



Pós-Graduação em Ciência da Computação

**LUIZ GUSTAVO DA ROCHA CHARAMBA**

**DESCRITOR DE FORMA PLANAR ROBUSTO A  
PROJETIVIDADES BASEADO EM VETORES DE RAZÃO  
CRUZADA**



Universidade Federal de Pernambuco

[posgraduacao@cin.ufpe.br](mailto:posgraduacao@cin.ufpe.br)

[www.cin.ufpe.br/~posgraduacao](http://www.cin.ufpe.br/~posgraduacao)

Recife

2018

LUIZ GUSTAVO DA ROCHA CHARAMBA

DESCRITOR DE FORMA PLANAR ROBUSTO A  
PROJETIVIDADES BASEADO EM VETORES DE RAZÃO  
CRUZADA

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

**Área de Concentração:** Visão Computacional  
**Orientador:** Prof. Silvio de Barros Melo

Recife  
2018

Catálogo na fonte  
Bibliotecária Arabelly Ascoli CRB 4-2068

C469d Charamba, Luiz Gustavo da Rocha  
Descritor de forma planar robusto a projetividades baseado em  
vetores de razão cruzada / Luiz Gustavo da Rocha Charamba –  
2018.  
156 f.: fig., tab.

Orientador: Silvio de Barros Melo.  
Dissertação (Mestrado) – Universidade Federal de  
Pernambuco. Cin. Ciência da Computação. Recife, 2018.  
Inclui referências.

1. Visão computacional. 2. Inteligência artificial. 3.  
Reconhecimento de forma. I. Melo, Silvio de Barros (orientador)  
II. Título.

006.37 CDD (22. ed.) UFPE-MEI 2019-01

**Luiz Gustavo da Rocha Charamba**

**Descritor de Forma Planar Robusto a Projetividades Baseado  
em Vetores de Razão Cruzada**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Aprovado em: 28/08/2018.

**BANCA EXAMINADORA**

---

Prof. Dr. Tsang Ing Ren  
Centro de Informática / UFPE

---

Prof. Dr. João Paulo Silva do Monte Lima  
Departamento de Computação / UFRPE

---

Prof. Dr. Carlos Alexandre Barros de Mello  
Centro de Informática / UFPE

---

Prof. Dr. Silvio de Barros Melo  
Centro de Informática / UFPE

---

## AGRADECIMENTOS

---

Agradeço a Deus que iluminou o meu caminho durante esta jornada acadêmica com força e perseverança.

À minha mãe querida, Maria das Graças da Rocha, que com sua sabedoria me ensinou a enfrentar os desafios da vida, além de torcer por mim em cada obstáculo que enfrento e que sem ela eu não teria nada, obrigado pelo seu amor.

Agradeço a minha amada noiva, Raissa Duarte, pelo seu amor, apoio e compreensão nessa trajetória difícil, tornando o caminho mais doce com sua agradável companhia.

Sou muito grato ao meu orientador, o professor Silvio Melo, na qual eu nutro grande admiração e respeito, por ter confiado na minha capacidade e ter me dado a honra de ser seu aluno de mestrado.

À aluna de iniciação científica, Ullayne Lima, que me ajudou na geração das imagens, na execução e análise dos experimentos.

Gostaria de agradecer também aos colegas do CIn, que me ajudaram de uma forma ou de outra, com discussões e dicas sobre o meu trabalho, principalmente aqueles que se alegram com o sucesso dos outros.

Aos meus amigos próximos e familiares que torcem por mim, meus sinceros agradecimentos.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pela bolsa de estudos à mim providenciada, sou muito grato por esse amparo e espero retribuir à sociedade com este trabalho.

Por fim, agradeço a todas as pessoas que por diferentes razões, contribuíram para a realização desta dissertação.

*“O horizonte está nos olhos e não na realidade”.*

(GANIVET, 1893)

---

## RESUMO

---

Reconhecimento de objetos planares tais como caracteres, símbolos e logomarcas é um problema difícil em visão computacional: a possibilidade dos objetos sofrerem uma perspectiva, um tipo de deformação projetiva, é um fator que dificulta ainda mais o processo de reconhecimento. Neste trabalho, um novo descritor de forma planar robusto a transformações projetivas é proposto. O descritor se baseia em representar objetos planares num conjunto de feixes de raios que se interceptam com o objeto gerando pontos colineares de alta frequência, onde é possível calcular valores de razão cruzada, um invariante projetivo, e armazená-los em vetores que são usados durante a comparação entre elementos dos objetos planares descritos. Foram realizados experimentos com caracteres e símbolos mais complexos sob efeito de transformações projetivas variadas, onde o descritor proposto se mostrou eficiente no reconhecimento correto dessas formas. Também foram realizados testes com oclusão e deformações geométricas não lineares em cenas reais, mostrando-se também resistente para essas situações.

**Palavras-chave**—Descritor de forma. Reconhecimento de forma. Invariante projetivo. Razão cruzada.

---

## ABSTRACT

---

Recognizing planar objects such as characters, symbols and logos is a difficult problem in computer vision: the possibility of objects suffering a perspectivity, a type of projective deformation, is a factor that makes the recognition process even more difficult. In this work, a new descriptor of planar forms that is robust to projective transformations is proposed. The descriptor is based on representing planar objects with a set of ray beams that intersect with the object generating high-frequency collinear points, where it is possible to calculate values of *cross-ratio*, which is a projective invariant, and storing them in arrays which are used during the comparison of the described planar objects features. Experiments were carried out with more complex characters and symbols under the effect of varied projective, where the proposed descriptor was proved efficient in the correct recognition of these forms. In addition to these deformations, tests were also performed with occlusion and nonlinear geometric deformations in real scenes, showing its robustness to such cases.

**Keywords**—Shape descriptor. Shape recognition. Projective invariant. Cross ratio.

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>12</b>
1.1	MOTIVAÇÃO E OBJETIVOS . . . . .	12
1.2	CONTRIBUIÇÕES . . . . .	16
1.3	ORGANIZAÇÃO DA DISSERTAÇÃO . . . . .	17
<b>2</b>	<b>VISÃO GERAL DO RECONHECIMENTO DE PADRÕES . . . . .</b>	<b>19</b>
2.1	ETAPAS DO RECONHECIMENTO DE PADRÕES EM IMAGENS . . . . .	19
<b>2.1.1</b>	<b>Aquisição da imagem . . . . .</b>	<b>20</b>
<b>2.1.2</b>	<b>Pré-processamento . . . . .</b>	<b>20</b>
<b>2.1.3</b>	<b>Segmentação. . . . .</b>	<b>20</b>
<b>2.1.4</b>	<b>Extração de características . . . . .</b>	<b>21</b>
<b>2.1.5</b>	<b>Classificação . . . . .</b>	<b>21</b>
2.2	EXTRAÇÃO E RECONHECIMENTO DE FORMAS PLANARES . . . . .	22
2.3	FLUXO BÁSICO DE RECONHECIMENTO . . . . .	25
2.4	DISCUSSÃO . . . . .	26
<b>3</b>	<b>FUNDAMENTOS BÁSICOS DE GEOMETRIA PROJETIVA NO PLANO . . . . .</b>	<b>27</b>
3.1	O ESPAÇO PROJETIVO . . . . .	28
3.2	UM MODELO PARA O PLANO PROJETIVO . . . . .	29
3.3	COORDENADAS HOMOGÊNEAS. . . . .	29
<b>3.3.1</b>	<b>Pontos finitos e ideais em <math>\mathbb{P}^2</math> . . . . .</b>	<b>29</b>
<b>3.3.2</b>	<b>Retas em <math>\mathbb{P}^2</math> . . . . .</b>	<b>30</b>
3.4	TRANSFORMAÇÕES PROJETIVAS . . . . .	31
3.5	HIERARQUIA DE TRANSFORMAÇÕES GEOMÉTRICAS . . . . .	32
<b>3.5.1</b>	<b>Classe I: Isometrias . . . . .</b>	<b>32</b>
<b>3.5.2</b>	<b>Classe II: Similaridades . . . . .</b>	<b>33</b>
<b>3.5.3</b>	<b>Classe III: Afinidades . . . . .</b>	<b>33</b>
<b>3.5.4</b>	<b>Classe IV: Projetividades. . . . .</b>	<b>34</b>
3.6	INVARIANTES PROJETIVOS. . . . .	35
<b>3.6.1</b>	<b>Razão Cruzada. . . . .</b>	<b>35</b>
<b>3.6.2</b>	<b>Razão Cruzada de cinco pontos coplanares . . . . .</b>	<b>37</b>
3.7	DISCUSSÃO . . . . .	38
<b>4</b>	<b>FUNÇÕES DE DISTÂNCIA, CORRELAÇÃO E <i>DYNAMIC TIME WARPING</i> . . . . .</b>	<b>39</b>
4.1	FUNÇÕES DE DISTÂNCIA. . . . .	39

4.2	CORRELAÇÃO . . . . .	41
<b>4.2.1</b>	<b>Coefficiente de Pearson.</b> . . . . .	<b>41</b>
4.3	<i>DYNAMIC TIME WARPING</i> . . . . .	41
<b>4.3.1</b>	<b>O Algoritmo</b> . . . . .	<b>42</b>
<b>4.3.2</b>	<b>Condições de Alinhamento</b> . . . . .	<b>42</b>
<b>4.3.3</b>	<b>Um Exemplo Prático</b> . . . . .	<b>43</b>
4.4	DISCUSSÃO . . . . .	46
<b>5</b>	<b>TRABALHOS RELACIONADOS</b> . . . . .	<b>47</b>
5.1	DESCRITORES DE FORMA INVARIANTES PROJETIVOS . . . . .	50
<b>5.1.1</b>	<b><i>Cross Ratio Spectrum</i></b> . . . . .	<b>51</b>
<b>5.1.2</b>	<b><i>Hierarchical Characteristic Number Contexts</i></b> . . . . .	<b>52</b>
5.2	DISCUSSÃO . . . . .	53
<b>6</b>	<b>EXTRAÇÃO DE CARACTERÍSTICAS E ESTRUTURA DO DESCRITOR DE FORMA PROPOSTO</b> . . . . .	<b>54</b>
6.1	ELEMENTOS E ESTRUTURA DO DESCRITOR . . . . .	54
<b>6.1.1</b>	<b>Vértices do Invólucro Convexo</b> . . . . .	<b>54</b>
<b>6.1.2</b>	<b>Feixes e Raios</b> . . . . .	<b>55</b>
<b>6.1.3</b>	<b>Características Invariantes Projetivas Fundamentais</b> . . . . .	<b>55</b>
6.1.3.1	Pontos de Borda ( <i>Edge Points</i> ) . . . . .	55
6.1.3.2	Vetor de Razão Cruzada ( <i>Cross Ratio Vector</i> ) . . . . .	55
<b>6.1.4</b>	<b>Estrutura Hierárquica.</b> . . . . .	<b>55</b>
6.2	PROCESSO DE EXTRAÇÃO DE CARACTERÍSTICAS . . . . .	56
<b>6.2.1</b>	<b>Discussão Sobre as Etapas Preliminares</b> . . . . .	<b>57</b>
<b>6.2.2</b>	<b>Etapas da Extração de Características</b> . . . . .	<b>58</b>
6.2.2.1	Obtenção dos Vértices do Invólucro Convexo e Contorno . . . . .	58
6.2.2.2	Emissão dos Raios em Feixes Divergentes . . . . .	59
6.2.2.3	Geração de Pontos de Borda Colineares . . . . .	59
6.2.2.4	Cálculo dos Valores do Vetor de Razão Cruzada . . . . .	60
<b>6.2.3</b>	<b>Integração das Etapas</b> . . . . .	<b>62</b>
6.2.3.1	Algoritmo . . . . .	63
6.3	DISCUSSÃO . . . . .	64
<b>7</b>	<b>COMPARAÇÃO ENTRE DESCRITORES DE FORMA E CLASSIFICAÇÃO</b> . . . . .	<b>65</b>
7.1	COMPARAÇÃO ENTRE ELEMENTOS DOS DESCRITORES. . . . .	66
<b>7.1.1</b>	<b>Comparação Raio-a-Raio</b> . . . . .	<b>66</b>
7.1.1.1	Comparação entre Vetores de Razão Cruzada (vetor-a-vetor) . . . . .	66

<b>7.1.2</b>	<b>Comparação Vértice-a-Vértice . . . . .</b>	<b>68</b>
7.1.2.1	Comparação entre Feixes de Raios (Feixe-a-Feixe) . . . . .	68
7.1.2.1.1	<i>Formação de espectro de topologia de Raios correspondentes.</i> . . . . .	69
7.1.2.1.2	<i>Comparação entre Espectros de Topologia via DTW</i> . . . . .	71
7.1.2.1.3	<i>Comparação entre espectros de topologia com sinais pós-processados</i> . . . . .	73
7.2	<b>ANÁLISE DA DISPOSIÇÃO DOS VÉRTICES DAS ENVOLTÓRIAS . . . . .</b>	<b>75</b>
<b>7.2.1</b>	<b>Pré-ajuste dos índices de Vértices e remoção de falsos positivos. . . . .</b>	<b>75</b>
7.2.1.1	Ajuste de deslocamento circular dos índices . . . . .	76
7.2.1.2	Remoção de <i>outliers</i> e ajuste de linha por Mínimos Quadrados . . . . .	78
7.2.1.3	Remoção de falsos positivos pela razão cruzada de cinco pontos coplanares . . . . .	81
<b>7.2.2</b>	<b>Correlação entre índices de Vértices . . . . .</b>	<b>86</b>
7.2.2.1	Cálculo do Coeficiente de Correlação Linear de Pearson . . . . .	86
7.2.2.1.1	<i>Estudo de Caso I: Comparando objetos de mesma classe com correlação positiva</i> . . . . .	86
7.2.2.1.2	<i>Estudo de Caso II: Comparando objetos de mesma classe com correlação negativa</i> . . . . .	89
7.2.2.1.3	<i>Estudo de Caso III: Comparando objetos de diferentes classes e análise da correlação</i> . . . . .	91
7.2.2.2	Uso da Correlação como critério para eliminação. . . . .	93
7.3	<b>MÉTRICA DE DISTÂNCIA GLOBAL COMPOSTA . . . . .</b>	<b>95</b>
<b>7.3.1</b>	<b>Média das distâncias entre Vértices . . . . .</b>	<b>95</b>
<b>7.3.2</b>	<b>Percentuais de correspondências entre Vértices . . . . .</b>	<b>96</b>
<b>7.3.3</b>	<b>Composição das Métricas . . . . .</b>	<b>96</b>
7.3.3.1	Métrica composta para abordagem Simétrica . . . . .	97
7.3.3.2	Métrica composta para abordagem Assimétrica . . . . .	99
7.4	<b>CLASSIFICAÇÃO PARA OBJETOS SIMÉTRICOS E ASSIMÉTRICOS. . . . .</b>	<b>100</b>
<b>7.4.1</b>	<b>Classificação para objetos <i>template</i> simétricos . . . . .</b>	<b>100</b>
<b>7.4.2</b>	<b>Classificação para objetos <i>template</i> Assimétricos . . . . .</b>	<b>103</b>
7.5	<b>TESTES DE COMPARAÇÃO PARA <i>TEMPLATE</i> SIMÉTRICOS E ASSIMÉTRICOS . . . . .</b>	<b>105</b>
<b>7.5.1</b>	<b>Estudo de Caso I: Comparando objeto <i>template</i> simétrico com objeto de mesma classe. . . . .</b>	<b>105</b>
<b>7.5.2</b>	<b>Estudo de Caso II: Comparando Objeto <i>template</i> simétrico com objeto de classe diferente . . . . .</b>	<b>106</b>
<b>7.5.3</b>	<b>Estudo de Caso III: Comparando objeto <i>template</i> assimétrico com objeto de mesma classe . . . . .</b>	<b>107</b>
<b>7.5.4</b>	<b>Estudo de Caso IV: Comparando objeto <i>template</i> assimétrico com objeto de classe diferente . . . . .</b>	<b>108</b>

7.6	ABORDAGEM DE CLASSIFICAÇÃO POR ESTIMAÇÃO DE HOMO- GRAFIA E ERRO DE TRANSFERÊNCIA . . . . .	109
<b>7.6.1</b>	<b>Métrica de similaridade baseada no erro de transferência . . . . .</b>	<b>109</b>
7.7	DISCUSSÃO . . . . .	112
<b>8</b>	<b>EXPERIMENTOS E RESULTADOS . . . . .</b>	<b>113</b>
8.1	TESTES COM IMAGENS SINTÉTICAS POSSUINDO DEFORMAÇÕES PROJETIVAS VARIADAS . . . . .	114
<b>8.1.1</b>	<b>Testes Comparativos . . . . .</b>	<b>114</b>
8.1.1.1	Testes comparativos com Caracteres . . . . .	114
8.1.1.2	Testes comparativos com símbolos e logomarcas . . . . .	118
8.1.1.3	Testes comparativos com formas extraídas do <i>MPEG-7 dataset</i> . . . . .	121
8.1.1.4	Testes comparativos com deformações mais acentuadas . . . . .	125
<b>8.1.2</b>	<b>Testes individuais com oclusão . . . . .</b>	<b>127</b>
8.2	ANÁLISE DAS CORRESPONDÊNCIAS DOS SÍMBOLOS E LOGOMAR- CAS OBTIDOS DE CENAS REAIS. . . . .	132
<b>8.2.1</b>	<b>Teste I: Chuteira Adidas . . . . .</b>	<b>132</b>
<b>8.2.2</b>	<b>Teste II: Loja Puma . . . . .</b>	<b>133</b>
<b>8.2.3</b>	<b>Teste III: Caixa de Tênis Nike . . . . .</b>	<b>134</b>
<b>8.2.4</b>	<b>Teste IV: Símbolo de Acessibilidade . . . . .</b>	<b>135</b>
<b>8.2.5</b>	<b>Teste V: Perfume Polo Ralph Lauren. . . . .</b>	<b>136</b>
<b>8.2.6</b>	<b>Teste VI: Desodorante Adidas . . . . .</b>	<b>139</b>
<b>8.2.7</b>	<b>Teste VII: Camisa da Coca-Cola vestida num corpo feminino . . . . .</b>	<b>141</b>
<b>8.2.8</b>	<b>Teste VIII: Garrafa com o brasão da UFPE . . . . .</b>	<b>142</b>
8.3	SOBRE A IMPLEMENTAÇÃO . . . . .	145
8.4	ANÁLISE DO TEMPO DE EXECUÇÃO . . . . .	145
8.5	DISCUSSÃO . . . . .	146
<b>9</b>	<b>CONCLUSÃO . . . . .</b>	<b>149</b>
9.1	TRABALHOS FUTUROS. . . . .	150
	<b>REFERÊNCIAS. . . . .</b>	<b>152</b>

---

## 1 INTRODUÇÃO

---

Reconhecer padrões e objetos em uma imagem é um problema clássico em visão computacional, e isso é importante para a execução de tarefas repetitivas, difíceis e/ou até mesmo perigosas que podem ser automatizadas com velocidade e precisão, substituindo o papel humano no processo. Já existem inúmeras aplicações que visam facilitar e resolver problemas dessa natureza que podem ser encontrados em diferentes áreas, desde um simples reconhecimento óptico de caracteres em placas veiculares, até a detecção de tumores em imagens médicas e inspeções de linhas de transmissão de energia por imagens termográficas capturadas por *drones*.

Apesar dos avanços, muitos problemas desse campo estão ainda em aberto. Para o reconhecimento de objetos em cena há diversos fatores que devem ser levados em consideração, como luminosidade, sombra, ruídos provenientes do sistema de captura, etc. O contexto da cena é um fator de peso e, dentro da grande variedade de problemas da área, está a deformação em perspectiva sofrida pelos objetos, que é inerente ao processo de captura da imagem.

Será vista neste trabalho uma metodologia completa para o reconhecimento de objetos planares mesmo sob o efeito da perspectiva baseado em invariantes geométricos, oferecendo uma maneira viável para aplicações em visão computacional robustas a deformações dessa natureza.

### 1.1 MOTIVAÇÃO E OBJETIVOS

No nosso cotidiano, estamos cercados de objetos planares, tais como símbolos, caracteres e logomarcas, todos esses são amplamente utilizados para representar e informar. Na Figura 1 são expostas diversas situações onde há a presença de objetos planares, e em diferentes pontos de vista com diferentes graus de deformação projetiva. Os seres humanos conseguem facilmente discernir diferentes símbolos e também reconhecer símbolos idênticos sob diferentes pontos de vista.

Porém, isso ainda não é uma realidade tão comum para aplicações em visão computacional. A maioria dos descritores de imagem são robustos a transformações geométricas de muitas naturezas, mas poucos se propõem a lidar com transformações projetivas; os sistemas comerciais de reconhecimento óptico de caracteres, por exemplo, não tratam desse problema.

Diversas aplicações em visão computacional podem se tornar possíveis com a detecção robusta desses símbolos, oferecendo base para muitas aplicações, como reconhecimento de símbolos de trânsito para carros autônomos, reconhecimento de caracteres em cenas reais e em documentos, detecção de logomarcas mesmo em perspectiva e reconhecimento de marcadores planares para aplicações de realidade aumentada.



Figura 1 – Diversas situações em que há objetos planares presentes, divididos entre logomarcas, sinalizações e caracteres de placas automotivas. Os símbolos são apresentados em diferentes ângulos e graus de perspectiva.

Podemos imaginar aplicações bastante interessantes em se tratando de detecção de logomarcas em perspectiva, independente do ponto de vista. Em eventos esportivos, como a Copa do Mundo e Olimpíadas, os torneios são patrocinados e televisionados, empresas possuem grandes interesses de exibirem suas marcas durante os jogos, e pagam caro para que sua marca fique visível durante as disputas, as quais são transmitidas para os televisores do mundo todo. Na Copa do Mundo que se passou recentemente, foi possível notar diversas propagandas na beira do campo durante as partidas. Entretanto, as empresas que patrocinam esses eventos podem talvez não possuir métricas de tempo de exibição das suas logomarcas na TV, assim sendo, um sistema que contabilizasse o tempo de exibição de logomarcas durante as transmissões dos jogos ajudariam as empresas patrocinadoras a terem uma noção mais exata do retorno do seu investimento em publicidade nessas competições. Uma maneira para tornar essa ideia possível, seria com o uso de um descritor de forma planar robusto a transformações em perspectiva, as logomarcas exibidas nos painéis de LEDs expostos na periferia do campo podem estar sendo visualizadas em diferentes pontos de vista, e para se ter um sistema que contabilize fielmente o número de aparições durante as disputas faz-se necessário um descritor com essas especificações. Na Figura 2 é mostrado um exemplo de uso dessa aplicação.

Detecção e reconhecimento para a sinalização de trânsito é um outro importante nicho de atuação onde o uso de descritores de forma robustos a transformações projetivas são muito necessários para o reconhecimento desses símbolos. Dentro deste cenário, existem as aplicações de sistemas de visão computacional para carros autônomos, onde será possível visualizar as pla-



Figura 2 – Jogador da Bélgica comemorando após o gol. Atrás do atleta pode ser visto os painéis de LEDs exibindo as logomarcas da Hyundai em perspectiva e sendo detectadas por uma possível aplicação de contagem de logomarcas, robusta a transformações projetivas dessa natureza.

cas de sinalização de trânsito e reconhecer e distinguir os mais variados símbolos em diferentes graus de deformação projetiva; com isso, um carro autônomo poderá respeitar a sinalização das placas, vias e estacionamentos, evitando invadir as faixas reservadas para as ciclovias e virar na contra mão; poderá também identificar os nomes de ruas pelas placas e se orientar (independente de GPS); ao estacionar, um carro autônomo poderá visualizar e reconhecer a sinalização da vaga e estacionar em lugares adequados às necessidades do motorista, respeitando os espaços reservados para pessoas com deficiência, gestantes, idosos, etc. A Figura 3 exibe um conjunto de situações reais contendo sinalizações de trânsito e em diferentes pontos de vistas, o sistema de visão computacional de um carro autônomo deverá ser possível de detectar e reconhecer esses símbolos para tomar decisões de forma correta, evitando acidentes e infrações.

A necessidade de descritores de forma invariantes a projetividades é real, e o leque de aplicações é extenso, desde o reconhecimento de caracteres para placas de automóveis até o reconhecimento de logomarcas detalhadas, e marcadores planares para aplicações de realidade aumentada.

Já existem alguns trabalhos que tratam do reconhecimento de símbolos planares com deformações em perspectiva severas, em (LI; TAN, 2010) que faz uso de espectros de *razão cruzada*, um invariante projetivo, mas o mesmo se limitou ao reconhecimento de caracteres e sinalizações de trânsito, não resolvendo para símbolos mais complexos como logomarcas. Algumas outras técnicas vieram depois, resolvendo de maneira similar o reconhecimento, com o uso de uma novo invariante projetivo, o *número característico*, CN: do inglês *characteristic number*, em (LUO et al., 2013) e (JIA et al., 2016), tais técnicas foram muito bem sucedidas para o reco-



Figura 3 – Exemplos de sinalizações de trânsito em diferentes níveis de deformação projetiva.

nhecimento de objetos planares com deformação projetiva, foram realizados testes com várias bases de formas com variados níveis de complexidade.

Este trabalho tem o objetivo de propor um novo descritor de forma que ofereça suporte a um reconhecimento robusto desde as transformações geométricas mais básicas, como rotação, espelhamento, escala, cisalhamento, etc; até as transformações mais gerais como a perspectiva e projetividade. Para isso, o descritor proposto possui elementos invariantes a projetividades, organizados de uma maneira hierárquica que explora características mais periféricas da forma

até características mais internas da mesma. Tais informações são extraídas do objeto planar e armazenadas em um descritor de forma que poderá ser comparado com outros descritores, que representam outros objetos, que podem ser de mesma classe, ou não. Além das etapas de extração de características invariantes projetivas em objetos planares, são vistas também neste trabalho as etapas de comparação e classificação dos descritores, elemento por elemento do descritor, e oferecendo uma métrica final entre objetos com o objetivo de classifica-los. E por fim, serão realizados experimentos entre formas mais simples, como caracteres, em diferentes graus de deformação, e entre logomarcas com diferentes graus de deformação projetiva.

## 1.2 CONTRIBUIÇÕES

Dentre as contribuições deste trabalho, podemos enumerar:

1. Um novo descritor invariante projetivo com estrutura hierárquica que liga elementos de representação mais periféricos do objeto planar até elementos que representam características mais interiores do objeto;
2. O desenvolvimento de um novo invariante projetivo: Vetor de Razão Cruzada;
3. Comparação entre vértices baseada em espectros de topologia;
4. Distinção entre as comparações dos descritores para símbolos simétricos e assimétricos;
5. Remoção de associações falsas positivas inédita entre objetos assimétricos baseada em razão cruzada de cinco pontos coplanares;
6. Uso de correlação entre índices de vértices correspondentes que aumentam o poder de discriminação para formas assimétricas;
7. Elaboração de métrica de comparação composta entre objetos;
8. Melhoria nos resultados com a substituição das métricas simétricas e assimétricas pelo erro de transferência através da estimativa da homografia, desenvolvendo uma abordagem mais simples e eficiente;
9. Extrapolação das expectativas sobre a robustez do descritor proposto em condições mais adversas que contêm **espelhamento, oclusão e deformações não lineares**.
10. Valores de acurácia superiores ao estado da arte para símbolos mais complexos.

Foram realizados testes, mostrando a robustez do descritor para diversos cenários e realizações de testes com controle de visão de câmera através das variações do *azimute* e *elevação*.

### 1.3 ORGANIZAÇÃO DA DISSERTAÇÃO

O trabalho é dividido em três grandes partes: A primeira parte trata dos fundamentos teóricos e estado da arte sobre descritores de forma planar; a segunda parte é responsável pela metodologia da técnica proposta; a terceira parte é responsável pelos resultados e experimentos da técnica. Abaixo, são exibidos os capítulos agrupados nas partes específicas.

- **Parte I: Fundamentos Teóricos e Estado da Arte**

**Capítulo 2:** Nesse capítulo são exibidos os conceitos básicos do reconhecimento de padrões para imagens, explicando as etapas do processo desde a captura da imagem até a classificação. São mostrados também alguns trabalhos que segmentam e reconhecem formas planares em cenários diversos através de abordagens geométricas e/ou através de técnicas de aprendizagem de máquina.

**Capítulo 3:** Esse capítulo trata dos fundamentos básicos de geometria projetiva no plano, conceituando os elementos básicos do espaço  $\mathbb{P}^2$  e expondo a hierarquia das transformações geométricas e seus invariantes; ainda nesse capítulo é conceituada também a razão cruzada para quatro pontos colineares e para cinco pontos coplanares.

**Capítulo 4:** O capítulo é uma miscelânea de medidas e o estudo de um algoritmo de comparação de séries temporais, além de uma breve explicação do Coeficiente de Pearson usado para medir correlação linear; esse capítulo enumera um conjunto de ferramentas que podem ser usadas durante as comparações entre os elementos dos descritores.

**Capítulo 5:** Trabalhos relacionados, nesse capítulo há referências de diversas técnicas, desde as mais antigas e tradicionais até as técnicas mais recentes e, alinhado ao escopo deste trabalho, são mostrados alguns descritores de forma planar concorrentes, que também possuem elementos invariantes a transformações projetivas.

- **Parte II: Reconhecimento de objetos planares robusto a projetividades**

**Capítulo 6:** Esse é o primeiro capítulo da metodologia; ele trata da etapa de extração de características dos objetos planares e desenvolve um novo descritor hierárquico que agrupa os elementos que representam as características do objeto baseados em invariantes projetivos.

**Capítulo 7:** Nesse capítulo são mostradas as etapas de comparação entre objetos planares através da comparação entre os elementos dos descritores divididos em três diferentes abordagens, onde as duas primeiras são complementares e fazem distinção entre formas simétricas e assimétricas e a terceira abordagem é baseada na estimação de homografia, é discutido as semelhanças e diferenças entre as abordagens quanto a comparação entre os elementos, remoção de *outliers* e falsos-positivos e elaboração de métrica de similaridade utilizadas nas diferentes abordagens. São mostrados também vários exemplos e estudos de casos em algumas etapas do processo.

- **Parte III: Experimentos, Resultados e Conclusão**

**Capítulo 8:** O capítulo exhibe os resultados dos experimentos comparativos com outra técnica concorrente, e testes avulsos no intuito de explorar situações variadas, em cenas reais, e até mesmo com oclusão e deformações não lineares, e uma análise de tempo de execução.

**Capítulo 9:** O último capítulo deste trabalho trata da conclusão, onde são discutidos os pontos relevantes sobre a metodologia do descritor proposto e das melhorias que foram feitas ao longo do trabalho e outras que podem ser feitas com o objetivo de aumento de acurácia e aumento de desempenho.

---

## 2 VISÃO GERAL DO RECONHECIMENTO DE PADRÕES

---

Este capítulo aborda algumas das razões que motivaram a criação de sistemas de descritores visuais, bem como suas etapas que possibilitam essa tarefa. O reconhecimento de formas e padrões é a parte essencial para vários sistemas de visão computacional. No intuito de reduzir a dimensão e a redundância, facilitar e viabilizar a comparação entre diferentes formas faz-se necessário o uso de extração de características dos objetos nas imagens, essas características podem ser pontos, números, vetores, todas essas características são organizadas e formam o descritor do objeto. As abordagens para reconhecimento podem ser divididas em duas: Estatísticas e Estruturais (ARICA, 2003). As principais diferenças entre essas são:

1. A abordagem estatística é quantitativa, enquanto a abordagem estrutural é qualitativa e produz uma descrição composta de relações entre primitivas;
2. A abordagem estatística discrimina com base sobre diferenças numéricas entre características de diferentes grupos, enquanto na abordagem estrutural são usadas gramáticas para definir uma "linguagem" englobando as configurações aceitáveis de primitivas para cada grupo.

Os sistemas híbridos podem combinar as duas abordagens como forma de compensar pelas desvantagens de cada abordagem, ao mesmo tempo em que conserva as vantagens de cada uma (GOLDFARB, 1984). As características estruturais podem ser usadas com um classificador estatístico ou estrutural. Características estatísticas não podem ser usadas com um classificador estrutural porque eles não possuem informações relacionais, no entanto a informação estatística pode ser associada a primitivas estruturais e usada para resolver ambiguidades durante a classificação ou incorporado diretamente no próprio classificador, (OLSZEWSKI, 2001).

### 2.1 ETAPAS DO RECONHECIMENTO DE PADRÕES EM IMAGENS

Os sistemas de reconhecimento possuem uma estrutura geral composta de cinco etapas, como pode ser visto na Figura 4.



Figura 4 – Etapas do reconhecimento de padrões em imagens.

As duas últimas etapas são o cerne deste trabalho, após realizar a extração de características os descritores são comparados na etapa de classificação. Um sistema de reconhecimento flexível e robusto é capaz de reconhecer um objeto independente da sua orientação, tamanho e localização, de forma mais geral um bom sistema de reconhecimento é robusto a transformações geométricas. Em seguida são discutidas brevemente todas as etapas do reconhecimento de padrões em imagens.

### **2.1.1 Aquisição da imagem**

Para que uma imagem digital seja adquirida são necessários: um instrumento formador de imagem, um dispositivo físico sensível a uma faixa do espectro eletromagnético capaz de produzir um sinal elétrico proporcional ao nível de energia percebida. E um digitalizador, que converterá a saída elétrica do dispositivo foto-sensível para a forma digital (GONZALEZ; WOODS, 2006).

### **2.1.2 Pré-processamento**

No pré-processamento da imagem é realizada uma série de operações que possuem o intuito de melhorar a imagem, removendo eventuais ruídos, atenuando o embaçamento causado pelo movimento de câmera na etapa de aquisição e/ou realçando detalhes de interesse (ARICA, 2003). Para remoção de ruídos podem ser aplicados filtros espaciais, tais como filtro de média ou filtro de mediana, dependendo da natureza do ruído, ou então pode ser feita uma filtragem no domínio da frequência com o uso da transformada de Fourier da imagem. Existem inúmeras técnicas que podem ser usadas nesta etapa de pré-processamento.

### **2.1.3 Segmentação**

Após o pré-processamento teremos a etapa de segmentação. Na visão computacional, a segmentação da imagem é um dos problemas mais antigos e amplamente estudados (SZELISKI, 2010). A segmentação pode ser entendida como o particionamento da imagem em regiões que apresentam propriedades semelhantes. A detecção destas regiões pode ser feita através do agrupamento de pixels por características semelhantes, ou através de detecção de bordas de objetos e/ou regiões de interesse (ERPEN, 2004). Para detecção de bordas podem ser usadas algumas técnicas já bastante conhecidas, tais como, operador de Sobel, método de Canny, citando apenas alguns métodos clássicos de segmentação por detecção de bordas. Depois de ter segmentado a imagem, o conjunto de pixels agrupados resultante é então representado e descrito de uma maneira adequada para o processamento computacional (GONZALEZ; WOODS, 2006). Neste trabalho não são discutidas em detalhes as técnicas de processamento de imagens durante a etapa de pré-processamento e segmentação, visto que existem diversas técnicas e estratégias que podem ser melhores ou piores de acordo com o contexto da imagem, e aquisição da mesma;

mas para um melhor embasamento e contextualização, estamos interessados em segmentar formas planares, símbolos como caracteres, sinalização de trânsito, logomarcas, etc.

#### **2.1.4 Extração de características**

Após a aquisição da forma de interesse, obtida na etapa de segmentação, um conjunto de técnicas pode ser aplicado para extrair informações da forma, para que possa ser analisado de maneira mais adequada. Esse processo é chamado de descrição da forma e geralmente resulta em um vetor de características (estruturas mais complexas que um vetor de características também podem ser vistos na literatura, onde iremos discutir adiante). Nesta etapa busca-se alguma informação quantitativa de interesse ou que seja básica para a discriminação entre classes de objetos (GONZALEZ; WOODS, 2006). As abordagens de representação de forma são divididas em dois grupos:

1. **Representação Externa:** Técnicas baseadas em bordas;
2. **Representação Interna:** Técnicas baseadas em regiões.

Geralmente, uma representação externa é escolhida quando o interesse estiver voltado para características de forma, enquanto uma representação interna é escolhida quando o interesse estiver direcionado para propriedades como cor ou textura. O uso das duas abordagens em conjunto também é viável. É importante que as características extraídas sejam invariantes a transformações geométricas; existem características que são robustas a rotação, translação e escala, e algumas outras que suportam até transformações geométricas mais gerais, como afinidade e perspectiva. O resultado final desta etapa será uma estrutura de dados contendo informações de características de interesse que tem por finalidade diminuir a redundância dos dados, reduzindo a dimensionalidade, viabilizando e facilitando assim a posterior etapa de classificação.

#### **2.1.5 Classificação**

Classificação é o processo pelo qual se atribui um rótulo de categoria para um conjunto de medidas previamente extraídas (NIXON; AGUADO, 2012). De uma forma mais direta, a tarefa do classificador propriamente dito de um sistema completo de reconhecimento é usar o vetor (ou estrutura mais complexa) de características fornecido pela etapa anterior para atribuir o objeto a uma categoria (ERPEN, 2004). Como o desempenho de classificação perfeita é muitas vezes impossível, uma tarefa mais geral é determinar a probabilidade para cada uma das categorias possíveis. A abstração fornecida pela representação do vetor de característica da imagem/dado de entrada permite o desenvolvimento de uma teoria de classificação amplamente independente de domínio (OLSZEWSKI, 2001). O resultado da classificação pode ser um valor de distância, ou um valor de probabilidade de chance da ocorrência de alguma classe, ou também pode ser uma imagem com algumas características enfatizadas para auxiliar o especialista em sua

tomada de decisão. No caso do classificador binário, dado um conjunto de classes e um padrão apresentado como entrada para o sistema, o problema consiste em decidir a que classe o padrão pertence; deve haver a alternativa de rejeição do padrão (ERPEN, 2004).

## 2.2 EXTRAÇÃO E RECONHECIMENTO DE FORMAS PLANARES

O objetivo deste trabalho é descrever características invariantes projetivas das formas planares extraídas das cenas para depois realizar uma classificação através das comparações dos descritores pelo vizinho mais próximo, então o foco de interesse é após a segmentação dessas formas planares, para isso existem diversas técnicas e estratégias que possuem a finalidade de extrair esses símbolos. A seguir, são citados vários trabalhos de extração e reconhecimento de caracteres, sinais de trânsito e logomarcas em cenas reais; alguns trabalhos se dedicam até a etapa de segmentação desses símbolos, outros abordam todas as etapas, até a classificação.

Existem muitos trabalhos que possuem esses propósitos, em (CASEY; LECOLINET, 1996) são revisadas e discutidas diversas técnicas e estratégias para a segmentação de **caracteres** em imagens de documentos, em (ZHANG et al., 2013) e (CHAVRE; GHOTKAR, 2015) são mostradas e revisadas outras diversas estratégias para a segmentação de textos em cenas naturais; em (CHEN; YUILLE, 2004) é mostrado uma técnica de detecção e reconhecimento de texto em cenas reais usando *features* baseadas em regiões com baixa entropia, histogramas de intensidade e variações de gradiente; em (GREENHALGH; MIRMEHDI, 2015) foi desenvolvido uma técnica para o reconhecimento de texto em placas de trânsito, onde eram encontradas áreas candidatas baseado em *Maximally Stable Extremal Regions* (MSERs), e uma limiarização no espaço de cores HSV, e para a etapa de reconhecimento foi usado um OCR (*Optical Characters Recognition*) nas regiões segmentadas; em (CHEN; LIAO, 2012) foi desenvolvido um reconhecedor de caracteres chineses para dispositivos móveis segmentados por MSERs; em (ZHANG; GAO; TIAN, 2018) foi desenvolvida uma outra técnica para reconhecimento de texto em cenas naturais que realiza uma segmentação baseada em MSERs combinada com uma estimação prévia local de cores através de *Stroke Width Transform* (SWT) e uma posterior classificação com uso de uma rede neural convolucional (*Convolutional Neural Network*, CNN) após a segmentação.

Para a segmentação e classificação de **sinais de trânsito** temos: em (MALDONADO-BASCÓN et al., 2007) a segmentação das placas de trânsito é feita através de uma limiarização no espaço de cores HSI e a classificação é feita com o uso de SVM (*Support vector Machine*); em (MATHIAS et al., 2013), faz uso de características variantes de HOG (*Histogram of Oriented Gradients*) para a detecção, e para classificação é feita através de uma representação esparsa; em (BAMBHORI; JALGAON, 2013) foi usado um segmentador de placas de trânsito baseado em *k-means* no espaço de cores  $L^*a^*b^*$  e uma análise da forma dos objetos segmentados quanto a geometria e a análise do perímetro e área dos objetos extraídos; em (LIU et al., 2016) é proposto uma técnica rápida para detecção e extração de sinais de trânsito baseada também em HOG e representação esparsa, onde os símbolos foram extraídos através de uma técnica

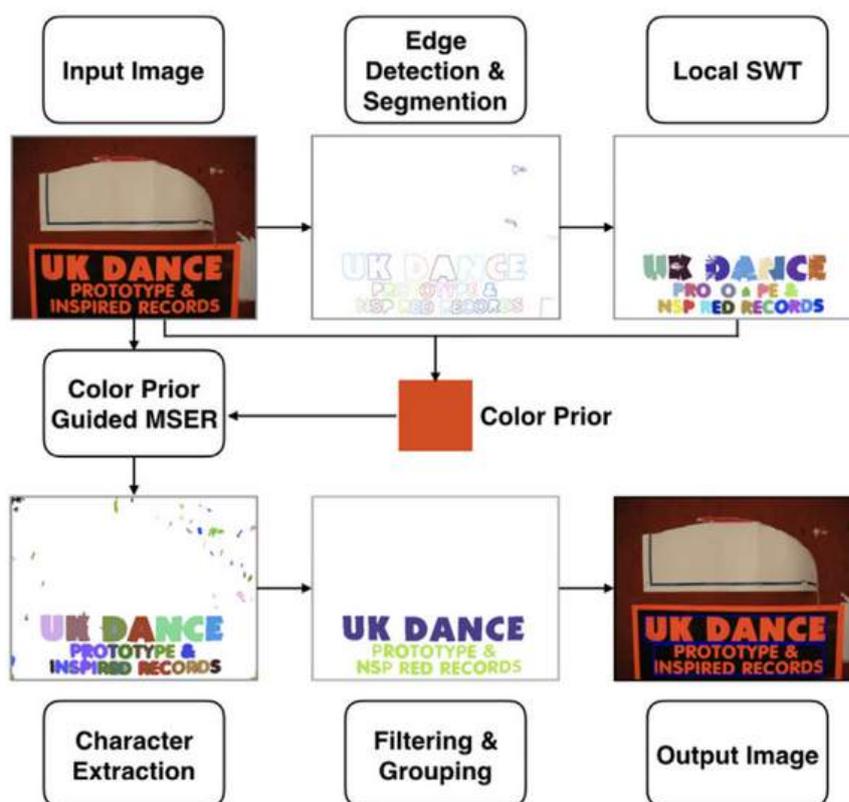


Figura 5 – Fluxo de operações para a segmentação de caracteres em cenas naturais através de MSERs e SWT, retirado de (ZHANG; GAO; TIAN, 2018).

própria baseada em regiões de interesse de alto contraste; em (KUANG; FU; YANG, 2018) foi feita uma detecção e reconhecimento em tempo real de símbolos de trânsito baseado em MSERs, depois são extraídas características geométricas e finalmente é construída uma árvore de decisão baseado em características HSV-HOG-LBP (*Localized Binary Pattern*, LBP) para o reconhecimento.



Figura 6 – Exemplo de segmentação de placa de trânsito em uma cena natural através de uma limiarização no espaço de cores HSI, retirado de (MALDONADO-BASCÓN et al., 2007).

Para o reconhecimento e segmentação de **logomarcas**, temos: em (POURGHASSEM, 2012) é proposta uma solução para a extração de logomarcas em documentos que é feita em dois estágios de segmentação constituídos de algoritmos baseados em wavelets e limiarização, e posterior classificação por MLP (*Multi Layer Perceptron*) e K-NN (*K-Nearest Neighbor*); em (DIXIT; SHIRDHONKAR, 2016) foi elaborada uma técnica de extração de logomarcas baseada em limiarização com decomposição por valor singular (*Singular Value Decomposition*, SVD) em documentos; em (RAVEENDRA; REDDY; KISHORE, 2018) são revisados e discutidos diversos trabalhos de detecção de logomarcas em documentos. Em (SHEN; WU; ZHENG, 2012) foi desenvolvido um método para a extração e reconhecimento de logomarcas de televisão baseado nas variações do *background* das imagens transmitidas pela TV com o uso de um método Otsu adaptativo local com uma máscara da logomarca, e para a classificação é feita uma análise em um histograma especial (*Projection Distribution Histogram Intersection*, PDHI); em (AOKI; OHYAMA; WAKABAYASHI, 2017), foi proposto um método de extração e reconhecimento automático para logomarcas em calçados contidas em cenas naturais usando um limitado número de amostras para o treinamento, para a extração das logomarcas foi usado um método baseado em MSERs e um algoritmo iterativo para agrupamento de regiões e características de gradiente, e para a classificação foi usado SVM justificando que o número de dados disponíveis para esse propósito (logomarcas em calçados) é pequeno para a utilização de uma técnica de aprendizagem profunda.

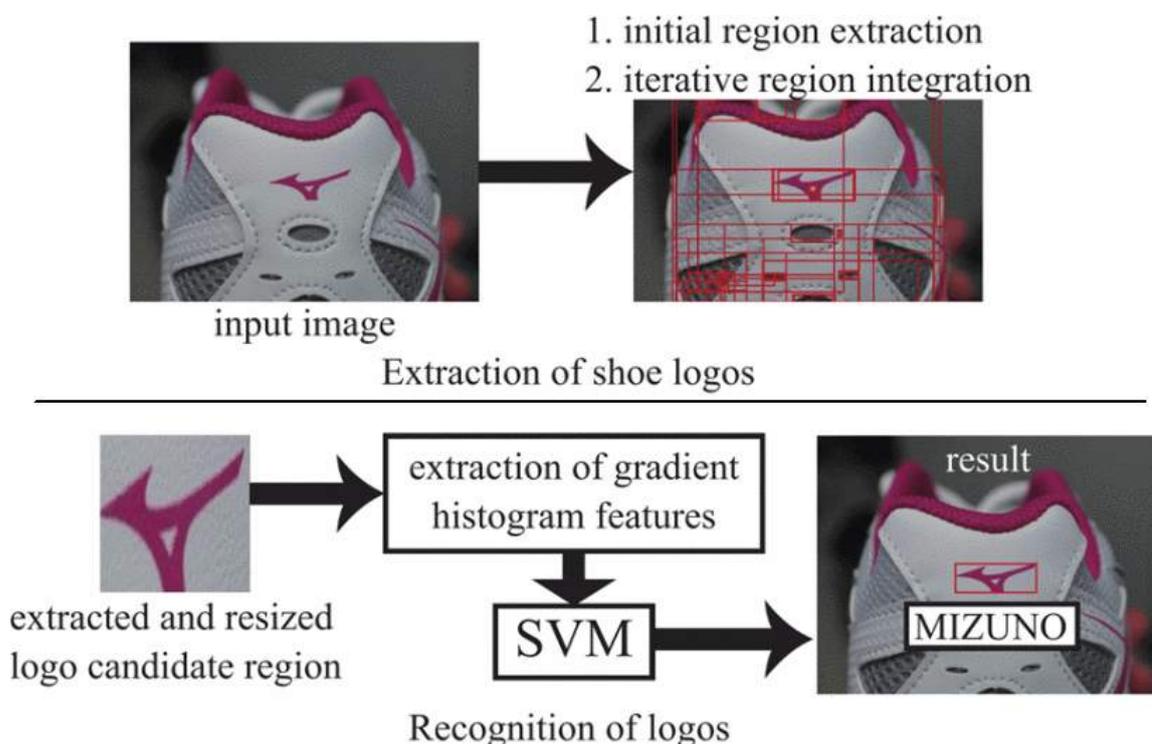


Figura 7 – Extração de logomarcas em imagens de calçados por MSERs e reconhecimento das formas através de SVM, retirado de (AOKI; OHYAMA; WAKABAYASHI, 2017).

Todos esses trabalhos tratam da segmentação, extração de características e classificação de símbolos em cenários diversos, mas nenhum desses trata do problema das deformações proje-

tivas que tais objetos planares estão suscetíveis; o descritor proposto neste trabalho é robusto para representar e reconhecer as formas planares também nessas circunstâncias; as etapas de aquisição, pré-processamento e segmentação das formas planares não são discutidas, pois não são o foco deste trabalho, mas foram citados diversos trabalhos nesta Seção que explicam diversas técnicas para a segmentação dos símbolos em documentos e até em cenas naturais que justificam o nosso foco de interesse e viabilidade deste trabalho como um módulo independente.

### 2.3 FLUXO BÁSICO DE RECONHECIMENTO

O fluxo básico para o reconhecimento de objetos planares adotado neste trabalho é exemplificado pelo diagrama abaixo, pode-se ver as etapas iniciais de aquisição, pré-processamento e segmentação, logo após vem a etapa de extração de características, que visa extrair e organizar de forma estruturada informações relevantes do objeto em um descritor, e na última etapa é feita a comparação entre os objetos através da comparação dos descritores de imagem. Ainda no diagrama da Figura 8 é exibido um objeto retirado de uma cena real, pré-processado, segmentado e convertido em um descritor após passar pela etapa de extração de características, e sendo comparado com vários objetos-modelos na etapa de classificação; o resultado final do processo aponta para qual classe o objeto investigado pertence. O objeto inspecionado que servirá de consulta é chamado de *query* e o objeto-modelo armazenado na base é chamado de *template*.

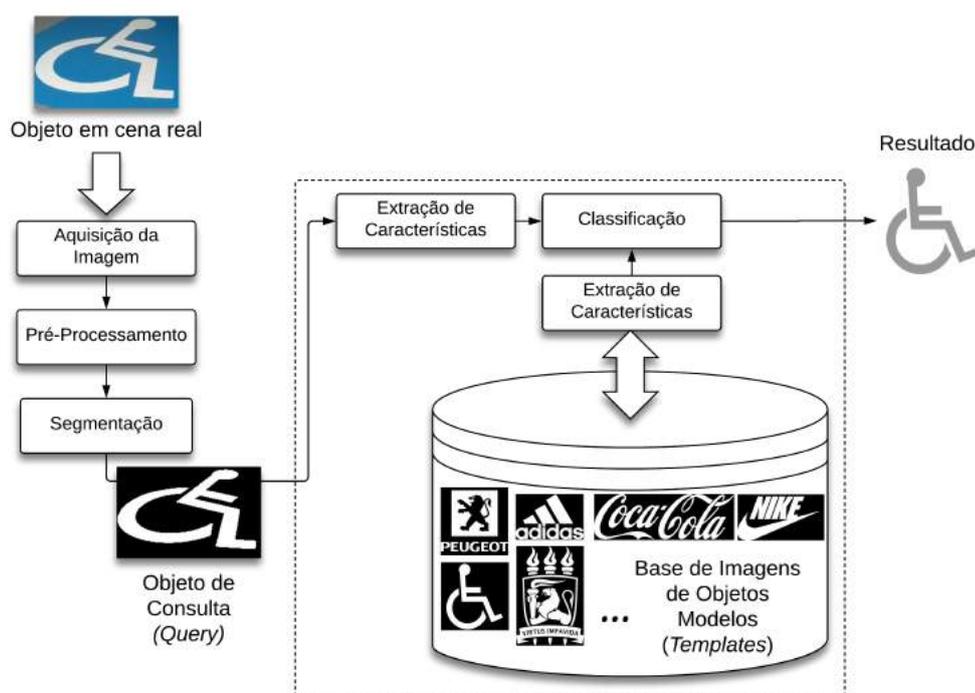


Figura 8 – Etapas do reconhecimento de objetos planares integrado com uma base de imagens de objetos modelos.

## 2.4 DISCUSSÃO

Este capítulo tem o objetivo introdutório de explicar alguns conceitos básicos do reconhecimento de imagens, explicando brevemente todas as cinco etapas necessárias para o reconhecimento.

Foram mostrados inúmeros trabalhos na Seção 2.2 que explicam variadas técnicas de segmentação, extração de características e classificação de caracteres, sinais de trânsito e logomarcas em diversos cenários, abordando desde a segmentação até a classificação dessas formas, fazendo uso de abordagens variadas, indo desde a análise de histogramas e formas até algumas outras que utilizam aprendizagem de máquina para a classificação dessas formas. Sendo assim, existem variadas estratégias para a segmentação desses símbolos, e variadas maneiras de representa-los e classifica-los, onde as técnicas para cada etapa podem ser encaradas como módulos independentes, que podem variar com o contexto onde os símbolos estão inseridos.

Diante disso, para este trabalho são abstraídas as etapas de aquisição, pré-processamento e segmentação, o foco deste trabalho são as etapas de **extração de características e classificação** adaptados para resolver o problema de reconhecimento de objetos planares que sofreram transformações projetivas.

---

### 3 FUNDAMENTOS BÁSICOS DE GEOMETRIA PROJETIVA NO PLANO

---

Este capítulo expõe os princípios geométricos da visão e como os objetos da cena são afetados pelas distorções intrínsecas ao processo de aquisição da imagem em perspectiva, e as propriedades invariantes às transformações nesses objetos à luz da Geometria Projetiva.

A História da Geometria Projetiva se inicia no Renascimento, na Itália do século XV. Os artistas, buscando um maior realismo para suas obras, introduziram os conceitos de ponto de fuga e perspectividade. A primeira obra conhecida por fazer uso da perspectiva linear foi criada pelo arquiteto e pintor Filippo Brunelleschi (1377-1446), ele resgatou os princípios da perspectiva cônica, que eram estudados pelos gregos e romanos, mas esse conhecimento ficou esquecido ao longo de toda a Idade Média. O sistema de perspectiva linear ofereceu às pinturas a ilusão de profundidade, baseada em linhas que convergiam nos pontos de fuga localizados na linha do horizonte. Logo após a pintura de Brunelleschi, a técnica foi disseminada entre outros artistas, como pode ser visto numa das obras de Pietro Perugino (1450-1523), Fig. 9. E até o final do século XV, todos os grandes pintores da Renascença dominavam a perspectiva na esfera da Arte.

Entretanto, demorou cerca de dois séculos para que essas ideias pudessem ser formuladas matematicamente. Foi apenas em 1639, com o trabalho pioneiro sobre a teoria geométrica das cônicas, o *Broullion Projet*, onde Gerard Desargues (1591-1661) formalizou esses conceitos. Na época, seu trabalho não foi bem aceito por outros estudiosos e somente no início do século XIX, Jean Victor Poncelet (1788-1867) pôde resgata-lo. Poncelet estudou na *École Polytechnique* de Metz e foi professor de matemática, fez parte do exército de Napoleão e lutou em 1812 contra a Rússia, acabou sendo capturado e ficou preso durante dezoito meses, e nesse período de cárcere escreveu sua principal obra: *Traité des propriétés projectives des figures*, publicada somente em 1822. Seu trabalho delineou os fundamentos da geometria projetiva, bem como alguns novos resultados. Após Poncelet, houveram muitas outras grandes contribuições nessa área e até o final do século XIX a geometria projetiva estava consolidada (AUFFINGER; VALENTIM, 2003).

Este capítulo aborda a geometria das transformações projetivas no plano. Essas transformações modelam a distorção geométrica que surge quando um plano é visualizado em perspectiva por uma câmera. Sob a perspectiva de imagens certas propriedades geométricas são preservadas, como a colinearidade de três ou mais pontos e a interseção de retas, enquanto outras propriedades não são, por exemplo, linhas paralelas não continuam como linhas paralelas após sofrerem uma perspectividade. A geometria projetiva propicia a modelagem da dinâmica dos objetos em manipulação de imagens e também fornece uma representação matemática apro-



Figura 9 – *Entrega de las llaves a San Pedro*, Capela Sistina (1481–1482) - Pietro Perugino

priada para cálculos. Será descrito como a manipulação de elementos do espaço projetivo é viabilizada através de uma representação dos pontos e retas em coordenadas homogêneas. Serão vistas também neste capítulo os elementos geométricos que são invariantes aos diferentes tipos de transformações que são lineares nas coordenadas homogêneas, e será dada uma atenção especial aos invariantes projetivos por serem mais gerais em imagens reais. Grande parte do conteúdo deste capítulo foi retirado do livro (HARTLEY; ZISSERMAN, 2004).

### 3.1 O ESPAÇO PROJETIVO

Como já é de conhecimento, num plano bidimensional um ponto pode ser representado como par de coordenadas  $(x, y)$  em  $\mathbb{R}^2$ . Então, é comum, identificar o plano com  $\mathbb{R}^2$ . Considerando  $\mathbb{R}^2$  como um espaço vetorial, o par de coordenadas  $(x, y)$  é um vetor. Um ponto é identificado como um vetor. A definição formal das funções de imagem estabelece que um ponto  $p = (x, y)$  qualquer deve pertencer ao espaço planar  $\mathbb{R}^2$  e que pode ser tomado como uma projeção planar de um ponto representante de uma reta inteira que passa na origem do espaço euclidiano tridimensional, sendo que a cada reta que passa na origem corresponde apenas um representante, como ocorre tipicamente em classes de equivalência. No caso, a relação de equivalência é a de que dois pontos estão numa mesma classe se e somente se pertencem à mesma reta que passa na origem. Topologicamente esta construção leva a um espaço bidimensional que corresponde ao  $\mathbb{R}^2$  mas com a adição de uma reta a mais, normalmente utilizada para representar pontos no infinito, que ocorrem por exemplo, em interseções de retas paralelas. A este espaço bidimensional, que corresponde ao  $\mathbb{R}^3$  quocientado pela relação de equivalência citada acima, denominamos espaço projetivo  $\mathbb{P}^2$ . De uma forma mais simples, para se obter o espaço

projetivo  $\mathbb{P}^2$ , toma-se um plano e, juntamente com a reta extra incorporada, denominada reta do infinito  $l_\infty$ , cuja resultante obtida por uma perspectividade corresponde à linha do horizonte em imagens com perspectiva. Na geometria projetiva a reta  $l_\infty$  não possui qualquer propriedade especial em relação as outras retas, onde existe a possibilidade de se operar com ela. No âmbito da geometria projetiva duas retas coplanares, ou são coincidentes ou possuem um ponto em comum. Duas retas, normalmente consideradas paralelas, que não se cruzam no espaço euclidiano ou no afim, no espaço projetivo cruzam-se num de seus elementos operáveis, ou seja, o seu ponto de interseção está precisamente em  $l_\infty$  o qual, em termos visuais numa imagem real, corresponde à direção das retas paralelas.

### 3.2 UM MODELO PARA O PLANO PROJETIVO

A interpretação de informações visuais em termos de elementos geométricos conhecidos (*i.e.*, direções, pontos, retas) e definidos sobre o espaço  $\mathbb{P}^2$  é especialmente vantajoso, pois representa de maneira natural as diferentes transformações projetivas (EVANGELISTA, 2016). Uma forma de pensar no  $\mathbb{P}^2$  é como um conjunto de raios no  $\mathbb{R}^3$  que contém sua origem. O conjunto de todos os vetores  $k(x, y, w)^T$ , com  $k$  não nulo variando como parâmetro de um raio passando pela origem, tal raio pode ser considerado como um único ponto no  $\mathbb{P}^2$ . Neste modelo, as linhas em  $\mathbb{P}^2$  são planos que passam pela origem. Observe que dois raios não idênticos se encontram em exatamente um plano e dois planos se cruzam em um raio. Isso é análogo a afirmar que dois pontos distintos definem uma reta, e duas retas distintas sempre se cruzam em um ponto. Pontos e retas podem ser obtidos ao cruzar este conjunto de raios e planos pelo plano  $w = 1$ . Resumidamente pontos e retas de  $\mathbb{P}^2$  são representados respectivamente por raios e planos que cruzam a origem de  $\mathbb{R}^3$ . Os raios que se encontram no plano  $xy$  representam pontos ideais (direções), e o plano  $xy$  paralelo ao plano  $w = 1$  representa  $l_\infty$ , conforme ilustrado na Fig.10 e na Fig. 11, onde pontos e retas são exibidos no plano  $\pi$  respectivamente.

### 3.3 COORDENADAS HOMOGÊNEAS

#### 3.3.1 Pontos finitos e ideais em $\mathbb{P}^2$

Seja o vetor  $p \in \mathbb{P}^2$ , representado pelo trio ordenado  $p = (x_p, y_p, w_p)^T$ . Geometricamente,  $p$  pode ser interpretado como um ponto próprio (ou finito) quando  $w \neq 0$  ou ponto impróprio (ou direção) quando  $w = 0$ . Triplas de números que representam pontos e direções projetivos são escaláveis, o que significa que  $p = (x_p, y_p, w_p)^T \equiv k(x_p, y_p, w_p)^T$ , para  $k \neq 0$ . O conceito de plano projetivo homogêneo é comumente aplicado para representar o plano da imagem, ou seja, tudo o que é visível no mundo real plano, e que é a entrada do problema. As triplas de números que definem esse plano possuem a terceira coordenada homogênea  $w = 1$ . Ao se aplicar uma transformação projetiva, eventualmente, tais vetores são mapeados para elementos com coordenada homogênea  $w'$  distinta. Se  $w' \neq 0$ , então basta dividir todas as coordenadas por  $w'$  e o mesmo é reprojetoado para o plano, essa operação é denominada normalização pela coordenada

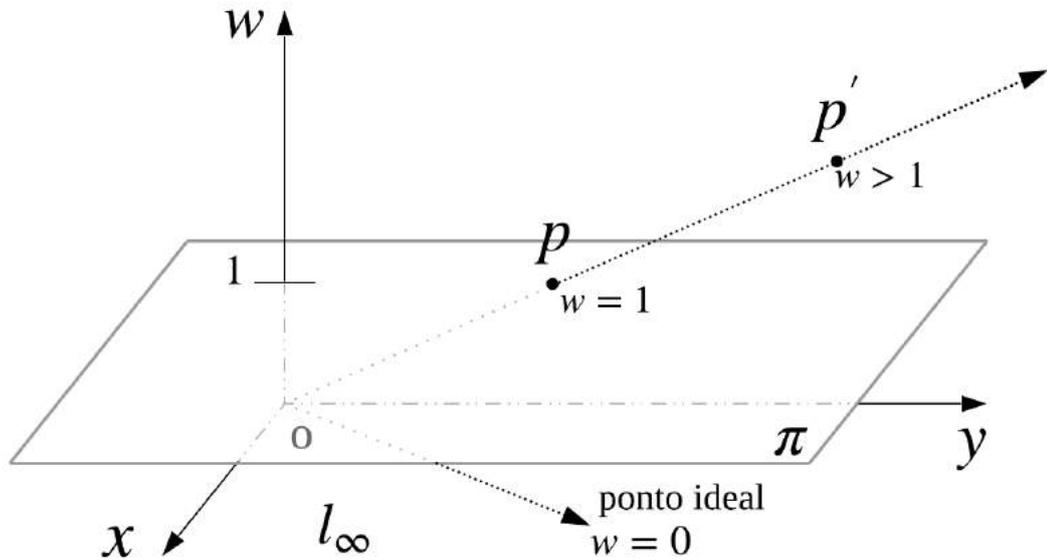


Figura 10 – Interpretação geométrica de pontos e vetores em  $\mathbb{P}^2$ . Pontos são representados por raios que passam pela origem de  $\mathbb{R}^3$ . O ponto  $p$  é equivalente ao ponto  $p'$ , isso quer dizer que ambos são representantes do mesmo ponto em  $\mathbb{P}^2$ , o ponto  $p$  está posicionado no plano  $w = 1$ , logo  $p = (x_p, y_p, 1)^T$ , então  $p' = (kx_p, ky_p, k)^T$ , para  $k > 1$ , neste caso onde  $p'$  está acima de  $p$ . As linhas que se encontram no plano  $xy$  representam pontos ideais (direções), e o plano  $xy$  representa  $l_\infty$ .

homogênea, e corresponde a recuperar o representante-padrão da classe de equivalência que é o ponto projetivo. Se  $w' = 0$ , então a tripla resultante é interpretada como uma direção, ou como se o mesmo fosse um ponto infinitamente distante da origem, conhecido como ponto ideal. Estes pontos ideais correspondem a vetores no  $\mathbb{R}^2$ .

### 3.3.2 Retas em $\mathbb{P}^2$

Retas respeitam à equação geral  $ax + by + c = 0$  e também são representadas como triplas  $r = (a, b, c)^T \in \mathbb{P}^2$ . Retas são escaláveis em  $\mathbb{P}^2$ , o que significa que  $r = (a, b, c)^T \equiv k(a, b, c)^T$ . Ou seja, independentemente da escala  $k \neq 0$ , o vetor normal representa a mesma reta. Geometricamente, uma reta em  $\mathbb{P}^2$  representa a intersecção entre o plano homogêneo e o plano de normal  $(a, b, c)^T$  e que passa pela origem do espaço vetorial tridimensional  $\mathbb{P}^2$ , conforme ilustra a Fig. 11, dois pontos representados por  $p$  e  $q$  cujas formas normalizadas são apresentadas sobre o plano homogêneo  $\pi$ . A reta representada por  $r$  que contém os pontos representados por  $p$  e  $q$  é dada por  $r = p \times q$ , onde  $\times$  indica o produto vetorial entre os vetores (do  $\mathbb{R}^3$ )  $p$  e  $q$  que representam os pontos. Como resultado, no contexto do  $\mathbb{R}^3$ , o espaço anfitrião,  $r$  é o vetor perpendicular tanto a  $p$  quanto a  $q$ , ou seja, o vetor normal ao plano gerado pelos vetores de  $p$  e  $q$  e que é o representante da reta exibida em  $\pi$ .

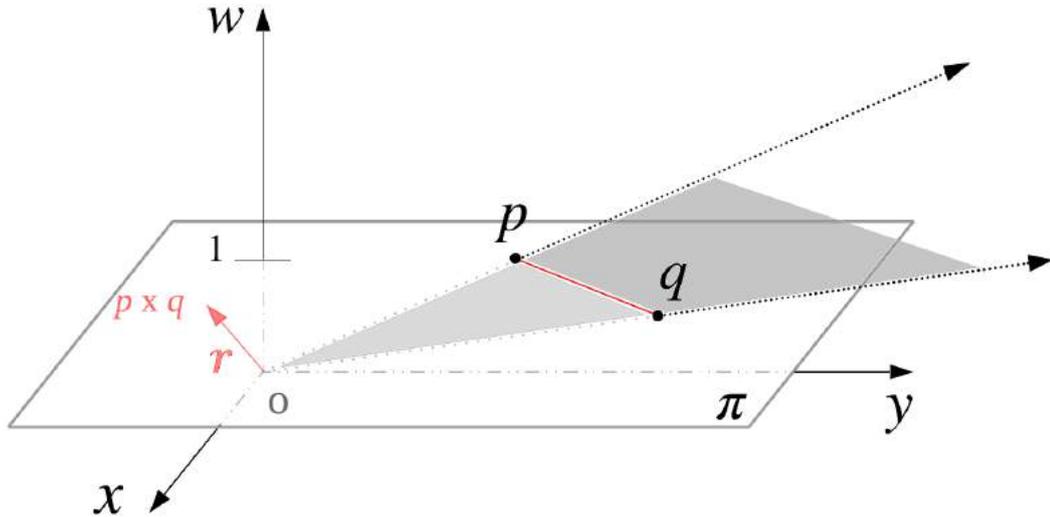


Figura 11 – Interpretação geométrica de uma reta em  $\mathbb{P}^2$  com o auxílio da geometria do espaço  $\mathbb{R}^3$ . O vetor (do  $\mathbb{R}^3$ )  $r$  que representa uma reta projetiva é obtido pelo produto vetorial dos vetores (do  $\mathbb{R}^3$ )  $p$  e  $q$  que representam dois pontos projetivos pertencentes à reta, ou seja,  $r = p \times q$ . O vetor  $r$  é normal ao plano que passa pela origem e que é gerado pelos vetores  $p$  e  $q$ , plano este que projetivamente é a reta representada por  $r$  e que por sua vez intersecta o plano homogêneo  $\pi$  numa reta da imagem na qual o segmento vermelho está contido.

### 3.4 TRANSFORMAÇÕES PROJETIVAS

O estudo da geometria projetiva num plano analisa as propriedades do plano projetivo  $\mathbb{P}^2$  que são invariantes sob um grupo de transformações conhecidas como *projetividades*. Uma *projetividade* é uma transformação inversível de pontos em  $\mathbb{P}^2$  para pontos em  $\mathbb{P}^2$  que mapeiam linhas em linhas. Mais precisamente, uma *projetividade* é um mapeamento inversível  $h$  de  $\mathbb{P}^2$  para si mesmo, tal que três pontos  $p_1$ ,  $p_2$  e  $p_3$  serão colineares se e somente se  $h(p_1)$ ,  $h(p_2)$  e  $h(p_3)$  também forem. As *projetividades* formam um grupo, uma vez que o inverso de uma *projetividade* também é uma *projetividade*, assim como a composição de duas projeções. Uma *projetividade* também é chamada de *colinação*, uma *transformação projetiva* ou uma *homografia*: os termos são sinônimos (HARTLEY; ZISSERMAN, 2004). Uma transformação projetiva planar é uma transformação linear em coordenadas homogêneas representada por uma matriz  $3 \times 3$  não-singular:

$$\begin{pmatrix} x'_p \\ y'_p \\ w'_p \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_p \\ y_p \\ w_p \end{pmatrix}, \quad (3.1)$$

onde pode ser escrito como  $p' = Hp$ .

Observe que a matriz  $H$  possui nove elementos, mas como se utilizam coordenadas homo-

gêneas a transformação é definida a menos de um fator de proporcionalidade não nulo. Assim sendo, esta transformação projetiva apresenta oito graus de liberdade. Uma transformação projetiva projeta cada figura em uma figura projetivamente equivalente, deixando todas as suas propriedades projetivas invariantes. No modelo de raios da Figura 10, uma transformação projetiva é simplesmente um operador linear inversível do  $\mathbb{R}^3$ .

### 3.5 HIERARQUIA DE TRANSFORMAÇÕES GEOMÉTRICAS

Nesta seção, descrevemos a importância das especializações de uma transformação projetiva e suas propriedades geométricas. Transformações projetivas podem ser classificadas em subgrupos considerando essas propriedades. Será vista a hierarquia de transformações e os efeitos de distorção de várias transformações começando das classes mais especializadas, seguindo a ordem: *isometrias*, *similaridades*, *afinidades* e por fim *projetividades*. São exibidos também os invariantes geométricos relativos a cada transformação; entende-se por invariantes como uma alternativa para descrever essas transformações além da forma algébrica (ou seja, como uma matriz atuando em coordenadas de um ponto ou curva). É descrever a transformação em termos de elementos ou quantidades que são preservadas (ou invariantes). Um invariante (escalar) de uma configuração geométrica é uma função de configuração cujo valor é inalterado por uma transformação em particular. Por exemplo, a separação de dois pontos é inalterada em uma transformação isométrica (translação e rotação), mas não por uma similaridade (que compõe translações e rotações com escalas isotrópicas). A distância é, portanto, um invariante isométrico, mas não é um invariante de similaridade. O ângulo entre duas linhas é tanto um invariante isométrico quanto um invariante de similaridade. Várias das transformações importantes para a visão computacional podem ser representadas pela ação de um grupo. É pertinente organizar estes grupos com base na generalidade das transformações. Quando se desce na hierarquia, os grupos de transformações se tornam menores e menos genéricos, a estrutura espacial correspondente torna-se mais rígida e existem mais invariantes. Por exemplo, uma transformação projetiva é mais genérica que uma transformação isométrica, pois aplica-se a mais situações, conseqüentemente haverá menos invariantes associados às transformações projetivas do que às transformações isométricas (HARTLEY; ZISSERMAN, 2004).

#### 3.5.1 Classe I: Isometrias

As *isometrias* (de *iso* = mesmo(a), *métrica* = medida) são transformações do plano  $\mathbb{R}^2$  que preservam a distância Euclidiana. Uma *isometria* é também conhecida como Transformação Euclidiana, e é representada algebricamente como:

$$\begin{pmatrix} x'_p \\ y'_p \\ 1 \end{pmatrix} = \begin{bmatrix} \epsilon \cos \theta & -\sin \theta & t_x \\ \epsilon \sin \theta & \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_p \\ y_p \\ 1 \end{pmatrix}, \quad (3.2)$$

onde  $\epsilon = \pm 1$ .

Se  $\epsilon = 1$ , então a *isometria* preserva a orientação. Se  $\epsilon = -1$  então a *isometria* inverte a orientação, um exemplo é a composição de uma reflexão com uma translação e/ou rotação. As Transformações Euclidianas modelam o movimento de um objeto rígido, são transformações que preservam a forma geral e a escala dos objetos envolvidos, os axiomas definidos pela geometria euclidiana são preservados. Os invariantes geométricos (e exclusivos) deste grupo são: comprimentos (distâncias entre dois pontos), ângulos e áreas. Além desses, os invariantes do grupo das *Similaridades* também são invariantes isométricos.

### 3.5.2 Classe II: Similaridades

Uma transformação de *similaridade* (ou mais simplesmente uma *similaridade*) é uma isometria composta com uma escala isotrópica. No caso de uma transformação euclidiana composta com uma escala (ou seja, sem reflexão) a *similaridade* é representada pela matriz:

$$\begin{pmatrix} x'_p \\ y'_p \\ 1 \end{pmatrix} = \begin{bmatrix} s \cos \theta & -s \sin \theta & t_x \\ s \sin \theta & s \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_p \\ y_p \\ 1 \end{pmatrix}, \quad (3.3)$$

onde o escalar  $s$  representa a escala isotrópica.

Uma transformação de similaridade também é conhecida como uma transformação *conforme*, porque preserva a "forma". Tais transformações, além de incluírem as isométricas, se caracterizam pela inclusão da escala uniforme de um objeto. Apesar de não preservarem a distância entre pontos, a razão entre distâncias e ângulos, são grandezas invariantes sob este grupo de transformações, incluindo os invariantes das *Afinidades* que englobam este grupo.

### 3.5.3 Classe III: Afinidades

Uma *Afinidade* ou Transformação Afim é uma transformação linear não singular seguido de uma translação. E possui a seguinte representação matricial:

$$\begin{pmatrix} x'_p \\ y'_p \\ 1 \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x_p \\ y_p \\ 1 \end{pmatrix} \quad (3.4)$$

As transformações afins formam um subgrupo especial das transformações projetivas que além de permitir transformações de similaridade e euclidianas, permitem a distorção do formato dos objetos, incorporando transformações de cisalhamento e escalas não-uniforme. Objetos que sofreram com Transformações afins possuem como invariantes o paralelismo entre retas, assim como a razão entre distâncias entre pontos pertencentes a retas paralelas, e a razão entre áreas, além dos invariantes projetivos.

### 3.5.4 Classe IV: Projetividades

As transformações projetivas foram definidas na Seção 3.4. É uma transformação linear não singular de coordenadas homogêneas. Engloba todos os outros grupos, e inclui também as transformações que mapeiam elementos geométricos em  $\mathbb{P}^2$  para um plano projetivo diferente, conhecidas como homografias.

Na Figura 12 podem ser vistos os pontos do plano mundial  $\pi$  com os pontos do plano  $\pi'$ , isso ilustra uma perspectividade, o mesmo vale em relação ao plano  $\pi$  para o segundo plano  $\pi''$ . As imagens de pontos de um plano estão relacionadas a pontos de imagem correspondentes em uma segunda vista por uma homografia como mostrado na Fig. 12. Esta é uma relação projetiva uma vez que depende apenas das interseções de planos com linhas. O mapeamento da homografia "transfere" pontos de uma vista para a outra como se fossem imagens de pontos no plano. Em outras palavras, uma transformação projetiva completa pode ser obtida como a aplicação de uma sequência finita de projeções de um plano sobre outro, dados pontos em comum de convergência. Outro ponto importante a ser destacado é sobre os invariantes: pode ser percebido, a partir da Figura 12, que ao se aplicar sequências de projeções de planos sobre planos a colinearidade dos pontos será mantida mesmo que as distâncias entre eles não o sejam. Além da colinearidade, a intersecção, concorrência, tangência, inflexão e razão cruzada, são algumas dos invariantes projetivos. Os invariantes projetivos, incluindo a razão cruzada, serão vistos com mais detalhes na Seção 3.6.

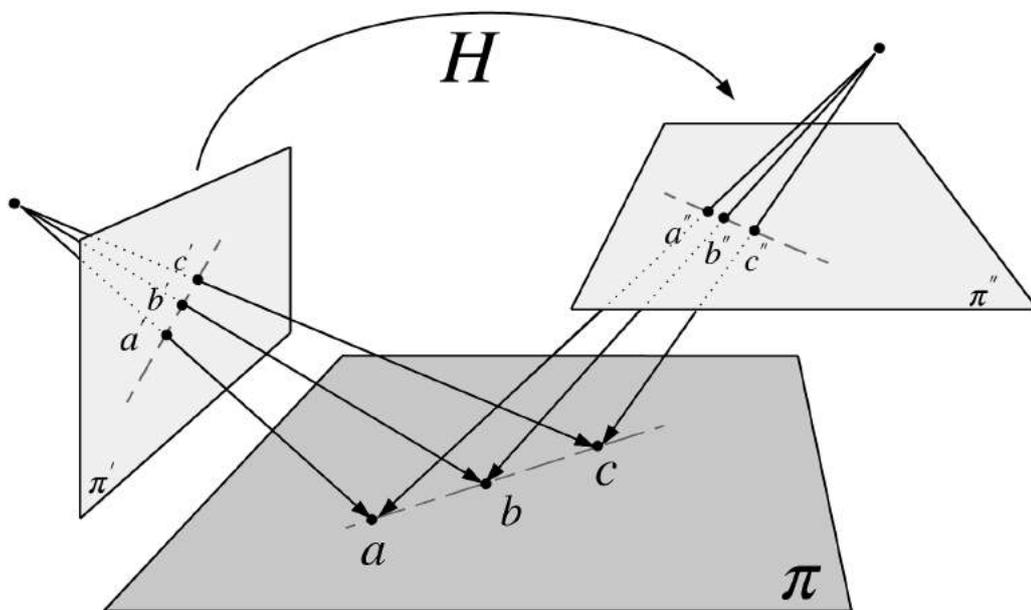


Figura 12 – O raio correspondente a um ponto  $a' \in \pi'$  é estendido até o encontro com o plano  $\pi$  em um ponto  $a$ , este ponto é projetado para um ponto  $a''$  no outro plano  $\pi''$ . O mapeamento de  $a'$  para  $a''$  é uma homografia induzida pelo plano  $\pi$ ; Existe uma perspectividade,  $a' = H_1(a)$ , entre o plano mundial  $\pi$  e o primeiro plano da imagem  $\pi'$ ; e uma perspectividade,  $a'' = H_2(a)$ , entre o plano  $\pi$  e segundo plano de imagem  $\pi''$ . A composição das duas perspectividades é uma homografia,  $a'' = H_2 H_1^{-1}(a') = H(a')$ , entre os planos da imagem  $\pi'$  e  $\pi''$ .

### 3.6 INVARIANTES PROJETIVOS

Um dos problemas centrais da visão computacional consiste em determinar as propriedades geométricas de um objeto utilizando imagens em perspectiva e com base nessas propriedades, efetuar o seu reconhecimento. Dessa forma, conclui-se que é essencial determinar propriedades da geometria do objeto que sejam invariantes perante o processo que rege a formação da imagem. O ponto fundamental será a possibilidade de descrever os objetos em termos desses invariantes. Nesta seção serão introduzidos os conceitos fundamentais sobre invariantes projetivos.

Grupos de Transformações	Invariantes Geométricos
<i>Projetividades</i>	Colinearidade; Concorrência; Ordem de contato: - Inflexão - Intersecção - Tangência Razão cruzada.
<i>Afinidades</i>	Paralelismo; Razão do comprimento de segmentos (Paralelos e Colineares); Razão entre áreas.
<i>Similaridades</i>	Ângulo; Razão de comprimentos.
<i>Isometrias</i>	Área; Comprimento.

Tabela 1 – Invariantes geométricos pertencentes a um determinado grupo da tabela servem também como invariantes para os grupos abaixo desses. Os invariantes projetivos são os invariantes mais gerais, se localizam no topo da tabela.

#### 3.6.1 Razão Cruzada

A razão cruzada (do inglês *cross-ratio*) de quatro pontos colineares é o invariante projetivo mais fundamental: uma razão de comprimentos em uma reta é invariante sob *afinidades*, mas não sob *projetividades*; no entanto, uma razão de razão ou razão cruzada de comprimentos em uma reta é um invariante projetivo (HARTLEY; ZISSERMAN, 2004).

A razão cruzada de quatro pontos colineares  $(P_1, P_2, P_3, P_4)$  dispostos nessa ordem, pode ser definida como:

$$RC(P_1, P_2, P_3, P_4) = \frac{P_1P_3}{P_2P_3} \bigg/ \frac{P_1P_4}{P_2P_4}, \quad (3.5)$$

onde  $P_iP_j$  é a distância entre os pontos  $P_i$  e  $P_j$ .

Podemos perceber que é possível variar a disposição dos pontos para o cálculo da razão cruzada, como a razão cruzada envolve quatro elementos, assim sendo diferentes ordens dos

índices dos pontos produzem  $4! = 24$  permutações possíveis para o cálculo desse invariante; no entanto, apenas seis deles têm valores distintos, e dado um invariante  $RC(P_1, P_2, P_3, P_4) = \lambda$ , todos os outros são funcionalmente dependentes dele, ou seja, podem ser derivados a partir de um valor de razão cruzada. Em resumo, todas as 24 diferentes permutações para realização do cálculo da razão cruzada expressarão apenas 6 valores de razão cruzada que são dependentes de um único valor de razão cruzada (SITTA; PEDROSO; TADINI, 2002). Dessa forma, os 6 únicos valores de razão cruzada em função de  $\lambda$  são:

$$\left\{ \lambda, \frac{1}{\lambda}, 1 - \lambda, \frac{1}{1 - \lambda}, \frac{\lambda - 1}{\lambda}, \frac{\lambda}{\lambda - 1} \right\} \quad (3.6)$$

Existem cinco conjuntos de quatro pontos colineares exibidos na Figura 13, onde cada conjunto está relacionado aos outros por uma *projetividade* linha-a-linha. Uma vez que a razão cruzada é invariante sob uma *projetividade*, a razão cruzada tem o mesmo valor para todos os conjuntos mostrados na figura.

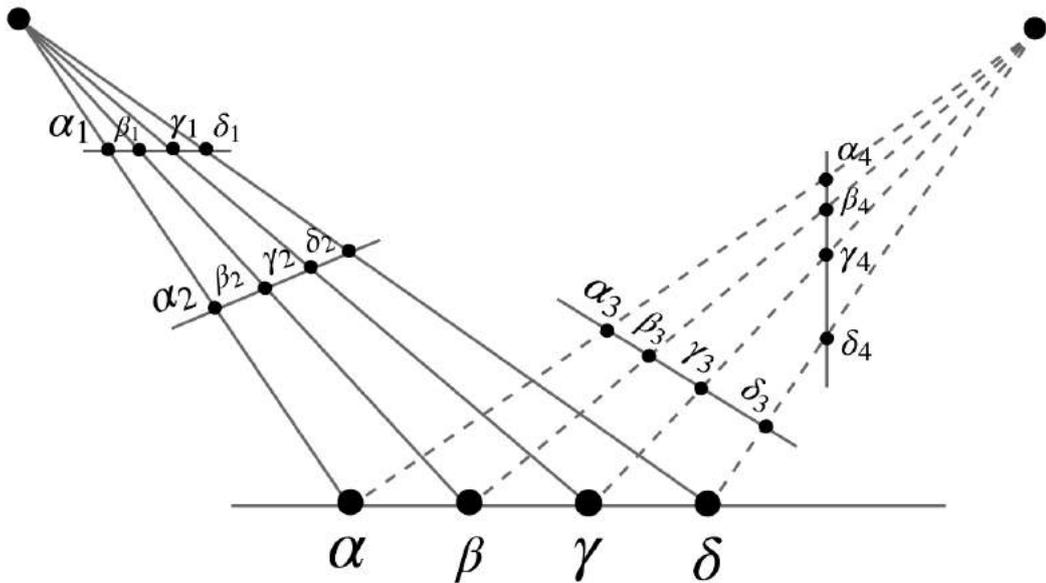


Figura 13 – A razão cruzada obtida pelo conjunto de pontos colineares  $(\alpha, \beta, \gamma, \delta)$  é a mesma para os outros quatro conjuntos de pontos também colineares  $(\alpha_i, \beta_i, \gamma_i, \delta_i)$ , onde  $i \in \{1, \dots, 4\}$ .

### 3.6.2 Razão Cruzada de cinco pontos coplanares

A razão cruzada de cinco pontos coplanares pode ser entendida como um caso especial de razão cruzada do comprimento de quatro segmentos de retas concorrentes com um único ponto de intersecção. Também pode ser entendida como a razão cruzada das áreas de quatro triângulos coplanares com um vértice em comum (SUK; FLUSSER, 2000). A razão cruzada de cinco pontos coplanares  $(P_1, P_2, P_3, P_4, P_5)$  ou da área de quatro triângulos coplanares com o vértice  $P_1$  em comum, pode ser definida como:

$$RC_5(P_1, P_2, P_3, P_4, P_5) = \frac{\Delta(P_1, P_2, P_3)\Delta(P_1, P_4, P_5)}{\Delta(P_1, P_2, P_4)\Delta(P_1, P_3, P_5)}, \quad (3.7)$$

onde  $\Delta(P_i, P_j, P_k)$  é a área do triângulo formado pelos pontos  $P_i, P_j$  e  $P_k$ .

Observe que  $\Delta(P_i, P_j, P_k)$  terá valor nulo se  $P_i, P_j$  e  $P_k$  forem colineares. Sendo assim a razão cruzada não está definida para todo tipo de arranjo geométrico, como aqueles que apresentarem colinearidade de no mínimo três pontos nos termos presentes no denominador da equação 3.7. Outra questão importante a ser discutida é a de que a equação 3.7 é apenas uma das definições para esse invariante; é possível extrair dois desses invariantes projetivos funcionalmente independentes (MARSICO et al., 2012) como ilustrado na Figura 14. Porém não estamos interessados aqui nas permutações desse invariante, isso é apenas para caráter informativo; para o descritor de forma discutido neste trabalho necessitamos apenas de uma única definição desse invariante projetivo.

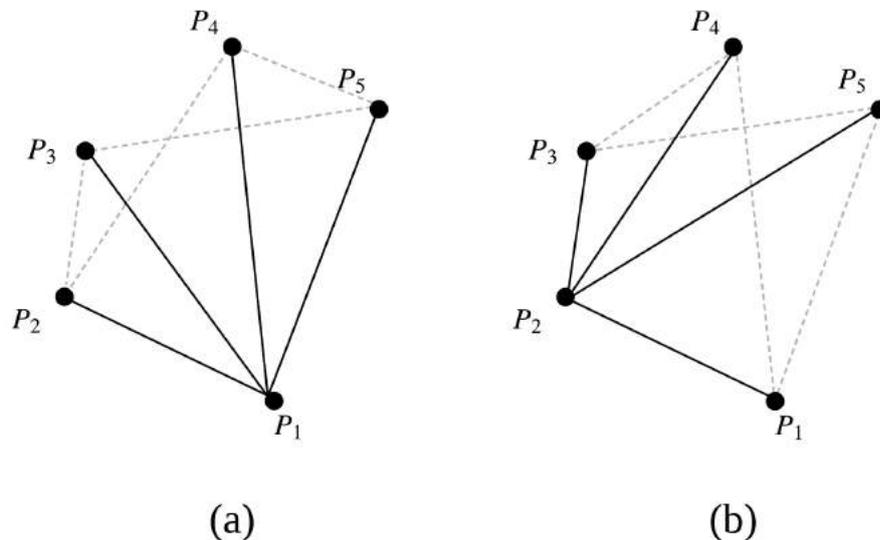


Figura 14 – O mesmo conjunto de cinco pontos coplanares em **(a)** e **(b)** dão origem a dois invariantes projetivos funcionalmente independentes. Cada uma delas podem ser expressas como a razão cruzada de quatro retas concorrentes num ponto base  $P_1$  em **(a)** e  $P_2$  em **(b)**; ou como a razão cruzada de quatro áreas de triângulos formados por combinações dos pontos  $(P_1, P_2, P_3, P_4, P_5)$ , tendo como vértices em comum  $P_1$  e  $P_2$  para **(a)** e **(b)** respectivamente.

### 3.7 DISCUSSÃO

Neste capítulo foram discutidos os princípios da obtenção de imagens pela visão de câmera através do estudo da geometria projetiva e os efeitos de deformação projetiva inerentes ao processo de visão nos objetos da cena.

Foi introduzido o espaço projetivo  $\mathbb{P}^2$ , onde foi mostrado o plano projetivo e os elementos geométricos básicos, como pontos, direções e retas sob a ótica da geometria projetiva. A hierarquia de transformações geométricas foi exibida e suas transformações foram brevemente estudadas. Os invariantes geométricos foram analisados sob o seu ponto de vista prático, e foi mostrado que os invariantes projetivos são mais gerais, isso quer dizer que podem ser utilizados para representar informações visuais extraídas de imagens de maneira mais robusta as diversas situações de cena; em outras palavras, esses invariantes são capazes de descrever objetos nas situações reais com uma flexibilização em discriminar superior aos outros invariantes geométricos. Sendo assim, por fim, foi apresentada a razão cruzada que é o invariante projetivo mais fundamental, e foram expostas suas definições para quatro pontos colineares, e cinco pontos coplanares.

Esses fundamentos servem de base para o entendimento das deformações projetivas que os objetos estão suscetíveis e como é possível extrair e descrever características flexíveis de objetos planares a partir de cenas reais.

---

## 4 FUNÇÕES DE DISTÂNCIA, CORRELAÇÃO E *DYNAMIC TIME WARPING*

---

Este capítulo é uma miscelânea de conceitos, demonstrações e listagem de funções de distância, correlação e a discussão de um algoritmo para comparações de séries temporais.

O acompanhamento do comportamento de um evento específico por tempo pode produzir informações relevantes. Uma grande variedade de aplicações do mundo real, como fisiologia, meteorologia, geologia, astrofísica, coletam observações que podem ser representadas como séries temporais. Uma série temporal pode ser definida como uma coleção  $S = (s_1, s_2, \dots, s_n)$  de dados observados ao longo do tempo, onde  $n$  é o número de observações, em outras palavras é um sinal discreto dependente do tempo. Séries temporais podem ser usadas em áreas que lidam com sinais, como em processamento de imagens. Por exemplo, os valores em sequência das distâncias dos pontos do contorno de um objeto planar, em uma imagem, até o centroide do mesmo, podem ser convertidos para séries temporais, sendo útil na descrição de características de forma (WANG et al., 2008). Então é necessário que existam métricas de comparação entre séries temporais para o problema de correspondência de similaridade. Uma métrica ou função de distância é uma função que define uma distância entre cada par de elementos de um conjunto. As séries temporais são essencialmente vetores de alta dimensão que podem ser comparados com outros vetores de igual dimensão, nesse sentido a função de distância mais utilizada é a Distância Euclidiana. Porém a Distância Euclidiana e suas variantes apresentam algumas desvantagens, que tornam inadequadas em determinadas aplicações (CASSISI et al., 2012). Por estas razões, outras técnicas de medida à distância foram propostas, com maior robustez à computação de similaridade. Neste capítulo são mostradas algumas métricas de distância já consagradas, alguns métodos estatísticos de correlação e técnicas mais complexas que usam programação dinâmica para comparação de séries temporais de maneira mais robusta e flexível às transformações de sinais, como deslocamento, diferença entre números de elementos, variações da amplitude e do tempo.

### 4.1 FUNÇÕES DE DISTÂNCIA

Uma forma de comparar os dados das séries temporais envolve o conceito de funções de distância. Segundo (HAN; PEI; KAMBER, 2012) uma função de distância é uma função que respeita as seguintes quatro propriedades fundamentais:

1. *Identidade*:  $f(x, y) = 0 \iff x = y$ ;
2. *Positivamente definida*:  $f(x, y) \geq 0$ ;

3. *Simetria*:  $f(x, y) = f(y, x)$ ;

4. *Desigualdade Triangular*:  $f(x, z) \leq f(x, y) + f(y, z)$ .

Sejam duas séries temporais  $T$  e  $S$ , vetores de comprimento  $n$ , e  $t_i$  e  $s_i$  os  $i$ -ésimos valores de  $T$  e  $S$ , respectivamente; abaixo é apresentada uma lista de funções de distância.

- **Distância Euclidiana (norma  $L^2$ )**. A mais conhecida e usada função de distância, é definida como:

$$d(T, S) = \sqrt{\sum_{i=1}^n (t_i - s_i)^2} \quad (4.1)$$

- **Distância Manhattan (norma  $L^1$ )**. Também conhecida como “*city block distance*”, porque é a distância baseada em número de blocos (quarteirões) entre dois pontos em uma cidade. É definida como:

$$d(T, S) = \sum_{i=1}^n |t_i - s_i| \quad (4.2)$$

- **Distância de Kolmogorov-Smirnov (norma  $L^\infty$ )**. É definida como o valor máximo de distância dos atributos:

$$d(T, S) = \max_{1 \leq i \leq n} |t_i - s_i| \quad (4.3)$$

- **Distância de Minkowski (norma  $L^p$ )**. As métricas de distâncias anteriores: A Distância Euclideana, a Distância Manhattan e a Distância de Kolmogorov-Smirnov são casos particulares da Distância Minkowski. E é definida como:

$$d(T, S) = \sqrt[p]{\sum_{i=1}^n (t_i - s_i)^p}, \quad (4.4)$$

onde  $p$  é chamada de ordem de distância *Minkowski*. Para Distância Manhattan,  $p = 1$ ; para a Distância Euclideana,  $p = 2$ ; enquanto para a Distância Máxima,  $p = \infty$ .

- **Distância de Hamming**. Esta métrica de distância é baseada na simples desigualdade entres os elementos de uma sequência e outra. E é definida como:

$$d(T, S) = \sum_{i=1}^n (t_i \oplus s_i), \quad (4.5)$$

onde o operador  $\oplus$  funciona de maneira similar a operação *XOR* (*eXclusive OR*) da álgebra booleana, porém é estendido para operandos com valores não binários. Ou seja: Se  $a = b$ , então  $a \oplus b = 0$ ; se  $a \neq b$ , então  $a \oplus b = 1$ . Assim sendo, a distância de Hamming dá como resultado a quantidade de elementos diferentes na comparação termo-a-termo entre as duas sequências.

## 4.2 CORRELAÇÃO

Uma tarefa comum é a estimativa de semelhança entre características de objetos. Uma medida de semelhança que pode ser usada para tal é uma correlação. Em (LIMA; GUEDES; SILVA, 2009) foi usado o coeficiente de correlação de Pearson na comparação de descritores de Fourier de imagens de cartões de inspeção para uso na indústria petrolífera. Uma correlação é qualquer relação dentro de uma ampla classe de relações estatísticas que envolva dependência entre duas variáveis. Intervalos comuns utilizados para mapear tal relação são  $[-1, 1]$  ou  $[0, 1]$ , onde 1 indica o máximo de similaridade.

### 4.2.1 Coeficiente de Pearson

O coeficiente de Pearson mede o grau da correlação linear (se positiva ou negativa) entre duas variáveis de escala métrica. Este coeficiente, normalmente representado por  $\rho$  assume apenas valores entre -1 e 1. O Coeficiente de Pearson entre duas séries temporais  $X$  e  $Y$  de comprimento  $n$ , é definido como:

$$\rho = \frac{cov(X, Y)}{\sqrt{var(X) \cdot var(Y)}}, \quad (4.6)$$

onde  $cov(X, Y)$  é a covariância de  $X$  e  $Y$ , e  $var(X) \cdot var(Y)$  é o produto das variâncias de  $X$  e  $Y$ . Os valores em absoluto de  $\rho$  superiores a 0.9 denotam uma correlação muito forte; no intervalo entre 0.7 a 0.9 indicam uma correlação forte; e nos demais intervalos:  $[0.5, 0.7]$ ,  $[0.3, 0.5]$  e  $[0, 0.3]$  correspondem a índices de correlação moderada, fraca e desprezível, na devida ordem.

## 4.3 DYNAMIC TIME WARPING

*Dynamic Time Warping* (DTW) é uma técnica baseada em Programação Dinâmica que serve para encontrar o alinhamento não-linear ótimo entre duas séries temporais; pode ser usada para alinhar qualquer tipo de dado que respeite uma ordem temporal, como áudio, vídeo e imagens. A princípio a DTW foi usada na comparação entre dois padrões no reconhecimento automático de fala (SAKOE; CHIBA, 1978). A fala é um sinal que varia dependente do tempo e pronúncias de uma mesma palavra poderão possuir diferentes durações e/ou diferentes amplitudes, dependendo da maneira de como estão sendo ditas, baseado na velocidade e intensidade (PEREIRA, 2015). Um método baseado em Programação Dinâmica é um método para resolver um problema complexo, dividindo-o em uma coleção de subproblemas mais simples, resolvendo cada um desses subproblemas apenas uma vez e armazenando suas soluções; a técnica baseada na DTW realiza cálculos para a estimativa de valores locais, que esses servem para calcular outros valores locais e seu acumulo oferece uma medida global que pode ser usado como medida de similaridade entre as duas séries (CASSISI et al., 2012).

### 4.3.1 O Algoritmo

Para obter um caminho de correspondência entre as duas séries temporais serão necessárias uma métrica de distância local e uma outra métrica de distância global; uma métrica de distância local é uma função de distância entre um termo de uma série com um termo da outra série, também conhecida como função de custo; a distância global se refere ao acúmulo global das distâncias locais entre as duas séries temporais. O algoritmo da DTW faz uso de uma matriz de distância acumulada  $D$  em que o valor de cada célula  $D_{ij}$  depende de valores de células imediatamente anteriores a  $(i, j)$  dessa mesma matriz, somada com a distância termo-a-termo das séries. Sejam  $X = \{x_1, x_2, \dots, x_N\}$  e  $Y = \{y_1, y_2, \dots, y_M\}$  duas séries temporais de tamanhos  $N$  e  $M$  respectivamente. Calculam-se os elementos da matriz  $D$  para índices  $1 \leq i \leq M$  e  $1 \leq j \leq N$  da seguinte forma:

$$D(i, j) = d(i, j) + \min \begin{cases} D(i-1, j-1), \\ D(i, j-1), \\ D(i-1, j) \end{cases} \quad (4.7)$$

onde  $d(i, j)$  é a uma função de custo (distância) que, por padrão DTW, é a distância Manhattan.

Como resultado final, a DTW fornece uma matriz com  $M$  linhas e  $N$  colunas e um alinhamento ótimo feito por um caminho entre o primeiro elemento da matriz  $D_{11}$  até o último elemento  $D_{MN}$ . Além disso, o último elemento  $D_{MN}$  representa a soma das distâncias acumuladas do melhor caminho (ou alinhamento ótimo) e pode ser usado como uma medida de similaridade na comparação entre duas séries temporais,  $DTW(X, Y) = D(M, N)$ . Apenas deve-se ter cuidado no uso de tal medida, pois o valor final DTW não obedece a propriedade fundamental de desigualdade triangular, o que formalmente não se constitui uma função de distância (MÜLLER, 2007).

### 4.3.2 Condições de Alinhamento

As tuplas de índices do caminho ótimo podem ser representadas por uma sequência de tuplas de índices  $P = (p_1, p_2, \dots, p_L)$  com  $p_l = (m_l, n_l)$ , onde  $n_l \in [1 : N]$ , e  $m_l \in [1 : M]$  para  $l \in [1 : L]$ . Para encontrar o melhor alinhamento entre as séries, é necessário localizar o caminho através da matriz  $D$ , satisfazendo as seguintes condições:

1. Condição de Borda:  $p_1 = (1, 1)$  e  $p_L = (M, N)$ ;
2. Condição de Monotonicidade:  $n_1 \leq n_2 \leq \dots \leq n_L$  e  $m_1 \leq m_2 \leq \dots \leq m_L$ ;
3. Condição de Continuidade:  $p_l - p_{l-1} \in \{(0, 1), (1, 0), (1, 1)\}$ .

O caminho ótimo  $P = (p_1, \dots, p_L)$  define um alinhamento entre as duas sequências  $X = (x_1, x_2, \dots, x_N)$  e  $Y = (y_1, y_2, \dots, y_M)$ , associando o elemento  $x_{n_l}$  de  $X$  para o elemento  $y_{m_l}$  de  $Y$ . Baseado em (MÜLLER, 2007), sobre as três condições: A **Condição de Borda** exige que

os primeiros elementos de  $X$  e  $Y$ , bem como os últimos elementos de  $X$  e  $Y$  estejam alinhados entre si. O caminho começa no primeiro elemento da matriz e termina no último elemento da mesma, garantindo que o alinhamento não seja parcial em uma das sequências; a Figura 15.(b) mostra essa condição sendo desobedecida: o início e fim do caminho não coincidem com o início e fim da matriz. A **Condição de Monotonicidade** garante que o caminho de mapeamento  $P$  é monotonicamente não-decrescente em seus argumentos dos elementos  $p_l = (m_l, n_l)$ , ou seja, o caminho não voltará sobre si mesmo; na Figura 15.(c) é destacado um trecho do caminho que há um decréscimo no índice da coluna, desrespeitando essa condição. O tamanho do passo expressa um tipo de **Condição de Continuidade**: nenhum elemento em  $X$  e  $Y$  pode ser omitido e não há repetições no alinhamento, o caminho avança um passo de cada vez, evitando saltos, a Figura 15.(d) expõe uma descontinuidade do caminho que infringe a condição citada. Figura 15 ilustra as três condições sendo respeitadas em (a), e sendo violadas em (b), (c) e (d).

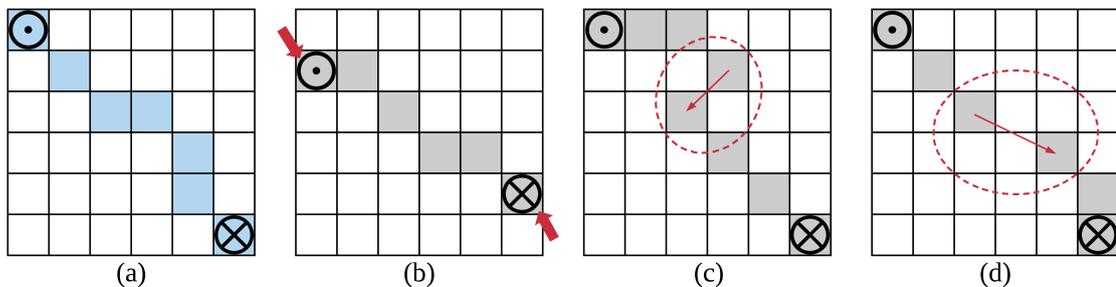


Figura 15 – Exemplos ilustrados de caminhos na matriz  $D$  do alinhamento entre uma sequência  $X$  de tamanho  $N = 6$  e uma outra sequência  $Y$  de tamanho  $M = 6$ ; início e fim do alinhamento são representados por  $\odot$  e  $\otimes$ , nessa ordem. (a) Caminho ótimo satisfazendo todas as três condições de alinhamento. (b) Condição de borda: Violada. (c) Condição de Monotonicidade: Violada. (d) Condição de Continuidade: Violada.

### 4.3.3 Um Exemplo Prático

Para um melhor entendimento da DTW, seguiremos com um exemplo didático. Sejam duas séries temporais  $X = (0, 0, 9, 2, 2, 7, 7, 0)$  de tamanho  $N = 8$  e  $Y = (0, 0, 0, 9, 2, 2, 2, 7, 7, 0)$  de tamanho  $M = 10$ , ilustradas na Figura 16, queremos usar a técnica do DTW para encontrar o melhor alinhamento entre as séries e obter um valor de similaridade.

Inicializamos a matriz de distância acumulada de tamanho  $10 \times 8$ , preenchendo os valores nas células da matriz de distância local com a função de custo  $d(i, j)$  na primeira linha e na primeira coluna da matriz, como ilustrado na Tabela 2, as células estão preenchidas com o módulo da diferença  $|x_j - y_i|$ , onde  $1 \leq i \leq 10$  e  $1 \leq j \leq 8$ , neste exemplo estamos usando a distância Manhattan como função de custo  $d(i, j)$ ; esse procedimento foi repetido para cada elemento  $x_j \in X$  com o primeiro elemento  $y_1 \in Y$  para a obtenção dos valores da primeira linha da matriz, e de forma análoga foi feita para a obtenção da primeira coluna, fixando um único elemento  $x_1 \in X$  e calculando a distância com os outros elementos  $y_i \in Y$ .

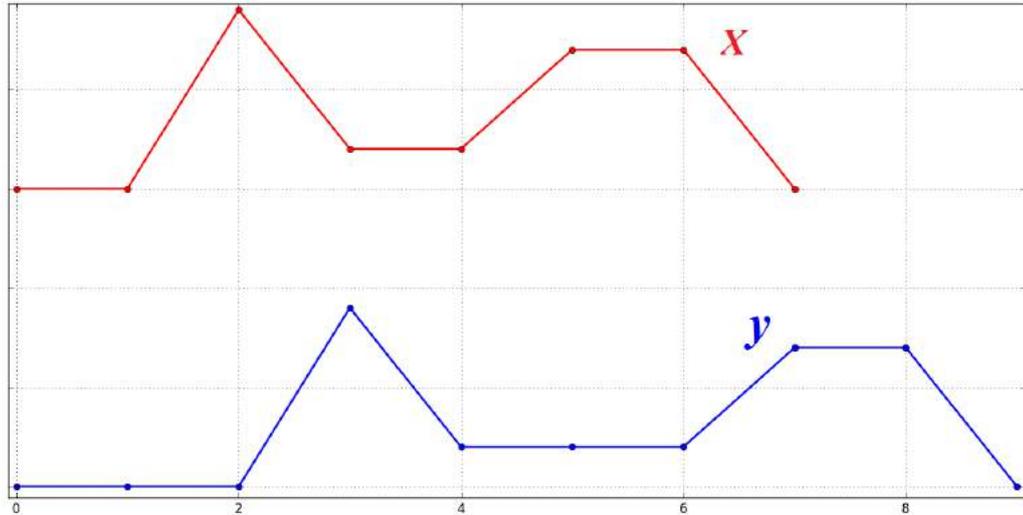


Figura 16 – Exemplo de duas séries temporais que possuem grande similaridade porém seus ritmos de variações são diferentes. O gráfico da série  $X$  está deslocado para cima na abscissa para efeito de melhor visualização.

		0	0	9	2	2	7	7	0
0	10-01	10-01	19-01	12-01	12-01	17-01	17-01	10-01	
0	10-01								
0	10-01								
9	10-91								
2	10-21								
2	10-21								
2	10-21								
7	10-71								
7	10-71								
0	10-01								

Tabela 2 – Matriz de distâncias acumuladas  $D$  de tamanho  $10 \times 8$  em sua primeira iteração do DTW. A sequência de números acima em vermelho pertencem a  $X$ , e na esquerda em azul à sequência  $Y$ ; as células em cinza contêm o módulo da diferença de um elemento  $x_j \in X$  com outro elemento de  $y_i \in Y$ , onde  $1 \leq i \leq 10$  e  $1 \leq j \leq 8$ .

Depois da geração da primeira linha e primeira coluna da matriz  $D$ , é possível usar a Equação 4.7 para calcular os valores das demais células da matriz, cada elemento  $D_{ij}$  é obtido através de um cálculo de distância entre os elementos  $x_j$  e  $y_i$  somando com o valor mínimo dos elementos anteriores ( $D_{(i-1,j-1)}$ ,  $D_{(i,j-1)}$ ,  $D_{(i-1,j)}$ ), como explanado na Figura 17, o valor do elemento  $D_{26}$  é calculado através da distância entre  $x_6$  e  $y_2$  somada com o mínimo entre  $D_{15}$ ,  $D_{25}$  e  $D_{16}$ .

Finalmente, a Tabela 3 mostra todas as células da matriz já preenchidas e ainda o alinhamento ótimo destacado pelas células em azul, respeitando as condições de alinhamento já mencionadas na Seção 4.3.2. O caminho começa do elemento  $D_{11}$ , para a escolha do segundo elemento, deve-se escolher o elemento com valor mais baixo, neste caso existem três possíveis candidatos:  $D_{12}$ ,  $D_{22}$  ou  $D_{21}$  e a escolha de qualquer um desses elementos não irá infringir as condições de alinhamento, sendo assim iremos priorizar o elemento candidato que está a

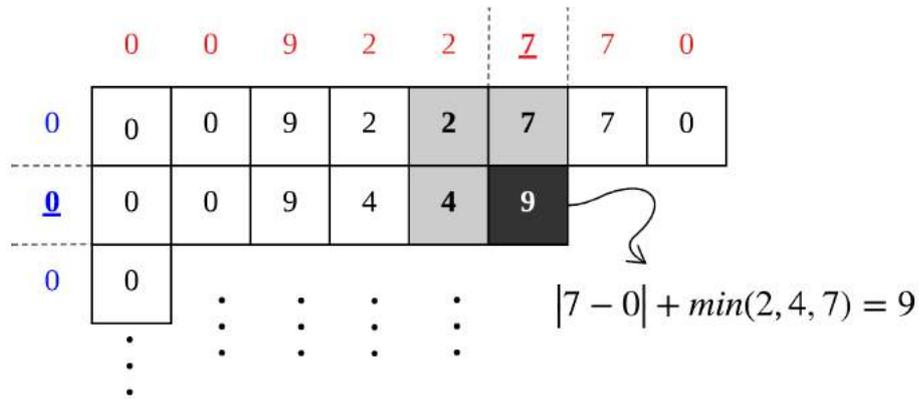


Figura 17 – Subconjunto de células da matriz  $D$  e cálculo de distância local. É exibido em destaque o cálculo para se obter o elemento  $D_{26}$ , que baseado na equação 4.7, tem-se:  $|x_6 - y_2| + \min(D_{15}, D_{25}, D_{16}) = |7 - 0| + \min(2, 4, 7) = 7 + 2 = 9$ .

diagonal de  $D_{11}$ , que no caso é  $D_{22}$ ; a partir de  $D_{22}$  irá haver a mesma análise na vizinhança para a escolha do próximo elemento do caminho, e há somente um elemento possível (que possua o valor mais baixo e que satisfaça as condições já citadas), e que no caso será  $D_{32}$ . Essas operações serão realizadas sucessivamente até alcançar o elemento  $D_{10,8}$ , concluindo assim o caminho completo de associações entre os elementos das séries. O último elemento da matriz  $D_{10,8}$  é a medida DTW entre as duas sequências  $X$  e  $Y$ , que no caso é nulo, isso significa que são sequências idênticas segundo o método.

0	0	9	2	2	7	7	0
0	0	9	4	4	9	14	0
0	0	9	6	6	11	16	0
9	9	0	7	13	8	10	9
2	4	7	0	0	5	10	11
2	4	11	0	0	5	10	12
2	4	11	0	0	5	10	12
7	9	6	5	5	0	0	7
7	14	8	10	10	0	0	7
0	0	9	10	12	7	7	0

Tabela 3 – Matriz de distâncias acumuladas  $D$  resultante e melhor caminho.

Pode-se perceber que neste exemplo o caminho ótimo que respeita as condições de alinhamento não é único, para este exemplo escolheu-se priorizar o elemento da diagonal como critério de desempate, mas também é possível realizar um outro caminho ótimo obedecendo uma outra regra de priorização de elementos vizinhos. Através dos pares de índices do caminho ótimo será possível realizar o alinhamento dos elementos correspondentes das duas séries  $X$  e  $Y$  e liga-los como ilustrados nos gráficos da Fig. 18. Podemos ver que a DTW permite associações de um termo para vários (1 para  $N$ ) e vice e versa.

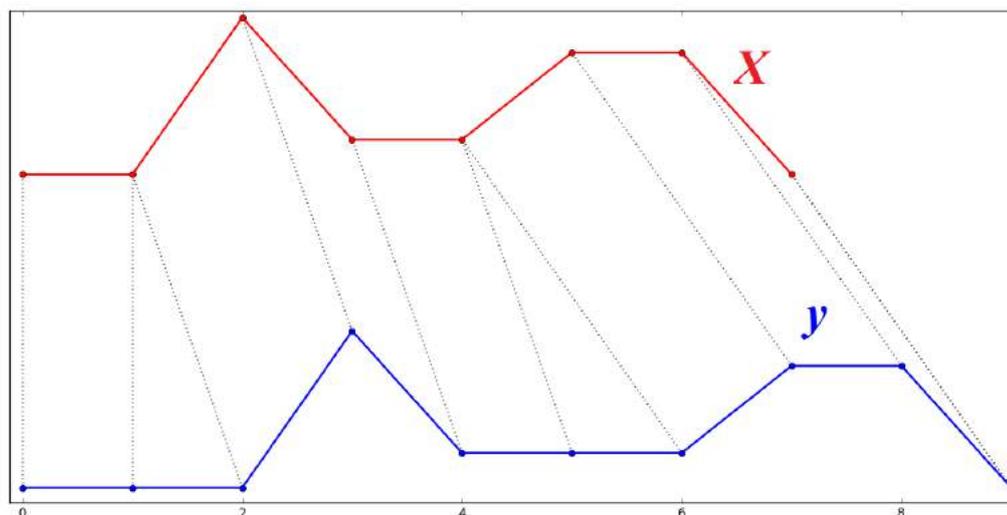


Figura 18 – Alinhamento pela técnica DTW das duas séries temporais  $X$  e  $Y$  do exemplo.

#### 4.4 DISCUSSÃO

Vimos neste capítulo algumas funções que são classificadas matematicamente como métricas ou funções de distância, uma vez que obedecem as propriedades de *identidade*, *não negatividade*, *simetria* e *desigualdade triangular* citadas na Seção 4.1. A Distância Euclidiana é a métrica mais utilizada na comparação de características em mineração de dados, descritores de padrões e imagens. Porém, a Distância Euclidiana e suas variantes apresentam diversas restrições, que tornam inadequado o seu uso em determinadas situações, como: Comparação apenas entre séries temporais do mesmo comprimento; não lida com *outliers* ou ruídos; É sensível a transformações de sinais como: deslocamento, deformação de tamanho linear e não linear de amplitude e tempo. Foi visto brevemente o Coeficiente de Pearson, uma medida de correlação linear que oferece como resultado o quão próximo de uma dependência linear um par de sequências se apresenta. Estudamos uma técnica baseada em programação dinâmica conhecida como *Dynamic Time Warping* e vimos como ela é mais robusta à computação de similaridade. Foi visto que a DTW oferece suporte a comparações de séries temporais de comprimentos diferentes pois, ao contrário da Distância Euclidiana, que faz comparações termo-a-termo, a DTW consegue comparar vários termos de uma série com um único termo da outra, e vice e versa (CASSISI et al., 2012). Como consequência, este método permite reconhecer sinais com formas similares, mesmo que tenham sofrido com deslocamentos e outras transformações.

Todas essas métricas e medidas mencionadas podem servir como ferramentas úteis para eventuais operações de comparação de características usadas em descritores de forma, desde que seu uso esteja apropriado com o tipo de característica que se deseja comparar.

---

## 5 TRABALHOS RELACIONADOS

---

Este capítulo tem o objetivo de revisar e discutir alguns trabalhos relacionados à descrição e classificação de formas planares; serão mencionados trabalhos clássicos da literatura e trabalhos mais atuais ligados diretamente ao problema de reconhecimento de objetos planares com deformações projetivas.

Alguns trabalhos de pesquisa em descritores de forma, como (ZHANG; LU, 2004), (LONCARIC, 1998), classificam diversos descritores pelas seguintes propriedades:

- Baseado em Contorno  $\times$  Baseado em Região
- Global  $\times$  Local (Estrutural)
- Domínio Espacial  $\times$  Domínio de Transformada

Os métodos baseados em região usam informações do interior do objeto, como cor e/ou textura; diferente dos métodos baseados em contorno que consideram apenas as bordas do objeto, negligenciando informações do interior da forma (KAZMI; YOU; ZHANG, 2013). Tanto os métodos Globais e Estruturais subdividem em abordagens baseadas em contorno e região. Existem diversas abordagens que se propõem a descrever características de formas para objetos planares que são invariantes a rotação, translação e escala e até transformações mais gerais como *afinidades* e *projetividades*. Um exemplo de descritor clássico invariante a escala e transformações Euclidianas é a Transformada Generalizada de Hough (BALLARD, 1981), é um algoritmo que usa informações de borda do objeto, como vetores gradientes, para definir um mapeamento num espaço de acumuladores de parâmetros referentes a ângulo de rotação  $\theta$ , localização  $(x, y)$  do centroide e escala  $S$  do objeto, constituindo-se num descritor invariante a *similaridade*. Descritores de Fourier são usados no reconhecimento de formas invariantes a *afinidades*, esses descritores trabalham com informações no domínio da frequência de uma curva planar fechada, componentes de frequências mais baixas são responsáveis pela aproximação da forma da curva, enquanto frequências mais altas descrevem os detalhes mais finos da forma a ser representada.

As abordagens típicas para reconhecimento de símbolos (formas planares) são textura, cor, e forma. Nessas últimas décadas, esforços significativos foram dedicados para características de textura e intensidade dos valores dos *pixels*, entre as quais o descritor SIFT (LOWE, 2004) e suas variantes, por exemplo, SURF (BAY et al., 2008) e PCA-SIFT (KE; SUKTHANKAR, 2004), ganharam grande sucesso. Esses descritores são principalmente derivados com base em informações de texturas locais (JIA et al., 2014). *Scale-Invariant Feature Transform* (SIFT)

é um descritor local afim-invariante que descreve uma região local em torno de um *key point*. Um descritor SIFT é robusto a oclusões e não requer segmentação. A avaliação comparativa de descritores locais baseados em região testada em (MIKOLAJCZYK; SCHMID, 2005) mostrou que o descritor SIFT funciona significativamente melhor do que outros descritores de mesma categoria; em (ROUTRAY; RAY; MISHRA, 2017) são explorados alguns descritores baseados em informações de intensidade, onde são realizados testes com os descritores SIFT, SURF e HOG em imagens com ruído, onde o SIFT se mostrou melhor, obtendo o máximo de pontos de características e pontos associados.

Porém existem numerosos cenários práticos em reconhecimento de imagens onde texturas mais ricas nem sempre estão disponíveis, mas apenas características geométricas distintas, especialmente para o reconhecimento de símbolos artificiais, como caracteres, sinalização de trânsito e logomarcas. Estudos mais recentes em reconhecimento de símbolos (formas planares) (LI; TAN, 2010) mostraram que um descritor SIFT possui baixa discriminação enquanto descritores baseados em contorno discriminam melhor (JIA et al., 2014).

Em (EVANGELISTA, 2016), no seu trabalho sobre monômios invariantes usados para descrever estruturas de junção, ainda sobre os descritores SIFT e SURF, é dito:

*"... não há dúvidas sobre a importância das informações de intensidade e gradiente obtidas a partir de uma vizinhança de interesse para descrever adequadamente a mesma. Mas apesar das regiões de interesse serem consideradas características locais, os descritores não levam em consideração a estrutura dos objetos em seu interior, tornando a invariância a transformações afins e projetivas computacionalmente complicada de se obter. Até mesmo as técnicas estabelecidas para descrever contornos, também locais, consideram contornos inteiros como característica, o que limita a solução para objetos de formato específico que possa ser totalmente segmentado do restante da cena. Além do que, esta última ainda possui limitações quanto a garantir invariância sob transformações afins e projetivas."*

Em meados de 2010 foi lançado um descritor de forma global baseado em contorno, que faz uso de trajetórias retilíneas que partem de pontos das arestas de um polígono convexo que envolve o objeto, tais trajetórias se interceptam com as bordas do objeto formando pontos colineares, dos quais se escolhem apenas quatro pontos para calcular um valor de razão cruzada de cada trajetória; como visto na seção 3.6, a razão cruzada é um invariante projetivo; esse descritor proposto por Li e Tan (LI; TAN, 2010) agrupa diversos valores de razão cruzada em espectros. O método foi apelidado de CRS, sigla do inglês: *Cross Ratio Spectrum* (CRS). O CRS foi bem sucedido para representar formas planares simples, como caracteres que sofreram com deformações projetivas severas. Uma representação para objetos planares invariante a transformações projetivas, baseada em aproximações de partes do contorno do objeto com pares de cônicas foi apresentada em (SRESTASATHIERN; YILMAZ, 2011). O método se mostrou bastante robusto, porém foi testado apenas em formas planares que não apresentavam estruturas internas,

apenas a silhueta. Em (LUO et al., 2013) foi proposto um método parecido com o CRS no uso de trajetórias que cortam o objeto para se calcular um novo invariante geométrico chamada de número característico (*characteristic number*, CN) e foi exibido um novo descritor de forma baseado em valores de CN calculados a partir de pontos colineares, com a vantagem de usar mais pontos colineares para obter um único valor de CN, aumentando o poder de descrição ao incluir mais informações para o cálculo do invariante. Foi mostrado em (JIA et al., 2016) um descritor de forma compacto chamado de *Hierarchical Characteristic Number Context* (HCNC), que se demonstrou robusto a deformações de perspectiva. Esse descritor faz uso do mesmo invariante já mencionado em (LUO et al., 2013), o número característico (CN); o descritor é construído em uma estratégia que parte das representações mais grosseiras para as mais finas que combinam a geometria global dada por invariantes projetivos e informações contextuais locais em um descritor de características hierárquicas. O trabalho de (YANG; WEI; YU, 2018) propõe um descritor de forma baseado em distâncias de centroides (*triangular centroid distances*, TCDs) com o propósito de estimar similaridade entre as formas planares que possuem contornos parcialmente parecidos; os TCDs são invariantes apenas a translação, rotação e escala, apesar dos resultados terem sido melhores que o HCNC em testes realizados para esses propósitos, porém não foram feitos experimentos com formas que sofreram com deformações projetivas.

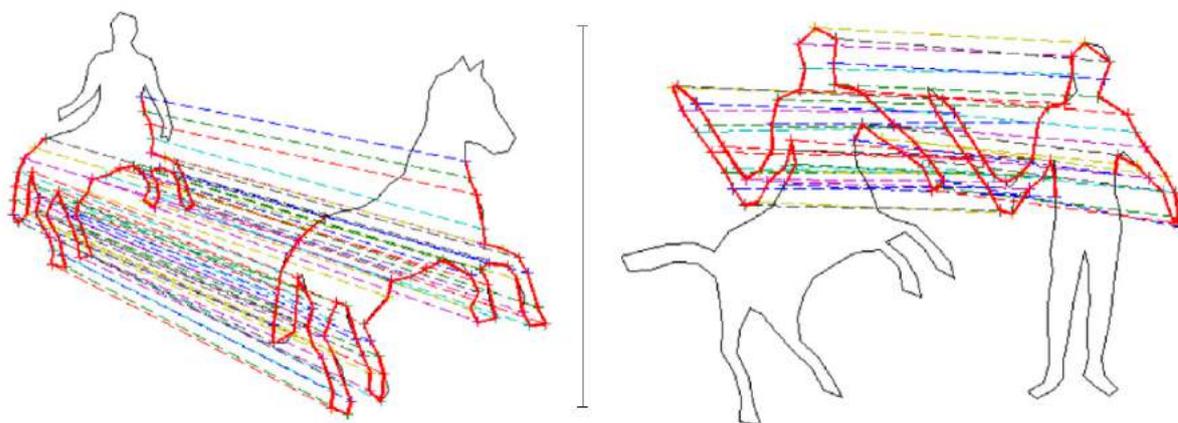


Figura 19 – O trabalho que faz uso dos TCDs possui o propósito de identificar partes semelhantes em objetos comparados; Na esquerda é possível observar as correspondências entre o centauro e o cavalo, na direita as correspondências entre o centauro portando uma espada com uma figura humana também portando uma espada; ambos exemplos retirados de (YANG; WEI; YU, 2018).

## 5.1 DESCRITORES DE FORMA INVARIANTES PROJETIVOS

Diante das diferentes abordagens, reduzimos o escopo para alguns poucos trabalhos que foram considerados concorrentes do método proposto nesta dissertação. Descritores do tipo SIFT, apesar do grande sucesso não são considerados concorrentes, primeiramente por não serem projetivamente invariantes, são invariantes por afinidades apenas; e segundo, por serem descritores baseados em texturas (não em formas), pois como já dito anteriormente, nem sempre haverá situações em que informações de textura estejam presentes na cena, inviabilizando ou impossibilitando o reconhecimento de formas planares através dessa estratégia.

O trabalho desenvolvido em (SRESTASATHIERN; YILMAZ, 2011) que aproxima os objetos planares em cônicas, não foi considerado um concorrente, apesar de ser um descritor de forma planar invariante projetivo, seus testes foram realizados apenas em formas sem estruturas internas. O trabalho realizado em (EVANGELISTA, 2016) desenvolve um invariante projetivo inédito baseado em junções, porém seus resultados ainda são preliminares apesar de alguns serem considerados bons, e não foram realizados testes em formas planares.

Dessa forma, para este trabalho, temos como concorrentes diretos os seguintes descritores de forma projetivamente invariantes: CRS, CN e HCNC. O primeiro foi citado em (LI; TAN, 2010), o segundo e o terceiro descritor foram desenvolvidos pelos mesmos autores, sendo o HCNC uma evolução do CN, com o uso do mesmo invariante projetivo (o número característico), disponíveis em (LUO et al., 2013) e (JIA et al., 2016). Como o terceiro descritor (HCNC) é uma versão mais atual do segundo descritor (CN), restaram apenas o CRS e o HCNC como concorrentes do descritor proposto neste trabalho. Nas seguintes sub-seções, foram explicados brevemente os descritores de forma CRS e HCNC.

### 5.1.1 Cross Ratio Spectrum

O CRS (*Cross Ratio Spectrum*) faz uso de um invariante projetivo já conhecido, a razão cruzada, a etapa de descrição de características desse descritor se baseia na geração de espectros de valores de razão cruzada obtidos através de pontos de intersecção de trajetórias retilíneas, usando apenas 4 pontos ao longo da trajetória para a realização do cálculo da razão cruzada, e para trajetórias que possuam um número inferior a 4 intersecções são usados pseudo-valores de razão cruzada na formação do espectro. As trajetórias são emitidas a partir de pontos localizados no invólucro convexo que envolve o objeto planar, e são emitidos agrupados em diferentes feixes com esses pontos que residem no invólucro em comum, para cada conjunto de trajetórias tem-se os espectros de razão cruzada obtidos. Ao se comparar um descritor CRS com outro, é usado na comparação dos espectros a DTW para o cálculo de distância entre os espectros, vista em detalhes na Seção 4.3, os pontos dos invólucros cujos espectros possuem menor distância, indica que são pontos correspondentes e seu valor é computado para uma métrica de distância final entre os dois objetos planares em comparação. Abaixo, na Figura 20 são mostrados dois caracteres 'H' em diferentes pontos de vista, e os espectros de razão cruzada gerados a partir dos pontos correspondentes  $P_1$  e  $P'_1$ .

Segundo os próprios autores Li e Tan em (LI; TAN, 2010), sobre o descritor:

*"Algumas informações de estruturas internas do símbolo são perdidos usando este método, e em teoria, nós ainda podemos estender o método para empregar outras intersecções da mesma maneira."*

Porém não foi mostrado como seria feita essa "extensão". Ainda segundo os autores do trabalho, em se tratando de oclusão:

*"Uma questão que não foi abordada neste documento é a condição de oclusão. Isto é um problema comum para todos descritores globais."*

A oclusão é realmente um problema em que os descritores globais tendem a sofrer mais em relação aos descritores locais, por usarem informações da totalidade da forma para descrever uma pequena porção da mesma, e a oclusão prejudica por degradar partes da forma que podem ser relevantes na descrição.

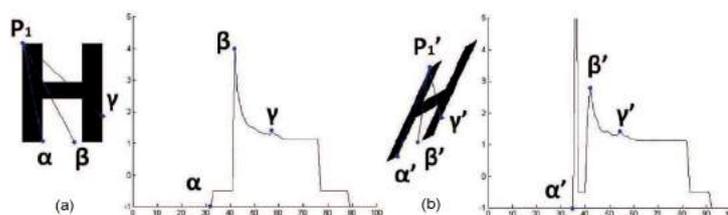


Figura 20 – Espectros de razão cruzada obtidos a partir de trajetórias emitidas de  $P_1$  e  $P'_1$  nos caracteres 'H' na vista frontal-paralela em (a), e com deformação projetiva em (b); retirado de (LI; TAN, 2010).

### 5.1.2 Hierarchical Characteristic Number Contexts

Esse descritor faz uso de um novo invariante projetivo desenvolvido pelos próprios autores (Jia, Luo e outros) batizada de número característicos, expressa pela sigla CN (Characteristic Number) apresentado em (LUO et al., 2013). Segundo os autores, o CN estende a razão cruzada, no sentido de que mais informações estruturais podem ser incorporados no cálculo do invariante, isso significa que é possível usar mais do que quatro pontos colineares para se obter um valor de CN. Os experimentos em logomarcas sob o efeito de transformações afins demonstraram resultados superiores ao CRS (LI; TAN, 2010) e ao descritor SC (*Shape Context*) publicado em (BELONGIE; MALIK; PUZICHA, 2002), porém só no trabalho seguinte que os autores comprovaram que o CN pode ser usado em descritores robustos a deformações projetivas.

No trabalho desenvolvido em (JIA et al., 2016), foi apresentado um descritor de forma compacto nomeado como *Hierarchical Characteristic Number Context* (HCNC), que faz uso do invariante projetivo CN, criado pelos próprios autores em um trabalho anterior, como já dito; além disso foi introduzido um novo invariante apelidado de FCN (*Five Points Characteristic Number*) que faz uso de cinco pontos obtidos da forma planar para o cálculo desse invariante. O HCNC fornece duas modificações que melhoram a robustez ao ruído, oferecendo representações mais estáveis, esse descritor faz uso de uma estratégia grosseira-a-fina (*coarse-to-fine*) que combina geometria global dado por invariantes projetivos e informação contextual local em um descritor de características de baixo nível (*low level feature descriptor*). Na Figura 21 são mostrados alguns exemplos de formas planares que o método se propõe a tratar. Como afirmam os autores, as comparações com o estado-da-arte com outros descritores de forma de baixo nível mostram desempenho favorável ao HCNC perante deformações em perspectiva.

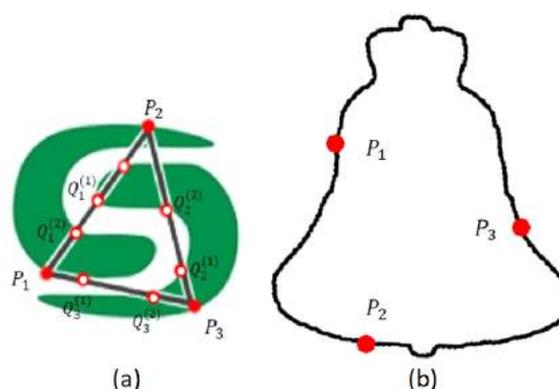


Figura 21 – Formas com e sem estruturas internas. Em (a),  $P_1$ ,  $P_2$  e  $P_3$  são pontos amostrados do contorno mais externo; as intersecções entre as arestas do triângulo  $\Delta P_1 P_2 P_3$  e as estruturas internas da forma geram os pontos  $Q_1^{(1)}$ ,  $Q_1^{(2)}$ ,  $Q_2^{(1)}$ ,  $Q_2^{(2)}$ ,  $Q_2^{(1)}$  e  $Q_3^{(2)}$  que são usado para calcular o CN. Em (b), é mostrado uma forma sem nenhuma estrutura interna, os autores do método HCNC desenvolveram um novo invariante apelidado de FCN (*Five Points Characteristic Number*) que faz uso dos pontos  $P_1$ ,  $P_2$  e  $P_3$  juntamente com mais outros dois pontos obtidos pela intersecção do prolongamento das arestas do triângulo  $\Delta P_1 P_2 P_3$  com as bordas do objeto planar; retirado de (JIA et al., 2016).

## 5.2 DISCUSSÃO

Foi mostrado neste Capítulo uma simples e didática classificação taxionômica de descritores de forma baseados nas diferentes estratégias de representação da forma. Foram citados descritores clássicos da literatura e especificados quanto a seu nível de invariância geométrica. Sobre os descritores mais atuais, do tipo SIFT, que são atualmente bastante utilizados em diversas soluções para problemas de visão computacional, foram exibidas suas limitações para a descrição de formas planares sob efeito de perspectiva, pois os mesmos se baseiam em texturas ricas para uma melhor representação da imagem, e em situações como formas planares artificiais (sinais de trânsito, logomarcas e caracteres), texturas mais ricas dificilmente estão presentes nessas formas, além de se tratar de um descritor invariante afim.

Dos descritores presentes na literatura, foram selecionados dois: O CRS e o HCNC, por se tratarem de descritores de forma invariantes projetivos que foram bem sucedidos na representação e no reconhecimento de símbolos como caracteres e logomarcas nessas condições. O CRS se baseia na razão cruzada para a formação dos seus espectros durante a etapa de descrição do objeto planar, porém faz uso de apenas 4 pontos colineares por trajetória que cruza as bordas do objeto planar a ser descrito para o cálculo do invariante projetivo, tornando mais pobre a representação da forma do objeto, porém o método se saiu muito bem no reconhecimento de objetos planares com baixa complexidade como caracteres. O HCNC é um descritor que faz uso de invariantes projetivos próprios: CN e FCN. Com isso, segundo os autores, o mesmo consegue descrever tanto formas simples como mais complexas com estruturas internas, o invariante CN pode ser obtido através de pontos adicionais sendo mais geral que a razão cruzada, e o FCN é mais utilizado em formas mais simples com pouca complexidade. Ambos os descritores obtiveram bons resultados em seus experimentos, como foram relatados em seus trabalhos.

O descritor proposto neste trabalho se propõe a representar formas planares sob o efeito da deformação projetiva com o uso de invariantes projetivos que serão discutidas mais à frente, dando suporte para o reconhecimento de objetos mesmo sob o efeito da perspectiva. Serão feitos experimentos com diferentes formas em diferentes níveis de deformação projetiva onde serão comparados com os outros descritores concorrentes mencionados através dos valores de acurácia obtidos nos experimentos.

---

## 6 EXTRAÇÃO DE CARACTERÍSTICAS E ESTRUTURA DO DESCRITOR DE FORMA PROPOSTO

---

Neste capítulo é mostrado detalhadamente o processo de extração de características de objetos planares do descritor de forma proposto e sua estrutura de dados de características.

Na etapa de extração de características estamos interessados em representar uma forma planar através de características visuais invariantes sob projetividades. Como visto no capítulo 3, na seção 3.6, características geométricas como **intersecção**, concorrência, **colinearidade**, tangência e **razão cruzada**, podem ser usadas como padrões de representação para a construção de um descritor de forma que seja robusto a transformações projetivas. As características são encontradas seguindo uma abordagem *top-down*: primeiramente são encontradas características mais periféricas da forma pelo invólucro convexo, e através de cada vértice do invólucro são emitidos vários raios, trajetórias retilíneas que se **interceptam** com as bordas do objeto planar, gerando pontos de borda (*edge points*) **colineares**, onde é possível obter valores de **razão cruzada**, e dessa forma é possível descrever também as características internas do objeto planar.

Todas essas características são armazenadas em um estrutura hierárquica que é exibida em detalhes mais adiante.

### 6.1 ELEMENTOS E ESTRUTURA DO DESCRITOR

Nesta seção são definidos alguns elementos do descritor de forma e como estão relacionados estruturalmente. Esses elementos são obtidos durante o processo de extração de característica que é discutido na seção 6.2 e, para facilitar a melhor compreensão desse processo, são primeiramente listados os elementos que compõem o descritor e sua estrutura. A estrutura hierárquica básica do descritor é composta de três níveis: **Invólucro Convexo**, que é formado por um conjunto de **Vértices**, cada um sendo o centro de um feixe de **Raios** que, por sua vez, armazenam características invariantes projetivas, como pontos de borda (colineares) e vetores de razão cruzada. A seguir é conceituado cada um desses elementos.

#### 6.1.1 Vértices do Invólucro Convexo

O **Invólucro Convexo** (*Convex Hull*) é a menor envoltória poligonal convexa que contém todo o objeto, ou seja, é um polígono convexo e, como tal, pode ser representado pelo conjunto de seus vértices e arestas. A coleção de Vértices compõe o Nível 1 da estrutura hierárquica do descritor, o número de Vértices é variado, dependendo da geometria do objeto, e atrelado a cada

Vértice está um conjunto de Raios. O invólucro convexo pode ser substituído também por um **Invólucro de Contorno**, que segue o contorno mais externo do objeto, recomendado apenas para símbolos mais simples como caracteres que não possuem partes descontínuadas; similar ao outro invólucro os pontos que residem no contorno realizam o mesmo papel dos Vértices.

### 6.1.2 Feixes e Raios

Raios são trajetórias retilíneas que partem de um Vértice do invólucro convexo e terminam numa aresta do mesmo invólucro. O número de Raios por Vértice é arbitrário e é igual para todos os Vértices do descritor; esse valor é configurado previamente e funciona como um parâmetro de amostragem do descritor. Cada Raio guarda uma coleção de Pontos de Borda e um Vetor contendo valores de razão cruzada.

Os Raios estão reunidos por Feixes de Raios divergentes que possuem o mesmo Vértice de origem em comum, esses Feixes são também conhecidos por *Fan-Beams*, devido a forma em leque que os Raios estão agrupados. Essas coleções de Raios unidos em Feixes constituem o Nível 2 da estrutura hierárquica do descritor.

### 6.1.3 Características Invariantes Projetivas Fundamentais

No Nível 3, mais baixo nível do Descritor, residem as características mais fundamentais (*Low-Level Features*). Como já estudado no Cap.3: intersecção, colinearidade e razão cruzada são características invariantes a projetividades e neste descritor são representados sob a forma de coleção de Pontos de Borda Colineares e Vetores que armazenam valores de razão cruzada.

#### 6.1.3.1 Pontos de Borda (*Edge Points*)

Uma coleção sequencial de pontos colineares pertencentes a um Raio; esses pontos se caracterizam por serem vizinhos a pixels de fundo (*background*, fora do objeto). O número de Pontos de Borda varia de acordo com o número de intersecções do Raio com os contornos do objeto.

#### 6.1.3.2 Vetor de Razão Cruzada (*Cross Ratio Vector*)

Uma coleção sequencial de valores de razão cruzada obtidos através dos Pontos de Borda; o tamanho do Vetor depende do número de pontos de borda obtidos, para um raio contendo  $K \geq 4$  Pontos de Borda é gerado um Vetor de tamanho  $K - 3$ .

### 6.1.4 Estrutura Hierárquica

Na Figura 22, é exibida a estrutura hierárquica do Descritor. O Nível 1 corresponde ao polígono convexo (*Convex-Hull*) que envolve o objeto constituído de um arranjo de  $M$  Vértices que obedecem uma ordem padrão que é invariante a transformações geométricas; o Nível 2

se trata dos agrupamentos de Raios organizados em *Fan-Beams* emitidos pelos Vértices, cada feixe possuindo um número  $N$  de Raios projetados; o Nível 3, o mais baixo, é onde residem as características mais internas do objeto planar, os  $N \cdot M$  Raios guardam coleções de Pontos de Borda e Vetores de Razão Cruzada obtidos pelo cruzamento dos raios com as estruturas interiores do objeto. Esse tipo de organização hierárquica dos elementos do descritor relaciona características mais externas com características mais internas do objeto planar, como exemplo temos: para cada Vértice do Invólucro Convexo que se encontra na periferia do objeto é lançado um conjunto de Raios que atravessam as estruturas **internas** do objeto, gerando Pontos de Borda; dessa forma foi possível relacionar o elemento Vértice (externo) com vários elementos Raios, e um Raio por sua vez se relaciona com uma coleção de Pontos de Borda (interno). A obtenção de todos esses elementos é explicada na Seção 6.2.

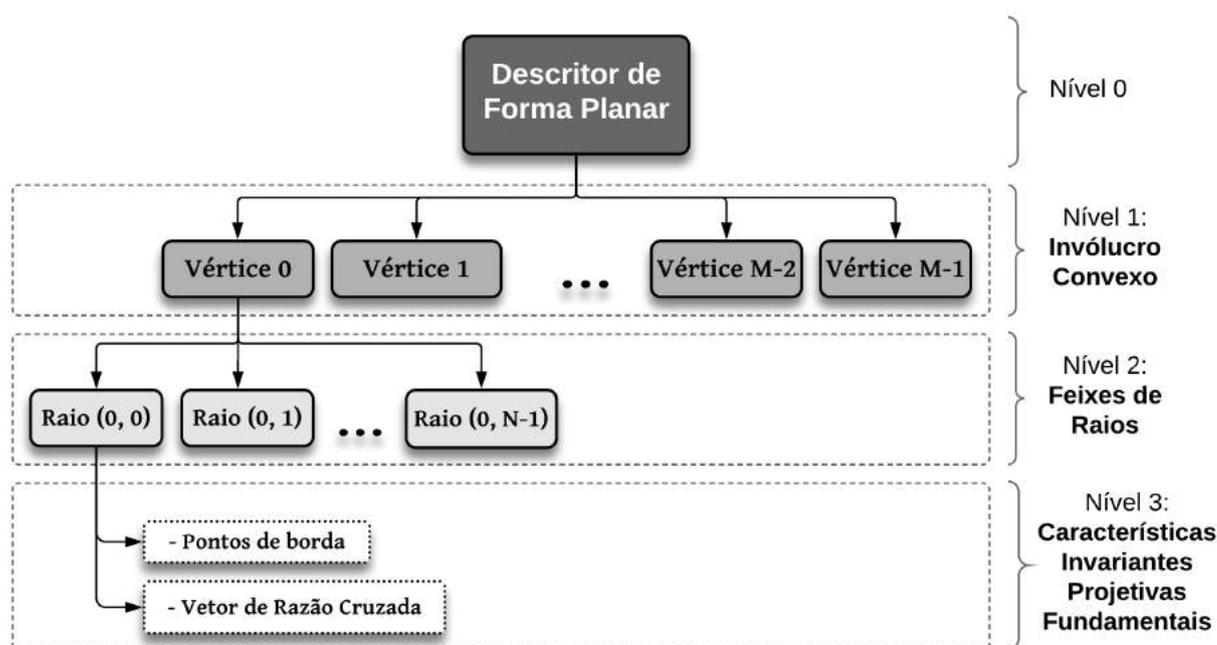


Figura 22 – Estrutura Hierárquica do descritor de Forma Planar proposto. O Invólucro Convexo é formado por  $M$  Vértices constituindo o Nível 1; para cada Vértice é associado um feixe com  $N$  Raios que habitam o Nível 2; e cada Raio possui uma coleção de Pontos de Borda e um Vetor de Razão Cruzada que representam o nível mais baixo do Descritor, o Nível 3.

## 6.2 PROCESSO DE EXTRAÇÃO DE CARACTERÍSTICAS

Como já mencionado anteriormente, a abordagem da extração de características invariantes projetivas do descritor proposto segue uma abordagem *top-down*. Características de mais alto nível são obtidas primeiramente, e a partir dessas são obtidas outras *features* que fornecem mais detalhes do objeto. É importante ressaltar que esse é um descritor de forma baseado em

contorno, e para que haja uma extração de características adequada, é necessário que o objeto nesta etapa esteja segmentado e esteja com os valores de *pixels* binários.

### 6.2.1 Discussão Sobre as Etapas Preliminares

Antes da etapa de Extração de Características, faz-se necessário detectar e isolar o objeto a ser descrito, e para isso são necessárias etapas preliminares de pré-processamento e segmentação, onde diversos filtros de remoção de ruídos podem ser aplicados, filtros detectores de borda podem auxiliar na segmentação, ou uma mudança no espaço de cores e finalmente uma limiarização podem ser aplicadas no intuito de entregar uma imagem com valores de *pixels* binários. Nota-se que existem inúmeras estratégias possíveis com o mesmo resultado, que é o de encaminhar uma imagem binária para a primeira etapa do reconhecimento (Extração de Características); como já discutido no Cap.2 as etapas de Aquisição de Imagens, Pré-processamento e Segmentação não são o foco deste trabalho, mas são explicadas brevemente para um maior entendimento do processo de reconhecimento de objetos como um todo e para chamar a atenção de que a etapa de Extração de Características exige como entrada uma imagem binária. Abaixo, na Fig. 23 é exposto um diagrama de uma aplicação de reconhecimento de objetos planares.

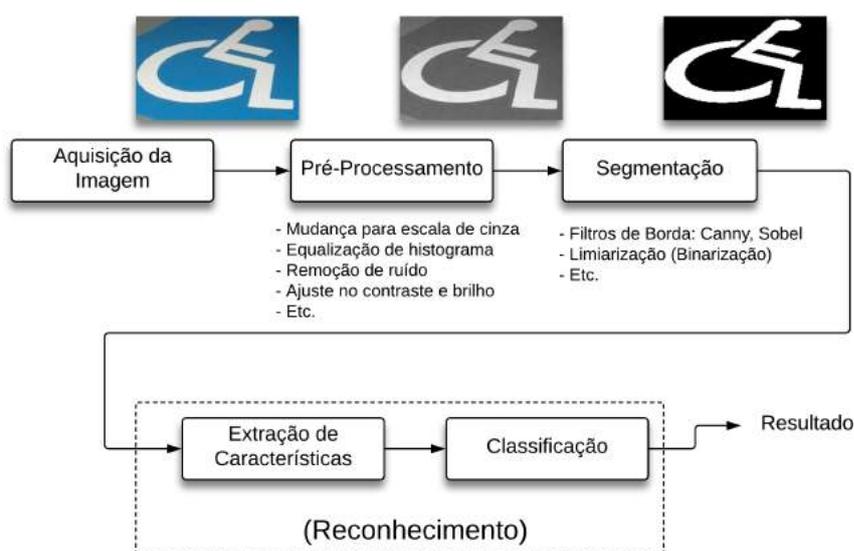


Figura 23 – Ciclo completo de uma aplicação de reconhecimento de objetos planares. As etapas que antecedem a supra-etapa de Reconhecimento tem o objetivo de alimentar a etapa de Extração de Características com uma imagem binária contendo uma forma planar a ser classificada. O resultado final aponta a que classe o objeto detectado faz parte.

Na sub-seção seguinte são estudadas em detalhes as sub-etapas da Extração de Características; são vistos os procedimentos internos do bloco desta etapa, que foram exibidos superficialmente no diagrama da Figura 23.

## 6.2.2 Etapas da Extração de Características

A partir de uma imagem binária contendo uma forma planar realiza-se as etapas para a representação e descrição do objeto planar com características invariantes projetivas.

### 6.2.2.1 Obtenção dos Vértices do Invólucro Convexo e Contorno

A partir das quinas que delimitam os *pixels* constituintes do objeto planar é possível usá-los como pontos de entrada para usar um algoritmo que encontre o *Convex-Hull* do objeto a ser representado. Um algoritmo rápido para a obtenção do invólucro convexo é o *Quickhull*, apropriado quando não há entrada contendo pontos localizados nos extremos (BARBER; DOBKIN; HUHDANPAA, 1996). Uma implementação da técnica escrita pelos autores pode ser encontrada em (BARBER; HUHDANPAA, 1995). O *Qhull* entregará uma coleção de vértices ordenados por ligações mais próximas numa sequência cíclica formando um polígono convexo. Na Figura 24 (Esq.) é exibida uma forma planar envolvida pelo *Convex-Hull* e seus vértices enumerados. Além dos vértices que definem o *Convex-Hull*, é possível inserir pontos extras no perímetro da envoltória com o objetivo de aumentar a amostragem, exercendo a mesma função prática dos vértices originais. O Invólucro de contorno é também mostrado na Fig. 24 (Dir.) envolvendo a letra ‘A’ e funciona de forma similar ao Invólucro Convexo.

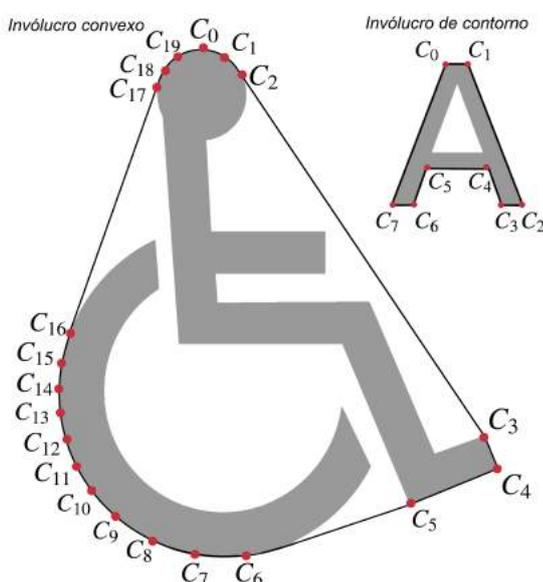


Figura 24 – Invólucro Convexo de um símbolo de acessibilidade: Os índices dos Vértices do Invólucro seguem uma ordenação circular que é invariante a transformações geométricas a menos de um deslocamento de indexação, na esquerda. De maneira análoga é ilustrado o Invólucro de Contorno, na direita, envolvendo o caractere ‘A’.

### 6.2.2.2 Emissão dos Raios em Feixes Divergentes

Com o objeto delimitado pelo Invólucro Convexo é possível usar seus  $M$  vértices como pontos de emissão de Raios, onde o número  $N$  de Raios emitidos por Vértice é dado como parâmetro de entrada. Para cada Vértice haverá uma amostragem de  $N$  pontos distribuídos uniformemente nas arestas do invólucro opostas ao Vértice emissor, esses são chamados Pontos-Alvos (*target-points*); os raios são traçados partindo do vértice atual até os  $N$  Pontos-Alvos.

Como já dito anteriormente o número total de raios do descritor é  $M \cdot N$ ; é possível endereçar um raio através da tupla de índices do seu vértice emissor com o índice do seu ponto-alvo. A disposição dos pontos-alvos segue uma ordem em um sentido conhecido, o que garante uma ordenação no próprio conjunto de raios. Os conjuntos de feixes de raios ordenados com um mesmo vértice de origem são chamados de *Fan-Beams*.

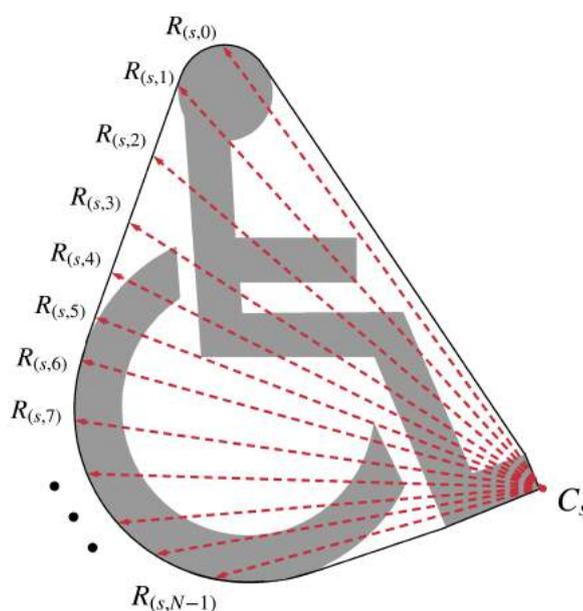


Figura 25 – *Fan-Beam* e seus Raios: São amostrados  $N$  Pontos-Alvos nas arestas não-adjacentes a um Vértice emissor do *Convex-Hull* e a partir das ligações de um Vértice emissor com os Pontos-Alvos são obtidos os Raios. Os Raios são endereçados através da combinação do índice do seu Vértice emissor seguindo do índice do Ponto-Alvo (*Target-Point*); para um Raio que parte de um Vértice  $C_s$  e atinge um *Target-Point*  $t$  é endereçado como  $R_{(s,t)}$ , onde  $t = \{0, 1, \dots, N - 1\}$ .

### 6.2.2.3 Geração de Pontos de Borda Colineares

Para cada raio emitido há intersecções com as bordas da forma planar a ser descrita, onde os pontos de borda surgem das variações de intensidade dos *pixels* ao longo da trajetória do Raio. O número de Pontos de Borda de um Raio pode ser de no mínimo dois até quantos forem possíveis a depender dos cruzamentos realizados com a forma planar. Quanto mais detalhes um símbolo possui, mais Raios com Pontos de Borda numerosos é gerado. Cada Raio possui como resultado final desse procedimento uma sequência de Pontos de Borda colineares. Na Figura

26 são mostrados vários Pontos de Borda que residem nos Raios emitidos pelo Vértice  $C_s$ , e o número de Pontos de Borda por Raio é expresso pela varável  $K$ .

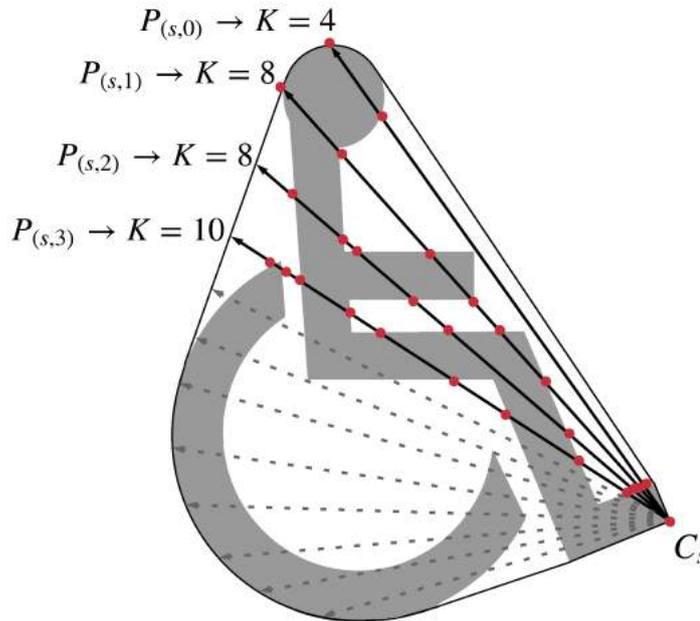


Figura 26 – Raios e seus Pontos de Borda (*Edge-Points*): São mostrados quatro Raios partindo do vértice  $C_s$  contendo coleções  $P_{(s,t)}$  de Pontos de Borda de tamanho variável  $K$ .

#### 6.2.2.4 Cálculo dos Valores do Vetor de Razão Cruzada

O Vetor de Razão Cruzada é uma estrutura gerada através de cálculos realizados com os pontos de borda presentes na trajetória linear (raio), onde cada valor de razão cruzada é calculado a partir de apenas quatro pontos colineares, como já mencionado na Sub-seção 3.6.1 sobre o cálculo dessa invariante; mas um raio pode ter mais do que quatro pontos de borda quando intercepta os contornos do símbolo a ser representado; então, para que haja o máximo poder de descrição sem que haja desperdício de informações é necessário usar todos os pontos de borda do raio, acarretando na obtenção de vários valores de razão cruzada. Neste trabalho criamos esse novo invariante projetivo, chamada de *Cross-Ratio Vector* (Vetor de Razão Cruzada), devido à necessidade de aproveitar o máximo de informação geométrica das trajetórias lineares que cruzam o objeto.

Cada célula do vetor é gerada a partir de dois pontos de bordas extremos (primeiro ponto e último ponto do raio) e outros dois pontos vizinhos intermediários; uma janela de comprimento dois desliza sobre a coleção de pontos de borda intermediários e juntamente com os dois pontos das extremidades, geram um conjunto de valores de razão cruzada. O processo é semelhante a uma filtragem espacial no processamento digital de sinais. Para uma coleção de pontos de borda de tamanho  $K$  a janela deslizante irá utilizar um par de pontos de borda vizinhos intermediários ( $P_1, \dots, P_{K-2}$ ) a cada iteração e agrupando sempre com os pontos de borda das extremidades

$(P_0, P_{K-1})$ , seguindo a fórmula geral da Equação 6.1; dessa forma temos  $K-2$  pontos de bordas intermediários que são agrupados em duplas, respeitando o critério de vizinhança, formando assim  $K-3$  pares de pontos intermediários que, juntamente com os dois pontos extremos, formando  $K-3$  valores de razão cruzada que constituem o vetor.

Assim como os pontos de borda, cada vetor de razão cruzada está atrelado a um raio. É importante ressaltar que os raios que não possuem o número mínimo de pontos necessários para realizar o cálculo da razão cruzada, ou seja  $K < 4$ , não possuirão vetores. Na Figura 27 é exibido um raio com pontos de borda e um exemplo de geração de um vetor de razão cruzada.

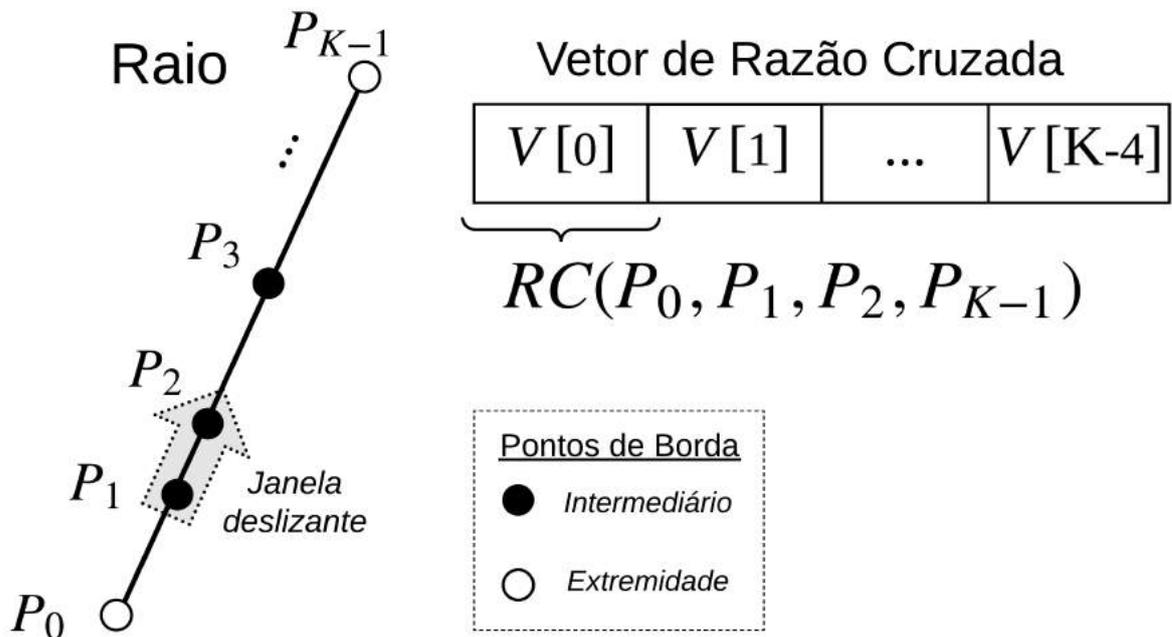


Figura 27 – Geração de valores de um vetor de razão cruzada: Os valores do vetor são obtidos a partir do cálculo da razão cruzada de dois pontos extremos com mais dois pontos intermediários vizinhos coletados pela janela deslizante; para um raio com  $K \geq 4$  pontos de borda é gerado um vetor com  $K-3$  valores de razão cruzada.

Baseado na Equação 3.5 sobre a Razão Cruzada de quatro pontos colineares calculam-se os valores do vetor; os componentes  $V[i]$  do Vetor de Razão Cruzada assumem a seguinte forma geral para raios com  $K$  pontos de borda:

$$V[i] = RC(P_0, P_{i+1}, P_{i+2}, P_{K-1}) \quad (6.1)$$

onde  $i$  é o índice do Vetor de Razão Cruzada e assume os seguintes valores:  $0 \leq i \leq K-4$ . Esta convenção de se calcular a razão cruzada usando primeiramente o primeiro ponto de extremidade  $P_0$  (vértice do invólucro), depois os dois pontos intermediários vizinhos e por fim o último ponto do raio  $P_{K-1}$  evitam-se as numerosas permutações para realização do cálculo da razão cruzada, muitas das quais equivalentes do ponto de vista das razões cruzadas geradas. Além disso, diminui-se o problema de se ter quádruplas com pontos de borda muito próximos, o que acarretaria em um ruído maior.

### 6.2.3 Integração das Etapas

A integração das etapas que compõem a Extração de Características segue uma simples sequência de operações já vistas. Primeiro se extrai o invólucro convexo da forma planar a ser representada, através de cada vértice haverá traçados de raios na segunda etapa, na terceira são extraídos os pontos de borda de cada raio a partir da intersecção do raio com os contornos do objeto e por último são calculados os vetores de razão cruzada já discutidos em 6.2.2.4; na Figura 28 é apresentada a integração de todas essas quatro etapas na devida sequência, e o resultado final é um Descritor de Forma Planar contendo vários elementos organizados numa estrutura hierárquica como já visto em 6.1.

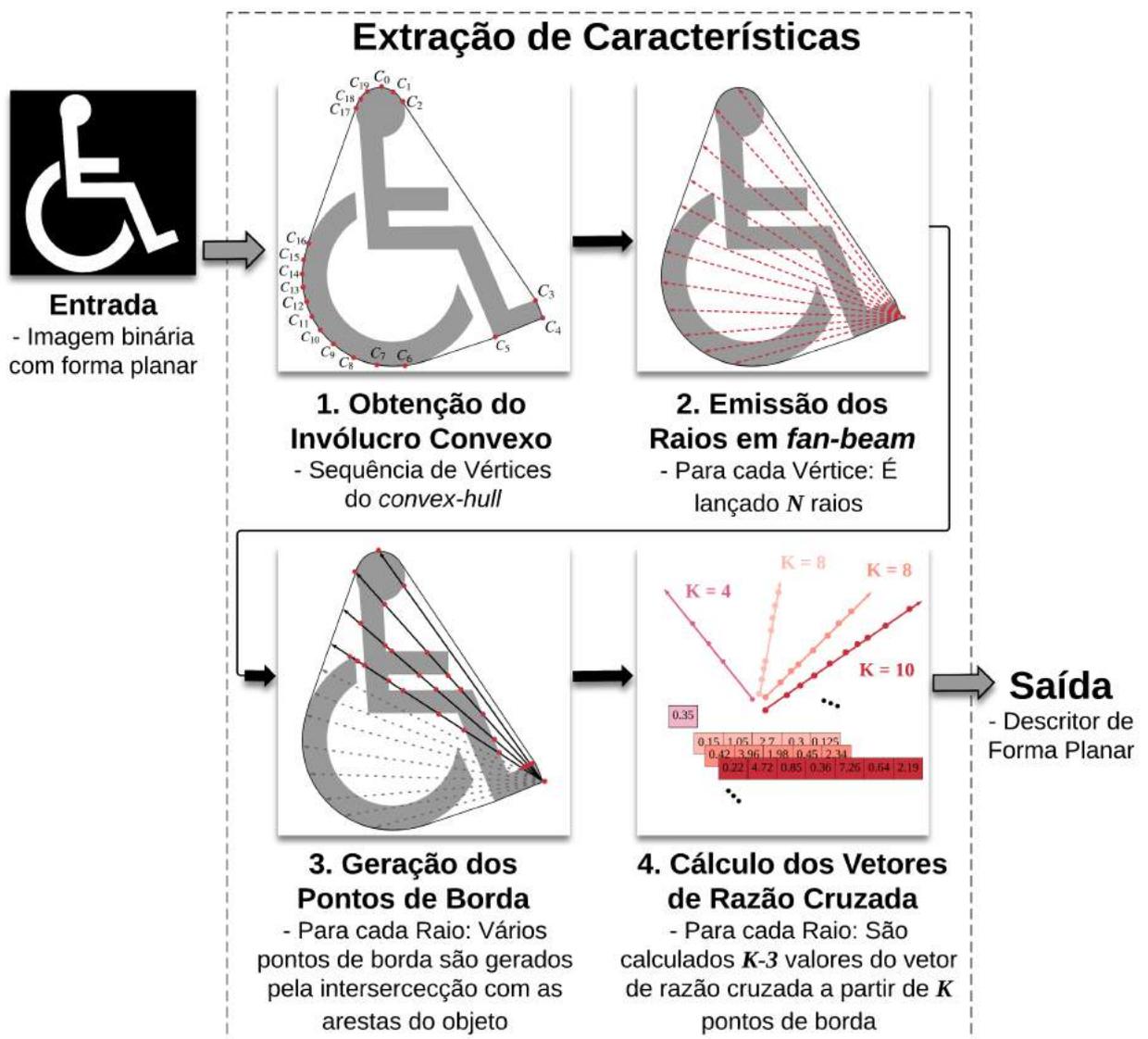


Figura 28 – Etapas da Extração de Características.

### 6.2.3.1 Algoritmo

Abaixo o pseudocódigo explicando todas as etapas do processo de Extração de Características é apresentado. O algoritmo recebe como entrada uma imagem planar representada por  $I$ , e um número  $N$  que representa o número de Raios por Vértice do Invólucro Convexo. O resultado final do algoritmo é uma coleção de  $M$  Vértices do *Convex-Hull* no formato  $C_s$ , e coleções de Raios, Pontos de Borda e Vetores de Razão Cruzada no formato  $R_{(s,t)}$ ,  $P_{(s,t)}$  e  $V_{(s,t)}$ , onde  $0 \leq s \leq M - 1$  e  $0 \leq t \leq N - 1$ .

---

#### Algoritmo 1 Extração de Características

---

**Entrada:**  $(I, N)$  ▷ (Img. Binária, N° Raios/ N° Vértices)

- 1:  $C_{[0, \dots, M-1]} \leftarrow ConvexHullVertices(I)$  ▷ 1. Gerando  $M$  Vértices
- 2: **para todo** Vértice  $C_s \in C$  **fazer**
- 3:  $F_{s[0, \dots, N-1]} \leftarrow FanBeamRays(C_s, N)$  ▷ 2. Lançando Feixe com  $N$  Raios
- 4: **para todo** Raio  $R_{(s,t)} \in F_s$  **fazer**
- 5:  $P_{(s,t)[0, \dots, K-1]} \leftarrow EdgePoints(R_{(s,t)}, I)$  ▷ 3. Encontrando  $K$  Pontos de Borda
- 6: **se**  $K \geq 4$  **então**
- 7:     **para**  $0 \leq i \leq K - 4$  **fazer**
- 8:          $P_0 \leftarrow P_{(s,t)}[0]$
- 9:          $P_{i+1} \leftarrow P_{(s,t)}[i + 1]$
- 10:         $P_{i+2} \leftarrow P_{(s,t)}[i + 2]$
- 11:         $P_{K-1} \leftarrow P_{(s,t)}[K - 1]$
- 12:         $V_{(s,t)}[i] \leftarrow RC(P_0, P_{i+1}, P_{i+2}, P_{K-1})$  ▷ 4. Calculando Razão Cruzada
- 13:     **fim para**
- 14:     **fim se**
- 15: **fim para**
- 16: **fim para**

**Saída:**  $\{C_0, \dots, C_{M-1}\} \cup \{R_{(0,0)}, \dots, R_{(M-1, N-1)}\} \cup \{P_{(0,0)}, \dots, P_{(M-1, N-1)}\} \cup \{V_{(0,0)}, \dots, V_{(M-1, N-1)}\}$

---

### 6.3 DISCUSSÃO

O objetivo do processo de Extração de Características apresentado neste trabalho é primeiramente representar características invariantes projetivas em elementos componentes estruturais de um descritor de forma planar que oferecem suporte à comparação com outros elementos (do mesmo tipo) pertencentes a um outro descritor, além da redução de informação que a "conversão" de uma imagem para um Descritor proporciona. A abordagem da Extração de Características segue uma abordagem *top-down*, começando de características mais periféricas de alto-nível, como invólucro convexo, até características de mais baixo-nível como pontos colineares de borda.

Neste capítulo foram listados e conceituados os elementos que compõem o descritor de forma planar invariante a transformações projetivas. A organização dos elementos do descritor foi exibida seguindo uma estrutura hierárquica que possibilita conectar características mais exteriores com características mais interiores do objeto planar a ser representado. Foi discutido brevemente a importância das etapas anteriores a esta e a necessidade de ter como entrada para a etapa de Extração uma imagem binária contendo uma forma planar. As etapas da Extração de Características foram explicadas em detalhes, a obtenção do Invólucro Convexo, o lançamento de Raios por cada Vértice, a geração dos Pontos de Borda obtidos pela intersecção dos Raios com os contornos do objeto e o cálculo de valores de Razão Cruzada feito de modo a aproveitar todos os Pontos de Borda colineares do Raio, aumentando o poder de descrição, formando um vetor desses valores invariantes projetivos.

O descritor de forma planar proposto apresenta semelhanças ao CRS relativo a emissão de Raios a partir de Vértices que residem no invólucro convexo e aos pontos de borda obtidos pelas intersecções entre o objeto e os Raios emitidos, a diferença nesta etapa é que formam-se vetores de razão cruzada ao invés de fazer uso apenas de um único valor de razão cruzada ao longo da trajetória nesta etapa de extração de características. É exibido no próximo capítulo outras diferenças além dessas mencionadas entre o descritor proposto e o CRS.

Foi visto neste capítulo que o descritor proposto permite realmente representar formas planares com características invariantes a projetividades, oferecendo assim também suporte à classificação mais robusto entre formas planares; a etapa de classificação é vista em detalhes no próximo Capítulo.

---

## 7 COMPARAÇÃO ENTRE DESCRITORES DE FORMA E CLASSIFICAÇÃO

---

No Capítulo anterior foi apresentado o descritor de características que se assemelha a uma árvore e foram estudadas em detalhes as operações necessárias para a extração de características da forma planar, e como tais características ficam organizadas no descritor. O processo de extração e descrição possui um comportamento *top-down*, ou seja, primeiro se extraem características da forma e essas são representadas em elementos do nível mais acima do descritor. Terminada a representação de um determinado nível desce-se para o nível seguinte e se extraem novas características que são representadas por elementos neste novo nível. Assim, primeiro se extrai a coleção de Vértices do invólucro convexo, e depois os armazena no primeiro nível do descritor e, para cada Vértice, são extraídas coleções de Raios que, por sua vez, para cada Raio é obtida uma sequência de Pontos de Borda que é armazenada no descritor nos seus devidos níveis. Neste Capítulo é mostrado como são realizadas as operações de comparação e eventuais correspondências (*matchings*) entre os elementos do descritor invariantes sob projetividades, proposto para duas formas planares em comparação.

Essas comparações aproveitam a estrutura hierárquica do descritor, porém o sentido de comparação segue uma orientação diferente da Extração de Características: primeiramente há comparações entre elementos dos níveis mais baixos do descritor, e as correspondências obtidas entre esses elementos do nível inferior ajudam a encontrar melhores correspondências entre os elementos no nível acima seguindo uma abordagem *bottom-up*; isto é, os Raios são primeiramente comparados, porém não seguindo qualquer ordem, os Raios de um descritor são comparados com os **Raios** do outro descritor por agrupamentos de Raios ligados aos seus **Vértices** de origem, e as correspondências entre **Raios** ajudam a encontrar as melhores correspondências entre os **Vértices** dos descritores das duas formas planares contrapostas. Maiores detalhes são vistos mais à frente; como resultado, as comparações desse nível determinam qual é a melhor correspondência de Vértices, e por fim a disposição e ordem das correspondências desses Vértices é avaliada obtendo uma medida global de semelhança entre as duas formas ou descarte imediato.

Neste Capítulo são mostradas as etapas de comparação e análise dos elementos entre descritores de formas planares semelhantes ou não; as métricas e medidas estudadas no Capítulo 4 são agora utilizadas com o propósito de se obter valores de semelhança entre elementos dos dois descritores e também se extrair uma medida final de comparação entre os mesmos.

## 7.1 COMPARAÇÃO ENTRE ELEMENTOS DOS DESCRITORES

Nesta Seção são exibidas as operações de comparação entre elementos componentes dos descritores vistos no Capítulo 6. É discutida a metodologia usada para comparar elemento-a-elemento de cada descritor com o outro.

Nesta Seção são exibidas as comparações entre elementos do descritor de diferentes formas, similares ou não, é mostrada a metodologia para a comparação entre esses componentes baseado em cálculos para estimar a semelhança entre esses, onde no final há "casamentos" entre os elementos de diferentes descritores baseado na melhor associação, e sem ambiguidades quanto às correspondências.

### 7.1.1 Comparação Raio-a-Raio

Os Raios são agrupados em Feixes de Raios pertencentes a um Vértice emissor em comum, como já dito no capítulo anterior. Para realizar a comparação de Raios é necessário comparar os Feixe-a-Feixe. E dentro desses Feixes tentar encontrar qual é a melhor correspondência de Raios baseada no número de Pontos de Borda e Vetores de Razão Cruzada. A comparação entre Raios começa comparando simplesmente o número de Pontos de Borda, e depois há a comparação entre Vetores de Razão Cruzada.

#### 7.1.1.1 Comparação entre Vetores de Razão Cruzada (vetor-a-vetor)

Como cada Vetor de Razão Cruzada está sempre relacionado a um Raio, a comparação entre Vetores de Razão Cruzada determina diretamente se há correspondência entre os mesmos.

Os vetores são comparados por distância Euclidiana com um valor de tolerância total baseado no somatório dos valores de razão cruzada multiplicados por um valor de tolerância total do vetor tido como o padrão (*template*). Se o valor da distância Euclidiana entre os Vetores de Razão Cruzada for inferior à tolerância total considera-se como sendo correspondentes e, caso contrário, descarta-se; para uma maior garantia de correspondência verdadeira-positiva entre Raios, foi feita uma última comparação de valores de razão cruzada baseado nos Pontos extremos ( $P_0, P_{K-1}$ ) e no segundo e penúltimo Pontos de Borda ( $P_1, P_{K-2}$ ), onde  $K$  é o número de Pontos de Borda do Raio. O ideal seria usar todas as combinações possíveis de Pontos de Borda de um Raio na comparação entre Raios, mas para efeitos práticos essa abordagem é desnecessária e bastante custosa, então usamos a combinação padrão já explicada no Capítulo anterior em 6.2.2.4 construindo um Vetor de Razão Cruzada mais um valor extra de razão cruzada dos Pontos mais extremos do Raio,  $RC(P_0, P_1, P_{K-2}, P_{K-1})$ . Esse valor extra de razão cruzada ajudou a reduzir o número de falsos positivos na comparação entre Raios. É possível visualizar na Figura 29 todo o processo de comparação entre dois Vetores de Razão Cruzada; a rotina de comparação de Raios recebe como entrada Vetores de Razão Cruzada pertencentes ao Descritor da imagem *Template* e ao Descritor da imagem *Query*,  $V$  e  $V'$  respectivamente, e são mostra-

das as etapas de cálculo de distância euclidiana, cálculo da tolerância total e cálculo de valores extras de razão cruzada e, por fim, um valor *Booleano* expressando o resultado da comparação.

Por fim há uma remoção nas associações que possuem ambiguidades, ou seja, os Raios do *Template* que tiverem mais de uma associação, e também os Raios da imagem *Query* que tiverem mais de uma associação, são eliminados deixando apenas as "melhores" correspondências de pares de Raios sem repetições. A abordagem é escolher simplesmente as que possuírem a menor distância euclidiana entre os Vetores de Razão Cruzada sobrando apenas associações únicas (1 : 1) entre os Raios.

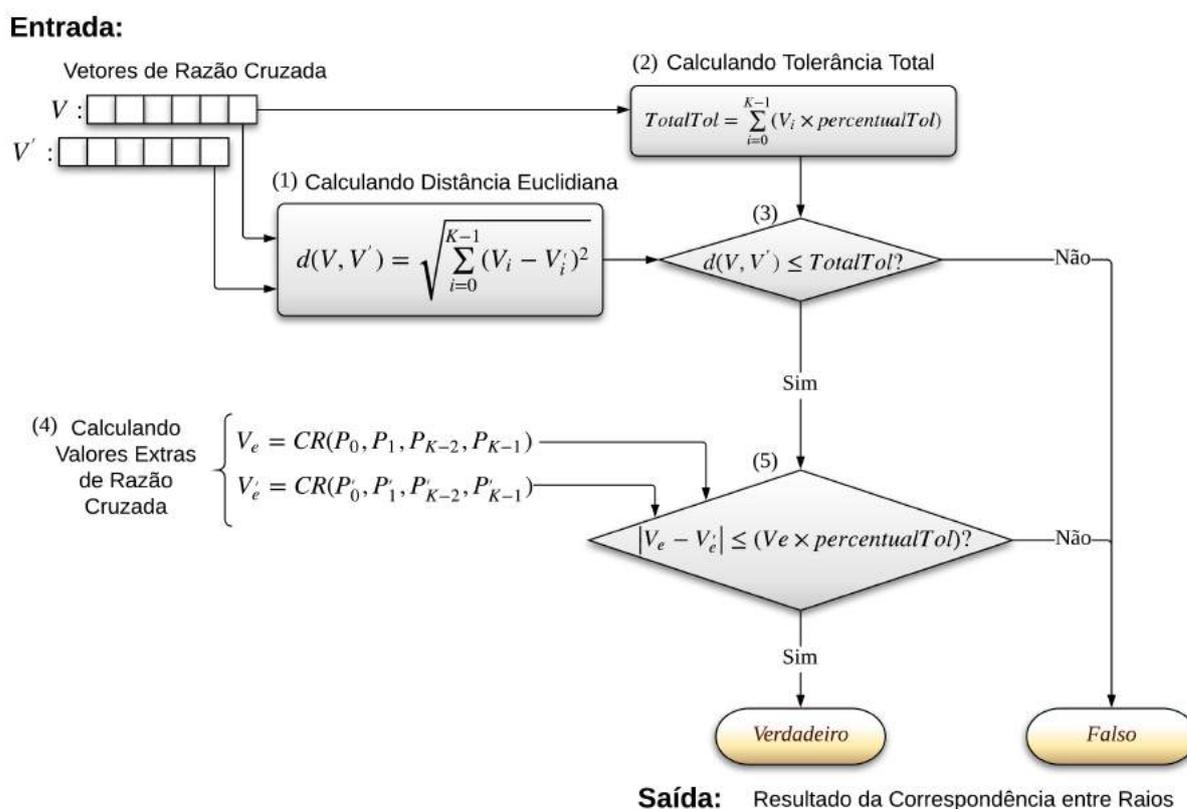


Figura 29 – Comparação entre dois Raios por Vetores de Razão Cruzada. Em (1) é realizado o cálculo da distância euclidiana entre os Vetores ( $V, V'$ ) fornecidos; o cálculo de tolerância total em (2) é obtido através do somatório dos elementos do vetor  $V$  com um valor de percentual de tolerância dado (por padrão: 10%); em (3) é verificado se a distância  $d(V, V')$  calculada em (2) está dentro da tolerância total; se sim, há a próxima verificação em (5), caso contrário, o resultado da correspondência já é tido como **Falso** entre esses Raios; em (4) são calculados os valores extras de Razão Cruzada ( $V_e, V'_e$ ) usando dois Pontos de Borda iniciais e dois Pontos finais para ambos os Raios; e em (5) é realizada a comparação entre a diferença absoluta dos valores extras de Razão Cruzada  $|V_e - V'_e|$  com o produto da tolerância percentual com  $V_e$ ; o resultado dessa comparação irá determinar se o valor da correspondência entre esses Raios é **Verdadeiro** ou **Falso**.

## 7.1.2 Comparação Vértice-a-Vértice

Após a comparação e conseqüente associações entre os Raios de feixes de diferentes descritores, há um conjunto de Raios que foram associados e sem ambigüidades. Como já dito anteriormente, as correspondências entre elementos de diferentes descritores obtidas num nível ajudam na comparação entre elementos num nível imediatamente superior. Em outras palavras, como um conjunto de Raios (Feixe ou *Fan-Beam*) está ligado a um Vértice residente do nível acima, os pares de Feixes de Raios correspondentes ajudam a encontrar os pares de Vértices correspondentes entre os Descritores dos Objetos Planares em comparação; resumidamente: encontrando a melhor correspondência entre Feixes de Raios encontra-se a melhor correspondência entre Vértices.

### 7.1.2.1 Comparação entre Feixes de Raios (Feixe-a-Feixe)

Assim como a comparação entre Vetores de Razão Cruzada está diretamente associada à comparação entre Raios, de maneira análoga a comparação entre Feixes está associada à comparação entre Vértices. Porém a comparação entre Feixes é feita aproveitando as correspondências individuais entre Raios já discutidas em 7.1.1, e depois dessa etapa o conjunto de Raios agrupados pelo mesmo Vértice em comum (Feixe de Raios ou *Fan-Beam*) é comparado de maneira coletiva com o Feixe de Raios do outro Descritor; esse tipo de comparação leva em conta não só a **quantidade** de *matchings* entre Raios, mas também considera a **ordem** dos Raios: uma característica importante a ser utilizada como um atributo que aumenta o poder comparativo entre os Feixes de Raios.

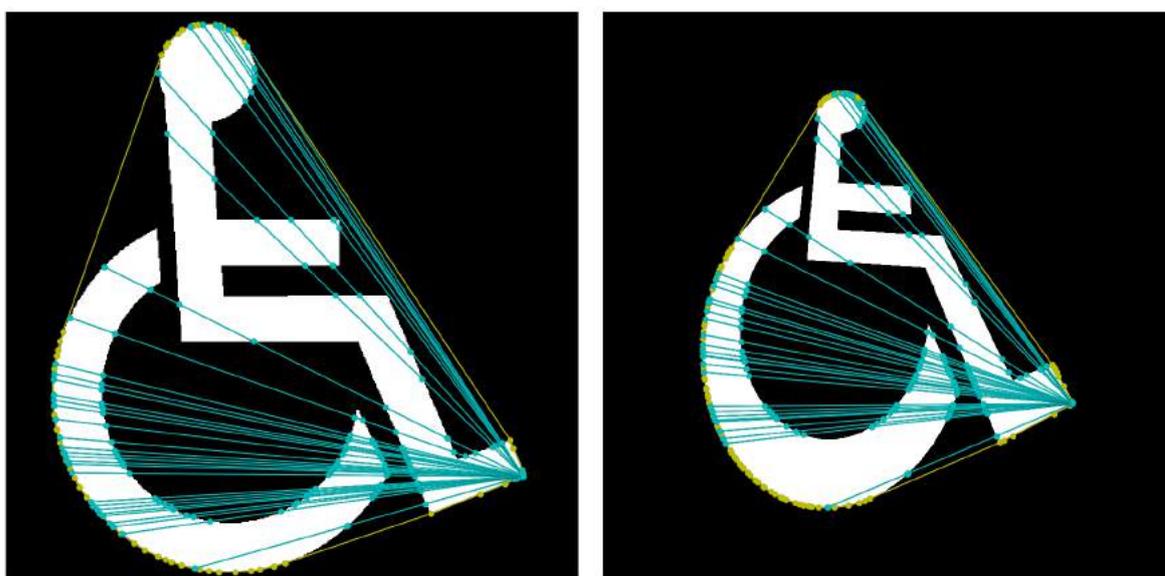


Figura 30 – Feixes de Raios que obtiveram correspondências durante a comparação entre os mesmo Feixes entre dois símbolos de mesma classe.

### 7.1.2.1.1 Formação de espectro de topologia de Raios correspondentes

A forma como as características de quantidade e ordem dos Raios correspondentes entram na comparação Feixe-a-Feixe começa pela construção de *espectros de topologia* de Raios que foram correspondidos durante a comparação entre seus Vetores de Razão Cruzada.

Para a construção desses espectros (sinais, ou séries) são usados Raios que apresentaram correspondência, mais precisamente as informações topológicas desses Raios; entende-se por informação topológica dos Raios o número de Pontos de Borda associado a cada Raio. Então o espectro de topologia que estamos formando é um espectro de número de Pontos de Borda de cada Raio associado: é um sinal contendo sequências de números inteiros que podem ser comparadas usando alguma técnica de comparação entre sequências. É importante ressaltar que os espectros gerados podem possuir distorções causadas pelas deformações projetivas, e que sequências de Raios correspondentes raramente estão em posições correspondentes, mesmo quando a comparação estiver associada a Vértices verdadeiramente correspondentes.

Para um melhor entendimento é ilustrada na Figura 31 a comparação de Feixes de Raios entre duas formas planares de mesma classe: as duas correspondendo a um mesmo símbolo de acessibilidade, apresentado sob vista paralela frontal na Figura 31(a), e deformado projetivamente na Figura 31(b), em que cada uma foi atravessada por um Feixe divergente partindo dos Vértices  $V$  em (a) e  $V'$  em (b). Notam-se os Raios indexados por letras gregas em ambos os símbolos: esses Raios obtiveram correspondência durante a comparação Raio-a-Raio e, como resultado, o Raio  $R_\alpha$  na Figura 31(a) possui um correspondente  $R_{\alpha'}$  na Figura 31(b), e o  $R_\beta$  possui o  $R_{\beta'}$  como correspondente, e assim por diante; tem-se então dois conjuntos de Raios:  $R_\alpha, R_\beta, R_\gamma, R_\delta$  e  $R_\epsilon$  em (a) e seus correspondentes  $R_{\alpha'}, R_{\beta'}, R_{\gamma'}, R_{\delta'}$  e  $R_{\epsilon'}$  em (b); o número de Pontos de Borda de cada um desses Raios compõe os valores dos espectros. Para o Raio  $R_\alpha$  temos oito Pontos, esse valor faz parte do espectro de topologia em (a) na posição reservada para  $R_\alpha$ , e também o mesmo acontece com o seu correspondente  $R_{\alpha'}$  com mesmo número de Pontos de Borda no espectro em (b). São mostrados em ambos os gráficos os rótulos dos Raios com seus índices em letras gregas relativos a cada valor inteiro de Pontos de Borda: os correspondentes  $R_\alpha$  e  $R_{\alpha'}$  possuem ambos o mesmo número de Pontos de Borda, no caso oito, e esse valor estará presente em ambos os espectros ( $S$  e  $S'$ ), o mesmo valendo para os outros Raios rotulados (que possuem correspondência).

Para os Raios que não tiveram *matchings* os valores que compõem os espectros são 0 para o espectro em (a), e 1 para o espectro em (b); essa diferença de valor tem o propósito de penalizar as posições do espectro que não obtiveram correspondências durante a comparação Raio-a-Raio. Então, para isso adotou-se uma convenção para esses Raios não pareados; para essas posições do espectro pertencentes à imagem *template* foi atribuído o valor 0, enquanto aos pertencentes à imagem *query* foi atribuído o valor 1.

Os espectros para os casos da Figura 31 possuem semelhanças e diferenças; devido à transformação projetiva sofrida na Figura 31(b) o espectro possui certas deformidades: é possível ver entre os Raios  $R_\alpha$  e  $R_\beta$  que existem dois Raios não correspondentes (com valor 0) em (a),

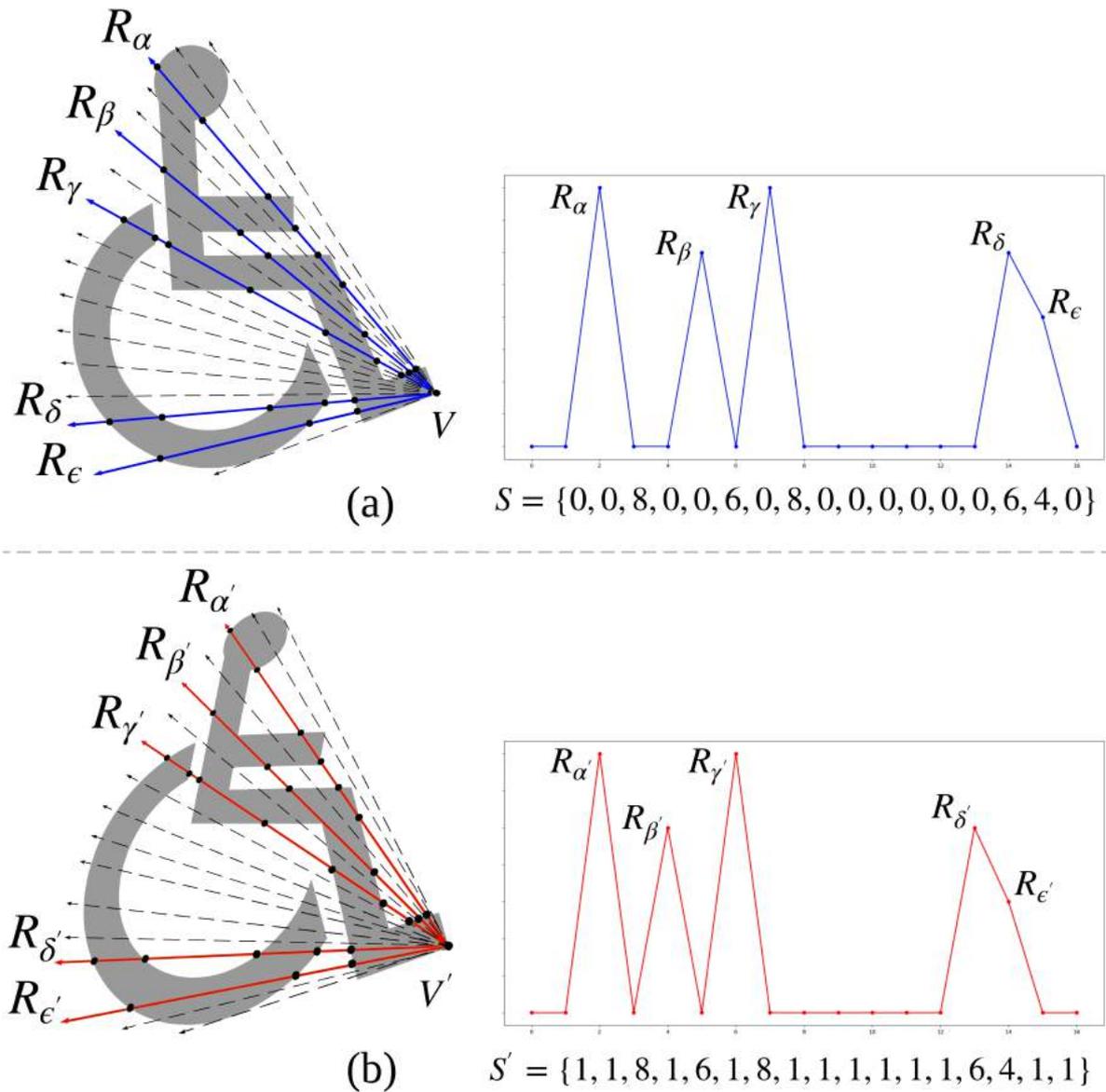


Figura 31 – Feixes de Raios e espectros de topologia. Na esquerda em (a) tem-se um símbolo de acessibilidade com vista paralela frontal e o conjunto de Raios atravessando a forma planar com alguns Raios rotulados; acontece de maneira similar com o mesmo símbolo deformado projetivamente na esquerda em (b); os espectros de topologia de ambos os símbolos na direita em (a) e (b) são formados pelo número de Pontos de Borda dos pares de Raios correspondentes que são mostrados com o rótulo  $R$  com letras gregas (Ex.:  $R_\alpha \leftrightarrow R_{\alpha'}$ ); para as posições relativas aos Raios que não obtiveram correspondências foram usados os valores 0 no espectro em (a), e 1 no espectro em (b), seguindo a convenção adotada.

enquanto que em (b) os Raios  $R_{\alpha'}$  e  $R_{\beta'}$  correspondentes de  $R_\alpha$  e  $R_\beta$  possuem apenas um Raio que não foi pareado (com valor 1) os separando. Devido a essa imprecisão de posicionamento entre os Raios correspondentes, teremos que recorrer a uma técnica de comparação entre os espectros que seja robusta a essas deformações de sinais.

### 7.1.2.1.2 Comparação entre Espectros de Topologia via DTW

Na Figura 31 foram ilustrados os espectros correspondentes aos símbolos de acessibilidade em vistas distintas. O espectro exibido em (b) é uma versão com regiões deslocadas do espectro do símbolo original em (a), impossibilitando uma comparação termo-a-termo entre as sequências. Para a comparação desses tipos de espectros, foi usado o algoritmo *Dynamic Time Warping* (DTW), já discutido no Capítulo 4, com a finalidade de alinhar os espectros como usados em diversos trabalhos que tratam desse tipo de comparação: Em (LI; TAN, 2010) a técnica DTW foi usada na comparação de espectros de razão cruzada (*Cross-Ratio Spectrum*) para reconhecimento de símbolos deformados projetivamente, e em (JIA et al., 2014) com seus espectros de razão características (*Characteristic Ratio Spectrum*) com o mesmo fim, e até nos histogramas do espaço de Radon em (SANTOSH, 2011) para reconhecimento de caracteres; seguindo a mesma abordagem desses descritores citados utilizamos a DTW na comparação entre os nossos espectros de topologia de Raios.

Para exemplificar o uso da técnica no contexto do descritor apresentado neste trabalho, são usados os espectros extraídos na comparação entre feixes esboçados na Figura 31,  $S = \{0, 0, 8, 0, 0, 6, 0, 8, 0, 0, 0, 0, 0, 0, 6, 4, 0\}$  e  $S' = \{1, 1, 8, 1, 6, 1, 8, 1, 1, 1, 1, 1, 1, 6, 4, 1, 1\}$ . Para a comparação desse tipo de espectro foi usada a DTW com a distância de Hamming como função de custo. Estamos interessados em saber quanta semelhança existe entre esses dois espectros, e a distância de Hamming retorna apenas dois possíveis valores quando for usada na comparação termo-a-termo dos espectros: 0, caso os valores dos termos sejam iguais ou 1, se forem distintos; como resultado final da comparação, teremos uma distância DTW que varia entre 0 e  $N$ , onde  $N$  é a amostragem baseada em número de Raios por Vértice do invólucro; a grande vantagem do uso da distância de Hamming é que teremos valores de distâncias conhecidos previamente, e com isso fica mais fácil estimar o grau de semelhança entre as sequências tendo como distância máxima o número de Raios  $N$ . Para as sequências com tamanhos iguais à do exemplo,  $S$  e  $S'$ , poderemos ter um valor conhecido máximo de distância DTW,  $N = 17$ .

Na Tabela 4 é mostrada a matriz de alinhamento entre as duas sequências dos espectros  $S$  e  $S'$  e um caminho destacado obedecendo as regras de condições para o alinhamento citados em 4.3.2. Cada termo  $D_{(i,j)}$  da matriz para  $0 \leq i < N$  e  $0 \leq j < N$  foi computado através da equação:

$$D(i, j) = (s_i \oplus s'_j) + \min \begin{cases} D(i-1, j-1), \\ D(i, j-1), \\ D(i-1, j) \end{cases} \quad (7.1)$$

onde  $s_i \in S$  e  $s'_j \in S'$ . A Equação 7.1 é uma versão da Eq. 4.7 com uma função de custo especificada; o termo  $(s_i \oplus s'_j)$  é a função de custo DTW, que nesse caso é a Distância de Hamming.

	<b>1</b>	<b>1</b>	<b>8</b>	<b>1</b>	<b>6</b>	<b>1</b>	<b>8</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>6</b>	<b>4</b>	<b>1</b>	<b>1</b>
<b>0</b>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
<b>0</b>	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
<b>8</b>	1	2	2	3	3	3	2	3	3	3	3	3	3	3	3	3
<b>0</b>	1	2	3	3	4	4	3	3	4	4	4	4	4	4	4	4
<b>0</b>	1	2	3	4	4	5	4	4	4	5	5	5	5	5	5	5
<b>6</b>	1	2	3	4	4	5	5	5	5	5	6	6	6	5	6	6
<b>0</b>	1	2	3	4	5	5	6	6	6	6	6	7	7	6	6	7
<b>8</b>	1	2	2	3	4	5	5	6	7	7	7	7	8	7	7	7
<b>0</b>	1	2	3	3	4	5	6	6	7	8	8	8	8	8	8	8
<b>0</b>	1	2	3	4	4	5	6	7	7	8	9	9	9	9	9	9
<b>0</b>	1	2	3	4	5	5	6	7	8	8	9	10	10	10	10	10
<b>0</b>	1	2	3	4	5	6	6	7	8	9	9	10	11	11	11	11
<b>0</b>	1	2	3	4	5	6	7	7	8	9	10	10	11	12	12	12
<b>0</b>	1	2	3	4	5	6	7	8	8	9	10	11	11	12	13	13
<b>6</b>	1	2	3	4	4	5	6	7	8	9	10	11	12	11	12	13
<b>4</b>	1	2	3	4	5	5	6	7	8	9	10	11	12	12	11	12
<b>0</b>	1	2	3	4	5	6	6	7	8	9	10	11	12	13	12	13

Tabela 4 – Matriz de Distância acumulada DTW com uso da distância de Hamming como função de custo para as sequências  $S = \{0, 0, 8, 0, 0, 6, 0, 8, 0, 0, 0, 0, 0, 0, 6, 4, 0\}$ ,  $S' = \{1, 1, 8, 1, 6, 1, 8, 1, 1, 1, 1, 1, 1, 6, 4, 1, 1\}$ , e um exemplo de caminho possível para o alinhamento ótimo; a última célula da matriz exibe o valor de distância DTW entre os dois espectros, que no caso é 13.

A Figura 32 mostra as ligações entre os termos dos espectros, como resultado da comparação entre os mesmos, de acordo com o caminho destacado na Matriz 4.

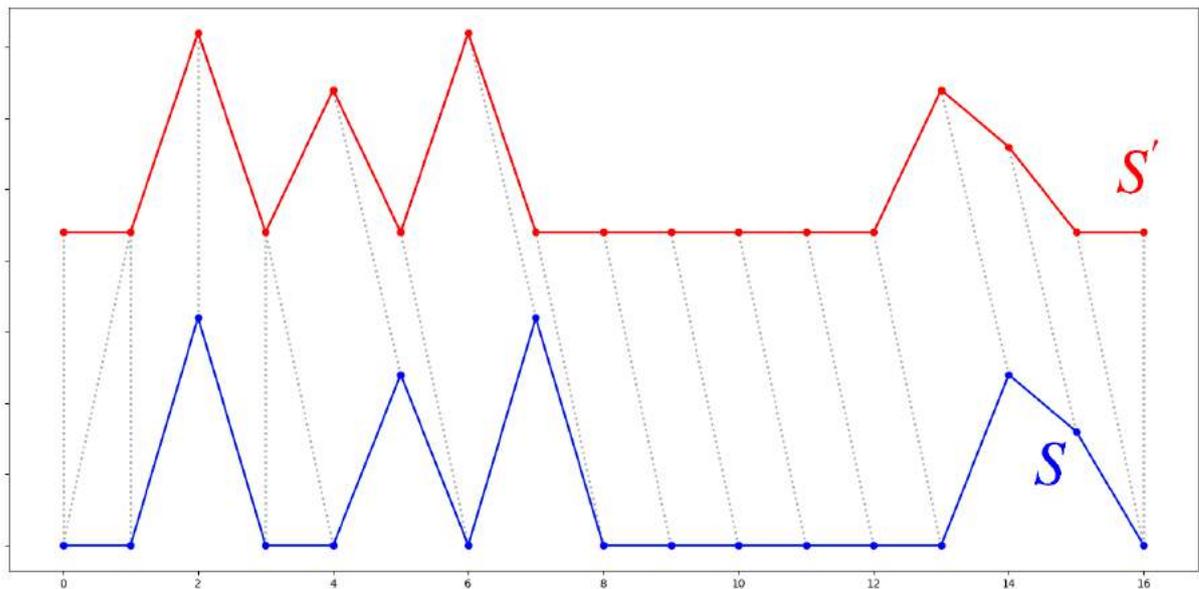


Figura 32 – Alinhamento ótimo resultante da comparação entre os Espectros  $S$  e  $S'$  via DTW.

O objetivo final da comparação desses espectros originados a partir da comparação feixe-a-

feixe é de encontrar as correspondências entre os Vértices baseado na menor distância obtida durante a comparação dos espectros. Cada feixe de Raios da imagem *Template* irá ser comparado com outros feixes da imagem *Query*, resultando num pareamento entre os Vértices dos invólucros convexos dos objetos em comparação.

#### 7.1.2.1.3 Comparação entre espectros de topologia com sinais pós-processados

Foi visto durante a formação dos espectros em 7.1.2.1.1 que os mesmos são gerados pela comparação feixe-a-feixe, e que são sinais contendo números inteiros dos Raios que foram correspondidos durante a comparação Raio-a-Raio, e os Raios que não foram associados com nenhum outro, atribuímos um valor padrão (0 ou 1) no espectro da imagem com o objetivo de preencher esse vazio entre os Raios correspondidos e fornecendo dois sinais com igual número de Raios emitidos por Vértice. Além disso, foi também explicado em 7.1.2.1.2 uma forma de comparação entre os espectros de topologia usando DTW, onde a técnica se mostrou adequada para espectros dessas natureza, para os quais os valores correspondentes não estão obrigatoriamente localizados nas mesmas posições devido às transformações geométricas sofridas.

Durante o curso do desenvolvimento dessa dissertação foi observada ao revisar a técnica originalmente proposta uma maneira alternativa e mais barata computacionalmente para a comparação dos espectros de topologia. Os espectros surgem como resultado da comparação Raio-a-Raio, por associações unívocas (1 : 1); uma característica desses sinais é que possuem muitas lacunas entre valores com correspondências, tais lacunas representam a parte do espectro formada por Raios que não obtiveram correspondências durante a comparação Raio-a-Raio. Sabendo-se que a comparação Raio-a-Raio tem como resultado o mesmo número de Raios correspondentes, é possível subdividir o espectro, gerando: uma parte dita "ativa", contendo os valores relativos aos Raios correspondentes, e uma parte "inativa", contendo os demais valores. Uma forma de modificar esses espectros sem que se altere a ordem dos valores de Raios correspondentes, e ainda assim manter o sinal com o mesmo tamanho é coalescer os valores que compõem a parte "ativa" dos espectros, formando um sequência conexa, disjunta da parte "inativa". Como foi visto durante a formação dos espectros anteriormente, tomando-se como exemplo o mesmo espectro recém-formado  $S = \{0, 0, 8, 0, 0, 6, 0, 8, 0, 0, 0, 0, 0, 0, 6, 4, 0\}$ , teremos depois do pós-processamento um espectro com valores  $Z = \{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 8, 6, 8, 6, 4\}$ . A ordem dos valores relativos aos Raios se mantêm mesmo depois desse biparticionamento, a única diferença sendo que agora os valores "ativos" e "inativos" estão dispostos em duas partições conexas, que viabiliza uma comparação mais rápida entre os espectros, onde é feita uma comparação direta somente entre as partes "ativas" pela DTW somada com o erro da parte "inativa" que é simplesmente uma comparação simples termo-a-termo que é computacionalmente mais simples e rápida (Figura 33).

Após o biparticionamento dos espectros  $S$  e  $S'$ , gerando os espectros  $Z$  e  $Z'$ , é possível compara-los termo-a-termo com uso da distância de Hamming na região inativa, da seguinte

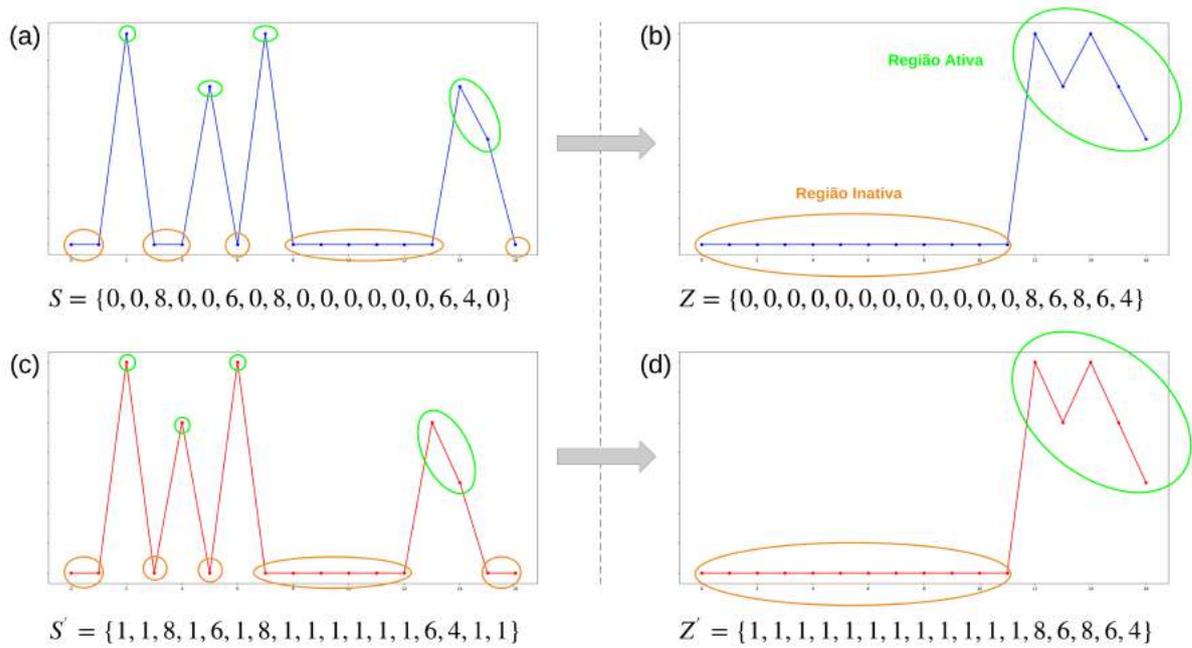


Figura 33 – Exemplos de espectros formados e pós-processados. Em (a) e (c) são mostrados os espectros  $S$  e  $S'$  e em (b) e (d) são mostrados os espectros após o coalescimento dos valores da região ativa na parte direita dos espectros. Em todos os espectros as regiões ativas e inativas estão circuladas na cor verde e laranja, respectivamente.

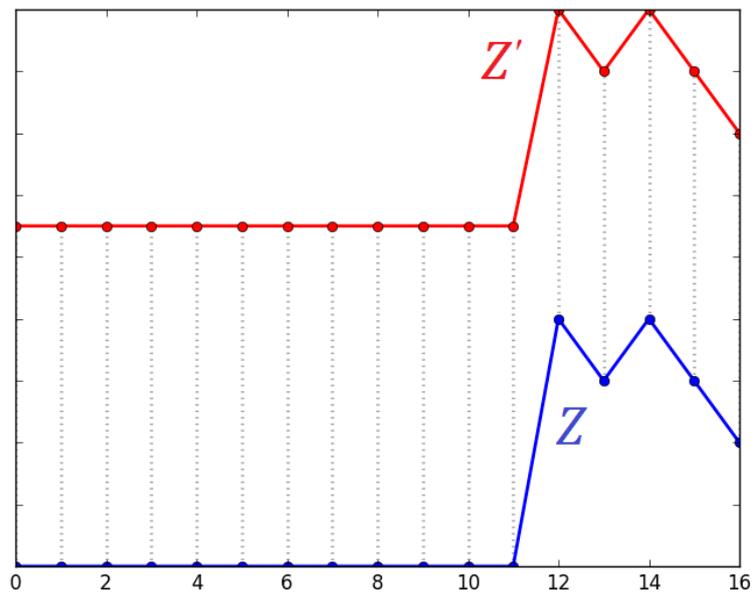


Figura 34 – Espectros biparticionados alinhados.

maneira:

$$d(Z, Z') = \sum_{i=0}^{n-1} (z_i \oplus z'_i), \tag{7.2}$$

onde  $z_i \in Z$  e  $z'_i \in Z'$ .

Como mostrado na Figura 34, com o uso da distância de Hamming o resultado da com-

comparação entre esses espectros  $Z$  e  $Z'$  (derivados de  $S$  e  $S'$ ) é numericamente próximo ao da comparação via DTW dos espectros  $S$  e  $S'$ . Para esse exemplo a região inativa de ambos os espectros possui doze elementos enquanto que a região ativa possui cinco elementos; diante disso, podemos calcular a distância entre os espectros usando a métrica citada, com a parte inativa computada como  $\sum_{i=0}^{11} (0 \oplus 1) = 12$ , e a parte ativa computada pela DTW, que para esse caso equivale a seguinte expressão:  $(8 \oplus 8) + (6 \oplus 6) + (8 \oplus 8) + (6 \oplus 6) + (4 \oplus 4) = 0$ . Somando o erro das duas parcelas, ativa e inativa, temos:  $12 + 0 = 12$ .

De maneira similar à que é feita na comparação via DTW nos espectros completos, o objetivo é encontrar as correspondências entre os Vértices baseado na menor distância obtida durante a comparação dos espectros, tendo como resultado final dessa operação, os Vértices pareados entre os dois objetos em comparação.

## 7.2 ANÁLISE DA DISPOSIÇÃO DOS VÉRTICES DAS ENVOLTÓRIAS

Depois da comparação a nível de Vértices, é possível analisar as correspondências entre os mesmos no que diz respeito à disposição e ordem dos Vértices correspondentes. Nesta etapa estamos interessados em observar como os vértices do modelo e de consulta estão associados. Em nossos experimentos, observamos que os índices de vértices de consulta aumentam linearmente (a menos de uma rotação de deslocamento) com os índices de vértices de modelo de aumento quando objetos similares são comparados. Na Figura 35 são mostradas algumas poucas associações entre os Vértices correspondentes das envoltórias como resultado da comparação entre objetos semelhantes.

É importante ressaltar que este tipo de análise só é possível quando estamos comparando objetos planares que não possuem simetria no seu desenho. Para formas planares simétricas não há uma garantia de que as correspondências entre os Vértices estão numa ordem correta, pois a simetria gera ambiguidades e durante a comparação Vértice-a-Vértice, podendo prejudicar a correta associação entre os Vértices. Mas quando o tipo de imagem não possui simetria na forma, podemos explorar a ordenação dos Vértices correspondidos. A técnica abordada neste trabalho usa uma correlação entre os índices dos Vértices dos objetos em comparação depois de algumas etapas de remoção de *outliers*, falsas correspondências entre vértices. Neste trabalho estamos usando duas, uma baseada em mínimos quadrados entre os índices dos Vértices correspondentes e outra baseada na razão cruzada de cinco pontos coplanares dos Vértices correspondentes.

### 7.2.1 Pré-ajuste dos índices de Vértices e remoção de falsos positivos

É possível estimar uma correlação entre os valores dos índices de Vértices do invólucro convexo dos objetos; foi observado ao longo deste trabalho que uma correlação entre esses índices é uma condição necessária mas não suficiente para o reconhecimento entre formas semelhantes (de mesma classe). Essa correlação é o primeiro crivo para prosseguir com o reconhecimento

ou imediata recusa. Em outras palavras, esta etapa é uma triagem na classificação. Porém, antes de realizar o cálculo da correlação é necessário ajustar os valores dos dados, pois os valores dos índices podem conter associações errôneas entre Vértices (falsos positivos) e/ou podem estar com seus valores de índices circularmente deslocados.

### 7.2.1.1 Ajuste de deslocamento circular dos índices

A Figura 35 ilustra as correspondências entre os Vértices do objeto modelo (*template*), na esquerda, e os do objeto de consulta (*query*), na direita; podemos ver todos os pares de vértices de correspondências:  $(T_0, Q_2)$ ,  $(T_1, Q_3)$ ,  $(T_2, Q_4)$ ,  $(T_3, Q_5)$ ,  $(T_4, Q_6)$ ,  $(T_5, Q_7)$ ,  $(T_6, Q_0)$ ,  $(T_7, Q_1)$ . Neste exemplo é possível notar que os Vértices do *template* estão associados a Vértices do *query* com deslocamento circular de seus índices: o primeiro Vértice da imagem modelo é  $T_0$ , e este vértice está ligado ao Vértice  $Q_2$  da imagem de consulta, e de maneira similar, o primeiro vértice na imagem de consulta  $Q_0$  está ligado ao Vértice  $T_6$  da imagem do modelo.

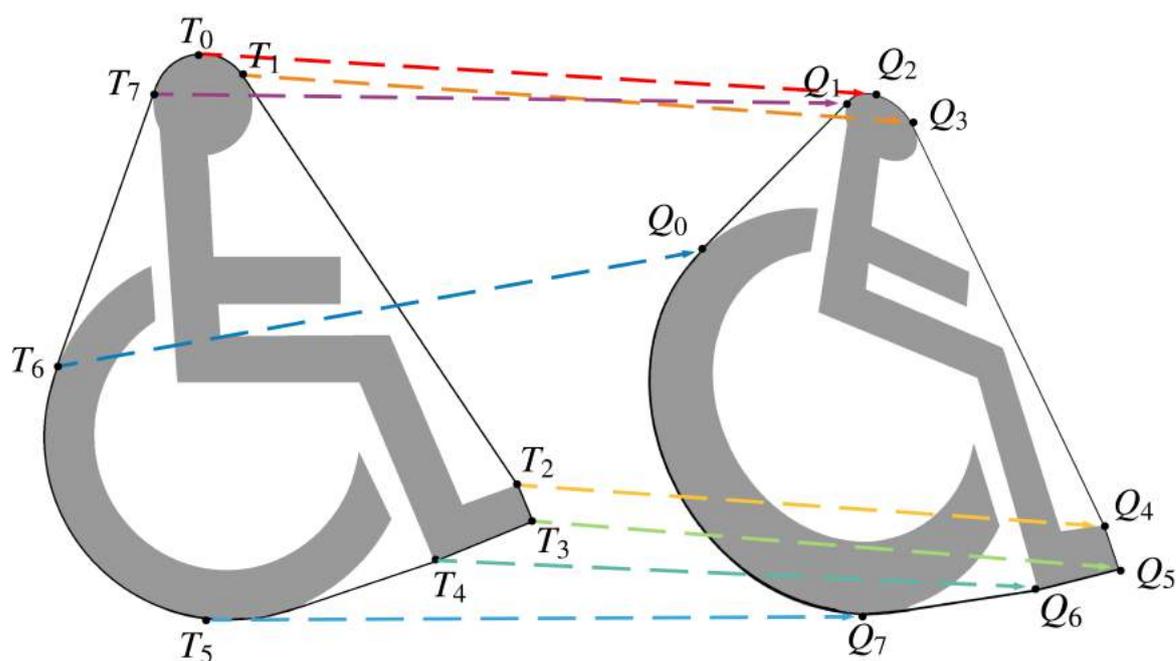


Figura 35 – Um exemplo de correspondência entre Vértices dos invólucros entre objetos de mesma classe: As correspondências Vértice-a-Vértice estão corretas, mas os valores dos índices dos Vértices estão circularmente deslocados: o último Vértice indexado do modelo,  $T_7$ , está associado ao segundo Vértice indexado do objeto de consulta,  $Q_1$ .

A Figura 36 (Esq.) mostra uma matriz de correspondências entre os índices de Vértices dos invólucros baseados no exemplo da Figura 35; nota-se que as relações entre os índices formam uma curva linear e crescente no trecho entre os pares de índices de Vértices  $(T_0, Q_2)$  até os pares  $(T_5, Q_7)$ , e no trecho final entre  $(T_6, Q_0)$  a  $(T_7, Q_1)$ . Essa descontinuidade se deve ao deslocamento circular sofrido durante a indexação dos Vértices obtidos durante a geração do *Convex-Hull* do objeto de consulta em relação aos índices dos Vértices do objeto modelo. Como

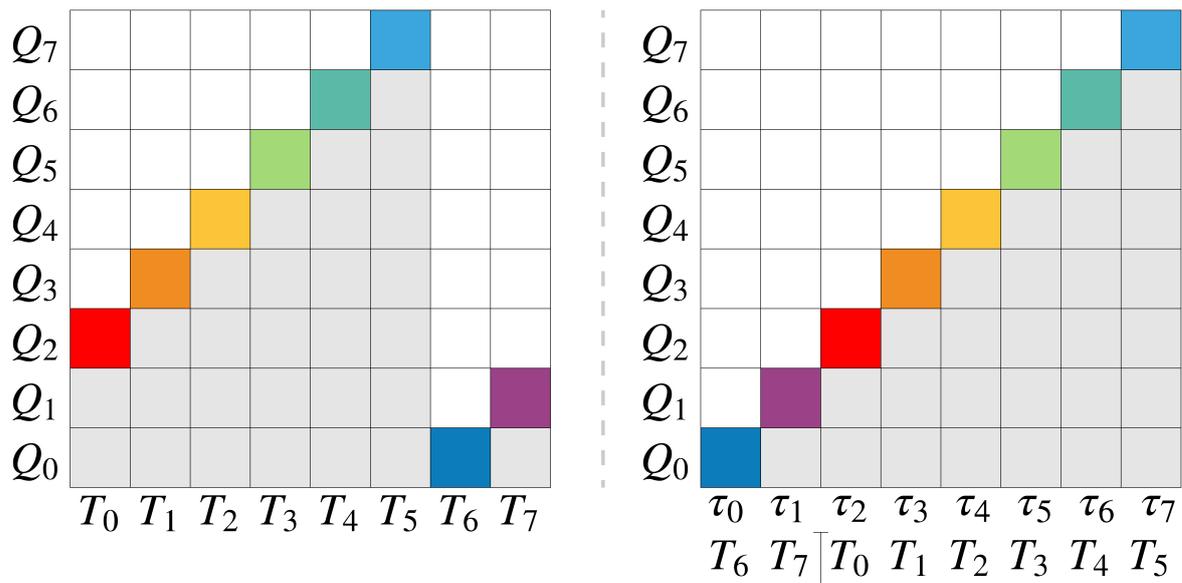


Figura 36 – **Esquerda:** Matriz de Correspondências baseada nas associações de pares de Vértices da Figura 35, as células coloridas representam as associações entre os Vértices *Template*,  $\{T_0, \dots, T_7\}$ , e *Query*,  $\{Q_0, \dots, Q_7\}$ . **Direita:** Re-ajuste na Matriz de Correspondências, as associações continuam as mesmas, porém houve uma mudança nas ordens de apresentação das associações de tal modo que forma uma linha reta crescente; é listada também uma nova indexação para os Vértices *Template* de  $T$  para  $\tau$  com o intuito de gerar valores sempre crescentes para posterior cálculo de correlação entre os índices de Vértices.

já falado anteriormente sobre a relação dos índices, quando a comparação é realizada entre objetos de mesma classe, foi observado que a relação entre os índices é sempre linearmente crescente (ou decrescente).

Então, a fim de estimar corretamente um valor de correlação é necessário aplicar uma permutação nos índices de consulta e criar novos índices para o modelo. A Figura 36 (Dir.) mostra uma permutação entre os pares de índices. A sequência de índices de consulta mudou de  $\{Q_2, \dots, Q_7, Q_0, Q_1\}$  para  $\{Q_0, Q_1, Q_2, \dots, Q_7\}$  e os índices do modelo mudaram de  $\{T_0, \dots, T_7\}$  para  $\{T_6, T_7, T_0, \dots, T_5\}$ ; note que a permutação continuou respeitando as correspondências entre os vértices. O objetivo final foi o de reordenar os índices do *query* para que o índice mais baixo seja o de  $Q_0$ . Inicialmente o termo de menor índice  $Q_0$  está ocupando a penúltima posição da sequência, e passa agora ocupar o primeiro lugar da mesma. Podemos dizer que a rotina de Ajuste de Deslocamento Circular dos Índices começa procurando o índice de menor valor de Vértice correspondido para daí então dividir a sequência em dois trechos: O trecho que começa do índice mais baixo e vai até o final da sequência, e o trecho que vai do primeiro termo até o elemento que antecede o termo de menor valor de índice; a Figura 37 exhibe o procedimento de permutação realizado nos índices de consulta: a seta vermelha está apontada para o termo de menor índice  $Q_0$ .

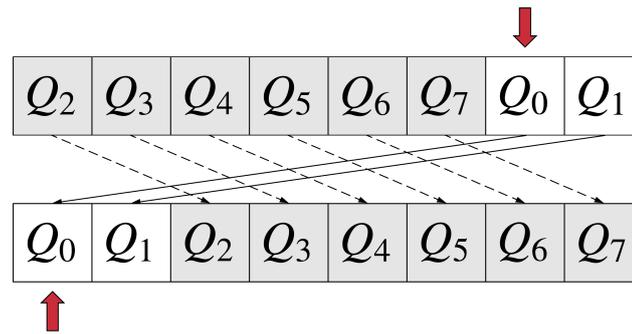


Figura 37 – Ajuste dos índices de consulta: Primeiramente busca-se o elemento de menor índice, que neste caso é  $Q_0$  (que está associado ao vértice  $T_6$  do *template*) e então a sequência é dividida em duas partes, antes e depois de  $Q_0$  e realiza-se o *swap* entre os trechos.

### 7.2.1.2 Remoção de *outliers* e ajuste de linha por Mínimos Quadrados

Além do deslocamento circular, um outro problema que também afeta uma correta medida de correlação entre os índices dos Vértices correspondentes, são os *outliers*; nesse contexto de associações entre Vértices, os *outliers* representam associações falso-positivas de Vértices. Antes de calcular o Coeficiente de Pearson, o índice de pares deve ser também livre de *outliers*, um par de índices falso positivo é um exemplo. Neste trabalho usamos o MMQ para encontrar a melhor reta que satisfaça alguns requisitos. Consideramos como *outliers*: os pares de índices fora do modelo de aproximação linear do conjunto de dados, que no nosso caso são pontos formados por índices de Vértices como coordenadas.

É importante ressaltar que o conjunto de dados deve seguir um modelo linear; para essa análise, dois parâmetros devem ser levados em consideração:

1. **t**: Um valor de *threshold* para determinar se o par de índices se ajusta ao modelo linear, ou seja se o seu erro é menor que **t**.
2. **d**: O número de pares de índices suficientemente próximos (abaixo do *threshold t*) necessários para afirmar que um modelo linear se ajusta bem aos dados.

Caso o modelo não satisfaça o número suficiente de pares de índices com valores de erro menores que  $d$  não há remoção de *outliers*, e a classificação segue para a próxima etapa. Mas caso o número seja satisfatório, poderemos então excluir os pares de índices que extrapolaram o *threshold* de erro  $t$ . Neste trabalho, para os valores de  $t$  e  $d$ , escolhemos:

$$t = 0.5 \cdot \mu_e \quad (7.3)$$

onde  $\mu_e$  é a média dos erros expressa por  $\mu_e = \frac{1}{\eta} \sum_{i=1}^{\eta} \epsilon_i$  e  $\eta$  é o número de pares de Vértices correspondentes, e o termo  $\epsilon_i$  expressa o erro do  $i$ -ésimo elemento do conjunto de dados (pares de Vértices correspondentes).

Para o parâmetro  $d$  que representa o número de dados com erro abaixo de  $t$ , consideramos pelo menos 50% do valor total de pares de Vértices como valor mínimo aceitável para validar o ajuste linear, como exibido abaixo:

$$d \geq 0.5 \cdot \eta \quad (7.4)$$

Onde  $\eta$  é o número de pares de Vértices correspondentes. Respeitando esses critérios expressos anteriormente por esse parâmetros se consegue determinar um modelo linear para o conjunto de dados, caso haja; em se tratando de verdadeiras associações falso-positivas de Vértices entre objetos de mesma classe isso sempre será verdade.

Para um melhor entendimento de como o ajuste linear pode ajudar na remoção de associações de Vértices falsas positivas, abaixo são ilustrados dois objetos planares de mesma classe, que são os mesmos símbolos de acessibilidade exibidos na Figura 35, porém agora tem-se já uma nova indexação para os Vértices do *Template* expressos pelas letras  $\tau$  com índices já ajustados e dois pares de Vértices com Falsas correspondências  $(\tau_3, Q_5)$  e  $(\tau_5, Q_3)$  ligados de forma errada de propósito; o objetivo desta análise é saber como se apresentam os pares de Vértices falsos positivas num gráfico de associação de valores de índices de *template* e *query*. Espera-se, pelas associações verdadeiras positivas de pares de Vértices, que estes sejam representados por pontos que definam um modelo linear, já os pares falsos positivos é esperado que estes sejam *outliers*, ou seja, removendo-os estaremos por tabela removendo as associações falsas positivas entre Vértices.

A Figura 39 mostra um gráfico com associações de índices de Vértices. Através do MMQ se conseguiu a reta crescente em ciano, enquanto os pontos em azul representam os pontos com correspondências verdadeiras e que são *inliers* pelo Método, e os pontos em vermelhos são *outliers* e também representam os pares de Vértices com falsas correspondências  $(\tau_3, Q_5)$  e  $(\tau_5, Q_3)$ . Os *outliers* foram identificados seguindo os critérios de valores de parâmetros já mencionados nas inequações 7.3 e 7.4, sobre os respectivos parâmetros  $t$  e  $d$ . É possível observar que o valor  $\eta$  que representa o número de pares de Vértices associados é igual a  $\eta = 8$  e que para que haja um ajuste linear tem-se que satisfazer pelo menos a metade do número total de associações baseado na inequação 7.4 ( $d \geq 0.5 \cdot \eta$ ), que no caso é 4; tem-se 6 pontos que estão com os valores abaixo do *threshold* expresso pela inequação 7.3 ( $t \leq 0.5 \cdot \mu_e$ ), e dessa forma foi possível para este caso ajustar a maior parte dos dados num modelo linear e com isso remover os *outliers*.

Essa etapa ajuda a remover grande parte dos falsos positivos baseado nos índices de pares de Vértices que não se ajustaram a um modelo linear, porém essa técnica não garante remover todos os falsos positivos, é mostrado mais adiante uma segunda etapa de remoção de falsos positivos.

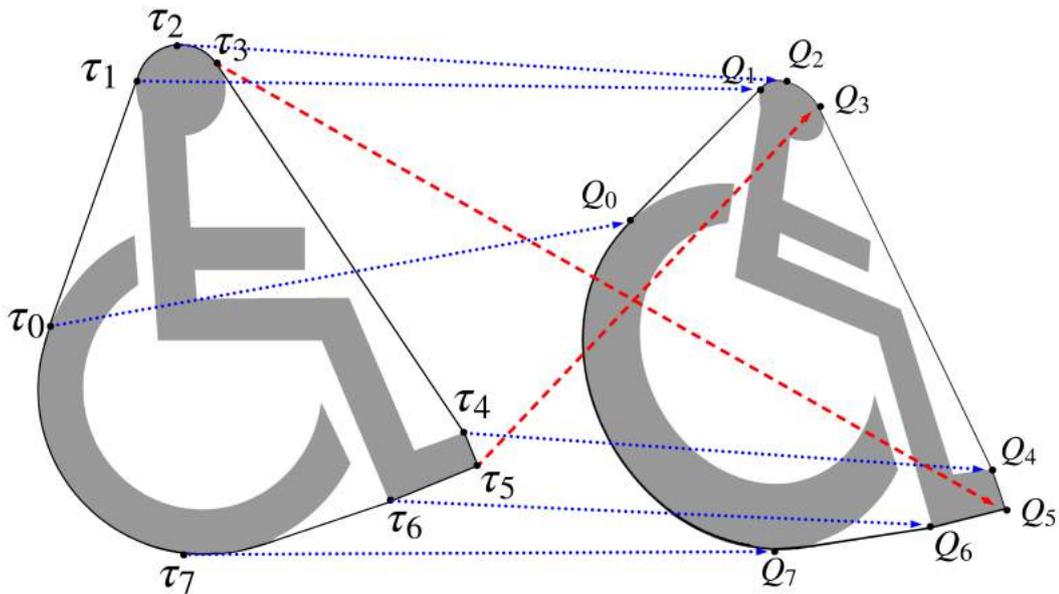


Figura 38 – Exemplo de associações entre Vértices contendo falsas correspondências. Os pares de Vértices associados  $(\tau_3, Q_5)$  e  $(\tau_5, Q_3)$  não representam verdadeiras correspondências e devem ser removidos.

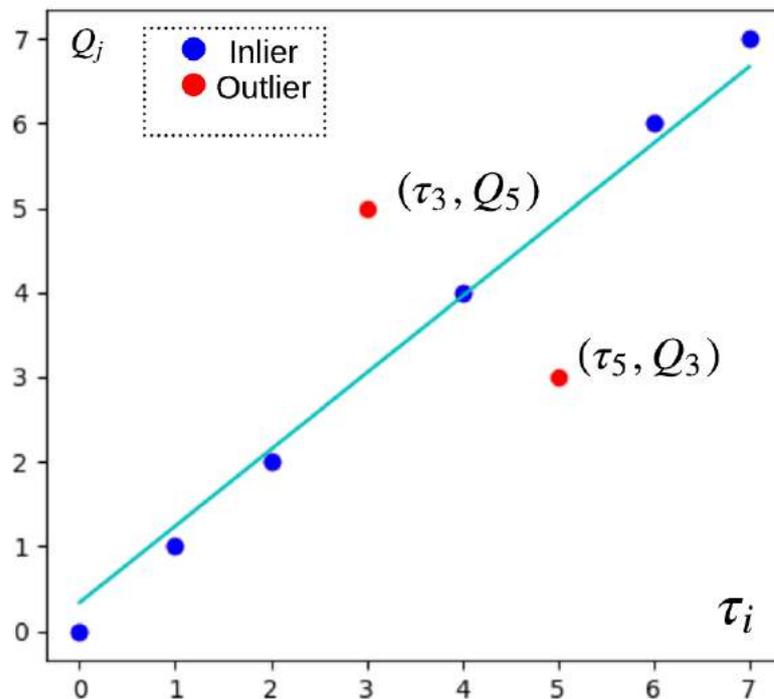


Figura 39 – Ajuste de Linha e Remoção de *Outliers* pelo MMQ. Os pontos são formados pelos índices dos Vértices associados; os pontos em azul são *inliers* e os pontos em vermelho são *outliers*. Os pares de Vértices  $(\tau_3, Q_5)$  e  $(\tau_5, Q_3)$  são associações falsas e podem ser removidos através da remoção de *Outliers* pelo MMQ, esses pontos são considerados *Outliers* seguindo os critérios baseado nos parâmetros de erro de *threshold*  $t$  e quantidade de elementos necessários *inliers*  $d$ .

### 7.2.1.3 Remoção de falsos positivos pela razão cruzada de cinco pontos coplanares

Como resultado parcial da classificação, a essa altura tem-se um conjunto de Vértices no descritor do objeto *template* associados aos pares com o outro conjunto de Vértices no descritor do objeto *query*. Além da análise dos índices dos Vértices realizada com o MMQ com o propósito de remoção dos falsos positivos é possível também remover as falsas correspondências de Vértices a partir de algum invariante geométrico. Foi visto no Capítulo 3 em 3.6.2 que a razão cruzada de cinco pontos coplanares é um invariante projetivo, então para um dado conjunto de cinco Vértices do *template* é esperado que a razão cruzada desse conjunto seja igual à razão cruzada dos seus cinco Vértices correspondentes no *query* se esta for uma versão de mesma classe do objeto *template* que tenha sofrido alguma transformação projetiva; caso contrário, isso indica que pelo menos uma dessas associações é um falso positivo em potencial.

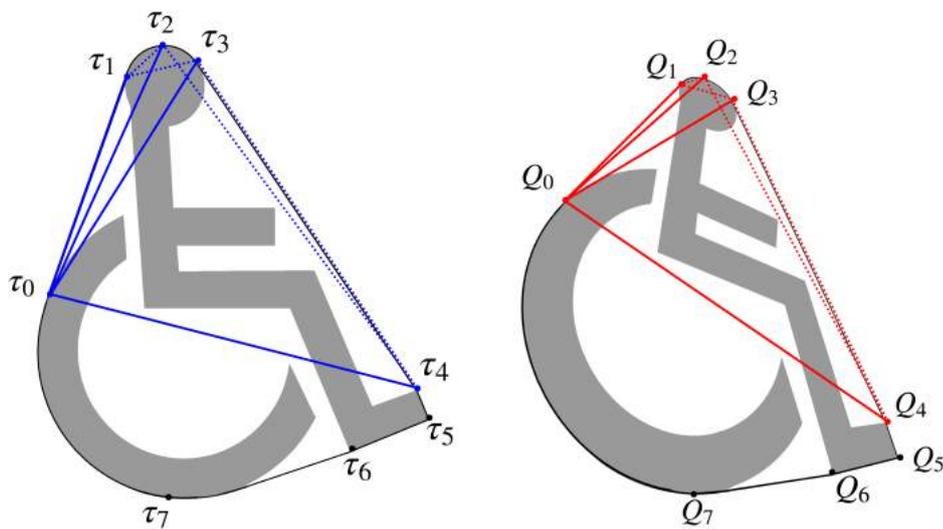


Figura 40 – Um exemplo de combinação de cinco pares de Vértices correspondentes (*template* e *query*) em objetos de mesma classe: No *template* (Esq.), a razão cruzada do conjunto de Vértices  $\{\tau_0, \tau_1, \tau_2, \tau_3, \tau_4\}$ , dispostos nessa ordem, será igual à razão cruzada do outro conjunto de Vértices pertencentes ao *query* (Dir.),  $\{Q_0, Q_1, Q_2, Q_3, Q_4\}$ , dispostos nessa ordem.

Então, sabemos que se as associações Vértice-a-Vértice estiverem corretas, a razão cruzada desses cinco vértices se mantém a mesma num dado conjunto de pares de vértices associados como ilustrado na Figura 40; esse é o princípio básico que rege esta análise baseada nesse invariante projetivo. Devemos procurar inicialmente uma combinação de cinco pares de vértices do *template* que possua valor de razão cruzada de cinco pontos coplanares iguais a seus correspondentes do *query*. A Figura 41 mostra um diagrama de como é feito o cálculo desse invariante com o uso da Equação 3.7 exibida no Capítulo 3 e a comparação entre esses valores baseada num valor de tolerância percentual que decide se os valores de razão cruzada desses dois conjuntos de Vértices estão suficientemente próximos para serem considerados verdadeiramente correspondentes ou não.

**Entrada:** Conjuntos de Vértices associados

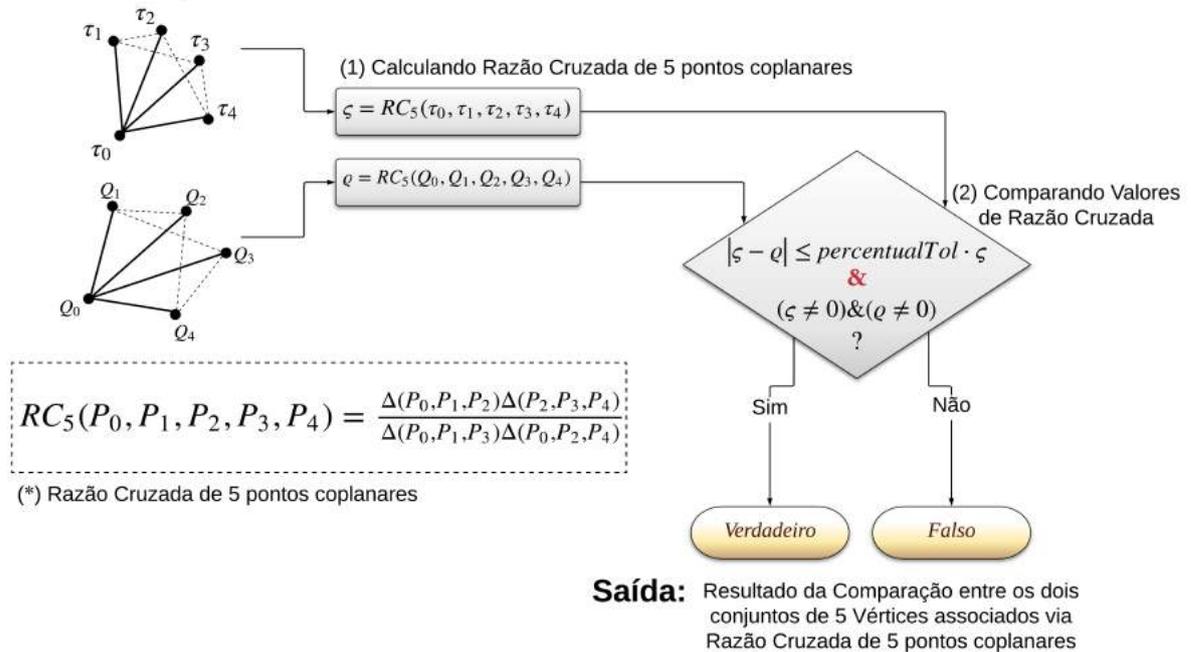


Figura 41 – Comparação dos valores de Razão Cruzada de cinco pontos coplanares. Em (\*) é mostrada a equação da razão cruzada de cinco pontos coplanares. Em (1) é realizado o cálculo da razão cruzada de cada conjunto de Vértices e em (2) os valores calculados  $\zeta$  e  $\varrho$  são comparados seguindo a inequação  $|\zeta - \varrho| \leq \text{percentualTol} \cdot \zeta$ , onde *percentualTol* representa o valor percentual de tolerância (por padrão: 10%) que deverá ser multiplicado com o valor de  $\zeta$ . Se  $|\zeta - \varrho|$  for menor ou igual ao produto de tolerância percentual com o valor da razão cruzada  $\zeta$  e também os valores de razão cruzada  $\zeta$  e  $\varrho$  devem ser diferentes de zero, então considera-se que as associações Vértice-a-Vértice entre esses dois conjuntos estão corretas.

Quando encontrada tal combinação, armazenamos numa lista de pares confiáveis que pode subsequentemente ser usada para se testarem pares de Vértices ainda não analisados, combinando quatro pares de Vértices confiáveis com um par a ser analisado; e aí é possível realizar o cálculo desse invariante entre os Vértices do lado do *template* e do *query*. Caso essa combinação possua valor de razão cruzada de cinco pontos coplanares iguais entre seus pares de Vértices (*template-query*), adicionamo-lo na lista de pontos confiáveis; caso não seja igual, testaremos com outra combinação de quatro pares confiáveis da lista até se esgotarem as chances; caso não haja comparação que seja próxima (baseada num valor de tolerância), então descartaremos esse par, pois se trata de uma associação falsa positiva de Vértices.

A Figura 42 explica o processo de remoção de associações falsas positivas de Vértices baseado nessas verificações dos valores de razão cruzada entre os conjuntos de cinco pares de Vértices associados por vez. Recebe-se como entrada as associações entre Vértices do *template* e *query* e, como já explicado, são testadas as associações Vértice-a-Vértice através da comparação dos valores de razão cruzada.

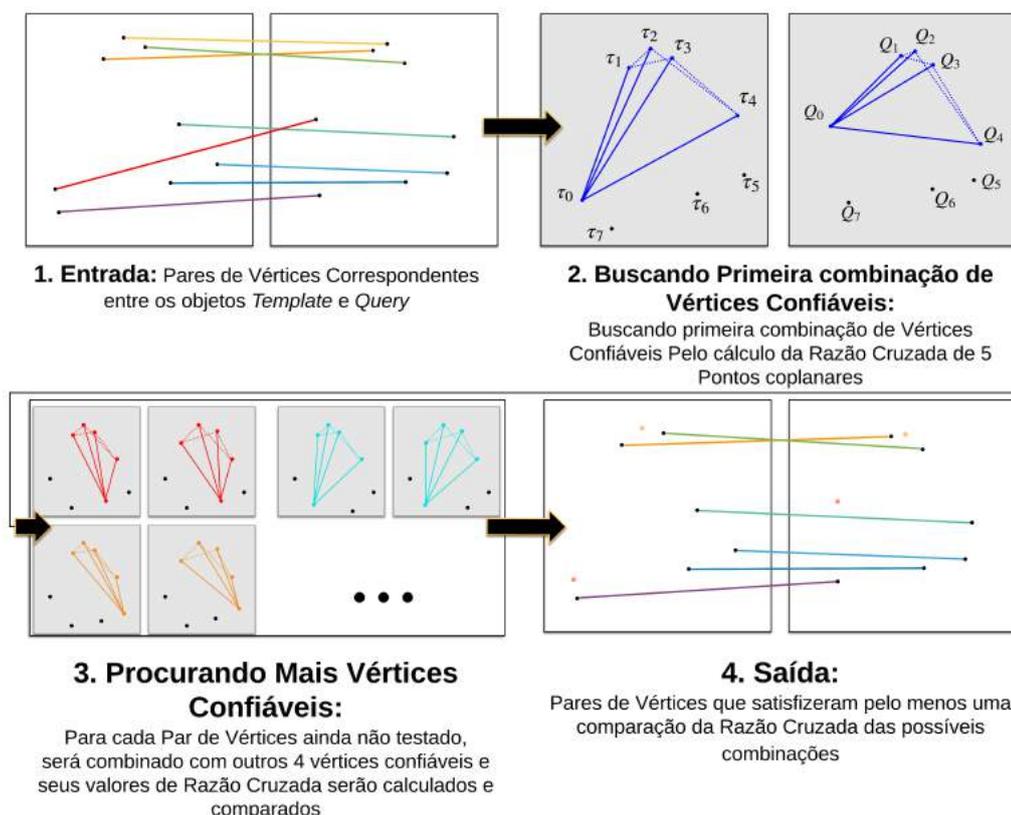


Figura 42 – Etapas do processo de remoção de associações falsas positivas de vértices pela razão cruzada de cinco pontos coplanares.

O resultado final desta etapa continua sendo as correspondências entre os Vértices dos dois descritores em comparação (*template* e *query*), porém agora temos uma filtragem mais criteriosa das associações, livre de falsos positivos, restando apenas pares selecionados segundo os critérios desta análise baseada na razão cruzada de cinco pontos coplanares.

A Figura 43 mostra duas imagens contendo os símbolos de acessibilidade em diferentes vistas e com diferentes dimensões da imagem. No caso, a imagem da esquerda (*template*) é de  $518 \times 518$  pixels e a imagem da direita (*query*) é de  $210 \times 134$  pixels; o número de Vértices do *Convex-Hull* também é diferente devido a diferentes resoluções do objeto em diferentes deformações, a imagem *query* foi extraída de uma cena real; o *template* possui 58 vértices, enquanto o *query* possui 31 vértices. O número de pares de Vértices associados é 18, fruto das comparações Vértice-a-Vértice, porém é possível notar ainda um grande número de falsos positivos, as falsas associações são ainda removidas.

São exibidas na Figura 44 as correspondências que restaram ao processo de seleção das associações entre os Vértices do *template* e *query*; sobraram apenas seis correspondências das 18 associações anteriores ao processo, caindo para 33,33%. Esse baixo número de associações sobreviventes e essa remoção considerável é explicada pela baixa resolução da imagem *query* que foi segmentada de uma cena real, porém a técnica se mostrou robusta mesmo nessas condições.

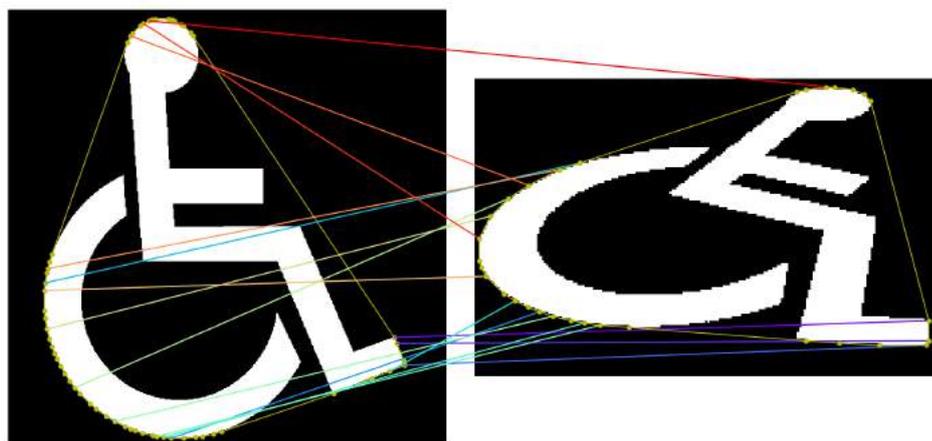


Figura 43 – Associações entre Vértices contendo ainda falsos positivos, o número de associações é 18. A imagem *template* (Esq.) possui dimensão de  $518 \times 518$  e a imagem *query* (Dir.)  $210 \times 134$ . O *template* possui 58 vértices no seu invólucro contra 31 do *query*.

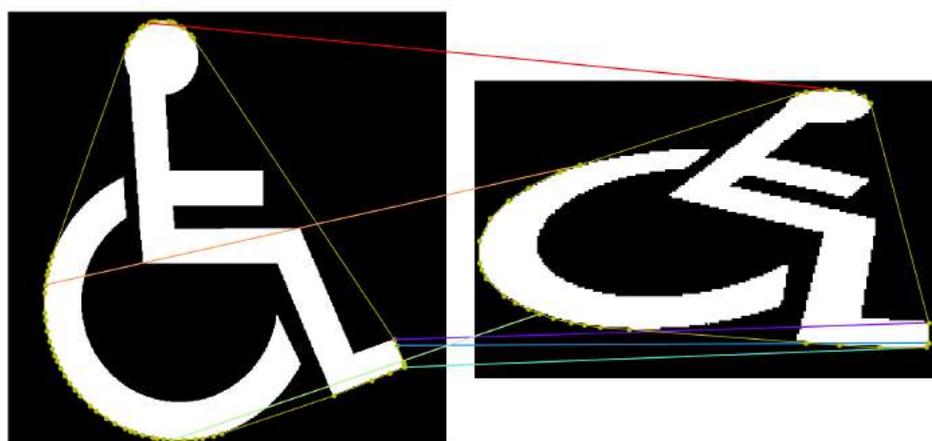


Figura 44 – Associações entre Vértices após a remoção dos falsos positivos, o número de associações entre vértices é 6, restando apenas 33,33% das associações anteriores ao processo de remoção de falsos positivos baseado em razão cruzada de cinco pontos.

Na Figura 45 são exibidas associações entre a imagem *template*, já usada anteriormente, com uma nova imagem *query* de dimensões ( $1728 \times 1728$ ) com uma outra deformação projetiva com 104 vértices em seu invólucro; o interessante dessa comparação é que não foi removida nenhuma associação de pares de vértices pelo processo baseado na razão cruzada; visualmente é possível perceber que as correspondências estão em grande maioria corretas.

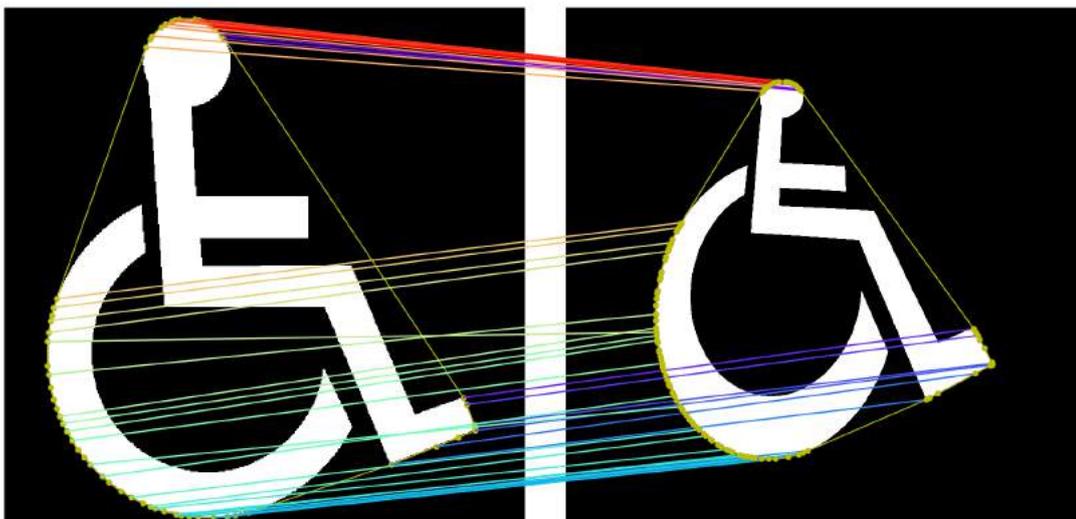


Figura 45 – Associações entre Vértices, possuindo 41 correspondências, sendo o *template* (Esq.) com dimensões de  $518 \times 518$  e o *query* (Dir.) com  $1728 \times 1728$ ; o *template* possui 58 vértices no seu invólucro contra 104 do *query*. Segundo o método de remoção explicado, não houve necessidade de remoção de nenhum par de vértices.

Antes de finalizar, algumas considerações sobre esse método de remoção proposto devem ser destacadas, sobre o número de comparações e sobre os valores de razão cruzada válidos para a comparação. Inicialmente procura-se um conjunto de cinco pares correspondentes de Vértices que possuam valores suficientemente próximos de razão cruzada que consideramos na prática como iguais a menos de uma tolerância, então no pior caso teremos  $C_{n,5}$  verificações iniciais para encontrar um conjunto de pontos ditos confiáveis. Depois podemos usar quatro desses pares de vértices confiáveis para testar os outros pares ainda não verificados e, à medida que encontramos novos pares confiáveis, o mesmo é incluso na lista de pares de vértices confiáveis. Para considerar um par de Vértices basta que apenas uma combinação seja bem sucedida; já para a eliminação de um par de vértices (no caso dos falsos positivos) o mesmo tem que ser combinado com os pares confiáveis, e só é excluído depois de testado com todas as possíveis combinações de pares já rotulados como confiáveis; no caso de não ter mostrado nenhuma igualdade nos valores esse par é descartado. Então podemos estimar que, para a segunda etapa da verificação, no pior caso teremos  $C_{k,4}$ , onde  $k$  é o número de elementos do conjunto atual de pares de vértices confiáveis. A razão cruzada dos cinco pontos coplanares, que pode ser entendida também como a razão cruzada da área de quatro triângulos, é expressa pela equação 3.7 do cálculo desse invariante,  $RC_5(P_1, P_2, P_3, P_4, P_5) = \frac{\Delta(P_1, P_2, P_3)\Delta(P_1, P_4, P_5)}{\Delta(P_1, P_2, P_4)\Delta(P_1, P_3, P_5)}$ . Os termos das áreas dos triângulos  $\Delta(P_i, P_j, P_k)$ , no caso desses pontos serem colineares, possuem valor nulo. Se o termo residir no numerador o valor da razão cruzada é zero, caso esteja no denominador o valor é indefinido, mas durante a comparação desconsideramos os valores de razão cruzada tanto iguais a zero como indefinidos, mesmo que as potenciais correspondentes possuam o mesmo valor, e pode continuar sendo testados com outras combinações de pares de Vértices.

## 7.2.2 Correlação entre índices de Vértices

Após o tratamento de ajuste do deslocamento circular dos índices, remoção das associações falsas positivas de vértices, remoção de *outliers* por MMQ e análise dos valores de razão cruzada de cinco pontos coplanares, foi percebido na análise dos índices dos Vértices correspondentes durante a comparação entre objetos de mesma classe que os mesmos cresciam/decresciam linearmente enquanto o outro crescia; e quando comparado com objetos de diferentes classes nem sempre havia uma correlação ao crescimento dos índices, então uma alta correlação entre os índices dos vértices correspondentes é uma condição necessária embora não suficiente para classificar o objeto como sendo de mesma classe.

É exibida nesta subseção a correlação entre os índices de vértices após o ajuste circular dos índices já mostrados em 7.2.1 e livre de falsos positivos, incluindo exemplos práticos desse ajuste e do cálculo do Coeficiente de Correlação Linear de Pearson a partir desses, no contexto de comparações entre objetos de mesma classe que sofreram diferentes transformações projetivas. O ajuste circular dos Vértices visto em 7.2.1.1 fará com que os Vértices *template*  $T = \{T_0, \dots, T_M\}$  mudem para uma nova indexação  $\tau = \{\tau_0, \dots, \tau_M\}$  de tal forma que os índices dos Vértices do *template* e *query* comecem inicialmente alinhados, e então uma sequência de pares de Vértices associados  $(T_k, Q_j)$  mudará para uma nova indexação  $(\tau_i, Q_j)$  viabilizando agora um cálculo correto da correlação.

### 7.2.2.1 Cálculo do Coeficiente de Correlação Linear de Pearson

Para o cálculo da correlação entre os índices já ajustados é usado o Coeficiente de Correlação Linear de Pearson, introduzido em 4.2.1, e como já explicado, trabalharemos com os pares de índices dos Vértices ajustados:  $\Phi(\tau) = \{0, \dots, M\}$  e  $\Phi(Q) = \{0, \dots, N\}$ , onde a função  $\Phi$  retorna uma lista de índices. Baseado na equação 4.6, o Coeficiente de Pearson das duas sequências de índices dos Vértices  $\tau$  e  $Q$  é calculado como:

$$\rho = \frac{cov(\Phi(\tau), \Phi(Q))}{\sqrt{var(\Phi(\tau)) \cdot var(\Phi(Q))}}. \quad (7.5)$$

Logo em seguida é mostrado o coeficiente de Pearson em diferentes estudos de caso. Trataremos aqui de analisar o comportamento entre as associações de índices de Vértices dos objetos depois da comparação. Iniciaremos analisando o caso de uma imagem do brasão da UFPE sem deformações sendo comparada com outra deformada por uma perspectiva.

#### 7.2.2.1.1 Estudo de Caso I: Comparando objetos de mesma classe com correlação positiva

É analisado neste estudo de caso a correlação de índices durante a comparação entre símbolos de mesma classe após o ajuste circular e a nova indexação; abaixo, na Figura 46, é ilustrado o resultado da comparação Vértice-a-Vértice entre os brasões da UFPE em vista paralela frontal

(Esq.) e após sofrer uma deformação projetiva (Dir.). Na Figura 47, podem-se ver as associações entre os Vértices *template* e *query*, com valores de índices ainda não ajustados, com atenção para as setas indicando o início da numeração dos Vértices em ambos os objetos, que estão em localizações diferentes:

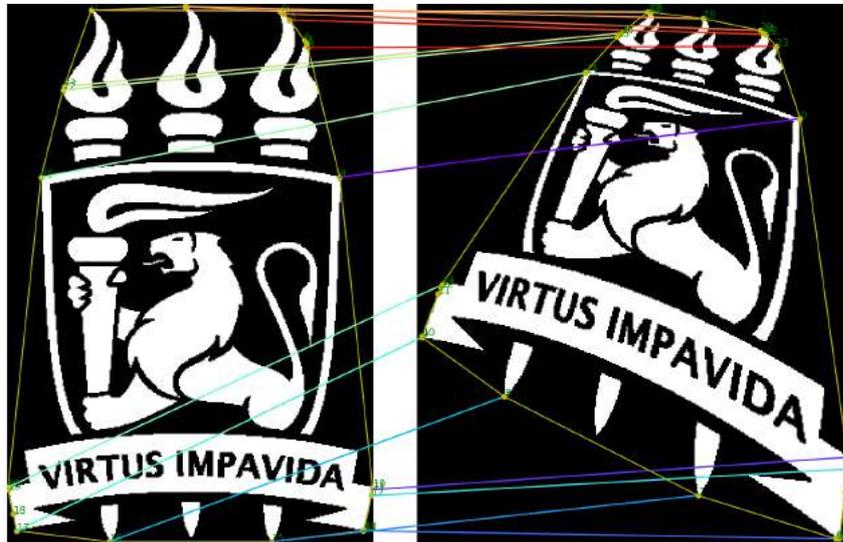


Figura 46 – Resultado da comparação entre os brasões da UFPE em diferentes vistas, possuindo 16 associações de Vértices; o *template* (Esq.) com dimensões de  $300 \times 446$  e o *query* (Dir.) com  $328 \times 410$ ; o *template* e *query* possuem o mesmo número de vértices em seus invólucros, 24 vértices cada.

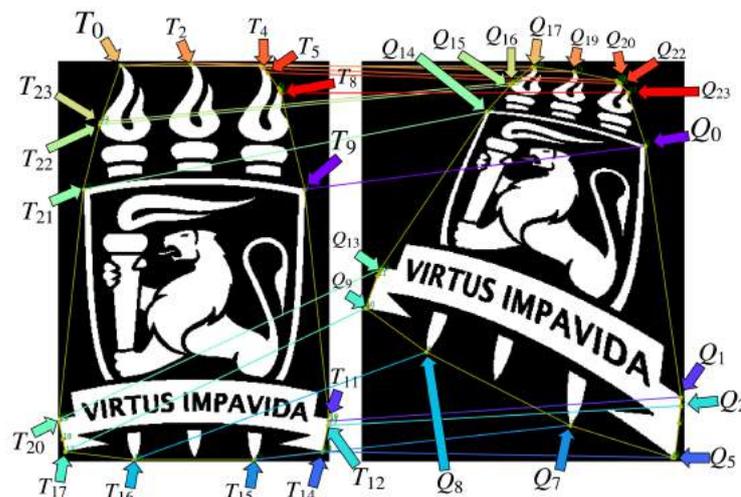


Figura 47 – Mesmo resultado da comparação anterior; as setas em laranja indicam a associação entre o primeiro Vértice enumerado do *template*,  $T_0$  com o Vértice  $Q_{17}$  do *query*; as setas em violeta indicam a associação entre o Vértice  $T_9$  do *template*, com o primeiro Vértice enumerado do *query*,  $Q_0$ . Apesar de ambos os pares estarem deslocados, os pares estão corretos quanto à sua localização nas formas, por isso faz-se necessário um ajuste circular para um correto cálculo da correlação.

Ainda na Figura 47, depois do ajuste circular dos índices, temos para o primeiro índice do objeto *template*,  $T_0$  sendo convertido para  $\tau_{17}$ , e o Vértice  $T_9$ , que é associado ao primeiro

Vértice do *query*, chamaremos agora de  $\tau_0$ ; então teremos as seguintes mudanças:  $(T_9, Q_0) \Rightarrow (\tau_0, Q_0)$  e  $(T_0, Q_{17}) \Rightarrow (\tau_{17}, Q_{17})$ . Em seguida é exibida uma lista de pares de vértices, antes do ajuste dos índices  $(T_k, Q_j)$ , e após o ajuste com o formato  $(\tau_i, Q_j)$ :

$$\begin{aligned} & \{(T_9, Q_0), (T_{11}, Q_1), (T_{14}, Q_5), (T_{15}, Q_7), (T_{16}, Q_8), (T_{12}, Q_2), (T_{17}, Q_9), \\ & (T_{20}, Q_{13}), (T_{21}, Q_{14}), (T_{22}, Q_{15}), (T_{23}, Q_{16}), (T_0, Q_{17}), (T_2, Q_{19}), (T_4, Q_{20}), \\ & (T_5, Q_{22}), (T_8, Q_{23})\} \implies \\ & \{(\tau_0, Q_0), (\tau_2, Q_1), (\tau_4, Q_2), (\tau_5, Q_5), (\tau_8, Q_7), (\tau_9, Q_8), (\tau_{11}, Q_9), \\ & (\tau_{12}, Q_{13}), (\tau_{14}, Q_{14}), (\tau_{15}, Q_{15}), (\tau_{16}, Q_{16}), (\tau_{17}, Q_{17}), (\tau_{20}, Q_{19}), (\tau_{21}, Q_{20}), \\ & (\tau_{22}, Q_{22}), (\tau_{23}, Q_{23})\} \quad (7.6) \end{aligned}$$

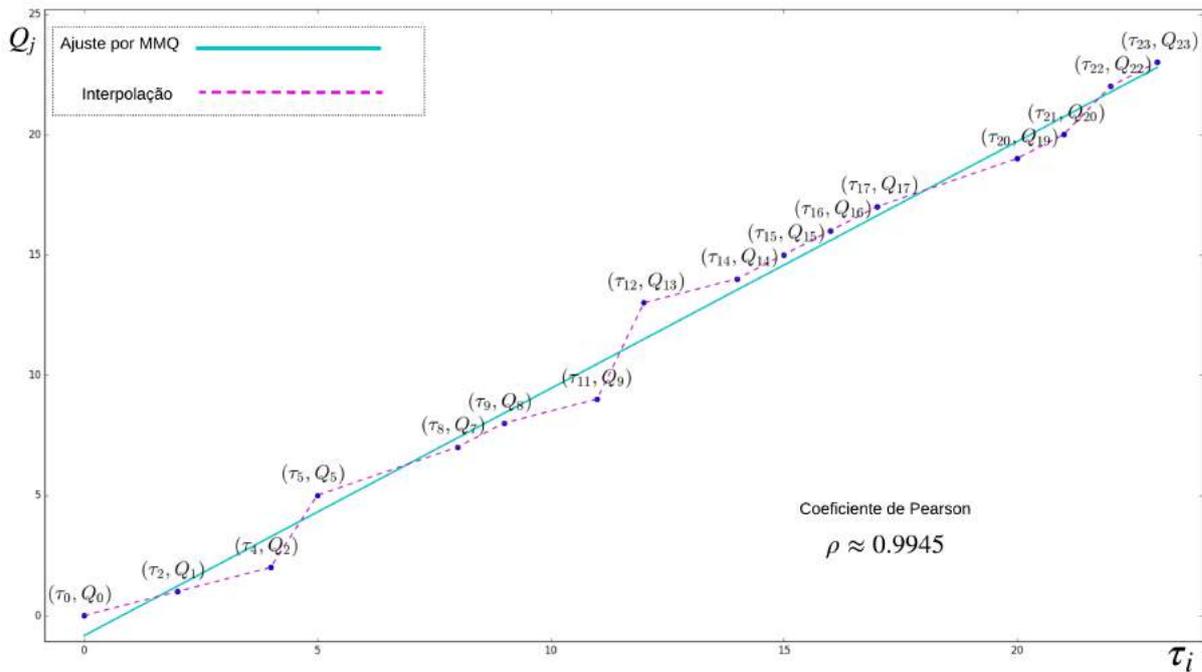


Figura 48 – Gráfico baseado nas 16 associações de vértices como resultado da comparação entre os braços da UFPE em diferentes vistas, com os eixos  $\tau_i$  e  $Q_j$  relativos aos vértices do *template* e *query*. Os pontos em azul representam os pares de vértices associados  $(\tau_i, Q_j)$  com indexação do *template* já ajustado de  $T \Rightarrow \tau$ . A reta em ciano é resultado do ajuste feito pelo MMQ, sem a presença de *outliers* segundo os critérios já discutidos em 7.2.1.2, o resultado da correlação das duas sequências de índices baseada no Coeficiente de Pearson é forte, com valor de  $\rho \approx 0.9945$ .

Como é possível ver na Figura 48, as associações de pares de Vértices com índices já ajustados possuem um modelo aproximadamente linear e crescente, e uma forma de estimar esse padrão é através do Coeficiente de Pearson entre os índices dos Vértices associados. Para os Vértices do *template*, já ajustados, tem-se  $\Phi(\tau) : \{0, 2, 4, 5, 8, 9, 11, 12, 14, 15, 16, 17, 20, 21, 22, 23\}$ , e para os índices de Vértices do *query*,

$\Phi(Q) : \{0, 1, 2, 5, 7, 8, 9, 13, 14, 15, 16, 17, 19, 20, 22, 23\}$ ; calculando o Coeficiente de Pearson baseado na Equação 7.5 entre as duas sequências de índices, obtêm-se:  $\rho = 0.9944613128359946 \approx 0.9945$ . Esse valor indica uma correlação forte e positiva, ou seja, que esboça um modelo aproximadamente linear e crescente. Como já mencionado, durante a comparação de objetos de mesma classe, para os seus pares de Vértices associados, espera-se que os valores dos índices do *query* **creçam** (ou decresçam) linearmente, enquanto os valores dos índices do *template* (já ajustados,  $\tau_i$ ) cresçam. Para a grande maioria dos casos, na comparação entre objetos com mesmas formas planares, mas com transformações geométricas diferentes (incluindo identidade), a correlação tenderá a ser forte e sempre positiva. Os únicos casos em que foram observadas correlações fortes e decrescentes entre objetos de mesma classe, se deram quando ocorreram transformações contendo espelhamento, que é exibido no próximo estudo de caso.

#### 7.2.2.1.2 Estudo de Caso II: Comparando objetos de mesma classe com correlação negativa

De maneira análoga ao estudo de caso anterior, iremos analisar e ajustar os índices de Vértices associados entre objetos e é estimada a correlação de índices entre esses. Abaixo, na Figura 49 é ilustrado o resultado da comparação Vértice-a-Vértice entre as logomarcas da Peugeot em vista paralela frontal (Esq.) e após sofrer uma deformação projetiva **combinada** com uma transformação de **espelhamento** (Dir.); na Figura 50 podem-se ver as associações entre os Vértices *template* e *query*, com valores de índices ainda não ajustados, com atenção para as setas indicando o início da numeração dos Vértices em ambos os objetos, que estão em localizações diferentes:

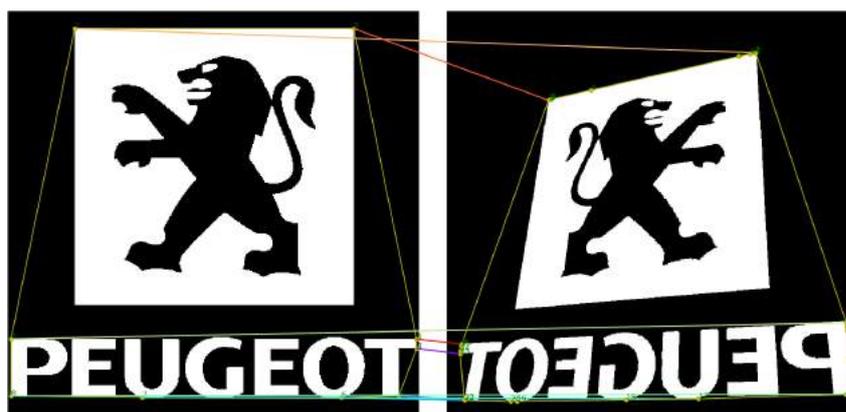


Figura 49 – Resultado da comparação entre as logomarcas da Peugeot em vista paralela frontal, na esquerda, e possuindo deformação projetiva com espelhamento, na direita, com 9 pares de Vértices correspondentes; o *template* (Esq.) possui dimensões de  $810 \times 810$  e o *query* (Dir.) com  $800 \times 800$ ; o *template* possui nove vértices, enquanto o *query* possui 26 vértices.

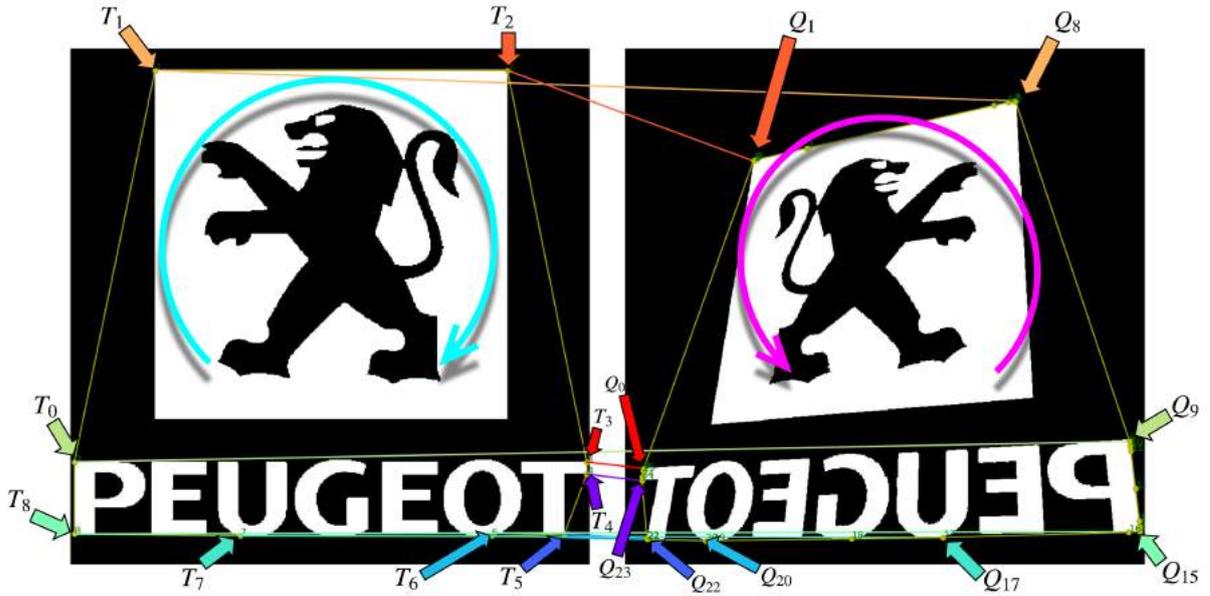


Figura 50 – Mesmo resultado da comparação anterior; as setas apontam para os pares de vértices correspondentes com cores diferentes para cada par; as setas circulares em ambos os objetos indicam o sentido das associações entre os pares de Vértices, partindo do primeiro Vértice do *template*,  $T_0$ , e seu par,  $Q_9$ , do *query*.

$$\{(T_4, Q_{23}), (T_5, Q_{22}), (T_6, Q_{20}), (T_7, Q_{17}), (T_8, Q_{15}), (T_0, Q_9), (T_1, Q_8), (T_2, Q_1), (T_3, Q_0)\} \implies \{(\tau_0, Q_{23}), (\tau_1, Q_{22}), (\tau_2, Q_{20}), (\tau_3, Q_{17}), (\tau_4, Q_{15}), (\tau_5, Q_9), (\tau_6, Q_8), (\tau_7, Q_1), (\tau_8, Q_0)\} \quad (7.7)$$

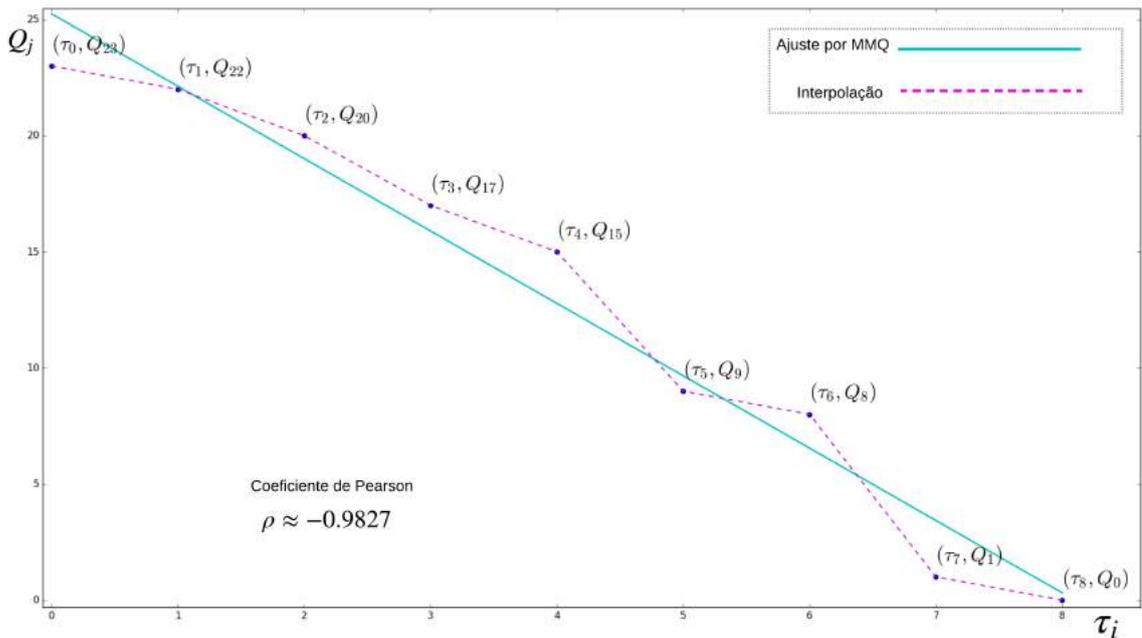


Figura 51 – Gráfico baseado nas nove associações dos Vértices; é apresentada uma correlação forte e negativa, com  $\rho \approx -0.9827$  entre os pares já ajustados no formato  $(\tau_i, Q_j)$ .

Ainda na Figura 50 observa-se uma seta circular no sentido horário no objeto *template*, e uma seta circular no sentido anti-horário no objeto *query*, essas setas se baseiam nos pares de Vértices associados:  $\{(T_0, Q_9), (T_1, Q_8), (T_2, Q_1), (T_3, Q_0)\}$ . Apesar dessas associações ainda não estarem ajustadas, analisando esse trecho de quatro pares de Vértices, já é o suficiente para perceber que à medida que os índices dos Vértices do *template*  $\{T_0, T_1, T_2, T_3\}$  crescem seguindo a seta circular no sentido horário, os índices dos Vértices do *query*  $\{Q_9, Q_8, Q_1, Q_0\}$  decrescem seguindo a seta circular no sentido anti-horário.

Na Expressão 7.7 foi realizado o ajuste circular dos pares de índices de vértices, após o ajuste podemos observar que a medida que os índices dos vértices  $\tau_i$  crescem, os índices  $Q_j$  decrescem. Na Figura 51 é mostrado um gráfico com as associações de pares de vértices com índices já ajustados e livre de *outliers* segundo o MMQ; essas associações de índices possuem um modelo aproximadamente linear e decrescente, e para estimar a correlação foi usado o Coeficiente de Pearson entre esses. Para os vértices do *template* já ajustados, tem-se  $\Phi(\tau) : \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$ , e para os índices de vértices do *query*,  $\Phi(Q) : \{23, 22, 20, 17, 15, 9, 8, 1, 0\}$ ; calculando o Coeficiente de Pearson baseado na Equação 7.5 entre as duas sequências de índices, obtêm-se:  $\rho = -0.9826692256650857 \approx -0.9827$ , esse valor indica uma correlação forte e negativa, ou seja, que esboça um modelo aproximadamente linear e decrescente. Como já dito inicialmente, durante a comparação de objetos de mesma classe, para os seus pares de Vértices associados, espera-se que os valores dos índices do *query* cresçam (**ou decresçam**) linearmente enquanto os valores dos índices do *template* (já ajustados,  $\tau_i$ ) cresçam.

### 7.2.2.1.3 Estudo de Caso III: Comparando objetos de diferentes classes e análise da correlação

São vistas neste caso comparações entre objetos de classes diferentes onde é feita uma análise da correlação entre os índices de Vértices da mesma forma como foi realizada nos estudos de casos anteriores, e espera-se que se tenha uma correlação fraca com associações difusas entre os índices de Vértices mesmo após as etapas de ajuste circular dos índices, e remoção dos pares de Vértices falsos positivos por MMQ e pela razão cruzada de cinco pontos coplanares.

Abaixo, na Figura 52 é exibido o resultado da comparação entre diferentes logomarcas: Adidas e Puma, onde ambas se apresentam em vista paralela frontal sem terem sofrido com deformações projetivas; percebe-se que as associações entre os Vértices não obedecem uma ordem circular como nos casos anteriores; em 7.8 é mostrada a associação do conjunto de pares de Vértices antes e depois do ajuste circular. Nota-se que à medida que os índices do *template* ( $\tau_i$ ) crescem, os índices do *query* ( $Q_j$ ) não seguem uma razão de crescimento ou decrescimento. A Figura 53 mostra o gráfico das associações entre os Vértices de maneira análoga aos casos anteriores. A curva interpolada segue um padrão distante de uma aproximação linear, com trechos de subida e descida, e a correlação entre os índices de vértices associados é desprezível com  $\rho \approx -0.2398$ .

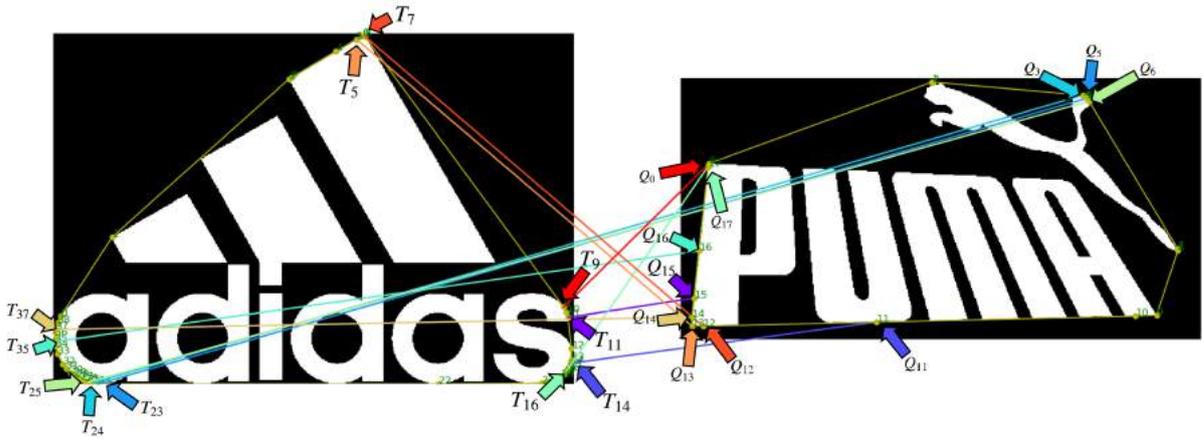


Figura 52 – Resultado da comparação entre as logomarcas da Adidas e Puma, com onze pares de vértices correspondentes; o *template* (Esq.) possui dimensões de  $406 \times 600$  e o *query* (Dir.) com  $260 \times 516$ ; o *template* possui 39 vértices, enquanto o *query* possui 18 vértices.

$$\begin{aligned} & \left\{ (T_{11}, Q_{15}), (T_{14}, Q_{11}), (T_{23}, Q_5), (T_{24}, Q_3), (T_{35}, Q_{16}), (T_{16}, Q_{17}), (T_{25}, Q_6), \right. \\ & \quad \left. (T_{37}, Q_{14}), (T_5, Q_{13}), (T_7, Q_{12}), (T_9, Q_0) \right\} \implies \\ & \left\{ (\tau_5, Q_{15}), (\tau_7, Q_{11}), (\tau_9, Q_{17}), (\tau_{11}, Q_5), (\tau_{14}, Q_3), (\tau_{16}, Q_6), \right. \\ & \quad \left. (\tau_{23}, Q_{16}), (\tau_{24}, Q_{14}), (\tau_{25}, Q_{13}), (\tau_{35}, Q_{12}), (\tau_{37}, Q_0) \right\} \quad (7.8) \end{aligned}$$

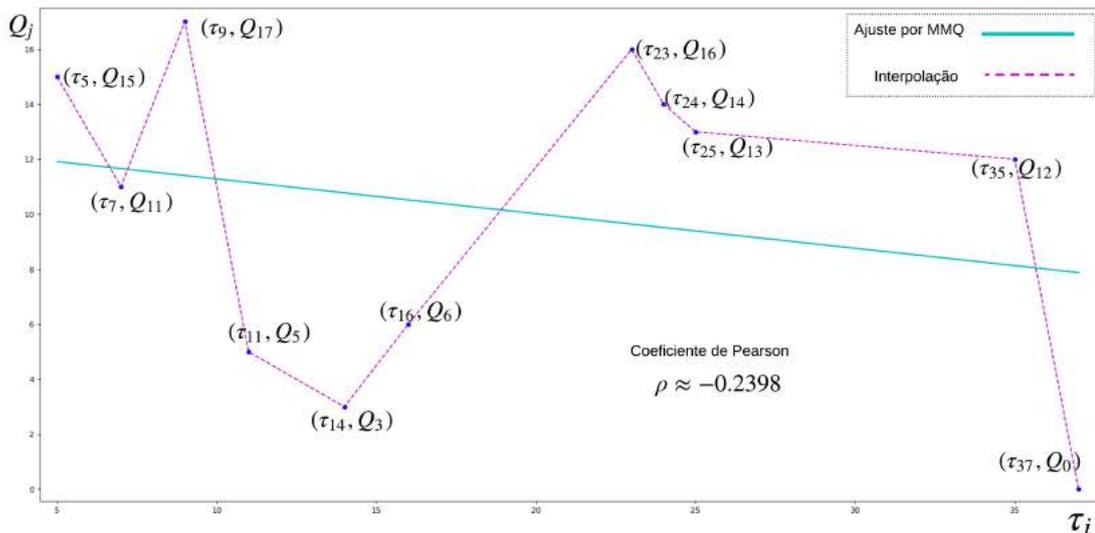


Figura 53 – Gráfico baseado nas 11 associações de vértices do caso abordado; é apresentado uma correlação desprezível e positiva, com  $\rho = -0.23980184951172517$  entre os índices dos pares de vértices já ajustados no formato  $(\tau_i, Q_j)$ , onde o Coef. de Pearson foi calculado a partir das sequências de índices  $\Phi(\tau) : \{5, 7, 9, 11, 14, 16, 23, 24, 25, 35, 37\}$  e  $\Phi(Q) : \{15, 11, 17, 5, 3, 6, 16, 14, 13, 12, 0\}$ .

Este último estudo de caso mostrou que, para as comparações entre objetos de classes diferentes, espera-se na grande maioria dos casos que a correlação entre os índices de vértices associados não seja forte, ao contrário dos casos I e II, em que o Coeficiente de Pearson foi muito forte ( $|\rho| > 0.9$ ) em ambos os casos. Então, é possível usar essa informação do Coeficiente de Pearson como um critério para eliminação durante a classificação do objeto de consulta (*query*). As comparações entre objetos de mesma classe possuem correlações fortes entre seus índices de Vértices associados enquanto na comparação entre objetos de diferentes classes, a correlação dos índices de Vértices dificilmente será forte ou muito forte.

Portanto, podemos afirmar que o cálculo do Coeficiente de Pearson entre as sequências de índices de Vértices associados e ajustados pode ser usado como uma etapa eliminatória, e é possível usar um *threshold* para eliminar os casos em que foram apresentadas correlações fracas ou desprezíveis.

#### 7.2.2.2 Uso da Correlação como critério para eliminação

Como recentemente discutido no final do Estudo de caso III, a correlação entre os índices de Vértices podem ajudar a eliminar candidatos durante a comparação entre os objetos. Durante a comparação entre objetos de mesma classe, espera-se sempre que o valor do Coeficiente de Pearson obtido entre os índices de vértices associados e ajustados apontem para uma correlação forte; já durante a comparação de objetos de classes diferentes não se espera uma forte correlação, não que seja impossível uma forte correlação também para esses casos, mas caso se obtenha um valor de Coeficiente de Pearson próximo de zero, podemos já excluir esse candidato (*query*) e afirmar que não se trata de um objeto da mesma classe do atual *template* comparado.

Os valores do Coeficiente de Pearson,  $\rho$ , variam de -1.0 à 1.0, quanto mais distante de 0, mais forte é a correlação; não interessa se a correlação é positiva ou negativa, como foi visto nos Estudos de caso I e II, ambos apresentaram fortes correlações, onde a primeira era positiva e a segunda era negativa; então é pertinente para a análise da correlação fazer o uso dos valores absolutos do coeficiente de Pearson, então os valores para  $|\rho|$  variam de 0 a 1.0. Abaixo, na Fig. 54 são mostrados dois intervalos de valores possíveis para  $\rho$  em (a), e para  $|\rho|$  em (b), além das setas de indicação que apontam para onde a correlação é mais forte.

Como o interesse da análise da correlação através do Coef. de Pearson se resume a identificar se uma correlação é fraca ou forte, então o uso dos valores absolutos é mais direto e útil para a essa análise. A análise é simplesmente excluir do processo de classificação os candidatos que apresentarem um valor de correlação fraco ou desprezível entre os índices de vértices já ajustados entre os objetos durante a comparação, para isso ser possível deve-se estabelecer previamente um valor de *threshold* para os valores em absoluto do Coef. de Pearson obtidos.

Em (MUKAKA, 2012) são discutidos os valores para uma interpretação qualitativa dos valores de correlação baseados no Coeficiente de Pearson, na Tabela 5 são listadas diferentes faixas de valores do coeficiente em números absolutos; os intervalos de valores da tabela para  $|\rho|$  podem ajudar a uma escolha mais adequada para o valor de *threshold* que determinará se a

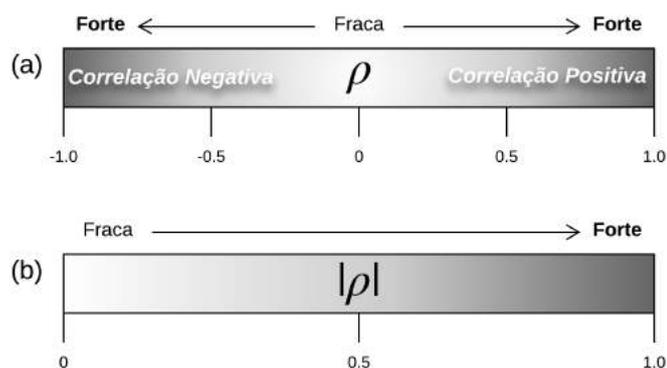


Figura 54 – Intervalos de valores possíveis para  $\rho$  e  $|\rho|$ ; em (a) é exibido uma barra com os valores que o coef. de Pearson pode assumir:  $-1.0 \leq \rho \leq 1.0$ , para a direita indica uma correlação positiva, e para a esquerda uma correlação negativa; em (b) é mostrado o intervalo do módulo de coef. de Pearson:  $0 \leq |\rho| \leq 1.0$ ; as setas apontam para onde a correlação é mais forte.

Intervalos de valores para $ \rho $	Interpretação
$0.9 \leq  \rho  \leq 1.0$	<i>Muito Forte</i>
$0.7 \leq  \rho  < 0.9$	<i>Forte</i>
$0.5 \leq  \rho  < 0.7$	<i>Moderada</i>
$0.3 \leq  \rho  < 0.5$	<i>Fraca</i>
$0.0 \leq  \rho  < 0.3$	<i>Desprezível</i>

Tabela 5 – Tabela com diferentes intervalos de valores para  $|\rho|$  e interpretação desses valores para a correlação.

comparação entre o *query* e *template* deve continuar ou não; de maneira óbvia se espera valores de *threshold* para  $|\rho|$  que incluam a faixa do intervalo de correlação *Moderada* ou acima dessa.

Após a escolha (prévia) de um valor adequado de *threshold* para a correlação, após todas as etapas anteriores de análise dos Vértices, há uma verificação condicional simples baseado nesse valor de *threshold* escolhido; caso o valor da correlação (em absoluto) for inferior a esse valor de *threshold*, finaliza-se imediatamente a comparação entre os objetos planares em análise e conclui-se que se trata de objetos de classes diferentes; caso o valor obtido seja maior ou igual ao *threshold* estipulado, a comparação segue para a próxima etapa, continuando a análise dos Vértices.

A próxima etapa estabelece uma distância de similaridade entre os dois objetos em comparação, baseado em dados já conferidos da análise comparativa entre os descritores das formas planares, sendo assim o último estágio da comparação entre dois símbolos planares.

### 7.3 MÉTRICA DE DISTÂNCIA GLOBAL COMPOSTA

Várias etapas de comparação elemento-a-elemento do descritor foram discutidas em 7.1, tais como: Raio-a-Raio, e Vértice-a-Vértice, onde os Raios são correspondidos através da comparação dos vetores de razão cruzada por distância Euclidiana, enquanto os Vértices são comparados através da distância de Hamming entre os espectros de topologia gerados. De maneira análoga faz-se necessária uma medida global final para identificação de um objeto com outro.

É mostrado mais adiante que existem diferenças entre as etapas do processo de comparação e classificação entre objetos com *template* simétrico e assimétrico, e isso irá impactar na composição das métricas para esses dois casos. Para ambos os casos podemos usar os valores de distância da comparação Vértice-a-Vértice e calcular um valor médio dos mesmos, e usar os percentuais de acertos entre os vértices do *template* e do *query*.

Em se tratando exclusivamente dos casos assimétricos é possível realizar uma composição de métricas com a correlação dos índices dos vértices, percentual de Vértices que foram corretamente associados do *template* e do lado do *query*, juntamente com a média das distâncias Vértice-a-Vértice, gerando uma métrica composta final de distância global. Isso tudo considerando-se as análises feitas em 7.2 para a eliminação de associações de pares de Vértices falsos positivos com remoção de *outliers* pelo MMQ explicado em 7.2.1.2, e através da comparação dos valores de razão cruzada de cinco pontos coplanares descrito em 7.2.1.3. Daí, finalmente extrai-se um valor de distância médio a partir dos pares de Vértices "sobreviventes".

É explicado nesta seção como se obter o valor de distância média global entre os objetos, e um valor de distância global composta para os casos com *template* simétrico e assimétrico. A métrica composta varia no número de elementos integrantes, sendo o simétrico com um elemento a mais, que é a correlação entre os índices de vértices, expressa pelo o módulo do Coeficiente de Pearson,  $|\rho|$ .

#### 7.3.1 Média das distâncias entre Vértices

Como dito, após todas as etapas de análise dos pares de vértices associados entre os objetos em comparação é possível estimar um valor de distância final entre esses. Para isso, inicialmente calcula-se o valor médio das distâncias da comparação Vértice-a-Vértices através dos espectros de topologia gerados em 7.1.2.1.3; é usado um valor médio pelo fato do número de vértices correspondentes poder ser variável, dependendo do objeto e do resultado da comparação Vértice-a-Vértice dos pares de Vértices após a remoção dos pares falsos positivos. Abaixo é mostrado o cálculo de distância média  $\sigma$  entre um objeto e outro através das distâncias entre os espectros de topologia dos Vértices associados  $Z_k$  e  $Z'_k$ :

$$\sigma = \frac{1}{\eta} \sum_{k=1}^{\eta} d(Z_k, Z'_k), \quad (7.9)$$

onde  $k$  é o índice do par de vértices correspondentes e  $\eta$  é o número total de pares correspondentes e a função  $d(Z_k, Z'_k)$  foi definida em 7.1. A média das distâncias Vértice-a-Vértice

já é uma métrica aceitável para uso prático nas comparações entre os objetos, porém ainda há como ser mais discriminante e explorar a correlação quanto aos índices dos vértices pareados e o percentual de correspondências entre os vértices, a depender do caso.

### 7.3.2 Percentuais de correspondências entre Vértices

Um parâmetro que também pode expressar semelhança entre os objetos após a comparação é o percentual de correspondências entre os vértices dos dois objetos, que é obtido pela simples razão do número de vértices correspondentes pelo número de vértices da envoltória convexa, porém os objetos em comparação podem possuir números diferentes de vértices e consequentemente os percentuais também são diferentes. Por exemplo, poderemos ter um objeto *template* contendo 10 vértices em sua envoltória sendo comparado com outro objeto (*query*) com 8 vértices e tendo como resultado 4 vértices correspondentes, os valores percentuais de correspondência para o *template* e o *query* são 40% e 50%, respectivamente.

Na métrica composta que está sendo proposta neste trabalho, usaremos os dois valores percentuais, chamaremos o valor percentual de correspondências de vértices do *template* de  $\alpha$ , e o valor percentual de correspondências de vértices do *query* de  $\beta$ . Como desejamos métricas de distâncias, usaremos as expressões  $1 - \alpha$  e  $1 - \beta$  para representar as distâncias de percentuais de *matching*, então quanto maior o valor percentual menor é a distância entre os objetos comparados, e vice e versa; os intervalos de valores para essas distâncias percentuais irão variar entre 0 e 1. As distâncias percentuais  $1 - \alpha$  e  $1 - \beta$  fazem parte da métrica final composta, que são integradas com outras métricas, a depender da abordagem, se simétrica ou assimétrica, tal integração será vista em detalhes a seguir.

### 7.3.3 Composição das Métricas

Para a composição de uma distância final que leve em consideração não apenas a média das distâncias dos pares de Vértices, mas que também abranja os valores de correlação dada por  $|\rho|$  e os percentuais de *matchings* de vértices que são representados por  $\alpha$  e  $\beta$ , é necessário antes entender os intervalos de valores para essas métricas; os intervalos de valores para  $\rho$ ,  $\alpha$  e  $\beta$  variam entre 0 e 1, cada um; já o valor da média das distâncias,  $\sigma$  varia entre 0 e o valor da amostragem, baseado no número de Raios emitidos por vértice ainda na etapa de Descrição explicada no Cap. 6 na subseção 6.2.2.2, pois a métrica de comparação Vértice-a-Vértice possui valores de distância que variam de 0 até o número de Raios por Vértice, então a média desses valores também estará dentro dessa faixa. Para os valores percentuais de *matching* de vértices,  $\alpha$  e  $\beta$ , quanto mais próximo de 1, melhor será para o quesito de semelhança entre os objetos em comparação; e o mesmo se aplica ao módulo do coeficiente de Pearson dos índices de vértices,  $|\rho|$ ; como desejamos construir uma métrica de distância de final, quanto menor o valor melhor é para efeito de semelhança.

É importante chamar a atenção, que para os casos em que o objeto *template* é simétrico não será possível utilizar o módulo do coeficiente de Pearson na construção da métrica final, pois

nesses casos existem problemas com *matchings* ambíguos em objetos simétricos, impedindo um cálculo correto de correlação entre os índices dos vértices correspondentes, esse problema será discutido mais adiante, na Seção 7.4. Por enquanto, podemos adiantar que a métrica final composta para esses casos conterà apenas os valores de distância percentual:  $1 - \alpha$  e  $1 - \beta$ ; e a média das distâncias Vértice-a-Vértice,  $\sigma$ .

Para formar uma métrica final composta de todas as métricas possíveis de acordo com cada abordagem, uma forma é utilizar um vetor que contenha os valores de tais métricas, porém deve-se atentar para os intervalos de valores das diferentes métricas componentes. Como já dito inicialmente, os intervalos de valores para  $1 - \alpha$ ,  $1 - \beta$  e  $1 - |\rho|$  pertencem a  $[0, 1]$ , e o intervalo para  $\sigma$  é  $[0, N]$ , onde  $N$  é o número de Raios emitidos por vértice. Dada essa diferença entre os valores de  $\sigma$  e as demais métricas, usaremos então o valor de  $\sigma$  como um escalar que multiplica o vetor que conterà as outras métricas restantes que variam entre 0 e 1. São mostradas as composições dessas métricas divididas entre abordagem simétrica e assimétrica.

### 7.3.3.1 Métrica composta para abordagem Simétrica

A métrica final composta para as situações em que o *template* é uma forma simétrica é obtida a partir de um vetor que contem em seus termos os valores de correspondência percentual dos pares de vértices, através da distância percentual:  $1 - \alpha$  e  $1 - \beta$ ; juntamente com o valor de  $\sigma$  que representa a média das distâncias Vértice-a-Vértice que possui intervalo definido em  $[0, N]$ , onde  $N$  é o número de Raios emitidos por vértice. Usaremos nessa métrica o valor de  $\sigma$  como um escalar que multiplica o vetor com as distâncias que variam entre  $[0, 1]$ , então o vetor fica definido como:  $\sigma \langle 1 - \alpha, 1 - \beta \rangle$ ; esse vetor possui duas dimensões e consegue reunir três métricas de distâncias, para a obtenção de um valor de distância final resultante, sendo assim podemos usar a magnitude desse vetor como a métrica de distância final entre os objetos em comparação, definida como:  $\sigma \sqrt{(1 - \alpha)^2 + (1 - \beta)^2}$ .

É importante observar que o módulo do vetor  $\langle 1 - \alpha, 1 - \beta \rangle$  possui valores que variam entre 0 e  $\sqrt{2}$ , e que há uma faixa de valores atenuantes entre  $[0, 1)$  e outra faixa com valores amplificadores, entre  $(1, \sqrt{2}]$ , como exibido na Figura 55. Na Figura 56 é mostrado um gráfico de uma função  $\mathbb{R}^2 \rightarrow \mathbb{R}$  expressa por  $\sqrt{(1 - \alpha)^2 + (1 - \beta)^2}$ , que são os valores de distâncias permitidos através de  $\alpha$  e  $\beta$ . Para fins de comparação entre objetos *templates* que podem ser simétricos ou assimétricos, deveremos **normalizar** os valores dos vetores para um intervalo comum, independente da abordagem, então dada essa razão iremos expressar a distância final para os casos com simetria na forma:

$$\| D \| = \frac{\sigma}{\sqrt{2}} \sqrt{(1 - \alpha)^2 + (1 - \beta)^2} \quad (7.10)$$

Como já visto, o intervalo do vetor  $\langle 1 - \alpha, 1 - \beta \rangle$  varia entre 0 e  $\sqrt{2}$ , então o termo  $\sqrt{2}$  no denominador limita o valor desse vetor, dando um intervalo de magnitude para  $\langle 1 - \alpha, 1 - \beta \rangle$  entre 0 e 1. Sendo assim, a Equação 7.10 é usada como métrica final para a classificação dos

objetos para os casos com *template* simétrico, num sistema onde se quer classificar o objeto inspecionado em comparação com todos os objetos contidos numa base de dados, portanto a comparação que resultar na menor distância indica fortemente que são objetos de mesma classe.

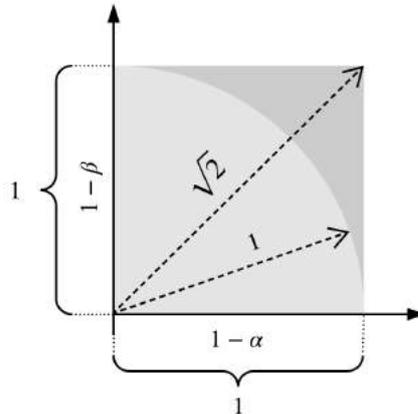


Figura 55 – Valores possíveis para o módulo do vetor  $\langle 1 - \alpha, 1 - \beta \rangle$ . Quando  $\alpha = 0$  e  $\beta = 0$  o vetor  $\langle 1 - \alpha, 1 - \beta \rangle$  possui o maior valor possível que é  $\sqrt{2}$ , e quando  $\alpha = 1$  e  $\beta = 1$ , o vetor possui o menor módulo que é 0. A região mais escura contém os valores possíveis entre 1 e  $\sqrt{2}$ .

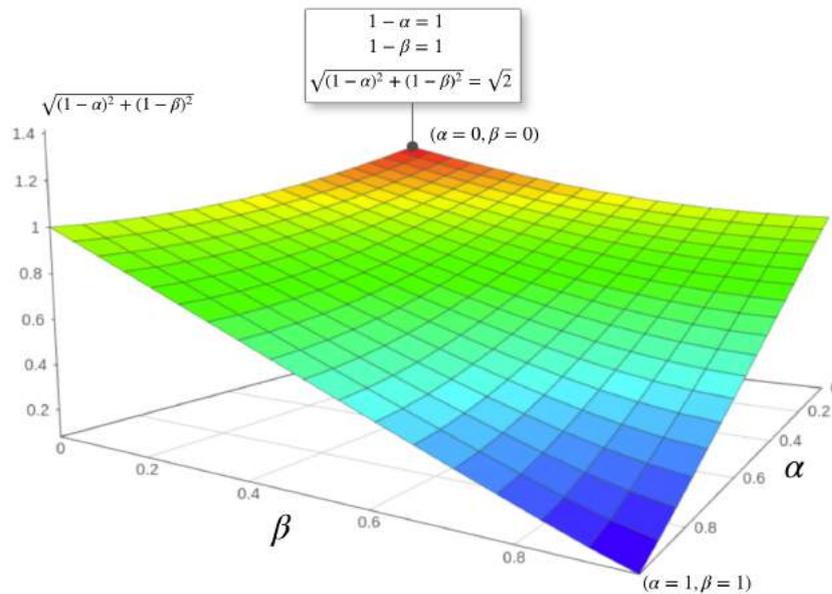


Figura 56 – Valores possíveis para o módulo do vetor  $\langle 1 - \alpha, 1 - \beta \rangle$ .

### 7.3.3.2 Métrica composta para abordagem Assimétrica

De maneira análoga à composição para a métrica de abordagem simétrica, nos casos com *templates* assimétricos teremos também uma composição onde as métricas estão sendo reunidas através de um vetor que integra as métricas, porém teremos também a contribuição do Coeficiente de Pearson em valor absoluto,  $\rho$ , que expressa a correlação entre os índices de Vértices. Sendo assim, teremos um vetor que contém em seus termos os valores de correspondência percentual dos pares de Vértices, através da distância percentual:  $1 - \alpha$  e  $1 - \beta$ ; e da correlação dos índices dos Vértices, com uso da distância de Pearson, na forma:  $1 - |\rho|$ ; juntamente com o valor de  $\sigma$  que representa a média das distâncias Vértice-a-Vértice que pertence ao intervalo  $[0, N]$ , onde  $N$  é o número de Raios emitidos por Vértice, então o vetor fica definido como:  $\sigma\langle 1 - \alpha, 1 - \beta, 1 - |\rho| \rangle$ ; esse vetor possui três dimensões e consegue reunir quatro métricas de distâncias, para a obtenção de um valor de distância final resultante, e assim podemos usar a magnitude desse vetor como a métrica de distância final entre os objetos em comparação, definida como:  $\sigma\sqrt{(1 - \alpha)^2 + (1 - \beta)^2 + (1 - |\rho|)^2}$ .

A magnitude do vetor  $\langle 1 - \alpha, 1 - \beta, 1 - |\rho| \rangle$  possui valores que variam entre 0 e  $\sqrt{3}$ , dado que os intervalos em cada dimensão sua possuem valores que variam entre 0 e 1, e a Figura 57 ilustra espacialmente esses valores.

De maneira similar à abordagem anterior, para fins de comparação entre objetos *templates* que podem ser simétricos ou assimétricos, deveremos **normalizar** os valores dos vetores para um intervalo comum, independente da abordagem, então iremos expressar a distância final para os casos com assimetria da seguinte forma:

$$\| D \| = \frac{\sigma}{\sqrt{3}} \sqrt{(1 - \alpha)^2 + (1 - \beta)^2 + (1 - |\rho|)^2} \quad (7.11)$$

O intervalo do vetor  $\langle 1 - \alpha, 1 - \beta, 1 - |\rho| \rangle$  varia entre 0 e  $\sqrt{3}$ , então o termo  $\sqrt{3}$  no denominador limita o valor desse vetor, dando um intervalo de magnitude para  $\langle 1 - \alpha, 1 - \beta, 1 - |\rho| \rangle$  entre 0 e 1. Sendo assim, a Equação 7.11 é usada como métrica final para a classificação dos objetos para os casos com *template* assimétrico; de forma parecida com a abordagem anterior o menor valor de distância durante a comparação é considerado a indicação dos objetos estarem na mesma classe.

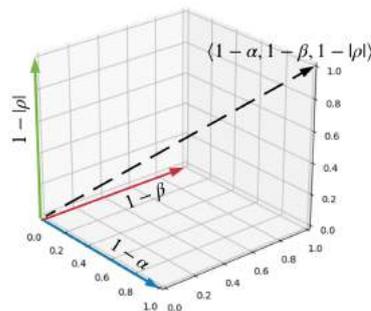


Figura 57 – Vetor  $\langle 1 - \alpha, 1 - \beta, 1 - |\rho| \rangle$  composto por diferentes métricas.

## 7.4 CLASSIFICAÇÃO PARA OBJETOS SIMÉTRICOS E ASSIMÉTRICOS

Foram vistas até agora todas as etapas de classificação de uma forma detalhada, quanto à comparação elemento-a-elemento, análise das correspondências dos pares de Vértices e remoção de falsos positivos e finalmente um valor final de distância entre os objetos comparados. O objetivo desta Seção é mostrar a integração dessas etapas numa visão macro do processo, onde também é feita uma distinção do processo de comparação e classificação para objetos **simétricos** e **assimétricos**, e em se tratando de objetos simétricos, teremos como exemplo principal o caractere ‘A’ (para algumas fontes), enquanto para objetos assimétricos o do caractere ‘G’.

O número total de etapas difere a depender de se o objeto contido na base de dados (*template*) é simétrico ou assimétrico durante a comparação com um objeto inspecionado (*query*): quando se trata de símbolos simétricos algumas etapas são desnecessárias e descartadas do processo, já para símbolos assimétricos na qual podemos explorar a disposição dos índices de Vértices através da correlação dos mesmos vista em 7.2.2.2. É importante destacar que os símbolos planares armazenados na base, os objetos *templates*, devem estar previamente rotulados como **Simétricos** e **Assimétricos**.

### 7.4.1 Classificação para objetos *template* simétricos

Após a geração dos descritores dos objetos *template* e armazenamento dos mesmos com o rótulo predefinido entre simétrico ou assimétrico, pode ser realizada agora a comparação entre os objetos contidos na base com outros objetos a serem inspecionados, *query*.

Para classificar objetos que possuem forma perfeitamente simétrica, teremos menos etapas de verificação e **não há** análise das associações de vértices, e conseqüentemente sem remoção de pares de Vértices falsos positivos e sem uma estimativa de correlação entre os índices de Vértices. Essa limitação se dá por conta da ambigüidade das associações entre formas simétricas; nessa situação o mesmo Vértice poderia estar associado com mais de um Vértice do outro objeto, e em ambos a associação está correta, porém não há como realizar uma análise dos Vértices em relação à disposição de Vértices pela razão cruzada de cinco pontos coplanares e a ordenação dos índices não é garantida por conta da ambigüidade. Baseado nessa limitação que a simetria da forma traz teremos um número reduzido de etapas em relação às comparações com formas assimétricas.

Para entender melhor as limitações relativas à análise durante a comparação das formas simétricas, é exibido a seguir, na Figura 58, o resultado entre formas simétricas onde estão ocorrendo ambigüidades em suas correspondências Vértice-a-Vértice: o Vértice  $V_1$ , do *template*, pode estar associado tanto com o vértice  $V_a$  quanto ao vértice  $V_b$  do *query*, o mesmo se aplica aos outros Vértices, onde cada um possui duas possibilidades de “casamento”, e em ambos os casos as associações estão corretas.

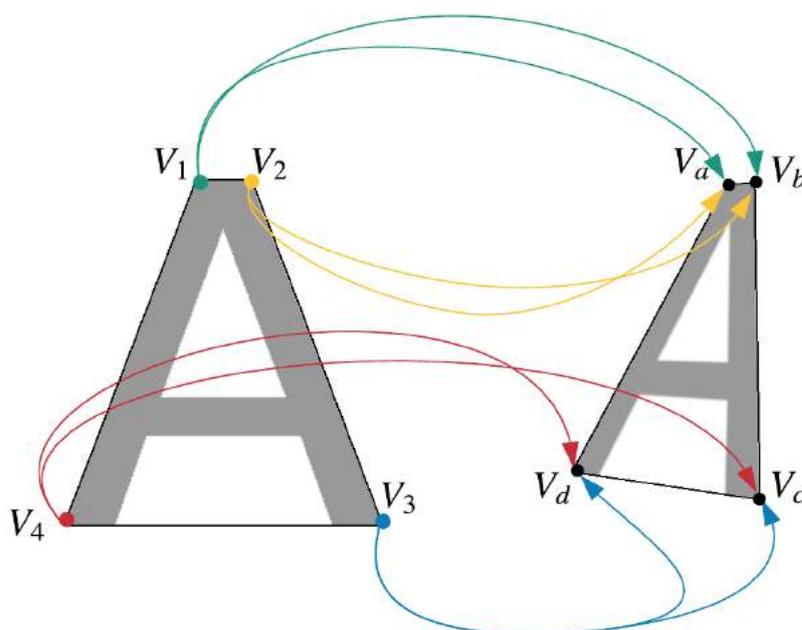


Figura 58 – Resultado da comparação entre os caracteres ‘A’ em diferentes vistas. Detalhe para as associações ambíguas entre os vértices.

Esse problema é comum em praticamente todos os descritores de forma, tanto para *features* locais, quanto para *features* que exigem mais informações, como é o caso dos descritores globais. Em se tratando do descritor abordado neste trabalho, os vértices são comparados por espectros de topologia que são obtidos pelo *matching* entre Raios, explicado em maiores detalhes na Subseção 7.1.2, e o resultado de tal comparação será um valor de distância Vértice-a-Vértice onde pareamos os vértices que possuem a menor distância formando associações; assim sendo, no contexto de formas simétricas as ambiguidades geradas irão possuir valores de distâncias iguais em teoria, mas na prática se observam valores próximos.

Em resumo, o que se observa na Figura 58, como já dito, o vértice  $V_1$  pode estar associado ao vértice  $V_a$  ou ao  $V_b$ , e o vértice  $V_2$  também possui as mesmas possíveis correspondências de  $V_1$ , esse tipo de confusão torna inviável uma análise dos Vértices quanto à correlação entre os índices dos pares de Vértices associados, e a remoção de associações falsas positivas pelo mesmo motivo, todas essas etapas vistas na Seção 7.2.

Abaixo, na Figura 59, é mostrado um fluxograma que contem as etapas do processo de classificação para os casos em que o objeto *template* possui simetria em sua forma. O sistema recebe os descritores de objetos planares como dados de entrada, e as primeiras etapas (em verde) da classificação tratam da comparação entre os elementos dos descritores, Raio-a-Raio, Vértice-a-Vértice em *loop* até que todos os Vértices de um descritor tenham sido comparados com todos os Vértices do outro descritor. Então são pareados os melhores casamentos entre Vértices, formando associações pela menor distância entre seus espectros de topologia. Todas essas etapas iniciais já foram discutidas na Seção 7.1; as etapas seguintes (em vermelho) são responsáveis pela computação da média das distâncias Vértice-a-Vértice e distância final entre os objetos em comparação. Já as etapas pertencentes a essa fase final foram vistas na Seção 7.3.

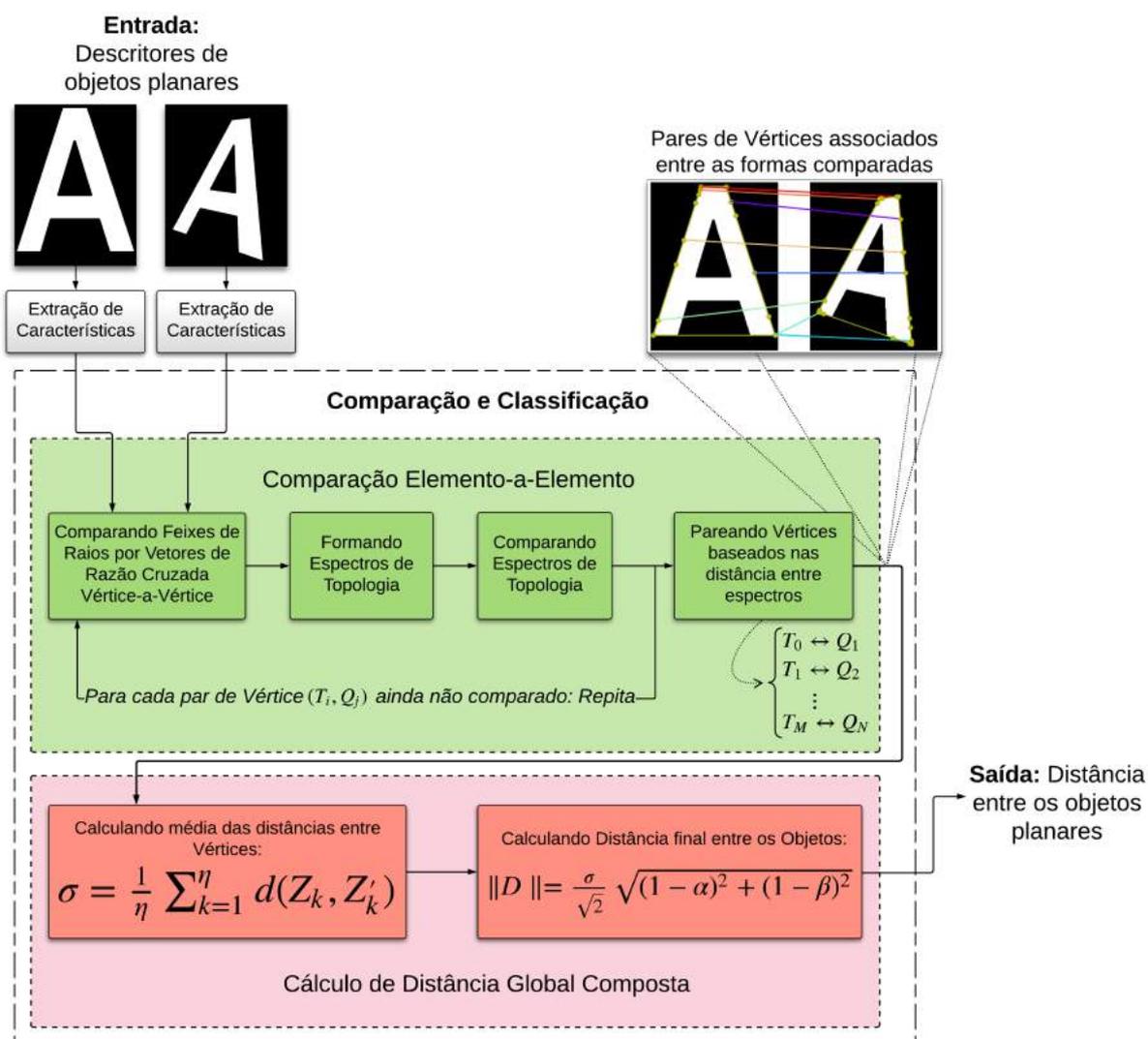


Figura 59 – Fluxograma de comparação e classificação entre objetos com formas planares simétricas. Descritores de forma de dois símbolos planares são recebidos como entrada, tendo o símbolo *template* como objeto previamente identificado como simétrico; as etapas em verde são responsáveis pela comparação entre os elementos de um descritor com os do outro; após a comparação entre os elementos é realizado o cálculo de distância entre os objetos comparados nas etapas em vermelho, e depois o valor da distância é repassado como saída do sistema.

Como visto nesta subseção, quando temos uma comparação com um objeto *template* **simétrico** sendo usado, as etapas de análise dos vértices das envoltórias (*convex-hull*) não são necessárias, tornando um processo mais simples e mais rápido, porém menos poderoso no quesito discriminação comparado aos casos em que se tem objetos *template* assimétricos, onde é possível extrair também valores de correlação entre os índices de Vértices, além da distância entre objetos.

### 7.4.2 Classificação para objetos *template* Assimétricos

Para a comparação e classificação onde existem objetos *template* que possuem forma assimétrica, teremos mais etapas de verificação além das etapas de comparação e cálculo de distância, onde também há a análise das associações de Vértices e remoção de pares de vértices falsos positivos e com uma estimativa de correlação entre os índices de Vértices através do cálculo do coeficiente de correlação de Pearson.

Se tratando de símbolos planares assimétricos é possível calcular o coeficiente de correlação de Pearson, pois não há o mesmo problema de ambiguidade apresentado nos casos com simetria, obviamente após o ajuste circular e remoção de falsos positivos via MMQ e Razão Cruzada de cinco pontos coplanares. Uma outra vantagem da abordagem para esses casos com assimetria é que, caso não haja nenhum par de vértices “sobreviventes” aos processos de remoção de falsos positivos, o sistema já descarta a possibilidade de as formas serem de mesma classe sem a necessidade de continuar para as etapas seguintes; isso é um grande vantagem da abordagem, pois possibilita a imediata recusa de símbolos que não possuem nenhuma chance de serem classificados como “iguais” (de mesma classe), diminuindo a chance de errar.

Abaixo, na Figura 60, é mostrado o fluxograma que possui as etapas do processo de classificação para as situações em que o objeto *template* é assimétrico; de maneira similar ao caso anterior, o sistema recebe os descritores de objetos planares como dados de entrada, e as primeiras etapas (em verde) da classificação tratam da comparação entre os elementos dos descritores, Raio-a-Raio, Vértice-a-Vértice em *loop* até que todos os vértices de um descritor tenham sido comparados com todos os vértices do outro descritor, então são pareados os melhores casamentos entre vértices, formando associações pela menor distância entre seus espectros de topologia: todas essas etapas iniciais já foram mostradas na Seção 7.1; a novidade contida neste processo são as etapas intermediárias de análise da disposição dos Vértices apresentadas em amarelo. Numa delas é realizado o ajuste circular dos índices dos vértices e um ajuste por MMQ, onde são removidos os primeiros falsos positivos pela exclusão dos *outliers*, e em seguida são removidos falsos positivos pela razão cruzada de cinco pontos coplanares; depois é analisado o percentual de pares de vértices sobreviventes no desvio condicional: caso todos os vértice tenham sido removidos ( $\alpha = 0$ ), o processo é abortado e as formas são classificadas como objetos pertencentes a classes diferentes. Caso o percentual de vértices seja maior que zero ( $\alpha > 0$ ), então o processo continua normalmente e há o cálculo do coeficiente de Pearson entre os índices dos Vértices, finalizando a análise dos Vértices das envoltórias estudadas na Seção 7.2. De maneira ainda parecida com a abordagem do caso simétrico, as etapas seguintes (em vermelho) são responsáveis pela computação da média das distâncias Vértice-a-Vértice e distância final entre os objetos, sendo que agora há o valor em módulo do coeficiente de Pearson  $|\rho|$  na composição da métrica de distância final. As etapas pertencentes a essa fase final do processo foram vistas na Seção 7.3.

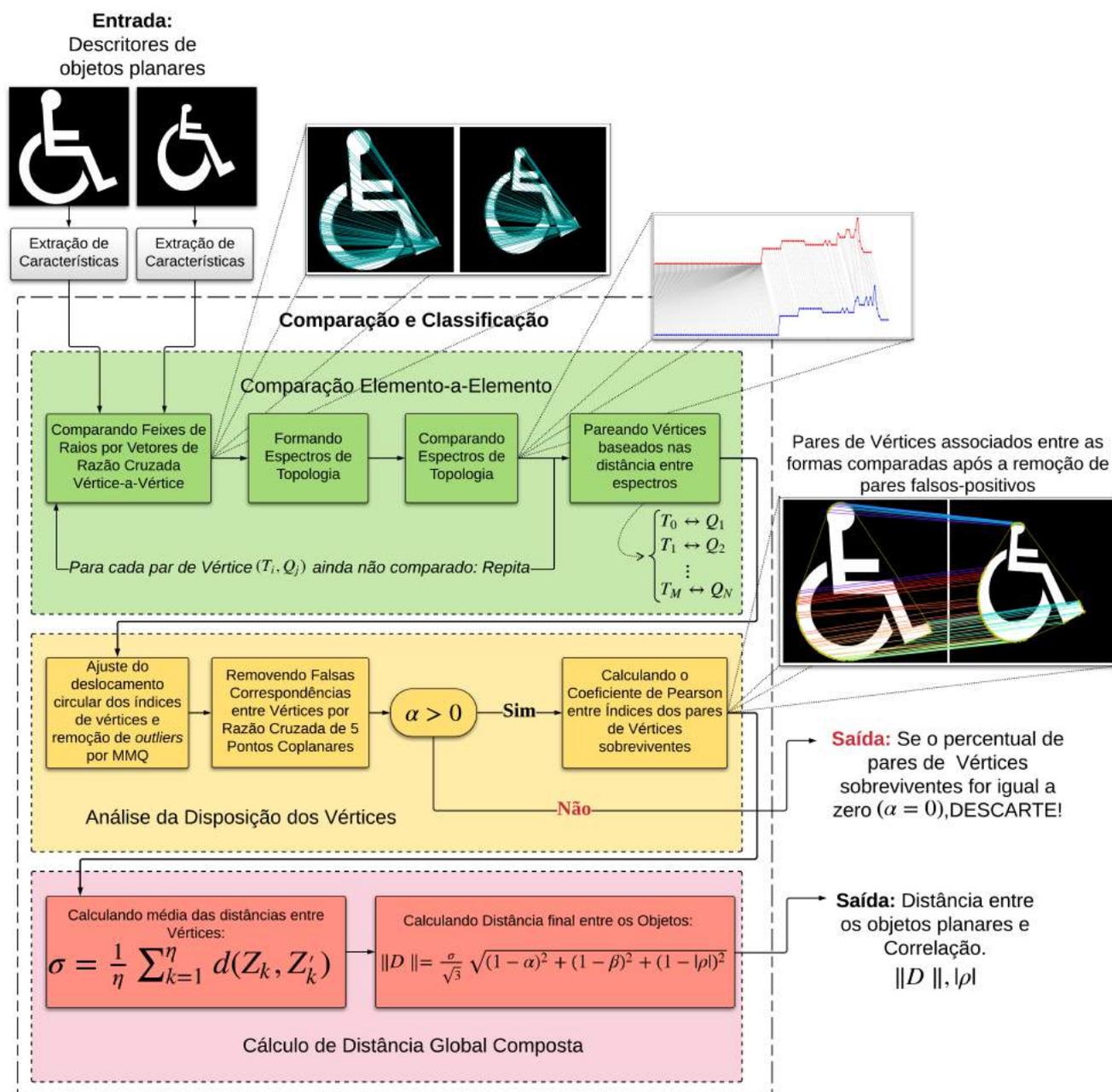


Figura 60 – Fluxograma de comparação e classificação entre objetos com formas planares assimétricas. Descritores de forma de dois símbolos planares são recebidos como entrada, tendo o símbolo *template* como objeto previamente identificado como assimétrico; as etapas em verde são responsáveis pela comparação entre os elementos de um descritor com os do outro; após a comparação entre os elementos, os pares de vértices associados são analisados nas etapas em amarelo e as associações falsas positivas são eliminadas; logo depois há uma verificação no número de pares sobreviventes que decide se aborta ou se dá prosseguimento onde há um cálculo da correlação, e por fim é realizado o cálculo de distância entre os objetos comparados nas etapas em vermelho, retornando o valor da distância  $\|D\|$  e o módulo da correlação,  $|\rho|$ .

Como visto nesta subseção que aborda os casos onde existe a comparação com um objeto *template* **assimétrico**, há a inclusão de uma fase intermediária a mais no processo, onde essa

fase apresenta etapas de remoção de associações falsas positivas e um cálculo de correlação; ainda nessa mesma fase existe uma verificação para o imediato descarte, indicando que são formas planares pertencentes a categorias diferentes, ajudando na classificação. E por fim há um cálculo de distância mais completo, que faz uso combinado da média entre distâncias Vértice-a-Vértice,  $\sigma$ , o percentual do número de vértices associados verdadeiros-positivos,  $\alpha$ , e ainda possui informação de correlação através do módulo do coeficiente de Pearson,  $|\rho|$ . Essas etapas adicionais oferecem uma grande vantagem na classificação, pois tendem a discriminar melhor com uma dimensão extra que representa a correlação, além da “limpeza” nas associações de pares de Vértices, porém aumenta-se o custo computacional para se realizar essas verificações.

## 7.5 TESTES DE COMPARAÇÃO PARA *TEMPLATE* SIMÉTRICOS E ASSIMÉTRICOS

Esta seção irá mostrar alguns exemplos práticos a fim de validar as partes do processo, onde será possível visualizar algumas etapas numa comparação entre objetos planares e realizará alguns estudos de casos com a finalidade de avaliar a métrica final de distância como resultado da comparação entre objetos de mesma classe e de classes diferentes, além do módulo do Coeficiente de correlação de Pearson para os casos onde o *template* é assimétrico. O primeiro estudo de caso irá comparar objetos simétricos e de mesma classe; o segundo estudo de caso irá abordar quando um objeto *template* simétrico for comparado com outro objeto de classes diferentes. O terceiro e quarto estudos de caso irão analisar a comparação de objetos *template* assimétricos entre objetos de mesma classe e entre classes diferentes, respectivamente. Espera-se que o valor da distância para a comparação entre objetos de mesma classe sejam menores quando comparados entre objetos diferentes. Para todos os casos, foram usados 300 raios por vértice.

### 7.5.1 Estudo de Caso I: Comparando objeto *template* simétrico com objeto de mesma classe

Este estudo de caso mostra a etapa final da comparação entre caracteres simétricos de mesma classe: foi escolhida a letra ‘A’ pertencente à fonte Arial Bold, em vista paralela frontal e a outra contendo uma deformação projetiva. O resultado da comparação são as associações entre os vértices de um objeto com o outro exibido na Figura 61. O número de correspondências é sete, para um objeto *template* contendo 17 vértices, e um *query* contendo 20 vértices e, com isso, podemos calcular um valor para o percentual de vértices correspondidos em relação ao *template*:  $\alpha = \frac{7}{17} = 0,4117647058823529$ ,  $\alpha \approx 41,2\%$ ; o percentual de vértices correspondidos do *query*:  $\beta = \frac{7}{20} = 0,35 = 35\%$  e o valor da média entre as distâncias Vértice-a-Vértice é  $\sigma = 7,538233264320222$ . Para compor a métrica de distância entre os objetos para os casos com a forma do *template* simétrico, não há a correlação dada por  $|\rho|$ , pelos motivos já explicados na Seção 7.4, e então a magnitude do vetor de distância desconsiderando o termo contendo o módulo do coeficiente de Pearson fica:  $\| D \| = \frac{\sigma}{\sqrt{2}} \sqrt{(1 - \alpha)^2 + (1 - \beta)^2} =$

$$\frac{7,538233264320222}{\sqrt{2}} \cdot \sqrt{(1 - 0,4117647058823529)^2 + (1 - 0,35)^2} = 4,67285576983035 \approx \mathbf{4,67}.$$

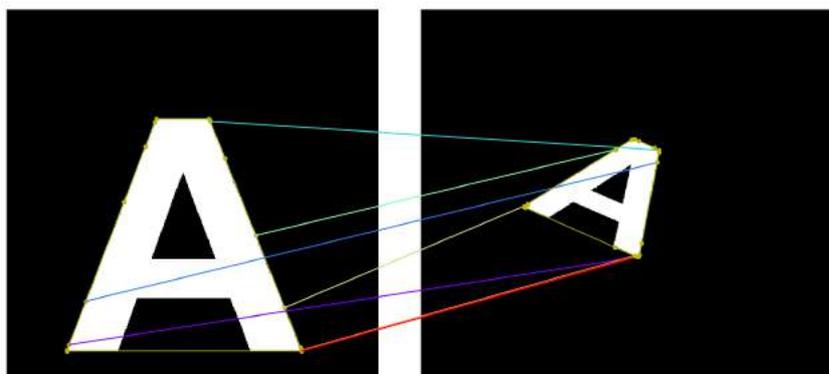


Figura 61 – Resultado da comparação entre os caracteres ‘A’ em fonte Arial Bold em diferentes vistas com correspondências entre sete vértices; o *template* (Esq.) possui 17 vértices com dimensões de  $400 \times 400$  pixels e o objeto *query* possui 20 vértices com  $730 \times 648$  pixels.

### 7.5.2 Estudo de Caso II: Comparando Objeto *template* simétrico com objeto de classe diferente

Para este estudo de caso foi comparado o mesmo caractere ‘A’ com deformação projetiva ao caractere ‘D’ como *template*. De maneira análoga ao estudo de caso anterior, após a comparação pode-se calcular a distância final através dos valores de  $\sigma$  e  $\alpha$ . Na Figura 62 são exibidas as associações entre os vértices de um objeto com o outro: o número de correspondências entre vértices é 10, para um objeto *template* contendo 45 vértices, e um *query* com 20 vértices; com isso podemos calcular um valor para o percentual de vértices correspondidos em relação ao *template*:  $\alpha = \frac{10}{45} = 0,2222\dots$ ,  $\alpha \approx 22,2\%$ ; o percentual de vértices correspondidos do *query*:  $\beta = \frac{10}{20} = 0,5 = 50\%$  e o valor da média entre as distâncias Vértice-a-Vértice é  $\sigma = 10,167074903318095$ . Então, de maneira similar ao estudo de caso I, a magnitude do vetor de distância fica:  $\| D \| = \frac{\sigma}{\sqrt{2}} \sqrt{(1 - \alpha)^2 + (1 - \beta)^2} = \frac{10,167074903318095}{\sqrt{2}} \cdot \sqrt{(1 - 0,2222222222)^2 + (1 - 0,5)^2} = 6,647347837465061 \approx \mathbf{6,64}$ .

Com esses dois primeiros estudos de caso com simetria e tendo como objeto *query* o mesmo caractere ‘A’ deformado projetivamente; foi possível mostrar que a métrica foi menor no primeiro caso, como esperado, onde se tratava da comparação entre objetos de mesma classe com o valor aproximado de **4,67**, e no segundo caso a métrica foi maior com valor igual a **6,64**, por se tratar de caracteres de classe diferente, ‘D’ e ‘A’.

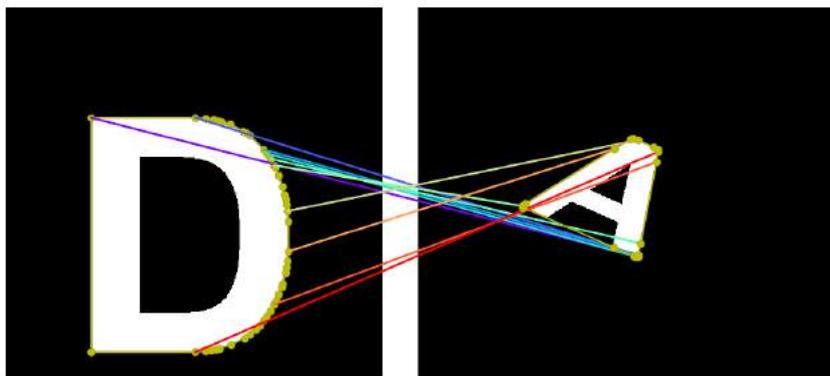


Figura 62 – Resultado da comparação entre os caracteres ‘D’ e ‘A’ em fonte Arial Bold em diferentes vistas com correspondências entre 10 Vértices; O *template* (Esq.) possui 45 Vértices com dimensões de  $400 \times 400$  *pixels* e o objeto *query* possui 20 Vértices com  $730 \times 648$  *pixels*.

### 7.5.3 Estudo de Caso III: Comparando objeto *template* assimétrico com objeto de mesma classe

Neste estudo de caso foram comparados dois objetos assimétricos e de mesma classe: o desenho de um sagitário em vista paralela frontal comparado ao mesmo símbolo contendo deformação projetiva. O resultado da comparação pode ser visto na Figura 63: o número de correspondências entre Vértices é 40, para um objeto *template* contendo 54 Vértices, e um *query* contendo 45 Vértices; com isso, podemos calcular um valor para o percentual de vértices correspondidos em relação ao *template*:  $\alpha = \frac{40}{54} = 0,7407407407407407 \approx 74,1\%$ ;

o percentual de vértices correspondidos do *query*:  $\beta = \frac{40}{45} = 0,8888888888888888 \approx 88,9\%$  e o valor da média entre as distâncias Vértice-a-Vértice é  $\sigma = 6,404085559045683$ . Para

compor a métrica de distância entre os objetos para os casos com a forma do *template* assimétrico, pode ser usado o módulo do coeficiente de Pearson como medida de correlação

entre os índices de vértices,  $|\rho| = 0,9973442303350751$ . Então a magnitude do vetor

de distância fica:  $\| D \| = \frac{\sigma}{\sqrt{3}} \sqrt{(1 - \alpha)^2 + (1 - \beta)^2 + (1 - |\rho|)^2} = \frac{6,404085559045683}{\sqrt{3}}$ .

$\sqrt{(1 - 0,7407407407407407)^2 + (1 - 0,8888888888888888)^2 + (1 - 0,9973442303350751)^2} = 1,0429559836290263 \approx \mathbf{1,04}$ .

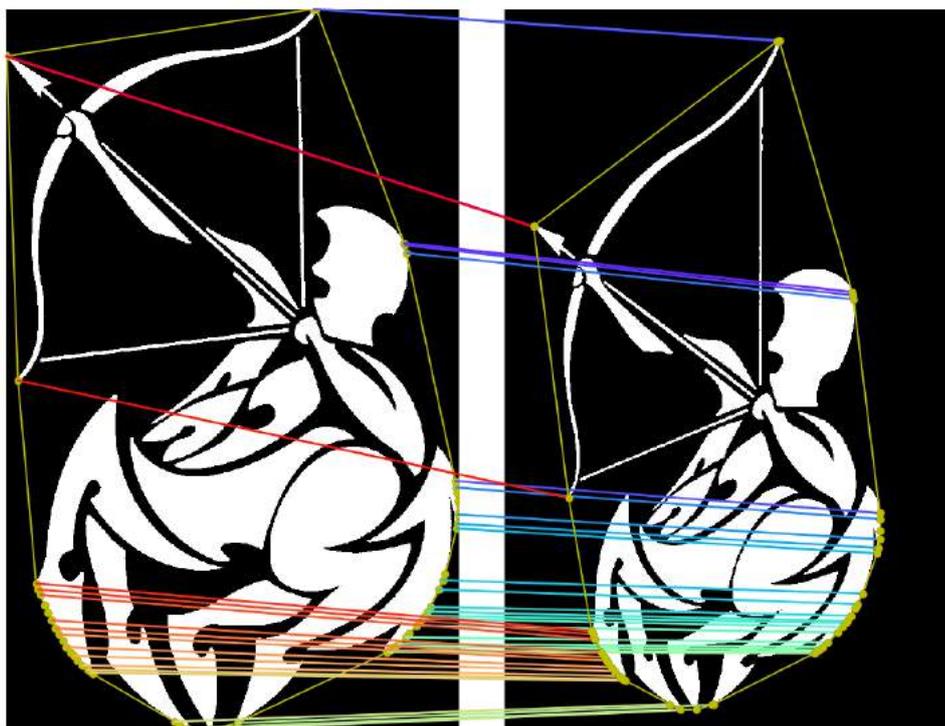


Figura 63 – Resultado da comparação entre os símbolos de um sagitário em diferentes Vistas com 40 pares de Vértices correspondentes; O *template* (Esq.) possui 54 Vértices com dimensões de  $670 \times 1076$  pixels e o objeto *query* possui 45 Vértices com  $670 \times 1076$  pixels.

#### 7.5.4 Estudo de Caso IV: Comparando objeto *template* assimétrico com objeto de classe diferente

Foram comparados dois objetos assimétricos de classes diferentes: o brasão da UFPE com o desenho de um sagitário contendo deformação projetiva. O resultado da comparação pode ser visto na Figura 64: O número de correspondências entre vértices é 10, para um objeto *template* contendo 24 Vértices, e um *query* contendo 45 Vértices; com isso, podemos calcular um valor para o percentual de vértices correspondidos em relação ao *template*:  $\alpha = \frac{10}{24} = 0,4166666666666667 \approx 41,6\%$ ; o percentual de vértices correspondidos do *query*:  $\beta = \frac{10}{45} = 0,2222222222222222 \approx 22,2\%$  e o valor da média entre as distâncias Vértice-a-Vértice é  $\sigma = 47,54895710350256$ . Para compor a métrica de distância entre os objetos para os casos com a forma do *template* assimétrico, poderá ser usado o módulo do coeficiente de Pearson como medida de correlação entre os índices de vértices,  $|\rho| = 0,28623699801737335$ . Então a magnitude do vetor de distância fica:  $\|D\| = \frac{\sigma}{\sqrt{3}} \sqrt{(1 - \alpha)^2 + (1 - \beta)^2 + (1 - |\rho|)^2} = \frac{47,54895710350256}{\sqrt{3}} \cdot \sqrt{(1 - 0,4166666666666667)^2 + (1 - 0,2222222222222222)^2 + (1 - 0,28623699801737335)^2} = 33,11030321907446 \approx \mathbf{33,11}$ . Como esperado, o Estudo de caso IV obteve uma distância maior do que o Estudo de caso III.

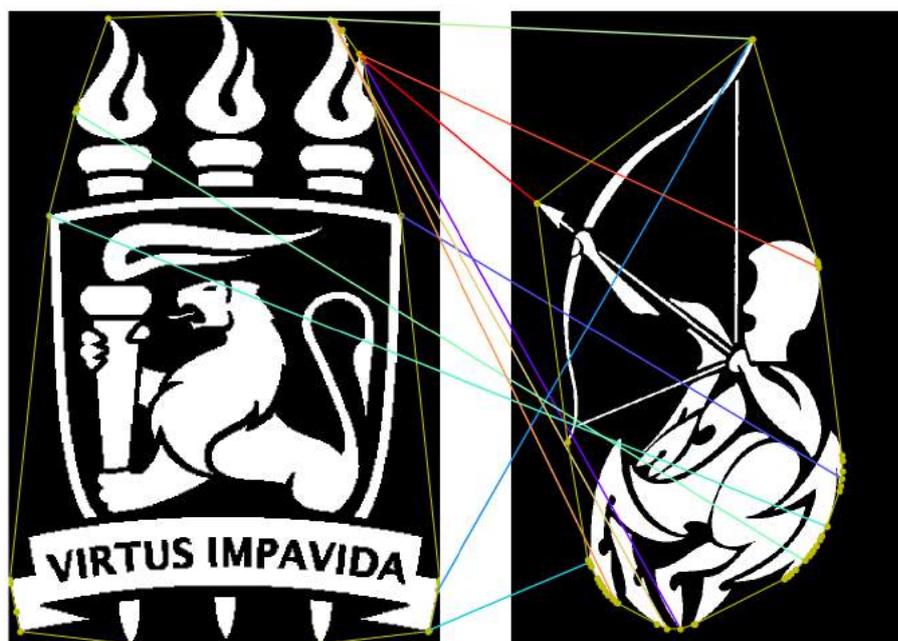


Figura 64 – Resultado da comparação entre o brasão da UFPE *template* com o sagitário em diferentes vistas possuindo 10 correspondências entre vértices; O *template* (Esq.) possui 24 vértices com dimensões de  $300 \times 446$  pixels e o objeto *query* possui 45 vértices com  $670 \times 1076$  pixels.

## 7.6 ABORDAGEM DE CLASSIFICAÇÃO POR ESTIMAÇÃO DE HOMOGRAFIA E ERRO DE TRANSFERÊNCIA

Além das abordagens complementares de Classificação para formas simétricas e assimétricas discutidas na Seção 7.4, foi desenvolvida uma terceira abordagem de Classificação que realiza a comparação entre elementos de maneira idêntica a das outras abordagens seguida de uma estimação de homografia entre os pares de Vértices correspondentes com remoção de *outliers* e cálculo de uma nova métrica de similaridade baseada em erro de transferência.

### 7.6.1 Métrica de similaridade baseada no erro de transferência

Para a estimativa da homografia é usado o algoritmo *Direct Linear Transformation* (DLT), explicado com mais detalhes em (HARTLEY; ZISSERMAN, 2004), que faz uso de no mínimo quatro correspondências de pontos para encontrar uma transformação que leva de um conjunto de pontos ao outro; no nosso caso, estamos trabalhando com muito mais do que quatro pares de pontos correspondentes, temos um sistema sobre-determinado e para isso foi usado um método de regressão *Least Median of Squares* (LMS) (ROUSSEEUW, 1984) para encontrar a melhor homografia; esse método substitui a soma pela mediana dos resíduos ao quadrado. O LMS funciona muito bem, se houver menos de 50% de *outliers* nas correspondências e tem a vantagem sobre o RANSAC (*RANdom SAMple Consensus*) que não requer limiares para número mínimo de pontos (DUBROFSKY, 2009).

Depois da estimativa correta da homografia  $H$  (estamos estimando a homografia dos pontos do objeto de cena, *query*, para os pontos do objeto *template*) calculamos o erro de transferência (*transfer error*) em uma imagem. O erro de transferência, também conhecido como distância geométrica, é a distância Euclidiana entre o ponto mapeado pela a homografia e onde o ponto correspondente foi originalmente encontrado. O total de erro de transferência para um conjunto de pontos correspondentes  $x_i \rightarrow x'_i$  é dado por:

$$\sum_i d(x'_i, Hx_i)^2, \quad (7.12)$$

onde  $d(x'_i, Hx_i)$  é a distância Euclidiana entre os pontos  $x'_i$  e  $Hx_i$ . A métrica final entre os objetos é uma variação desse erro total de transferência: é a média das distâncias Euclidianas das correspondências consideradas *inliers* pelo LMS, ficando dessa forma:

$$\frac{1}{n} \sum_i^n d(x'_i, Hx_i), \quad (7.13)$$

onde  $n$  é o número de correspondências *inliers*. Para os experimentos realizados que visam reconhecimento, testamos com as métricas já discutidas no capítulo anterior, na Seção 7.4, e com essa nova métrica mostrada nesta seção.

A Fig. 65 ilustra os Vértices correspondentes e os pontos estimados por uma transformação do objeto *query* para o objeto *template* através da homografia  $H$ , pode-se observar os erros gerados após a transformação pela distância entre os pontos  $x'_i$  e os pontos  $Hx_i$ , onde  $i$  representa o índice dos pares de Vértices correspondentes.

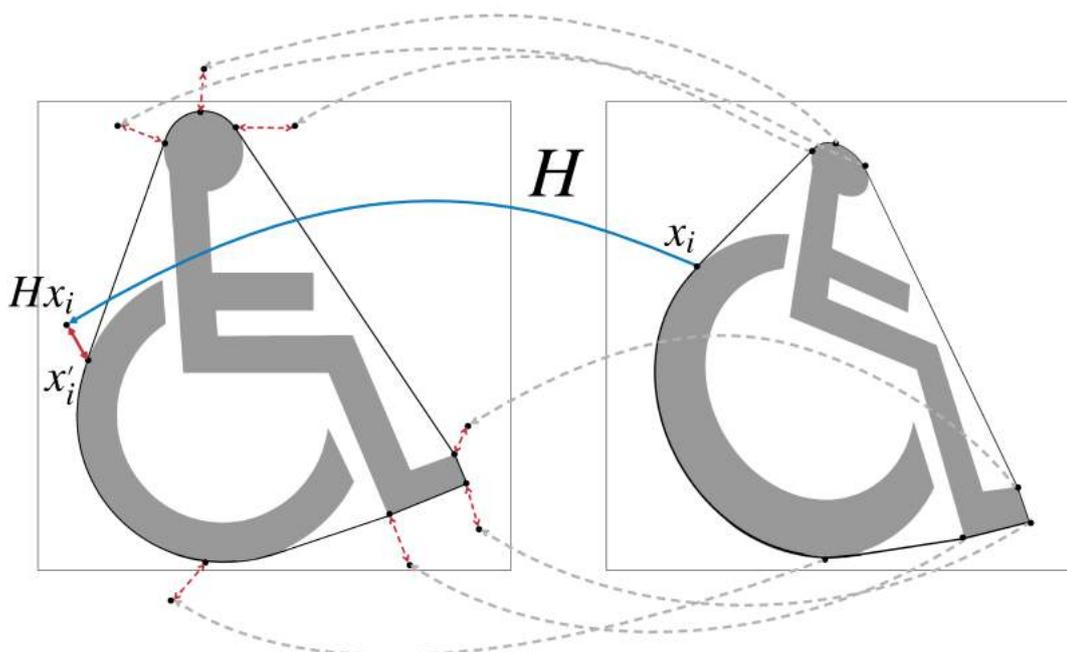


Figura 65 – Estimativa da homografia dos Vértices do objeto *query*  $x_i$  para os Vértices  $x'_i$  do objeto *template* e erro de transferência.

Sendo assim, esta terceira abordagem de Classificação possui as seguintes semelhanças e diferenças: a etapa de comparação Elemento-a-Elemento continua a mesma, mas a etapa de remoção de falsos positivos e cálculo do Coeficiente de Pearson (usados somente na comparação para formas assimétricas), e a última etapa de cálculo de distância (comum a ambas) foram eliminadas e substituídas pela a estimativa de homografia e cálculo de distância final baseado no erro de transferência, como ilustrado em azul na Fig. 66. Dessa forma foi possível tornar o processo de Classificação muito mais simples e eficiente.

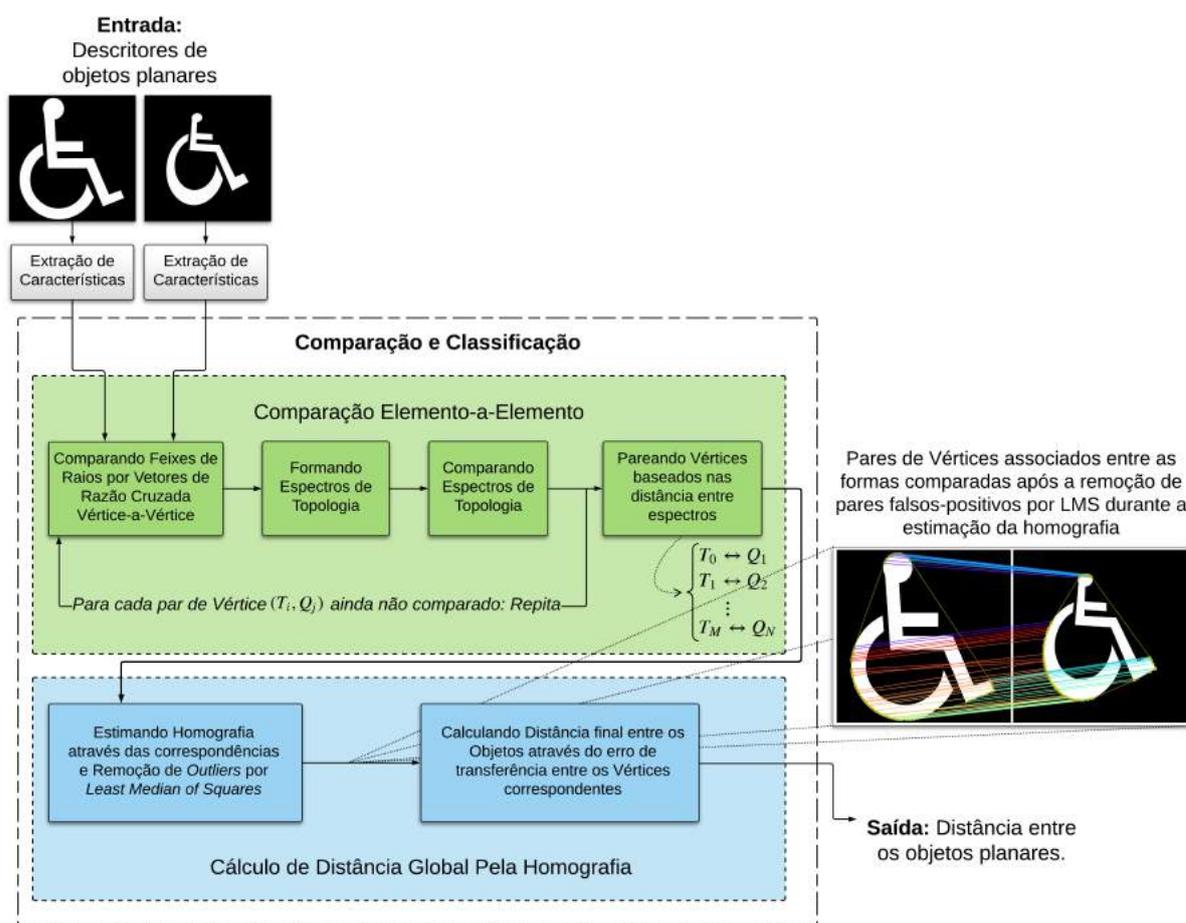


Figura 66 – Fluxograma da nova abordagem utilizada para a classificação contendo novas etapas responsáveis pela a estimativa da homografia, remoção de *outliers* e cálculo de distância baseado no erro de transferência usada como métrica de similaridade entre os objetos. Essas novas etapas em azul substituem as etapas finais das abordagens simétricas e assimétricas, apenas as etapas em verde de comparação Elemento-a-Elemento do descritor continuam idênticas.

## 7.7 DISCUSSÃO

Foram explicadas neste capítulo todas as etapas de comparação e classificação entre os descritores. Inicialmente foram mostradas todas as etapas de comparação elemento-a-elemento, onde os elementos são pareados hierarquicamente: primeiramente, tenta-se gerar correspondências Raio-a-Raio pela comparação dos vetores de razão cruzada, que por sua vez extraem-se espectros topológicos que viabilizam as comparações Vértice-a-Vértice através dos espectros de topologia com uso da DTW, ou de uma maneira menos custosa, por um simples biparticionamento, dividindo os espectros em regiões ativas e inativas, usando a DTW apenas na comparação entre as regiões ativas e calculando o erro das regiões inativas pela distância de Hamming, de maneira trivial e mais barata computacionalmente, e finalmente soma-se o erro das duas partes. Dessa forma cada Vértice do *template* é comparado com todos os Vértices do *query* através dessa distância entre os espectros, e a menor distância entre esses indica que os Vértices são correspondentes. O resultado dessa fase é uma coleção de correspondências que é passada para a fase seguinte.

Foi mostrado que existem diferenças nas etapas de classificação para objetos simétricos e assimétricos, e que após a comparação entre os elementos dos descritores e com o uso de objetos *template* simétricos, é então possível realizar a análise da disposição dos Vértices das envoltórias correspondentes, e que tal verificação ajuda a elaborar uma métrica de classificação mais completa, mas que só é possível para objetos *template* que não possuem simetria.

A fase final da classificação consiste em estimar um valor de distância entre os dois objetos comparados, através da composição de alguns valores dependendo se o caso se trata de uma comparação com *template* simétrico ou assimétrico.

Além das duas abordagens já discutidas, foi desenvolvida também uma terceira abordagem baseada na estimativa de homografia por DLT e uma nova métrica de similaridade baseado na média do erro de transferência dos pares de Vértices correspondentes. Essa nova abordagem, mais simples, possui a etapa inicial de comparação Elemento-a-Elemento idêntica as outras duas abordagens diferenciando-se apenas durante segunda etapa dessa abordagem.

---

## 8 EXPERIMENTOS E RESULTADOS

---

Este capítulo apresenta os experimentos e a análise dos resultados obtidos. Os testes realizados têm por objetivo atestar a robustez do descritor proposto diante de transformações projetivas, as transformações foram geradas através da variação dos ângulos de *azimute* e *elevação*, e os testes foram comparados com uma técnica concorrente mais atual no contexto de descritores robustos a projetividades com o objetivo de comparar os valores de acurácia, onde a acurácia é a razão do número de objetos reconhecidos corretamente sobre o número de objetos de consulta (*query*).

Também foram realizados testes com cenas reais capturadas sob diferentes pontos de vista para diferentes símbolos e logomarcas extraídas por uma segmentação baseada na limiarização, existem diversas estratégias citadas no Cap. 2 para a segmentação de formas planares em cenas naturais que também poderiam ser utilizadas como extratores de símbolos para o descritor proposto. Alguns outros testes simulados com oclusão também foram realizados visando explorar a robustez do descritor proposto e os limites das correspondências entre os pares de vértices dos objetos em comparação.

Dessa forma, dividiremos os experimentos da seguinte maneira:

- Testes com imagens sintéticas possuindo deformações projetivas variadas
  - Testes comparativos
    - \* Testes comparativos com caracteres
    - \* Testes comparativos com símbolos e logomarcas
    - \* Testes comparativos com formas extraídas do *MPEG-7 dataset*
  - Testes individuais com oclusão
- Análise das correspondências dos símbolos e logomarcas obtidos de cenas reais

Além dessa divisão estruturada de testes, é discutido também a linguagem utilizada no desenvolvimento dos algoritmos e os tempos médios de execução das etapas de Extração de Características e Classificação.

## 8.1 TESTES COM IMAGENS SINTÉTICAS POSSUINDO DEFORMAÇÕES PROJETIVAS VARIADAS

A primeira parte desta seção exibe os testes para o reconhecimento entre as formas planares sem deformação projetiva (*templates*) e com deformações projetivas geradas artificialmente (*query*); esses testes são feitos nas mesmas condições no descritor proposto e com um outro descritor concorrente (HCNC), infelizmente não foi possível realizar testes com o CRS por não possuirmos o código do mesmo, foi obtido apenas o código do HCNC, que é a técnica concorrente mais atual. A segunda parte desta seção é reservada para alguns testes realizados com oclusão simulada em diferentes níveis apenas com o descritor proposto com a finalidade de explorar os limites do descritor também nessas condições.

### 8.1.1 Testes Comparativos

Esta subseção exibe os resultados dos experimentos realizados com caracteres e logomarcas em diferentes ângulos de *azimute* (*az*) e *elevação* (*el*), ângulos pertencentes ao sistema horizontal de coordenadas celestes usados em cartografia e astronomia ilustrado na Fig. 67.

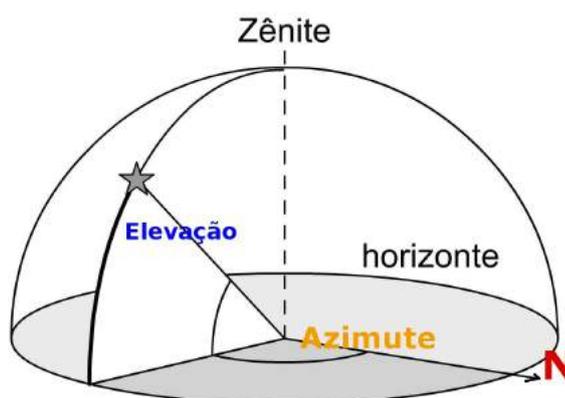


Figura 67 – Sistema horizontal de coordenadas celestes: Os ângulos de azimute e elevação são relativos à localização da estrela com respeito ao centro. O azimute é o ângulo que começa partindo do Norte geográfico e cresce no sentido horário variando de  $0^\circ$  a  $360^\circ$ ; o ângulo de elevação varia de  $0^\circ$  a  $90^\circ$ , do horizonte até o zênite.

#### 8.1.1.1 Testes comparativos com Caracteres

A metodologia de testes foi seguida de forma parecida com a do trabalho de Li e Tan (LI; TAN, 2010). Um conjunto de imagens *templates* ( $400 \times 400$ ) em vista paralela frontal, contendo 25 caracteres maiúsculos do alfabeto latino e mais nove dígitos, resultando em 34 símbolos alfanuméricos em fonte *Arial Bold* foi usado nos experimentos para a geração dos objetos *templates*. Para esse conjunto de experimentos, não foi incluído o caractere ‘I’ por não apresentar características que possam ajudar na descrição, os caracteres ‘N’ e ‘Z’ foram considerados de mesma

classe pela grande semelhança, e o dígito ‘0’ também não foi incluído por conta da grande semelhança com o caractere ‘O’. O conjunto de testes foi dividido em três ângulos de elevação: 10°, 20° e 50°; para cada ângulo de elevação foram variados três ângulos de azimute: 0°, 30° e 90°, gerando  $34 \times 3 = 102$  símbolos deformados para cada ângulo de elevação. Para cada experimento com um ângulo de elevação fixado foram realizadas 3468 ( $34 \times 102$ ) comparações entre as formas. Abaixo, na Figura 69 é exibido o caractere ‘A’ nos diferentes ângulos de azimute e elevação utilizados nos experimentos.

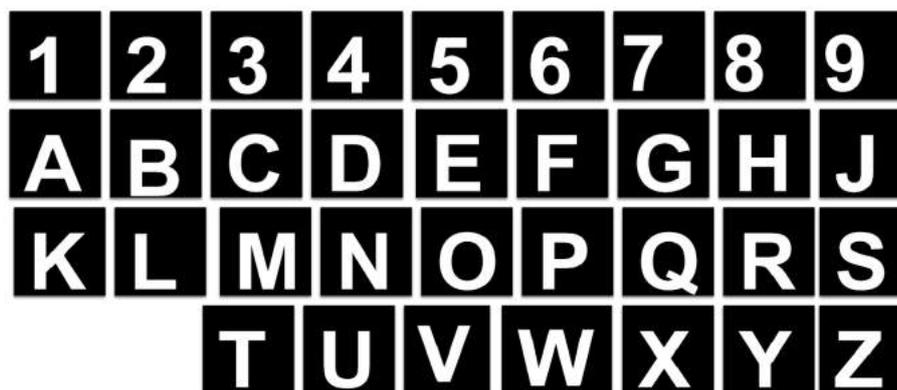


Figura 68 – Conjunto de caracteres Arial Bold utilizados como *templates* nos experimentos.

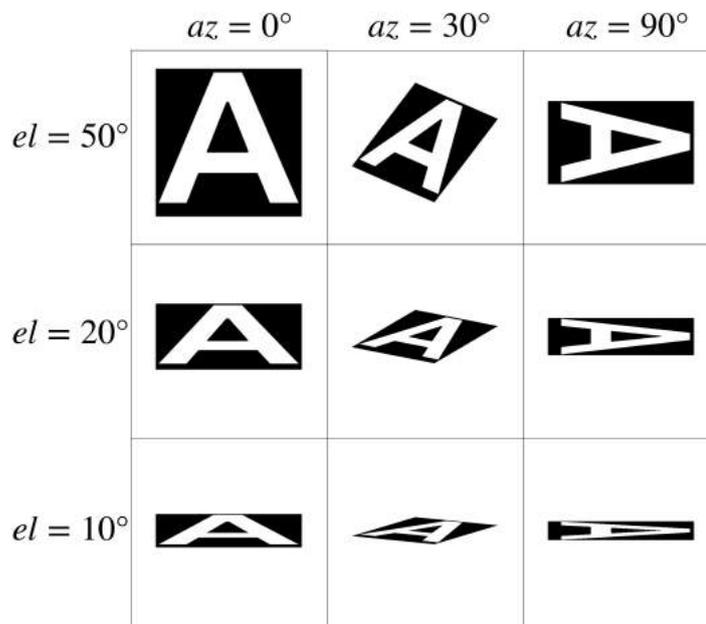


Figura 69 – Exemplos do caractere ‘A’ com deformações geradas a partir de variações de ângulos de azimute e elevação usadas nos experimentos.

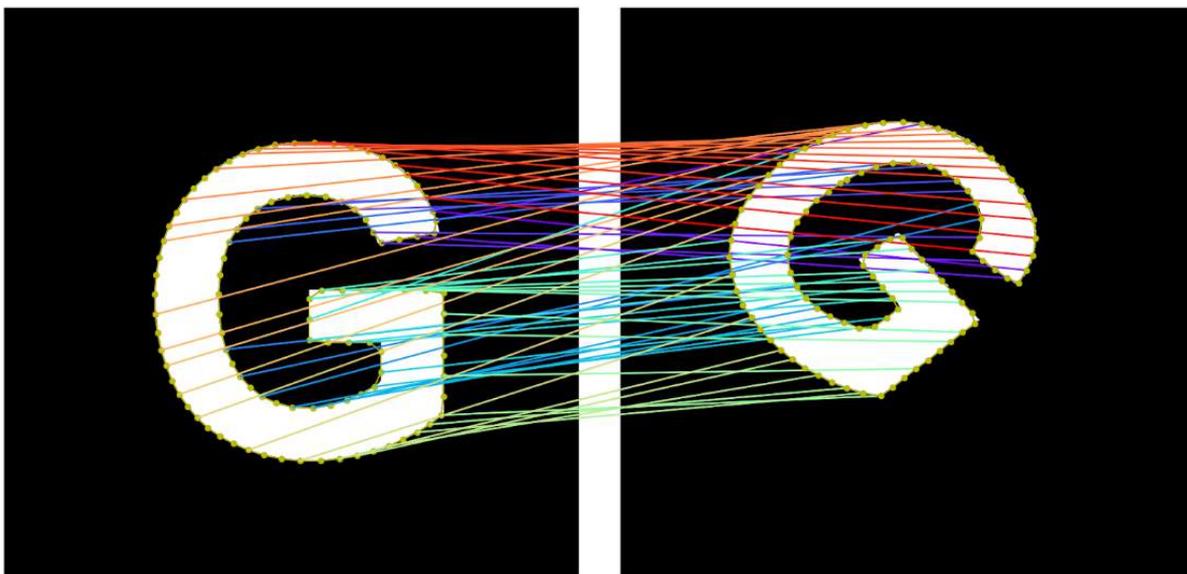


Figura 70 – Exemplo de comparação entre caracteres de mesma classe ('G') com pontos nos invólucros de contorno e suas correspondências. Imagem *query* com deformação de  $el=50^\circ$  e  $az=30^\circ$ .

<b>Elevação</b>	10°	20°	50°
<b>Acurácia</b>	57,2%	59,8%	61,7%

Tabela 6 – Resultado dos experimentos para caracteres com uso do **descritor proposto** para  $el = 10^\circ$ ,  $el = 20^\circ$ ,  $el = 50^\circ$  com amostragem de 150 vértices e 150 raios por vértices com uso de métrica baseado na métrica proposta.

<b>Elevação</b>	10°	20°	50°
<b>Acurácia</b>	61,7%	68,6%	84,3%

Tabela 7 – Resultado dos experimentos para caracteres com uso do **descritor proposto** para  $el = 10^\circ$ ,  $el = 20^\circ$ ,  $el = 50^\circ$  com amostragem de 150 vértices e 150 raios por vértices com uso de métrica pela projeção com a estimativa de homografia.

<b>Elevação</b>	10°	20°	50°
<b>Acurácia</b>	67,6%	83,3%	96,1%

Tabela 8 – Resultado dos experimentos para caracteres com uso do HCNC.

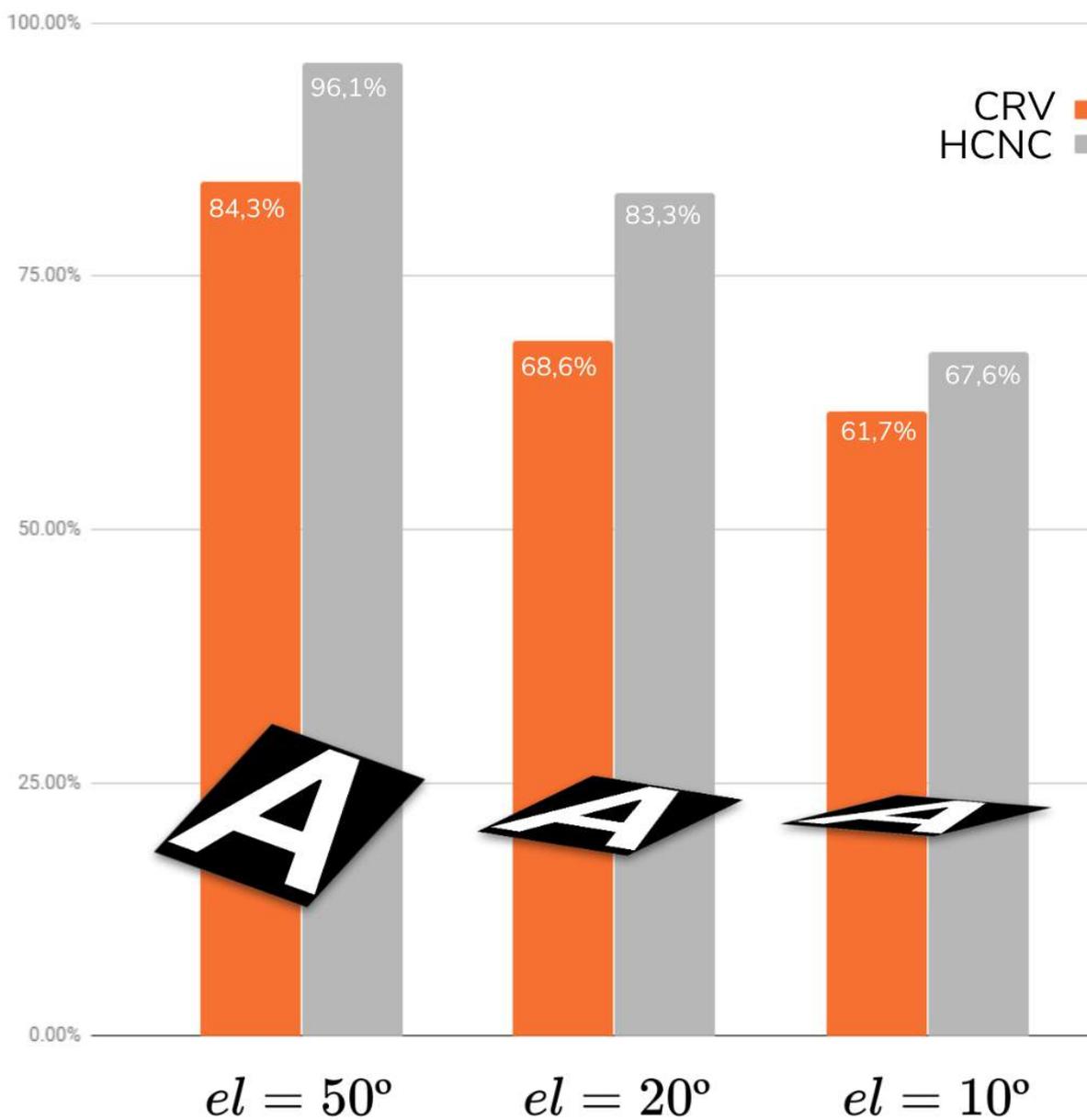


Figura 71 – Gráfico de barras com valores de acurácia do HCNC e do CRV (*Cross Ratio Vector*), o método proposto, com uso de métrica baseada na estimação de homografia para caracteres. A maior diferença de acurácia foi obtida nos testes para  $el = 20^\circ$ , uma diferença de 14,7%.

### 8.1.1.2 Testes comparativos com símbolos e logomarcas

Para os testes com símbolos e logomarcas, foram utilizadas dez formas *templates*, a Fig. 72 exibe esse *dataset* de *templates*, e de maneira similar aos testes realizados em caracteres, foram geradas variações de cada uma dessas formas através dos ângulos de elevação e azimute para a formação dos objetos de teste (*query*); os conjuntos de testes foram divididos em três ângulos de elevação: 10°, 20° e 50°; para cada ângulo de elevação foram variados três ângulos de azimute: 0°, 30° e 90°, a Fig. 73 mostra as variações do mesmo símbolo com diferentes transformações. Então, para cada experimento com ângulo de elevação fixado, temos  $3 \times 10 = 30$  formas compondo o *dataset* de objetos de consulta, resultando num total de 300 ( $30 \times 10$ ) comparações entre objetos *query* e *template* para cada experimento com um ângulo de elevação fixado.



Figura 72 – Conjunto de imagens dos objetos *templates* usados nos experimentos e suas dimensões.

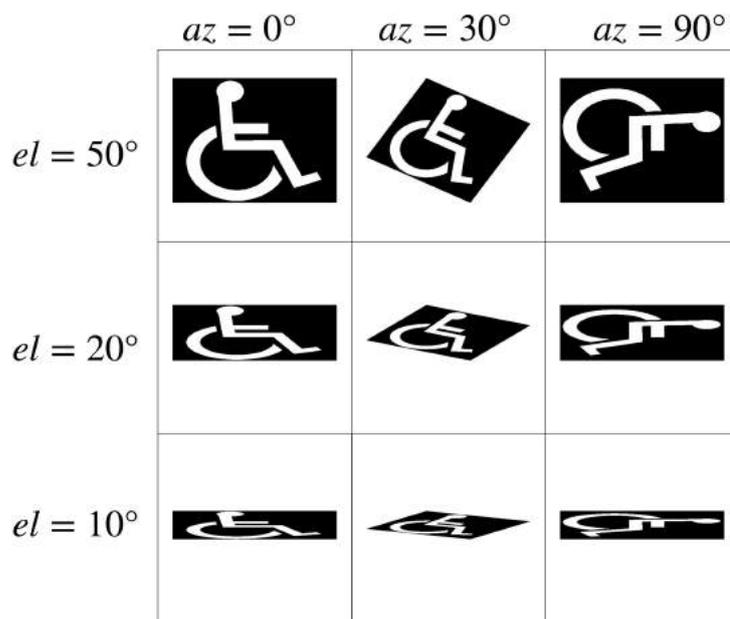


Figura 73 – Exemplos do símbolo de acessibilidade com deformações geradas através das variações de ângulos de azimute e elevação usadas nos experimentos.

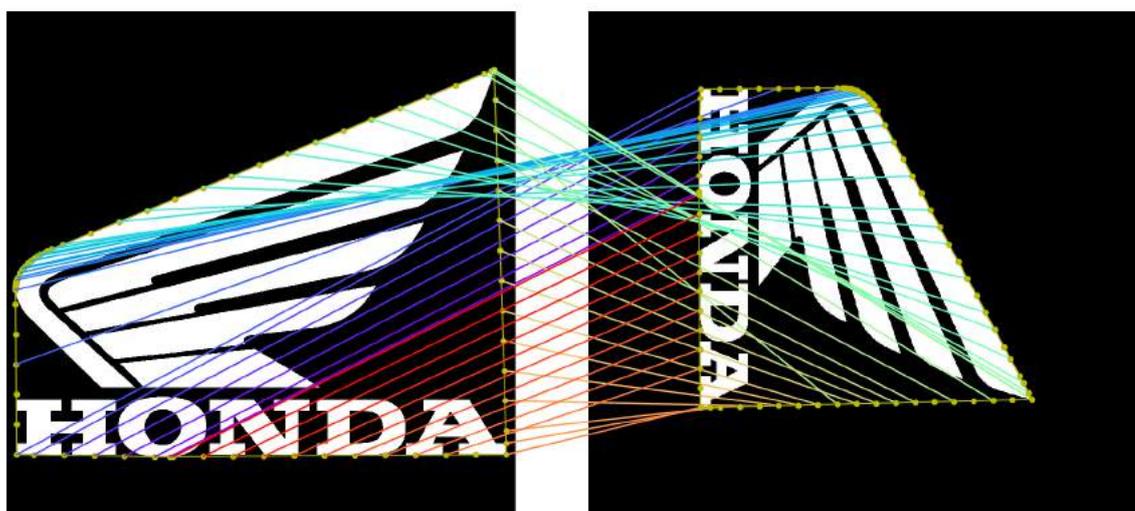


Figura 74 – Exemplo de comparação entre formas de mesma classe com pontos adicionais nos invólucros convexos e suas correspondências. Imagem *query* com deformação de  $el=50^\circ$  e  $az=90^\circ$ .

Elevação	$10^\circ$	$20^\circ$	$50^\circ$
Acurácia	68.6%	90%	73.3%

Tabela 9 – Resultado dos experimentos para símbolos e logomarcas com uso do **descritor proposto** para  $el = 10^\circ$ ,  $el = 20^\circ$ ,  $el = 50^\circ$  com 150 vértices e 150 raios por vértices com uso de métrica baseado na métrica proposta.

Elevação	10°	20°	50°
Acurácia	90%	100%	100%

Tabela 10 – Resultado dos experimentos para símbolos e logomarcas com uso do **descritor proposto** para  $el = 10^\circ$ ,  $el = 20^\circ$ ,  $el = 50^\circ$  com 150 vértices e 150 raios por vértices com uso de métrica pela reprojeção com a estimativa de homografia.

Elevação	10°	20°	50°
Acurácia	23%	93%	100%

Tabela 11 – Resultado dos experimentos para símbolos e logomarcas com uso do HCNC.

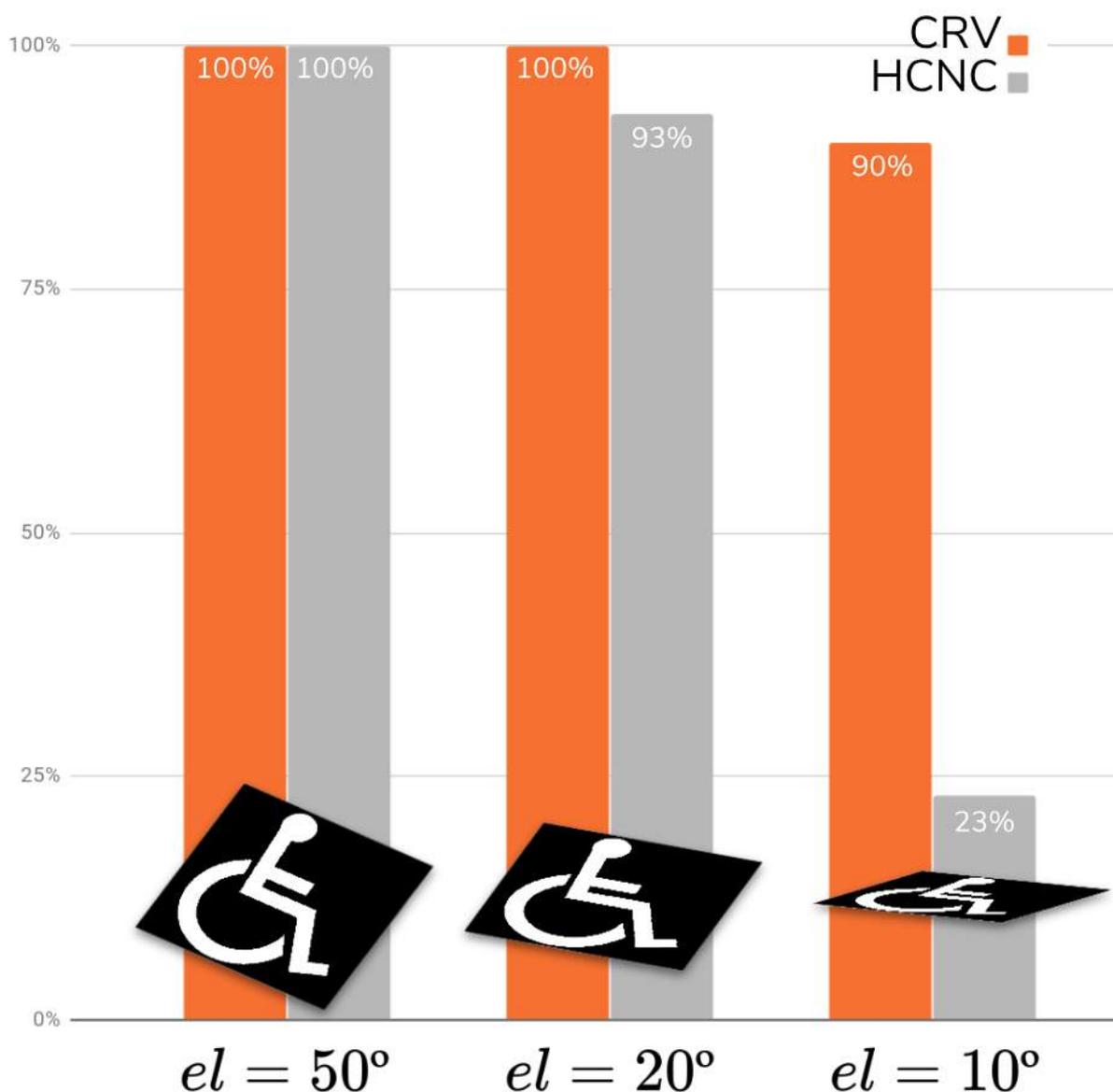


Figura 75 – Gráfico de barras com valores de acurácia do HCNC e do CRV com uso de métrica baseada na estimação de homografia para símbolos e logomarcas. O CRV errou apenas no reconhecimento da logomarca da Nike em  $el = 10^\circ$ , o HCNC teve um desempenho bem abaixo do CRV nos testes para objetos deformados com  $el = 10^\circ$ , uma diferença de 67% da acurácia.

### 8.1.1.3 Testes comparativos com formas extraídas do MPEG-7 *dataset*

No trabalho de (JIA et al., 2016) foram realizados testes em dois *datasets*: o MCD (*Multi-view Curve Dataset*) e o PLD (*Projective Landmark Dataset*); ambas as bases são subconjuntos do MPEG-7 com algumas formas selecionadas onde foram aplicadas deformações projetivas aleatórias nas mesmas, o MCD e o PLD incluem 40 e 10 classes, respectivamente. Além dessas duas bases utilizadas, os autores do HCNC construíram também um *dataset* usando um exemplo de cada uma das 70 classes do MPEG-7 e gerando novas formas através de transformações projetivas de maneira análoga a construção do MCD e do PLD.

Neste trabalho houveram várias buscas para a obtenção do MCD e do PLD, porém não se obteve êxito, foi obtido o MPEG-7 *dataset* onde selecionamos aleatoriamente, a partir de 12 classes, uma forma para cada classe e aplicamos transformações projetivas a cada forma através das variações de ângulos de azimute e elevação de maneira análoga aos testes comparativos realizados anteriormente, produzindo  $3 \times 12 = 36$  formas para cada ângulo de elevação. De maneira análoga aos testes anteriores os conjuntos de testes foram divididos em três ângulos de elevação:  $10^\circ$ ,  $20^\circ$  e  $50^\circ$ ; para cada ângulo de elevação foram variados três ângulos de azimute:  $0^\circ$ ,  $30^\circ$  e  $90^\circ$ , a Fig. 73 mostra as variações do mesmo símbolo com diferentes transformações. Então, para cada experimento com ângulo de elevação fixado, temos 36 formas compondo o *dataset* de objetos de consulta, resultando num total de 432 ( $36 \times 12$ ) comparações entre objetos *query* e *template* para cada experimento com um ângulo de elevação fixado.

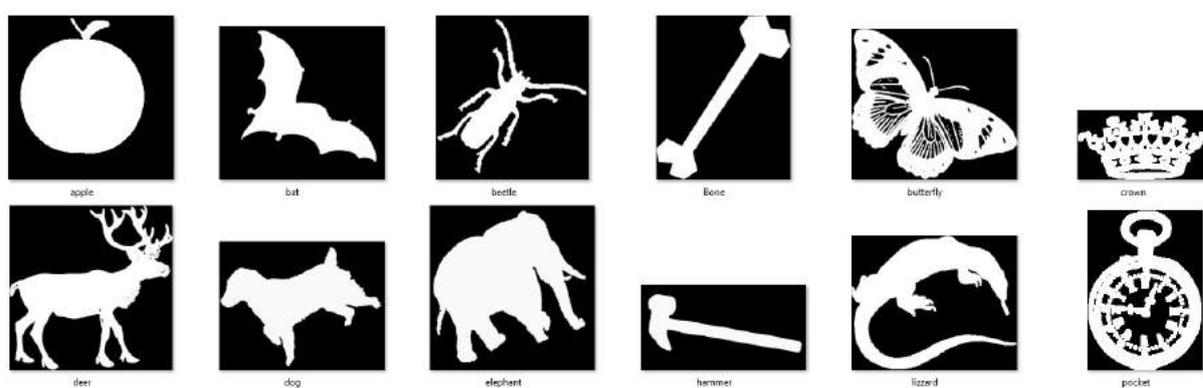


Figura 76 – Conjunto de doze imagens dos objetos *templates* extraídos do MPEG-7 *dataset* usados nos experimentos.

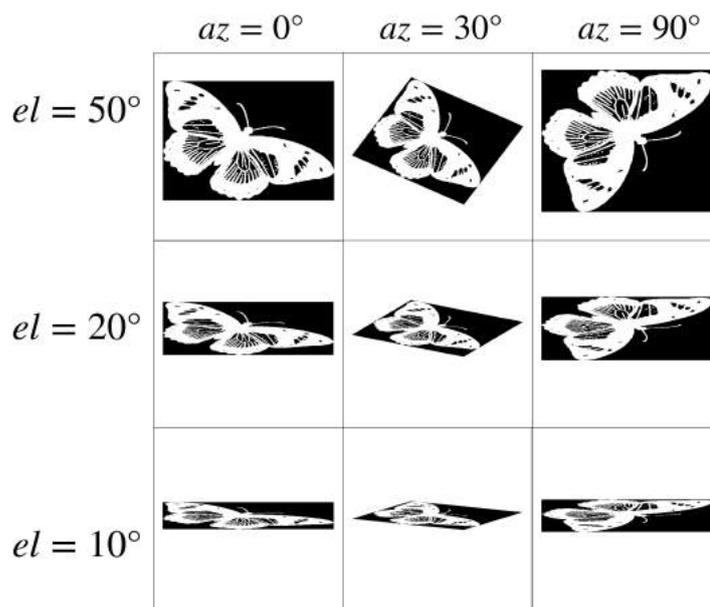


Figura 77 – Exemplos de deformações geradas da forma de borboleta extraído do MPEG-7 geradas através das variações de ângulos de azimute e elevação usadas nos experimentos.

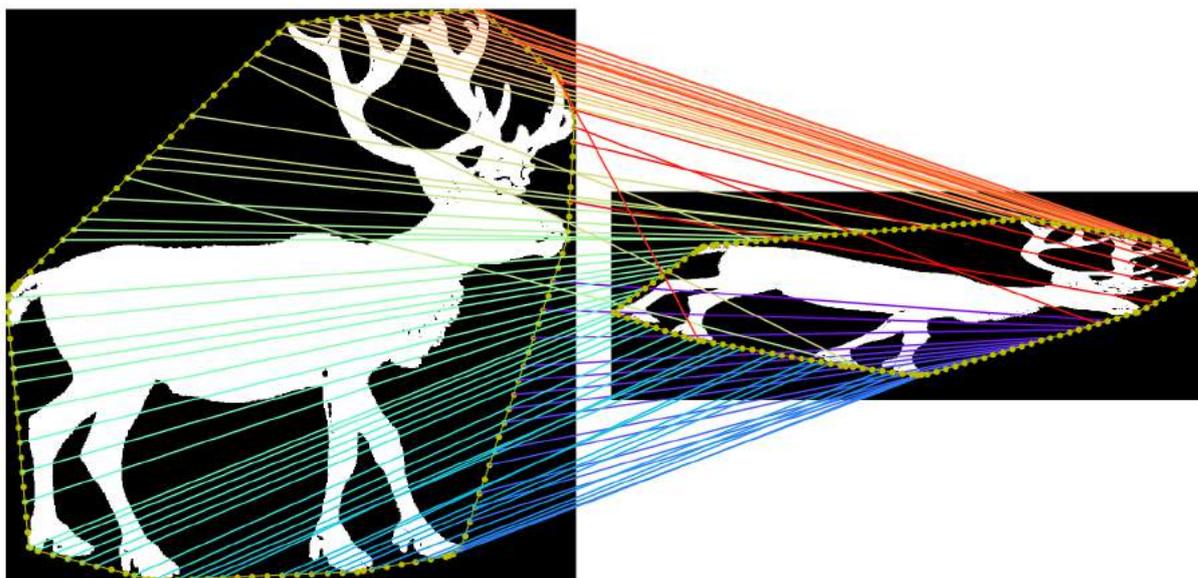


Figura 78 – Exemplo de comparação entre formas de mesma classe com pontos adicionais nos invólucros convexos e suas correspondências. Imagem *query* com deformação de  $el=10^\circ$  e  $az=30^\circ$ .

<b>Elevação</b>	10°	20°	50°
<b>Acurácia</b>	75%	83,3%	83,3%

Tabela 12 – Resultado dos experimentos para formas extraídas do MPEG-7 *dataset* com uso do **descritor proposto** para  $el = 10^\circ$ ,  $el = 20^\circ$ ,  $el = 50^\circ$  com 150 vértices e 150 raios por vértices com uso de métrica baseado na métrica proposta.

<b>Elevação</b>	10°	20°	50°
<b>Acurácia</b>	77,7%	88,8%	91,6%

Tabela 13 – Resultado dos experimentos para formas extraídas do MPEG-7 *dataset* com uso do **descritor proposto** para  $el = 10^\circ$ ,  $el = 20^\circ$ ,  $el = 50^\circ$  com 150 vértices e 150 raios por vértices com uso de métrica pela reprojeção com a estimativa de homografia.

<b>Elevação</b>	10°	20°	50°
<b>Acurácia</b>	91,6%	100%	100%

Tabela 14 – Resultado dos experimentos para formas extraídas do MPEG-7 *dataset* com uso do HCNC.

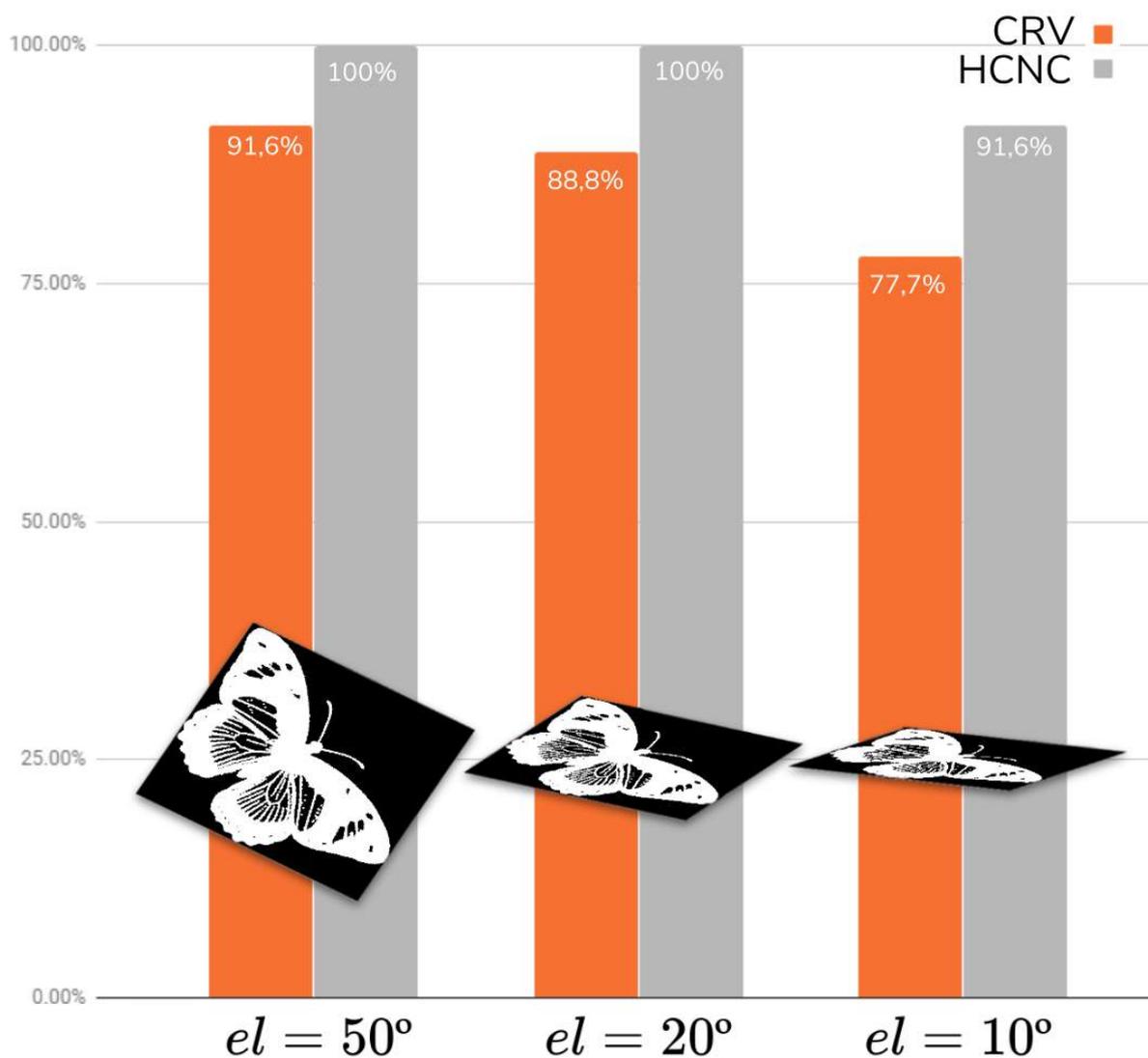


Figura 79 – Gráfico de barras com valores de acurácia do HCNC e do CRV com uso de métrica baseada na estimativa de homografia para formas extraídas do MPEG-7 *dataset*. O CRV não errou em nenhum objeto rica em estruturas internas como a *crown*, *butterfly* e o *pocket*. A maior diferença percentual obtida foram nos testes para  $el = 10^\circ$ , uma diferença de 13,9%.

#### 8.1.1.4 Testes comparativos com deformações mais acentuadas

Foi realizado um pequeno conjunto de testes com cinco objetos *templates*, sem deformações, já conhecidos (Peugeot, Polo, Puma, Sagitário e UFPE) confrontadas com cinco versões deformadas desses mesmos símbolos no intuito de comparar os valores das métricas de similaridade. As imagens de teste foram geradas a partir do software de manipulação gráfica Gimp 2, que oferece a ferramenta de transformação em perspectiva, como ilustrado na Fig. 80. A Fig. 81 exibe o conjunto de objetos deformados gerados contendo deformações avulsas. Para esse experimento foi utilizado o descritor proposto e o HCNC.

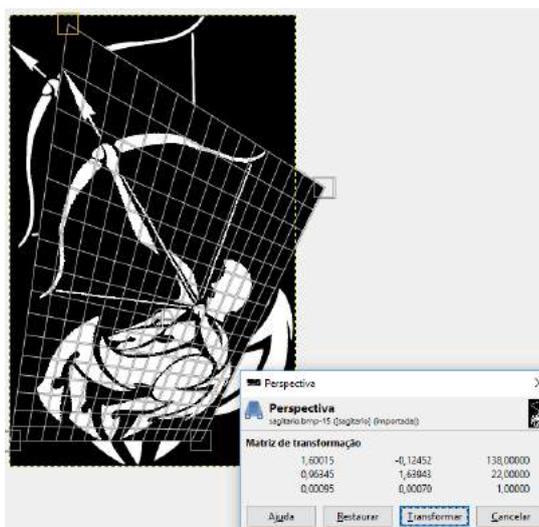


Figura 80 – Ferramenta de transformação projetiva do Gimp 2, e matriz de transformação.

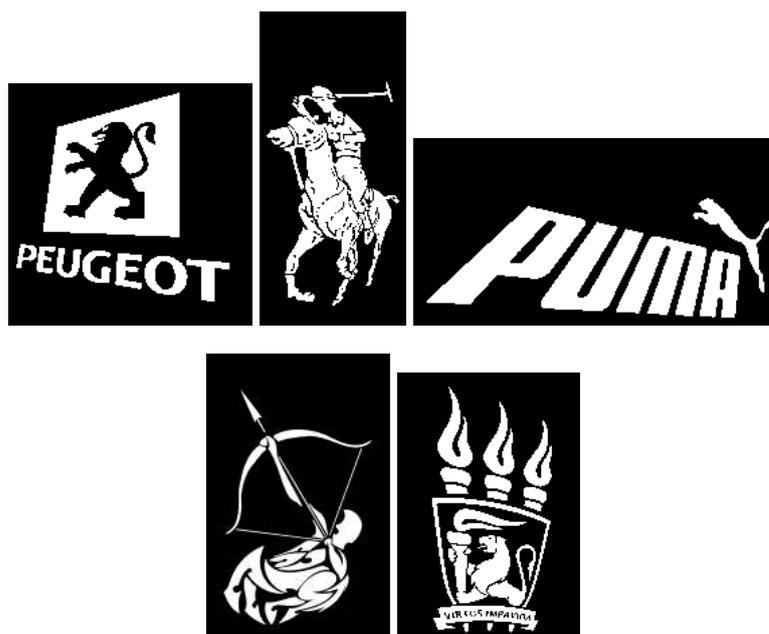


Figura 81 – Conjunto dos símbolos utilizados nos testes sob o efeito de transformações projetivas variadas.

					
peugeot	211,645	321,887	363,736	338,328	368,41
polo	330,625	265,301	367,173	356,413	365,056
puma	340,314	361,685	388,374	365,208	349,023
sagitário	343,318	359,119	381,229	338,075	314,65
ufpe	368,877	367,986	375,32	337,236	90,1538

Tabela 15 – Matriz de distâncias resultante das comparações realizadas através do **HCNC**. Houve duas classificações erradas.

					
peugeot	2,10102	363,218	41,6492	40,9396	19,4936
polo	43,5925	0,213807	43,6224	36,3337	39,4343
puma	54,7467	57,7181	1,56115	1,10349	33,4779
sagitário	37,5751	41,0238	28,5382	0,511174	18,5048
ufpe	43,4973	44,186	15,9783	7,15083	0,134791

Tabela 16 – Matriz de distâncias resultante das comparações realizadas através do **Descritor proposto** com uso da métrica baseada no erro de transferência com amostragem de 50 vértices e 150 raios por vértices.

As Tabelas 15 e 16 exibem as matrizes de confusão com os valores de distância obtidos das comparações entre as formas mencionadas originais (sem deformação) com sua versão deformada projetivamente. Como foi possível reparar nesses experimentos, o HCNC não foi tão resistente para classificar objetos sob condições de deformação projetiva mais severas, já o descritor proposto conseguiu acertar todas as comparações realizadas. Além disso, é importante observar os valores de similaridade apresentados: no HCNC, na maiorias das comparações realizadas de um objeto teste (*query*) com os *templates* os valores eram proporcionalmente próximos; na coluna da Tabela 15 que contém o Puma deformado projetivamente, além do reconhecimento ter falhado nesse caso para HCNC, os valores são bem próximos, já na coluna referente ao mesmo objeto na Tabela 16, o reconhecimento está correto e apresenta um valor de distância de similaridade muito pequeno comparado aos outros valores da mesma coluna, mostrando que os valores obtidos pelo descritor proposto indicam corretamente o objeto mais semelhante com um grau de confiança maior do que o descritor concorrente.

### 8.1.2 Testes individuais com oclusão

Em cenas reais, um objeto planar pode estar incompleto, faltando partes por conta de uma oclusão ou por erros durante o processo de segmentação. Com a finalidade de testar os limites do descritor proposto quanto a problemas dessa natureza, foram realizados testes com oclusão em um objeto deformado projetivamente comparado com um objeto *template* sem deformação. Foram aplicadas diferentes proporções de oclusão nas formas com o objetivo de verificar até onde o descritor continua fornecendo correspondências corretas entre os Vértices.

Diferente dos testes anteriores, nesse conjunto de testes estamos interessados apenas em analisar qualitativamente a comparação entre objetos de mesma classe contendo oclusões em suas formas. Os resultados são analisados pelas correspondências entre os objetos das imagens, não se tem a finalidade de explorar o reconhecimento, mas são exibidos para caráter informativo os valores em absoluto do Coeficiente de Pearson obtidos. O objetivo é apenas mostrar a robustez do descritor quanto às correspondências entre os Vértices em condições mais críticas como essas. A análise é feita baseada na razão entre o número de correspondências corretas (verdadeiras-positivas) e o número de associações totais entre os Vértices dos invólucros convexos obtidas após a comparação à medida que a oclusão cresce percentualmente. Os valores percentuais de oclusão são obtidos pela razão da área subtraída pela área total do objeto.

A Fig. 82 exibe as correspondências entre os vértices dos invólucros convexos entre as logomarcas Polo Ralph Lauren, com e sem deformação projetiva. O número de associações obtidas foi de 30, e dessas associações: 100% são correspondências verdadeiras-positivas. À medida que cresce a oclusão, há um aumento no número das associações falsas-positivas, como podem ser vistas nas Figuras 83, 84, 85, 86, 87, 88 e 89, onde o percentual de oclusão varia de 9, 13% até 91, 38%, e o percentual de falsos positivos varia de 4, 17% até 81, 25% no pior caso. O nível de erro percentual acompanha a taxa de oclusão ficando atrás em valores percentuais, e mesmo nos piores casos ainda há correspondências corretas entre as logomarcas comparadas. Isso é bastante surpreendente em se tratando de um descritor de forma **global**, pois segundo (LI; TAN, 2010) a oclusão é um problema para esses tipos de descritores, e isso se deve à necessidade de uso de informações globais das formas para encontrar as correspondências. A explicação pela qual o descritor proposto se mostra resistente a oclusões leves e severas é que há uma grande quantidade de raios emitidos a partir de cada vértice do *convex-hull*, e o espectros são comparados após o *matchings* de raios, sendo mais flexíveis na comparação. A seguir são exibidos os resultados das comparações contendo oclusão: as formas deformadas em miniatura exibem partes sinalizadas em vermelho que representam as regiões que sofreram oclusão; também são exibidos os valores em absoluto do Coeficiente de Pearson  $|\rho|$ , que diminuem à medida que a oclusão aumenta (na maioria dos casos), por conta da quantidade crescente de falsos positivos, resultando em uma medida ruim de correlação entre os índices de vértices correspondentes. Observa-se um pequeno aumento do  $|\rho|$  entre as Fig. 86 e Fig. 87, contrariando a expectativa, porém trata-se apenas de uma oscilação momentânea, após essa etapa a correlação volta a diminuir com o aumento da oclusão.

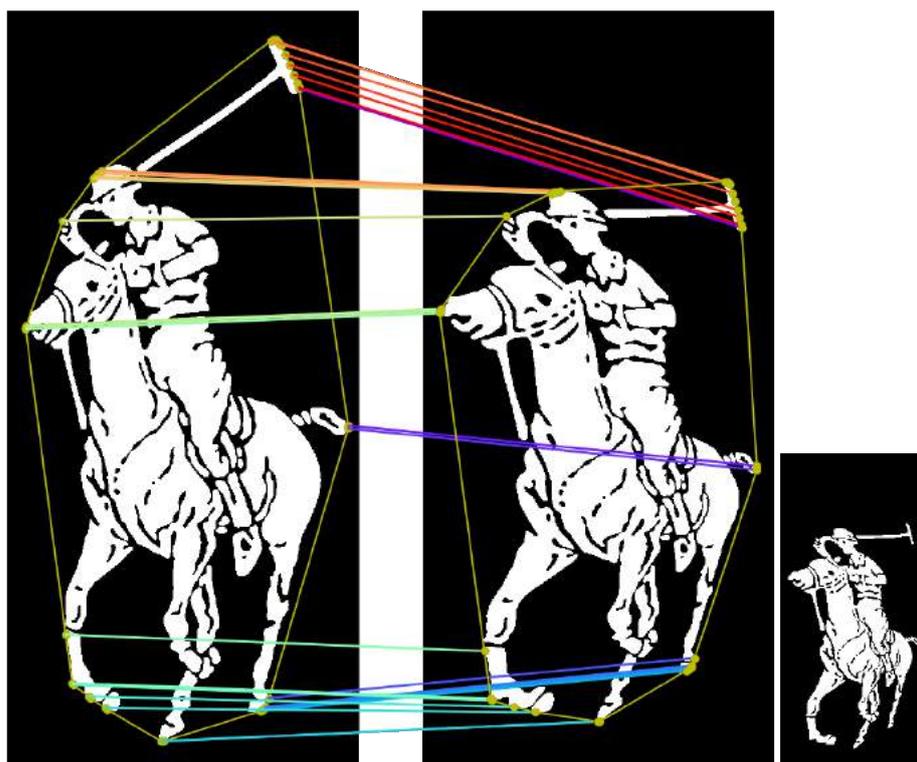


Figura 82 – Correspondências com objeto *query* (Dir.) deformado projetivamente e sem oclusão, com  $|\rho| \approx 0,9981$ .

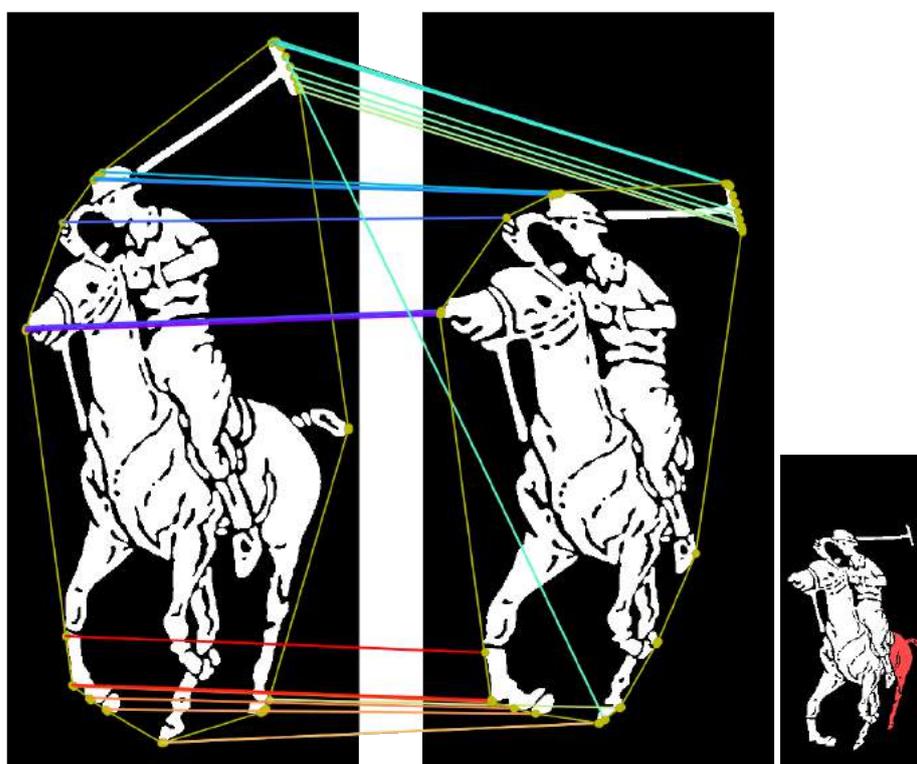


Figura 83 – Oclusão em torno de 9,13%. Há 24 associações, da qual apenas uma é incorreta, representando 4,17% do total de correspondências.  $|\rho| \approx 0,9684$ .

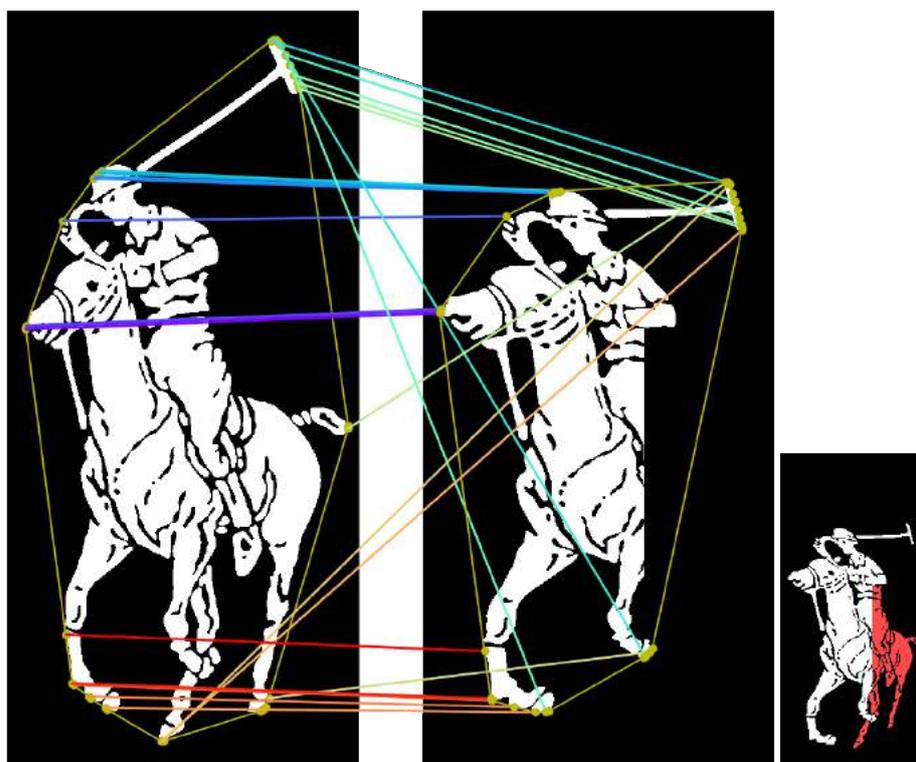


Figura 84 – Oclusão em torno de 23,31%. Existem 25 associações, das quais cinco são incorretas, representando 20% do total com  $|\rho| \approx 0,9419$ .

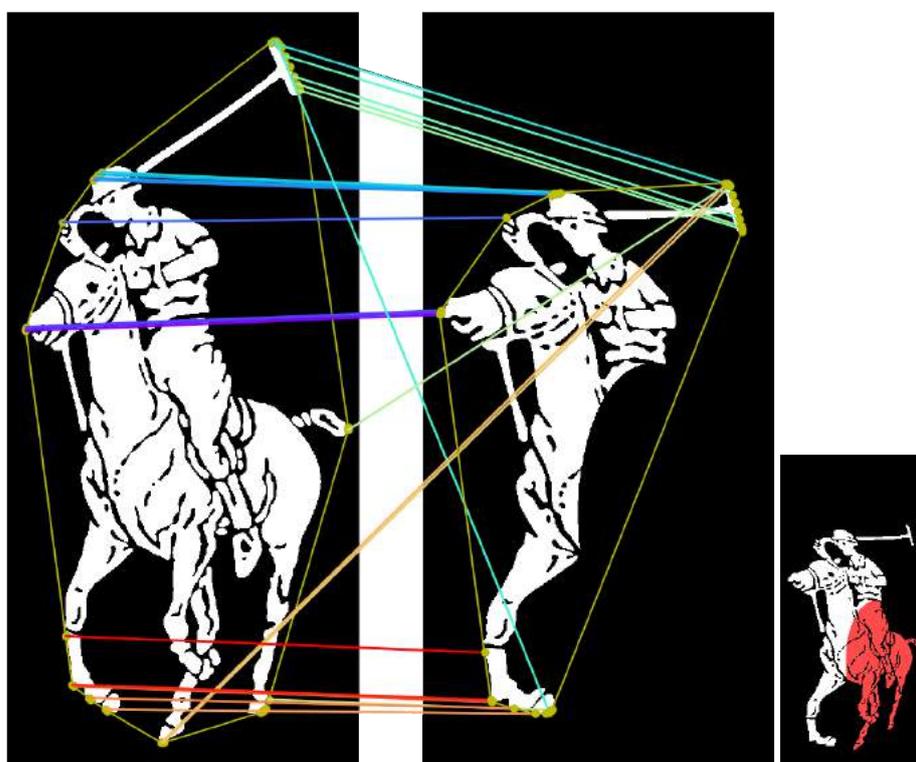


Figura 85 – Oclusão em torno de 40,92%. Existem 24 associações, onde quatro estão incorretas, representando 16% do total de correspondências com  $|\rho| \approx 0,9331$ .

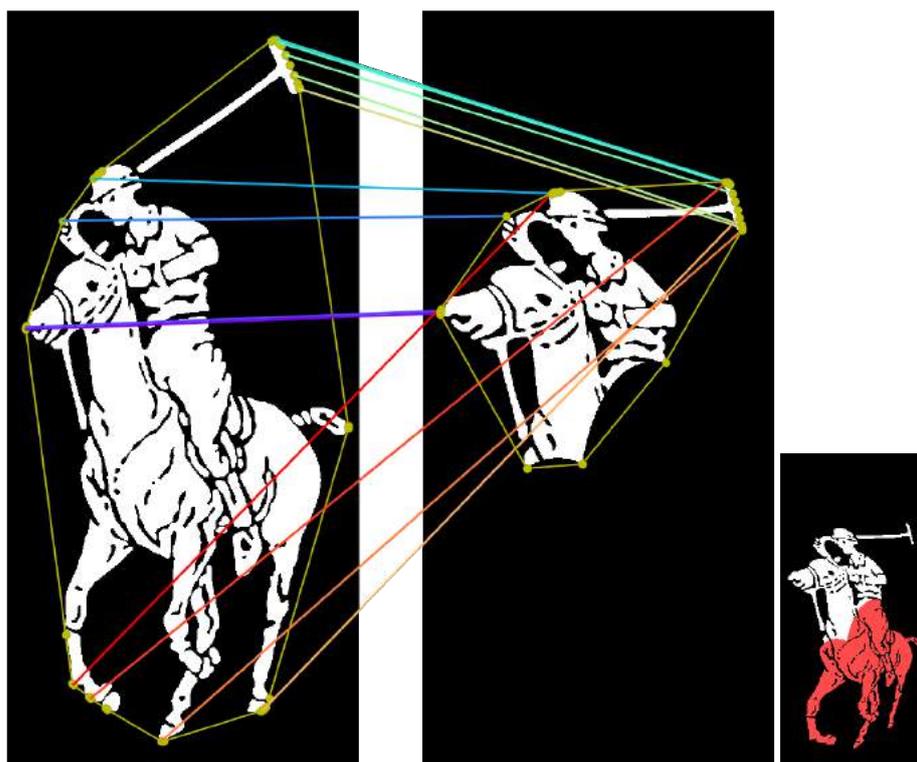


Figura 86 – Oclusão em torno de 56,42%. Há 13 associações, onde 4 delas estão incorretas, 30,77% do total com  $|\rho| \approx 0,8595$ .

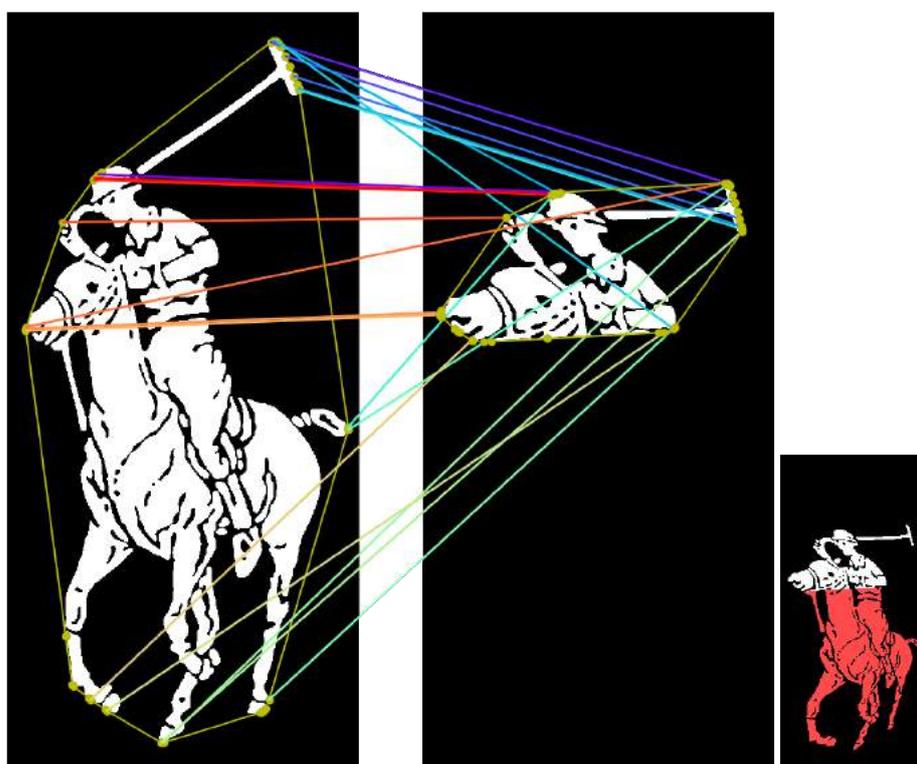


Figura 87 – Oclusão em torno de 73,29%. O número de associações é 21, dos quais dez estão incorretas, 47,62% do total com  $|\rho| \approx 0,9168$ .

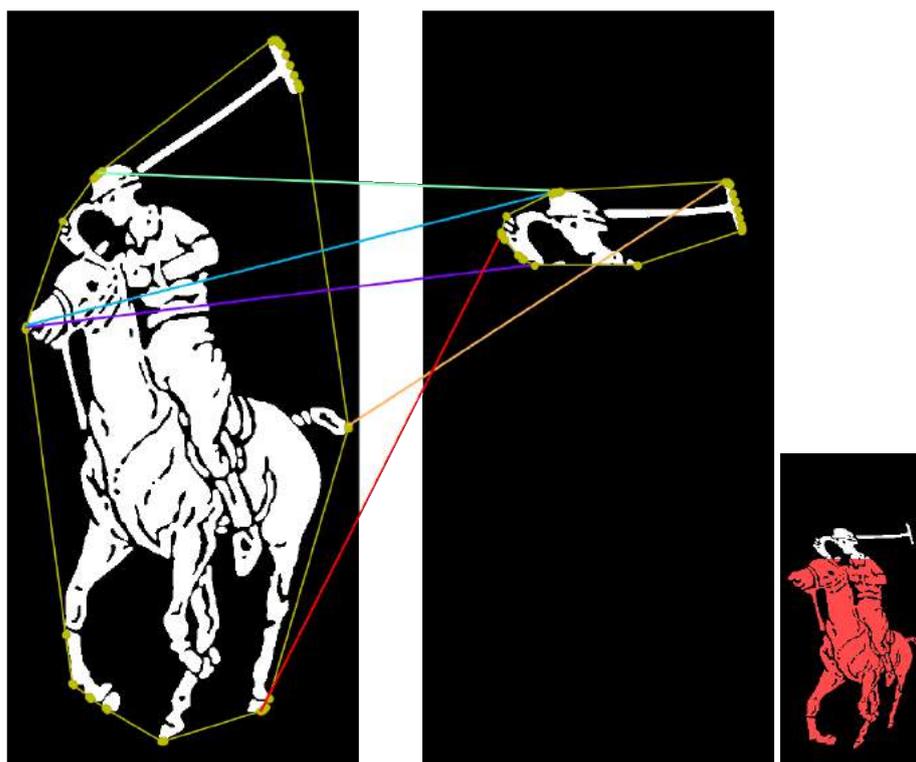


Figura 88 – Oclusão em torno de 91, 38%. Há 5 associações, das quais quatro estão incorretas, 80% do total com  $|\rho| \approx 0,6352$ .

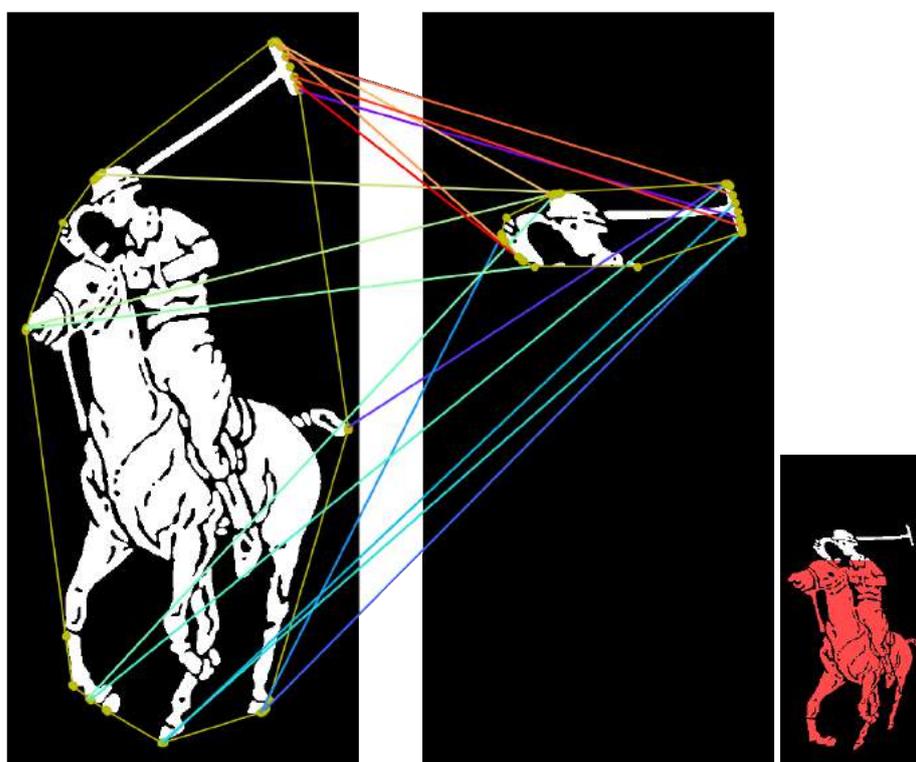


Figura 89 – Oclusão em torno de 91, 38%. Sem remoção de falsos-positivos. Há 16 associações, onde 13 estão incorretas, 81, 25% do total com  $|\rho| \approx 0,4529$ .

## 8.2 ANÁLISE DAS CORRESPONDÊNCIAS DOS SÍMBOLOS E LOGOMARCAS OBTIDOS DE CENAS REAIS

Para afirmar a viabilidade da técnica em casos reais, um conjunto de imagens contendo logomarcas e símbolos em diferentes pontos de vista é analisado quanto à qualidade das correspondências e os valores obtidos através do Coeficiente de Pearson, que apresenta valores altos em imagens de mesma classe com uma boa quantidade de associações Vértice-a-Vértices nas ordens corretas, por conta da correlação alta entre os índices vértices ajustados circularmente, como já estudado.

As imagens das cenas reais utilizadas receberam um pré-processamento e uma segmentação dos objetos de interesse. Na etapa de pré-processamento foi feita uma mudança na grande maioria dos valores de *pixels* para tons de cinza seguida de uma binarização na imagem baseado em um limiar de valores de intensidade, dando como resultado parcial objetos segmentados, e em seguida o objeto de interesse teve suas características extraídas e armazenadas no descritor e comparado com um descritor de um objeto modelo. E para caráter de exibição foi posto lado a lado o objeto modelo com a imagem original da cena real contendo um objeto planar de mesma classe. Abaixo alguns testes realizados com símbolos e logomarcas em cenas reais.

### 8.2.1 Teste I: Chuteira Adidas

Este teste foi realizado com a imagem de uma chuteira contendo a logomarca da Adidas rotacionada ( $2000 \times 1360$ ), após o processo de segmentação exibido na Fig. 90. O resultado da comparação com o *template* da logomarca Adidas ( $900 \times 900$ ) é exibido na Fig. 91, o resultado mostra que o descritor proposto é robusto a transformações geométricas simples, como a rotação. O valor em absoluto do Coeficiente de Pearson para este teste foi de  $|\rho| \approx 0,9874$ , indicando uma forte correlação entre os índices de vértices dos dois objetos.



Figura 90 – Processo de segmentação da logomarca da Adidas na chuteira.

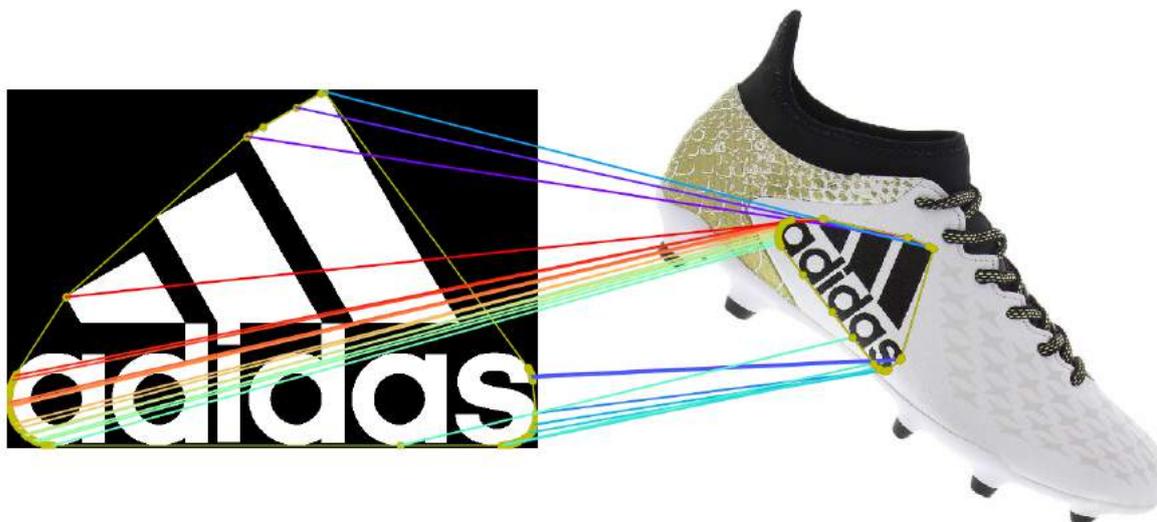


Figura 91 – O número de associações entre os vértices é 24, o objeto *template* contém 80 Vértices e o objeto *query* possui 32.

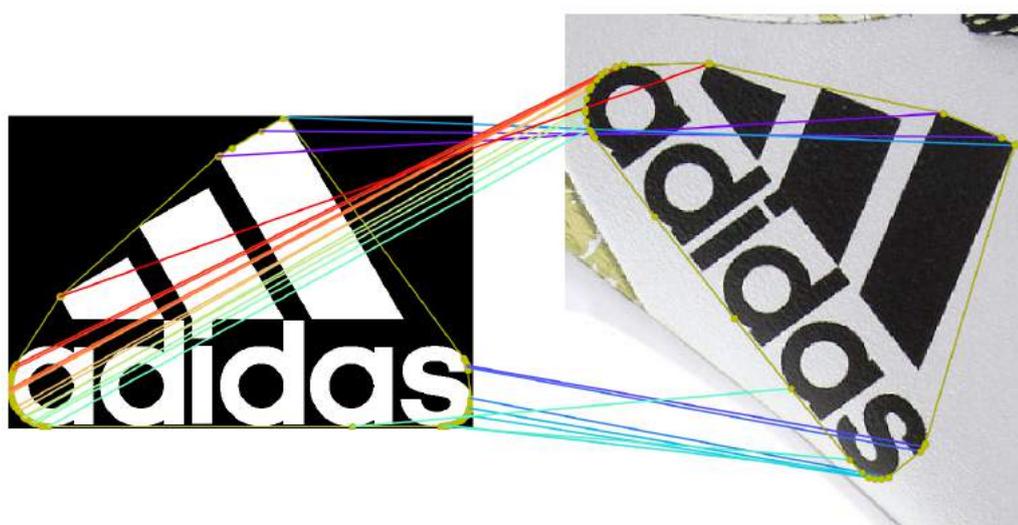


Figura 92 – O resultado da comparação com um zoom na imagem real. Pode-se observar os *matchings* entre os vértices dos *convex-hulls* em maiores detalhes.

### 8.2.2 Teste II: Loja Puma

Foi usada uma imagem do letreiro de uma loja da Puma sob efeito da perspectiva ( $254 \times 264$ ), após o processo de segmentação exibido na Fig. 93. O resultado da comparação com o *template* da logomarca Puma ( $600 \times 310$ ) é mostrada na Figura 94: as correspondências foram corretas e mostrou a robustez do descritor perante a deformações projetivas. O valor do módulo do Coeficiente de Pearson para este teste foi de  $|\rho| \approx 0,9472$ , indicando uma forte correlação entre os índices de Vértices dos objetos *template* e *query*.

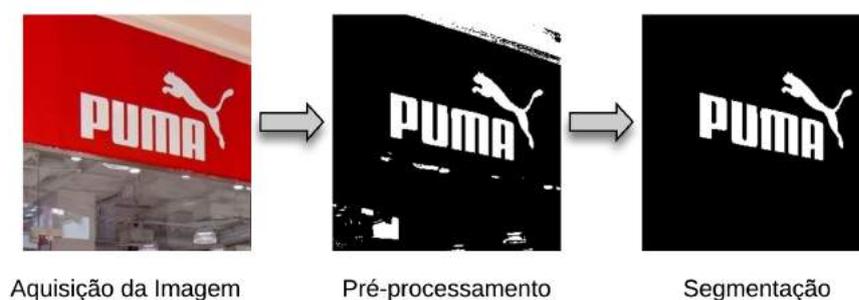


Figura 93 – Processo de segmentação e binarização da logomarca da Puma no letreiro da loja.



Figura 94 – Resultado da comparação: O número de associações entre os Vértices é sete, o objeto *template* contém dez vértices e o objeto *query* possui 15 vértices.

### 8.2.3 Teste III: Caixa de Tênis Nike

Para este teste foi usada uma imagem de uma caixa de tênis Nike sob efeito de uma perspectiva ( $1814 \times 1114$ ), após o processo de segmentação exibido na Fig. 95, notam-se oclusões na logomarca nas bordas laterais esquerda e direita. O resultado da comparação com o *template* da logomarca Nike ( $2068 \times 1076$ ) é mostrado na Fig. 96, as correspondências foram corretas e mostrou a robustez do descritor para deformações projetivas e com leves oclusões. O valor do módulo do Coeficiente de Pearson para este teste foi de  $|\rho| \approx 0,9910$ , indicando uma forte correlação entre os índices de vértices dos objetos *template* e *query*.



Figura 95 – Processo de segmentação/binarização da logomarca da Nike na tampa da caixa, observa-se oclusões na forma.

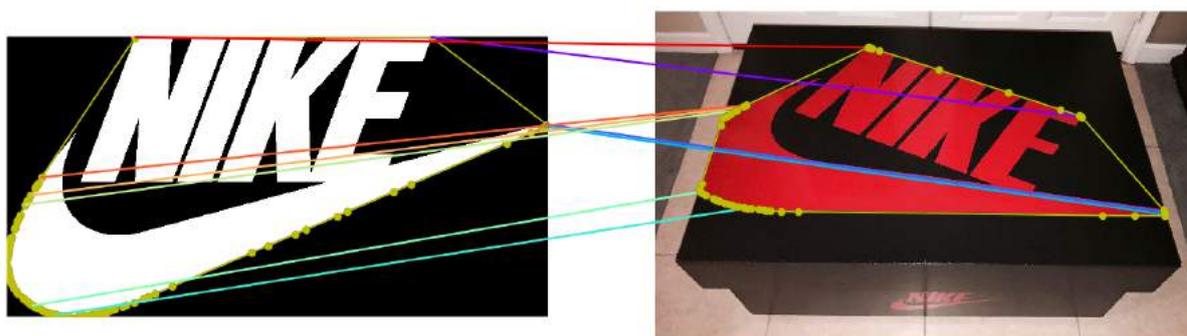


Figura 96 – O número de associações entre os vértices é nove, o objeto *template* contém 93 vértices e o objeto da cena real possui 48 vértices, o mesmo contém oclusões leves nas laterais.

#### 8.2.4 Teste IV: Símbolo de Acessibilidade

Foi realizado também um teste com o símbolo de acessibilidade para deficientes físicos fixada na parede em perspectiva ( $1024 \times 1024$ ). Na Figura 97 é ilustrado o processo de segmentação/binarização do símbolo de acessibilidade. O resultado, comparado com o objeto *template* ( $518 \times 518$ ), é exibido em 98, observa-se uma boa quantidade de *matchings* entre os vértices, e alguns poucos *matchings* falsos positivos. O valor em absoluto do Coeficiente de Pearson foi de  $|\rho| \approx 0,9595$ .



Figura 97 – Processo de segmentação do símbolo de acessibilidade.

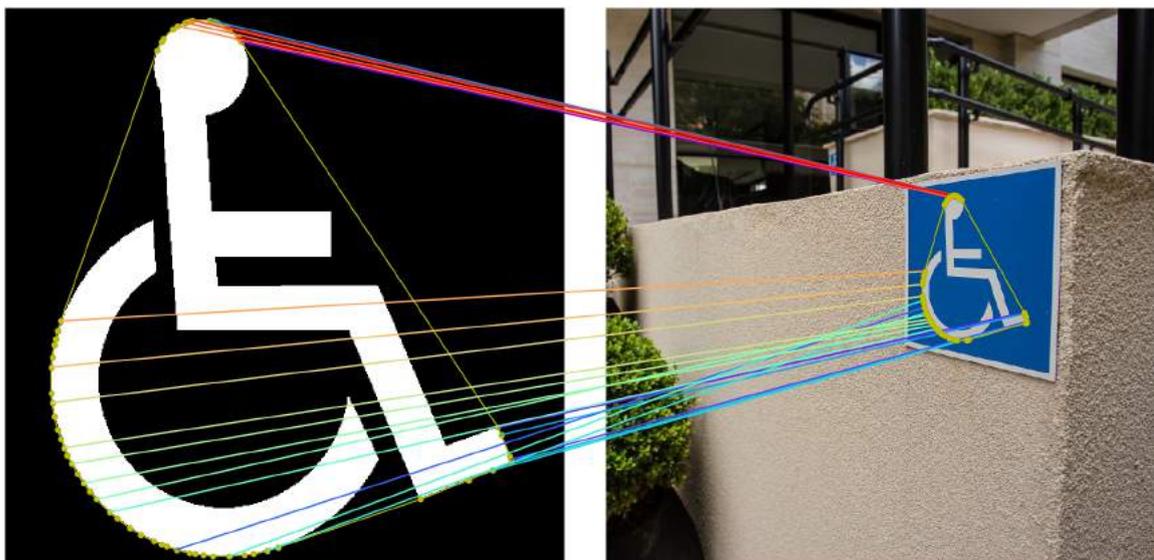


Figura 98 – Resultado da comparação: O número de associações entre os Vértices é 23, o objeto *template* contém 58 Vértices e o objeto da cena real possui 39 Vértices.

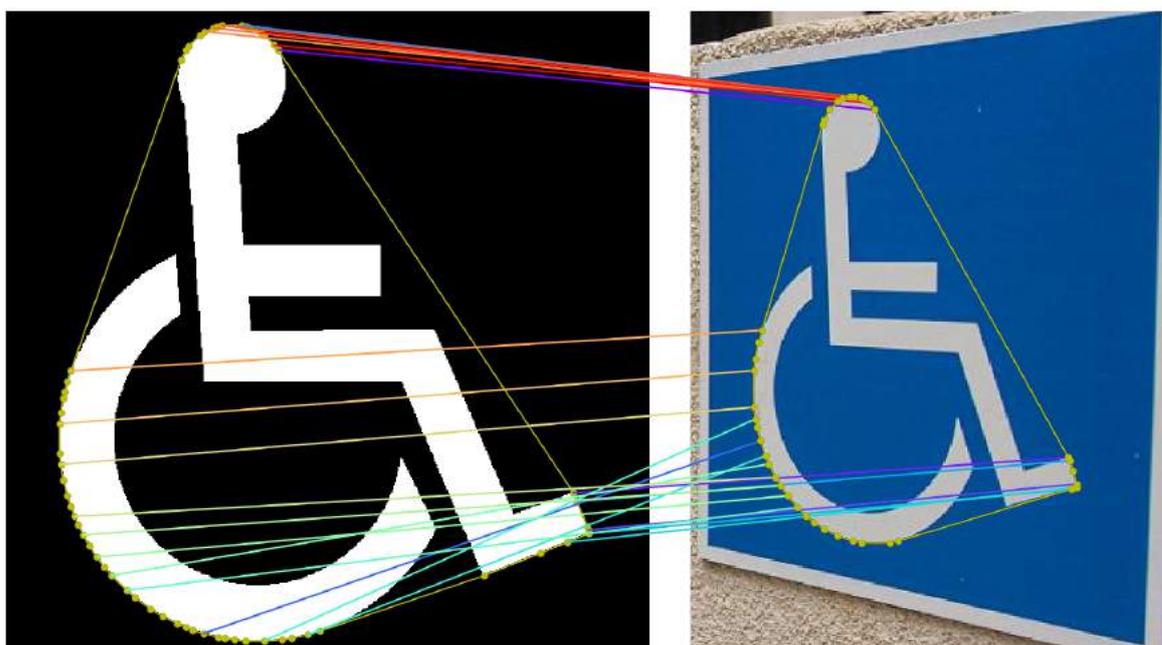


Figura 99 – Ampliação da imagem real para melhor visualização das correspondências dos vértices.

### 8.2.5 Teste V: Perfume Polo Ralph Lauren

Nesta subseção discutimos o teste realizado com a logomarca Polo Ralph Lauren gravada no vidro de um perfume e com leve efeito da perspectiva ( $2448 \times 3264$ ). Na Figura 100 é mostrado o resultado da segmentação/binarização da logomarca. Na Fig. 101 é exibido o resultado da comparação com o objeto *template* ( $600 \times 1300$ ). O valor em absoluto do Coeficiente de Pearson foi de  $|\rho| \approx 0,9948$ .



Figura 100 – Processo de segmentação da logomarca Polo Ralph Lauren em vidro de perfume.

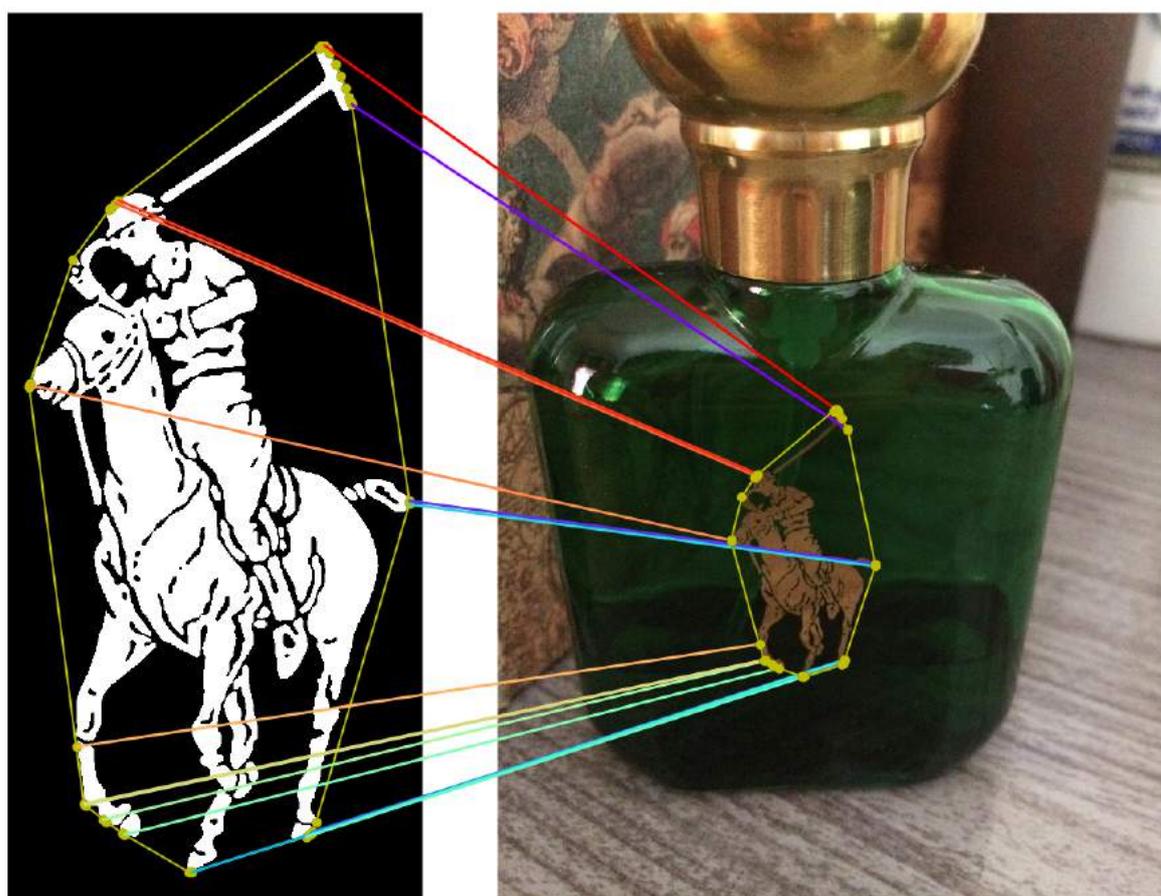


Figura 101 – O número de associações entre os Vértices é 17, o objeto *template* contém 37 Vértices e o objeto da cena real possui 31 Vértices.

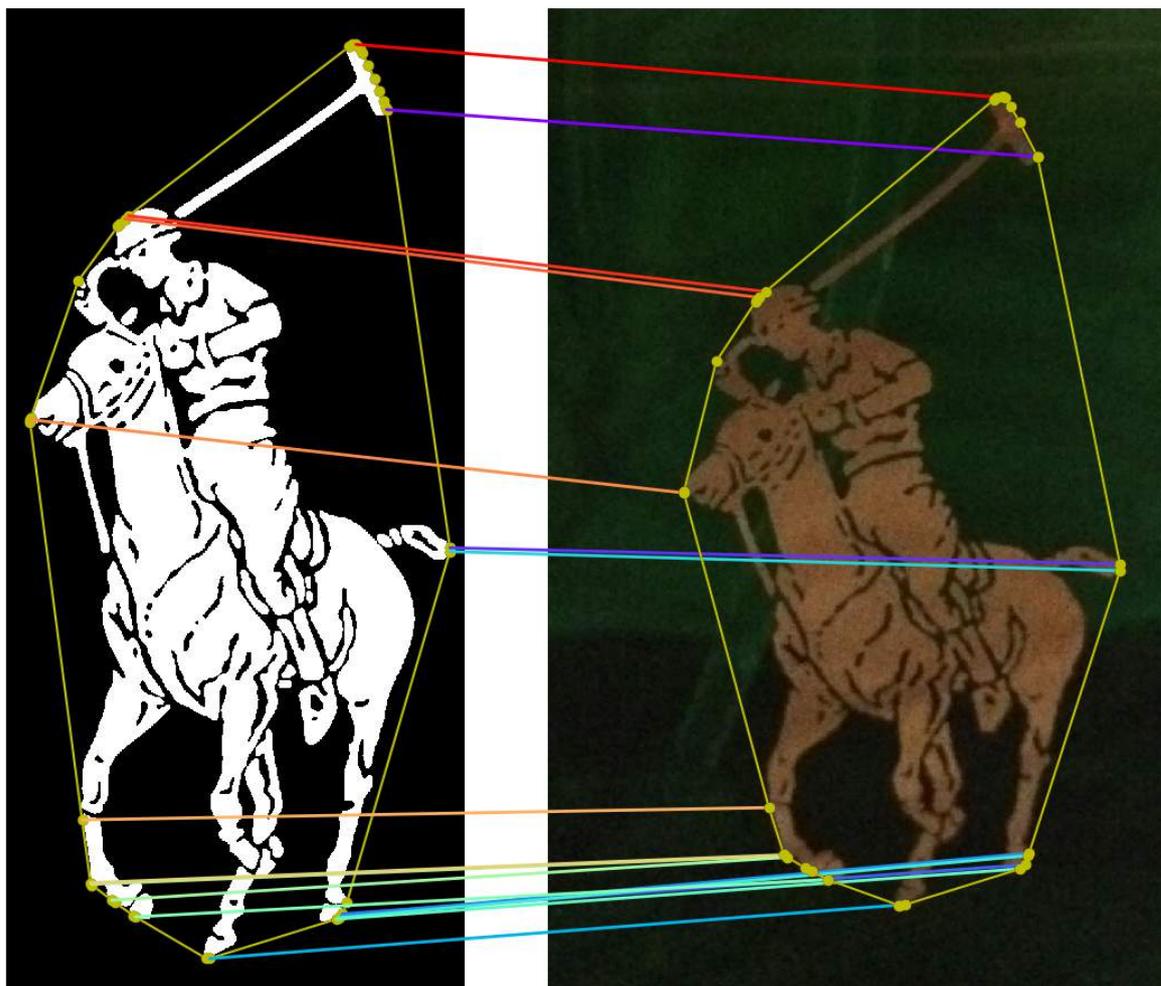
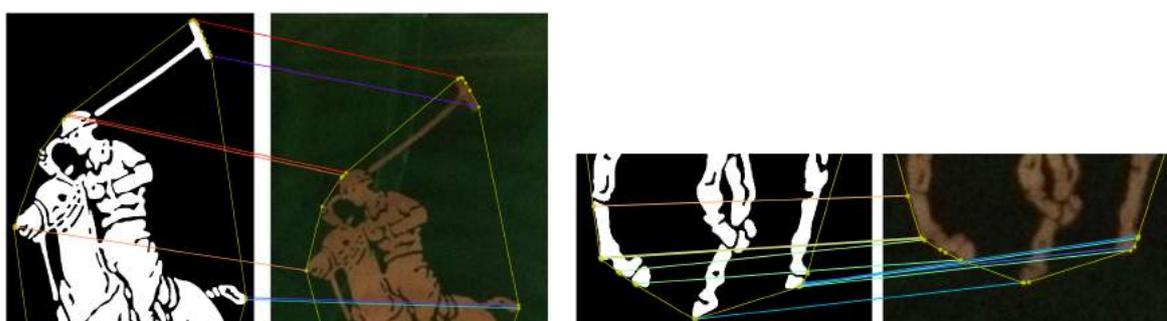


Figura 102 – Resultado da comparação com um zoom na imagem real. É possível notar uma boa qualidade nos *matchings* entre os Vértices.



(a) Correspondências na parte superior.

(b) Correspondências na parte inferior.

Figura 103 – Detalhes das correspondências em zoom.

### 8.2.6 Teste VI: Desodorante Adidas

Este teste foi realizado com a logomarca Adidas impressa na superfície cilíndrica do desodorante ( $1540 \times 2448$ ); é mostrado o processo para a segmentação/binarização da forma na Fig. 104. O resultado da comparação com o objeto *template* ( $2000 \times 1360$ ), é exibido na Figura 105: o objeto na cena real contém uma deformação projetiva não linear e mesmo assim o descritor foi capaz de fazer boas correspondências entre os Vértices. O valor em absoluto do Coef. de Pearson foi de  $|\rho| \approx 0,9877$ .



Figura 104 – Processo de segmentação da logomarca Adidas impressa no desodorante.



Figura 105 – O número de associações entre os vértices é 34, o objeto *template* contém 80 Vértices e o objeto da cena real possui 47 Vértices.

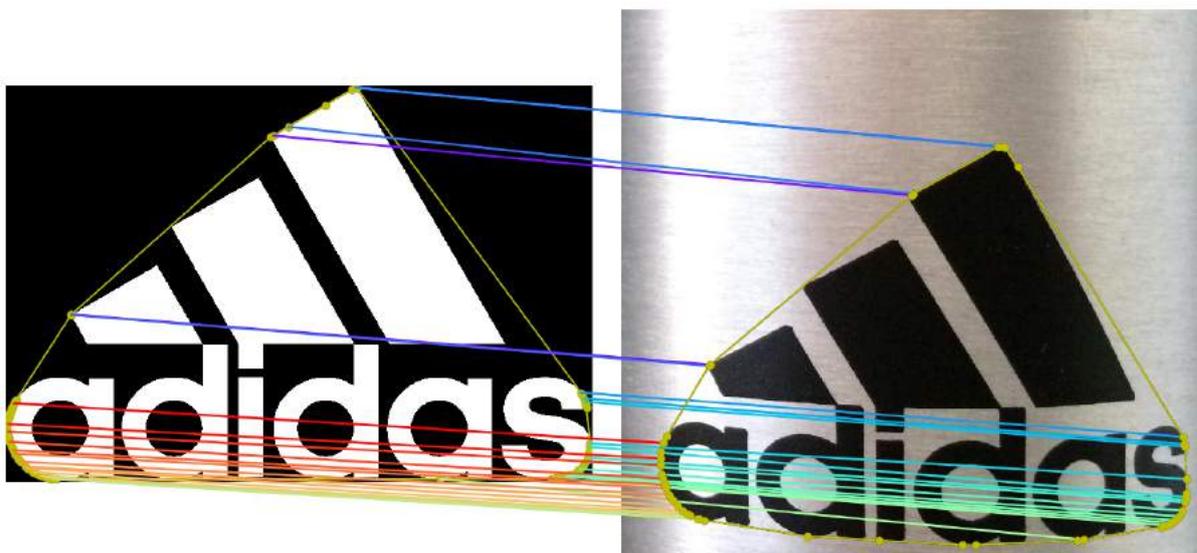
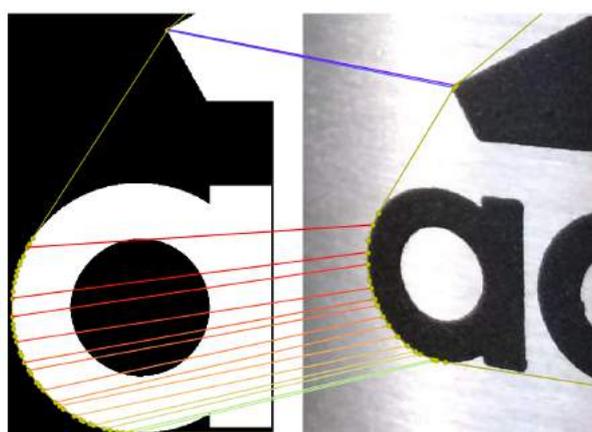


Figura 106 – Zoom na imagem real, o descritor proposto mostrou-se robusto a pequenas deformações não lineares, como é possível observar.



(a) Correspondências no primeiro carácter da logomarca.



(b) Correspondências nos últimos caracteres da logomarca.

Figura 107 – Detalhes das correspondências em zoom.

### 8.2.7 Teste VII: Camisa da Coca-Cola vestida num corpo feminino

Para este teste foi usada uma imagem ilustrando uma mulher vestindo uma camisa da Coca-Cola ( $1200 \times 1000$ ); é mostrado o processo para a segmentação/binarização da logomarca na Fig. 108. O resultado da comparação com o objeto *template* ( $800 \times 262$ ). Na Fig. 109 é exibido o objeto na cena real contendo uma deformação projetiva não linear, com uma pequena oclusão do lado esquerdo e mesmo assim o descritor foi robusto oferecendo boas correspondências entre os Vértices como resultado da comparação. O valor em absoluto do Coef. de Pearson foi de  $|\rho| \approx 0,9772$ .

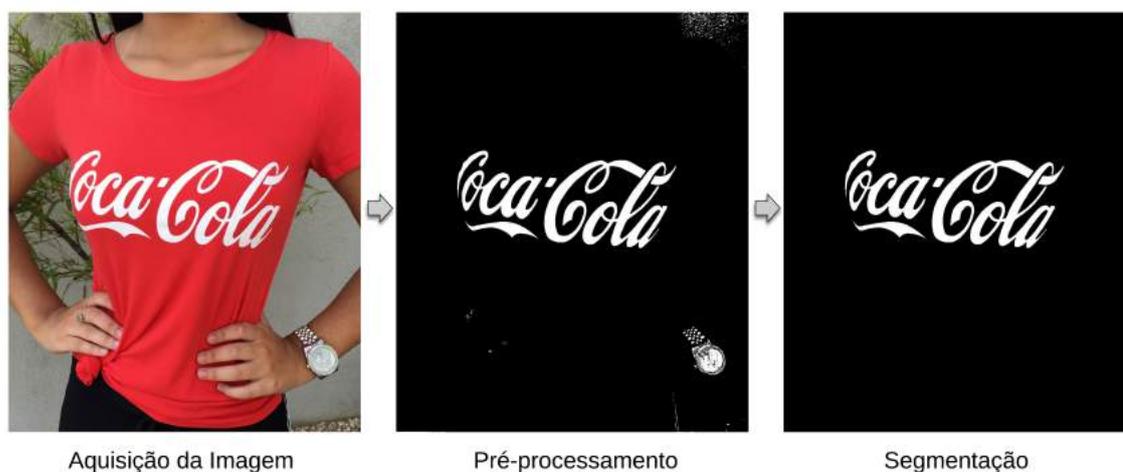


Figura 108 – Processo de segmentação da logomarca da Coca-Cola. É possível observar que a logomarca possui uma leve deformação não linear com uma pequena oclusão do lado esquerdo.



Figura 109 – Resultado da comparação: O número de associações entre os Vértices é 22, o objeto *template* contém 53 Vértices e o objeto da cena real possui 41 Vértices.



Figura 110 – Imagem ampliada para melhor visualização das correspondências, o descritor se mostrou robusto também a deformações dessa natureza.

### 8.2.8 Teste VIII: Garrafa com o brasão da UFPE

Neste teste foi usada uma garrafa contendo o brasão da Universidade Federal de Pernambuco, juntamente com a sigla UFPE (1340 × 2592), é mostrado o processo para a segmentação da logomarca na Fig. 108. O resultado da comparação com o objeto *template* (3428 × 5092), é exibido na Figura 109; o objeto na cena real contém uma deformação projetiva não linear por conta do formato cilíndrico da garrafa, e é possível notar que a sigla UFPE não foi retirada durante o processo de segmentação, mas mesmo assim as correspondências entre os vértices foram totalmente corretas, sem falsos positivos, demonstrando mais um tipo de robustez do descritor. O valor em absoluto do Coef. de Pearson foi de  $|\rho| \approx 0,9622$ .



Figura 111 – Processo de segmentação/binarização do brasão da UFPE, é importante ressaltar que nesse exemplo a sigla UFPE não foi descartada da segmentação com o intuito de testar a robustez do descritor proposto para comparações desse tipo.



Figura 112 – Como resultado, o número de associações entre os Vértices é 13, o objeto *template* contém 49 Vértices e o objeto da cena real possui 38 Vértices. Todas as associações entre Vértices estão corretas.

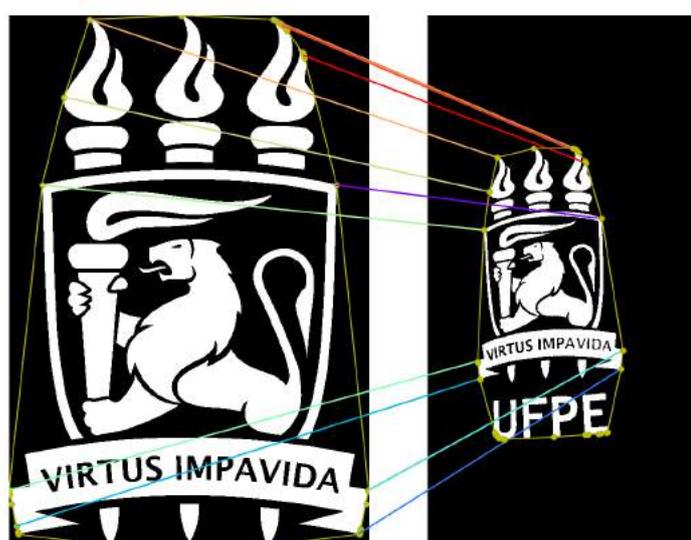
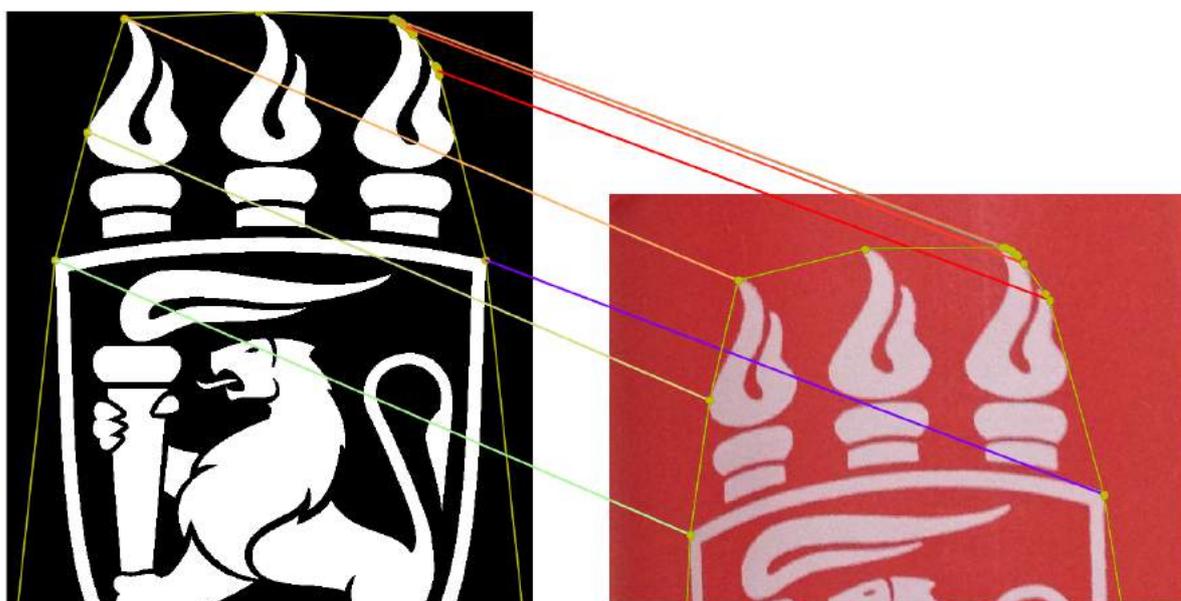


Figura 113 – Na imagem segmentada a partir da cena real (direita), o brasão e a sigla UFPE foram descritas em conjunto, como se fosse um mesmo objeto, apesar disso o descritor proposto foi capaz de associar corretamente os Vértices entre os objetos.



(a) Detalhe das Correspondências na parte superior



(b) Detalhe das Correspondências na parte inferior

Figura 114 – O resultado deste teste apresentou boas associações, sem falsos positivos.

### 8.3 SOBRE A IMPLEMENTAÇÃO

O descritor de forma planar proposto, juntamente com os algoritmos de extração de características e de comparação/classificação foram todos desenvolvidos em Python 3 com a biblioteca de visualização Matplotlib, outras bibliotecas também foram usadas, como: Numpy e Scipy. Para a estimação da homografia utilizada na métrica de comparação da terceira abordagem foi usado o método *findHomography* do OpenCV que faz uso do DLT para estimar a transformação projetiva sofrida entre os Vértices correspondentes. O hardware utilizado possui um processador Intel Core i7-7700HQ ( $4 \times 2.8$  GHz) e 16 GB de RAM. Não foi usada nenhuma técnica que explore o paralelismo nessa primeira versão do algoritmo, o sistema operacional usado foi o Microsoft Windows 10 Home Single Language.

### 8.4 ANÁLISE DO TEMPO DE EXECUÇÃO

Uma desvantagem da técnica proposta é o tempo de execução dos algoritmos, o objetivo principal deste trabalho é validar a técnica em termos de qualidade do descritor e na comparação entre os objetos para uma classificação simples pelo vizinho mais próximo; melhorias futuras em relação a performance são necessárias e discutidas mais a frente.

Na etapa de extração de características o tempo de execução é proporcional ao número de raios total que cruza o objeto, que depende do número de raios por *fan-beam* multiplicada pelo número de pontos inseridos e/ou Vértices presentes no invólucro do objeto, e depende também das dimensões do objeto.

As comparações entre objetos descritos são feitas aos pares, durante esse processo cada feixe de raios (*fan-beam*) de tamanho  $N$  que partem dos  $M$  Vértices do invólucro de um objeto é comparado com todos os outros feixes de Raios do outro objeto; então temos  $(N \times M)^2$  comparações de Raios no pior caso, porém a comparação completa (por vetores de razão cruzada) entre os Raios muitas vezes não é necessária, só haverá comparação completa entre Raios que possuem os mesmos números de pontos de borda. A comparação entre os Raios é feita Feixe-a-Feixe como explicada no Cap. 7, o tempo de comparação entre dois feixes de Raios de tamanho  $N$  é  $O(N^2)$  e após o término das comparações entre Raios, são gerados espectros de topologia com os valores dos números de pontos de borda dos Raios que obtiveram correspondências e em seguida esses espectros são comparados por DTW; o tempo de comparação entre duas séries via DTW de mesmo tamanho  $N$  é também  $O(N^2)$ .

A seguir, na Fig. 115, um exemplo de comparação entre formas de mesma classe: o símbolo de uma coroa extraída do MPEG-7 e uma versão da mesma deformada projetivamente com ângulos de  $el = 10^\circ$  e  $az = 90^\circ$ . Neste experimento foram usados 150 Vértices  $\times$  150 Raios por Vértice na extração de características, as imagens *template* e *query* das formas possuem dimensões  $178 \times 94$  e  $936 \times 1920$  respectivamente. O tempo de execução da etapa de extração de características do *template* foi em torno de 60 segundos e para o *query* foi de 280 segundos; e o tempo de comparação entre as formas foi de aproximadamente 147 segundos. Na tabela 17 são

n°Vértices × n° Raios por Vértice	150×150	100×100	50×50
tempo de varredura do <i>template</i>	60.59 s	24.55 s	5.58 s
tempo de varredura do <i>query</i>	280.05 s	125.73 s	25.62 s
tempo de comparação	146.93 s	25.36 s	1.21 s
<b>tempo total</b>	<b>487.57 s</b>	<b>175.64 s</b>	<b>32.41 s</b>

Tabela 17 – Tabela com os tempos de execução das etapas de extração de características (varredura do objeto) e comparação entre as formas da Fig.115 em diferentes amostragens (150 × 150, 100 × 100 e 50 × 50).

exibidos os tempos de varredura e comparação das mesmas formas da Fig. 115 em diferentes amostragens; é importante observar também que os tempos de varredura do *template* e *query* diferiram significativamente dentro de um mesmo valor de amostragem, o tempo de varredura do *query* foi em média aproximadamente cinco vezes o tempo de varredura do *template*, isso se deve ao fato do *query* possuir dimensões maiores do que a do *template* nesse exemplo.

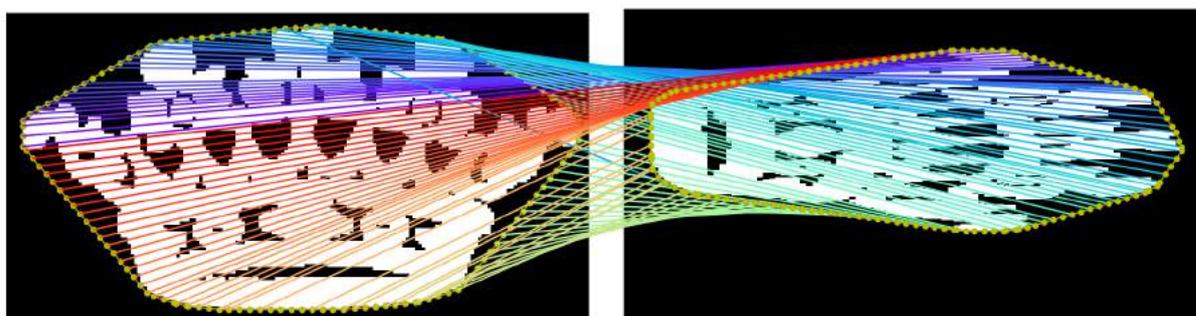


Figura 115 – Comparação entre formas de mesma classe: Um exemplo de *Crown* extraída do MPEG-7 (Esq.) e sua versão deformada projetivamente (Dir.), a amostragem usada neste exemplo foi de 150 vértices por 150 raios em ambas as formas.

## 8.5 DISCUSSÃO

Neste capítulo foram apresentados os experimentos realizados para atestar a robustez do descritor em diferentes situações. Na Seção 8.1, foram realizados testes com formas de consulta deformadas através de variações de ângulos de azimute e elevação, e para o reconhecimento foram usadas as métricas desenvolvidas no Cap. 7 nas três diferentes abordagens, onde foi constatado que a terceira abordagem, por estimação da homografia, mostrou resultados melhores de acurácia em testes realizados perante as outras abordagens; foram realizados experimentos com objetos planares variados, divididos entre caracteres, logomarcas (juntamente com o símbolo de acessibilidade) e formas extraídas do MPEG-7 *dataset*, onde foi gerado versões deformadas dos mesmos seguindo a metodologia do trabalho de (LI; TAN, 2010) com variações de ângulos de elevação e azimute e os resultados foram comparados com uma técnica concorrente atual, o HCNC de (JIA et al., 2016).

Nos experimentos com caracteres, os resultados não foram tão bons quanto o HCNC, mas uma análise feita nos experimentos indicaram a natureza do erro que pode ser corrigido adequadamente, alguns símbolos foram confundidos por possuírem semelhança total ou parcial, para exemplificar: o símbolo ‘O’ deformado foi confundido com o símbolo ‘9’, isso se deve ao fato dos pontos de contorno do primeiro símbolo casarem com a parte arredondada do dígito, para esse problema deve-se usar uma estratégia que use todos os pontos após uma estimação de homografia, a maior diferença de acurácia foi de 14,7%. Nos testes realizados com logomarcas o descritor proposto obteve resultados iguais acurácia iguais ou superiores ao HCNC, onde a diferença era mais notável enquanto o grau de elevação crescia, o descritor proposto obteve bons resultados até em deformações mais severas, fornecendo um valor de acurácia de 90% no cenário com deformação mais acentuada ( $el = 10^\circ$ ), enquanto o HCNC ofereceu um valor de acurácia de apenas 23% no mesmo cenário, uma diferença de 67%. Também foram realizados testes com doze formas originais extraídas do MPEG-7 *dataset* (que foi utilizado no trabalho de (JIA et al., 2016)) confrontadas com 36 formas derivadas dessas formas originais, onde o descritor proposto teve um resultado inferior ao HCNC nos testes realizados, porém a diferença de taxa de acerto não foi tão grande, no pior cenário a diferença foi de 13,9%, analisando os erros, percebeu-se que o descritor proposto errou nos casos onde os objetos possuem formas com pouca complexidade, como: *apple*, *bat*, *bone* e *hammer*, e em alguns cenários o *dog* também foi confundido, o descritor proposto teve 100% de acerto em relação aos objetos com maiores complexidades, que possuem estruturas internas mais detalhadas e/ou reentrâncias, como o *beetle*, *butterfly*, *crown*, *deer*, *elephant*, *lizzard* e *pocket*; o HCNC teve alguns insucessos com o *pocket*, por ser uma forma que possui estruturas internas mais ricas. Ainda nesta mesma seção foram realizados pequenos testes com oclusão num objeto de cenas deformado em diferentes percentuais de obstrução do objeto, demonstrando que o descritor proposto é robusto também a oclusões além de deformações projetivas, e isso é bastante animador em se tratando de um descritor que necessita de informações globais da forma para representar uma pequena parte do objeto, mostrando resistência também nessas condições.

Na Seção 8.2, foram feitos testes com o intuito de analisar as correspondências entre os Vértices dos objetos em cenas reais. Foram realizados desde testes simples com logomarcas rotacionadas, em perspectiva e até com **deformações não lineares** em superfícies cilíndricas, como pode ser visto nos testes com a logomarca da Adidas num desodorante e o brasão da UFPE impressa na lateral da garrafa, além da logomarca da Coca-Cola numa camisa vestida num corpo feminino com leves oclusões na forma. Esse pequeno conjunto de testes mostrou que o descritor pode ser usado também para o reconhecimento de objetos com deformações não lineares leves. Foi realizado uma pequena análise dos tempos de execução das etapas de descrição de características e comparação entre as formas com diferentes valores de amostragem na Seção 8.4; e como esperado os tempos de execução de ambas as etapas aumentam de acordo com o aumento da amostragem e o tempo da etapa de varredura do objeto (extração de características) aumenta também de acordo com as dimensões do objeto.

Neste capítulo foi possível comprovar a robustez do descritor quanto a deformações projetivas, que foi a proposta principal deste trabalho, sendo comparada com uma técnica pertencente ao estado-da-arte, tendo resultados inferiores em cenários que existiam em sua maioria objetos com pouca complexidade (caracteres e algumas formas do MPEG-7) e superior em cenários com objetos mais complexos (logomarcas e algumas formas do MPEG-7). Além disso, foi possível obter bons resultados em outras situações adversas, como oclusão e deformações não lineares. A métrica de similaridade foi repensada, e a abordagem final foi modificada não diferenciando mais entre formas simétricas e assimétricas, a distância final entre os objetos é computada através de uma estimação de homografia baseada no erro de transferência.

---

## 9 CONCLUSÃO

---

Este trabalho apresenta um novo descritor de forma planar que dá suporte ao reconhecimento de símbolos sob o efeito de diversas transformações geométricas, desde transformações mais simples até transformações mais gerais, como as projetividades.

Foi mostrado que o descritor proposto possui uma estrutura hierárquica de *features* (características do objeto) que se baseia em informações mais periféricas até informações mais internas da forma, começando com os Vértices pertencentes a um invólucro convexo ou de contorno que envolve o objeto, e a partir de cada Vértice são emitidos vários Raios que cruzam o objeto planar gerando pontos colineares através da intersecção desses Raios com a forma. Foi visto que com quatro pontos colineares podemos calcular o invariante projetivo conhecido como razão cruzada, que é o invariante projetivo mais fundamental, mas nas intersecções dos Raios com a forma conseguimos obter mais do que quatro pontos colineares, e sendo assim, podemos obter mais de um valor de razão cruzada por Raio emitido, onde a trajetória que cruza o objeto pode ser identificada a partir de um **Vetor de Razão Cruzada**, que é um invariante projetivo desenvolvido neste trabalho; todas essas *features* compõem esse novo descritor de maneira estruturada e com valores de amostragem configuráveis.

Além da etapa de Extração de Características do descritor, foi estudada a forma como os elementos do descritor são comparados, começando pela comparação Raio-a-Raio, que faz comparações entre os Vetores de Razão Cruzada indo até comparações entre os espectros de topologia via DTW gerados pelos eventuais casamentos de Raios. Após todas essas etapas de comparação, vimos que para objetos não simétricos podemos calcular um valor de correlação entre os índices do Vértices correspondentes através do módulo do coeficiente de Pearson, além dos ajustes e remoção de falsos-positivos realizados através de MMQ e razão cruzada de cinco pontos coplanares; em se tratando de formas simétricas, como alguns caracteres, não é possível extrair esse tipo de informação, forçando a divisão em duas abordagens, para formas simétricas e assimétricas como foi explicado na Seção 7.4. As únicas diferenças dessas abordagens é que na abordagem assimétrica existe uma etapa intermediária de remoção de falsos positivos e cálculo de correlação, mas as etapas de comparação elemento-a-elemento e cálculo da distância final continuam semelhantes.

Durante os experimentos realizados em larga escala, houve a oportunidade de se testar uma métrica de similaridade mais confiável, e foi desenvolvida uma nova métrica baseada no erro de transferência explicada no capítulo anterior na Subseção 7.6.1, sendo assim a abordagem da Classificação muda em algumas etapas: a etapa de comparação Elemento-a-Elemento continua a mesma, mas a etapa de remoção de falsos positivos e cálculo do Coeficiente de Pearson (usa-

dos somente na comparação para formas assimétricas), e a última etapa de cálculo de distância (comum a ambas) foram eliminadas e substituídas pela a estimativa de homografia e cálculo de distância final baseado no erro de transferência, como discutido em 7.6. Dessa forma foi possível obter valores melhores de acurácia durante os experimentos e tornando as etapas de Classificação muito mais simples e eficientes.

Após todo esse desenvolvimento e estudo desse novo descritor, o mesmo foi submetido a testes variados, em cenas reais, com deformações projetivas e oclusões simuladas, com diferentes símbolos e logomarcas, além de testes comparativos com outro descritor de forma invariante projetivo pertencente ao estado-da-arte. Os resultados são promissores, o descritor extrapolou as expectativas, mostrando resistências em alguns graus de deformação e oclusão que não eram inicialmente previstos, apresentando boas correspondências entre Vértices até em condições de deformações mais severas. Nos testes comparativos com HCNC, em relação à métrica de similaridade usada nos testes para o reconhecimento, foram testadas as métricas propostas exibidas no Cap. 7 e além dessas houve uma terceira métrica desenvolvida durante os experimentos que se mostrou mais eficiente para o reconhecimento, e que não há a necessidade de distinção entre forma simétrica ou assimétrica.

Foi observado que o descritor proposto se saiu melhor em relação ao HCNC quando comparado com formas mais detalhadas, ricas em estruturas internas, como no caso das logomarcas e algumas formas do MPEG-7 que possuem detalhes internos e reentrâncias, isso se deve ao fato de que a *feature* desenvolvida neste trabalho, o vetor de razão cruzada, pode carregar mais informação do objeto durante a extração de características e ter uma maior chance de acerto de casamento entre raios durante a etapa de comparação e classificação para formas mais complexas. Sendo assim, o descritor proposto pode ainda ser melhorado para os casos de reconhecimento entre formas mais simples, pobres em detalhes.

## 9.1 TRABALHOS FUTUROS

Alguns pontos do descritor proposto podem ser explorados e melhorados. Ainda há espaço para a melhoria da métrica para objetos mais simples, como caracteres, como discutido recentemente, melhorando algumas etapas finais do processo é possível obter valores melhores de acurácia.

Os testes com o descritor proposto ofereceram bons casamentos entre os pares de Vértices, que são pontos que ficam nas partes mais externas do objeto, além desses, é possível também obtermos casamentos com os pontos de borda dos Raios, oferecendo ainda mais associações, ajudando (talvez) a estimar a homografia de maneira mais precisa e dessa forma podendo melhorar a métrica para situações em que o descritor proposto ainda não se sai tão bem, nos casos contendo objetos mais simples.

A métrica baseada no erro de transferência obteve bons resultados durante os testes de reconhecimento com símbolos e logos, nos testes com caracteres foi possível verificar o motivo do erro na classificação para esses casos. Dessa forma, como um trabalho futuro, iremos me-

lhorar o *template matching* após a retificação dos pontos, para isso iremos explorar diferentes metodologias que podem ajudar nessa fase final do reconhecimento, tais como a distância de Hausdorff, que mede a distância entre dois conjuntos de pontos, que no nosso contexto podem ser os pontos de contorno ou do invólucro convexo dos objetos comparados, com os pontos do objeto de consulta já retificados; outras estratégias de comparação entre dois conjuntos de pontos podem ser também estudadas e testadas.

Estratégias para o aumento de performance são necessárias, é possível obter melhorias dos tempos nas comparações entre os espectros via DTW, substituindo-a por alguma versão mais rápida da mesma ou alguma outra técnica de comparação entre séries temporais que seja mais rápida. Uma versão futura do método proposto em tempo real pode ser desenvolvida com uso de técnicas de processamento em paralelo com uso de GPU (*Graphics Processing Unit*) dado que a geração e comparação dos elementos componentes do descritor são independentes; como na extração de características: onde o objeto é varrido por um grande número de Raios e vetores de razão cruzada são gerados independentemente de outros vetores; e também na comparação Raio-a-Raio, na etapa de comparação entre os descritores: onde os vetores são comparados aos pares também de maneira independente e paralelizável.

Em condições mais adversas, contendo oclusões e deformações não lineares, o descritor já apresentou um bom nível de robustez, porém é necessário um estudo mais profundo e um número maior de testes para o entendimento da robustez do descritor nessas situações, já que não existe na literatura uma forma padronizada de se testar robustez a deformações não lineares, e nem a robustez a oclusões.

Uma outra forma que é teoricamente prevista para a estimação da transformação projetiva sofrida pelos objetos, é através da obtenção da linha do horizonte, que é a imagem da reta no infinito após sofrer uma deformação projetiva, e que pode ter seus pontos de fuga encontrados pela relação da razão simples de três pontos colineares com a razão cruzada de quatro pontos colineares; no descritor proposto, existem Raios na qual residem esses pontos colineares e isso pode ser usado para a obtenção da linha do horizonte.

---

## REFERÊNCIAS

---

- AOKI, K.; OHYAMA, W.; WAKABAYASHI, T. Automatic extraction and recognition of shoe logos with a wide variety of appearance. In: IEEE. *Machine Vision Applications (MVA), 2017 Fifteenth IAPR International Conference on*. [S.l.], 2017. p. 211–214. 24
- ARICA, N. *SHAPE : Representation, Description, Similarity and Recognition*. Tese (Doutorado) — Middle East Technical University (METU), School of Natural and Applied Sciences, Department of Computer Engineering; Ankara, Turkey, 9 2003. 19, 20
- AUFFINGER, A. C. T. de C.; VALENTIM, F. J. da S. *Introdução à Geometria Projetiva*. Universidade Federal do Espírito Santo, Departamento de Matemática; Vitória, ES, 2003. 27
- BALLARD, D. Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition*, v. 13, n. 2, p. 111 – 122, 1981. ISSN 0031-3203. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0031320381900091>>. 47
- BAMBHORI, J.; JALGAON, I. Recognition of traffic symbols using k-means and shape analysis. *International Journal of Engineering*, v. 2, n. 5, 2013. 22
- BARBER, C. B.; DOBKIN, D. P.; HUHDANPAA, H. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, v. 22, n. 4, p. 469–483, 1996. 58
- BARBER, C. B.; HUHDANPAA, H. Qhull. 1995. Acessado em: 23/02/2018. Disponível em: <<http://www.qhull.org>>. 58
- BAY, H. et al. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, v. 110, n. 3, p. 346 – 359, 2008. ISSN 1077-3142. Similarity Matching in Computer Vision and Multimedia. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1077314207001555>>. 47
- BELONGIE, S.; MALIK, J.; PUZICHA, J. Shape matching and object recognition using shape contexts. In: . [S.l.: s.n.], 2002. v. 24, n. 4, p. 509–522. ISSN 0162-8828. 52
- CASEY, R. G.; LECOLINET, E. A survey of methods and strategies in character segmentation. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 18, n. 7, p. 690–706, 1996. 22
- CASSISI, C. et al. Similarity measures and dimensionality reduction techniques for time series data mining. In: KARAHOCA, A. (Ed.). *Advances in Data Mining Knowledge Discovery and Applications*. Rijeka: InTech, 2012. cap. 03. Disponível em: <<http://dx.doi.org/10.5772/49941>>. 39, 41, 46
- CHAVRE, P. B.; GHOTKAR, A. A survey on text localization method in natural scene image. *International Journal of Computer Applications*, Foundation of Computer Science, v. 112, n. 13, 2015. 22
- CHEN, R.; LIAO, S. Mobile chinese translator using maximally stable extremal regions ( mser ). In: . [S.l.: s.n.], 2012. 22

- CHEN, X.; YUILLE, A. L. Detecting and reading text in natural scenes. In: IEEE. *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. [S.l.], 2004. v. 2, p. II–II. 22
- DIXIT, U. D.; SHIRDHONKAR, M. Automatic logo detection and extraction using singular value decomposition. In: IEEE. *Communication and Signal Processing (ICCSP), 2016 International Conference on*. [S.l.], 2016. p. 0787–0790. 24
- DUBROFSKY, E. Homography estimation. *Diplomová práce. Vancouver: Univerzita Britské Kolumbie*, 2009. 109
- ERPEN, L. R. C. *Reconhecimento de Padrões em Imagens por Descritores de Forma*. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, 6 2004. 20, 21, 22
- EVANGELISTA, R. dos S. *Detecção e Descrição de Características Visuais Invariantes à Transformações de Similaridade, Afins e Projetivas*. Dissertação (Mestrado) — Universidade Federal Fluminense, Niterói, RJ, 4 2016. 29, 48, 50
- GANIVET, A. Epistolario. *Cap. xiii*, 1893. 5
- GOLDFARB, L. A unified approach to pattern recognition. *Pattern Recognition*, v. 17, p. 575–582, 1984. 19
- GONZALEZ, R. C.; WOODS, R. E. *Digital Image Processing (3rd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2006. ISBN 013168728X. 20, 21
- GREENHALGH, J.; MIRMEHDI, M. Recognizing text-based traffic signs. *IEEE Trans. Intelligent Transportation Systems*, v. 16, n. 3, p. 1360–1369, 2015. 22
- HAN, J.; PEI, J.; KAMBER, M. *Data mining: Concepts and Techniques (3rd Edition)*. [S.l.]: Elsevier, 2012. ISBN 9780123814791. 39
- HARTLEY, R.; ZISSERMAN, A. *Multiple View Geometry in Computer Vision (2nd Edition)*. [S.l.]: Cambridge University Press, 2004. ISBN 9780521540513. 28, 31, 32, 35, 109
- JIA et al. Hierarchical projective invariant contexts for shape recognition. *Pattern Recognition*, v. 52, p. 358 – 374, 2016. ISSN 0031-3203. 14, 49, 50, 52, 121, 146, 147
- JIA, Q. et al. A new geometric descriptor for symbols with affine deformations. *Pattern Recognition Letters*, v. 40, p. 128 – 135, 2014. ISSN 0167-8655. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0167865513004261>>. 47, 48, 71
- KAZMI, I. K.; YOU, L.; ZHANG, J. J. A survey of 2d and 3d shape descriptors. In: *2013 10th International Conference Computer Graphics, Imaging and Visualization*. [S.l.: s.n.], 2013. p. 1–10. 47
- KE, Y.; SUKTHANKAR, R. Pca-sift: A more distinctive representation for local image descriptors. In: *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 2004. (CVPR'04), p. 506–513. Disponível em: <<http://dl.acm.org/citation.cfm?id=1896300.1896374>>. 47

- KUANG, X.; FU, W.; YANG, L. Real-time detection and recognition of road traffic signs using mser and random forests. *International Journal of Online Engineering (iJOE)*, v. 14, n. 03, p. 34–51, 2018. 23
- LI, L.; TAN, C. L. Recognizing planar symbols with severe perspective deformation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 32, n. 4, p. 755–762, April 2010. ISSN 0162-8828. 14, 48, 50, 51, 52, 71, 114, 127, 146
- LIMA, F. S. de; GUEDES, L. A. H.; SILVA, D. R. Application of fourier descriptors and pearson correlation for fault detection in sucker rod pumping system. In: *2009 IEEE Conference on Emerging Technologies Factory Automation*. [S.l.: s.n.], 2009. p. 1–4. ISSN 1946-0740. 41
- LIU, C. et al. Fast traffic sign recognition via high-contrast region extraction and extended sparse representation. *IEEE transactions on Intelligent transportation systems*, IEEE, v. 17, n. 1, p. 79–92, 2016. 22
- LONCARIC, S. A survey of shape analysis techniques. *Pattern Recognition*, v. 31, p. 983–1001, 1998. 47
- LOWE, D. G. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, v. 60, n. 2, p. 91–110, Nov 2004. ISSN 1573-1405. Disponível em: <<https://doi.org/10.1023/B:VISI.0000029664.99615.94>>. 47
- LUO, Z. et al. A shape descriptor based on new projective invariants. In: *2013 IEEE International Conference on Image Processing*. [S.l.: s.n.], 2013. p. 2862–2866. ISSN 1522-4880. 14, 49, 50, 52
- MALDONADO-BASCÓN, S. et al. Road-sign detection and recognition based on support vector machines. *IEEE transactions on intelligent transportation systems*, IEEE, v. 8, n. 2, p. 264–278, 2007. 22, 23
- MARSICO, M. D. et al. Moving face spoofing detection via 3d projective invariants. In: *2012 5th IAPR International Conference on Biometrics (ICB)*. [S.l.: s.n.], 2012. p. 73–78. ISSN 2376-4201. 37
- MATHIAS, M. et al. Traffic sign recognition—how far are we from the solution? In: *IEEE. Neural Networks (IJCNN), The 2013 International Joint Conference on*. [S.l.], 2013. p. 1–8. 22
- 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, Oct 2005. ISSN 0162-8828. 48
- MUKAKA, M. M. A guide to appropriate use of correlation coefficient in medical research. *Malawi Medical Journal*, Medical Association of Malawi, v. 24, n. 3, p. 69–71, 2012. 93
- MÜLLER, M. *Information retrieval for music and motion*. [S.l.]: Springer, 2007. v. 2. 42
- NIXON, M. S.; AGUADO, A. S. *Feature extraction & image processing for computer vision*. [S.l.]: Academic Press, 2012. 21
- OLSZEWSKI, R. T. *Generalized feature extraction for structural pattern recognition in time-series data*. Tese (Doutorado) — Carnegie Mellon University, School of Computer Science; Pittsburgh, PA, 2001. 19, 21

- PEREIRA, L. F. *Desenvolvimento e Avaliação de Desempenho do Mecanismo de Reconhecimento Automático de Voz de um Sistema Tutor Inteligente*. Dissertação (Mestrado) — Universidade Estadual do Ceará, Centro de Ciências e Tecnologia, Fortaleza, CE, 8 2015. 41
- POURGHASSEM, H. A hierarchical logo detection and recognition algorithm using two-stage segmentation and multiple classifiers. In: IEEE. *Computational Intelligence and Communication Networks (CICN), 2012 Fourth International Conference on*. [S.l.], 2012. p. 227–231. 24
- RAVEENDRA, K.; REDDY, P.; KISHORE, P. A review on signature and logo identification and extraction using automatic logo based document image retrieval methods. *HELIX*, BIOAXIS DNA RESEARCH CENTRE PRIVATE LIMITED 13-51, SRI LAKSHMI NAGAR COLONY, BESIDES BLG BAZAR, NEAR KAMINENI HOSPITALS, GSI POST BANDAGUDA, HYDERBAD, 500068, INDIA, v. 8, n. 1, p. 2726–2729, 2018. 24
- ROUSSEEUW, P. J. Least median of squares regression. *Journal of the American statistical association*, Taylor & Francis, v. 79, n. 388, p. 871–880, 1984. 109
- ROUTRAY, S.; RAY, A. K.; MISHRA, C. Analysis of various image feature extraction methods against noisy image: Sift, surf and hog. In: *2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT)*. [S.l.: s.n.], 2017. p. 1–5. 48
- SAKOE, H.; CHIBA, S. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, v. 26, n. 1, p. 43–49, Feb 1978. ISSN 0096-3518. 41
- SANTOSH, K. Character recognition based on dtw-radon. In: *IEEE International Conference on Document Analysis and Recognition (ICDAR)*. [S.l.: s.n.], 2011. p. 264–268. 71
- SHEN, L.; WU, W.; ZHENG, S. Tv logo recognition based on luminance variance. *IET*, 2012. 24
- SITTA, A. M.; PEDROSO, H. A.; TADINI, W. M. Razão cruzada: aspectos algébricos e topológicos. *Revista Brasileira de Matemática e Estatística*, v. 20, p. 67–77, 2002. São Paulo, SP. ISSN:0102-0811. 36
- SRESTASATHIERN, P.; YILMAZ, A. Planar shape representation and matching under projective transformation. *Computer Vision and Image Understanding*, v. 115, n. 11, p. 1525 – 1535, 2011. ISSN 1077-3142. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S107731421100169X>>. 48, 50
- SUK, T.; FLUSSER, J. Point-based projective invariants. *Pattern Recognition*, v. 33, n. 2, p. 251 – 261, 2000. ISSN 0031-3203. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0031320399000497>>. 37
- SZELISKI, R. *Computer vision: algorithms and applications*. [S.l.]: Springer Science & Business Media, 2010. 20
- WANG, X. et al. Annotating historical archives of images. In: *ACM. Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries*. [S.l.], 2008. p. 341–350. 39
- YANG, C.; WEI, H.; YU, Q. A novel method for 2d nonrigid partial shape matching. *Neurocomputing*, Elsevier, v. 275, p. 1160–1176, 2018. 49

ZHANG, D.; LU, G. Review of shape representation and description techniques. *Pattern Recognition*, v. 37, n. 1, p. 1 – 19, 2004. ISSN 0031-3203. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0031320303002759>>. 47

ZHANG, H. et al. Text extraction from natural scene image: A survey. *Neurocomputing*, Elsevier, v. 122, p. 310–323, 2013. 22

ZHANG, X.; GAO, X.; TIAN, C. Text detection in natural scene images based on color prior guided msr. *Neurocomputing*, Elsevier, 2018. 22, 23