vista diferente. Além disso, nestas imagens induzimos alterações de brilho/contraste realizando; mudanças de fundo, variação de iluminação e/ou efeitos de sombras. O tamanho das imagens Q varia tipicamente entre 50x50 e 70x70 pixels, entanto o tamanho das imagens A é 500x375 pixels. Na Fig.5-3 são apresentadas as imagens modelos (Q) usadas neste conjunto.



Fig. 5-3. Imagens modelo Q utilizadas no Conjunto-2.

As nove imagens de análise do grupo D2 do Conjunto-2 são apresentadas na Fig.5-4.





Fig. 5-4. Imagens de análise (A) pertencentes ao grupo D2 do Conjunto-2.

Note na Fig.5-4 como as variações do fundo, iluminação e os efeitos de sombras geram grandes mudanças no brilho/contraste das imagens.

5.1.1.3. Conjunto 3: Imagens que incluem padrões repetitivos

Na Fig.5-5 são apresentadas as imagens modelos (Q) usadas no Conjunto 3.

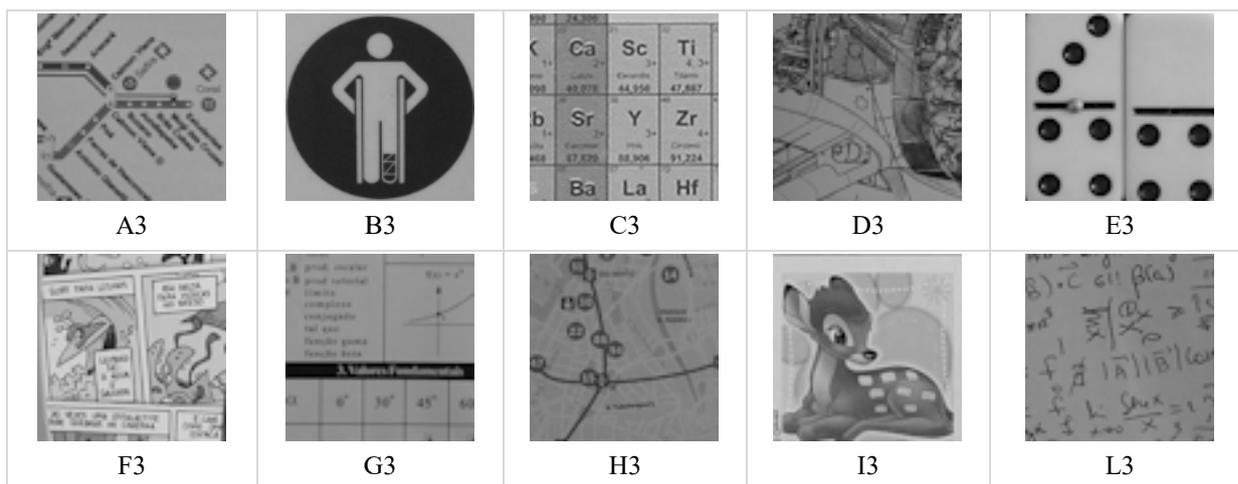


Fig. 5-5. Imagens Q utilizadas no Conjunto-3.

Este conjunto-3 consiste de 100 imagens, divididas em 10 grupos. Cada grupo contém uma imagem modelo (Q) e nove imagens a serem analisadas (A). Estas imagens, além de ter mudanças no ponto de vista, possuem padrões repetitivos. O

tamanho das imagens Q varia tipicamente entre 50x50 e 70x70 pixels, enquanto que o tamanho das imagens A foi 500x375 pixels.

As nove imagens de análise do grupo C3 do Conjunto-3 são apresentadas na Fig.5-6.

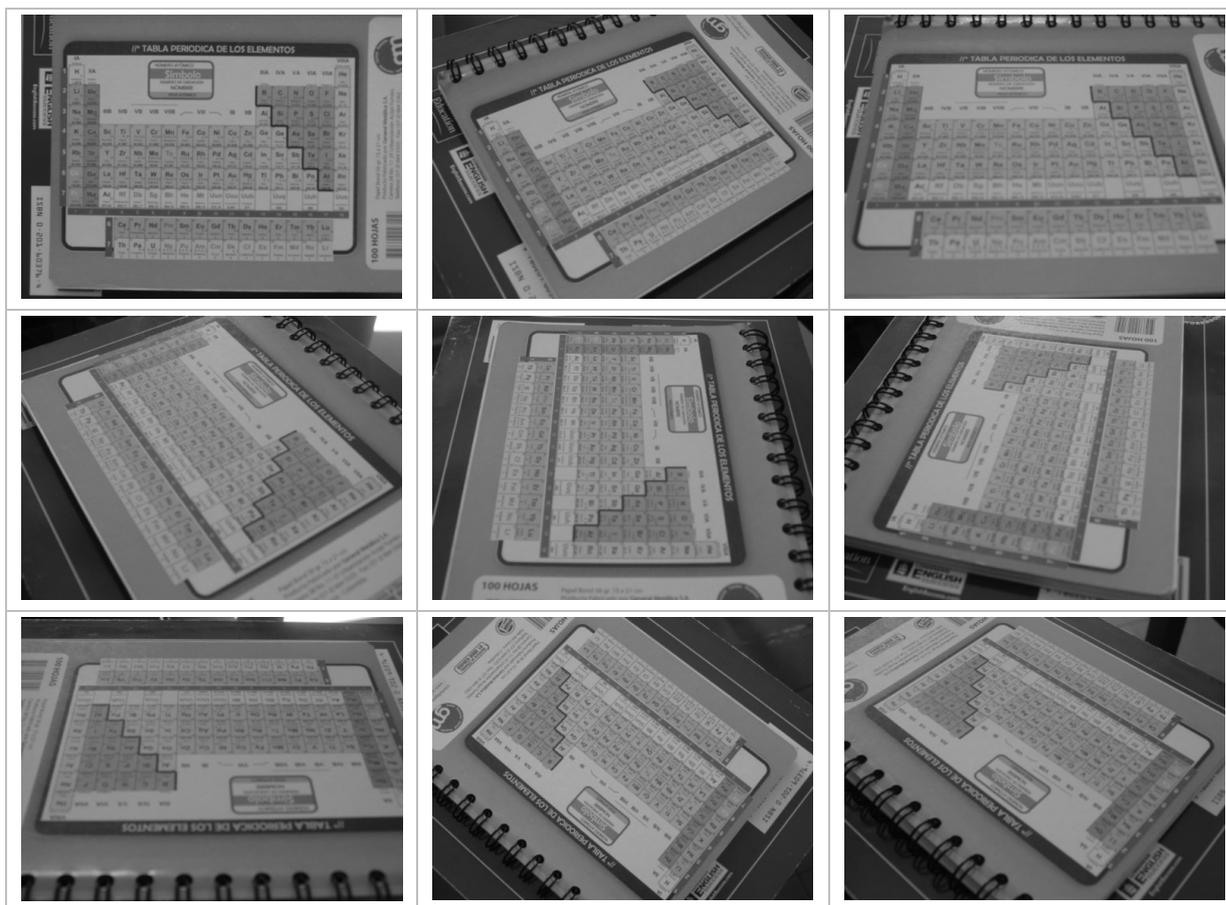


Fig. 5-6. Imagens A pertencentes ao grupo C3 do Conjunto-3.

Neste grupo foram utilizadas fotografias de: mapas do metrô, revista de desenhos, sinais informativas, mapas da cidade, tabela periódica, fichas de dominó e rascunhos de fórmulas matemáticas. Note que as imagens Q (Fig.5-5) foram selecionadas para favorecer a repetição dos seus padrões (subpartes) em outras partes da imagem A .

5.1.2. Testes de invariância a pontos de vista

Para testar a invariância a pontos de vista foram utilizadas as imagens do Conjunto-1. Estas imagens, além de permitir comparar Aciratefi e Aforapro com Asift, também permitem uma comparação indireta para outros três métodos invariantes a transformações afins: Mser, Harris-affine, Hessian-affine. O trabalho de Morel e You

(2009), utilizando as mesmas imagens, concluiu que nenhum destes métodos consegue achar objetos com uma mudança de ponto de vista $\theta \geq 70^\circ$. Na Fig.5-7 é apresentado um exemplo dos resultados obtidos.

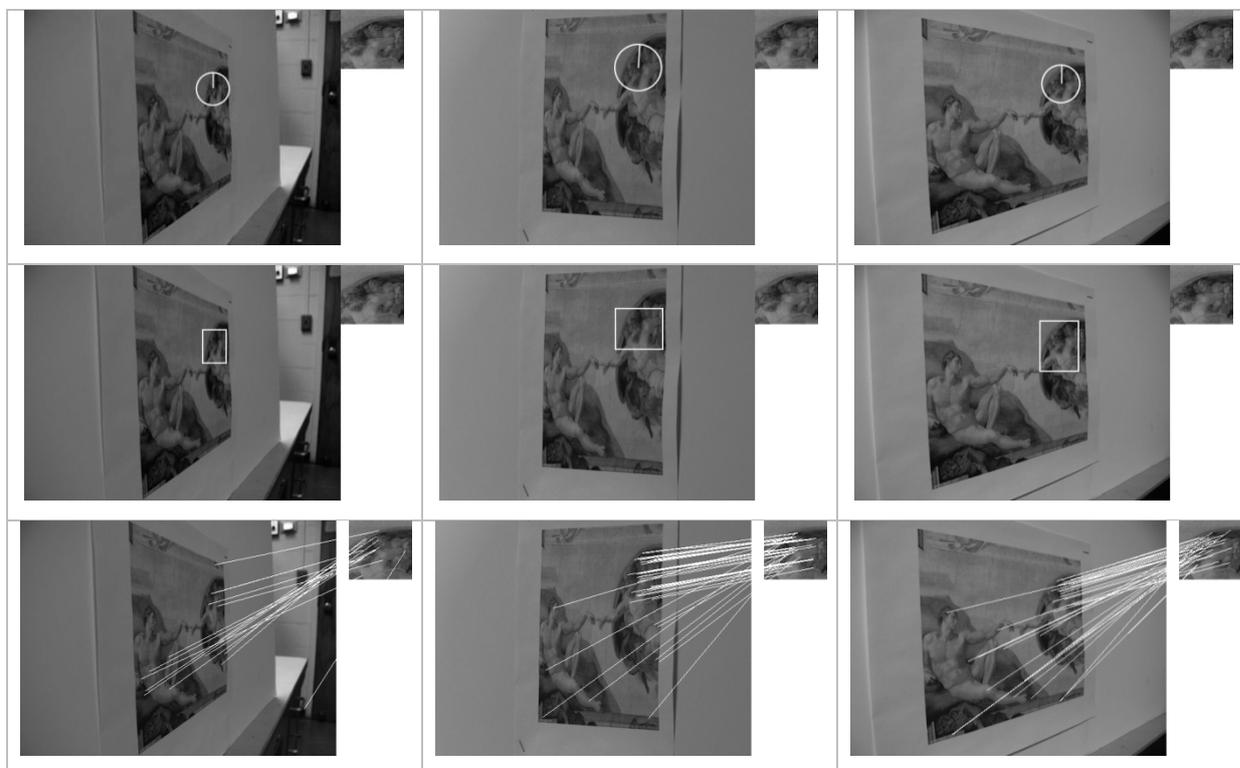


Fig. 5-7. Exemplo dos resultados da invariância a pontos de vista. De cima para baixo Aciratefi, Aforapro e Asift.

A Tab.5-1 resume os resultados obtidos neste item.

Tab. 5-1. Taxas de acerto e tempos de processamento no Conjunto-1.

Grupo	Possíveis casamentos	ASIFT		Aforapro		Aciratefi	
		acertos (%)	tempo (s)	acertos (%)	tempo (s)	acertos (%)	tempo (s)
A1	16	100	45,2	100	255,4	100	210,6
B1	23	100	38,1	100	257,9	91	305,2
C1	8	100	56,1	100	259,5	100	247,4
Total	47	100	46,5	100	257,3	97	254,4

O parâmetro “possíveis casamentos” indica o número de instâncias de Q existentes no grupo avaliado. Este número coincide com o número imagens do grupo, já que cada imagem A tem uma única instância de Q . O parâmetro “tempo” é calculado como o valor médio do tempo total gasto no processamento das imagens do grupo.

Aforapro e Asift tiveram 100% de acertos e Aciratefi acertou 97%. Aciratefi errou em casos onde o ângulo de inclinação foi $\theta > 80^\circ$. Na Fig.5-8 apresentamos os dois casos onde Aciratefi errou.



Fig. 5-8. Limitações de Aciratefi em invariância a pontos de vista. São apresentados dois exemplos onde Aciratefi realizou casamento errado, e a vista frontal da imagem de análise com a indicação (em quadro branco) da instância procurada.

No tempo de processamento, Asift foi aproximadamente 5 vezes mais rápido do que Aciratefi e Aforapro.

5.1.3. Testes de invariância a mudanças de brilho/contraste

Os objetivos destes testes foram: (a) demonstrar que Aciratefi e Aforapro são invariantes a pontos de vista e mudanças de brilho/contraste; (b) comparar o desempenho destes dois algoritmos com o algoritmo do estado da arte (Asift). Na realização dos testes foram utilizadas as imagens do Conjunto-2. Um exemplo dos resultados obtidos é apresentado na Fig.5-9.



Fig. 5-9. Exemplo dos resultados da invariância a mudanças de brilho/contraste. De cima para baixo: Aciratefi, Aforapro e Asift. Aciratefi e Aforapro acertam todos os 3 testes, enquanto que Asift acerta apenas 1 dos 3 (exemplo 2), com mais de 3 casamentos corretos de pontos-chaves.

Os resultados obtidos com o Conjunto-2 são apresentados na Tab.5-2.

Tab. 5-2. Taxas de acerto e tempos de processamento no Conjunto-2.

Grupo	Possíveis casamentos	ASIFT		Aforapro		Aciratefi	
		acertos (%)	tempo (s)	(%)	(s)	(%)	(s)
A1	10	60	120,4	100	230,8	80	181,3
B1	10	40	100,7	100	150,1	100	202,4
C1	10	50	54,2	80	164,8	90	216,0
D1	10	20	56,6	90	204,3	100	189,7
E1	10	30	49,0	100	162,5	70	174,5
F1	10	40	66,3	90	207,3	90	168,7
G1	10	60	59,8	100	176,0	100	210,1
H1	10	100	57,1	100	208,4	90	185,3
I1	10	80	70,6	100	223,7	100	180,8
L1	10	50	90,2	90	180,5	100	120,5
Total	100	53	72,5	95	190,8	92	182,9

Os resultados da Tab. 5-2 demonstram que Aciratefi e Aforapro superam Asift quando as imagens apresentam variações de brilho/contraste (taxa de acertos 92%,

95% e 53% respectivamente). Asift continua sendo o mais rápido dos três algoritmos. No entanto, o tempo de processamento de Aciratefi e Aforapro diminuíram comparado aos resultados da Tab. 5-1, enquanto que o tempo de processamento de Asift diminuiu. Isto aconteceu porque as imagens do Conjunto-2 têm bastante textura o que aumenta a complexidade de busca dos pontos-chaves de Asift. Estes resultados demonstram que Aciratefi e Aforapro são de fato invariantes a mudanças de brilho/contraste, e que superam Asift em situações com grandes variações de brilho/contraste.

5.1.4. Testes de invariância a padrões repetitivos

Para avaliar a invariância a padrões repetitivos foram utilizadas as imagens do Conjunto-3. Na Fig.5-10 são apresentados três exemplos dos resultados obtidos por cada algoritmo.

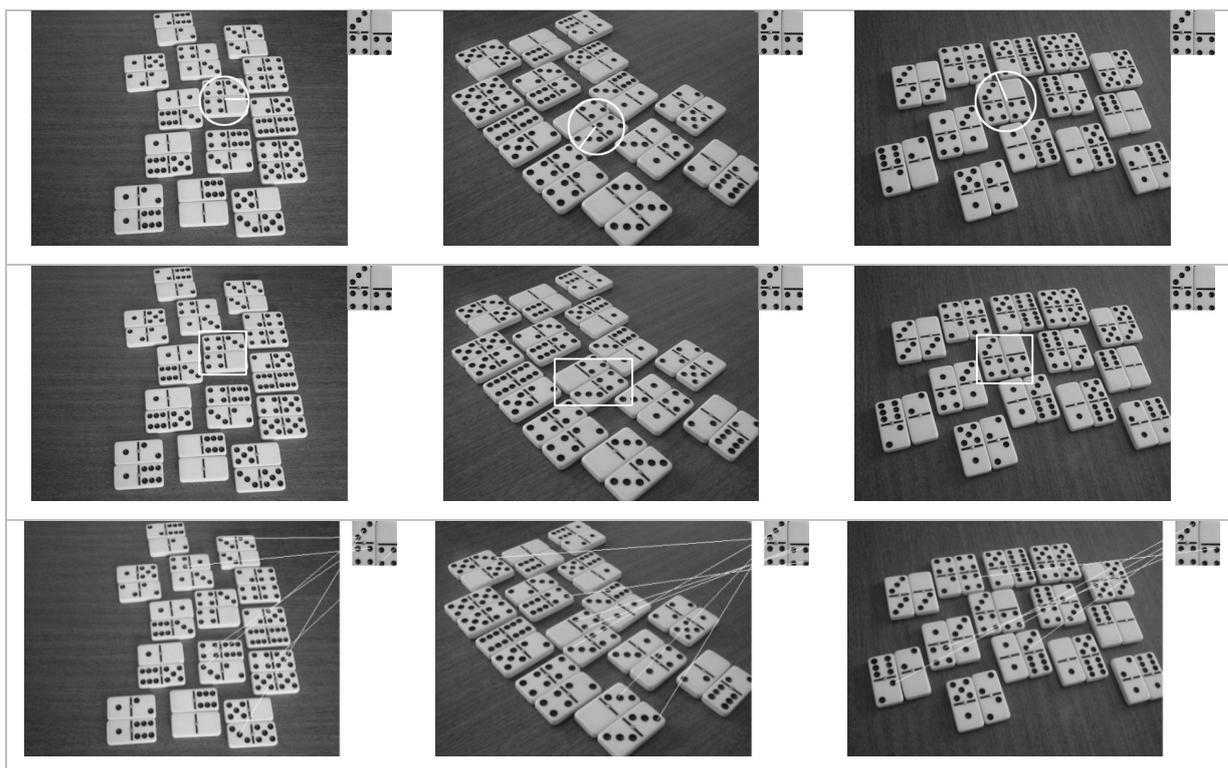


Fig. 5-10. Exemplo dos resultados da invariância a padrões repetitivos. De cima para baixo, Aciratefi, Aforapro e Asift. Aciratefi e Aforapro acertam todos os três testes, enquanto que Asift acerta apenas 1 dos 3 (exemplo 2), com mais de 3 pontos-chaves casados corretamente.

Na Tab.5-3 são apresentados os resultados obtidos com imagens do Conjunto-3. A Tab.5-3 mostra superioridade do Aciratefi e Aforapro para Asift, com taxas de acerto de 91, 92 e 35% respectivamente. Asift continua sendo o algoritmo mais rápido

dos três. Estes resultados sugerem que Aciratefi e Aforapro são robustos a padrões repetitivos, e que superam neste item o Asift.

Tab. 5-3. Taxas de acerto e tempos de processamento no Conjunto-3.

Grupo	Possíveis casamentos	Asift		Aforapro		Aciratefi	
		acertos (%)	tempo (s)	(%)	(s)	(%)	(s)
A1	10	10	50,3	90	230,4	100	180,7
B1	10	10	56,9	80	218,9	90	210,5
C1	10	40	110,5	70	183,1	90	204,8
D1	10	60	168,1	100	240,4	100	194,1
E1	10	30	60,6	90	220,8	100	209,3
F1	10	40	54,3	100	190,3	80	225,4
G1	10	50	140,6	100	205,4	80	183,2
H1	10	30	58,7	100	228,0	90	246,4
I1	10	20	50,8	100	259,5	100	239,6
L1	10	60	150,7	90	216,2	80	195,2
Total	100	35	90,1	92	219,3	91	208,9

5.1.5. Análise dos resultados

Foram realizados três classes de experimentos, para comparar Aciratefi, Aforapro e Asift. Estes experimentos avaliaram a invariância a pontos de vista, mudanças de brilho/contraste e padrões repetitivos. O algoritmo Asift foi utilizado como referência do estado da arte. O Gráfico 5.1 sumariza os resultados de acertos obtidos.

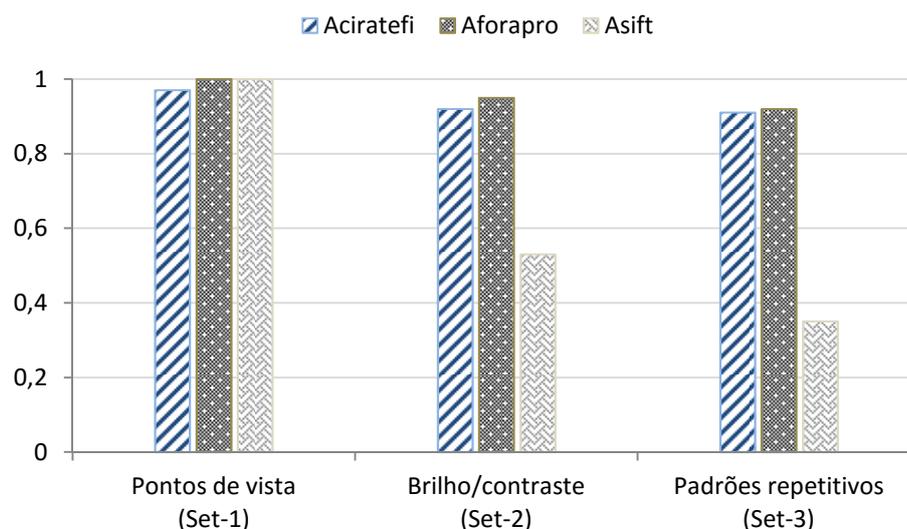


Gráfico 5- 1. Taxa de acertos de Aciratefi, Aforapro e Asift

Estes resultados apresentam um ótimo desempenho de Aciratefi e Aforapro, em todos os casos superam a marca dos 90%. Enquanto, Asift apresenta baixo desempenho nos Conjunto-2 (53%) e Conjunto-3 (35%).

Os resultados obtidos com o Conjunto-1 demonstram que Aciratefi e Aforapro são invariantes a pontos de vista. Aciratefi apresenta limitações para pontos de vista com ângulo de inclinação $\theta > 80^\circ$ (ver Fig.5-8). Estas limitações devem-se basicamente as diferenças existentes no tamanho da região de casamento. A região de casamento refere-se à quantidade de pixels que são casados no processo. Aciratefi utiliza uma região maior do que a utilizada pelos outros dois algoritmos. Na Fig.5-11 exemplificamos a relação de tamanhos da região de casamento dos três algoritmos.

Algoritmo	Região de casamento	Tamanho
Asift	<i>Ponto chave</i>	
Aforapro	<i>Submodelo</i>	
Aciratefi	<i>Modelo</i>	

Fig. 5-11. Quadro comparativo da região de casamento do Asift, Aforapro e Aciratefi. O desenho quadrado representa a área da imagem modelo Q e o desenho em circular representa o tamanho da região de casamento.

Aciratefi utiliza a área toda do modelo como área de casamento aumentando as chances de erros com grandes mudanças de vista. As simulações de pontos de vista para ângulos de inclinação $\theta > 80^\circ$ perdem precisão. Aforapro e Asift levam vantagem neste item por precisarem somente de pelo menos três regiões de casamento para ter sucesso. No entanto, utilizar a imagem toda como região de casamento robustece Aciratefi em outras aplicações, como nos casos com padrões repetitivos.

Os resultados obtidos com Conjunto-2 e Conjunto-3 sugerem que Aciratefi e Aforapro superam o estado-da-arte na combinação de pontos de vista, padrões repetitivos e mudanças de brilho/contraste. Asift apresenta deficiência nesta combinação por herdar estas limitações do Sift.

Asift utiliza o algoritmo Ransac (Random Sample Consensus) (FISCHLER; BOLLES, 1981; WEI; KOSECKA, 2006) para remover casamentos incorretos em cada imagem simulada. Ransac tem uma limitação quando processa imagens que incluem padrões repetitivos. Ransac estima a matriz homográfica usando uma amostragem dos casamentos achados. A matriz homográfica é usada para traçar linhas epipolares entre as duas imagens. Um ponto na primeira imagem produzirá uma linha epipolar na segunda imagem, e vice-versa (HARTLEY; ZISSERMAN, 2004). O problema existe quando uma linha epipolar passa perto de um casamento falso. Ransac calcula que o casamento-falso satisfaz as restrições da linha epipolar e classifica o casamento-falso como verdadeiro. Estes casos usualmente acontecem quando uma linha epipolar falsa cruza pelo menos uma repetição do mesmo padrão (LE BRESE, et al., 2010).

Aciratefi e Aforapro não possuem estas limitações porque, no seu processo de extração de características, calculam o valor médio de várias trajetórias circulares e radiais sobre a vizinhança do pixel.

No tempo de execução, Asift superou Aciratefi e Aforapro. O Gráfico 5.2 resume os resultados obtidos nos testes realizados.

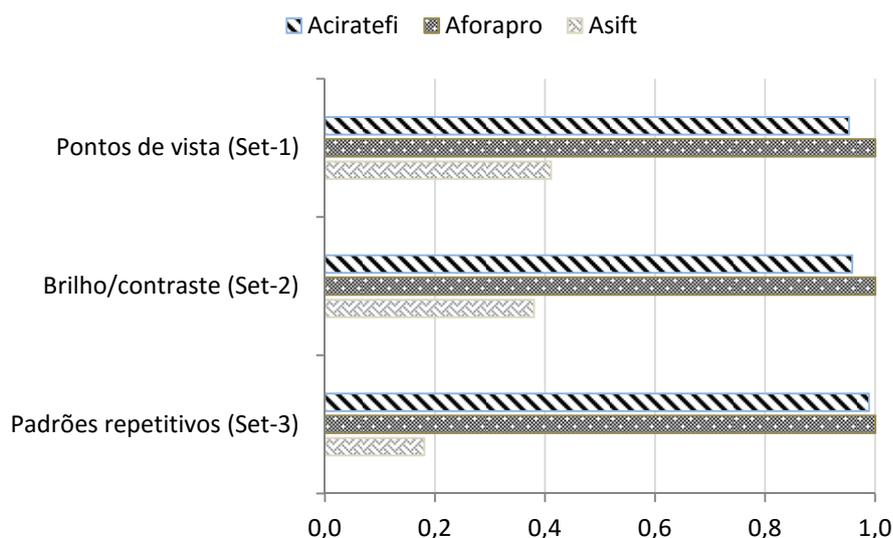


Gráfico 5- 2. Tempo de execução, Aciratefi e Aforapro vs. Asift.

A Tab. 5-4 mostra os tempos médios de execução.

Tab. 5-4. Tempos médios de execução.

	Aciratefi (s)	Aforapro (s)	Asift (s)
Conjunto-3	254,4	257,3	90,1
Conjunto-2	182,9	190,8	72,5
Conjunto-1	208,9	219,3	46,5

O tempo de execução no ASIFT varia de forma significativa de um conjunto para outro. Uma das possíveis causas dessa variação é que Asift gasta muito mais tempo nas imagens que apresentam quantidade elevada de detalhes (texturas), como no caso das imagens do Conjunto-2 e Conjunto-3.

O tempo de execução de Aciratefi e Aforapro poderia ser diminuído implementando-os em hardware, fazendo processamento paralelo em GPU, ou usando instruções vetoriais. As implementações atuais são seriais, e não utilizam todos os núcleos existentes nos processadores atuais. Aciratefi e Aforapro executam os mesmos conjuntos de operações para todos os pixels e poderiam ser facilmente paralelizados. Paralelizar Asift é um pouco mais difícil, embora não seja impossível, já que ele realiza sequências de operações diversas.

Outros fatores que fazem diferença entre estes três algoritmos são: inclinação de transição (detalhes no Capítulo 3) e o ajuste dos parâmetros.

Inclinação de transição refere-se à capacidade do algoritmo para realizar casamentos onde as duas imagens apresentam vistas inclinadas ($\theta \neq \text{ângulo frontal} = 90^\circ$). Aciratefi e Aforapro supõe que a imagem modelo foi fotografada de frente, enquanto que Asift consegue casamentos mesmo que as duas imagens (de busca e de análise) estejam inclinadas.

A implementação atual do Aforapro necessita vários parâmetros que dependem das características das imagens testadas. Por exemplo: o conjunto de relações de escala entre Q e A ; o número de projeções radiais e circulares; o número de candidatos a serem selecionados; etc. Estes parâmetros devem ser fornecidos pelo usuário. Aciratefi e Asift também precisam de alguns parâmetros de entrada. Porém, estes algoritmos funcionam adequadamente com parâmetros pré-estabelecidos. Assim, o usuário praticamente não precisa selecionar nenhum parâmetro manualmente para utilizar Asift ou Aciratefi. Isto não impede do usuário alterar os parâmetros pré-estabelecidos caso queira.

Na Tab. 5-5 resumimos as características dos três algoritmos.

Tab. 5-5. Comparação de características: Aciratefi, Aforapro e Asift.

Características	Asift	Aforapro	Aciratefi
Especialização	casamento de pontos-chaves	casamento de submodelos	casamento de modelos
Transição do ângulo de inclinação	sim	não	não
Invariância a pontos de vista	excelente	excelente	bom
Invariância a mudanças de brilho/contraste	regular	bom	bom
Invariância a padrões repetitivos	limitado	bom	bom
Tempo de execução	bom	limitado	limitado
Obtenção automática dos parâmetros	bom	limitado	bom

Estes resultados sugerem que cada algoritmo tem suas próprias vantagens/desvantagens. Por exemplo: Aforapro responde excelentemente a variações de pontos de vista, no entanto, vários parâmetros devem ser ajustados manualmente; Aciratefi tem limitações para grandes variações de pontos de vista, no entanto, pode-se utilizar os parâmetros de entrada pré-selecionados; Aciratefi e

Aforapro superam as limitações de Asift na invariância a padrões repetitivos e mudanças de brilho/contraste, no entanto, são computacionalmente mais lentos.

5.2. Testes de avaliação da etapa de Filtragem de casamentos

O objetivo destes testes é analisar a filtragem baseada em coerência geométrica, nos modos 'n' e 'f'. Os algoritmos foram implementados utilizando o algoritmo Sift disponibilizado pelo autor (LOWE, 2015), a biblioteca de OpenCV (INTEL, 2014), e a biblioteca Cekeikon (KIM HAE, 2015b).

Foram testadas duas ideias, descritas no Cap. 4: O 'modo n' ordena os pontos-chaves pelo número de casamentos coerentes; e o 'modo f' ordena os pontos-chaves pela função f que considera o número de casamentos coerentes e a razão R entre as distâncias euclidianas do melhor e do segundo melhor casamento. Os desempenhos destes dois modos são comparados com a ordenação obtida pela razão R , utilizada no algoritmo original do Sift.

Os limiares de escala/ângulo testados foram 0,39/13,5 e 0,50/20,0. As distorções das imagens testadas foram: variações de escala, pontos de vista, iluminação, borrão de foco, e compressão JPEG.

5.2.1. Descrição da base de imagens utilizada

As imagens utilizadas foram obtidas da base de imagens de Mikolackjzy (2015). Esta base contém oito grupos de imagens com fotografias de ambientes naturais. Cada grupo tem uma imagem de referência (I_1), e mais cinco imagens ($I_2 \dots I_6$) da mesma cena, mas fotografadas sob diferentes variações de iluminação, escala, ponto de vista, compressão JPEG ou efeitos de borrão. Nas imagens com mudanças do ponto de vista, a câmera varia desde uma vista frontal ($\theta = 0^\circ$) para pequenas inclinações ($\theta \leq 60^\circ$). As mudanças de escala são obtidas pela variação do zoom da lente, e os efeitos de borrão são obtidos pelo ajuste de foco da câmera. Todas as imagens são de resolução média, aproximadamente 800×640 pixels. Na Fig.5-12 são apresentadas as imagens de referência utilizadas nos testes.

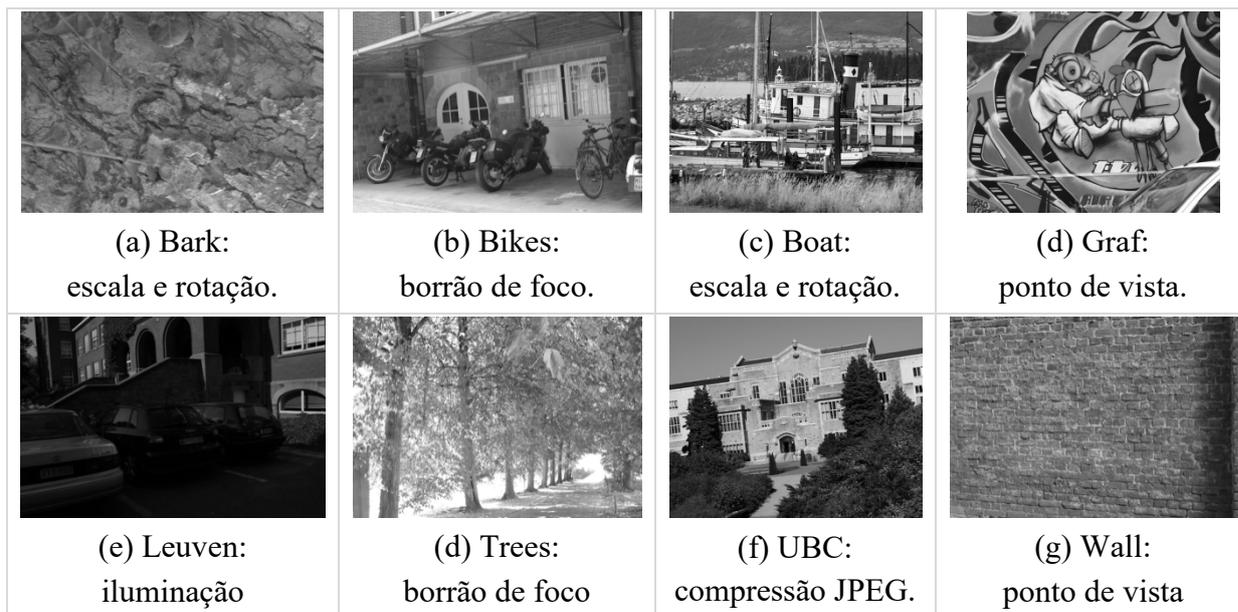


Fig. 5-12. Imagens de referência I_1 utilizadas nos testes de filtragem de casamentos pela coerência geométrica.

Esta base de imagens também inclui uma matriz homográfica (ver Capítulo 2) para cada combinação de imagens (I_1, I_k). Utilizando a matriz homográfica, é possível classificar cada casamento de ponto chave como “verdadeiro” ou “falso”, já que ela associa cada ponto (x_1, y_1) na imagem I_1 com um ponto (x_k, y_k) na imagem I_k . Na Fig.5-13 apresentamos, como exemplo, a sequência de imagens “Boat”.

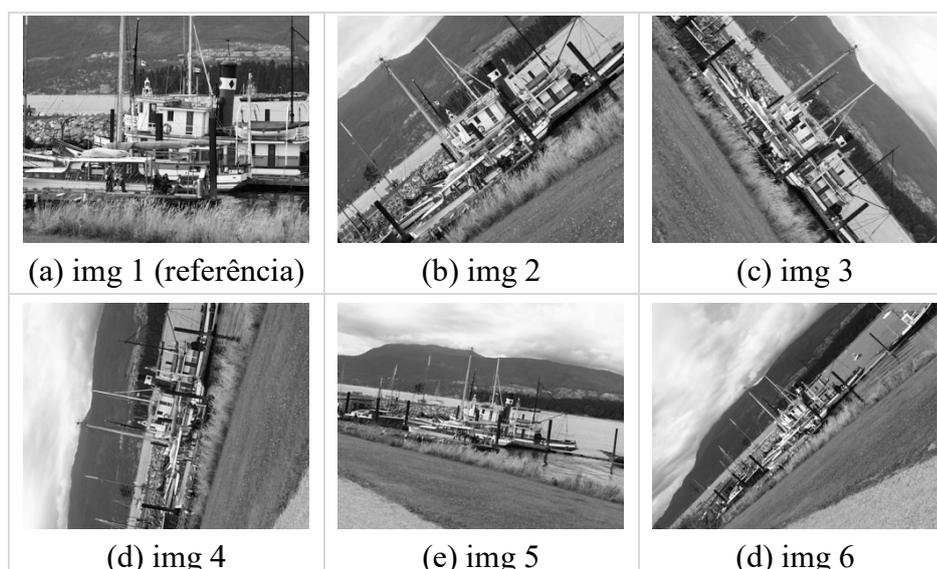


Fig. 5-13. Sequência de imagens “Boat” com variações de escala e rotação. (a) imagem de referência I_1 , (b-d) imagens com mudanças de escala e rotação I_k . Imagens obtidas da data base de Mikolajczyk (2015).

5.2.2. Testes com variações de escala e rotação

Tab. 5-6 apresenta as taxas de erros obtidos com as sequências de imagens “Bark” e “Boat” com variações de escala e rotação.

Tab. 5-6. Taxas de erro e tempos de processamento para imagens com variações de escala e rotação, aceitando casamentos com $R < 0.8$.

Imagens	Número de casamentos (N)	Critérios de avaliação									
		Sift		* Limiar L1(0.39, 13.5)				Limiar L2(0.50, 20.0)			
		erros (%)	tempo (s)	modo n		modo f		modo n		modo f	
		(%)	(s)	(%)	(s)	(%)	(s)	(%)	(s)	(%)	(s)
Bark 1 e 2	649	5,08	4,828	0,00	5,211	0,00	5,242	0,00	5,408	0,00	5,490
Bark 1 e 3	570	4,74	5,788	0,00	6,380	0,00	6,162	0,00	6,368	0,00	6,400
Bark 1 e 4	716	5,31	6,452	2,93	7,052	2,65	7,036	2,23	7,052	2,51	7,038
Bark 1 e 5	469	5,33	6,140	1,49	6,599	1,49	6,615	1,28	6,702	1,28	6,680
Bark 1 e 6	294	13,27	6,310	10,20	6,849	10,20	6,880	10,20	6,865	10,20	6,860
Boat 1 e 2	2565	3,98	19,582	0,00	23,496	0,27	23,348	0,04	23,458	0,31	23,460
Boat 1 e 3	1945	6,79	15,878	0,21	19,632	0,41	19,868	0,21	19,434	0,62	19,432
Boat 1 e 4	858	20,86	13,498	2,56	16,990	4,43	16,862	3,03	16,772	4,55	16,760
Boat 1 e 5	626	24,60	12,806	8,15	16,180	11,98	16,120	8,31	16,030	12,78	16,012
Boat 1 e 6	341	46,04	11,094	17,89	14,468	25,22	14,468	19,65	14,430	26,98	14,378

* Limiar L(escala, ângulo)

Em todos os casos, os ‘modos n’ e ‘modo f’ apresentam taxas de erros menores do que Sift original. A maior diferença registrada foi nas imagens ‘Boat 1 e 6’ onde o ‘modo n’ registra 28,15% menos erros do que Sift. O ‘modo n’, na maioria dos casos, apresenta taxa de erros ligeiramente menor do que o ‘modo f’.

O maior aumento de tempo de processamento foi de 19,9%. Esta é um dos casos que apresenta grande número de casamentos aceitos (Boat 1 e 2).

No Gráfico 5.3 apresentamos um exemplo da taxa de acertos entre duas imagens da sequência Boat, variando a taxa de aceitação. Evidentemente, a taxa de sucesso é definida como um menos a taxa de erros.

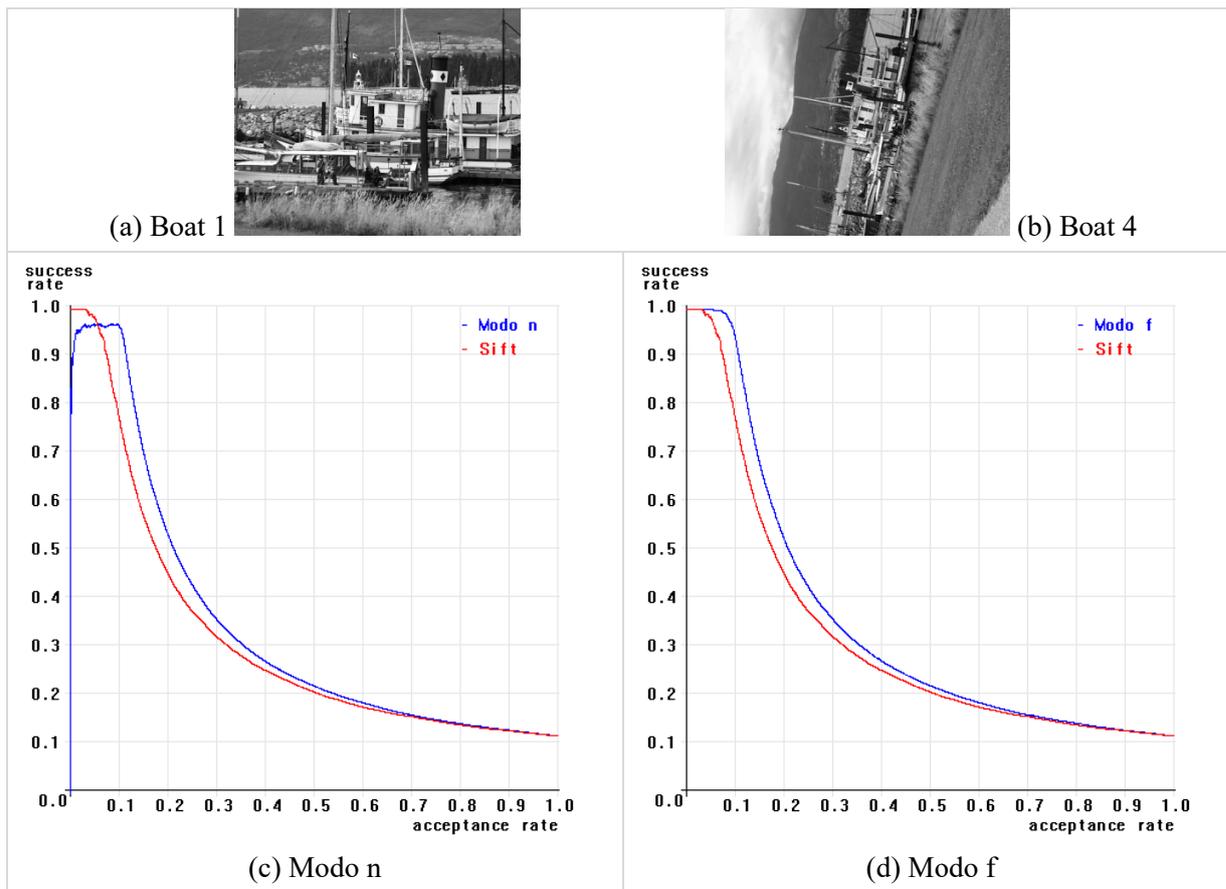


Gráfico 5- 3. Taxa de acertos entre Boat 1 e Boat 4.

(a-b) Imagens utilizadas. (c-d) taxa de acertos vs. porcentagem de casamentos aceitos como verdadeiros, 'modo n' e 'modo f' em azul, e Sift original em vermelho. Note que 'modo n' é pior que Sift original para baixa taxa de aceitação; porém 'modo f' é sempre melhor que Sift original.

Os gráficos mostram que o 'modo n' e 'modo f' superam a taxa de acertos do Sift original. No entanto, o 'modo n' apresenta problemas no início do gráfico, tendo taxa de sucesso menor do que Sift original. Provavelmente, há um ou mais casamentos errados com alto número de casamentos coerentes. Isto é problemático, já que o Sift é normalmente utilizado na região de baixa taxa de aceitação. Este problema pode ser superado usando 'modo f', já que a taxa de sucesso no 'modo f' é maior do que a do Sift original para todas as taxas de aceitação.

5.2.3. Testes com borrão de foco

Na Tab. 5-7 são apresentados os resultados obtidos com as duas sequências de imagens que contém distorções causadas pelo borrão de foco (bikes e trees).

Tab. 5-7. Taxas de erro e tempos de processamento para imagens com borrão de foco, aceitando casamentos com $R < 0.8$.

Imagens	Número de casamentos (N)	Critérios de avaliação									
		Sift		* Limiar L1(0.39, 13.5)				Limiar L2(0.50, 20.0)			
		erros (%)	tempo (s)	modo n		modo f		modo n		modo f	
		(%)	(s)	(%)	(s)	(%)	(s)	(%)	(s)	(%)	(s)
Bikes 1 e 2	893	10,64	4,570	3,25	5,112	3,70	4,838	4,93	4,830	4,26	4,808
Bikes 1 e 3	731	20,66	3,948	3,97	4,480	7,11	4,378	4,38	4,358	7,52	4,360
Bikes 1 e 4	475	25,68	3,546	12,21	4,008	14,11	3,918	11,37	3,916	13,89	3,926
Bikes 1 e 5	470	42,98	3,366	32,55	3,832	34,04	3,804	32,13	3,726	34,26	3,376
Bikes 1 e 6	414	54,59	3,168	43,96	3,714	44,69	3,628	44,93	3,628	44,93	3,608
Trees 1 e 2	1926	2,80	36,810	1,09	44,541	0,16	48,252	0,21	46,610	0,16	45,618
Trees 1 e 3	1638	4,52	47,612	0,12	58,834	0,43	60,504	0,24	56,570	0,61	58,620
Trees 1 e 4	752	12,10	32,596	1,46	42,542	1,86	44,022	1,86	42,936	2,66	43,402
Trees 1 e 5	435	25,29	18,876	7,59	26,366	1,84	27,230	6,21	29,706	2,07	29,046
Trees 1 e 6	287	41,11	12,900	14,29	20,958	12,54	23,426	17,70	21,770	13,24	22,296

* Limiar L(escala, ângulo)

Em todos os casos, os 'modos n' e 'modo f' apresentam taxa de erros menores do que Sift original. A maior diferença observada foi nas imagens 'Trees 1 e 6' onde o 'modo f' registra 28,57% menos erros do que Sift. O 'modo n' apresenta normalmente taxa de erros ligeiramente menor do que o 'modo f'. No Gráfico 5.4 apresentamos um exemplo da taxa de acertos entre duas imagens da sequência Bikes, para diferentes taxas de aceitação.

O maior aumento no tempo de execução foi de 31,0%, no caso em que apresenta o maior número de casamentos aceitos (Trees 1 e 2).

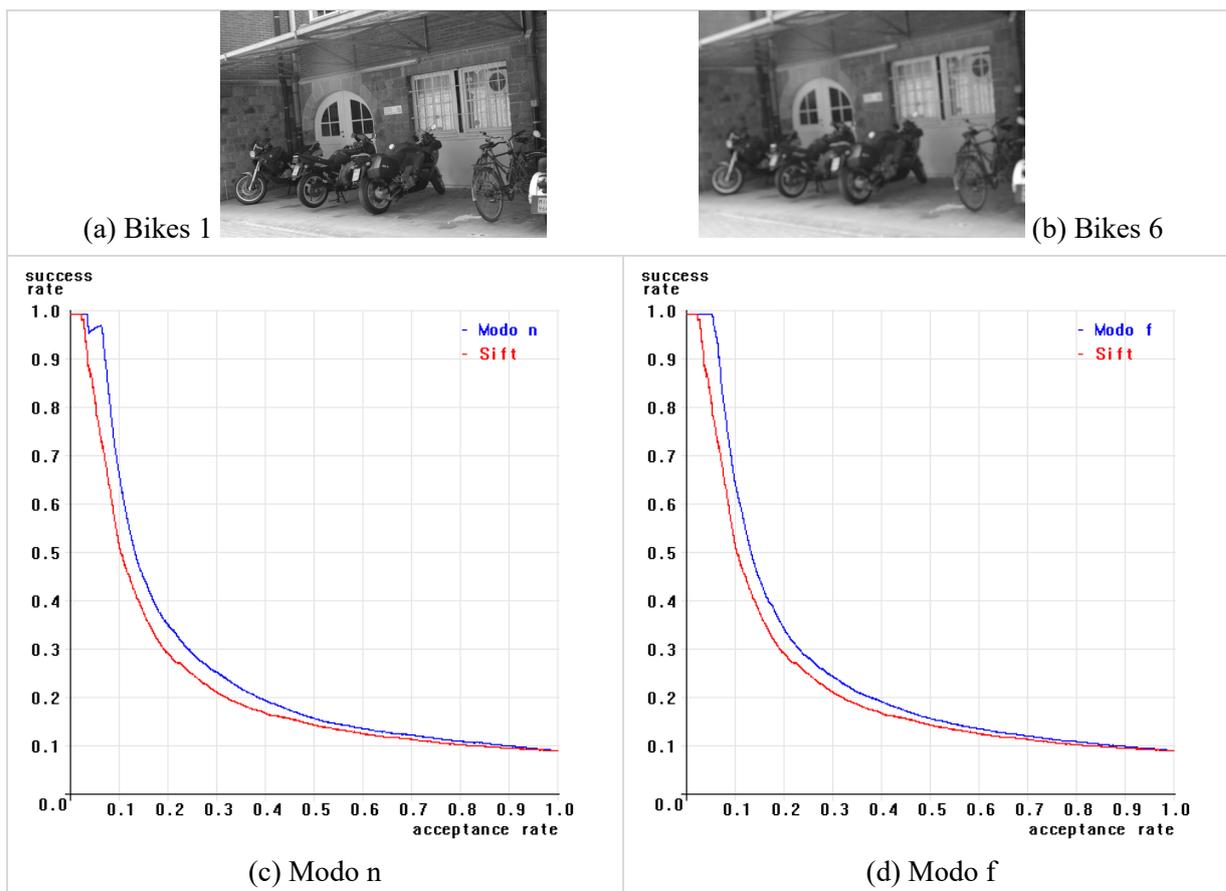


Gráfico 5- 4. Taxa de acertos entre Bikes 1 e Bikes 6. (a-b) Imagens utilizadas. (c-d) taxa de acertos vs. porcentagem de casamentos aceitos como verdadeiros, 'modo n' e 'modo f' em azul, e Sift original em vermelho.

Os gráficos mostram que o 'modo n' e 'modo f' superam a taxa de acertos de Sift.

5.2.4. Testes com compressão JPEG

Na Tab. 5-8 são apresentados os resultados obtidos com a sequência de imagens UBC que comprime a imagem de referência com diferentes taxas de compressão JPEG.

Tab. 5-8. Taxas de erro e tempos de processamento para imagens com diferentes níveis de compressão JPEG, aceitando casamentos com $R < 0,8$.

Imagens	Número de casamentos (N)	Critérios de avaliação									
		Sift		* Limiar L1(0.39, 13.5)				Limiar L2(0.50, 20.0)			
		erros (%)	tempo (s)	modo n		modo f		modo n		modo f	
(%)	(s)			(%)	(s)	(%)	(s)	(%)	(s)		
UBC 1 e 2	3186	1,10	10,886	0,00	12,988	0,16	13,130	0,03	13,185	0,19	13,126
UBC 1 e 3	2550	2,43	12,042	0,00	14,244	0,24	14,052	0,08	14,320	0,31	14,026
UBC 1 e 4	1654	4,96	11,868	0,24	13,752	0,60	13,468	0,03	13,916	1,09	13,956
UBC 1 e 5	803	9,09	9,964	0,75	11,746	0,62	11,236	1,12	11,338	0,62	11,040
UBC 1 e 6	415	20,96	7,010	1,45	8,562	0,16	8,524	3,86	8,232	2,65	8,470

* Limiar L(escala, ângulo)

Em todos os casos os ‘modos n’ e ‘modo f’ apresentam taxas de erros menores do que Sift original. A maior diferença registrada foi nas imagens ‘UBC 1 e 6’ onde o ‘modo n’ registra 19,51% menos erros do que Sift. O ‘modo n’ normalmente apresenta taxas de erros ligeiramente menores do que o ‘modo f’.

O maior aumento de tempo de execução foi 18,9%, correspondendo ao caso que apresenta o maior número de casamentos aceitos (UBC 1 e 3).

No Gráfico 5.5 apresentamos um exemplo da taxa de acertos entre duas imagens da sequência UBC, variando a taxa de aceitação.

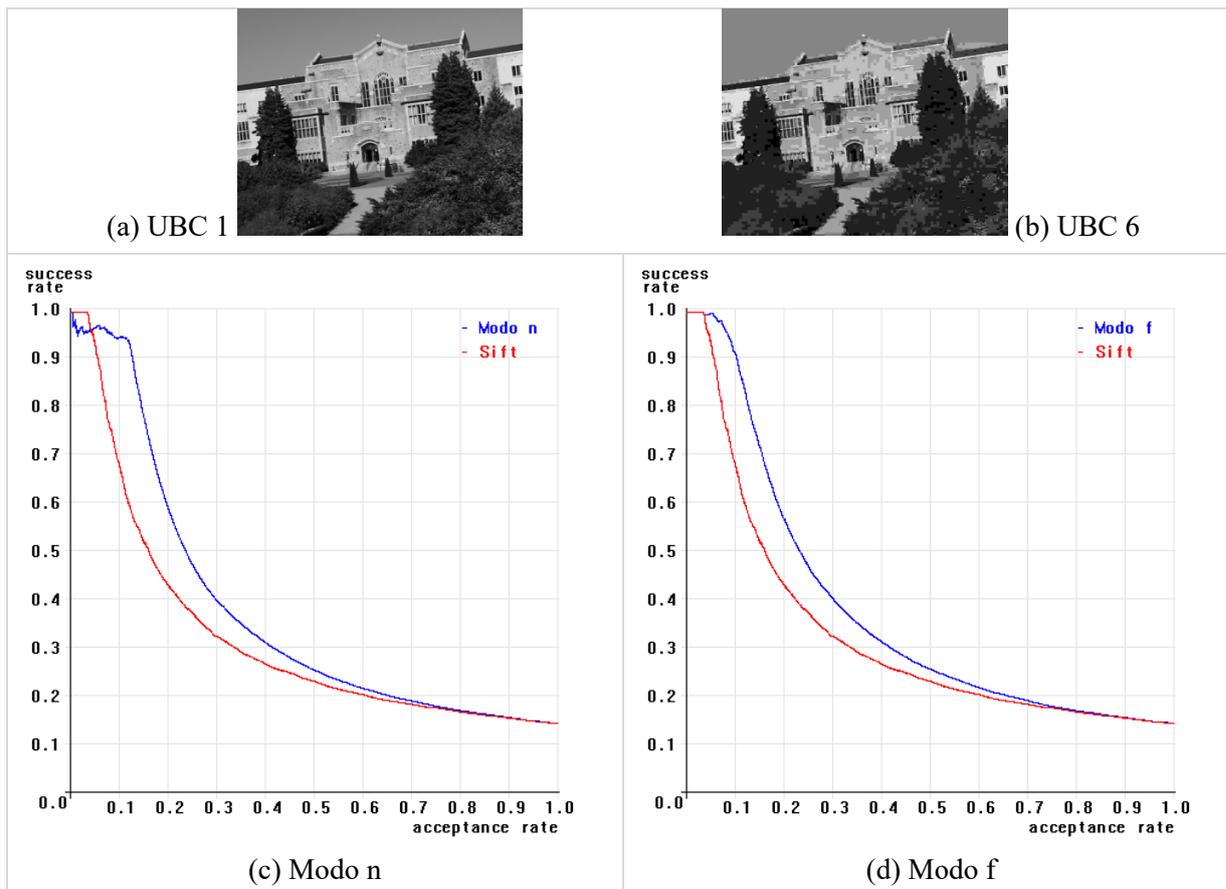


Gráfico 5- 5. Taxa de acertos entre UBC 1 e UBC 6.

(a-b) Imagens utilizadas. (c-d) taxa de acertos vs. porcentagem de casamentos aceitos como verdadeiros, 'modo n' e 'modo f' em azul, e Sift original em vermelho. Note que, para baixas taxas de aceitação, 'modo n' é pior que Sift original; porém 'modo f' é melhor do que Sift original em qualquer situação.

Os gráficos mostram que o 'modo n' e 'modo f' superam a taxa de acertos de Sift. No entanto, o 'modo n' apresenta problemas no início do gráfico, sendo pior do que Sift original nesta região. Esta limitação é superada pelo 'modo f'.

5.2.5. Testes variações de iluminação

A Tab.5-9 apresenta os resultados obtidos com a sequência de imagens Leuven com variações de iluminação.

Tab. 5-9. Taxas de erro e tempos de processamento para imagens com variações de iluminação, aceitando casamentos com $R < 0.8$.

Imagens	Número de casamentos (N)	Critérios de avaliação									
		Sift		* Limiar L1(0.39, 13.5)				Limiar L2(0.50, 20.0)			
		erros (%)	tempo (s)	modo n		modo f		modo n		modo f	
		(%)	(s)	(%)	(s)	(%)	(s)	(%)	(s)	(%)	(s)
Leuven 1 e 2	1245	4,02	3,636	0,80	4,118	1,37	4,272	2,01	3,928	1,45	3,968
Leuven 1 e 3	974	5,34	3,492	0,00	4,032	1,33	3,858	0,00	3,708	1,54	3,726
Leuven 1 e 4	781	6,15	3,228	1,02	3,698	2,43	3,638	1,79	3,536	2,43	3,510
Leuven 1 e 5	685	9,78	3,156	1,31	3,606	2,63	3,516	2,19	3,408	3,21	3,438
Leuven 1 e 6	504	16,47	2,982	0,20	3,252	5,75	3,198	0,40	3,176	6,75	3,216

* Limiar L(escala, ângulo)

Em todos os casos, os modos 'n' e 'f' apresentam taxas de erros menores do que Sift original. A maior diferença observada foi nas imagens 'Leuven 1 e 6' onde o 'modo n' registra 16,27% menos erros do que Sift. O 'modo n' apresenta normalmente taxas de erros ligeiramente menores do que o 'modo f'.

O maior aumento de tempo de execução do foi de 17,4%, no caso com o maior número de casamentos aceitos (Leuven 1 e 2).

No Gráfico 5.6 apresentamos a taxa de acerto versus taxa de aceitação de duas imagens da sequência Leuven.

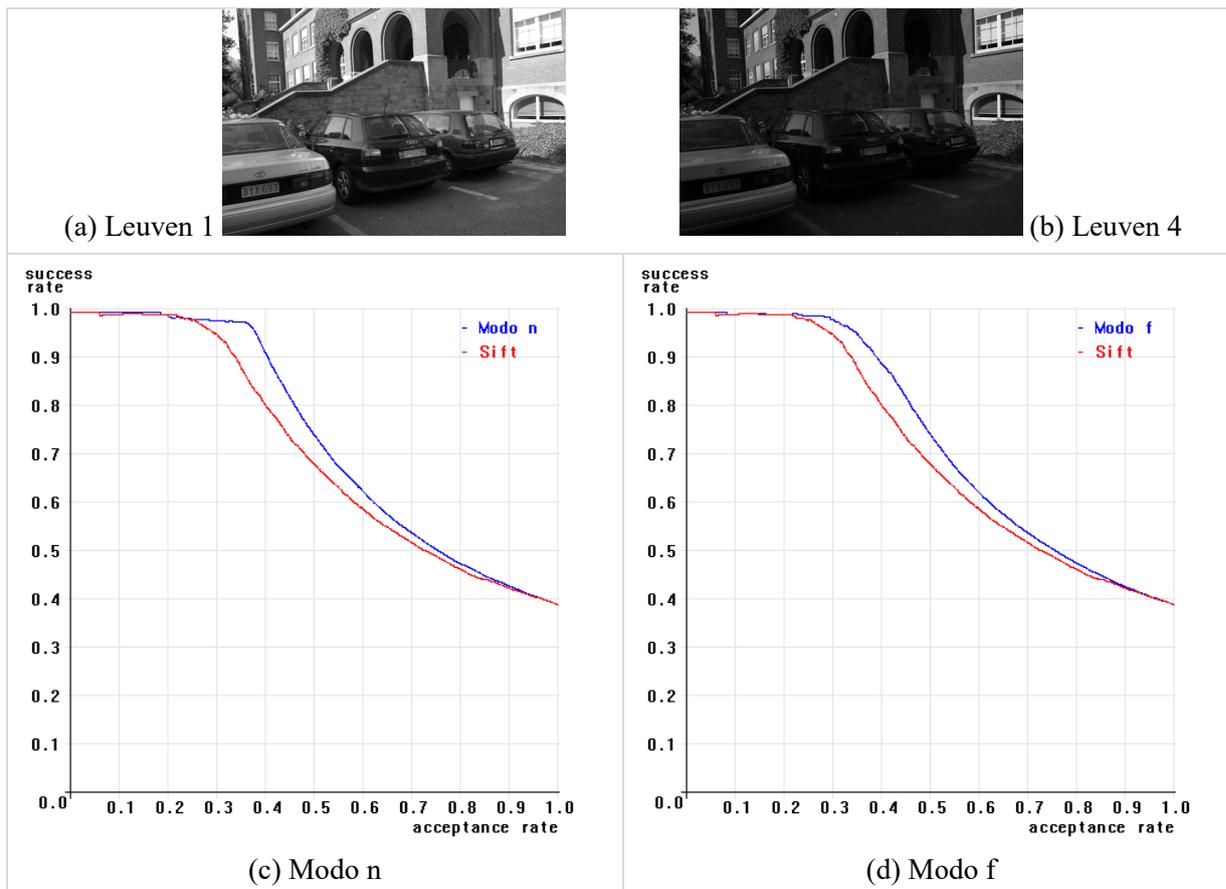


Gráfico 5- 6. Taxa de acertos para variações de iluminação. (a-b) Imagens utilizadas. (c-d) taxa de acertos vs. porcentagem de casamentos aceitos como verdadeiros, 'modo n' e 'modo f' em azul, e Sift original em vermelho.

Os gráficos mostram que o 'modo n' e 'modo f' superam a taxa de acerto de Sift.

5.2.6. Testes com variações do ponto de vista

As mudanças do ponto de vista, nestas seqüências de imagens, incluem pequenas variações do ângulo de inclinação $\theta \leq 60^\circ$. Isto é apropriado para a avaliação, já que Sift não é invariante a pontos de vista. Na Tab. 5-10 são apresentados os resultados obtidos com as seqüências de imagens Graf e Wall com variações do ponto de vista.

Tab. 5-10. Taxas de erro e tempos de processamento para imagens com variações do ponto de vista, aceitando casamentos com $R < 0,8$.

Imagens	Número de casamentos (N)	Critérios de avaliação									
		Sift		* Limiar L1(0.39, 13.5)				Limiar L2(0.50, 20.0)			
		erros (%)	tempo (s)	modo n		modo f		modo n		modo f	
(%)	(s)			(%)	(s)	(%)	(s)	(%)	(s)		
Graffiti 1 e 2	1182	6,26	4,398	0,00	4,778	0,93	5,088	0,08	4,820	1,27	4,950
Graffiti 1 e 3	679	19,15	4,718	3,53	5,232	1,18	5,230	1,33	5,050	0,88	5,058
Graffiti 1 e 4	232	54,31	4,762	11,21	5,262	12,07	5,162	9,48	5,130	9,91	5,166
Graffiti 1 e 5	156	91,03	5,090	77,56	5,430	80,13	5,342	81,41	5,310	80,13	5,280
Graffiti 1 e 6	106	95,28	5,700	92,45	6,130	93,40	5,972	91,51	5,988	94,34	5,968
Wall 1 e 2	5446	0,50	28,332	0,28	35,300	0,00	36,366	0,61	36,120	0,02	36,266
Wall 1 e 3	4228	0,69	27,966	0,17	35,164	0,02	35,576	0,33	35,190	0,02	35,638
Wall 1 e 4	2313	2,12	27,056	1,56	35,286	0,09	34,172	2,08	33,022	0,09	33,870
Wall 1 e 5	592	9,29	28,344	4,90	34,170	0,84	33,590	3,21	34,004	0,68	33,738
Wall 1 e 6	94	75,53	27,986	85,11	34,272	44,68	32,300	80,85	33,118	44,68	33,400

* Limiar L(escala, ângulo)

Em todos os casos (exceto Wall 1 e 6), os modos 'n' e 'f' apresentam taxas de erros menores do que Sift original. A maior diferença registrada foi nas imagens 'Graf 1 e 4' onde o 'modo n' registra 44,83% menos erros do que Sift. Não há um vencedor claro entre modos 'n' e 'f'.

O maior aumento de tempo de execução foi de 28,3% correspondendo ao caso com o maior número de casamentos aceitos (Wall 1 e 2).

No Gráfico 5.7 apresentamos um exemplo de taxa de acertos entre duas imagens da sequência Graf, variando taxa de aceitação. Note que 'modo f' é melhor do que 'modo n' para baixas taxas de aceitação, justamente a situação em que Sift normalmente é usado.

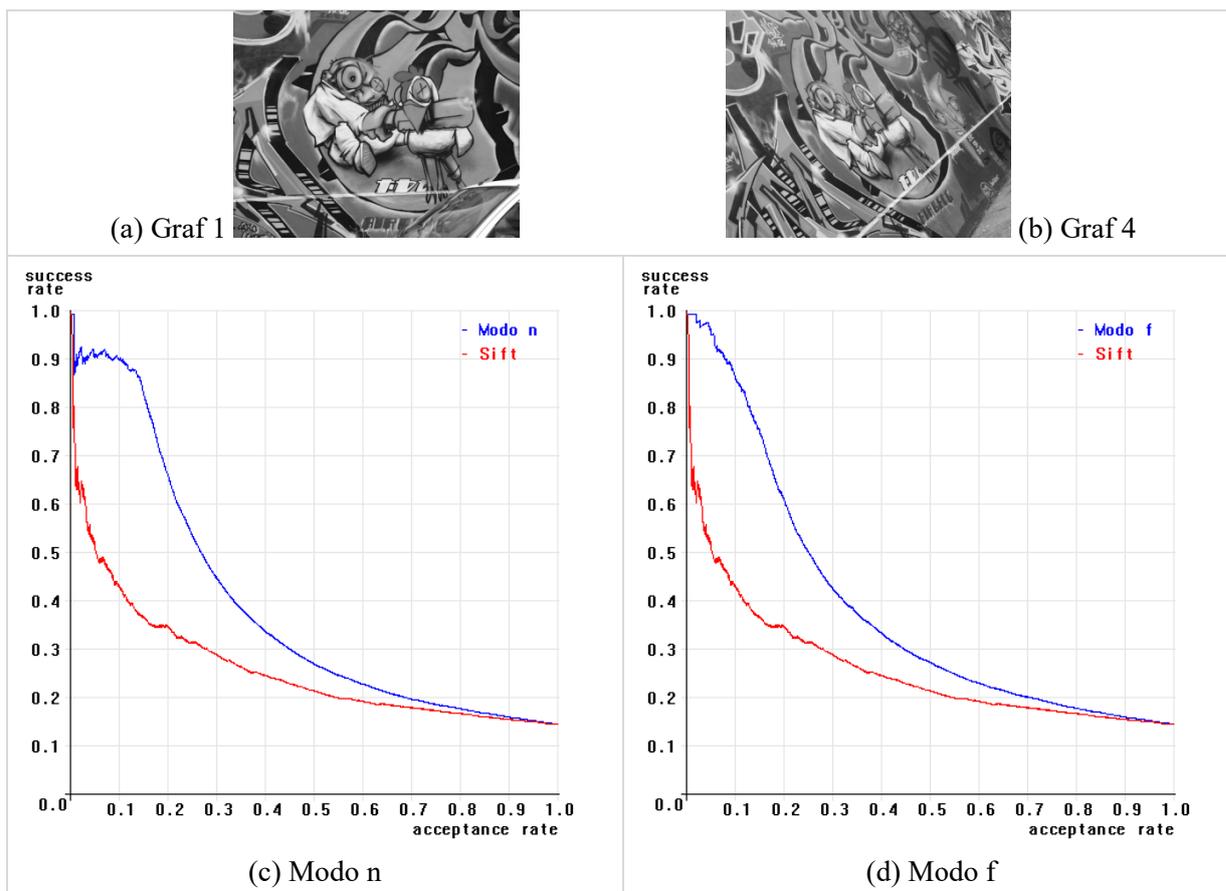


Gráfico 5- 7. Taxa de acertos entre Graf 1 e Graf 4. (a-b) Imagens utilizadas. (c-d) taxa de acertos vs. porcentagem de casamentos aceitos como verdadeiros, 'modo n' e 'modo f' em azul, e Sift original em vermelho.

Estes gráficos mostram que o 'modo n' e 'modo f', em azul, superam a taxa de acertos de Sift, em vermelho.

5.2.7. Análise da avaliação da etapa de filtragem de casamentos

Nesta avaliação testamos o desempenho das filtragens baseadas em coerência geométrica, nos modos 'n' e 'f', e comparamos para o algoritmo original Sift. Consideramos como métricas as taxas de erro e acertos (taxa de erros é igual a um menos taxa de acertos). Foram utilizadas 48 imagens da base de Mikolajczyk. Testamos dois limiares (L1 e L2) utilizados para determinar o intervalo de tolerância dos fatores de escala e ângulo de rotação (ver Cap.4). Os resultados desta avaliação sugerem que:

- Em todos os casos (exceto num único caso) os modos 'n' e 'f' melhoram o desempenho do Sift. Os modos 'n' e 'f' apresentam menores taxas de erro.

- O 'modo n' na maioria dos casos apresenta uma taxa de erro ligeiramente menor do que o 'modo f'.
- O 'modo n' apresenta dificuldades quando a taxa de aceitação é baixa. Isto é problemático na prática, já que é justamente nesta região que Sift é utilizado. Esta limitação é superada no 'modo f'. Isto sugere que o melhor na prática é usar o 'modo f'.
- O incremento no tempo de execução do Sift, causado pelas filtragem, depende do número N de casamentos aceitos. No pior caso, teve incremento de 31,0% para $N=1926$ casamentos.
- Nas sequências com variação de ponto de vista, Sift apresentou quatro taxas de erro muito elevadas (54,31%, 91,03%, 95,28%, e 75,53%). Só no primeiro caso esta taxa diminuiu significativamente usando filtragem geométrica (de 54,31% para 9,48% ou para 9,91%). Isto indica que Sift, mesmo com filtragem geométrica, não é robusto contra grandes variações de ponto de vista.
- Em alguns casos, as taxas de erros diminuíram significativamente de Sift para a filtragem geométrica. Por exemplo, de 46% para 18%; 41% para 13%; 21% para 1,5%; 16% para 0,2%; e 54% para 9,4%.

6. CONCLUSÕES

6.1. Conclusões gerais

Nesta tese, apresentamos duas propostas:

- Aciratefi, um algoritmo para o casamento de modelos. Os dados experimentais indicam que Aciratefi supera o algoritmo estado da arte Asift na combinação das distorções causadas por: mudança de ponto-de-vista, mudança de brilho/contraste, e presença de padrões repetitivos.
- Uma etapa de filtragem para casamento de pontos-chaves, denominada filtragem geométrica, que conseguiu melhorar o desempenho do algoritmo Sift.

6.2. Comentários finais

6.2.1. Comentários sobre Aciratefi

Os resultados obtidos com Aciratefi foram comparados com os resultados de Asift e Aforapro. Asift constitui atualmente o melhor método para casamento de pontos-chaves invariante a mudanças de ponto de vista. Por outro lado, Aforapro foi proposto em nossa dissertação de mestrado. Os resultados sugerem que: (a) Aciratefi supera Asift na presença de mudanças de brilho/contraste e de padrões repetitivos; (b) Aciratefi supera a Aforapro na facilidade de ajustar os parâmetros. Enquanto no Aforapro o usuário precisa ajustar manualmente vários parâmetros para cada aplicação (e.g. relações de escala, número de círculos, precisão, separação, etc.), Aciratefi utiliza parâmetros pré-definidos e consegue bom desempenho na maioria dos casos. No entanto, Aciratefi e Aforapro são mais lentos do que Asift. Nos experimentos, Asift foi pelo menos três vezes mais rápido que Aciratefi e Aforapro.

6.2.2. Comentários sobre a filtragem geométrica

A filtragem baseada na coerência geométrica dos casamentos pode ser aplicada em algoritmos que fazem casamento de pontos-chaves, por exemplo, Sift ou Surf

(BAY, et. al., 2008). Nesta tese, todos os testes foram com o algoritmo Sift. O objetivo da nossa etapa de filtragem foi melhorar o desempenho do Sift, aumentando a taxa de acertos. Nossa proposta inclui duas alternativas denominadas 'modo n' e modo f'. O 'modo n' considera apenas a coerência geométrica dos casamentos. O 'modo f' considera uma combinação entre a coerência geométrica dos casamentos e a razão das distâncias Euclidianas. Realizamos testes comparativos com o Sift original, que permitem concluir:

- A filtragem geométrica supera a taxa de acertos do Sift original.
- O 'modo n' muitas vezes apresenta maior taxa de erro do que Sift original. Estas limitações do 'modo n' acontecem para pequenas taxas de aceitação, justamente na região onde Sift é utilizado na prática. O 'modo f' supera Sift original e as limitações do 'modo n', inclusive para pequenas taxas de aceitação. Isto torna o 'modo f' utilizável na prática.

6.3. Trabalhos Futuros

Os trabalhos futuros visando melhorar e/ou aproveitar as nossas propostas, Aciratefi e a filtragem geométrica, são:

- Explorar a utilização de submodelos como área de casamento no Aciratefi. Isto teoricamente melhoraria a invariância a pontos de vista, porém aumentaria o tempo de execução.
- Diminuir o tempo de execução do Aciratefi, analisando uma diminuição no número de imagens simuladas.
- Explorar áreas de aplicação adequadas para Aciratefi, visando expor suas potencialidades.
- Aplicar a filtragem geométrica para outros algoritmos de casamento de pontos-chaves, como Surf.
- Explorar áreas de aplicação da filtragem geométrica, objetivando destacar sua relevância como complemento de algoritmos de casamento de pontos-chaves.

REFERÊNCIAS

- ADELSON, H. et al. Pyramid methods in image processing. **RCA Engineer**, v. 29, n. 6, 1984.
- AGUADO, A.; MONTIEL, E.; NIXON, M. Invariant characterization of the Hough transform for pose estimation of arbitrary shapes. **Pattern Recognition**, v. 35, p. 1083–1097, 2002.
- ALVAREZ, L. et al. Affine Invariant Distance Using Multiscale Analysis. Springer Science, Business Media New York. **J Math Imaging Vis**. DOI 10.1007/s10851-015-0585-9. 2015.
- ALVAREZ, L.; MORALES, F. Affine morphological multi-scale analysis of corners and multiple junctions. **International Journal of Computer Vision**. v. 2, n. 25, p. 95-107. 1997.
- ARAÚJO, S; KIM, H. Ciratefi: an RST-invariant template matching with extension to color images. **Integrated Computer-Aided Engineering**, v. 18, n. 1, p. 75-90, 2011.
- ARAÚJO, S. Casamento de padrões em imagens digitais livre de segmentação e invariante sub transformações de similaridade. 2009, 142p. **Tese (Doutorado)**. EPUSP, 2009.
- BALLARD, D. Generalizing the Hough transform to detect arbitrary shapes. **Pattern Recognition**. v. 13, n. 2, p. 111–122, 1981.
- BAUMBERG, A. Reliable feature matching across widely separated views. In: Conference on Computer Vision and Pattern Recognition, **Proceedings...** Hilton Head Island, South Carolina, USA, p. 774-781, 2000.
- BAY, H.; TUYTELAARS, T; GOOL, L. Surf: Speeded up robust features. In: **European Conference on Computer Vision**, p. 404–417, 2006.
- BAY, H.; ESS, A.; TUYTELAARS, T; GOOL, L. Speeded-Up Robust Features (SURF). **Computer Vision and Image Understanding**, v. 110, n. 3, p. 346-359, 2008. ISSN 1077- 3142.
- BERGOUNIOUX, M. (ed.). Mathematical Image Processing. In: **Springer Proceedings in Mathematics 5**, DOI 10.1007/978-3-642-19604-1 5, Springer-Verlag Berlin Heidelberg, 2011.
- BRADSKI, G; KAEHLER, A. **OpenCV, O'Reilly Media**. Sebastopol, CA: O'Reilly, 2008. ISBN 978-0596516130.

- BRAHMBHATT, S. **Practical OpenCV**. S.l.: Technology in Action, 2014. Apress Publisher. ISBN 9781430260790. (2014).
- BRUNELLI, R. **Template Matching Techniques in Computer Vision: Theory and Practice**. John Wiley & Sons, 2009. ISBN: 978-0-470-51706-2.
- CAO, F.; et al. **Securing SIFT with a Contraries Techniques**. Springer-Verlag Berlin Heidelberg. ISBN: 978-3-540-68480-0. (2008).
- CAO, S.; et al. An edge-based scale- and affine-invariant algorithm for remote sensing image registration, **International Journal of Remote Sensing**, v. 34, n. 7, p. 2301-2326, 2013.
- CHEN, L. **Digital and discrete geometry, theory and algorithms**. s.l.: Springer, 2015. ISBN: 978-3-319-12098-0.
- CHEN, M; SHAO, Z; LI, D; LIU, J. Invariant matching method for different viewpoint angle images. *Applied Optics*. vol. 52, No. 1/1. (2013).
- CYGANEK, B. **Object Detection and Recognition in Digital Images: Theory and Practice**, J. Wiley, ISBN: 978-1-118-61836-3. (2013).
- DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. *Proc. Computer Vision and Pattern Recognition (CVPR'05)*, v. 1, p. 886-893, 2005.
- DELLINGER, F. et al. SAR-SIFT: A SIFT-Like Algorithm for SAR Images. **IEEE Transactions On Geoscience And Remote Sensing**, v. 53, n. 1, 2015.
- DI RUBERTO, C. Generalized Hough transform for shape matching. **International Journal of Computer Applications**, v. 0975–8887, 2012.
- DOUBEK, P.; MATAS, J.; PERDOCH, M.; CHUM, O. Image Matching and Retrieval by Repetitive Patterns. *ICPR*. p. 3195-3198. (2010).
- ECABERT, O.; THIRAN, J. Adaptive Hough Transform for the Detection of Natural Shapes Under Weak Affine Transformations. **Pattern Recognition Letters**, v. 25, n. 12, p. 1411–1419, 2004.
- ETHAN, R.; RABAUD, V.; KONOLIGE, K.; BRADSKI, G. ORB: An Efficient Alternative to SIFT or SURF. *ICCV11*, **Proceedings of the International Conference on Computer Vision**. (2011).
- FAN, B. et al.: Registration of optical and SAR satellite images by exploring the spatial relationship of the improved SIFT. **IEEE Geosci. Remote Sens. Lett.**, v. 10, n. 4, p. 657–661, 2013.
- FISHER, R; DAWSON-HOWE, K; FITZGIBBON, A; ROBERTSON, C; TRUCCO, E **Dictionary of Computer Vision and Image Processing**. 2nd ed. John Wiley and Sons, ISBN: 0-470-01526-8. 2014.

- FISCHLER, M.; BOLLES, R. Random Sample Consensus: a Paradigm for Model Fitting with Application to Image Analysis and Automated Cartography. **Communications of the ACM**, v. 24, n. 6, p. 381-395, 1981.
- FLUSSER, J.; SUK, T. Pattern Recognition by Affine Moment Invariants. **Pattern Recognition**. v. 26, n. 1, p. 167–174, 1993.
- FUNT, B. V.; BARNARD, K.; MARTIN, L. Is machine colour constancy good enough?, In: **Proceedings of the 5th European Conference on Computer Vision, LNCS**, v. 1406, p. 445-459, 1998.
- GAO, K.; ZHANG, Y.; ZHANG, W.; LIN, S. Affine stable characteristic based sample expansion for object detection. In: **CIVR'10. Xi'an China**. 2010.
- GONZALES, R.; WOODS, R. **Digital Images Processing**. 2nd ed. New Jersey Prentice Hall, 2002. Upper Saddle River, ISBN 0201180758.
- GRAUMAN, K.; LEIBE, B. **Visual Object Recognition**. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool, 2011. ISBN 9781598299694.
- HARTLEY, R.; ZISSERMAN, A. **Multiple View Geometry in Computer Vision**. 2nd ed. New York: Cambridge University Press, 2004. ISBN-13: 978-0-521-54051-3.
- HARRIS, C; STEPHENS, M. A combined corner and edge detector. In: **Proceedings from the Fourth Alvey Vision Conference**. v. 15, p. 147–151, 1988.
- HASSNE, T; MAYZELS, V; ZELNIK-MANOR, L. On sifts and their scales. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**, p. 1522–1528, 2012.
- HASSANEIN, A. et al. A Survey on Hough Transform, Theory, Techniques and Applications. arXiv preprint arXiv:1502.02160. (2015).
- HAYS, J.; LEORDEANU, M.; EFROS, A.; LIU, Y. Discovering texture regularity as a higher-order correspondence problem, In: Proceedings 9th European Conference Comput. Vis., p. 522–535, 2006.
- HOUGH, P. Method and Means for Recognizing Complex Patterns. **US Patent, 3069654**, 1962.
- HSIAO, J.; CHEN, C.; CHIEN, L.; CHEN, M. A new approach to image copy detection based on extended feature sets, **IEEE Transactions on Image Processing**. v. 16, n. 8, p. 2069–2079, 2007.
- HSIANG, T.-R. et al. A MapReduce-based Indoor Visual Localization System Using Affine Invariant Features. **Computers and Electrical Engineering**, v. 39, p. 2369–2378, 2013.

- HU, M. K. Visual Pattern Recognition by Moment Invariants. **IEEE Transactions in Information Theory**. v. 8, p 179-187, 1962.
- HUO, C.; PAN, C.; HUO, L.; ZHOU, Z. Multilevel SIFT matching for large-size VHR image registration. **IEEE Geosci. Remote Sens. Lett.**, v. 9, n. 2, p. 171–175, 2012.
- INTEL. OpenCV - Open Source Computer Vision Library. 2014. Disponível em: <<http://www.intel.com/technology/computing/opencv/>>. Acesso em: fev 2014.
- JIANG, H.; YU, S. Linear Solution to Scale and Rotation Invariant Object Matching. **Proc. IEEE Conf. Computer Vision and Pattern Recognition**. 2009.
- JU, H.; KAI-KUANG, M. Rotation-invariant and scale-invariant Gabor feature for texture image retrieval. **Image and Vision Computing**. Nayang Singapore. v. 25, n. 9, p. 1474-1481, 2007.
- KADIR, T.; ZISSERMAN, A.; BRADY, M. An affine invariant salient region detector. In: **Proceedings of the 8th European Conference on Computer Vision**. Prague, Czech Republic. p. 345–457, 2004.
- KIM, H. *ProEikon - Rotinas e programas em C++ para processamento de imagens e visão computacional*. São Paulo, 2009. Disponível em: <<http://www.lps.usp.br/~hae/software>>. Acesso em: jan 2015a.
- KIM, HAE: *Cekeikon - Rotinas e programas em C++ para processamento de imagens e visão computacional*. São Paulo, 2009. Disponível em: <<http://www.lps.usp.br/~hae/software>>. Acesso em: jan, 2015b.
- KIM, H. FORAPRO: Rotation-Discriminating Template Matching Based on Fourier Coefficients of Radial Projections with Robustness to Scaling and Partial Occlusion. **Pattern Recognition**. v. 43, n. 3, p. 859-872, 2010.
- KIM, H.; ARAÚJO, S. Grayscale template-matching invariant to rotation, scale, translation, brightness and contrast. In: IEEE Pacific-Rim Symposium on Image and Video Technology, **Lecture Notes in Computer Science**, v. 4872, p. 100-113, 2007.
- KIM, H.; PÉREZ, G. Reconhecimento de Padrões Invariante Afim Baseado nos Coeficientes de Fourier de Projeções Radiais. In: **VI Workshop de Visão Computacional**, UNESP Campus de Presidente Prudente – Brasil. 2010.
- KLINGER, A. Pattern and Search Statistics. In: Rustagi J.S. (ed.). **Optimizing Methods in Statistics**. New York: Academic Press, 1971.
- LE BRESE, C.; ZOU, J.; UY, B. An Improved Asift Algorithm for Matching Repeated Patterns. In: **Proceedings of 2010 IEEE 17th. International Conference on Image Processing**. 2010.

- LEUNG, T.; MALIK, J. DETECTING, localizing and grouping repeated scene elements from an image. In: **Proc. 4th Eur. Conf. Comput.** p. 546–555, 1996.
- LI, Q.; WANG, G.; LIU, J.; CHEN, S. Robust scale-invariant feature matching for remote sensing image registration. **IEEE Geosci. Remote Sens. Lett.**, v. 6, n. 2, p. 287–291, 2009.
- LIN, W.; LIU, Y. A lattice-based MRF model for dynamic near-regular texture tracking. **IEEE Trans. Pattern Anal. Mach. Intell.**, v. 29, n. 5, p. 777–792, 2007.
- LINDE, O.; LINDBERG, T. Object recognition using composed receptive field histograms of higher dimensionality. In: **Proceedings 17th International Conference on Pattern Recognition (ICPR'04)**. 2004.
- LINDE, O.; LINDBERG, T. Composed complex-cue histograms: An investigation of the information content in receptive field based image descriptors for object recognition. **Computer Vision and Image Understanding**, v. 116, p. 538-560. 2012.
- LINDBERG T; BRETZNER, L. Real-time scale selection in hybrid multi-scale representations. In: **Proceedings Scale-Space'03, Springer Lecture Notes in Computer Science**, v.2695, p. 148–163, 2003.
- LINDBERG, T. Image matching using generalized scale-space interest points. In: **Proceedings of International Conference on Scale-Space and Variational Methods for Computer Vision (SSVM 2013), Lecture Notes in Computer Science**, v. 7893, p. 355–367. Springer, 2013.
- LINDBERG, T. Image matching using generalized scale-space interest points. **J Math Imaging Vis**, v. 52, p. 3–36, 2015. DOI 10.1007/s10851-014-0541-0.
- LINDBERG, T.; GARDING, J. Shape-adapted smoothing in estimation of 3-D shape cues from affine deformations of local 2-D brightness structure. **Image and Vision Computing**, v. 15, n. 6, p. 415-434, 1997.
- LIU, X; TIAN, Z; LU, Q; YANG, L; CHAI, C. A new affine invariant descriptor framework in shearlets domain for SAR image multi scale registration. **International Journal of Electronics and Communications (AEÜ)**. China, 2013.
- LOWE, D. Object Recognition from Local Scale-Invariant Features. In: **Proceedings of the International Conference on Computer Vision**, p. 1150-1157, 1999.
- LOWE, D. Distinctive image features from scale invariant keypoints, **International Journal on Computer Vision**, v. 60, n. 2, p. 91-110, 2004. ISSN 1573-1405.
- LOWE, D. **Demo Software:** Sift keypoints detector. Vancouver, Canada, 2005. Disponível em: <<http://www.cs.ubc.ca/~lowe/keypoints/>>. Acesso em: Jan 2015.

- MA, X. et al. A fast affine-invariant features for image stitching under large viewpoint changes. **Neurocomputing**, Elsevier, v. 151, p. 1430–1438, 2015.
- MAJI, P; PAUL, S. **Scalable Pattern Recognition Algorithms**, Springer, 2014. ISBN: 978-3-319-05629-6.
- MALLAT, S. G. A Theory of Multiresolution Signal Processing: the Wavelet Representation. **IEEE Transactions on Pattern Analysis and Machine Intelligence**. v. PAMI-11, n. 6, p 674-693, 1989.
- MATAS, J; CHUM, O; Urban, M; Pajdla, T. Robust Wide-baseline Stereo from Maximally Stable Extremal Regions. **Image Vis. Computer**. v. 22, p. 761–767, 2004.
- MIKOLAJCZYK, K. **Images dataset**. Disponível em: <<http://lear.inrialpes.fr/people/mikolajczyk/Database/>>. Acesso em: jan. 2015.
- MIKOLAJCZYK, K.; SCHMID, C. An affine invariant interest point detector. In: **European Conference on Computer Vision (ECCV)**. Copenhagen, Denmark. p. 128-142, 2002.
- MIKOLAJCZYK, K.; SCHMID, C. Scale and Affine Invariant Interest Point Detectors. **International Journal of Computer Vision**, Springer Netherlands. v. 60, n. 1, p. 63-86, oct 2004.
- MIKOLAJCZYK, K; SCHMID, C. A Performance Evaluation of Local Descriptors. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 27, n. 10, p. 1615-1630, 2005. ISSN 0162-8828.
- MOESLUND, T. **Introduction to Video and Image Processing**, Springer, 2012. ISBN: 978-1-4471-2502-0.
- MOLTON, N; DAVISON, A; REID, I. Locally planar patch features for real-time structure from motion. In: **Proc. British Machine Vision Conference (BMVC)**. 2004
- MOREL, J. M.; YU, G. ASIFT: A new framework for fully affine invariant image comparison. **SIAM Journal on Image Sciences**, v. 2, n. 2, 2009.
- MOREL, J. M.; YU, G. Is Sift Scale Invariant? **Inverse Problems and Imaging**, v. 5-1, p 115-136, 2010. ISSN 19308337.
- MOREL, J. M.; YU, G. **Images dataset**. Disponível em: <<http://www.cmap.polytechnique.fr/~yu/research/ASIFT/>>. Acesso em: jan 2014.
- NIXON, M; AGUADO, A. **Feature Extraction and Image Processing**. 3rd ed. Newnes, ISBN: 0750650788. 2012.

- NOURY, N; SUR, F; BERGER, M. How to Overcome Perceptual Aliasing in Asift. **6th International symposium on visual computing**. ISVC 2010. Inria. 2010.
- OUYANG, P. et al. A FAST Extreme Illumination Robust Feature in Affine Space. **22nd International Conference on Pattern Recognition. IEEE Computer Society**. p. 2365-2370, 2014. DOI: 10.1109/ICPR.2014.410.
- PANG, YANWEI et al. Fully Affine Invariant SURF for Image Matching. **Neurocomputing**, v. 85, p. 6–10, 2012.
- PARK, M.; BROCKLEHURST, K.; COLLINS, R.; LIU, Y. Deformed lattice detection in real-world images using mean-shift belief propagation, **IEEE Trans. Pattern Anal. Mach. Intell**, v. 31, n. 10, p. 1804–1816, 2009.
- PEREZ, G. **AFORAPRO**: Reconhecimento de imagens invariante sob transformações afins. 2011. Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo (2011). Disponível em: <<http://www.teses.usp.br/teses/disponiveis/3/3142/tde-31052011-155411/>>. Acesso em: 2015-06-29.
- PEREZ, G.; KIM, H. Y. Novo Algoritmo para Reconhecimento de Objetos Invariante Afim. In: **Simpósio Brasileiro de Telecomunicações**, 2011, Curitiba. SBRT 2011,
- PÉREZ, G.; KIM, H.: Comparison of viewpoint invariant template matchings. IEEE Computer Society, **International Telecommunication Symposiums ITS**, 2014.
- PETROU, M.; PETROU, C. **Image Processing: The Fundamentals**. 2nd ed. John Wiley & Sons, 2010. ISBN: 978-0-470-74586-1.
- PERWASS, C. **Geometric Algebra with Applications in Engineering**, Springer, 2009. ISBN: 978-3-540-89067-6.
- PODBREZNIK, P.; POTOČNIK, B. Estimating correspondence between arbitrarily selected points in two widely-separated views. **Advanced Engineering Informatics**. v. 24, n. 3, p. 367-366, 2010.
- PODBREZNIK, P.; POTOČNIK, B. A self-adaptive ASIFT-SH method. **Adv. Eng. Inform**, v. 27, n. 1, p. 120–130, 2013.
- PRATT, W. **Digital image processing**. 4th ed. Los Altos, California, Wiley-interscience, 2007. ISBN: 978-0-471-76777-0.
- RABIN, J.; DELON, J.; GOUSSEAU, Y. A statistical approach to the matching of local features. **SIAM Journal on Imaging Sciences**. v. 2, p 931-958, 2009.
- RAMESH, Y.; PRIYA, J. Illumination Invariant Iterative Based Image Matching. **International Journal of Review in Electronics & Communication Engineering (IJRECE)**, v. 1, n. 5, 2013. ISSN: 2321-3159.

- RASMUSSEN, C; WILLIAMS, C. **Gaussian Processes for Machine Learning**, MIT Press, 2006. ISBN-13: 978-0-262-18253-9.
- REY-OTERO, I.; DELBRACIO, M. Anatomy of the SIFT Method. **Image Processing On Line**, v. 4, p. 370-396, 2014. ISSN 2105-1232.
- REZA, O. An Automatic Algorithm for Object Recognition and Detection Based on Asift Keypoints. **Signal & Image Processing: An International Journal (SIPIJ)**, n. 5, 2012.
- RUBNER, Y.; TOMASI, C. Texture-Based Image Retrieval Without Segmentation. **Proc. ICCV**. p. 1018-1024, 1999.
- SCHAEFER, G. How useful are colour invariants for image retrieval, In: **Proceedings of 2nd Int. Conference on Computer Vision and Graphics**, 2004.
- SCHAFFALITZKY, F.; ZISSERMAN, A. Geometric grouping of repeated elements within images, in Shape, Contour and Grouping in Computer Vision, **Lecture Notes in Computer Science**, v. 1681, p. 165–181, 1999.
- SCHAFFALITZKY, F.; ZISSERMAN, A. Viewpoint Invariant Texture Matching and Wide Baseline Stereo, **Proc. ICCV**. v. 2, p. 636-643, 2001.
- SCHWIND, P. et al. Applicability of the SIFT operator to geometric SAR image registration. **Int. J. Remote Sci.**, v. 31, n. 8, p. 1959–1980, 2010.
- SCOTT, K. **Computer Vision Metrics: Survey, Taxonomy, and Analysis**. Apress, 2014. ISBN: 978-1-4302-5930-5.
- SHECHTMAN, E.; IRANI, M. Matching local Self-Similarities across images and videos. In: **Proceedings of CVPR**. 2007.
- SHIN, D.; TJAHJADI, T. Clique descriptor of affine invariant regions for robust wide baseline image matching. **Pattern Recognition**, v. 43, p. 3261-3272, 2010.
- SWAIN, M.; BALLARD, D. Color indexing. **International Journal of Computer Vision**, v. 7, n. 1, p. 11-32, 1991. DOI:10.1007.
- SZELISKI, R. **Computer Vision: Algorithms and Applications**. Springer, 2011. ISBN: 978-1-84882-934-3.
- TAM, P. **Object recognition**. Croatia: Intech publisher, 2011. ISBN 978953-3072227.
- TAO, C. et al. Airport detection from large IKONOS images using clustered SIFT keypoints and region information. **IEEE Geosci. Remote Sens. Lett.**, v. 8, n. 1, p. 128–132, 2011.

- TEAGUE, M. R. Image Analysis via the General Theory of Moments. **Journal of Optical Society of America**. v. 70, n. 8, p. 920-930, 1980.
- TELL, D.; CARLSSON, S. Combining Appearance and Topology for Wide Baseline Matching. In: **Proceedings of ECCV**, p. 68-81, 2002.
- THEODORIDIS, S.; KOUTROUMBAS, K. **Pattern Recognition**. 4th ed. San Diego, California, USA, 2009. ISBN: 978-1-59749-272-0.
- TONG, G.; MENG, X.; YE, N. A GPU-based Affine and Scale Invariant Feature Transform Algorithm. **Journal of Information & Computational Science**. v. 10, n. 13, p. 3991-3998, 2013.
- TREIBER, M. **An Introduction to Object Recognition**. London Dordrecht Heidelberg, New York: Springer, 2010. ISBN 978-1-84996-234-6.
- TUYTELAARS, T.; GOOL, L. J. Wide baseline stereo matching based on local, affinity invariant regions. In: **Proceedings of the British Machine Vision Conference**. p. 42–56, 2000.
- VAN GOOL; MOONS, T.; UNGUREANU, D. Affine/Photometric Invariants for Planar Intensity Patterns. **Proceedings of European Conference on Computer Vision**. v. 1, p. 642–651, 1996.
- WANG, X. et. al. An ASIFT-Based Local Registration Method for Satellite Imagery. **Remote Sens**. 2015. ISSN 2072-4292.
- WANG, Y. et al. A Novel Form of Affine Moment Invariants of Grayscale Images. **Elektronika Ir Elektrotechnika**, v. 19, n. 1, 2013. ISSN 1392-1215.
- WEI, Z.; KOSECKA, J. Generalized RANSAC Framework for Relaxed Correspondence Problems. In: **3D Data Processing, Visualization, and Transmission, 3rd International Symposium on**, Chapel Hill, NC, p. 854-860, 2006.
- WENDEL, A.; DONOSER, M.; BISCHOF, H. Unsupervised Facade Segmentation Using Repetitive Patterns. Lecture notes in computer science. **Pattern Recognition**. v. 6376, p. 51-60, 2010.
- WU, J. et al. A Comparative Study of Sift and its Variants. **Measurement Science Review**, v. 13, n. 3, 2013.
- WU, W.; YANG, J. Object fingerprints for content analysis with applications to street landmark localization. **ACM Multimedia**. 2008.
- XU, W.; HUANG, X.; LIU, Y.; ZHANG, W. A local characteristic scale selection method based on Gabor wavelets, **J. Image Graph**, v. 16, n. 1, p. 72–78, 2011.

YANG, X.; WANG, M. Seamless image stitching method based on ASIFT, **Comput. Eng**, v. 39, n. 2, p. 241–244, 2013.

YANG, Y.; NEWSAM, C. Geographic image retrieval using local invariant features. **IEEE Trans. Geosci. Remote Sens.**, v. 51, n. 2, p. 818–832, 2013.

YU, G.; MOREL, J.M. Asift: an algorithm for fully affine invariant comparison. **Image Processing On Line**, 2011.

ZHANG, Q. et al. Registration of images with affine geometric distortion based on Maximally Stable Extremal Regions and phase congruency. **Image and Vision Computing**, v. 36, p. 23–39, 2015.