

**Construção e seleção de janelas  
na combinação de W-operadores**

Marta Magda Dornelles

TESE APRESENTADA  
AO  
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA  
DA  
UNIVERSIDADE DE SÃO PAULO  
PARA  
OBTENÇÃO DO TÍTULO  
DE  
DOUTOR EM CIÊNCIAS

Programa: Pós-Graduação em Ciência da Computação

Orientadora: Profa. Dra. Nina S. T. Hirata

Durante o desenvolvimento deste trabalho a autora recebeu auxílio financeiro da Universidade Estadual de Santa Cruz - UESC, Ilhéus-BA

São Paulo, abril de 2015

# Construção e seleção de janelas na combinação de $W$ -operadores

Esta tese contém as correções e alterações sugeridas pela Comissão Julgadora durante a defesa realizada por Marta Magda Dornelles em 23/07/2015.

O original encontra-se disponível no Instituto de Matemática e Estatística da Universidade de São Paulo.

Comissão Julgadora:

- Profa. Dra. Nina Sumiko Tomita Hirata - IME-USP
- Prof. Dr. Ronaldo Fumio Hashimoto - IME - USP
- Prof. Dr. Flávio Soares Corrêa da Silva - IME-USP
- Prof. Dr. Carlos da Silva dos Santos - UFABC
- Prof. Dr. Hae Yong Kim - EP - USP

# Agradecimentos

Agradeço a Deus e a Nossa Senhora da Penha pela sabedoria para concluir esta tese.

A minha querida mãe Maria e ao meu pai Pedro. Sem a dedicação e amor deles eu não chegaria até aqui.

Aos meus irmãos Juliano e Risieiri, a minha tia Thereza, a toda minha família por entenderem minhas ausências e estarem sempre ao meu lado.

A minha avó Irina, sempre em meu coração.

A professora Nina S. T. Hirata pela paciência e orientação durante estes anos.

Ao professor Roberto Hirata pelo apoio, mesmo que indiretamente, em alguns momentos.

Ao Wonder pela ajuda, incentivo, motivação e carinho dispensados no decorrer deste trabalho.

Aos meus amigos Maysa Macedo, Daniel Chaada, Jesus Mena, David e Alexandre pelos auxílios em diferentes momentos desta caminhada.

Aos colegas David Martins e Carlos Santos por me auxiliarem na utilização de seus códigos.

A Maridalva de Souza Penteado, pela paciência e por incentivar a conclusão deste trabalho.

A minha grande amiga Geloca pela força e paz.

Aos professores e funcionários do IME-USP.

Aos meus colegas de trabalho Jorge, Paulo Ambrósio, Martha, Mathias, Esbel, Suzana, Dany, Honda, a todos da área de computação da Universidade Estadual de Santa Cruz (UESC) e da UAB/UESC, que mesmo indiretamente me auxiliaram no andamento do doutorado.

À UESC por financiar estes estudos.

A todos, meu muito obrigado.



# Resumo

DORNELLES, M. M. **Construção e seleção de janelas na combinação de W-operadores**. 2015. 114 f. Tese (Doutorado) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2015.

W-operadores são uma classe de operadores morfológicos utilizados no processamento de imagens em diferentes áreas do conhecimento. Dada a sua importância, diversos trabalhos vêm sendo realizados no intuito de automatizar sua construção. Em particular, esta pesquisa visa avançar o conhecimento sobre métodos de projeto de W-operadores binários que utilizam técnicas de aprendizagem computacional. Os W-operadores são caracterizados por uma janela de vizinhança  $W$  e quanto maior ela for, maior é a imprecisão dos operadores projetados quando se considera uma quantidade fixa de dados de treinamento. Uma abordagem utilizada para contornar esse problema é o projeto multi níveis de operadores, do qual o projeto dois níveis é um caso particular. No projeto dois níveis realiza-se o treinamento de operadores em dois níveis, tal que no primeiro nível alguns operadores baseados em diferentes janelas são treinados e em seguida, no segundo nível, eles são combinados para gerar o operador final. Uma dificuldade no uso dessa abordagem é a necessidade de escolha manual das janelas dos operadores de primeiro nível, uma vez que a escolha pode depender do conhecimento de um especialista além do fato do espaço de possibilidades ser exponencial. Uma primeira contribuição deste trabalho é a proposta de métodos para a construção de coleções de janelas coerentes com um problema a ser tratado, buscando assim restringir o espaço de possibilidades. Três coleções de janelas são propostas, sendo uma formada por janelas de formas básicas, outra por janelas obtidas por meio de um algoritmo proposto recentemente em uma tese e que explora a magnitude da informação de interação, e a terceira consistindo da união das duas primeiras. Além disso, é proposto um método para estimar um intervalo adequado para o tamanho das janelas e também para delimitar automaticamente o domínio de definição dessas janelas. Uma segunda importante contribuição é a proposta de dois procedimentos para a seleção de combinações de janelas a partir de uma dada coleção de janelas. O primeiro é um algoritmo genético e o segundo baseia-se na ordenação das janelas da coleção, utilizando como critério uma versão corrigida da entropia condicional proposta nesta tese. Os resultados mostram que tanto o algoritmo genético quanto o método baseado em ordenação geram resultados equivalentes ou superiores aos obtidos manualmente. Em especial, o método baseado em ordenação, que é muito mais eficiente que o algoritmo genético, quando utilizado juntamente com uma coleção de janelas básicas definidas sobre um domínio automaticamente estimado, mostra-se como uma opção promissora para a completa automatização do projeto dois níveis de W-operadores.

**Palavras-chave:** W-operador, algoritmo genético, entropia condicional, janelas, ordenação.



# Abstract

DORNELLES, M. M. **Window construction and selection in W-operator combination.** 2015. 114 f. Tese (Doutorado) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2015.

W-operators are a class of morphological operators used in image processing in different fields of knowledge. Given its importance, many studies have been undertaken in order to automate its construction. In particular, this research aims to advance knowledge on methods for the design of binary W-operators that are based on machine learning techniques. W-operators are characterized by a neighborhood window  $W$  and the larger it is, the greater the uncertainty of the operator designed when considering a fixed amount of training data. An approach used to overcome this problem is the multilevel design of operators, of which the two-level design is one particular case. In two-level design training is performed in two levels such that in the first level some operators based on different windows are trained and then, in the second level, they are combined to generate the final operator. A difficulty on using this approach is the need to manually select the windows of the first level operators, since the choice may depend on the knowledge of a specialist, besides the fact that the space of possibilities is exponential. A first contribution of this thesis is the proposal of methods for building collections of windows that are coherent with a problem to be addressed, seeking to restrict the space of possibilities. Three window collections are proposed, being one formed by windows of basic shapes, another by windows obtained by an algorithm proposed in a recent thesis that exploits the magnitude of the interaction information, and the third consisting of the union of the first two. In addition, methods to estimate an appropriate range for the size of the windows and to automatically limit the window definition domain are also proposed. A second important contribution is the proposal of two procedures for selecting window combinations from a given window collection. The first one is a genetic algorithm and the second one is based on sorting the windows of the collection using as the criterion a corrected version of the conditional entropy, newly proposed in this thesis. The results show that both the genetic algorithm and the sorting based method produce results equivalent or superior to those obtained manually. In particular, the sorting based method, which is more efficient than the genetic algorithm, when used with a collection of basic windows defined over an automatically estimated window definition domain, shows to be a promising option for a full automation of the two-level design of W-operators.

**Keywords:** W-operator, genetic algorithm, conditional entropy, windows, sorting.





# Sumário

<b>Lista de Abreviaturas</b>	<b>ix</b>
<b>Lista de Símbolos</b>	<b>xi</b>
<b>Lista de Figuras</b>	<b>xiii</b>
<b>Lista de Tabelas</b>	<b>xvii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Projeto de W-operadores . . . . .	1
1.2 Problema tratado nesta tese . . . . .	3
1.3 Objetivos e contribuições . . . . .	5
1.4 Organização do texto . . . . .	6
<b>2 Referencial teórico</b>	<b>7</b>
2.1 Operadores morfológicos e W-operadores . . . . .	7
2.2 Projeto de W-operadores a partir de exemplos . . . . .	9
2.2.1 Influência da janela no projeto de W-operadores . . . . .	11
2.3 Projeto dois níveis de W-operadores . . . . .	13
2.3.1 Janelas e o tempo de processamento de um projeto dois níveis . . . . .	14
2.3.2 Janelas e a qualidade de um operador dois níveis . . . . .	15
2.4 Elementos estruturantes e janelas . . . . .	16
2.5 Teoria da informação e W-operadores . . . . .	17
<b>3 Coleções de janelas</b>	<b>21</b>
3.1 Estimação de um intervalo de tamanho de janelas . . . . .	21
3.2 Criação de janelas . . . . .	28
3.2.1 Comentários gerais sobre a implementação . . . . .	34
<b>4 Seleção de combinações de janelas usando algoritmo genético</b>	<b>35</b>
4.1 Algoritmo genético para seleção de janelas . . . . .	35
4.1.1 Cromossomo . . . . .	36
4.1.2 Função de avaliação . . . . .	37
4.1.3 Operadores genéticos . . . . .	38
4.1.4 O algoritmo genético WGA . . . . .	41
4.1.5 Comentários gerais sobre a implementação . . . . .	43

4.2	Parâmetros do WGA . . . . .	44
4.2.1	Conjunto de dados . . . . .	44
4.2.2	Experimentos . . . . .	44
4.2.3	Resultados e discussões . . . . .	45
<b>5</b>	<b>Seleção de combinações de janelas usando ordenação</b>	<b>47</b>
5.1	Ordenação para selecionar janelas . . . . .	47
5.2	Entropia condicional como critério de ordenação de janelas . . . . .	49
5.2.1	Entropia condicional corrigida . . . . .	49
5.3	O algoritmo WES . . . . .	50
5.3.1	Comentários sobre a implementação da WES . . . . .	51
5.3.2	Validade da ordenação de uma CJ pela WES . . . . .	51
<b>6</b>	<b>Avaliação das metodologias propostas</b>	<b>55</b>
6.1	Conjuntos de dados . . . . .	55
6.2	Experimentos com o WGA . . . . .	56
6.2.1	Resultados e discussões . . . . .	56
6.3	Experimentos com a WES . . . . .	62
6.3.1	WES versus seleção manual . . . . .	62
6.3.2	Avaliação das coleções de janelas e domínio automático com a WES . . . . .	65
6.3.3	Testes complementares . . . . .	68
6.4	WGA versus WES . . . . .	70
6.5	Operadores um nível versus combinações do WGA e WES . . . . .	73
<b>7</b>	<b>Discussões finais</b>	<b>75</b>
<b>A</b>	<b>Imagens</b>	<b>77</b>
<b>B</b>	<b>Procedimentos de construção de janelas para <math>CJB_a</math></b>	<b>85</b>
B.1	Procedimentos padrões . . . . .	85
B.2	Cálculo automático do domínio de definição das janelas . . . . .	86
<b>C</b>	<b>Publicações</b>	<b>89</b>
	<b>Referências Bibliográficas</b>	<b>91</b>

# Lista de Abreviaturas

AG	Algoritmo genético
<i>BoxCirc</i>	Conjunto de dados referente a extração de retângulos tracejados em imagens de circuitos eletrônicos
<i>Char_s</i>	Conjunto de dados referente a extração do caractere <b>s</b> em textos
<i>CircCirc</i>	Conjunto de dados referente a extração de círculos em imagens de circuitos eletrônicos
CJB	Coleção de janelas básicas
<i>CJB<sub>f</sub></i>	Coleção de janelas básicas fixas, em que a forma está fixada
<i>CJB<sub>a</sub></i>	Coleção de janelas básicas em que a forma está fixada, mas o número de pontos da janela é escolhido aleatoriamente em um intervalo
CJI	Coleção de janelas obtida pela minimização da magnitude da informação de interação
CJBI	União de CJB com CJI
CMIM	Maximização da informação mútua condicional (do inglês <i>Conditional Mutual Information Maximization</i> )
EE	Elemento estruturante
<i>gMax</i>	Número máximo de gerações do algoritmo genético WGA
<i>gMaxMC</i>	Número máximo de gerações do WGA sem aprimorar o melhor cromossomo
ISI	Particionamento Sucessivo de Intervalos (do inglês <i>Incremental Splitting of Intervals</i> )
MAE	Erro absoluto médio (do inglês <i>mean absolute error</i> )
<i>Mapa</i>	Conjunto de dados referente a segmentação de regiões em mapas com uma dada textura
MDL	Comprimento mínimo de descrição (do inglês <i>Minimum Description Length</i> )
<i>MMII</i>	Algoritmo de minimização da magnitude da informação de interação
<i>np</i>	Número de pontos de uma janela.
<b>pc</b>	Probabilidade de crossover
<b>pm</b>	Probabilidade de mutação
<i>RuidoBool</i>	Conjunto de dados referente a filtragem de ruído booleano
SFS	Busca sequencial para frente (do inglês <i>Sequential Forward Search</i> )
<i>TexCirc</i>	Conjunto de dados referente a extração de texto em imagens de circuitos eletrônicos
<i>TexRev</i>	Conjunto de dados referente a segmentação de texto em páginas de revistas
<i>TipoCJ</i>	Tipo de coleções de janelas: CJB, CJI ou CJBI
<i>tp</i>	Tamanho da população de um algoritmo genético
WES	Seleção de janelas usando entropia e baseada em ordenação (do inglês <i>Window selection using Entropy based Sorting</i> )
WGA	Seleção de janelas usando AG (do inglês <i>Window Selection using Genetic Algorithm</i> )



# Lista de Símbolos

$\alpha$	Constante que define o número máximo de pontos de uma janela
$\beta$	Constante que define o número mínimo de pontos de uma janela
$\mathcal{B}_W(\Psi)$	Base de um operador
$\mathbf{D}$	Domínio de uma janela
$ D $	Tamanho do domínio $\mathbf{D}$
$\Delta_i$	Decréscimo relativo da entropia condicional $c_0$ de uma janela $i$ em relação a janela 0
$\delta_B(S)$	Dilatação de $S$ por $B$
$\varepsilon_B(S)$	Erosão de $S$ por $B$
$e_1$	Erro de estimação
$e_2$	Erro de restrição
$E[X]$	Esperança de $X$
$\mathbb{E}$	$\mathbb{Z}^2$
$H(X)$	Entropia
$H(Y X)$	Entropia condicional
$\hat{H}_{c_M}(y X)$	Entropia condicional corrigida proposta por (Martins-Jr <i>et al.</i> , 2004, 2005)
$\hat{H}_{c_P}(y X)$	Entropia condicional corrigida proposta por (Porta <i>et al.</i> , 1998)
$\hat{H}_{c_R}(y X)$	Entropia condicional corrigida proposta por (Reis, 2012)
$\hat{H}_c(y X)$	Entropia condicional corrigida proposta por esta pesquisa
$\hat{H}_{c_0}$	Entropia condicional considerando probabilidade 0 para padrões não observados
$I(Y; X)$	Informação mútua
$I(Y; X_1; X_2)$	Informação de interação
$I$	Imagem ideal
$\mathbb{I}$	Conjunto de imagens ideais referentes a $\mathbb{S}$
$I_W$	Intervalo de tamanho de janelas
$\mathcal{K}_W(\Psi)$	Núcleo ou kernel de um operador
$\lambda$	Operador sup-gerador
$lg$	Função logaritmo na base dois
$\mathbb{N}$	Conjunto dos números naturais
$\mathbb{N}^*$	Conjunto dos números naturais maiores que zero
$o$	Elemento neutro do grupo abeliano $(\mathbb{E}, +)$
$o_i$	Total de observações do padrão $X_i$ em $\hat{H}_{c_M}$
$p(X)$	Distribuição de probabilidade de $X$
$p(X, Y)$	Distribuição de probabilidade conjunta
$p(Y X)$	Probabilidade de $Y$ ocorrer dado que $X$ ocorreu

$\mathcal{P}(\mathbb{E})$	Conjunto das partes de $\mathbb{E}$
$\Psi$	Operador morfológico
$\hat{\Psi}$	Operador morfológico estimado
$\Psi_{otm}$	Operador morfológico ótimo
$\Psi_{Cotm}$	Operador morfológico ótimo restrito ao subespaço $C$
$\psi$	Função característica ou função booleana de um operador
$\hat{\psi}$	Função booleana estimada
$\mathbb{R}$	Conjunto dos reais
$\mathbb{R}^*$	Conjunto dos reais maiores que zero
$R_i$	Decréscimo relativo da entropia condicional $c0$ de duas janelas consecutivas $W_{i-1}$ e $W_i$
$S$	Imagem de entrada
$\mathbb{S}$	Conjunto de imagens de entrada
$S^t$	Reflexão de $S$ em relação a origem
$S + p$	Translação de $S$ por um ponto $p$
$S^c$	Operação de complemento de um conjunto $S$
$T$	Imagem de saída
$\mathbb{T}$	Conjunto de imagens de saída referentes a $\mathbb{S}$
$\mathbf{X}$	Conjunto de todos os padrões definidos por uma janela $W$
$X_i$	Um padrão $i$
$-x$	Oposto de $x$
$W$	Janela
$W^{(2)}$	Janela de segundo nível no projeto dois níveis de W-operadores
$W_{otm}$	Janela ótima
$ W $	Número de pontos de $W$ , tamanho de $W$
$y$	Valor de um pixel em uma imagem ideal
$\mathbb{Z}$	Conjunto dos inteiros
$\mathbb{Z}^*$	Conjunto dos inteiros maiores que zero
$+$	Operação usual de adição
$\cup$	Operação de união entre dois conjuntos
$\cap$	Operação de interseção entre dois conjuntos
$\subset$	Relação de inclusão (propriamente contido) entre dois conjuntos
$\subseteq$	Relação de subconjunto entre dois conjuntos
$\in$	Relação de pertinência entre um elemento e um conjunto
$\omega$	Tamanho da primeira janela em $I_W$ em que $R_i > 1.005$
$\omega_{min}$	Tamanho da janela mínima de $I_W$
$\omega_{max}$	Tamanho da janela máxima de $I_W$

# Lista de Figuras

1.1	Visão geral do procedimento proposto em (Barrera <i>et al.</i> , 1997, 2000) para o projeto de W-operadores. . . . .	2
1.2	Exemplo da coleta de um (padrão, valor de saída), sob uma janela W, em uma imagem de entrada e em sua respectiva imagem ideal. . . . .	2
2.1	Exemplo de uma imagem binária. . . . .	7
2.2	Exemplo de um par de imagens ( $S_i, I_i$ ). . . . .	9
2.3	Exemplo de uma coleta para estimar um W-operador. . . . .	11
2.4	Procedimento básico de treinamento para projetar W-operadores. . . . .	11
2.5	Esquema ilustrativo sobre o erro do operador projetado em relação ao operador ótimo ( $\Delta(\hat{\Psi}_C, \Psi_{otm})$ ). . . . .	12
2.6	MAE para diferentes tamanhos de janelas e conjuntos de treinamento. . . . .	13
2.7	Exemplo de um projeto dois níveis para um W-operador. . . . .	14
2.8	Imagens de saída versus janelas do projeto dois níveis. . . . .	15
2.9	Exemplos de janelas do trabalho de Santos (2010). . . . .	19
3.1	Curvas dos MAEs, sobre imagens de teste, de operadores projetados com janelas crescentes sobre conjuntos de treinamento de tamanhos diferentes e um intervalo de tamanho de janelas em que o menor erro está presente. . . . .	21
3.2	Pré-requisitos para se determinar a entropia condicional e o MAE de um operador individual. . . . .	22
3.3	Gráfico exibindo o comportamento de $\hat{H}_{c_M}(y X)$ , $\hat{H}_{c_0}(y X)$ e do MAE de teste em janelas crescentes para o problema de extrair o caractere S em imagens de texto. . . . .	24
3.4	(a) Imagem original; (b) e (c) Janela de 3 pontos, em vermelho, sobre a imagem original onde o pixel de análise está sob o ponto central da janela; (d) e (e) Janela de 9 pontos onde o pixel de análise está sob o ponto central da janela. . . . .	25
3.5	Gráficos exibindo o comportamento do MAE de teste $\times \hat{H}_{c_0}(y X) \times \hat{H}_{c_M}(y X)$ , em relação ao tamanho de janelas crescentes para cinco problemas diferentes. . . . .	26
3.6	Exemplos de janelas crescentes. . . . .	27
3.7	Exemplo de decréscimo relativo $\Delta_i$ e $\Delta_{i-1}$ . . . . .	27
3.8	Distribuição normal padronizada e a distribuição de $I_W$ . . . . .	29
3.9	$CJB_f$ sobre o domínio $9 \times 9$ e a janela (última janela inferior da direita) obtida pela minimização da entropia condicional de (Martins-Jr <i>et al.</i> , 2004) para o problema de extração do caractere S em imagens de texto. . . . .	31

3.10	Exemplos de janelas verticais para $CJB_a$ : (1) domínio definido pelo usuário com tamanho da janela aleatório, (2) domínio definido pelo usuário com janela padrão e (3) domínio automático com tamanho da janela aleatória. . . . .	32
3.11	$CJB_a$ com domínio do usuário $9 \times 9$ e tamanho das janelas sorteado em $I_W$ . . . . .	32
3.12	$CJB_a$ com o tamanho de janelas escolhido aleatoriamente em $I_W$ e domínio automático para um problema de extração de linhas pontilhadas em diagramas funcionais (Hirata, 2009). . . . .	33
4.1	Esquema básico de funcionamento de um AG. . . . .	36
4.2	Exemplo de um cromossomo contendo duas janelas onde cada gene representa uma janela disposta sobre um domínio ( $5 \times 5$ ). . . . .	36
4.3	No topo da figura as janelas da CJ, os respectivos operadores de primeiro nível e as suas imagens de treinamento para o segundo nível. No meio, um cromosso contendo duas janelas e na parte inferior a execução da função de avaliação para esse cromossomo. . . . .	38
4.4	Exemplo de um <i>crossover</i> entre dois cromossomos de uma população. . . . .	39
4.5	Exemplo de mutaçãoT em um gene de um cromossomo. . . . .	40
4.6	Exemplo de mutaçãoA para um cromossomo. . . . .	41
4.7	Exemplo de mutaçãoE para um cromossomo. . . . .	41
4.8	MAE sobre $\mathbb{S}$ e $\mathbb{I}$ dos indivíduos da primeira e última gerações do WGA para o conjunto de dados $Char_s$ . . . . .	46
5.1	Esquema geral da metodologia de ordenação de janelas para seleção de uma combinação de janelas para um primeiro nível. . . . .	48
5.2	Comportamento dos operadores resultantes das combinações obtidas pela ordenação crescente e decrescente para as duas primeiras janelas da ordenação, para as três primeiras janelas, até as quinze primeiras janelas. . . . .	53
6.1	MAE sobre $\mathbb{S}$ e $\mathbb{I}$ do melhor cromossomo de cada geração do WGA para o conjunto de dados $Char_s$ . . . . .	58
6.2	Janelas usadas nos testes de Hirata (2009) e as melhores janelas obtidas pelo WGA. . . . .	60
6.3	Fragmentos de uma imagem de entrada em $\mathbb{S}_t$ para cada conjunto de dados, suas respectivas imagens ideais, os resultados da aplicação dos operadores manuais e os resultados da aplicação dos operadores obtidos pela combinação do WGA. . . . .	61
6.4	Janelas usadas nos testes de Hirata (2009) e as melhores janelas obtidas pela WES. . . . .	64
6.5	Fragmentos de uma imagem de entrada em $\mathbb{S}_t$ para cada conjunto de dados, suas respectivas imagens ideais, os resultados da aplicação dos operadores manuais e os resultados da aplicação dos operadores obtidos pela combinação selecionada pela WES. . . . .	65
6.6	Janelas usadas nos testes de Hirata (2009) e as melhores combinações de janelas obtidas pela WES. . . . .	67
6.7	Uma imagem de entrada em $\mathbb{S}_t$ , sua respectiva imagem ideal, o resultado obtido pela janelas manuais e o resultado obtido pelas janelas da $WES\_CJB_a$ com domínio automático para $BoxCirc$ . . . . .	68



6.8	Uma imagem de entrada em $\mathbb{S}$ , sua respectiva imagem ideal, o resultado obtido pela janelas manuais e o resultado obtido pelas janelas da $WES\_CJB_a$ com domínio automático para <i>Mapa</i> . . . . .	69
6.9	Janelas manuais e as janelas selecionadas pela $WES\_CJB_a$ para <i>BoxCirc</i> e <i>Mapa</i> . . .	69
6.10	Janelas usadas nos testes de Hirata (2009), as melhores janelas obtidas pelo WGA e pela WES. . . . .	72
6.11	Fragmentos de imagens de entrada para cada conjunto de dados, suas respectivas imagens ideais, os resultados da aplicação dos operadores manuais e os resultados da aplicação dos operadores obtidos pela combinação WGA e pela WES. . . . .	73
A.1	Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual de janelas, o resultado obtido pela combinação do $WGA\_CJB_fI$ e pela combinação da $WES\_CJB_fI$ para <i>Char-s</i> $9 \times 7$ . . . . .	78
A.2	Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual e o resultado obtido pela combinação do $WGA\_CJB_fI$ para <i>Char-s</i> $11 \times 9$ . . . . .	79
A.3	Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual e o resultado obtido pela combinação do $WGA\_CJB_fI$ para <i>Mapa</i> . .	79
A.4	Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual e o resultado obtido pela combinação do $WGA\_CJB_fI$ e pela combinação da $WES\_CJB_f$ para <i>CircCirc</i> . . . . .	80
A.5	Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual e o resultado obtido pela combinação do $WGA\_CJB_f$ e pela combinação da $WES\_CJB_f$ para <i>TexCirc</i> . . . . .	81
A.6	Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual e o resultado obtido pela combinação da $WES\_CJB_a$ com domínio automático para <i>BoxCirc</i> . . . . .	82
A.7	Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual e o resultado obtido pela combinação do $WGA\_CJB_fI$ e pela combinação da $WES\_CJB_f$ para <i>TexRev</i> . . . . .	83
A.8	Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela respectiva combinação manual e o resultado obtido pela combinação do $WGA\_CJB_f$ e pela combinação da $WES\_CJB_f$ para <i>RuidoBool</i> . . . . .	84



# Lista de Tabelas

2.1	Tempos de processamento para o projeto dois níveis. . . . .	15
3.1	Descrição geral das janelas que compõem uma $CJB_f$ . . . . .	30
3.2	Descrição geral das janelas que compõem uma $CJB_a$ . . . . .	31
4.1	Descrição dos conjuntos de dados usados nos experimentos com o WGA. Fonte: Hirata (2009). . . . .	44
4.2	Parâmetros para os testes 1 e 2. . . . .	45
4.3	Parâmetros do WGA que produziram as melhores combinações de janelas e o MAE sobre $(\mathbb{S}, \mathbb{I})$ . . . . .	46
5.1	Descrição dos conjuntos de dados usados nos testes 1, 2 e 3. Fonte: Hirata (2009). . . . .	52
5.2	MAEs de teste resultantes da aplicação dos operadores definidos por escolha aleatória; por operadores definidos pela WES e pela versão decrescente da WES. . . . .	52
5.3	Comparação dos MAEs sobre $(\mathbb{S}, \mathbb{I})$ obtidos pelos operadores das combinações WES com $CJB_f$ utilizando $\hat{H}_c(y X)$ e $\hat{H}_{c_R}(y X)$ . . . . .	53
6.1	Descrição dos conjuntos de dados usados nos experimentos deste capítulo. Fonte*: Hirata (2009). . . . .	56
6.2	Comparação dos MAEs sobre $(\mathbb{S}, \mathbb{I})$ obtidos pelos operadores das combinações manual, WGA_CJI, WGA_CJB_f e WGA_CJB_fI. . . . .	57
6.3	Comparação dos MAEs sobre $(\mathbb{S}_t, \mathbb{I}_t)$ obtidos pelos operadores das combinações manual, metodologia de Santos (2010), WGA_CJI, WGA_CJB_f e WGA_CJB_fI. Subscritas estão as quantidades de janelas do primeiro nível. . . . .	57
6.4	Tempo em horas, número de gerações executadas pelo WGA e o tamanho final da população (TFP) para os testes. Os experimentos foram realizados em uma máquina com o sistema operacional Ubuntu 11.04/Linux e arquitetura com quatro processadores Intel i5-2410M de 2.30GHz cada um. . . . .	59
6.5	Comparação dos MAEs sobre $(\mathbb{S}, \mathbb{I})$ obtidos pelos operadores das combinações manual e WES_CJI, WES_CJB_f e WES_CJB_fI. . . . .	62
6.6	Comparação dos MAEs sobre $(\mathbb{S}_t, \mathbb{I}_t)$ obtidos pelos operadores das combinações de janelas manual, WES_CJI, WES_CJB_f e WES_CJB_fI. . . . .	63
6.7	Tempo de execução da WES com as coleções CJI, CJB_f e CJB_fI em uma máquina com o sistema operacional Ubuntu 11.04/Linux e arquitetura com quatro processadores Intel i5-2410M de 2.30GHz cada um. . . . .	63

6.8 Comparação dos MAEs sobre  $(\mathbb{S}_t, \mathbb{I}_t)$  obtidos pelos operadores das combinações manual e WES para as coleções de janelas obtidas com domínio definido pelo usuário e domínio automático. . . . . 66

6.9 MAEs obtidos pelos operadores das combinações manual e WES. . . . . 68

6.10 MAEs sobre  $(\mathbb{S}_t, \mathbb{I}_t)$  obtidos pelos operadores das combinações manual, WGA e WES. O subscrito representa o número de janelas do primeiro nível, em que  $b$ =básica e  $i$ = janela obtida pela informação de interação. . . . . 70

6.11 MAEs sobre  $(\mathbb{S}_t, \mathbb{I}_t)$  obtidos pelos operadores individuais, WGA e WES. O subscrito representa o número de janelas do primeiro nível, em que  $b$ =básica e  $i$ = janela obtida pela informação de interação. . . . . 74

# Capítulo 1

## Introdução

Operadores morfológicos são ferramentas utilizadas em diferentes tipos de transformações no contexto de processamento de sinais e imagens. Informalmente, eles realizam uma transformação mediante a análise da forma dos objetos contidos em imagens de interesse. Em geral, os operadores são caracterizados por elementos estruturantes que são pequenos padrões preestabelecidos, usados para investigar uma imagem de estudo (Soille, 2003). Neste trabalho são consideradas imagens binárias.

Uma classe de operadores morfológicos que se destaca é a dos operadores invariantes a translação e localmente definidos por uma janela  $W$ , denominados **W-operadores** (Banon e Barrera, 1991; Barrera *et al.*, 2000). Essa classe abrange uma grande variedade de operadores dos quais fazem parte a erosão e a dilatação (operadores elementares), abertura, fechamento, gradiente morfológico, *hit-miss*, filtro da mediana e esqueletonização (Martins-Jr, 2004). Eles podem ser utilizados em diferentes aplicações, em diversas áreas do conhecimento, como na indústria de chapas metálicas para medir deformações nas mesmas (Tuzikov *et al.*, 1992); na saúde, para segmentar imagens mamográficas auxiliando no processo de identificação de câncer de mama (Basha e Prasad, 2009) ou na composição de um sistema de processamento de imagens de angiogramas da retina, a fim de facilitar o diagnóstico e acompanhamento da degeneração macular (centro do campo visual) relacionada a idade, que é uma das causas de cegueira nos países industrializados (Barthes *et al.*, 2001); na bioinformática, compondo um mecanismo de segmentação de genes em imagens de microarrays (Hirata-Jr *et al.*, 2002); em Sistemas de Informação Geográfica, na constituição de métodos para extração de rodovias em imagens de satélite (Ishikawa *et al.*, 2010), dentre outras.

Apesar do uso de W-operadores (também chamados de operadores neste texto) abranger diferentes domínios de aplicação, a obtenção de resultados satisfatórios depende de uma composição adequada de operadores elementares e da escolha apropriada de elementos estruturantes. Portanto, o seu emprego requer um especialista no assunto, alguém que conheça os operadores e saiba como aplicá-los para se obter um resultado desejado (Soille, 2003).

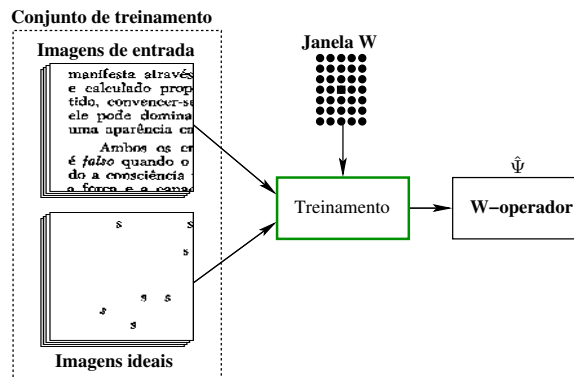
### 1.1 Projeto de W-operadores

Visando facilitar o uso dos operadores por profissionais da área e por aqueles que não são familiarizados com o tema, pesquisas vêm sendo realizadas no intuito de projetá-los automaticamente ou semi-automaticamente. W-Operadores projetados automaticamente já foram aplicados na extração de pontos extremos em segmentos de reta, extração de bordas, na filtragem de ruídos, na segmentação de caracteres, segmentação de códigos de barra, identificação de fissuras em metais, na segmentação de texto, em reconhecimento de textura e em reconhecimento de padrões em diagramas (Hirata, 2000; Tomita, 1996).

Uma abordagem utilizada em alguns trabalhos é a de projetar  $W$ -operadores a partir de um conjunto de exemplos, obtidos de imagens de entrada (referentes a um problema a ser resolvido) e suas respectivas imagens de saída desejadas, também conhecidas por imagens ideais. Geralmente essas imagens são chamadas de **imagens de treinamento** ou **conjunto de treinamento**.

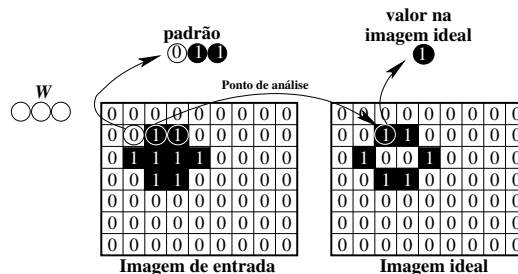
Um dos primeiros trabalhos que faz uso desta metodologia restrita à classe de operadores crescentes é devido a Dougherty (1992a,b), que apresenta uma proposta baseada em estimação estatística. Redes neurais também foram exploradas usando imagens de treinamento para se determinar operadores (Araújo *et al.*, 2007).

Barrera *et al.* (1997, 2000) propuseram um procedimento que recebe como entrada imagens de treinamento e uma janela  $W$ , definida manualmente de acordo com o conhecimento do projetista. A partir dessas entradas, o procedimento constrói um operador  $\hat{\Psi}$  para o problema estabelecido pelas imagens. Uma vez projetado o operador, ele pode ser aplicado em outras imagens pertencentes ao mesmo contexto. Uma visão geral dessa metodologia pode ser vista na Figura 1.1.



**Figura 1.1:** Visão geral do procedimento proposto em (Barrera *et al.*, 1997, 2000) para o projeto de  $W$ -operadores.

A janela  $W$  delimita a vizinhança de um ponto a ser analisado em uma imagem de entrada. Esse ponto e sua vizinhança compõem um **exemplo**, também chamado de **padrão**. O objetivo é "aprender" que valor deve ser associado ao ponto de análise, de acordo com sua vizinhança, em sua respectiva imagem de saída. Em linhas gerais, esse aprendizado ocorre a partir de padrões coletados em imagens de entrada, do conjunto de treinamento, e os correspondentes valores dos pontos de análise nas imagens ideais. Os padrões coletados determinam uma estimativa da probabilidade de ocorrência do par (padrão, valor de saída) que é utilizada para definir o operador. Um exemplo de uma coleta pode ser observado na Figura 1.2.



**Figura 1.2:** Exemplo da coleta de um (padrão, valor de saída), sob uma janela  $W$ , em uma imagem de entrada e em sua respectiva imagem ideal.

Após o treinamento, todos os padrões possíveis de serem coletados através de um janela devem ter um valor de saída associado, mesmo aqueles que não aparecem nas imagens de treinamento, pois posteriormente eles podem estar presentes em imagens diferentes dessas e, nesse caso, o sistema

deve ser capaz de vincular um valor a eles, ou seja, deve ser capaz de generalizar. Ao final de todo o processo existe uma associação para cada padrão, definindo assim o operador. Uma vez projetado, esse operador pode ser aplicado sobre imagens de teste para avaliar sua qualidade. A avaliação de qualidade é feita através do erro absoluto médio (*mean absolute error* - **MAE**). Logo, o objetivo do procedimento é projetar operadores com menor MAE possível.

Em relação à qualidade do operador projetado, quando a janela é pequena, normalmente ela não é capaz de capturar padrões representativos em um dado problema (**erro de restrição**), ainda que o operador fique bem definido. Assim, janelas grandes são desejáveis pois estabelecem uma vizinhança mais representativa. No entanto, quanto maior a janela mais padrões existem para serem avaliados, implicando um processo de generalização mais falho (**erro de estimação**). Dessa forma, definir o número de pontos de uma janela é uma tarefa que deve levar em conta um balanceamento entre o erro de restrição e o erro de estimação. Nesse contexto, é apresentado por Hirata (2009) o projeto multi níveis de operadores morfológicos.

O projeto multi níveis baseia-se em uma técnica conhecida por *stacked generalization* proposta por Wolpert (1992), que explora a combinação de classificadores em múltiplos níveis de treinamento. No projeto multi níveis, vários operadores são projetados em um primeiro nível de treinamento e os resultados da aplicação desses operadores são usados para um segundo nível de treinamento, de forma a determinar como combinar esses resultados. Níveis adicionais de treinamento podem ser repetidos sucessivamente. Nesse esquema, a cada operador está associada uma janela que define qual vizinhança do pixel sendo processado é levada em consideração.

Resultados experimentais apresentados em Hirata (2009) mostram que, em geral, um treinamento em dois níveis produz melhores resultados. Mais que isso, os melhores resultados são obtidos quando, no operador de segundo nível, considera-se apenas um ponto (o pixel sendo processado) de cada uma das imagens resultantes dos operadores de primeiro nível. Esta configuração particular pode ser interpretada como uma forma de combinação de operadores de imagens, isto é, cada um dos operadores de primeiro nível gera um resultado e o operador de segundo nível decide um resultado final que seria uma espécie de consenso. Note, porém, que o consenso não é dado por uma regra fixa como a da maioria, mas determinado pelo operador de segundo nível que é obtido também por treinamento.

Desta forma, janelas de tamanhos grandes que em geral são inviáveis no projeto de um operador que utiliza apenas um nível, por requerer muita memória e tempo de processamento, podem ser subdivididas em janelas menores e essas podem ser utilizadas para treinar os operadores de primeiro nível. Uma vez que o operador de segundo nível combina os resultados dos operadores de primeiro nível, indiretamente ele leva em consideração uma vizinhança maior na imagem de entrada, do que as consideradas individualmente pelos operadores de primeiro nível. Uma possível explicação para a melhoria de desempenho desses operadores de dois níveis é que, apesar de estarem associadas a janelas maiores (menor erro de restrição), o erro de estimação é mantido pequeno.

## 1.2 Problema tratado nesta tese

No método de combinação de operadores descrito na seção anterior, a escolha do número de operadores a serem combinados assim como das respectivas janelas é realizada manualmente. A escolha de janela, mesmo no caso de projeto de um único operador, é um problema difícil para o qual não existe uma solução consolidada. Algumas abordagens foram propostas usando medidas da teoria da informação para a escolha (Vaquero, 2006) ou criação (Martins-Jr *et al.*, 2005) de janelas para operadores de apenas um nível.

Visando uma maior automação do processo de projeto de operadores, nesta tese foi considerado o problema de determinar:

- O número de operadores a serem combinados, e
- A forma e tamanho das respectivas janelas

no contexto de combinação de operadores em dois níveis. Os operadores obtidos em dois níveis de treinamento são denominados **operadores dois níveis**.

Para solucionar o problema proposto, caso seja considerado uma formulação como um problema de busca no espaço de todas as possíveis soluções, esse espaço pode ser caracterizado como segue. Suponha  $\mathbf{D}$  o domínio de definição de janelas, isto é, todas as janelas a serem consideradas estão necessariamente restritas a esse domínio. Se denotarmos por  $|D|$  o número de pontos deste domínio, o número de possíveis janelas em  $\mathbf{D}$  é dado por  $2^{|D|}$ . Assim, o número possível de combinações de operadores, ou operadores dois níveis, é dado por:

$$\begin{aligned}
 k &= \begin{array}{c} \text{n}^\circ \text{ de combinações} \\ \text{diferentes com} \\ 2 \text{ janelas} \end{array} + \begin{array}{c} \text{n}^\circ \text{ de combinações} \\ \text{diferentes com} \\ 3 \text{ janelas} \end{array} + \dots + \begin{array}{c} \text{n}^\circ \text{ de combinações} \\ \text{diferentes com} \\ n \text{ janelas} \end{array} \\
 k &= \frac{2^{|D|}}{2!(2^{|D|}-2)!} + \frac{2^{|D|}}{3!(2^{|D|}-3)!} + \dots + \frac{2^{|D|}}{n!(2^{|D|}-n)!} \\
 k &= \sum_{j=2}^n \frac{2^{|D|}}{j!(2^{|D|}-j)!},
 \end{aligned}$$

em que  $n$  é o número máximo de janelas no primeiro nível e cada parcela da soma é uma *r-combinação*<sup>1</sup>. Portanto, experimentar todas as possibilidades para combinações com até  $n$  janelas no primeiro nível é de complexidade assintótica exponencial.

Um ponto a se observar na escolha de janelas é que se elas forem muito parecidas, os operadores de primeiro nível irão fornecer, após sua aplicação, um conjunto de imagens de entrada muito semelhantes para o segundo nível, comprometendo o ganho do primeiro nível para o segundo nível.

Outra observação é que as janelas podem não corresponder a toda a vizinhança necessária para se estabelecer um operador para um determinado cenário de aplicação. Nessa situação, o operador irá realizar a transformação de maneira insatisfatória. Portanto, qualquer procedimento que pretenda automatizar a combinação de operadores deve observar a perda de informação (falta de janelas representativas) e a redundância entre operadores (Santos, 2010).

Como o espaço de busca das possíveis combinações é exponencial, conforme comentado acima, uma metodologia que pode ser utilizada é restringir adequadamente esse espaço. O primeiro trabalho seguindo essa ideia foi apresentado em (Santos, 2010). Ele utilizou conceitos da teoria da informação (Shannon, 1948) e propôs um algoritmo de minimização da magnitude da informação de interação (Jakulin e Bratko, 2003) como base para criar uma coleção de janelas. Essas são processadas e submetidas ao primeiro nível de treinamento. Os operadores resultantes desse processo são ordenados em ordem crescente segundo o critério de maximização da informação mútua condicional (*Conditional Mutual Information Maximization - CMIM*) (Fleuret e Guyon, 2004). Em seguida, o projeto dois níveis é executado para a primeira e segunda janelas da ordenação, depois para a primeira, segunda e terceira janelas, e assim sucessivamente. Os operadores dois níveis resultantes desses projetos são avaliados pelo princípio do comprimento mínimo de descrição (Minimum Description Length - MDL) (Grünwald, 2007) para a escolha da melhor combinação realizada. Os

<sup>1</sup> $C_n^r = \frac{r!}{n!(r-n)!}$  (Lipschutz e Lipson, 2004)



resultados apresentados não foram suficientes para uma comparação mais definitiva com aqueles estabelecidos manualmente em (Hirata, 2009).

Outros critérios podem ser pesquisados e usados no processo de ordenação de janelas, para que posteriormente as melhores janelas dessa ordenação sejam escolhidas para uma combinação de operadores. A entropia condicional, por exemplo, pode ser avaliada como critério de ordenação, pois já foi usada na definição de janelas para operadores de apenas um nível em Vaquero (2006) e em Martins-Jr *et al.* (2005).

O uso de técnicas de buscas, como *simulated annealing* (Kirkpatrick *et al.*, 1983), busca tabu (Glover e Laguna, 1997) e algoritmos genéticos (Holland, 1975) são alternativas plausíveis para auxílio na seleção de janelas. Dentre essas destacam-se os **algoritmos genéticos (AGs)**, que podem ser definidos como técnicas de busca baseadas no processo biológico de evolução natural das espécies.

Os AGs já foram aplicados com sucesso em diferentes áreas como na alocação de recursos, no setor elétrico, bioinformática, medicina, análise de séries temporais, aeronáutica, robótica e outros (Coley, 1998; Linden, 2006). Atualmente, existem diversos produtos comerciais e não comerciais que são baseados em um algoritmo genético. Os AGs também foram usados no projeto de filtros morfológicos como nos trabalhos de Kraft *et al.* (1995) e Jelodar *et al.* (2006). Yoda *et al.* (1994) utilizaram AGs para definir uma sequência de erosões e dilatações (definidas sobre 4 ou 8 vizinhos) que realizam um determinado mapeamento estabelecido por um par de imagens entrada/ideal.

No que diz respeito à escolha de operadores a serem combinados, o uso de AGs é apropriado uma vez que essa escolha pode ser vista como um problema de busca, em que o objetivo é encontrar combinações em um vasto espaço de possibilidades. Existem vários trabalhos, em diferentes áreas de conhecimento, relatando o uso de AGs para a seleção de parâmetros que estabelecem arquiteturas/topologias como, por exemplo:

- Na determinação de arquiteturas de redes neurais (Arifovic e Gencay, 2001; Fiszlelew *et al.*, 2007; Ileanã *et al.*, 2004) estabelecendo quantas camadas, o número de neurônios em cada camada e a ligação entre os neurônios.
- Na definição automática do número de clusters, bem como na estipulação das partições no uso do método k-means (Liu *et al.*, 2008).
- Na localização ótima de um conjunto de *switches* em uma rede de comunicação móvel (Salcedo-Sanz *et al.*, 2008).

### 1.3 Objetivos e contribuições

O objetivo desta pesquisa foi investigar e propor procedimentos para restringir o espaço de busca de janelas, bem como definir mecanismos para a seleção de combinação de janelas a partir deste espaço restrito.

Uma contribuição deste trabalho é a proposição de procedimentos que, encadeados, podem compor uma ferramenta automática para o projeto de W-operadores binários. Tal ferramenta contribuiria na utilização de W-operadores por pessoas com pouco ou nenhum conhecimento no tema, bem como auxiliaria o trabalho de especialistas da área.

As contribuições resultantes deste trabalho são:

- a) Uma metodologia para estimar um intervalo de tamanho de janelas para um dado problema. Esse intervalo pode ser usado para a construção de janelas;
- b) Procedimentos para a construção de coleções de janelas das quais uma combinação pode ser selecionada;
- c) Um AG para a seleção de combinação de janelas em uma coleção de janelas predefinida;
- d) Uma versão corrigida da entropia condicional como critério de ordenação de uma coleção de janelas e um método para seleção de uma combinação de janelas a partir desta ordenação.

Um aspecto relevante é que cada metodologia proposta pode também ser utilizada individualmente na composição de outras ferramentas ou como instrumento para auxiliar um especialista no projeto de um operador.

Como exemplo, o procedimento de ordenação segundo a entropia condicional corrigida pode ser usado para definir uma combinação de janelas para o projeto dois níveis. Essa, por sua vez, pode ser aprimorada pelo especialista que pode adicionar ou remover janelas da combinação. Ou ainda, o especialista pode adicionar novos algoritmos ao arcabouço que constrói uma coleção de janelas ou mesmo adicionar diretamente janelas à coleção criada.

Uma contribuição interessante, mesmo que empírica, foi a definição de um limite superior para o número de janelas no primeiro nível. Esse limite permite reduzir ainda mais o espaço de busca.

## 1.4 Organização do texto

Este texto está organizado basicamente em duas partes. Na primeira, são descritos brevemente os trabalhos prévios relacionados a esta pesquisa. Na segunda, as metodologias propostas são apresentadas, avaliadas e por fim comparadas.

No Capítulo 2 são descritos os conceitos básicos necessários para o entendimento deste trabalho. São revisados alguns pontos da morfologia matemática com foco nos W-operadores, bem como é exposta a ideia geral de funcionamento do projeto dois níveis. Algumas medidas da teoria da informação e trabalhos relacionados com o tema estão postos no capítulo.

As metodologias para a construção de coleções de janelas são relatadas no Capítulo 3. Basicamente três coleções são propostas. A primeira, baseada em janelas básicas utilizadas em outros trabalhos. A segunda, que utiliza o algoritmo de minimização da informação de interação (Santos, 2010) para construção das janelas. A terceira que é a união das duas primeiras coleções.

Os Capítulos 4 e 5 expõem as metodologias para a seleção de combinação de janelas. Um algoritmo genético é proposto no Capítulo 4 e no seguinte, um procedimento baseado na ordenação de janelas utilizando uma versão corrigida, proposta por esta pesquisa, da entropia condicional é apresentado.

Comparações entre as metodologias são descritas e discutidas no Capítulo 6 e as discussões finais estão no Capítulo 7.

# Capítulo 2

## Referencial teórico

Neste capítulo são revistos conceitos sobre W-operadores e seu projeto, incluindo definições utilizadas ao longo deste trabalho. São expostas também discussões sobre a influência das janelas na qualidade de um operador projetado e no tempo de processamento do projeto. Nas últimas seções são apresentados trabalhos relacionados à construção de janelas no projeto de W-operadores, os quais contribuíram para o desenvolvimento desta tese.

### 2.1 Operadores morfológicos e W-operadores

Operadores morfológicos são objetos de estudo da Morfologia Matemática (MM) que teve sua origem nos trabalhos de [Matheron \(1975\)](#) e [Serra \(1982\)](#). Inicialmente, a MM foi formulada para lidar com imagens binárias e, posteriormente, foi estendida para mapeamentos entre reticulados completos ([Serra, 1988](#)).

Seja  $\mathbb{Z}$  o conjunto dos números inteiros e  $\mathbb{E} = \mathbb{Z}^2$ . O par  $(\mathbb{E}, +)$  é um grupo Abelianiano com respeito à operação binária usual de adição. O elemento neutro (chamado de origem) de  $(\mathbb{E}, +)$  é denotado por  $o$ . O oposto de  $z \in \mathbb{E}$  é denotado por  $-z$ .

Uma imagem binária, ou simplesmente imagem neste trabalho, pode ser representada por uma função  $f : \mathbb{E} \rightarrow \{0, 1\}$ . Informalmente, ela é um conjunto não vazio de pontos (**pixels**) que podem assumir o valor 0 (considerado aqui como fundo da imagem) ou 1 (objeto na imagem), como no exemplo da Figura 2.1. No contexto da morfologia matemática, os objetos de uma imagem  $f$  são vistos como subconjuntos de  $\mathbb{E}$ , isto é  $S = \{z \in \mathbb{E} : f(z) = 1\}$ , onde as operações usuais de conjuntos como união ( $\cup$ ), interseção ( $\cap$ ) e complemento ( $S^c$ ) podem ser usadas.

0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

**Figura 2.1:** Exemplo de uma imagem binária onde 0 representa o seu fundo e 1 diz respeito a um objeto na imagem.

Um operador morfológico binário  $\Psi$  pode ser definido como um mapeamento de  $\mathcal{P}(\mathbb{E})$  em  $\mathcal{P}(\mathbb{E})$ , tal que  $\mathcal{P}(\mathbb{E})$  é o conjunto das partes de  $\mathbb{E}$ .

Vários operadores são caracterizados por **elementos estruturantes** (EEs). Os EEs são subconjuntos de  $\mathbb{E}$  usados para examinar uma imagem, buscando características de interesse.

Além dos EEs, duas operações sobre conjuntos são importantes para a definição dos operadores morfológicos elementares: a **translação** e a **reflexão**. Seja  $S$  uma imagem binária e  $p$  um ponto em  $\mathbb{E}$ . A translação de  $S$  por  $p$  é dada por  $S + p = \{z + p : z \in S\}$ , e a reflexão de  $S$  em relação a origem é dada por  $S^t = \{-z : z \in S\}$ .

Dois operadores elementares da MM são a **erosão** e a **dilatação**. A erosão de uma imagem  $S$  por um EE  $B$  pode ser definida por  $\varepsilon_B(S) = \{z \in \mathbb{E} : B + z \subseteq S\}$ . Por outro lado, a dilatação de  $S$  por  $B$  é dada por  $\delta_B(S) = \{z \in \mathbb{E} : (B^t + z) \cap S \neq \emptyset\}$ . Geralmente, uma imagem  $S$  é chamada de **imagem de entrada**, e uma imagem obtida após a aplicação de um operador é chamada de **imagem de saída**.

Um ponto importante sobre os operadores morfológicos são as propriedades que os caracterizam. Duas delas são destacadas neste texto. A primeira é a **invariância a translação (i.t.)**. Um operador é dito ser invariante a translação quando  $\Psi(S + z) = \Psi(S) + z$  para todo  $z$  pertencente a  $\mathbb{E}$ . A erosão e a dilatação são operadores invariantes a translação.

A segunda propriedade sobre os operadores é a definição local por uma **janela  $W$** , onde uma janela é um subconjunto finito e não vazio de  $\mathbb{E}$ . Um operador é **localmente definido (l.d.)** por uma janela  $W$  se, para todo  $z \in \mathbb{E}$  e  $S \subseteq \mathbb{E}$ ,  $z \in \Psi(S \cap (W + z)) \iff z \in \Psi(S)$ . Assim, o resultado desse tipo de operador sobre uma imagem de entrada em um pixel  $z$ , depende apenas da vizinhança desse pixel estabelecida pela janela  $W$ . Operadores que apresentam essas duas propriedades, i.t. e l.d., são denominados **W-operadores**.

Um operador importante para o projeto de W-operadores é o chamado **operador sup-gerador** ( $\lambda$ ). Ele pode ser expresso como:  $\lambda_{(A,B)}^W(S) = \varepsilon_A(S) \cap (\delta_{(B^c)^t}(S))^c$ , onde  $A$  e  $B$  são dois elementos estruturantes tal que  $A \subseteq B \subseteq W$ .

O operador sup-gerador é o eixo de um importante teorema para o projeto de W-operadores: o **teorema da decomposição canônica** (Banon e Barrera, 1991). Esse teorema afirma que dado um W-operador  $\Psi : \mathcal{P}(\mathbb{E}) \rightarrow \mathcal{P}(\mathbb{E})$  tem-se:

$$\Psi(S) = \bigcup \{\lambda_{(A,B)}^W(S) : [A, B] \subseteq \mathcal{K}_W(\Psi)\}, \quad (2.1)$$

no qual  $[A, B]$  é o intervalo fechado com extremidades  $A$  e  $B$ , isto é  $[A, B] = \{X | A \subseteq X \subseteq B\}$ , e  $\mathcal{K}_W(\Psi)$  é o **núcleo** ou **kernel** do operador. Esse último pode ser definido por  $\mathcal{K}_W(\Psi) = \{X \in \mathcal{P}(W) : o \in \Psi(X)\}$ , ou seja, são aquelas configurações da janela mapeadas para 1 na imagem de saída pela aplicação do operador. Esse teorema garante que qualquer W-operador pode ser decomposto em termos de erosões, dilatações, complementos, interseções e uniões.

Na prática, existem redundâncias nestas decomposições, e uma forma minimal é apresentada pelo **teorema da decomposição minimal** (Banon e Barrera, 1991). Ele fundamenta-se na **base** de um W-operador em vez de seu núcleo. A base é o conjunto de todos os intervalos maximais do núcleo. Um intervalo, pertencente a uma coleção de intervalos, é dito ser maximal se ele não está contido propriamente em nenhum outro intervalo pertencente à coleção. Formalmente o teorema afirma que dado um W-operador  $\Psi$  tem-se:

$$\Psi(S) = \bigcup \{\lambda_{(A,B)}^W(S) : [A, B] \in \mathcal{B}_W(\Psi)\}, \quad (2.2)$$

na qual  $\mathcal{B}$  é a base do W-operador.

Esse teorema estabelece uma fundamentação sólida para a construção automática de W-operadores.

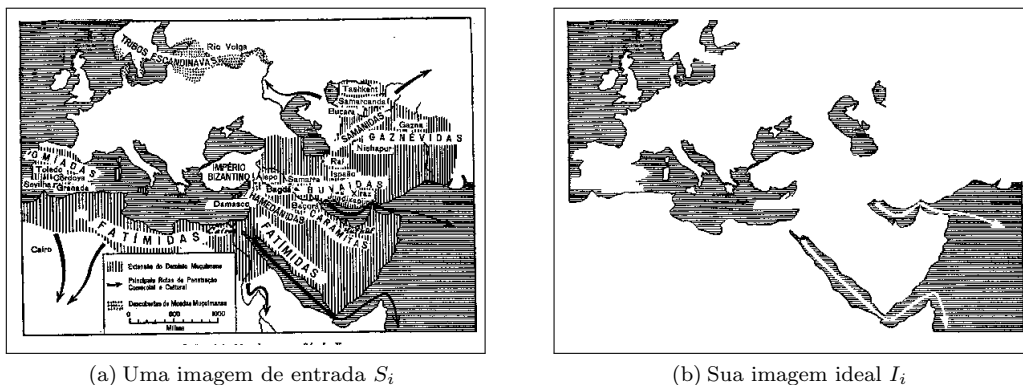
Outro fato importante sobre a forma canônica e minimal dos W-operadores binários é que elas correspondem à forma canônica e minimal das funções booleanas (Tomita, 1996). Assim, um W-operador pode ser caracterizado por uma função booleana  $\psi : \mathcal{P}(W) \rightarrow \{0, 1\}$ , tal que  $z \in \Psi(S) \iff \psi((S - z) \cap W) = 1$ .

Assumindo que cada ponto  $w_j \in W$  está associado a uma variável binária  $b_j$ , e fazendo  $b_j = 1$  se e somente se  $w_j \in (S - z) \cap W$ , a função  $\psi$  pode ser vista como uma função booleana de  $v = |W|$  variáveis, onde  $|W|$  é o **número de pontos de  $W$**  (também referenciado neste texto como **tamanho de  $W$** ). Dessa forma, determinar um W-operador pode ser visto como encontrar sua **função booleana  $\psi$** , também conhecida por **função característica**.

## 2.2 Projeto de W-operadores a partir de exemplos

O projeto de W-operadores a partir de **imagens de treinamento** considera pares compostos por uma imagem de entrada  $S_i$  e uma imagem ideal  $I_i$ , tal que  $S_i$  pertence a um conjunto de imagens de entrada  $\mathbb{S}$ ,  $I_i$  pertence a um conjunto de imagens ideais  $\mathbb{I}$ , e  $i$  é um inteiro maior que zero. A imagem ideal  $I_i$  é aquela que se deseja obter após o processamento de  $S_i$  por um operador. Um exemplo de um par de imagens de treinamento pode ser observado na Figura 2.2.

Baseando-se nos pares de imagens  $(S_i, I_i)$ , o objetivo do projeto de W-operadores é encontrar um operador que consiga produzir imagens de saída (onde cada uma será denominada por  $T_i$ ) tão próximas quanto possíveis das ideais. Logo, o propósito é obter um operador  $\Psi$ , tal que  $T_i = \Psi(S_i)$  seja o mais próximo possível de  $I_i$ . Essa proximidade pode ser quantificada por alguma medida de erro.



**Figura 2.2:** Exemplo de um par de imagens  $(S_i, I_i)$ .

Nos trabalhos com W-operadores é comum o uso do **erro absoluto médio (MAE - mean absolute error)** como medida de erro, adotando um modelo probabilístico no qual as imagens em  $\mathbb{S}$  e  $\mathbb{I}$  são consideradas realizações de conjuntos aleatórios discretos  $S$  e  $I$ , respectivamente, definidos em  $\mathbb{E}$ , conjuntamente estacionários em relação a uma janela  $W$ , ou seja, a probabilidade de um dado **padrão  $X$**  ser observado através de  $W$  em uma imagem  $S_i$ , associado a um valor booleano na imagem de saída  $I_i$ , é a mesma para qualquer translação de  $W$  sobre  $S_i$ .

Este fato permite tratar um padrão  $X_z$ <sup>1</sup> em  $S_i$  e o valor booleano  $y_z$ <sup>2</sup> em  $I_i$  como uma realização de um vetor aleatório  $\mathbf{X}$  e de uma variável aleatória  $\mathbf{y}$ . Ou seja, o par  $(\mathbf{X}, \mathbf{y})$  obedece a uma distribuição de probabilidade  $p(\mathbf{X}, \mathbf{y})$ . Dessa forma, para caracterizar quanto  $I_i$  e  $T_i$  estão próximas pode-se calcular o MAE sem definir uma posição específica (Dougherty, 1998). Essa medida pode ser expressa como:

$$\begin{aligned} MAE(\Psi) &= E[|\psi(\mathbf{X}) - \mathbf{y}|] \\ &= \sum_{(X,y)} |\psi(X) - y| p(X, y) \\ &= \sum_X \psi(X) p(X, 0) + |\psi(X) - 1| p(X, 1) \\ MAE(\Psi) &= \sum_{\{X:\psi(X)=1\}} p(X, 0) + \sum_{\{X:\psi(X)=0\}} p(X, 1). \end{aligned} \quad (2.3)$$

Um **operador ótimo** é aquele que apresenta MAE mínimo (Dougherty, 1992a). A partir da equação 2.3 um operador ótimo pode ser caracterizado como :

$$\psi_{otm}(X) = \begin{cases} 1, & \text{se } p(X, 0) < p(X, 1) \\ 0, & \text{se } p(X, 0) > p(X, 1) \\ 0 \text{ ou } 1, & \text{se } p(X, 0) = p(X, 1). \end{cases} \quad (2.4)$$

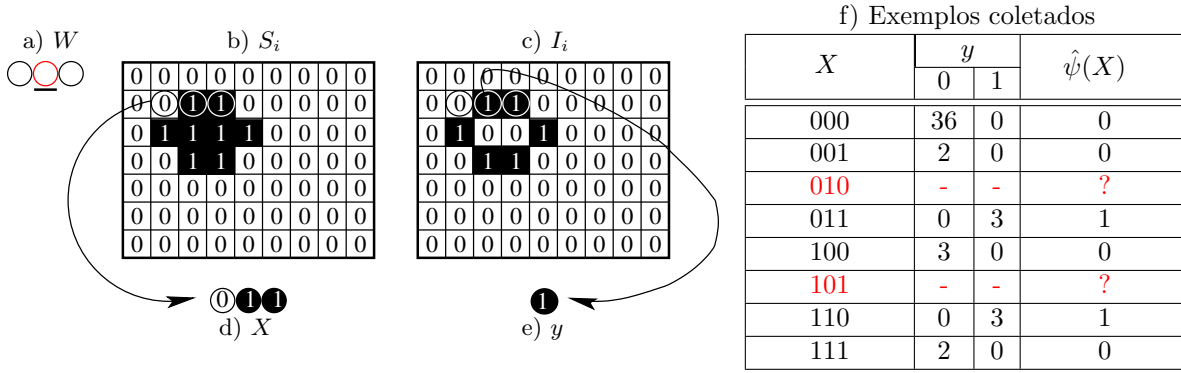
Portanto, conhecendo as probabilidades conjuntas é possível determinar o operador ótimo. Na prática, dificilmente essas probabilidades são conhecidas. O que se utiliza são estimativas obtidas a partir de imagens de treinamento e, portanto, o operador obtido poderá ter erro maior que o operador  $\Psi_{otm}$ . Logo, uma boa estimativa influencia na qualidade do operador encontrado.

Três passos principais podem ser definidos para se estimar W-operadores a partir de pares de imagens de treinamento  $(\mathbb{S}, \mathbb{I})$  e de uma janela  $W$  preestabelecida (Hirata, 2011):

- 1) **Estimação de  $p(\mathbf{X}, \mathbf{y})$ :** padrões (ou exemplos ou também observações) são coletados das imagens de entrada por meio de uma janela  $W$ . Essa coleta ocorre transladando-se  $W$  sobre as imagens em  $\mathbb{S}$ . Para cada pixel  $z$  em uma imagem  $S_i$ , o padrão observado  $X_z$  é anotado bem como o valor correspondente em  $I_i$ , ou seja  $y_z$ . Um exemplo dessa coleta pode ser visto na Figura 2.3.
- 2) **Decisão ótima:** para cada padrão observado  $X$  é associado um valor ao operador estimado  $\hat{\psi}(X)$ , que minimiza o MAE (baseado na equação 2.4) usando-se as probabilidades estimadas. Quando  $\hat{p}(X, 1) = \hat{p}(X, 0)$ , adota-se  $\hat{\psi}(X) = 0$ . A última coluna da tabela, exposta na Figura 2.3, é um exemplo do resultado desse passo. Esse resultado define parcialmente os elementos que pertencem ao núcleo do operador.
- 3) **Generalização:** é possível verificar que na tabela da Figura 2.3 existem dois padrões, 010 e 101, que não foram observados na imagem de entrada. Assim, a função  $\hat{\psi}(X)$  não está completamente especificada. Para solucionar esse problema, algumas técnicas de aprendizagem computacional como árvores de decisão (Breiman *et al.*, apud Hirata (2000)), redes neurais (Bishop (1996), apud Hirata (2000)) e outras, podem ser utilizadas. O trabalho de Tomita (1996) fez uso do algoritmo ISI (Incremental Splitting of Intervals) no processo de generalização. Esse algoritmo realiza a minimização de funções booleanas, sendo que uma vantagem de seu uso é a facilidade de mapeamento entre funções booleanas e operadores morfológicos.

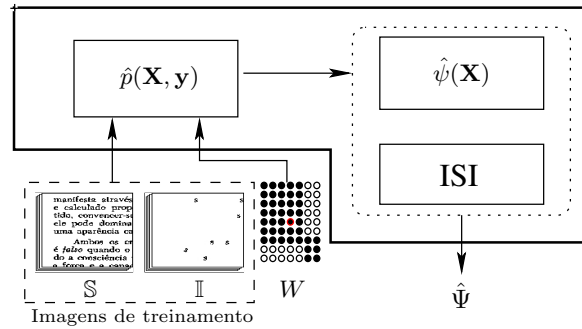
<sup>1</sup> $X_z$  é a configuração da janela observada no pixel  $z$  em  $S_i$ .

<sup>2</sup> $y_z$  é o valor observado através do centro de  $W$  no pixel  $z$  em  $I_i$



**Figura 2.3:** (a) Uma janela  $W$  ( $1 \times 3$ ) com o centro destacado em vermelho; (b)  $W$  centrada sobre um pixel  $z$  em uma imagem de entrada  $S_i$ ; (c)  $W$  sobre o correspondente de  $z$  na respectiva imagem ideal  $I_i$ ; (d) padrão  $X = 011$  observado em  $S_i$  através de  $W$ ; (e)  $y = 1$  valor do pixel  $z$  em  $I_i$  sob  $W$ ; (f) tabela resultante do passeio de  $W$  sobre  $(S_i, I_i)$ .

Estes três passos, utilizando o ISI na generalização, compõem o que é chamado neste trabalho, de **procedimento básico de treinamento** de um W-operador. Parte desse procedimento pode ser visto como um problema de classificação, onde o objetivo é classificar padrões como pertencentes ao núcleo ou não. Um esquema desse procedimento básico pode ser visualizado na Figura 2.4.



**Figura 2.4:** Esquema do procedimento básico de treinamento para projetar W-operadores com exemplos de parâmetros de entrada.

Na Figura 2.4, observa-se que  $W$  (círculos escuros) está disposta sobre uma matriz retangular. Essa representação facilita a visualização da janela, uma vez que a mesma nem sempre é retangular. No contexto desta pesquisa, essa matriz estabelece o **domínio** de uma janela, ou seja, os limites máximos de sua forma.

### 2.2.1 Influência da janela no projeto de W-operadores

A janela dada como entrada em um procedimento básico de treinamento tem grande influência na qualidade do operador resultante desse procedimento. Para descrever essa influência, será analisado um problema qualquer  $Q$  de transformação de imagens para o qual deseja-se projetar um W-operador. Para  $Q$  existe um operador ótimo ( $\Psi_{otm}$ ) e admite-se uma janela  $W_{otm}$  (desconhecida) associada a esse operador (Hirata, 2011).

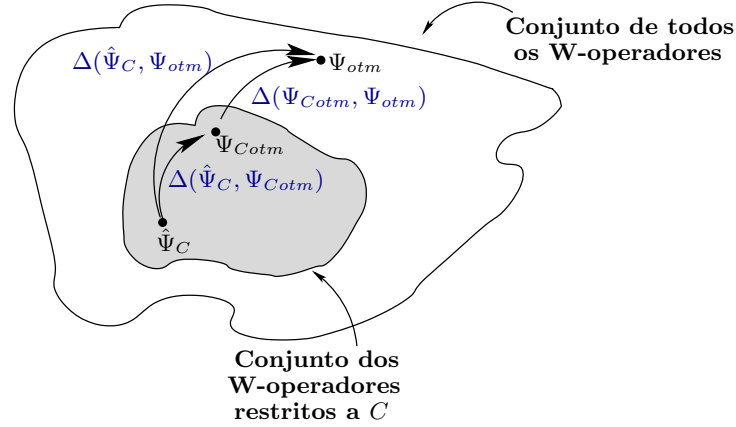
Seja  $W$  uma janela fornecida ao procedimento básico de treinamento e um conjunto preestabelecido de imagens de treinamento para  $Q$ . Quais as implicações de se usar  $W$  no projeto do W-operador?



$W$  estabelece um subespaço  $C$  dos  $W$ -operadores associados a  $W$ . O operador ótimo restrito a  $C$  é denotado por  $\Psi_{Cotm}$ . O procedimento básico fornece como resultado um operador estimado  $\hat{\Psi}_C$ . Ao se projetar  $\hat{\Psi}_C$  busca-se minimizar o erro desse operador em relação ao operador ótimo  $\Psi_{otm}$ . Esse erro pode ser escrito como:

$$\begin{aligned}\Delta(\hat{\Psi}_C, \Psi_{otm}) &= MAE(\hat{\Psi}_C) - MAE(\Psi_{otm}) \\ &= MAE(\hat{\Psi}_C) - MAE(\Psi_{Cotm}) + MAE(\Psi_{Cotm}) - MAE(\Psi_{otm}) \\ &= \Delta(\hat{\Psi}_C, \Psi_{Cotm}) + \Delta(\Psi_{Cotm}, \Psi_{otm}).\end{aligned}$$

O erro do operador projetado pode ser ilustrado como na Figura 2.5.



**Figura 2.5:** Esquema ilustrativo sobre o erro do operador projetado em relação ao operador ótimo ( $\Delta(\hat{\Psi}_C, \Psi_{otm})$ ).

Considerando todos os possíveis conjuntos de treinamento, o erro esperado de  $\hat{\Psi}_C$  relativo ao operador ótimo é:

$$E[\Delta(\hat{\Psi}_C, \Psi_{otm})] = \underbrace{E[\Delta(\hat{\Psi}_C, \Psi_{Cotm})]}_{e_1} + \underbrace{\Delta(\Psi_{Cotm}, \Psi_{otm})}_{e_2}. \quad (2.5)$$

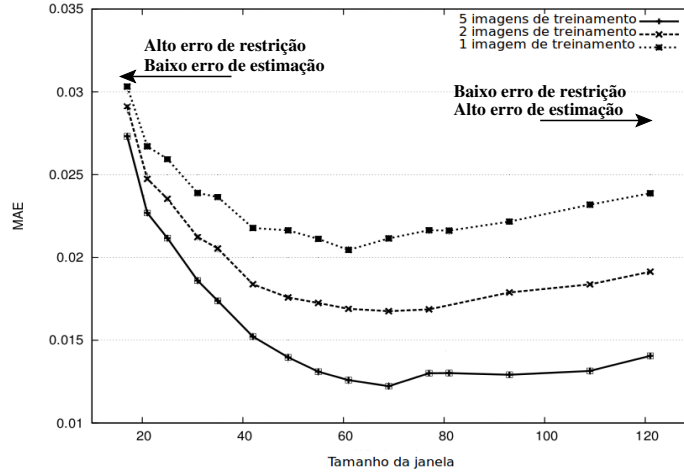
A Equação 2.5 é composta por dois termos:

- $e_1$ ) **Erro de estimação** restrito ao subespaço  $C$ , onde o crescimento desse subespaço implica aumento desse erro quando o conjunto de imagens de treinamento é fixo;
- $e_2$ ) **Erro de restrição** o qual tem seu valor elevado quando o subespaço  $C$  é muito restrito. Uma vez definido  $C$ , esse erro é uma constante (desconhecida).

Portanto, a escolha de  $W$  influencia na qualidade do operador estimado, ou seja, o número de pontos da janela afeta diretamente o erro do operador.

O gráfico da Figura 2.6 reflete a influência de  $W$  no valor do MAE. Quando a janela é pequena o MAE é grande, devido a restrição severa do espaço. Nesse caso a vizinhança estabelecida pela janela não é suficiente para capturar uma transformação adequada para  $Q$ . A medida que o número de pontos de  $W$  é incrementado o erro diminui até um mínimo, e então volta a aumentar (mais lentamente) com o crescimento da janela (aumenta  $e_1$ ). É possível observar na figura que independente do número de imagens de treinamento, o comportamento do MAE em relação a variação do tamanho da janela é similar.





**Figura 2.6:** MAE para diferentes tamanhos de janelas e conjuntos de treinamento. Fonte: (Hirata, 2009).

Uma maneira de diminuir o erro do operador é estabelecer uma janela grande (para garantir  $e_2$  pequeno) e incrementar o número de imagens de treinamento, diminuindo o erro de estimação  $e_1$ . Na prática, o número de imagens de treinamento é fixo e, portanto, definir a janela  $W$  implica fazer escolhas que representem o melhor compromisso entre o erro de estimação e o erro de restrição para um determinado número de imagens de treinamento (Hirata, 2011).

Uma abordagem que surgiu neste contexto é o **projeto dois níveis** de W-operadores (Hirata, 2009). A seção 2.3 apresenta essa abordagem.

## 2.3 Projeto dois níveis de W-operadores

O projeto dois níveis é um caso particular do projeto multi níveis (Hirata, 2009) que fundamenta-se na *stacked generalization*, uma técnica de combinação de classificadores baseada em múltiplos níveis de treinamento proposta por Wolpert (1992).

Em cada nível do projeto multi níveis vários operadores são construídos usando-se o procedimento básico de treinamento. Quando o projeto multi níveis possui apenas um nível, ele fica reduzido ao procedimento básico. Dessa forma, o procedimento básico de treinamento será denominado por **projeto um nível**.

A Figura 2.7 apresenta um exemplo de projeto dois níveis. Nesse esquema, as janelas  $W_1^{(1)}, W_2^{(1)}, \dots, W_n^{(1)}$  assim como o número de janelas são estabelecidos manualmente, de acordo com o conhecimento do projetista. Cada uma dessas janelas é submetida ao projeto um nível, utilizando imagens de treinamento pertencentes a  $\mathbb{S}^{(1)}, \mathbb{I}^{(1)}$ . O resultado são os operadores  $\hat{\Psi}_1^{(1)}, \hat{\Psi}_2^{(1)}, \dots, \hat{\Psi}_n^{(1)}$ . No segundo nível, esses operadores são aplicados a um grupo de imagens em  $\mathbb{S}^{(2)}$ , distintas daquelas em  $\mathbb{S}^{(1)}$  usadas na entrada do primeiro nível. As imagens resultantes dessas aplicações  $\hat{\Psi}_1^{(1)}(\mathbb{S}^{(2)}), \hat{\Psi}_2^{(1)}(\mathbb{S}^{(2)}), \dots, \hat{\Psi}_n^{(1)}(\mathbb{S}^{(2)})$ , e a janela de segundo nível são fornecidas como entrada para o treinamento do segundo nível. A janela do segundo nível é usada para examinar em paralelo todas as  $n$  saídas do primeiro nível. Dessa forma, o operador projetado em dois níveis (**operador dois níveis**) consegue usar uma vizinhança maior na imagem de entrada, mesmo que indiretamente, diminuindo o erro de restrição. Ao mesmo tempo, os operadores individuais do primeiro e segundo níveis são baseados em uma vizinhança menor, contribuindo para que o erro de estimação não aumente.

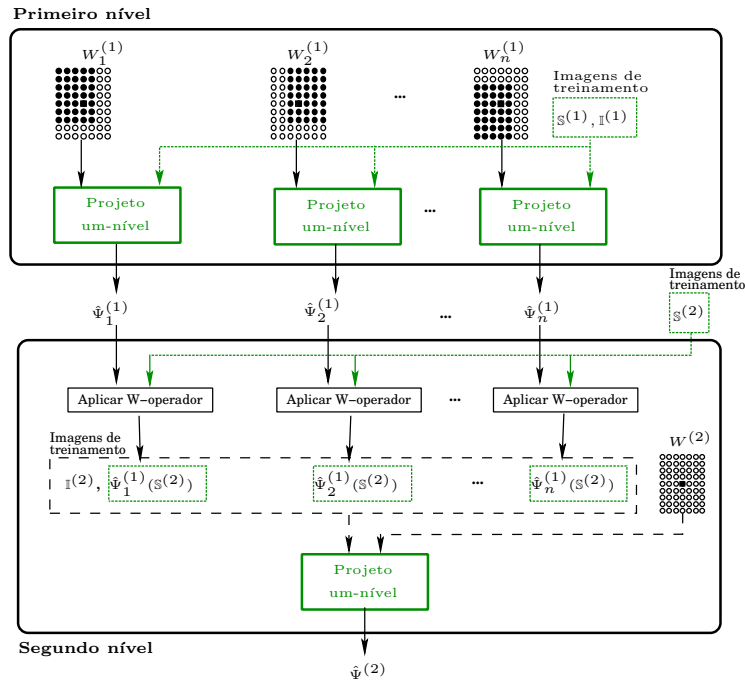


Figura 2.7: Exemplo de um projeto dois níveis para um  $W$ -operador.

O projeto dois níveis considerado nesta tese pode ser caracterizado apenas pelo conjunto das janelas dos operadores de primeiro nível, uma vez que será adotado para o segundo nível a janela de um ponto central.

### 2.3.1 Janelas e o tempo de processamento de um projeto dois níveis

A complexidade de tempo do projeto dois níveis é determinada pela complexidade de tempo do procedimento básico de treinamento, pois esse é aplicado para cada janela no primeiro nível e no treinamento do segundo nível. O procedimento básico inclui a execução do algoritmo de generalização ISI que é responsável por grande parte do processamento.

Não foi encontrada na literatura a análise de complexidade do ISI, visto que a mesma é de difícil definição (Tomita, 1996), mas sabe-se que ele precisa tratar instâncias do problema de cobertura de vértices (Hirata *et al.*, 2002) que não possui solução em tempo polinomial (Cormen, 2002) e sua solução com busca exaustiva pertence a classe assintótica  $\Theta(2^n)$ .

Sabe-se que o número de pontos da janela de entrada do projeto um nível determina o número de variáveis da função booleana a ser minimizada pelo ISI. Dessa forma, a janela influencia diretamente no tempo de execução do algoritmo.

Na dissertação de mestrado e tese de doutorado de Hirata (2000); Tomita (1996) e na publicação (Hirata *et al.*, 2002) são apresentados resultados de testes práticos, para algumas versões do ISI, que comprovam a relação entre número de variáveis e tempo de execução. Também é possível verificar em (Hirata, 2009) que quanto maior a janela, maior é o tempo de execução do projeto dois níveis. Para ilustrar o quanto o tamanho de uma janela pode influenciar no tempo de execução do projeto dois níveis, foi realizada uma pequena comparação entre dois testes relatados em (Hirata, 2009). Essa comparação está exposta na Tabela 2.1. Portanto, deve-se ter cautela na definição das janelas do projeto dois níveis.

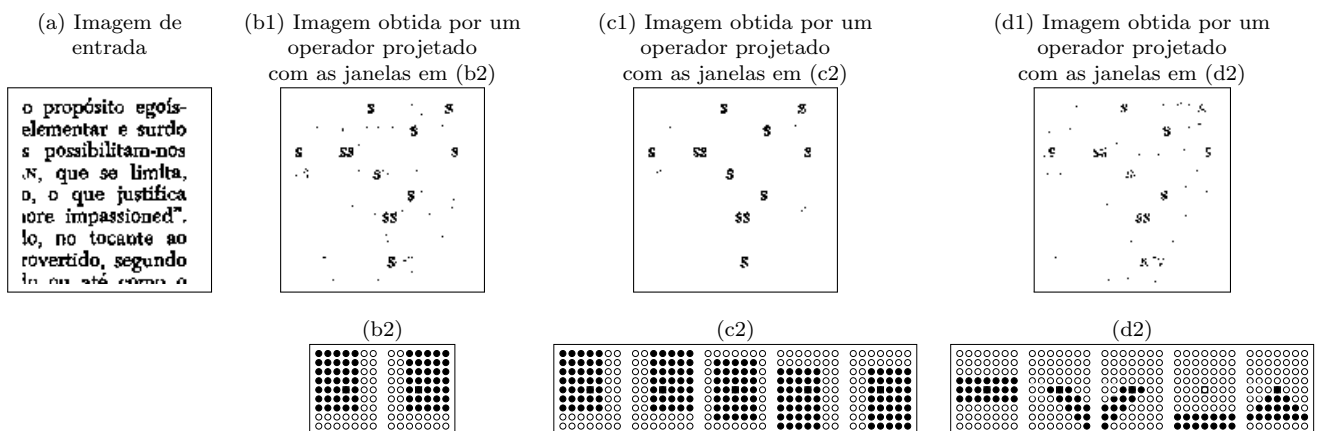
**Tabela 2.1:** Comparação dos tempos de processamento, em uma máquina AMD Athlon 64 X2 4200 2.2GHz com 3GB RAM, do projeto dois níveis de acordo com o número de pontos das janelas usadas no primeiro nível para dois conjuntos de dados. Extraído de (Hirata, 2009).

Dados	Total de imagens em $\mathbb{S}^{(1)}$	Total em $\mathbb{S}^{(2)}$	Total de pontos de cada janela do 1º nível	tempo de processamento
Segmentação do caractere <b>S</b> em imagens de texto	6	4	5 janelas com 35 pontos 5 janelas com 63 pontos	3.2 minutos 14.0 minutos
Segmentação de texto em imagens de páginas de revistas	3	2	5 janelas com 35 pontos 4 jan. com 49, 1 com 100 e 2 com 55	5.8 horas 17.0 horas

### 2.3.2 Janelas e a qualidade de um operador dois níveis

Os argumentos expostos na seção 2.2.1 sobre a influência de uma janela no projeto de W-operadores também são válidos para a escolha de combinação de janelas para o primeiro nível. Além disso, o número de janelas no primeiro nível e a diversidade de forma entre elas também afetam a qualidade do operador projetado. Essas questões podem ser observadas nos exemplos apresentados na Figura 2.8. A figura mostra resultados da aplicação de diferentes operadores sobre uma mesma imagem. O objetivo desses operadores é extrair o caractere **S** da imagem de entrada. Uma janela de um ponto central foi usada no segundo nível para o projeto destes operadores.

Na Figura 2.8(b1) a imagem é resultado da aplicação do operador projetado utilizando no primeiro nível as duas janelas em (b2). A imagem em (c1) foi obtida pela aplicação do operador projetado utilizando as janelas em (c2) no primeiro nível. Comparando as imagens em (b1) e (c1), verifica-se que a primeira é inferior a segunda, mostrando que o número de janelas pode influenciar na qualidade do operador projetado. A imagem resultante em (d1) comparada com a imagem em (c1), mostra que a forma das janelas também influencia no operador projetado.



**Figura 2.8:** Exemplos de imagens resultantes da aplicação de operadores dois níveis contruídos com diferentes conjuntos de janelas no primeiro nível para o problema de extração do caractere **S** em imagens de texto.

Portanto, a escolha de janelas para o projeto dois níveis envolve três momentos: definir o número de pontos que uma janela pode assumir; estabelecer a forma adequada das janelas para o problema em questão; e determinar qual o número de janelas do primeiro nível.

As próximas seções tratam de trabalhos e ideias que podem nortear o processo de definir uma combinação de janelas para o projeto dois níveis.

## 2.4 Elementos estruturantes e janelas

Como apresentado, uma janela tem a finalidade de delimitar os elementos estruturantes que são utilizados em uma transformação. Logo, trabalhos que envolvam a definição de elementos estruturantes podem nortear a construção de uma janela para um dado problema.

Segundo Soille (2003), muitas transformações morfológicas consistem em aplicar operadores elementares, caracterizados por EEs “básicos”, sobre imagens de interesse. Entende-se por “básicos” as aproximações digitais (no caso das imagens binárias) de figuras geométricas como: hexágono, diamante e quadrado, que podem ser usados para representar discos, utilizados, por exemplo, para recuperar imagens com ruídos sal/pimenta; conjunto de pontos representando segmentos de retas horizontais, verticais e diagonais, frequentemente usados para extrair estruturas alongadas; triângulos; dentre outros.

EEs básicos foram utilizados em vários trabalhos. Em (Béliz-Osorio *et al.*, 2011) discos são usados em uma metodologia para segmentação de células cancerígenas em imagens de citologia. Cruz, quadrado, losango, octógono e discos de diferentes raios foram usados na dilatação e erosão de imagens binárias de anéis, cruz, pontos e imagens arbitrarias em (Ji *et al.*, 1989). O openCV (Bradski, 2000) (biblioteca para visão computacional) permite definir os EEs quadrado, cruz e elipse. O seu elemento estruturante padrão para as funções de erosão e dilatação é um quadrado  $3 \times 3$ . A biblioteca de processamento de imagens do Matlab permite criar os seguintes EEs básicos: diamante (losango), retângulo, disco, octógono e linhas de diferentes comprimentos e ângulos (MAT, 2012).

Na maioria das vezes, os EEs são simples e escolhidos em função das propriedades de convexidade, não-convexidade, isotropia (vizinhos de um ponto são equidistantes) e anisotropia (Facon, 1996).

Algumas abordagens podem ser usadas para definir um conjunto de elementos estruturantes. Duas delas podem ser utilizadas para computar aberturas e fechamentos (Soille, 2003): *i*) uso de EEs com pontos conectados (sempre possuem vizinhos) e área (número de pontos) máxima definida; *ii*) dado um EE de entrada, o uso de subconjuntos conectados ou não desse EE.

Outra ideia que pode ser aproveitada para a criação de janelas vem do trabalho de Salembier (1992). Ele propõe um método para adaptar EEs a um determinado par de imagens entrada/ideal (imagens de treinamento). Essa metodologia sugere a criação de uma janela adaptativa, criada de acordo com características extraídas das imagens de treinamento. Acredita-se que, com isso, o operador projetado é mais preciso.

Todas as abordagens citadas anteriormente podem ser usadas para guiar a escolha do formato e tamanho de janelas no projeto de operadores. Baseando-se na estratégia adaptativa, a teoria da informação já foi utilizada na criação de janelas. A próxima seção apresenta dois trabalhos nessa área.

## 2.5 Teoria da informação e escolhas de janelas para o projeto de W-operadores

A teoria matemática da comunicação ou teoria da informação iniciou-se na década de 1940 com [Shannon \(1948\)](#) e seu objetivo era definir uma medida de informação que refletisse a novidade contida em uma certa mensagem. Uma das primeiras medidas baseada nessa ideia foi a **entropia**.

Seja  $X$  uma variável aleatória discreta cujo contradomínio é  $R_X$  e  $p$  a distribuição de probabilidade de  $X$ . Informalmente, pode-se dizer que a entropia é uma medida de incerteza sobre a variável aleatória  $X$ , e quanto maior o seu valor, maior a incerteza sobre  $X$ . A entropia é a base para outras medidas da teoria da informação e é definida como ([Cover e Thomas, 1991](#); [Shannon, 1948](#)):

$$H(X) = - \sum_{x \in R_X} p(x) \lg(p(x)), \quad (2.6)$$

em que  $\lg$  é a função logaritmo na base dois. Para  $p(x) = 0$  adota-se  $\lg(0) = 0$ .

Seja  $Y$  uma variável aleatória discreta diferente de  $X$  com contradomínio  $R_Y$ . Para medir a incerteza de  $Y$  na presença de todos os valores em  $X$ , utiliza-se a **entropia condicional**  $H(Y|X)$ , dada por ([Hall e Smith, 1999](#); [Shannon, 1948](#)):

$$H(Y|X) = - \sum_{x \in R_X} p(x) \sum_{y \in R_Y} p(y|x) \lg(p(y|x)) \quad (2.7)$$

$$H(Y|X) = - \sum_{x \in R_X} p(x) H(Y|x). \quad (2.8)$$

A entropia condicional foi utilizada por [Martins-Jr et al. \(2004, 2005\)](#) na construção de janelas para o procedimento básico de treinamento. Como a probabilidade é estimada, a entropia condicional sofreu uma correção que reflete os padrões não encontrados no conjunto de treinamento:

$$\hat{H}_{c_M}(y|X) = \frac{2^{|W|} - N}{2^{|W|} + totalO} + \sum_{i=1}^N \frac{(o_i + 1)}{2^{|W|} + totalO} \hat{H}(y|X_i), \quad (2.9)$$

em que  $y$  é um dos valores 0 ou 1 em uma imagem ideal,  $|W|$  é o número de pontos da janela considerada;  $o_i$  é o número de observações do padrão  $X_i$ ;  $N$  é o total de padrões observados; e  $totalO$  é o número total de exemplos coletados. A  $\hat{H}(y|X_i)$  dos padrões não observados é considerada a máxima, ou seja 1, dando origem a primeira parcela da soma.

A metodologia de [Martins-Jr et al. \(2004, 2005\)](#) pode ser descrita, de maneira geral, por passos tal que: no passo 1 procura-se, no domínio dado como entrada, uma janela de um ponto com menor entropia condicional corrigida  $\hat{H}_{c_M}(y|X)$ . No passo 2, é selecionada a janela de dois pontos, caso exista, obtida acrescentando-se um ponto na janela escolhida no passo 1, que possui menor entropia condicional corrigida. O processo repete-se até que em um passo  $P$  não se encontre janela com  $\hat{H}_{c_M}(y|X)$  menor que a do passo anterior, ou  $P$  tenha atingido o número de pontos do domínio.

Outra medida da teoria da informação é a **informação mútua**:

$$I(X;Y) = H(Y) - H(Y|X). \quad (2.10)$$

Ela mede o grau de dependência entre duas variáveis aleatórias e quanto menor for  $I(X; Y)$  mais independentes são as variáveis.

Segundo Frénay *et al.* (2013) a informação mútua é um dos critérios mais utilizados em seleção de características. Em seu trabalho, os autores analisam o comportamento do erro quadrático médio (MSE) e do MAE em função da entropia condicional, quando o erro de estimação tem uma distribuição gaussiana, laplaciana ou uniforme sobre problemas de regressão. São consideradas variáveis aleatórias contínuas. Nessas condições, o trabalho mostra que, escolher um subconjunto de característica que minimiza a entropia condicional, e conseqüentemente a informação mútua, também minimiza o MAE.

A **informação de interação** mede o grau de interação entre duas variáveis aleatórias, por exemplo  $X_1$  e  $X_2$ , para a determinação de uma terceira variável  $Y$ , e pode ser dada pela equação (Jakulin e Bratko, 2003):

$$I(Y; X_1; X_2) = I(Y; (X_1, X_2)) - (I(Y; X_1) + I(Y; X_2)), \quad (2.11)$$

em que  $I(Y; (X_1, X_2))$  é a informação mútua entre  $Y$  e o par  $(X_1, X_2)$  considerado conjuntamente.

Esta medida é usada por Santos (2010) na construção de janelas para o primeiro nível em um projeto dois níveis. Um algoritmo de minimização da magnitude da informação de interação foi definido e empregado como base para criar um conjunto de janelas. Essas são processadas e submetidas ao procedimento básico de treinamento. Os operadores resultantes desse procedimento são ordenados em ordem crescente segundo o critério de maximização da informação mútua condicional (*Conditional Mutual Information Maximization - CMIM*) (Fleuret e Guyon, 2004). Em seguida, o projeto dois níveis é executado para a primeira e segunda janelas da ordenação, depois para a primeira, segunda e terceira janelas, e assim sucessivamente. A janela de um ponto é utilizada no segundo nível. Os operadores dois níveis resultantes desses projetos são avaliados pelo princípio do comprimento mínimo de descrição (Minimum Description Length - MDL) (Grünwald, 2007) para a escolha de uma combinação de janelas para o primeiro nível em um projeto dois níveis. Os resultados apresentados ainda não foram suficientes para uma comparação mais definitiva com aqueles estabelecidos manualmente em (Hirata, 2009).

Uma observação sobre esta abordagem é que as janelas obtidas pela técnica, nos poucos experimentos realizados, não possuem uma forma geométrica específica. Essa característica pode ser observada na Figura 2.9 que ilustra alguns exemplos de janelas construídas para um problema de extração do caractere **S** em textos de livros.

Todos os trabalhos e técnicas apresentados nesta seção nortearam a definição de uma metodologia para a construção de janelas a serem selecionadas para um primeiro nível em projetos de W-operadores dois níveis.

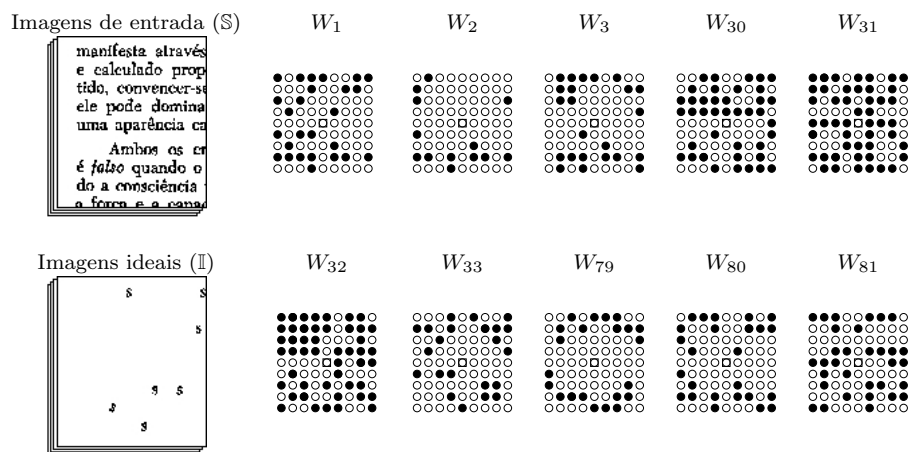


Figura 2.9: Exemplos de janelas dispostas sobre um domínio  $9 \times 9$  obtidas pela minimização da magnitude da informação de interação no problema de extração do caractere  $S$ .





## Capítulo 3

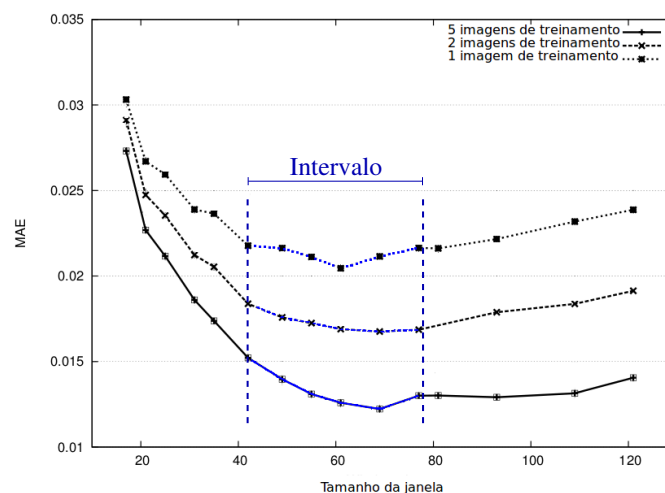
# Coleções de janelas

Este capítulo apresenta procedimentos para a criação de coleções de janelas que visam reduzir o espaço de busca de combinações de janelas a serem usadas no primeiro nível de um projeto dois níveis. Antes de abordar os procedimentos de criação, é apresentada uma metodologia para estimar, a partir de um conjunto de pares de imagens de entrada-saída, um intervalo de tamanho de janelas que pode ser utilizado na criação das coleções.

### 3.1 Estimação de um intervalo de tamanho de janelas

Conforme mencionado na introdução, observações empíricas indicam que combinações de operadores resultam em operadores melhores que os projetados individualmente. Apesar dessa melhora não ser garantida, pode-se afirmar que o desempenho do operador final será, ao menos equivalente ao do melhor operador individual projetado no primeiro nível da combinação. Portanto, escolher janelas "boas", aquelas que resultam em operadores com erro pequeno, é estrategicamente importante.

Já é conhecido do Capítulo 2 que o tamanho de uma janela é uma característica que influencia na qualidade de um operador individual. Quando estima-se o MAE sobre um conjunto de imagens de teste para diferentes janelas de tamanho crescente, considerando uma quantidade fixa de dados de treinamento, o resultado é uma curva como as exibidas na Figura 3.1.



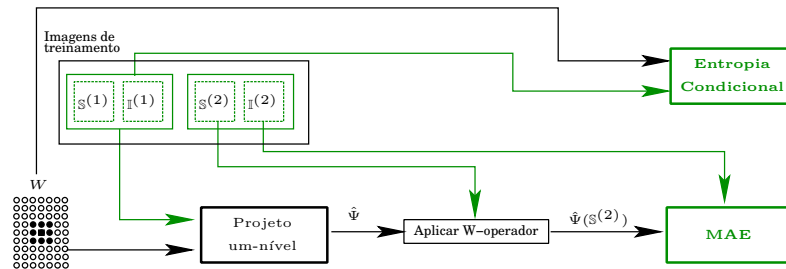
**Figura 3.1:** Curvas dos MAEs, sobre imagens de teste, de operadores projetados com janelas crescentes sobre conjuntos de treinamento de tamanhos diferentes e um intervalo de tamanho de janelas em que o menor erro está presente.

O ponto de mínimo das curvas do MAE de teste corresponde à janela ótima estimada. Vamos supor que o tamanho da janela ótima, dado um conjunto de treinamento, encontra-se em um intervalo em torno deste ponto de mínimo com alta probabilidade. Esse intervalo pode, então, ser utilizado para delimitar o tamanho das janelas a serem utilizadas no projeto dois níveis. Para calcular uma curva do MAE de teste, no entanto, é necessário treinar o operador e avaliar o MAE para cada uma das janelas na sequência de janelas crescentes.

Uma vez que o treinamento é um processo computacionalmente caro, este método torna-se inapropriado do ponto de vista prático. Essa observação sugere que um método prático para estimar o tamanho de janelas não deve depender de treinamento de operadores. Ou seja, o método deve ser capaz de estimar o MAE de teste, ou alguma medida equivalente, sem a necessidade de se treinar os operadores.

A proposta deste trabalho é utilizar a entropia condicional como medida em substituição ao MAE de teste, uma vez que ela pode ser vista, no escopo do projeto de W-operadores, como a medida de incerteza que se tem dos valores dos pixels em imagens de saída, dado que se conhece os valores dos padrões coletados mediante a translação de uma janela sobre imagens de entrada. Trata-se também de uma medida já utilizada em outros trabalhos relacionados a seleção ou criação de janelas ou ainda em substituição à minimização do MAE como mencionado no Capítulo 2.

A vantagem em se escolher a entropia condicional está no fato de que não é preciso treinar operadores. Ela é obtida a partir das estimativas  $p(X, y)$  calculadas sobre um conjunto de treinamento, que é de ordem linear no número de pixels das imagens desse conjunto, ao passo que o treinamento do operador é de ordem exponencial. Outra vantagem é que não é preciso um conjunto de validação para obter a entropia condicional. Essas vantagens podem ser observadas na Figura 3.2, onde são exibidos os pré-requisitos necessários para o cálculo do MAE e da entropia condicional.



**Figura 3.2:** Pré-requisitos para se determinar a entropia condicional e o MAE de um operador individual.

No próximo tópico é apresentada uma comparação da entropia condicional com o MAE, mostrando que o comportamento das duas medidas é similar no contexto de W-operadores e que uma versão da entropia condicional, para um conjunto de treinamento fixo, pode substituir o MAE na busca de um intervalo desejado.

### Entropia condicional versus MAE

Seja  $\mathbf{X}$  o processo que representa os padrões observados em um dado problema de imagens e  $\mathbf{y} = \{0, 1\}$  os possíveis valores dos pixels em uma imagem ideal. A entropia condicional pode ser escrita como:

$$H(\mathbf{y}|\mathbf{X}) = \sum_{X \in \mathbf{X}} p(X)H(\mathbf{y}|X) \quad (3.1)$$

$$= - \sum_{X \in \mathbf{X}} p(X)p(0|X)\lg(p(0|X)) + p(X)p(1|X)\lg(p(1|X))$$

pela lei de Bayes:

$$= - \sum_{X \in \mathbf{X}} p(X, 0)\lg(p(0|X)) + p(X, 1)\lg(p(1|X))$$

$$H(\mathbf{y}|\mathbf{X}) = \sum_{(\mathbf{X}, \mathbf{y})} p(\mathbf{X}, \mathbf{y})(-\lg(p(\mathbf{y}|\mathbf{X}))). \quad (3.2)$$

A equação do MAE, já conhecida:

$$MAE(\Psi) = \sum_{(\mathbf{X}, \mathbf{y})} p(\mathbf{X}, \mathbf{y})|\psi(\mathbf{X}) - \mathbf{y}|. \quad (3.3)$$

Assumindo que se conheça todas as probabilidades, para o respectivo operador, das imagens de treinamento e teste, pode-se realizar uma comparação entre a entropia condicional e o MAE teóricos e verificar como se comportam.

Fazendo uma análise superficial, quando o MAE teórico é pequeno significa que as probabilidades definem bem o operador, ou seja o nível de incerteza é baixo, logo a entropia também é baixa. Quando o MAE teórico é alto, as probabilidades deixam uma margem de incerteza alta e portanto entropia condicional alta. Assim, na teoria ambas as medidas se comportam de forma semelhante.

É necessário então investigar como as medidas se comportam na prática. Para isso, esta pesquisa verificou como a entropia condicional se comporta para janelas crescentes em relação ao MAE de teste, considerando que o conjunto de treinamento é fixo e, portanto, as probabilidades são estimadas. Nesse caso, um fator de correção ao cálculo da entropia condicional pode ser aplicado para os padrões que não aparecem no conjunto de treinamento.

Martins-Jr *et al.* (2004, 2005) propôs um fator de correção para a entropia condicional, já apresentado no Capítulo 2, e transcrito abaixo por conveniência:

$$\hat{H}_{cM}(y|X) = \frac{2^{|W|} - N}{2^{|W|} + totalO} + \sum_{i=1}^N \frac{(o_i + 1)}{2^{|W|} + totalO} \hat{H}(y|X_i), \quad (3.4)$$

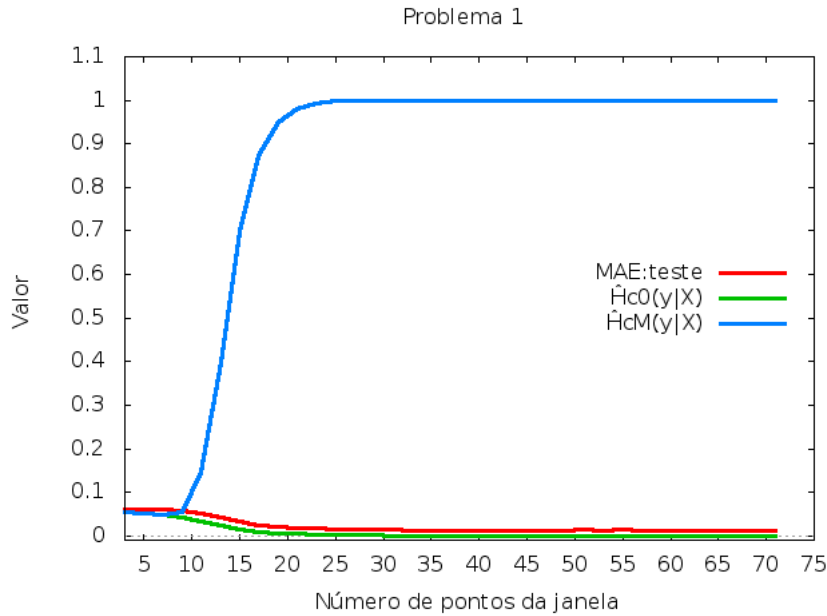
no qual  $|W|$  é o número de pontos da janela considerada;  $o_i$  é o número de observações do padrão  $X_i$ ;  $N$  é o total de padrões observados; e  $totalO$  é o número total de exemplos coletados. A  $\hat{H}(y|X_i)$  dos padrões não observados é considerada a máxima, ou seja 1, dando origem a primeira parcela da soma na equação.

A correção sugerida por Martins-Jr *et al.* (2004, 2005) leva a uma penalidade excessiva à medida que a janela cresce. Isso ocorre pois, nessa situação,  $2^{|W|}$  é muito maior que  $N$  e  $totalO$  e  $N \leq totalO$ . Considerando que  $totalO$  é fixo, a primeira parcela da  $\hat{H}_{cM}(y|X)$  assumirá valores próximos de 1.

Uma alternativa é assumir probabilidade zero para os padrões não observados no conjunto de treinamento. Isso é uma forma de generalização no qual o fator de correção é zero. Essa versão da entropia condicional é aqui chamada de **entropia condicional c0** e denotada por  $\hat{H}_{c0}(y|X)$ :

$$\hat{H}_{c0}(y|X) = \sum_{\hat{p}(X_i) > 0} \hat{p}(X_i) \hat{H}(y|X_i). \quad (3.5)$$

Na Figura 3.3 é exibido um gráfico mostrando o comportamento da  $\hat{H}_{cM}(y|X)$  e da  $\hat{H}_{c0}(y|X)$  calculadas sobre imagens de treinamento e para janelas de tamanho crescente, considerando o problema de extrair o caractere S em imagens de texto. Também é exibido o valor do MAE sobre imagens de teste. Para esse problema, a entropia condicional c0 se aproxima mais do MAE de teste do que a  $\hat{H}_{cM}(y|X)$ .



**Figura 3.3:** Gráfico exibindo o comportamento de  $\hat{H}_{cM}(y|X)$ ,  $\hat{H}_{c0}(y|X)$  e do MAE de teste em janelas crescentes para o problema de extrair o caractere S em imagens de texto.

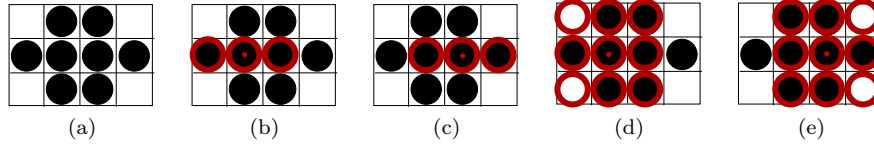
Analisando o comportamento da  $\hat{H}_{c0}(y|X)$  para janelas crescentes, observa-se que:

*Quando a janela é pequena* existe uma concentração maior de exemplos coletados para um mesmo padrão, o que pode ocasionar maior confusão. Isso ocorre pois parte dos exemplos pode estar associada a saída 0 e parte a saída 1, implicando alta entropia condicional.

*Quando a janela cresce*, o número de padrões diferentes coletado, no geral, também cresce, fazendo com que haja uma distribuição maior dos exemplos observados entre esses padrões, diminuindo a confusão e, conseqüentemente, a entropia condicional.

O crescimento da janela continua, até o ponto em que existe apenas um valor de saída para cada padrão observado, ou seja, não há confusão. Nesse ponto a  $\hat{H}_{c0}(y|X)$  é zero.

Essa análise sugere que a  $\hat{H}_{c0}(y|X)$  captura (em parte ou senão todo) o erro de restrição. Um pequeno exemplo dessa situação pode ser visualizado na Figura 3.4.



**Figura 3.4:** (a) Imagem original; (b) e (c) Janela de 3 pontos, em vermelho, sobre a imagem original onde o pixel de análise está sob o ponto central da janela; (d) e (e) Janela de 9 pontos onde o pixel de análise está sob o ponto central da janela.

Na Figura 3.4(a) uma imagem de entrada é exibida. Em (b) e (c) duas coletas de exemplos na imagem sob uma janela de três pontos são apresentadas. As duas observações são atribuídas ao mesmo padrão (111). Nas coletas em (d) e (e), com uma janela de nove pontos, os pixels que contribuíram para um mesmo padrão em (b) e (c), agora contribuem para padrões diferentes.

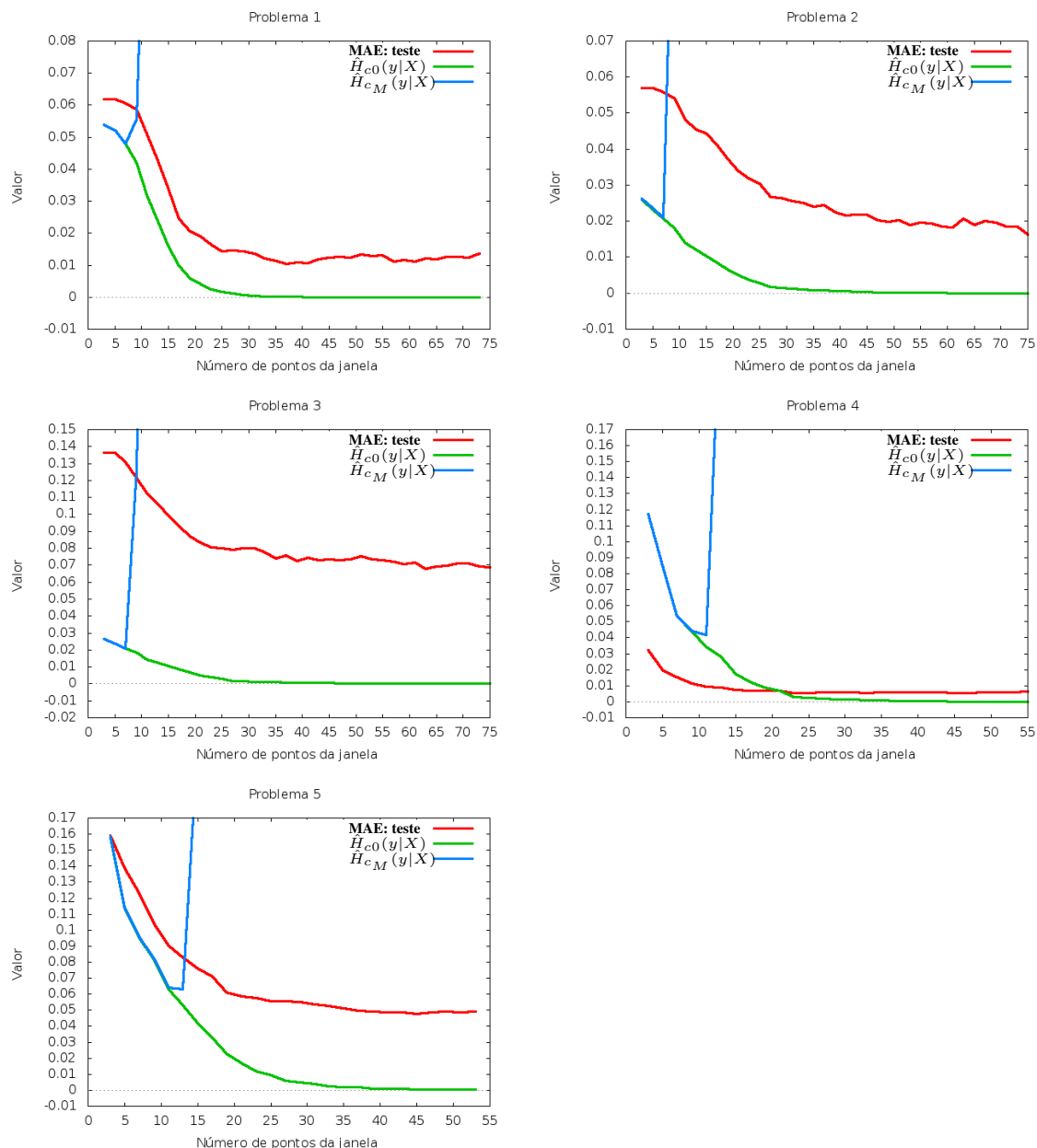
### Comparação empírica

Foi realizada uma comparação empírica do MAE de teste versus a  $\hat{H}_{c_M}(y|X)$  e  $\hat{H}_{c_0}(y|X)$  em relação ao tamanho de janelas crescentes. As entropias condicionais foram calculadas para imagens de treinamento. O MAE das imagens de teste foi usado, pois o que queremos é uma medida que possa substituí-lo. Cinco problemas distintos foram utilizados na comparação. Na Figura 3.5 são exibidos os cinco gráficos do MAE de teste versus a  $\hat{H}_{c_M}(y|X)$  e versus a  $\hat{H}_{c_0}(y|X)$ .

Verifica-se que a  $\hat{H}_{c_M}(y|X)$  cresce rapidamente com o crescimento da janela. Já a  $\hat{H}_{c_0}(y|X)$ , embora tenha sido calculada sobre imagens de treinamento, o seu comportamento é semelhante ao MAE de teste nos problemas utilizados, pelo menos até o ponto de mínimo das curvas, ainda que o MAE de teste apresente oscilações devido o seu cálculo usar imagens diferentes daquelas do treinamento. Essas medidas geram curvas decrescentes conforme o aumento do tamanho da janela até que a  $\hat{H}_{c_0}(y|X)$  chega a zero. Nesse ponto, o tamanho da janela para as duas medidas é próximo. Geralmente os valores de MAE mínimo ocorreram depois do mínimo da entropia condicional  $c_0$ .

Nota-se que, no geral, a entropia condicional  $c_0$  subestima o MAE, mas a similaridade de comportamento entre as medidas permite que se possa usar o ponto de inflexão da curva da  $\hat{H}_{c_0}(y|X)$  como um ponto de referência para estabelecer um intervalo para o tamanho de janelas. Portanto, em vez de determinar o ponto de mínimo na curva do MAE, é proposto estabelecer um intervalo em torno do ponto de inflexão da curva da  $\hat{H}_{c_0}(y|X)$ .

É importante destacar que essa abordagem faz sentido no cenário considerado nesta tese, em que se tem inicialmente um número de exemplos coletados bem maior que o número de padrões observados, permitindo que se obtenha a curva decrescente à medida que a janela cresce. Nesse contexto, as estimativas referentes aos padrões de maior probabilidade provavelmente estão próximas do valor correto; as estimativas mais incertas são referentes aos padrões menos frequentes, e por isso o impacto no erro total é também menor. Em situações com poucos exemplos, essa abordagem dificilmente será útil.

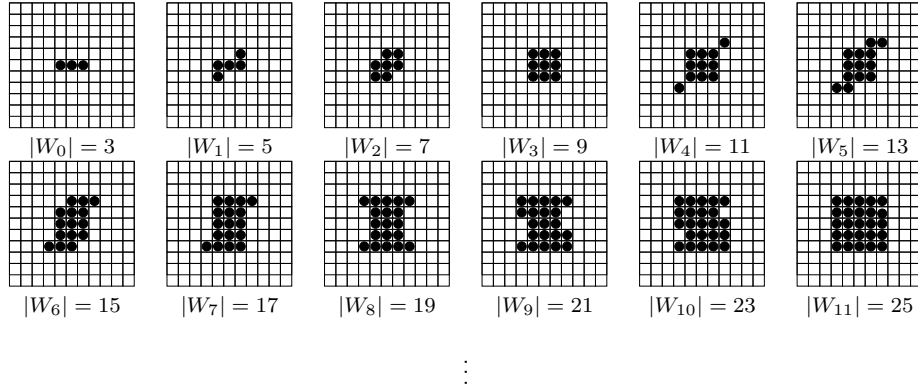


**Figura 3.5:** Gráficos exibindo o comportamento do MAE de teste  $\times \hat{H}_{c0}(y|X) \times \hat{H}_{cM}(y|X)$ , em relação ao tamanho de janelas crescentes para cinco problemas diferentes.

### Procedimento para estimar um intervalo de tamanho de janelas

Sejam  $W_0 \subseteq W_1 \subseteq W_2 \subseteq \dots \subseteq W_j$ ,  $j \in \mathbb{N}^*$ , janelas organizadas crescentemente pelo seu tamanho tal que:  $|W_0| = 3$ ;  $|W_1| = 5$ ;  $|W_2| = 7$ ;  $\dots$ ;  $|W_j| = |W_{j-1}| + 2$ . Exemplos de janelas crescentes e seus tamanhos são apresentados na Figura 3.6.

Considere  $\hat{H}_{c0}(y|X)_0, \hat{H}_{c0}(y|X)_1, \hat{H}_{c0}(y|X)_2, \dots, \hat{H}_{c0}(y|X)_j$  as entropias condicionais c0 das janelas, calculadas sobre um dado conjunto de treinamento.



**Figura 3.6:** Exemplos de janelas crescentes.

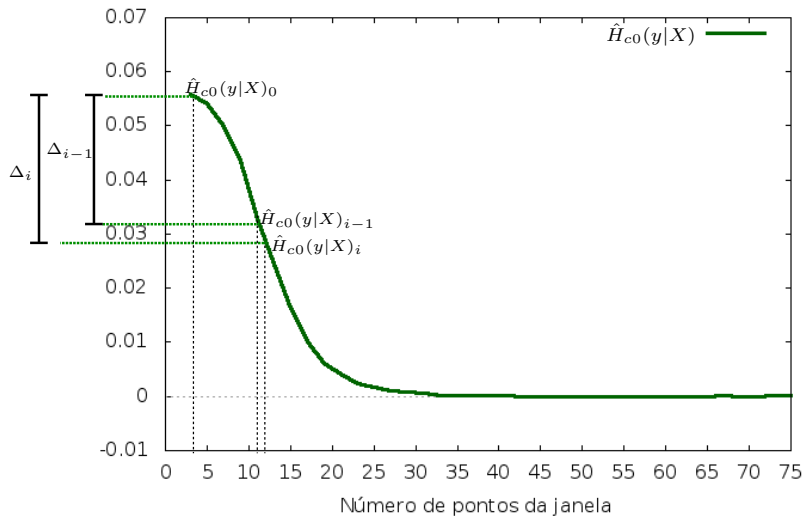
O ponto de inflexão é determinado analisando-se a razão entre o decréscimo relativo de  $\hat{H}_{c0}$  entre duas janelas consecutivas na sequência de janelas crescentes. Essa razão sinaliza como está o decréscimo da curva da  $\hat{H}_{c0}(y|X)$ . O decréscimo relativo é definido por:

$$\Delta_i = \hat{H}_{c0}(y|X)_0 - \hat{H}_{c0}(y|X)_i, \quad i = 1, 2, 3, \dots, j. \quad (3.6)$$

A razão entre decréscimos relativos sucessivos é então definida por:

$$R_i = \frac{\Delta_i}{\Delta_{i-1}}, \quad i = 2, 3, \dots, j. \quad (3.7)$$

Inicialmente, a razão é maior que 1, pois  $\Delta_i > \Delta_{i-1}$  uma vez que a entropia condicional  $c0$  está decrescendo. A medida em que  $i$  aumenta, a curva da  $\hat{H}_{c0}(y|X)$  vai se estabilizando e a razão fica próxima de 1. Quando essa razão é 1 significa que não houve decréscimo e a entropia condicional  $c0$  atingiu um valor constante. Essa análise pode ser observada no exemplo apresentado na Figura 3.7.



**Figura 3.7:** Exemplo de decréscimo relativo  $\Delta_i$  e  $\Delta_{i-1}$ .

O objetivo é determinar o ponto onde a curva da entropia condicional  $c0$  começa a se estabilizar, ou seja onde  $R_i$  é próxima de 1. Seja  $\omega$  o tamanho de  $W_i$ , em que  $R_i$  é próxima de 1. Esse valor é utilizado para estimar o intervalo procurado, definido por:

$$I_W = [\omega_{min}, \omega_{max}], \quad (3.8)$$

no qual  $\omega_{min} = \lfloor \omega * \alpha \rfloor$  e  $\omega_{max} = \lceil \omega * \beta \rceil$ , com  $\alpha < \beta$ .

Supondo que as boas janelas possuam tamanhos próximos ao do ponto de estabilização da curva do MAE e que a curva da  $\hat{H}_{c0}$  captura de forma aproximada o comportamento dessa curva,  $\alpha$  e  $\beta$  podem ser definidos de forma a tratar adequadamente a diferença entre elas. Por exemplo, se adotarmos  $\alpha = 0.9$  e  $\beta = 1.5$ , isso significa que o centro do intervalo será  $1.2 * \omega$ . Uma escolha dessas leva em consideração o fato da entropia condicional  $c0$  subestimar o MAE e que o ponto de estabilização do MAE está em geral à direita de  $\omega$ .

O  $\omega$  utilizado para estabelecer o intervalo é obtido, por definição, quando  $R_i < 1,005$ . Um limite superior MAX para o tamanho da janela também é estabelecido.

O algoritmo 1 descreve os passos para encontrar  $\omega$ , em que as imagens de treinamento são usadas no cálculo das entropias condicionais  $c0$ .

---

**Algoritmo 1:** Determina  $\omega$ .

---

**Entrada:** Imagens de treinamento, MAX.

**Saída:**  $\omega$ .

```

1  Constrói  $W_0$  com 3 pontos ;
2  Constrói  $W_1$  com 5 pontos ;
3   $\Delta_1 \leftarrow \hat{H}_{c0}(y|X)_0 - \hat{H}_{c0}(y|X)_1$  ;
4   $i \leftarrow 1$ ;
5  repita
6      |            $i \leftarrow i + 1$ ;
7      |           Constrói  $W_i$  ;
8      |            $\omega \leftarrow$  tamanho de  $W_i$  ;
9      |            $\Delta_i \leftarrow \hat{H}_{c0}(y|X)_0 - \hat{H}_{c0}(y|X)_i$  ;
10     |            $R_i \leftarrow \Delta_i / \Delta_{i-1}$  ;
11 até  $R_i < 1,005$  OU  $\omega \geq MAX$ ;
12 devolva  $\omega$  ;
```

---

► **Consumo de tempo de *Determina*  $\omega$**

Seja  $m$  o número total de pixels das imagens no conjunto de treinamento. A classe assintótica a qual o consumo de tempo do algoritmo pertence é definida pela linha 9, devido ao cálculo da entropia condicional  $c0$ . O consumo de tempo da entropia condicional  $c0$  pertence a  $\Theta(\omega * m)$ . Como MAX é o limite superior para  $\omega$  e a execução da linha 9 também é limitada superiormente por MAX, o consumo de tempo do algoritmo pertence a  $O(MAX^2 * m)$ . Como  $MAX^2$  é uma constante e geralmente muito menor que  $m$ , o consumo de tempo pertence a  $O(m)$ .

## 3.2 Criação de janelas

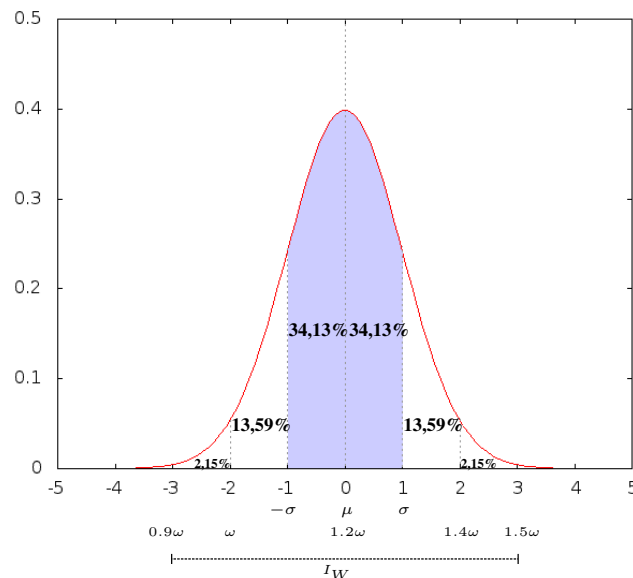
Nesta seção são descritas propostas de procedimentos para criação de coleções de janelas. Todas as janelas a serem criadas são restritas a um domínio de janelas, de dimensões  $l$  (linhas)  $\times$   $c$  (colunas) e tamanho máximo  $\omega_{max}$ .



### Tamanho de uma janela

No processo de criação das janelas, o tamanho de uma janela é sorteado de um intervalo  $[\omega_{min}, \omega_{max}]$ . Considerando a possibilidade de se utilizar um intervalo calculado como descrito na seção anterior, é definida uma distribuição aproximadamente normal sobre  $I_W = [\omega_{min}, \omega_{max}]$ . A ideia é que tamanhos mais próximos ao ponto médio desse intervalo tenham maior probabilidade de serem sorteados.

A vantagem em utilizar a distribuição normal padronizada é que a área sob sua distribuição é conhecida matematicamente. Essa área corresponde à probabilidade de ocorrência de um valor no eixo das abcissas. Portanto, considerando esta distribuição para o intervalo  $I_W$ , em que o ponto médio de  $I_W$  coincide com a média  $\mu$  da distribuição, a probabilidade de que um tamanho esteja entre  $1.1\omega$  e  $1.3\omega$  é de 68,26%; e a probabilidade de um tamanho ser sorteado próximo à fronteira do intervalo é de 4,3%. Existem 99,74% de chances do tamanho estar no intervalo. Diante do exposto, existe uma garantia razoável que os tamanhos sorteados ficarão dentro do intervalo e próximos ao seu ponto médio. Todo este discurso pode ser visualizado na Figura 3.8.



**Figura 3.8:** Distribuição normal padronizada e a distribuição de  $I_W$ .

Muitas linguagens de programação disponibilizam um procedimento, baseado no método polar de [Box e Muller \(1958\)](#) e descrito em [\(Knuth, 1981\)](#), que retorna um número aleatório sobre uma distribuição normal padronizada. Pode-se aproveitar esse fato e converter o número sorteado para o intervalo desejado, conforme o [Algoritmo 2](#).

---

#### Algoritmo 2: Sorteia tamanho de Janela

---

**Entrada:**  $\omega, \alpha, \beta$ .

**Saída:** Tamanho de uma janela

- 1  $media \leftarrow (\frac{\alpha-\beta}{2} + \beta) * \omega$  ;
  - 2  $desvio \leftarrow \frac{(\alpha-\beta)}{6} * \omega$  ;
  - 3  $numSorteado \leftarrow$  número aleatório sobre uma distribuição normal padronizada ;
  - 4  $tamanho = \lceil numSorteado * desvio + media \rceil$  ;
  - 5 **retorna** tamanho ;
-

## Janelas

Foram definidas três abordagens para a construção de coleções de janelas (CJ) de acordo com alguns critérios que estabelecem a forma das janelas.

- a) **CJI** é uma coleção formada pelas janelas obtidas pelo algoritmo de minimização da magnitude da informação de interação (será aqui identificado por *MMII*), proposto por Santos (2010). As janelas construídas pelo algoritmo não possuem uma forma específica.

Quando o projetista não fornece a dimensão do **domínio** ( $l$ =linhas e  $c$ =colunas), é assumido que  $l = c = \lceil \sqrt{\omega_{max}} \rceil + 2$  se  $\lceil \sqrt{\omega_{max}} \rceil$  for ímpar ou  $l = c = \lceil \sqrt{\omega_{max}} \rceil + 1$  se for par. O número de janelas da coleção é definido pelo algoritmo de Santos (2010) como  $l * c$ .

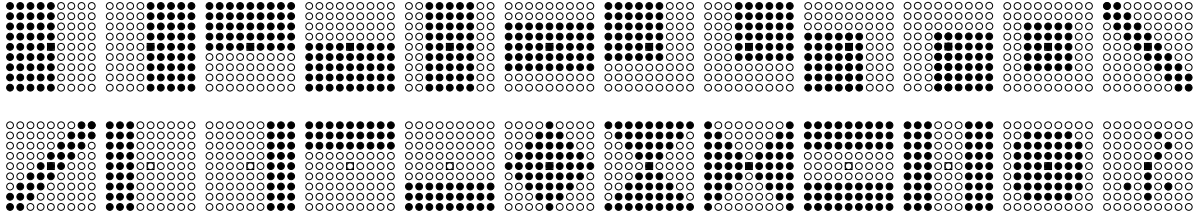
- b) **CJB** é uma coleção de **janelas básicas** definida fundamentando-se nos trabalhos sobre elementos estruturantes (EEs) apresentados no Capítulo 2 e nas janelas usadas por Hirata (2009). Foram estabelecidos 24 procedimentos para construção de janelas básicas tais como segmentos de retas, retângulos e outros objetos, criadas mediante algoritmos simples.

Dois coleções de janelas básicas são definidas:

**CJB<sub>f</sub>**: esta versão foi a primeira elaborada por esta pesquisa. Nela, o domínio é definido pelo usuário e os pontos das janelas não são obtidos aleatoriamente de um intervalo, mas são determinados de acordo com a forma predefinida de cada janela. A janela 24 desta coleção é criada pelo algoritmo proposto em (Martins-Jr *et al.*, 2004), mencionado no Capítulo 2. Na Tabela 3.1 estão as descrições gerais dos procedimentos para a criação das janelas e na Figura 3.9 são apresentados exemplos das 24 janelas.

**Tabela 3.1:** Descrição geral das janelas que compõem uma CJB<sub>f</sub>.

$W_1$ : janela vertical ocupando o lado esquerdo do domínio até o seu centro.	$W_{13}$ : janela diagonal secundária do domínio com espessura dada pelo maior inteiro ímpar menor ou igual a $(\lceil \frac{c}{3} \rceil + 1)$ .
$W_2$ : janela vertical ocupando o lado direito do domínio até o seu centro.	$W_{14}$ : janela vertical ocupando o lado esquerdo do domínio com espessura de $\lfloor \frac{c}{3} \rfloor$ .
$W_3$ : janela horizontal ocupando o lado superior do domínio até o seu centro.	$W_{15}$ : janela vertical ocupando o lado direito do domínio com espessura de $\lfloor \frac{c}{3} \rfloor$ .
$W_4$ : janela horizontal ocupando o lado inferior do domínio até o seu centro.	$W_{16}$ : janela horizontal ocupando o lado superior do domínio com espessura de $\lfloor \frac{l}{3} \rfloor$ .
$W_5$ : janela vertical central. Sua espessura é dada por $2\lfloor \frac{c}{4} \rfloor + 1$ .	$W_{17}$ : janela horizontal ocupando o lado inferior do domínio com espessura de $\lfloor \frac{l}{3} \rfloor$ .
$W_6$ : janela horizontal central. Sua espessura é dada por $2\lfloor \frac{l}{4} \rfloor + 1$ .	$W_{18}$ : janela losango centrada no domínio.
$W_7$ : janela ocupando o quadrante esquerdo superior do domínio.	$W_{19}$ : janela ampulheta horizontal centrada no domínio.
$W_8$ : janela ocupando o quadrante direito superior do domínio.	$W_{20}$ : janela ampulheta vertical centrada no domínio.
$W_9$ : janela ocupando o quadrante esquerdo inferior do domínio.	$W_{21}$ : janela horizontal superior e inferior com espessura de $\lfloor \frac{l}{3} \rfloor$ .
$W_{10}$ : janela ocupando o quadrante direito inferior do domínio.	$W_{22}$ : janela vertical esquerda e direita com espessura de $\lfloor \frac{c}{3} \rfloor$ .
$W_{11}$ : janela retangular ocupando o centro do domínio com base igual a $2\lfloor \frac{c}{4} \rfloor + 1$ e altura igual a $2\lfloor \frac{l}{4} \rfloor + 1$ .	$W_{23}$ : janela é uma aproximação de um octógono circunscrito em um quadrado de lado igual a $\min(l, c) - 2$ e com centro igual ao do domínio.
$W_{12}$ : janela diagonal principal do domínio com espessura dada pelo maior inteiro ímpar menor ou igual a $(\lceil \frac{c}{3} \rceil + 1)$ .	$W_{24}$ : janela obtida pela minimização da entropia condicional (Martins-Jr, 2004).



**Figura 3.9:**  $CJB_f$  sobre o domínio  $9 \times 9$  e a janela (última janela inferior da direita) obtida pela minimização da entropia condicional de (Martins-Jr et al., 2004) para o problema de extração do caractere  $S$  em imagens de texto.

$CJB_a$ : o número de pontos de cada janela, nomeado aqui por  $np$ , é definido pelo Algoritmo 2. Aqui, algumas janelas da  $CJB_f$  foram substituídas por outras. Por exemplo, a janela fornecida pelo algoritmo de (Martins-Jr et al., 2004) foi retirada, pois sempre apresentava uma janela com poucos pontos e em experimentos preliminares nunca era escolhida para uma combinação de primeiro nível. Além disso, algumas janelas que eram subjanelas de outras também foram substituídas, uma vez que o número de pontos da janela pode ser sorteado. Essa substituição também aumenta a diversidade da coleção. Na Tabela 3.2 estão as descrições gerais dos procedimentos para a construção das janelas.

**Tabela 3.2:** Descrição geral das janelas que compõem uma  $CJB_a$ .

$W_1$ : janela vertical ocupando o lado esquerdo do domínio.	$W_{13}$ : janela diagonal secundária do domínio.
$W_2$ : janela vertical ocupando o lado direito do domínio.	$W_{14}$ : janela triângulo com o ângulo reto no canto superior esquerdo do domínio.
$W_3$ : janela horizontal ocupando o lado superior do domínio.	$W_{15}$ : janela triângulo com o ângulo reto no canto superior direito do domínio.
$W_4$ : janela horizontal ocupando o lado inferior do domínio.	$W_{16}$ : janela triângulo com o ângulo reto no canto inferior esquerdo do domínio.
$W_5$ : janela vertical central (coluna).	$W_{17}$ : janela triângulo com o ângulo reto no canto inferior direito do domínio.
$W_6$ : janela horizontal central (linha).	$W_{18}$ : janela losango centrada no domínio.
$W_7$ : janela ocupando o quadrante esquerdo superior do domínio.	$W_{19}$ : janela ampulheta horizontal centrada no domínio.
$W_8$ : janela ocupando o quadrante direito superior do domínio.	$W_{20}$ : janela ampulheta vertical centrada no domínio.
$W_9$ : janela ocupando o quadrante esquerdo inferior do domínio.	$W_{21}$ : janela cruz centrada no domínio.
$W_{10}$ : janela ocupando o quadrante direito inferior do domínio.	$W_{22}$ : janela anel centrada no domínio. O lado externo do anel é aproximadamente duas vezes o lado interno do anel.
$W_{11}$ : janela retangular ocupando o centro do domínio.	$W_{23}$ : janela definida com pontos aleatórios sobre o domínio.
$W_{12}$ : janela diagonal principal do domínio.	$W_{24}$ : janela definida com pontos aleatórios sobre o domínio.

Para a construção desta coleção, o domínio pode ser dado pelo usuário ou pode ser automático. Quando o domínio automático é selecionado, para cada janela é calculado um domínio de maneira que permita a construção da forma definida na Tabela 3.2. Os procedimentos para a determinação do domínio de cada janela estão no Apêndice B.

Quando o usuário fornece um domínio  $l \times c$  e não é suficiente para construir alguma janela da Tabela 3.2, ela é criada por procedimento padrão para aquela forma, onde o domínio do usuário é mantido, mas o tamanho sorteado é abandonado, pois o domínio não o comporta. Para determinar se o domínio do usuário é suficiente ou não, foi definido que  $l * c \geq 1.7 * np$ . Os procedimentos para a construção das janelas padrão estão no Apêndice B.

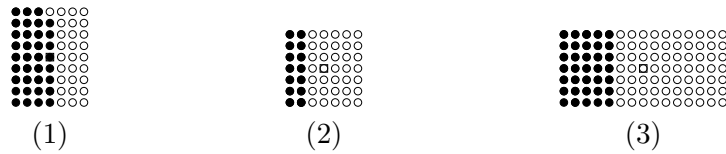
Para exemplificar a construção de uma janela em  $CJB_a$ , considere a construção de  $W_1$  (janela vertical à esquerda do domínio). Suponha que  $np = 35$ .

A seguir são relatadas as três possibilidades:

1) O usuário escolheu domínio automático. Para a janela vertical automática à esquerda do domínio, é definido que o lado maior da janela é duas vezes o seu lado menor. O lado maior define o número de linhas do domínio e o número de colunas é três vezes o lado menor da janela, sempre tomando números ímpares. Para o exemplo, em que o número de pontos sorteado foi 35, o lado menor é dado por  $\lceil \sqrt{\frac{35}{2}} \rceil = 5$  e o lado maior por  $\lceil \frac{35}{\text{lado menor}} \rceil = 7$ . A janela é como a da Figura 3.10(3).

2) O usuário forneceu o domínio  $l = 9$  e  $c = 7$ . Nesse caso  $l * c = 63 > 59,5 = 1,7 * 35$ . O domínio é suficiente para a criação da janela. Os 35 pontos são distribuídos da esquerda para a direita, preenchendo as colunas de baixo para cima, como definido no respectivo procedimento de construção da janela. A janela é a exibida na Figura 3.10(1).

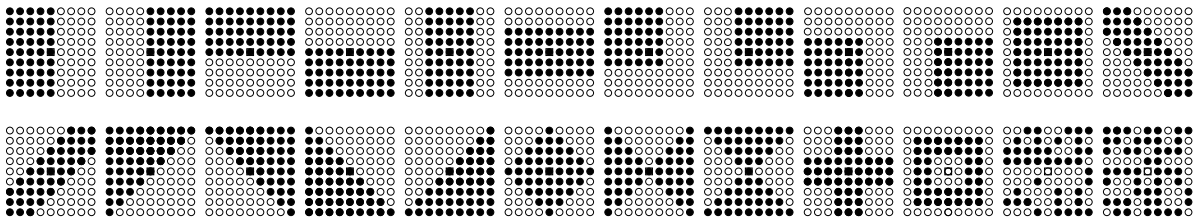
3) O usuário forneceu  $l = 7$  e  $c = 7$ . Como  $l * c = 49 < 59,5$ , agora é assumida uma janela padrão como a da Figura 3.10(2), em que  $np$  é abandonado.



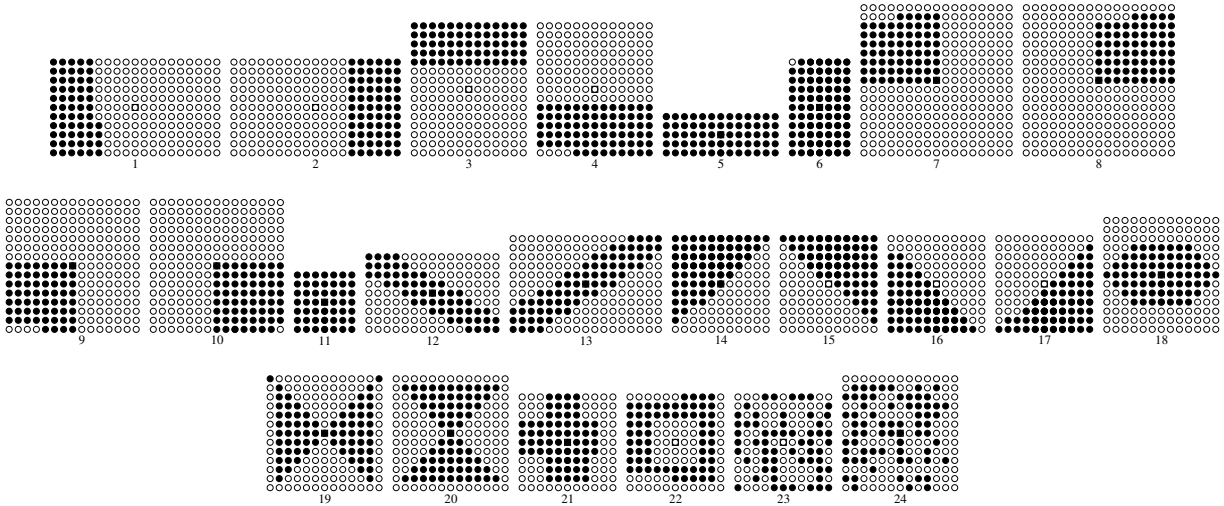
**Figura 3.10:** Exemplos de janelas verticais para  $CJB_a$ : (1) domínio definido pelo usuário com tamanho da janela aleatório, (2) domínio definido pelo usuário com janela padrão e (3) domínio automático com tamanho da janela aleatória.

Para garantir a simetria do domínio em relação a origem,  $l$  e  $c$  são considerados valores ímpares. Também não são permitidas dimensões menores que  $5 \times 5$ . Nesse caso, o projeto um nível resolve o problema.

Exemplos de janelas da  $CJB_a$  com domínio definido em  $l = 9$  e  $c = 9$  e pontos sorteados em  $I_W$  podem ser observados nas Figura 3.11. Na Figura 3.12 estão exemplos de janelas para domínio aleatório.



**Figura 3.11:**  $CJB_a$  com domínio do usuário  $9 \times 9$  e tamanho das janelas sorteado em  $I_W$ .



**Figura 3.12:**  $CJB_a$  com o tamanho de janelas escolhido aleatoriamente em  $I_W$  e domínio automático para um problema de extração de linhas pontilhadas em diagramas funcionais (Hirata, 2009).

c) **CJBI** é uma coleção que é a união das janelas obtidas em CJB e CJI.

O procedimento para construir uma CJ pode ser resumido no Algoritmo 3.

A coleção é construída de acordo com o tipo de CJ passado como parâmetro ao procedimento. Para cálculo de  $I_W$  ou da coleção  $CJI$ , são utilizadas imagens de treinamento de entrada  $\mathbb{S}^{(1)}$  e ideais  $\mathbb{I}^{(1)}$ . A função *mod* calcula o resto da divisão por 2.

$CJB_a$  e CJI podem ser categorizados como coleções adaptativas, pois ambas dependem de um conjunto de imagens de treinamento de um dado problema. Consequentemente qualquer versão de CJBI também é adaptativa.

#### ► Consumo de tempo de *Coleção de Janelas*

O consumo de tempo para encontrar  $\omega$  pertence a  $O(m)$  como descrito na seção 3.1. Lembrando que  $m$  é o total de pixels nas imagens de treinamento.

O tempo para construir uma coleção de janelas básicas é dado pela soma dos tempos para construir cada janela da coleção. Seja  $\omega_{max}$  a maior janela que pode ser construída por um procedimento, o tempo pertence a  $O(\omega_{max})$ , que é uma constante.

A coleção que requer mais tempo para a sua construção é a CJI e por conseguinte sua união com alguma versão da CJB. O consumo de tempo para construir CJI pertence a  $O(MAX\_ITER * L)$  (Santos, 2010), onde  $L$  é o número de pares  $(X, y)$ . No pior caso, existirá um par para cada pixel das imagens do conjunto de treinamento. Assim, podemos redefinir  $L$  como  $m$ .  $MAX\_ITER$  é uma constante que limita o número de iterações do procedimento. Portanto a construção de uma CJI é  $O(MAX\_ITER * m)$  levando o tempo do algoritmo pertencente a  $O(MAX\_ITER * m)$ .

**Algoritmo 3:** Coleção de Janelas.

---

```

Entrada:  $\mathbb{S}^{(1)}$ ,  $\mathbb{I}^{(1)}$ , TipoCJ, TipoDominio, l, c, MAX,  $\alpha$ ,  $\beta$ 
Saída: CJ: Coleção de janelas.
1  se (TipoCJ incluir  $CJB_f$ ) então
2      |            $CJB \leftarrow CJB_f(l, c)$ ;
3  fim se
4  se (TipoDominio = automático) OU (TipoCJ incluir  $CJB_a$ ) então
5      |            $\omega \leftarrow$  Determina  $\omega(\mathbb{S}^{(1)}, \mathbb{I}^{(1)}, MAX)$ ;
6  fim se
7  se TipoCJ incluir  $CJB_a$  então
8      |           se TipoDominio = usuário então
9          |            $CJB \leftarrow CJB_a(l, c, \omega, \alpha, \beta)$ ;
10         |           senão
11             |            $CJB \leftarrow CJB_a(\omega, \alpha, \beta)$ ;
12         |           fim se
13 fim se
14 se TipoCJ incluir CJI então
15     |           se TipoDominio = usuário então
16         |            $CJI \leftarrow MMII(\mathbb{S}^{(1)}, \mathbb{I}^{(1)}, l, c)$ ;
17         |           senão
18             |            $\omega_{max} = \lceil \beta * \omega \rceil$ ;
19             |            $l \leftarrow \lceil \sqrt{\omega_{max}} \rceil + 1 + \text{mod}(\lceil \sqrt{\omega_{max}} \rceil, 2)$ ;
20             |            $CJI \leftarrow MMII(\mathbb{S}^{(1)}, \mathbb{I}^{(1)}, l, l)$ ;
21         |           fim se
22 fim se
23 se (TipoCJ =  $CJB_aI$ ) OU (TipoCJ =  $CJB_fI$ ) então
24     |            $CJ \leftarrow CJB \cup CJI$ ;
25 senão
26     |           se TipoCJ = CJI então
27         |            $CJ \leftarrow CJI$ ;
28     |           senão
29         |            $CJ \leftarrow CJB$ ;
30     |           fim se
31 fim se
32 devolva CJ;

```

---

**3.2.1** Comentários gerais sobre a implementação

O algoritmo foi implementado em Java (Oracle, 2015) para a execução dos experimentos realizados ao longo desta pesquisa. O algoritmo usado para construir as janelas em CJI foi desenvolvido pelo autor da proposta Santos (2010) em Python (Pyt, 2015). Essa implementação foi utilizada neste trabalho.

## Capítulo 4

# Seleção de combinações de janelas usando algoritmo genético

Estabelecida uma coleção de janelas (CJ), um mecanismo de seleção pode ser aplicado sobre ela, a fim de escolher aquelas mais adequadas para compor uma combinação de janelas para o primeiro nível de um projeto dois níveis. Neste capítulo é apresentada a proposta de um algoritmo genético para realizar essa seleção.

A motivação para a utilização de AGs é o fato de diferenciarem-se dos esquemas enumerativos (*força bruta*) por não procurarem em todos os pontos do espaço de busca, mas sim em um subconjunto desses pontos, obtendo (na maioria das vezes) resultados satisfatórios (Linden, 2006). Essa propriedade indica que combinações razoáveis podem ser selecionadas sem a necessidade de testar todas as possibilidades.

Existem ainda outras características como: paralelização; facilidade de lidar com espaços de buscas intratavelmente grandes; desempenho robusto para diferentes domínios de aplicação (Oliveira, 1998), dentre outros, que encorajam a sua utilização na escolha de uma combinação de janelas.

### 4.1 Algoritmo genético para seleção de janelas

Os AGs, formalizados por Holland (1975), procuram simular o mecanismo de sobrevivência e reprodução das espécies, onde indivíduos de uma mesma população competem entre si pela sobrevivência, seja por alimento ou pela reprodução. Em um AG, um **indivíduo** representa uma **solução** para um problema a ser resolvido e um agrupamento de indivíduos compõem uma **população**.

Os indivíduos mais aptos possuem maiores chances de sobreviverem e gerarem descendentes, ao passo que os mais fracos (menos aptos) acabam morrendo. A aptidão de cada membro da população é medida através de uma **função de avaliação**.

Dois indivíduos, ou seja duas soluções, podem trocar informações ou ainda sofrerem alguma modificação. A primeira, o **crossover**, é equivalente a um mecanismo de reprodução sexuada e a segunda é similar a uma alteração genética, a **mutação**. Tais mecanismos fazem parte dos **operadores genéticos**. Esses operadores atuam até um determinado momento preestabelecido, quando o melhor indivíduo é devolvido como resposta do AG.

A uma execução completa de todas as etapas de um AG dá-se o nome de **geração**, estabelecendo assim o funcionamento do AG. Para que esse AG forneça um indivíduo de qualidade é necessária a execução de várias gerações.



Um esquema básico de funcionamento de um AG pode ser observado na Figura 4.1.

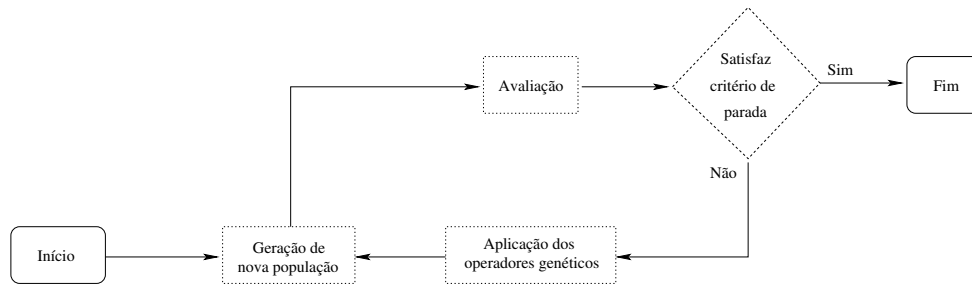


Figura 4.1: Esquema básico de funcionamento de um AG.

Para o projeto de um AG, dois itens devem estar bem definidos: a forma de representação de uma solução, chamada de **cromossomo**, onde a menor unidade que o compõe é denominada de **gene** (característica); e sua função de avaliação. O primeiro é a base para se definir os **operadores genéticos**, responsáveis pela evolução dos cromossomos, e o segundo estabelece uma ligação do AG com o problema a ser resolvido.

Nas próximas seções são definidos: o cromossomo, a função de avaliação, os operadores genéticos e por último um esboço do AG que seleciona uma combinação de janelas de uma CJ.

#### 4.1.1 Cromossomo

Adotou-se nesta pesquisa que um gene de um cromossomo simboliza uma janela. Um exemplo de um cromossomo com dois genes pode ser observado na Figura 4.2.

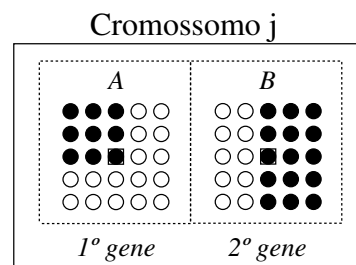


Figura 4.2: Exemplo de um cromossomo contendo duas janelas onde cada gene representa uma janela disposta sobre um domínio  $(5 \times 5)$ .

Uma vez que este trabalho adota uma janela de um ponto central no segundo nível, apenas as janelas de primeiro nível precisam ser representadas em um cromossomo. Isso posto, o **tamanho mínimo** de um cromossomo, ou seja o número mínimo de janelas que o compõe, deve ser um número maior ou igual a dois, pois uma única janela no primeiro nível não faz sentido, uma vez que a janela de segundo nível tem apenas um ponto. O tamanho mínimo é nomeado por  $minJ$ .

Quanto ao seu **tamanho máximo**, é importante lembrar que a combinação escolhida é usada posteriormente no treinamento de um operador, implicando na execução do projeto dois níveis por completo. Portanto, o tamanho máximo de um cromossomo não deve ser grande e é denominado de  $maxJ$ .



### 4.1.2 Função de avaliação

A função de avaliação deve ser capaz de expressar a qualidade do operador projetado correspondente à combinação de janelas para o primeiro nível representada em um cromossomo. Nesse sentido, o MAE foi adotado como medida de avaliação, uma vez que pode ser usado para comparar dois operadores e já é bem fundamentado por trabalhos neste contexto.

Esta escolha implica na necessidade de se executar o projeto dois níveis (Capítulo 2) para se obter um operador dois níveis correspondente à combinação definida pelo cromossomo antes do cálculo do MAE. A avaliação necessita de três conjuntos de imagens: as imagens de treinamento entradas/ideais do primeiro nível ( $\mathbb{S}^{(1)}, \mathbb{I}^{(1)}$ ); as imagens de treinamento do segundo nível ( $\mathbb{S}^{(2)}, \mathbb{I}^{(2)}$ ); e as imagens para calcular o MAE ( $\mathbb{S}, \mathbb{I}$ ).

Seja um cromossomo com  $t$  janelas,  $t \in \mathbb{N}^*$ . O projeto um nível deve ser executado para cada janela  $W_1^{(1)}, W_2^{(1)}, \dots, W_t^{(1)}$  do cromossomo, utilizando as imagens  $\mathbb{S}^{(1)}, \mathbb{I}^{(1)}$ . Os operadores projetados  $\hat{\Psi}_1^{(1)}, \hat{\Psi}_2^{(1)}, \dots, \hat{\Psi}_t^{(1)}$  são aplicados sobre as imagens  $\mathbb{S}^{(2)}$  e o resultado dessa aplicação, mais as imagens  $\mathbb{I}^{(2)}$  e a janela de segundo nível são fornecidos como entrada ao segundo nível. O resultado é um operador  $\Psi^{(2)}$  e consequentemente o operador dois níveis  $\hat{\Psi} = \hat{\Psi}^{(2)}(\hat{\Psi}_1^{(1)}, \hat{\Psi}_2^{(1)}, \dots, \hat{\Psi}_t^{(1)})$ .

A avaliação do cromossomo pode agora ser realizada. Para isso, o operador dois níveis é aplicado ao conjunto de imagens  $\mathbb{S}$ . As imagens resultantes, mais as imagens  $\mathbb{I}$  são usadas para obter o MAE.

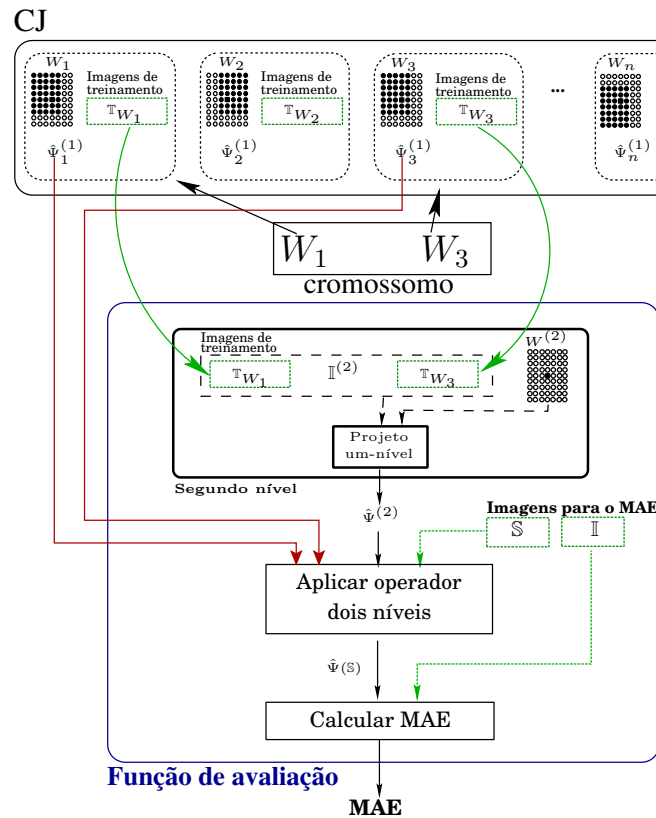
Devido ao alto custo dos treinamentos, é necessário buscar mecanismos que diminuam ao máximo a execução do projeto um nível referente às janelas do cromossomo, para não inviabilizar a aplicação do AG, uma vez que a cada geração é necessário executar a função de avaliação para cada cromossomo. Isso sugere que a população do AG não deve ser formada por muitos indivíduos, para que o tempo de execução de uma geração seja aceitável.

O procedimento adotado para melhorar o desempenho da função de avaliação é a execução preliminar do projeto um nível para cada janela da CJ. Assim, quando uma combinação for avaliada, executa-se apenas o segundo nível. Para que o propósito do procedimento seja atingido, para cada janela  $W_j$  em uma CJ, são armazenados os respectivos operadores projetados e as imagens resultantes das aplicações desses sobre as imagens  $\mathbb{S}^{(2)}$ . Os resultados destas aplicações são as imagens de entrada para o treinamento do segundo nível. Estas imagens são aqui identificadas por  $\mathbb{T}_{W_j} = \hat{\Psi}_j^{(1)}(\mathbb{S}^{(2)})$ .

A função de avaliação pode ser representada pelo Algoritmo 4. Na linha 1 do algoritmo são recuperadas as imagens  $\mathbb{T}_{W_j}$  para cada janela  $W_j$  pertencente ao cromossomo. Na linha 2, essas imagens e suas respectivas imagens ideais  $\mathbb{I}^{(2)}$  são usadas para executar o treinamento do segundo nível do projeto dois níveis, que retorna o operador de segundo nível  $\hat{\Psi}^{(2)}$ . Os operadores de primeiro nível referentes a cada janela do cromossomo em avaliação, são recuperados na linha 3. Com isso, o operador dois níveis pode ser aplicado sobre as imagens  $\mathbb{S}$  (linha 4). O resultado  $\hat{\Psi}(\mathbb{S})$  mais as respectivas imagens ideais  $\mathbb{I}$ , são então comparadas para se obter o MAE (linha 5). Esse é devolvido pela função. Um exemplo de seu funcionamento pode ser observado na Figura 4.3.

**Algoritmo 4:** Função de avaliação**Entrada:** cromossomo,  $\mathbb{I}^{(2)}$ ,  $\mathbb{S}$ ,  $\mathbb{I}$ ,  $CJ$ **Saída:** MAE

- 1  $\mathbb{T}_W \leftarrow \bigcup \mathbb{T}_{W_j} \in CJ$  tal que  $W_j$  pertence ao cromossomo ;
- 2  $\hat{\Psi}^{(2)} \leftarrow$  Treinamento do segundo nível(  $\mathbb{T}_W, \mathbb{I}^{(2)}$  ) ;
- 3  $\hat{\Psi}_1^{(1)}, \dots, \hat{\Psi}_t^{(1)} \leftarrow$  Recuperar em  $CJ$  os operadores de primeiro nível pertencentes ao cromossomo ;
- 4  $\hat{\Psi}(\mathbb{S}) \leftarrow$  Aplicar operador dois níveis(  $\hat{\Psi}_1^{(1)}, \dots, \hat{\Psi}_t^{(1)}, \hat{\Psi}^{(2)}, \mathbb{S}$  ) ;
- 5  $MAE \leftarrow$  Calcula o MAE(  $\hat{\Psi}(\mathbb{S}), \mathbb{I}$  ) ;
- 6 **devolva MAE ;**



**Figura 4.3:** No topo da figura as janelas da  $CJ$ , os respectivos operadores de primeiro nível e as suas imagens de treinamento para o segundo nível. No meio, um cromossomo contendo duas janelas e na parte inferior a execução da função de avaliação para esse cromossomo.

### 4.1.3 Operadores genéticos

Os operadores genéticos são os responsáveis em permitir que uma população evolua, gerando diversidade e produzindo novas soluções. A definição dos operadores está atrelada a codificação dos cromossomos mas, independente disso, o conceito geral dos operadores se mantém o mesmo. Três operadores se destacam: **seleção**, **crossover** e **mutação**.

### ► Seleção

Seguindo o processo de evolução das espécies, sobrevive e se reproduz o indivíduo mais apto ao meio ao qual está inserido. Os AGs tentam imitar esse princípio usando a função de avaliação para indicar quais são os mais aptos. Aqueles menos aptos, eventualmente, podem gerar descendentes e isso deve ser permitido, pois um gene desse cromossomo pode ser essencial para gerar um outro melhor.

Existem alguns mecanismos de seleção como: seleção baseada em classificação (Baker (1985) apud Goldberg e Deb (1991)) que atribui um valor de classificação a cada cromossomo, ao invés de usar sua função de avaliação diretamente; seleção por torneio (Brindle (1981) apud Goldberg e Deb (1991)), onde dois cromossomos selecionados aleatoriamente, por exemplo, disputam entre si uma vaga na próxima população; o método da roleta; o elitismo e outras técnicas particulares podem ser implementadas ou adequadas a um problema ou ainda a combinação entre os métodos.

Como mecanismo de seleção adotou-se o **elitismo**, método indicado para populações pequenas (Linden, 2006) que é o caso da população do AG aqui proposto. Neste método, a avaliação do melhor indivíduo da população atual é comparada com a do melhor indivíduo obtido até o momento. Caso o primeiro não seja melhor, o melhor indivíduo é inserido na população atual. Esse processo pode fazer com que o tamanho da população aumente gradativamente.

### ► Crossover

Para a realização do crossover, pares de cromossomos (pais) são selecionados aleatoriamente, tendo como parâmetro uma **probabilidade de crossover (pc)** fornecida ao AG. Na literatura sobre AG, geralmente essa probabilidade é alta, em torno de 0.8, pois deseja-se explorar ao máximo a diversidade genética. Caso o número sorteado seja menor ou igual a **pc**, o cromossomo participará do crossover. Neste trabalho, um mesmo cromossomo nunca é selecionado duas vezes. Pode ocorrer do último cromossomo selecionado não ter parceiro. Nesse caso ele não é utilizado pelo operador.

Um ponto de troca de material genético é estabelecido e como os cromossomos pais podem ser de tamanhos diferentes, o ponto é definido aleatoriamente no cromossomo de tamanho menor. A troca de janelas ocorre do final dos cromossomos até o ponto de troca. Quando uma janela a ser trocada já existe em outro cromossomo, a troca dessa janela não ocorre. Os filhos gerados substituem os pais. Um exemplo desse processo pode ser observado na Figura 4.4.

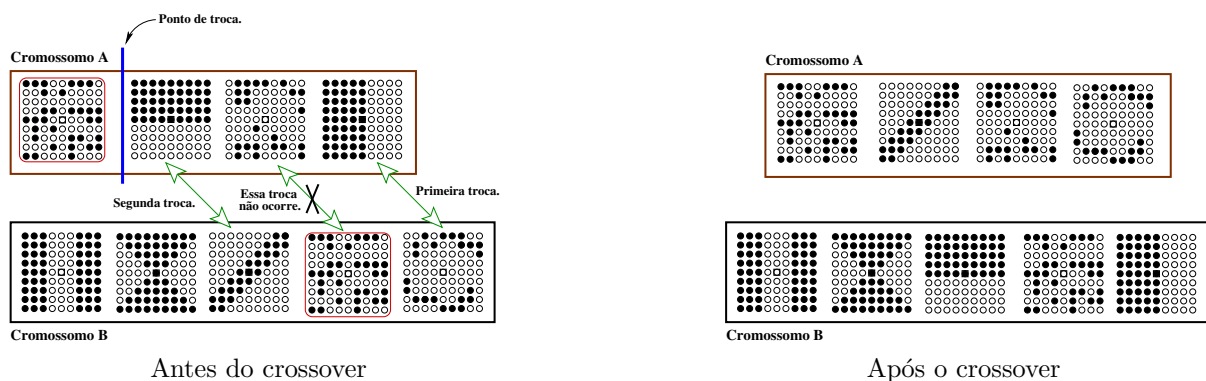


Figura 4.4: Exemplo de um crossover entre dois cromossomos de uma população.

### ► Mutação

Uma mutação tem o papel de modificar um ou mais genes de um cromossomo, promovendo a variabilidade genética. Geralmente ela está associada a uma **probabilidade de mutação (pm)** que costuma ser baixa, como ocorre na natureza. Usualmente em torno de 0.005. Frequentemente essa probabilidade é usada para selecionar se um gene será alterado ou não. Para cada gene de um cromossomo sorteia-se um valor em  $[0, 1]$  e se esse valor for menor ou igual a **pm** ele será mutado.

Este trabalho definiu três tipos de mutação. Para cada cromossomo filho, resultante do *crossover*, um número é sorteado entre 1 e 3, indicando a mutação a ser aplicada sobre o indivíduo. O cromossomo mutado substitui o original. Os tipos de mutações são:

- 1) **mutaçãoT**: um número aleatório em  $[0, 1]$  é gerado para cada gene (janela) do cromossomo. Quando esse número for menor que a **pm**, a janela em questão é identificada para a mutação. Para cada janela identificada é sorteada uma janela da CJ para a troca. Esse sorteio em CJ se repete até que a janela sorteada seja diferente de qualquer janela do cromossomo. Supondo que não se encontre na CJ janela diferente das existentes no cromossomo, não ocorre a substituição. Esse fato pode ocorrer, por exemplo, se o tamanho da coleção for igual ao tamanho do cromossomo. Nesta mutação o tamanho do cromossomo não é modificado. A Figura 4.5 apresenta um exemplo desse mecanismo.

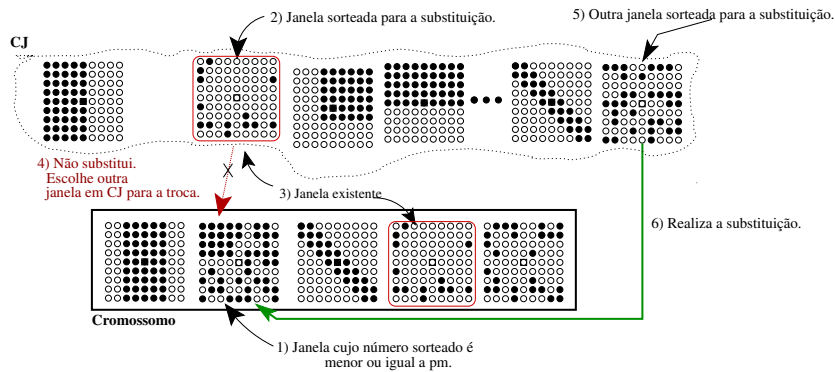


Figura 4.5: Exemplo de mutaçãoT em um gene de um cromossomo.

- 2) **mutaçãoA**: um número aleatório  $x \in [1, maxJ - tamanho\ atual\ do\ cromossomo]$  de janelas são escolhidas da CJ e adicionadas ao cromossomo. Essas janelas adicionadas são diferentes das existentes no indivíduo. A adição ocorre até que o número sorteado seja atingido ou, se não for possível, até que existam janelas em CJ diferentes daquelas do cromossomo. Ocorrendo a possibilidade de não haver na CJ janelas diferentes, a operação não é realizada. O cromossomo aumenta de tamanho e caso ele já esteja em seu tamanho máximo, a mutação não ocorre. Um exemplo pode ser observado na Figura 4.6.

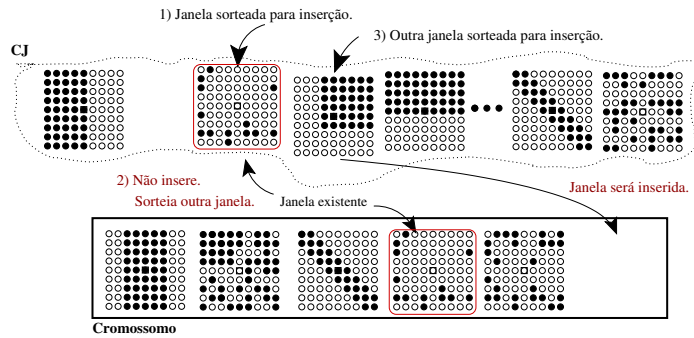


Figura 4.6: Exemplo de mutação A para um cromossomo.

3) **mutação E:** neste mecanismo um cromossomo pode sofrer um decréscimo de genes caso ele não esteja em seu tamanho mínimo. Um número aleatório  $x \in [1, \text{tamanho atual do cromossomo} - \text{min}J]$  define quantas janelas serão eliminadas. Elas são sorteadas uniformemente no cromossomo. Dessa maneira, se o melhor cromossomo tiver um número menor de janelas do que os cromossomos atuais, existirá a possibilidade dele ser alcançado. Um exemplo é exibido na Figura 4.7.

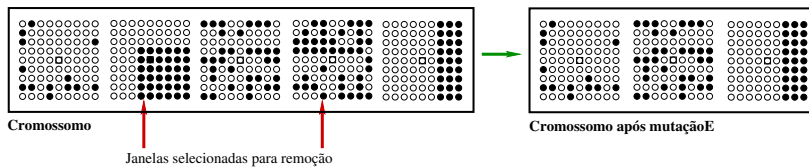


Figura 4.7: Exemplo de mutação E para um cromossomo.

#### 4.1.4 O algoritmo genético WGA

Com base no que foi exposto nas seções anteriores é proposto um algoritmo genético para seleção de uma combinação de janelas para o primeiro nível de um projeto dois níveis, denominado por WGA: **Window selection using Genetic Algorithm** e apresentado em Algoritmo 5. Os parâmetros desse algoritmo são a probabilidade de *crossover* (**pc**), a probabilidade de mutação (**pm**), o tamanho inicial da população (**tp**), o número máximo de gerações do AG (**maxG**), o número máximo de gerações permitidas ao AG sem que o melhor cromossomo evolua (**maxGSemA**), *minJ*, *maxJ*, a CJ e as imagens de treinamento.

**Algoritmo 5: WGA.**


---

```

1   Projeta operador para cada  $W_j \in CJ$  e aplica os operadores sobre  $\mathbb{S}^{(2)}$  ;
2   Constrói População ;
3    $numGeracoes, numGeracoesSemA \leftarrow 0$  ;
4   Avalia População ;
5   Obtém MelhorIndividuo da População ;
6   enquanto ( $numGeracoes < maxG$ ) E ( $numGeracoesSemA < maxGSemA$ ) faça
7       |   Crossover sobre a População ;
8       |   Mutação sobre a População ;
9       |   Avalia População ;
10      |   Realiza eletismo sobre a População;
11      |   Obtém MelhorIndividuo da População ;
12      |    $numGeracoes \leftarrow numGeracoes + 1$  ;
13      |   se MelhorIndividuo não melhorou então
14      |       |    $numGeracoesSemA \leftarrow numGeracoesSemA + 1$  ;
15      |   senão
16      |       |    $numGeracoesSemA \leftarrow 0$  ;
17      |   fim se
18   fim enquanto
19   devolva MelhorIndividuo ;

```

---

► **Projeta operador para cada  $W_j \in CJ$  e aplica os operadores sobre  $\mathbb{S}^{(2)}$**

Este passo consiste em projetar o operador um nível para cada janela  $W_j$  da CJ (linha 1), usando as imagens em  $\mathbb{S}^{(1)}$  e  $\mathbb{I}^{(1)}$ . Uma vez projetado, ele é aplicado sobre as imagens em  $\mathbb{S}^{(2)}$ , obtendo as imagens  $\mathbb{T}_{W_j}$  que serão fornecidas como entrada para o segundo nível no cálculo na função de avaliação.

► **Constrói População**

Neste passo, inicia-se a execução do AG propriamente dito. A população inicial é composta por  $tp$  cromossomos que são gerados de forma aleatória, ou seja, para cada cromossomo é sorteado o seu número de janelas, entre  $[minJ, maxJ]$ , que são escolhidas aleatoriamente na CJ, sem repetição.

► **Avalia População**

Cada cromossomo da população é avaliado através da função de avaliação apresentada no Algoritmo 4.

► **Obtém MelhorIndividuo da População**

O indivíduo de melhor avaliação da população é armazenado em MelhorIndividuo ( linha 5 ). Esse processo se repete ao término de cada geração ( linha 11 ). O melhor indivíduo é utilizado no eletismo.

► **Crossover**

Para a realização do *crossover*, pares de cromossomos (pais) são selecionados mediante a **pc** fornecida ao AG e sobre eles é aplicado o crossover de um ponto. Esse processo substitui, na população, os pais pelos filhos.

► **Mutação**

Sobre os filhos resultantes do *crossover* é aplicada a mutação. Esse processo substitui os filhos do crossover pelos filhos modificados.

► ***Eletismo***

Após a avaliação de cada cromossomo da população atual, o melhor indivíduo dessa população é comparado com o indivíduo armazenado em *MelhorIndividuo*. Caso o primeiro não seja melhor, o melhor indivíduo é inserido na população atual.

► ***Critério de parada***

Quando o número de gerações prefixado (**maxG**) for atingido ou o melhor cromossomo não for aprimorado por um número de gerações estabelecido (**maxGSemA**), o WGA encerra o seu funcionamento.

► ***Retorno do algoritmo***

O WGA retorna o indivíduo de melhor avaliação na última geração. Na verdade, ele é o melhor cromossomo encontrado entre todas as gerações executadas. Esse cromossomo contém as janelas de primeiro nível para o projeto dois níveis.

► ***Complexidade de tempo do WGA***

Para se obter o MAE na função de avaliação, é necessário antes treinar o operador dois níveis. Isso, no pior caso, é exponencial, o que estabelece a complexidade de tempo para o WGA.

#### 4.1.5 Comentários gerais sobre a implementação

A metodologia aqui proposta pode ser implementada em qualquer linguagem de programação atual. Neste trabalho, a linguagem Java (Oracle, 2015) foi utilizada para a implementação do WGA. O projeto dois níveis é um sistema já pronto que foi desenvolvido em *C*, integrado ao sistema Khoros (Rasure *et al.*, 1990).

► ***Detalhes de implementação do cromossomo***

É importante comentar alguns detalhes sobre a implementação do cromossomo, uma vez que a mesma influencia nos outros blocos do programa, bem como no desempenho do AG.

Uma alternativa é codificar cada gene do cromossomo para armazenar um identificador para uma determinada janela do CJ. Isso permite economia de memória e a aplicação dos operadores genéticos de forma eficaz. Por exemplo, no *crossover* apenas os identificadores serão trocados e não a janela inteira.

Como não devem existir janelas iguais em um cromossomo, uma verificação para se evitar duplicatas deve ser realizada, mas isso pode ser feito rapidamente com uma busca sequencial, uma vez que *maxJ* não é um número grande.

Outra alternativa de implementação é representar um cromossomo como um vetor de  $N$  bits, sendo que  $N$  é o número de janelas do CJ. Cada posição do cromossomo indica se a correspondente janela faz (bit 1) ou não (bit 0) parte do indivíduo. Essa configuração permite verificar diretamente se uma janela pertence a um cromossomo ou não, bastando olhar o bit de sua posição. Apesar dessa facilidade, um cuidado deve ser tomado com o *crossover* e mutação, para que os mesmos não gerem indivíduos inválidos, ou seja, cromossomos sem nenhuma janela ou com um número grande de janelas. No pior caso o cromossomo seria o próprio CJ.

Nas duas alternativas deve-se percorrer o cromossomo para realizar validações. Como na primeira forma de codificação o cromossomo terá no máximo  $maxJ$  posições, sendo esse valor bem menor que  $N$ , optou-se por esta representação.

## 4.2 Parâmetros do WGA

Geralmente vários testes são realizados com um AG, modificando seus parâmetros, a fim de estabelecer valores satisfatórios para o algoritmo.

No sentido de ajustar os parâmetros para o WGA, foram realizados alguns testes preliminares. Esses testes também tiveram o objetivo de averiguar se o WGA realmente é capaz de selecionar janelas adequadas para um dado problema. Testes comparativos com as combinações manuais estão expostos no Capítulo 6.

### 4.2.1 Conjunto de dados

Os dados utilizados nos experimentos foram extraídos do trabalho de Hirata (2009) e sua descrição encontra-se na Tabela 4.1.

Nas colunas *Primeiro nível* e *Segundo nível* estão relacionados os números de imagens usadas nesses níveis para a execução de cada teste.  $\mathbb{S}^{(1)}$  e  $\mathbb{S}^{(2)}$  representam as imagens de entrada para o primeiro e segundo níveis e  $\mathbb{I}^{(1)}$  e  $\mathbb{I}^{(2)}$  indicam suas respectivas imagens ideais.

Para mensurar a qualidade de uma combinação de janelas, o MAE é calculado baseando-se no resultado da aplicação do operador projetado sobre as imagens do conjunto  $\mathbb{S}$ , mais as correspondentes imagens ideais em  $\mathbb{I}$ . É importante frisar que os conjuntos de imagens são diferentes entre si.

**Tabela 4.1:** Descrição dos conjuntos de dados usados nos experimentos com o WGA. Fonte: Hirata (2009).

Conjunto de dados	Descrição	Imagens de treinamento		
		Primeiro nível ( $\mathbb{S}^{(1)}, \mathbb{I}^{(1)}$ )	Segundo nível ( $\mathbb{S}^{(2)}, \mathbb{I}^{(2)}$ )	MAE WGA ( $\mathbb{S}, \mathbb{I}$ )
<i>Char_s</i>	Extração do caractere <i>s</i> em textos.	6	4	10
<i>Mapa</i>	Segmentação de regiões em mapas com uma dada textura.	2	1	2
<i>CircCirc</i>	Extração de círculos em imagens de circuitos eletrônicos.	5	3	10
<i>BoxCirc</i>	Extração de retângulos tracejados em imagens de circuitos eletrônicos.	5	3	10

### 4.2.2 Experimentos

A execução ocorreu em uma máquina com o sistema operacional Debian GNU/Linux e arquitetura com quatro processadores de 2.40GHz cada um.

O AG testado utilizou-se do crossover proposto, da mutaçãoT e da mutaçãoA. Três tipos de testes foram executados.



**Teste 1 (T1):** para cada conjunto de dados foi contruída uma CJ com janelas básicas, em que o domínio das janelas foi o mesmo definido para as janelas manuais em (Hirata, 2009), com exceção no conjunto *Mapa* e *BoxCirc*, onde para o primeiro definiu-se um domínio  $3 \times 7$  e para o segundo  $9 \times 9$ . Nesse caso, uma combinação manual para cada um dos dois conjuntos de dados, diferente da apresentada no trabalho de (Hirata, 2009), foi estabelecida para fins de comparação. As janelas manuais foram inseridas na CJ e um indivíduo contendo a combinação manual foi adicionado na população inicial. Dessa forma, fica garantido que a combinação encontrada pelo WGA nunca é pior que a manual. O objetivo deste teste foi verificar a existência de uma combinação melhor que a manual fazendo uso apenas de janelas básicas, bem como averiguar o comportamento do algoritmo com diferentes parâmetros. Os valores de  $pc$  e  $pm$  foram  $\mathbf{pc} \in \{0.5, 0.6, 0.7, 0.8\}$  e  $\mathbf{pm} \in \{0.001, 0.001, 0.1\}$ . O restante dos parâmetros podem ser observados na Tabela 4.2. No total, 48 variações de testes foram executadas.

**Tabela 4.2:** Parâmetros para os testes 1 e 2.

Tamanho inicial da população ( $tp$ )	Número de gerações ( $maxG$ )	Máximo de gerações sem aprimorar o melhor cromossomo ( $maxGSemA$ )
5	50	10
5	100	20
10	100	20
10	200	40

**Teste 2 (T2):** este teste é equivalente a T1 com uma exceção: não foi inserido na população inicial o cromossomo com a combinação manual. A ideia foi verificar se o algoritmo seria capaz de encontrar uma combinação equivalente ou superior a manual sem nenhuma informação prévia em sua população inicial.

**Teste 3 (T3):** uma única CJ para os quatro conjuntos de dados foi construída, contendo todas as janelas das CJs usadas nos quatro conjuntos de dados do teste 1, ou seja as janelas manuais desses conjuntos estavam presentes na CJ. Além disso, outras janelas foram incluídas na coleção sem nenhum critério especial. O domínio máximo foi de  $12 \times 9$ . Nenhum cromossomo específico foi inserido na população inicial. Foram adotados os seguintes parâmetros para execução deste teste:  $\mathbf{pc} = 0.8$  e  $\mathbf{pm} = 0.01$ , pois esses foram os que forneceram os melhores resultados em T1 e T2;  $tp = 20$ ;  $maxG = 200$  e  $maxGSemA = 50$ . Este teste teve a finalidade de averiguar se o algoritmo iria se comportar bem com uma CJ maior.

Quando estes testes foram realizados, as coleções de janelas apresentadas no Capítulo 3 não estavam ainda definidas como procedimentos, mas as CJs usadas nos testes 1 e 2 continham a maioria das janelas da  $CJB_f$ , com exceção da janela 24. O número de janelas nestas coleções ficou em torno de 30 janelas.

Em todos os três testes o número máximo de janelas em um cromossomo foi 15 ( $maxJ = 15$ ) e  $minJ = 3$ .

### 4.2.3 Resultados e discussões

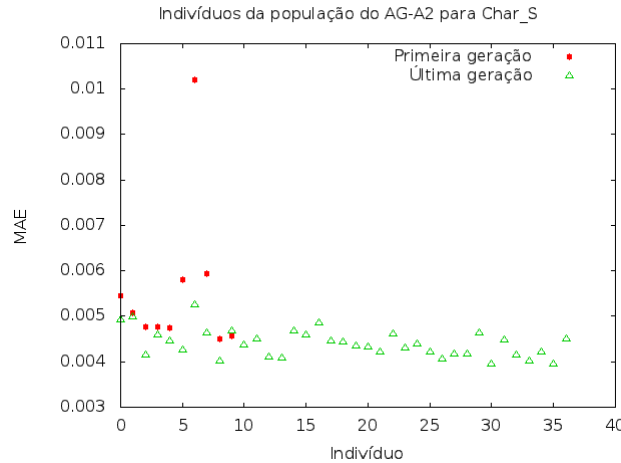
A Tabela 4.3 apresenta os parâmetros dos testes que forneceram os melhores resultados, bem como exibe o valor do MAE sobre  $(\mathbb{S}, \mathbb{I})$  obtido pelos operadores manuais e pelos operadores projetados mediante as combinações selecionadas pelo WGA.

**Tabela 4.3:** Parâmetros do WGA que produziram as melhores combinações de janelas e o MAE sobre  $(\mathbb{S}, \mathbb{I})$ .

Parâmetros	Char_s	Mapa	CircCirc	BoxCirc
Teste	1	3	2	3
Tamanho da pop. inicial (tp)	10	10	10	10
Tamanho da pop. final	15	39	24	35
maxG	100	200	200	200
maxGSemA	20	50	40	50
<b>pc</b>	0.5	0.8	0.7	0.8
<b>pm</b>	0.1	0.01	0.001	0.01
Número de gerações executadas pelo AG	94	193	106	158
Núm. de janelas do melhor cromossomo	7	13	8	10
Núm. de janelas da combinação manual	5	3	6	6
MAE do melhor cromossomo	0.003707	0.014969	0.005879	0.045121
MAE da combinação manual	0.005520	0.043178	0.007562	0.079702

O teste 1 mostrou que existem combinações melhores que as manuais e o AG conseguiu selecionar tais combinações. Os resultados dos testes 2 e 3 complementam o teste 1, atestando a eficácia do AG proposto que retornou, em todos os testes, combinações melhores que as manuais.

O comportamento do AG durante suas gerações nos diferentes testes foi analisado. Observou-se que o algoritmo produz uma população diversificada, onde a melhoria genética ocorre a medida que as gerações evoluem, ou seja, ele se comporta como o esperado. Esse comportamento pode ser observado na Figura 4.8, que exhibe a avaliação do melhor cromossomo na primeira e última gerações para o conjunto de dados *Char\_s*.

**Figura 4.8:** MAE sobre  $\mathbb{S}$  e  $\mathbb{I}$  dos indivíduos da primeira e última gerações do WGA para o conjunto de dados *Char\_s*.

Analisando os resultados dos 96 testes (testes 1 e 2) foi possível estabelecer alguns valores para os parâmetros do WGA. O tamanho inicial da população pode ser estabelecido em 10 cromossomos. As probabilidades **pc** e **pm** podem ser fixadas em 0.8 e 0.01, respectivamente. *maxG* igual a 200 e *maxGSemA* igual a 50. Outro ponto importante é o tamanho máximo que um cromossomo pode assumir. Foi estabelecido *maxJ* = 15 e na maioria dos resultados os cromossomos possuíam um número de janelas menor que 15, sugerindo que esse valor é um bom parâmetro.

Portanto, estes testes preliminares estabelecem parâmetros para o WGA e sugerem que sua utilização na seleção de combinação de janelas é uma ferramenta eficaz.

## Capítulo 5

# Seleção de combinações de janelas usando ordenação

Neste capítulo é descrita uma proposta para combinar janelas baseada na ordenação de uma coleção de janelas. A ideia é ordenar as janelas mediante um critério que as qualifique para compor uma combinação de primeiro nível em um projeto dois níveis. A combinação escolhida é aquela que apresenta menor MAE dentre as primeiras quinze janelas da ordenação. Esta metodologia treina um número fixo e reduzido de operadores para encontrar uma combinação de primeiro nível. Essa característica é uma vantagem sobre o WGA.

### 5.1 Ordenação para selecionar janelas

O projeto de um operador dois níveis pode ser visto como uma combinação de classificadores (Hirata, 2009), em que a escolha de cada janela define individualmente cada classificador participante da combinação, determinando o  $W$ -operador final. Quando um desses classificadores comete erros, pode-se agregar à combinação outro classificador, ou seja outra janela, a fim de melhorar a resposta final do modelo (Kuncheva, 2004). Baseando-se nesse conceito, é proposto um procedimento em que janelas, previamente mensuradas e ordenadas, são adicionadas sequencialmente a uma combinação de primeiro nível a fim de encontrar, dentre as avaliadas, uma com menor MAE.

Esta proposta, assim como no WGA, utiliza para o segundo nível uma janela de um ponto central, fazendo com que a ordenação fique restrita às janelas de primeiro nível.

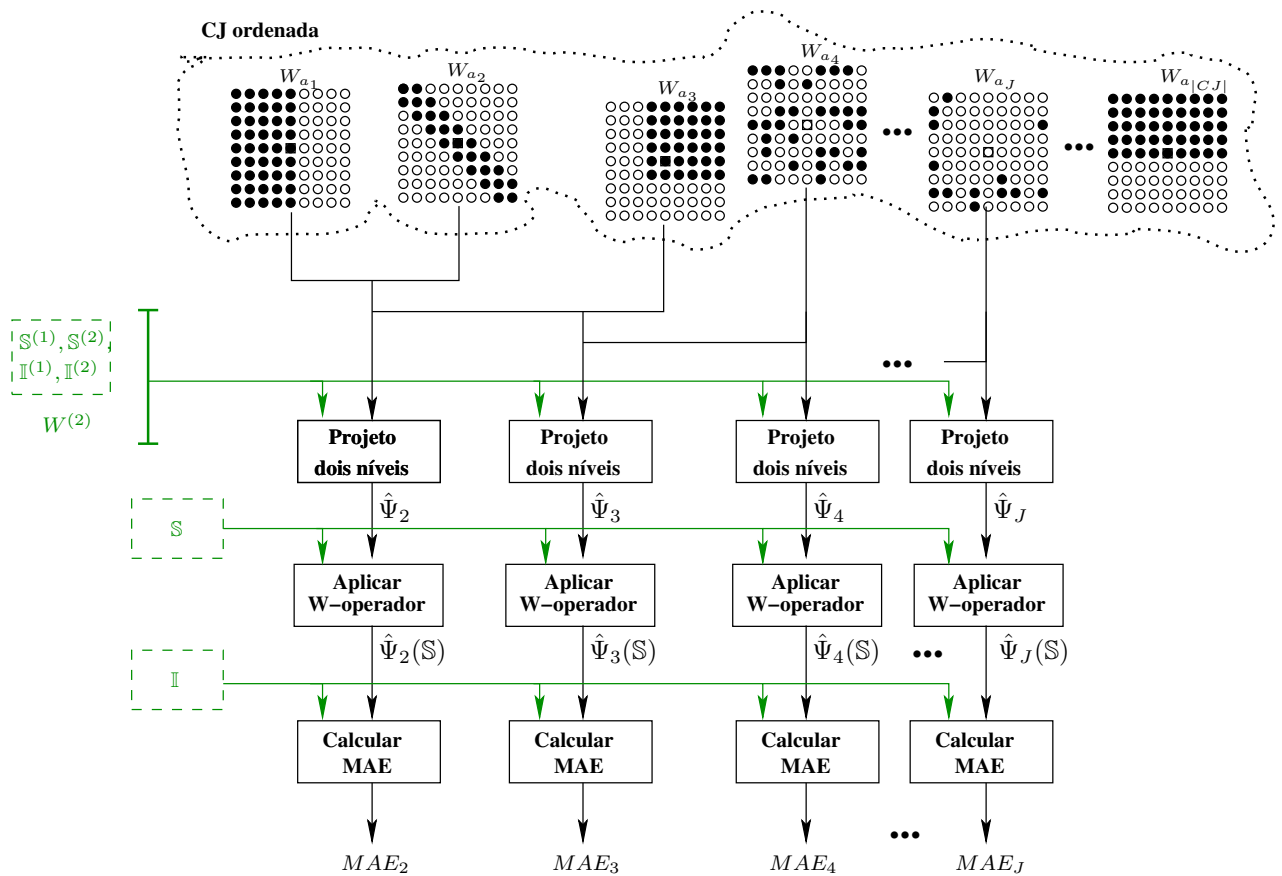
A metodologia está organizada nos seguintes passos:

1. Calcular, para cada janela de uma coleção de janelas  $CJ$ , o valor do critério de qualidade preestabelecido.
2. Ordenar  $CJ$  da melhor para a pior janela de acordo com o valor calculado no item anterior. Considere, após a ordenação,  $W_{a_1}$  a melhor janela,  $W_{a_2}$  a segunda melhor janela e assim sucessivamente, até  $W_{a_{|CJ|}}$ , em que  $|CJ|$  é o número de janelas da  $CJ$ .
3. Treinar o operador dois níveis para as duas melhores janelas da ordenação obtendo  $\hat{\Psi}_2$ ; treinar o operador dois níveis para as três melhores janelas da ordenação obtendo  $\hat{\Psi}_3$ ; repetir o processo até  $J$  janelas,  $J \in \mathbb{N}$ ,  $1 < J \leq |CJ|$ . Em relação ao valor máximo de  $J$ , quanto maior ele for, maior será o número de execuções do projeto um nível. A depender do problema, essa execução é custosa computacionalmente. Portanto, o número de janelas do primeiro nível não deve ser elevado. De acordo com os testes preliminares para o WGA, verificou-se que 15 janelas é um limite superior razoável para o primeiro nível.

Nesta fase utilizam-se imagens de treinamento para o primeiro nível pertencentes a  $\mathbb{S}^{(1)}, \mathbb{I}^{(1)}$  e imagens de treinamento para o segundo nível em  $\mathbb{S}^{(2)}, \mathbb{I}^{(2)}$ .

4. Aplicar os operadores dois níveis  $\hat{\Psi}_2, \hat{\Psi}_3, \hat{\Psi}_4, \dots, \hat{\Psi}_J$  em imagens de teste  $\mathbb{S}$ , obtendo como resultado o conjunto de imagens  $\hat{\Psi}_2(\mathbb{S}), \hat{\Psi}_3(\mathbb{S}), \hat{\Psi}_4(\mathbb{S}), \dots, \hat{\Psi}_J(\mathbb{S})$ .
5. Calcular o MAE para o conjunto de imagens  $\hat{\Psi}_2(\mathbb{S}), \hat{\Psi}_3(\mathbb{S}), \dots, \hat{\Psi}_J(\mathbb{S})$ , obtendo os valores  $MAE_2, MAE_3, \dots, MAE_J$ . Para o cálculo leva-se em consideração as respectivas imagens ideais  $\mathbb{I}$ .
6. Escolher o conjunto de  $g$  janelas, de  $W_{a_1}$  até  $W_{a_g}$ , tal que  $MAE_g$  é o menor entre  $MAE_2, MAE_3, \dots, MAE_J$ , para  $1 < g \leq J, g \in \mathbb{N}$ .

Uma versão gráfica exibindo a ideia geral da metodologia pode ser vista na Figura 5.1.



**Figura 5.1:** Esquema geral da metodologia de ordenação de janelas para seleção de uma combinação de janelas para um primeiro nível.

A escolha do critério de ordenação deve levar em conta o custo computacional, uma vez que uma CJ pode ter um número razoável de janelas.

É proposto utilizar como critério de ordenação a entropia condicional, já que não necessita do operador na obtenção de seu valor e sua aplicação na definição de janelas é bem fundamentada por outros trabalhos (Martins-Jr *et al.*, 2004; Santos, 2010).

## 5.2 Entropia condicional como critério de ordenação de janelas

Na combinação de classificadores, em termos gerais, espera-se que cada classificador de um conjunto seja especialista em uma área do problema a ser resolvido e quanto mais diverso for esse conjunto, melhor é o resultado final (Kuncheva, 2004).

A questão é como estabelecer a diversidade e a qualidade de um conjunto de classificadores ou, no contexto do projeto de operadores dois níveis, a diversidade e a qualidade de uma coleção de janelas do primeiro nível, possibilitando a ordenação dessa coleção.

Algumas medidas são propostas na literatura para tratar a questão. São exemplos (Kuncheva, 2004): as medidas de discordância e de falha dupla, que verificam as saídas dos pares de classificadores; as que calculam a diversidade analisando, de uma única vez, todos os classificadores do conjunto; dentre outras. Para todas as medidas pesquisadas é necessário ter a saída do classificador, ou seja ter projetado o W-operador. Portanto, elas não são adequadas, no escopo desta proposta, como critério de ordenação de janelas.

A entropia condicional de Shannon não mede o grau de diversidade entre duas janelas, mas pode ser adotada como um indicador de qualidade de uma janela. Essa consideração pode ser feita, pois quando a entropia condicional de uma janela é alta, tem-se uma grande incerteza sobre a sua escolha, caso contrário a janela é uma forte candidata a compor uma combinação de primeiro nível em um projeto dois níveis. Deve-se apenas tomar cuidado para que as janelas analisadas não sejam similares na forma, senão o ganho na combinação será desprezível.

A estruturação da coleção de janelas básicas (CJB) garante uma certa distinção de forma entre as janelas. Em relação às janelas obtidas pela informação de interação (CJI), não se pode afirmar que existe uma diversidade de forma entre as janelas, mas acredita-se que ela exista dada a sua construção. No contexto, onde alguma diversidade é garantida, a entropia condicional parece ser uma medida adequada para adjetivar uma janela e ser utilizada como critério na ordenação.

### 5.2.1 Entropia condicional corrigida

Para a utilização da entropia condicional como medida de qualidade na seleção de janelas, é necessário estabelecer um critério de correção em sua fórmula, uma vez que o conjunto de treinamento é fixo e as probabilidades são estimadas. Caso não ocorra correção, as janelas maiores terão valores menores para a entropia condicional, como apresentado no Capítulo 3. Logo, para que haja uma comparação justa no processo de ordenação, é necessário um fator de correção.

Já foi mostrado no Capítulo 3 que a correção sugerida por Martins-Jr *et al.* (2004, 2005) em  $\hat{H}_{c_M}(y|X)$  é severa demais. Assim, na busca por uma correção mais adequada, foi encontrado o artigo de Porta *et al.* (1998) que descreve um método para medir a regularidade de um processo ao longo de sequências curtas de dados. Ele apoia-se na definição de uma versão corrigida da entropia condicional e na extração do seu mínimo. O fator de correção baseia-se nos padrões que aparecem apenas um vez no conjunto de observações coletadas (**padrões únicos**) e nos padrões que aparecem mais de uma vez (**padrões não únicos**). Aos padrões que não aparecem, é atribuída entropia condicional zero. O artigo atribui para a entropia condicional dos padrões únicos um valor que representa a distribuição de probabilidade da série esperada de dados obtidos de ventilação mecânica em gatos.

Reis (2012) em seu trabalho descreve uma ideia semelhante a de Porta *et al.* (1998) para criar uma versão corrigida da entropia condicional, que foi utilizada como função custo na avaliação de algoritmos para a seleção de características em projetos de W-operadores um nível. Sua versão empregou valor 1 à entropia condicional dos padrões únicos e valor 0 para os padrões que não apareciam no conjunto de treinamento. Ele justifica esse valor argumentando que padrões com apenas uma observação sugerem que o conjunto de treinamento é insuficiente para representá-los. Sua equação é dada por:

$$\hat{H}_{cR}(y|X) = \frac{U}{totalO} + \sum_{\hat{p}(X_i) > \frac{1}{totalO}} \hat{p}(X_i) \hat{H}(y|X_i), \quad (5.1)$$

em que  $U$  é o total de padrões únicos.

A equação original da entropia condicional associa à  $\hat{H}(y|X_i)$  dos padrões únicos o valor zero, uma vez que esse padrão é vinculado a saída 0 ou 1, exclusivamente, indicando certeza absoluta sobre a saída que deve ser associada a ele.

A consideração feita por Reis (2012) sobre a insuficiência do conjunto de treinamento é válida, mas, na prática, o valor que ele atribuiu à entropia condicional dos padrões únicos é o oposto da equação original. Ele concedeu total incerteza sobre os padrões únicos, quando isso não é verdade, pois existe uma amostra de cada padrão único que carrega alguma informação. Parece que a correção é severa demais.

A proposta deste trabalho é usar a ideia de Porta *et al.* (1998) e, ao contrário da proposta de Reis (2012), atribuir uma correção baseada em um valor próximo de zero, ou seja mais próximo da entropia condicional real de um padrão único. A função proposta é nomeada de  $\hat{H}_c(y|X)$  e definida como:

$$\hat{H}_c(y|X) = H_c * \frac{U}{totalO} + \sum_{\hat{p}(X_i) > \frac{1}{totalO}} \hat{p}(X_i) \hat{H}(y|X_i), \quad (5.2)$$

em que  $H_c$  é o valor da entropia condicional considerado para os padrões únicos. Este trabalho adota o valor 0,001 para  $H_c$ . Esse valor representa 0,1 por cento da entropia máxima, parecendo um valor razoável com o conjunto de treinamento existente na prática, informando que apesar do padrão possuir apenas um representante, existe uma pequena incerteza sobre ele.

O procedimento de ordenação utilizando a  $\hat{H}_c(y|X)$  é denominado neste texto por **WES**: Window selection using Entropy based Sorting.

### 5.3 O algoritmo WES

O algoritmo 6 representa a metodologia proposta utilizando  $\hat{H}_c(y|X)$  e supondo que a coleção de janelas é diversificada. As imagens em  $\mathbb{S}^{(1)}$ ,  $\mathbb{I}^{(1)}$  são usadas para dois fins. O primeiro, no cálculo da  $\hat{H}_c(y|X)$  para cada janela da CJ (Linha 1). O segundo, no treinamento do primeiro nível para as  $J$  primeiras janelas ordenadas. Isso ocorre nas linhas 5 e 7. Na linha 8 ocorre o treinamento do segundo nível com os  $i$  operadores projetados e as imagens em  $\mathbb{S}^{(2)}$ ,  $\mathbb{I}^{(2)}$ . Esse treinamento gera o operador do segundo nível  $\hat{\Psi}_i^{(2)}$ . O resultado final da linha é o operador dois níveis  $\hat{\Psi}_i = \hat{\Psi}_i^{(2)}(\hat{\Psi}_1^{(1)}, \hat{\Psi}_2^{(1)}, \dots, \hat{\Psi}_i^{(1)})$ . O operador dois níveis é aplicado às imagens  $\mathbb{S}$  e as imagens provenientes desta aplicação são comparadas às imagens em  $\mathbb{I}$ , obtendo o  $MAE_i$  (Linha 10) usado para identificar a melhor combinação avaliada.

---

**Algoritmo 6:** WES: Seleção de janelas por ordenação da entropia condicional corrigida.

---

```

Entrada: CJ,  $\mathbb{S}^{(1)}, \mathbb{I}^{(1)}, \mathbb{S}^{(2)}, \mathbb{I}^{(2)}, \mathbb{S}, \mathbb{I}, J$ ,
Saída: Operador dois níveis, janelas selecionadas.
1   $\{W_{a_1}, W_{a_2}, \dots, W_{a_{|CJ|}}\} \leftarrow$  Ordenar crescentemente CJ por  $\hat{H}_c(y|X)$  ( $\mathbb{S}^{(1)}, \mathbb{I}^{(1)}$ ) ;
2   $g \leftarrow 2$  ;
3   $MAE_{menor} \leftarrow \infty$  ;
4   $i \leftarrow 2$  ;
5   $\hat{\Psi}_1^{(1)} \leftarrow$  Treinamento do primeiro nível ( $W_{a_1}, \mathbb{S}^{(1)}, \mathbb{I}^{(1)}$ ) ;
6  enquanto  $i \leq J$  faça
7       $\hat{\Psi}_i^{(1)} \leftarrow$  Treinamento do primeiro nível ( $W_{a_i}, \mathbb{S}^{(1)}, \mathbb{I}^{(1)}$ ) ;
8       $\hat{\Psi}_i \leftarrow$  Treinamento do segundo nível ( $\{\hat{\Psi}_1^{(1)}, \dots, \hat{\Psi}_i^{(1)}\}, \mathbb{S}^{(2)}, \mathbb{I}^{(2)}$ ) ;
9       $\hat{\Psi}_i(\mathbb{S}) \leftarrow$  Aplicar operador( $\hat{\Psi}_i, \mathbb{S}$ ) ;
10      $MAE_i \leftarrow$  Calcular MAE( $\hat{\Psi}_i(\mathbb{S}), \mathbb{I}$ ) ;
11     se  $MAE_i < MAE_{menor}$  então
12          $g \leftarrow i$  ;
13          $\hat{\Psi} \leftarrow \hat{\Psi}_i$  ;
14     fim se
15 fim enqto
16 devolva  $\hat{\Psi}, \{W_{a_1}, W_{a_2}, \dots, W_{a_g}\}$  ;

```

---

### ► Consumo de tempo do algoritmo

Devido às linhas 5, 7 e 8 o algoritmo é de ordem exponencial no pior caso.

#### 5.3.1 Comentários sobre a implementação da WES

O treinamento e aplicação dos operadores e o cálculo do MAE utilizou a biblioteca TRIOS: Training Image Operators from Samples (Montagner, 2015). Para facilitar a integração com a biblioteca, o Algoritmo 6, utilizado nos experimentos deste trabalho, foi desenvolvido em Python, versão 2.7 (Pyt, 2015). A coleção de janelas é criada por uma implementação em Java (Oracle, 2015).

#### 5.3.2 Validade da ordenação de uma CJ pela WES

Para avaliar se o valor atribuído a uma janela pela  $\hat{H}_c(y|X)$  é capaz de qualificá-la para uma combinação de primeiro nível, três testes foram realizados:

1. Executou-se a WES para as  $CJB_f$  de alguns problemas.
2. Para cada problema foram escolhidas 15 janelas aleatoriamente da  $CJB_f$ . Tomou-se as duas primeiras janelas, projetou-se o operador e calculou-se o MAE sobre as imagens  $\mathbb{S}, \mathbb{I}$ . Esse processo foi repetido até a décima quinta janela. A combinação que apresentou o menor MAE foi indicada como a eleita do processo.
3. Executou-se a WES, mas realizando a ordenação decrescente.

Espera-se que os operadores resultantes das seleções por ordenação sejam superiores aos outros. Testes comparativos com as combinações manuais de Hirata (2009) são apresentados no Capítulo 6.

## Experimentos

Na Tabela 5.1 estão elencados os conjuntos de dados usados nos testes deste capítulo. As imagens em  $\mathbb{S}^{(1)}$  são usadas no cálculo da entropia condicional corrigida e, juntamente com as imagens em  $\mathbb{I}^{(1)}$ , são usadas no treinamento do primeiro nível. As imagens em  $(\mathbb{S}^{(2)}, \mathbb{I}^{(2)})$  são utilizadas no treinamento do segundo nível e as combinações são avaliadas sobre  $(\mathbb{S}, \mathbb{I})$ .

**Tabela 5.1:** Descrição dos conjuntos de dados usados nos testes 1, 2 e 3. Fonte: Hirata (2009).

Conjunto de dados	Descrição	Imagens de treinamento		
		Primeiro nível ( $\mathbb{S}^{(1)}, \mathbb{I}^{(1)}$ )	Segundo nível ( $\mathbb{S}^{(2)}, \mathbb{I}^{(2)}$ )	MAE WES ( $\mathbb{S}, \mathbb{I}$ )
<i>Char_s</i>	Extração do caractere <i>s</i> em textos.	6	4	10
<i>CircCirc</i>	Extração de círculos em imagens de circuitos eletrônicos.	5	3	10
<i>TexCirc</i>	Extração de texto em imagens de circuitos eletrônicos	5	3	10
<i>RuidoBool</i>	Filtragem de ruído booleano	3	2	5

Além das imagens, outro parâmetro dos testes é a coleção de janelas. Essa foi uma coleção de janelas básicas fixas  $CJB_f$ , como definida no Capítulo 3, em que o domínio foi o mesmo utilizado para as combinações manuais<sup>1</sup>. Para cada teste, dez execuções da WES foram realizadas.

Na Tabela 5.2 pode ser observada uma comparação entre os resultados dos operadores das combinações definidas aleatoriamente, das encontradas pela ordenação crescente e pela ordenação decrescente. Nessa tabela estão dispostos os MAEs calculados a partir das imagens decorrentes da aplicação dos operadores projetados sobre o conjunto de imagens  $\mathbb{S}$ . Também está relacionado para cada execução, subscrito ao MAE, o número de janelas encontrado para o primeiro nível.

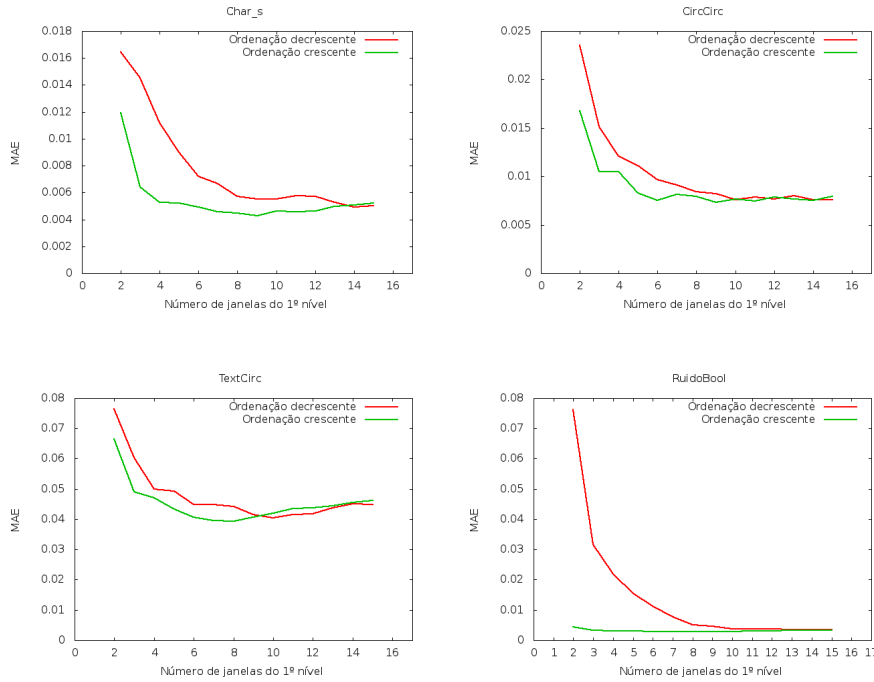
**Tabela 5.2:** MAEs de teste resultantes da aplicação dos operadores definidos por escolha aleatória; por operadores definidos pela WES e pela versão decrescente da WES.

Execução	MAE sobre $(\mathbb{S}, \mathbb{I})$			
	<i>Char_s</i>	<i>CircCirc</i>	<i>TexCirc</i>	<i>RuidoBool</i>
1	0.004526 <sub>10</sub>	0.006862 <sub>15</sub>	0.041663 <sub>8</sub>	0.003099 <sub>12</sub>
2	0.004799 <sub>11</sub>	0.007444 <sub>14</sub>	0.041818 <sub>10</sub>	0.003116 <sub>7</sub>
3	0.004551 <sub>12</sub>	0.007153 <sub>15</sub>	0.043571 <sub>13</sub>	0.002779 <sub>9</sub>
4	0.004839 <sub>11</sub>	0.006394 <sub>13</sub>	0.041993 <sub>13</sub>	0.003249 <sub>14</sub>
5	0.004531 <sub>12</sub>	0.007309 <sub>14</sub>	0.040085 <sub>11</sub>	0.003157 <sub>14</sub>
6	0.004713 <sub>8</sub>	0.006862 <sub>13</sub>	0.039494 <sub>9</sub>	0.002631 <sub>7</sub>
7	0.004713 <sub>11</sub>	0.007454 <sub>15</sub>	0.039620 <sub>9</sub>	0.002896 <sub>8</sub>
8	0.004627 <sub>11</sub>	0.008161 <sub>15</sub>	0.042661 <sub>9</sub>	0.003101 <sub>15</sub>
9	0.004768 <sub>10</sub>	0.007038 <sub>15</sub>	0.040666 <sub>7</sub>	0.002959 <sub>13</sub>
10	0.004541 <sub>10</sub>	0.007423 <sub>15</sub>	0.041925 <sub>14</sub>	0.002960 <sub>11</sub>
MAE WES crescente	0.004460 <sub>8</sub>	0.006279 <sub>13</sub>	0.039465 <sub>7</sub>	0.002420 <sub>9</sub>
MAE WES decrescente	0.004945 <sub>14</sub>	0.007569 <sub>14</sub>	0.040414 <sub>10</sub>	0.002829 <sub>8</sub>

Na Figura 5.2 são exibidos gráficos mostrando o comportamento dos operadores resultantes da seleção por ordenação crescente e decrescente sobre imagens de teste.

<sup>1</sup>Valores extraídos de Hirata (2009)





**Figura 5.2:** Comportamento dos operadores resultantes das combinações obtidas pela ordenação crescente e decrescente para as duas primeiras janelas da ordenação, para as três primeiras janelas, até as quinze primeiras janelas.

Também foi realizada uma comparação entre a entropia condicional corrigida  $\hat{H}_c(y|X)$  proposta por este trabalho e a entropia condicional corrigida proposta por Reis (2012)  $\hat{H}_{cR}(y|X)$ , como critérios de ordenação sobre uma coleção de janelas  $CJB_f$  com domínios iguais aos definidos em Hirata (2009).

**Tabela 5.3:** Comparação dos MAEs sobre  $(\mathbb{S}, \mathbb{I})$  obtidos pelos operadores das combinações WES com  $CJB_f$  utilizando  $\hat{H}_c(y|X)$  e  $\hat{H}_{cR}(y|X)$ .

Dados	Domínio de uma janela	MAE sobre $(\mathbb{S}_t, \mathbb{I}_t)$	
		WES com $\hat{H}_c(y X)$	WES com $\hat{H}_{cR}(y X)$
<i>Char_s</i>	$9 \times 7$	0.004460 <sub>8</sub>	0.00409 <sub>7</sub>
<i>CircCirc</i>	$9 \times 9$	0.006279 <sub>13</sub>	0.00707 <sub>15</sub>
<i>TexCirc</i>	$9 \times 9$	0.039465 <sub>7</sub>	0.04241 <sub>15</sub>
<i>RuídoBool</i>	$9 \times 9$	0.002420 <sub>9</sub>	0.00315 <sub>15</sub>
<i>TexRev</i>	$9 \times 7$	0.037389 <sub>10</sub>	0.03928 <sub>9</sub>

## Discussões

Pode-se observar na Tabela 5.2 e nas Figuras ?? e 5.2 que o operador resultante da metodologia de ordenação crescente pela  $\hat{H}_c(y|X)$  é superior à seleção aleatória e decrescente. Tanto em relação ao número de janelas determinado para o primeiro nível quanto à qualidade do operador treinado. A ordenação decrescente encontrou, na maioria dos casos, um número maior de janelas. Isso não é bom, pois eleva o tempo de execução de treinamento do respectivo operador. A escolha aleatória, nas dez execuções, sempre selecionou combinações piores que a WES.

Os resultados apresentados na Tabela 5.3 mostram que em 80% dos casos o uso da  $\hat{H}_c(y|X)$  fornece combinações de melhor qualidade do que a  $\hat{H}_{cR}(y|X)$ , validando seu uso como critério de

ordenação.

Os testes com a metodologia de ordenação foram executados em uma máquina com o sistema operacional Ubuntu 11.04/Linux e arquitetura com quatro processadores Intel i5-2410M de 2.30GHz cada um. O menor tempo de execução foi para *Char\_s* com 4 minutos e o maior tempo para o teste 2 com *RuidoBool* que levou 6 horas e 57 minutos. Esse tempo foi para uma execução do procedimento.

Do exposto, a ordenação pela  $\hat{H}_c(y|X)$  fornece uma combinação mais enxuta e consequentemente mais eficiente, com uma qualidade superior à escolha da ordenação decrescente, aleatória e do uso da  $\hat{H}_{c_R}(y|X)$ , sugerindo sua utilização na seleção de janelas.

## Capítulo 6

# Avaliação das metodologias propostas

Neste capítulo são apresentados experimentos na intenção de avaliar o WGA e a WES. As duas metodologias de seleção também são comparadas e uma avaliação das coleções de janelas e domínio automático é realizada.

### 6.1 Conjuntos de dados

Os dados utilizados nos experimentos estão expostos na Tabela 6.1. Parte desses dados foram usados na determinação dos parâmetros do WGA e na validação da WES. Lembrando que eles foram extraídos do trabalho de [Hirata \(2009\)](#).

Nas colunas *Primeiro nível* e *Segundo nível* estão relacionados os números de imagens usadas nesses níveis para a execução de cada teste.  $\mathbb{S}^{(1)}$  e  $\mathbb{I}^{(1)}$  representam as imagens de entrada e respectivas imagens ideais para o treinamento do primeiro e  $\mathbb{S}^{(2)}$  e  $\mathbb{I}^{(2)}$  as imagens para o treinamento do segundo nível.

Para mensurar a qualidade de uma combinação, o MAE é calculado baseando-se no resultado da aplicação do operador projetado sobre as imagens do conjunto  $\mathbb{S}$ , mais as correspondentes imagens ideais em  $\mathbb{I}$ .

$\mathbb{S}_t, \mathbb{I}_t$  representam imagens de teste. Elas são utilizadas para cálculo do MAE na intenção de verificar se o operador é eficaz em imagens diferentes das que foram utilizadas no treinamento e seleção.

As imagens dos conjuntos de dados foram obtidas através de um scanner de mesa e foram convertidas a resolução de 100dpi e podem ser acessadas em ([Hirata, 2015](#)). O Apêndice A apresenta exemplos de pares de imagens entrada/ideal para cada conjunto de dados da Tabela 6.1.

Uma observação deve ser feita sobre o conjunto *Mapa*. Para esse, não foi possível obter imagens para compor o conjunto  $\mathbb{S}_t$  (última coluna da Tabela 6.1).

**Tabela 6.1:** Descrição dos conjuntos de dados usados nos experimentos deste capítulo. Fonte\*: Hirata (2009).

Conjunto de dados	Descrição	Imagens de treinamento		Imagens para cálculo do MAE de qualificação ( $\mathbb{S}, \mathbb{I}$ )	Imagens de teste ( $\mathbb{S}_t, \mathbb{I}_t$ )
		Primeiro nível ( $\mathbb{S}^{(1)}, \mathbb{I}^{(1)}$ )	Segundo nível ( $\mathbb{S}^{(2)}, \mathbb{I}^{(2)}$ )		
<i>Char_s</i>	Extração do caractere <b>s</b> em textos.	6	4	10	5
<i>CircCirc</i>	Extração de círculos em imagens de circuitos eletrônicos.	5	3	10	2
<i>TexCirc</i>	Extração de texto em imagens de circuitos eletrônicos	5	3	10	2
<i>RuidoBool</i>	Filtragem de ruído booleano	3	2	5	5
<i>TexRev</i>	Segmentação de texto de páginas de revistas	3	2	5	5
<i>BoxCirc</i>	Extração de retângulos tracejados em imagens de circuitos eletrônicos.	5	3	10	2
<i>Mapa</i>	Segmentação de regiões em mapas com uma dada textura.	2	1	2	–

\*As imagens em  $\mathbb{S}_t, \mathbb{I}_t$  não foram usadas em Hirata (2009).

## 6.2 Experimentos com o WGA

Esta seção tem por objetivo verificar a eficácia de combinações de janelas selecionadas pelo WGA em relação as respectivas combinações manuais de (Hirata, 2009), utilizando a coleção de janelas fixas  $CJB_f$ , a  $CJI$  e a  $CJB_fI$ , todas propostas no Capítulo 3.

Os experimentos foram realizados em uma máquina com o sistema operacional Ubuntu 11.04/Linux e arquitetura com quatro processadores Intel i5-2410M de 2.30GHz cada um.

Três baterias de testes foram realizadas:

**Teste WGA\_CJI :** executou-se o WGA com a  $CJI$ . O número de janelas da coleção é igual a  $l = \text{linhas} * c = \text{colunas}$ , em que  $l$  e  $c$  são as dimensões do domínio.

**Teste WGA\_CJB<sub>f</sub> :** executou-se o WGA com a  $CJB_f$ . Lembrando que a coleção possui 24 janelas.

**Teste WGA\_CJB<sub>f</sub>I:** executou-se o WGA com a  $CJB_fI = CJB_f \cup CJI$ .

O domínio das janelas das CJs adotado nos testes foi igual ao usado manualmente. Os parâmetros utilizados na execução dos três testes foram aqueles definidos no Capítulo 4:  $\mathbf{pc} = 0.8$ ;  $\mathbf{pm} = 0.01$ ;  $\mathbf{maxG} = 200$ ;  $\mathbf{maxGsemA} = 50$ ;  $\mathbf{tp} = 10$ , o número máximo de janelas de um cromossomo  $\mathbf{maxJ} = 15$ , o número mínimo  $\mathbf{minJ} = 3$ , os conjuntos de imagens ( $\mathbb{S}^{(1)}, \mathbb{I}^{(1)}$ ), ( $\mathbb{S}^{(2)}, \mathbb{I}^{(2)}$ ) e ( $\mathbb{S}, \mathbb{I}$ ).

### 6.2.1 Resultados e discussões

Na Tabela 6.2 estão expostos os MAEs referentes ao trabalho de Hirata (2009) e do WGA, calculados para as imagens em ( $\mathbb{S}, \mathbb{I}$ ). Comparamos alguns poucos resultados com os de Santos (2010). O domínio das janelas está relacionado na segunda coluna da tabela. As quantidades de janelas no primeiro nível estão subscritas ao valor dos MAEs. Um número subscrito com  $b$  indica a quantidade de janelas selecionadas em  $CJB_f$ . Um número subscrito por  $i$ , a quantidade de janelas em  $CJI$ . Os MAEs destacados foram os melhores obtidos entre os três grupos de testes. Observa-se que as combinações encontradas pelo WGA são superiores às definidas manualmente.

**Tabela 6.2:** Comparação dos MAEs sobre  $(\mathbb{S}, \mathbb{I})$  obtidos pelos operadores das combinações manual,  $WGA\_CJI$ ,  $WGA\_CJB_f$  e  $WGA\_CJB_fI$ .

Dados	Domínio de uma janela	MAE sobre $(\mathbb{S}, \mathbb{I})$				
		Combinação manual	Metodologia de Santos (2010)	Combinação $WGA\_CJI$	Combinação $WGA\_CJB_f$	Combinação $WGA\_CJB_fI$
<i>Char_s</i>	$9 \times 7$	0.00552 <sub>5</sub>	0.0073 <sub>7</sub>	0.00653 <sub>8</sub>	0.00396 <sub>8</sub>	0.00383 <sub>8_b, 2_i</sub>
	$11 \times 9$	0.00425 <sub>5</sub>	—	0.00420 <sub>9</sub>	0.00346 <sub>9</sub>	0.00336 <sub>5_b, 4_i</sub>
<i>Mapa</i>	$9 \times 9$	0.04395 <sub>8</sub>	0.0600 <sub>8</sub>	0.03750 <sub>10</sub>	0.03431 <sub>10</sub>	0.03318 <sub>6_b, 4_i</sub>
<i>CircCirc</i>	$9 \times 9$	0.00756 <sub>6</sub>	—	0.00557 <sub>15</sub>	0.00559 <sub>15</sub>	0.00529 <sub>2_b, 13_i</sub>
<i>TexCirc</i>	$9 \times 9$	0.03977 <sub>6</sub>	—	0.04002 <sub>8</sub>	0.03384 <sub>8</sub>	0.03561 <sub>5_b, 3_i</sub>
<i>RuidoBool</i>	$9 \times 9$	0.00347 <sub>5</sub>	—	0.00329 <sub>8</sub>	0.00234 <sub>8</sub>	0.00223 <sub>6_b, 2_i</sub>
<i>TexRev</i>	$9 \times 7$	0.04635 <sub>5</sub>	—	0.04164 <sub>3</sub>	0.03197 <sub>3</sub>	0.03197 <sub>3_b, 0_i</sub>

A Tabela 6.3 permite verificar se o operador projetado fornece bons resultados para imagens diferentes daquelas usadas na função de avaliação do WGA.

**Tabela 6.3:** Comparação dos MAEs sobre  $(\mathbb{S}_t, \mathbb{I}_t)$  obtidos pelos operadores das combinações manual, metodologia de Santos (2010),  $WGA\_CJI$ ,  $WGA\_CJB_f$  e  $WGA\_CJB_fI$ . Subscritas estão as quantidades de janelas do primeiro nível.

Dados	Domínio de uma janela	MAE sobre $(\mathbb{S}_t, \mathbb{I}_t)$			
		Combinação manual	Combinação $WGA\_CJI$	Combinação $WGA\_CJB_f$	Combinação $WGA\_CJB_fI$
<i>Char_s</i>	$9 \times 7$	0.00561 <sub>5</sub>	0.00672 <sub>8</sub>	0.00461 <sub>8</sub>	0.00433 <sub>8_b, 2_i</sub>
	$11 \times 9$	0.00501 <sub>5</sub>	0.00482 <sub>9</sub>	0.00402 <sub>9</sub>	0.00401 <sub>5_b, 4_i</sub>
<i>Mapa</i>	$9 \times 9$	0.04395 <sub>8</sub>	0.03750 <sub>10</sub>	0.03431 <sub>10</sub>	0.03318 <sub>6_b, 4_i</sub>
<i>CircCirc</i>	$9 \times 9$	0.00651 <sub>6</sub>	0.00461 <sub>15</sub>	0.00510 <sub>15</sub>	0.00396 <sub>2_b, 13_i</sub>
<i>TexCirc</i>	$9 \times 9$	0.04196 <sub>6</sub>	0.04685 <sub>8</sub>	0.03784 <sub>8</sub>	0.04147 <sub>5_b, 3_i</sub>
<i>RuidoBool</i>	$9 \times 9$	0.00331 <sub>5</sub>	0.21672 <sub>8</sub>	0.00224 <sub>8</sub>	0.00215 <sub>6_b, 2_i</sub>
<i>TexRev</i>	$9 \times 7$	0.02600 <sub>5</sub>	0.03280 <sub>3</sub>	0.01248 <sub>3</sub>	0.01248 <sub>3_b, 0_i</sub>

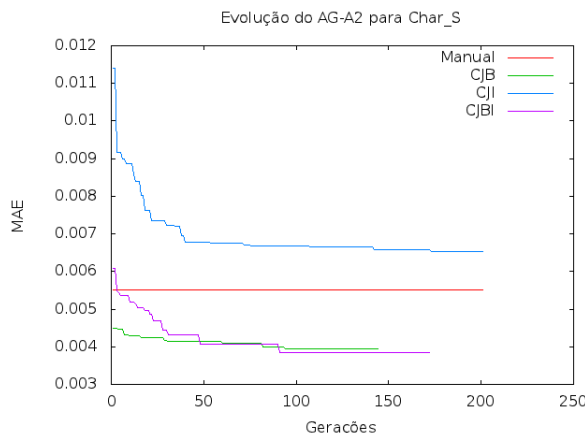
Quando se confronta os MAEs sobre  $(\mathbb{S}, \mathbb{I})$  obtidos mediante os operadores definidos pela metodologia de Santos (2010), com os MAEs obtidos pelo  $WGA\_CJI$ , esse último apresentou melhores resultados. Apesar do número pequeno de comparações entre as metodologias, esses resultados indicam que a abordagem usada em Santos (2010) para o pós-processamento das janelas e escolha de uma combinação pode ser aprimorada.

Observando os testes relacionados ao conjunto de dados *Char\_s*, os resultados encontrados pelo  $WGA\_CJB_f$  e  $WGA\_CJB_fI$ , para um domínio  $9 \times 7$ , são melhores que o respectivo resultado manual e melhores que o resultado manual para o domínio  $11 \times 9$ . Esse fato sinaliza a eficácia do WGA em relação às escolhas manuais.

É possível verificar na Tabela 6.3 que em 83% dos testes, os melhores resultados foram obtidos pelo  $WGA\_CJB_fI$ . Observa-se ainda que uma escolha baseada em um conjunto de janelas prefixado, seja um  $CJB_f$  ou  $CJB_fI$ , sempre forneceu um resultado melhor que o manual, apontando que a metodologia proposta é superior a escolha manual e que as coleções de janelas básicas atendem a problemas de contextos diferentes.

Verifica-se ainda que o ganho com o uso da  $CJB_fI$  sobre a  $CJB_f$ , em termos de MAE, é pequeno e que o número de janelas selecionadas pelo  $WGA\_CJB_f$  e  $WGA\_CJB_fI$ , na maioria das vezes foi igual. Pensando na eficiência do WGA, a escolha pela  $CJB_f$  seria uma opção razoável e o espaço de busca seria menor.

Na Figura 6.1 são exibidas a evolução dos melhores cromossomos para cada coleção de janelas nas gerações do WGA para o conjunto de dados *Char\_s*. Verifica-se que a  $CJB_f$  e a  $CJB_fI$  possuem resultados próximos, enquanto que para a  $CJI$  os resultados estão acima da combinação manual.



**Figura 6.1:** MAE sobre  $S$  e  $I$  do melhor cromossomo de cada geração do WGA para o conjunto de dados *Char\_s*.

Um fato interessante é o resultado obtido nos testes  $WGA\_CJB_f$  e  $WGA\_CJB_fI$  para *TexRev*. Eles encontraram o mesmo conjunto de janelas como solução e esse conjunto é menor que o manual.

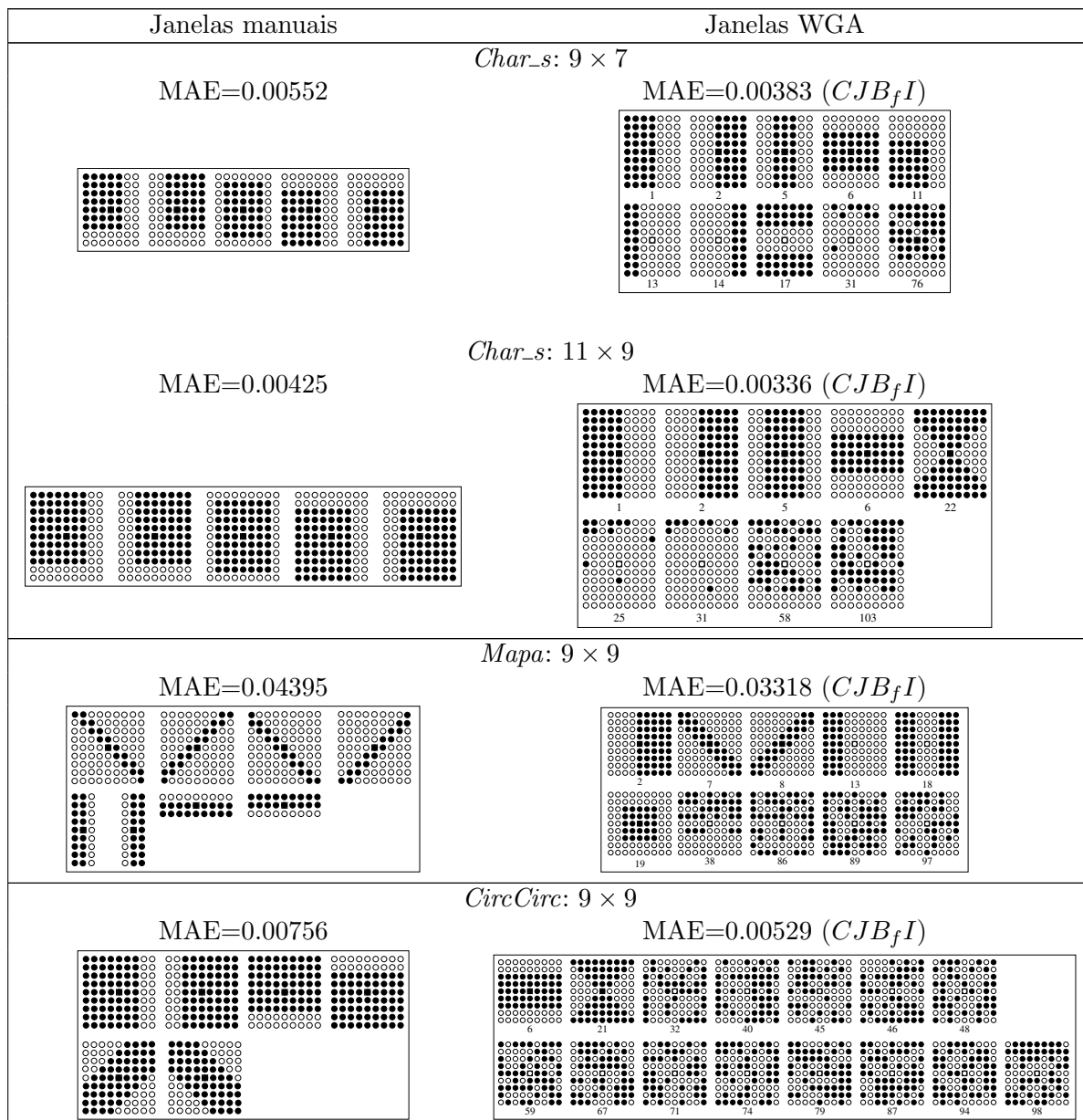
Para os outros conjuntos de dados o número de janelas obtido sempre foi maior. No caso do *CircCirc* o resultado do WGA foi um conjunto com 15 janelas. O máximo permitido. Será que o WGA consegue encontrar um cromossomo com um número menor de janelas e MAE igual ou inferior ao de 15 janelas? Para verificar isso, o WGA com  $CJB_fI$  foi executado novamente para *CircCirc* com  $maxJ = 10$ . O resultado encontrado apresentou o melhor indivíduo com 10 janelas e MAE de 0.00545. Esse resultado não foi melhor que o encontrado para  $maxJ = 15$ .

A Tabela 6.4 apresenta o tempo gasto em cada teste para cada conjunto de dados, o número de gerações executadas e o tamanho final da população (TFP). Apenas foi computado o tempo com o AG, desconsiderando o tempo com a obtenção da CJ e os operadores de primeiro nível. Vários fatores influenciam no tempo de execução do AG. Dentre eles estão o tamanho da população, o tamanho do indivíduo, o número de *crossovers* e mutações, mas a maior parte do tempo é devida a execução da função de avaliação que envolve o projeto dois níveis. Para alguns casos, um especialista deve levar um tempo razoável para determinar uma combinação de janelas para um primeiro nível. Infelizmente não existe o registro de tempo para as combinações manuais, impossibilitando uma comparação nesse sentido.

**Tabela 6.4:** Tempo em horas, número de gerações executadas pelo WGA e o tamanho final da população (TFP) para os testes. Os experimentos foram realizados em uma máquina com o sistema operacional Ubuntu 11.04/Linux e arquitetura com quatro processadores Intel i5-2410M de 2.30GHz cada um.

Conjunto de dados	Domínio de uma janela	WGA_CJI			WGA_CJB <sub>f</sub>			WGA_CJB <sub>f</sub> I		
		Tempo (hs)	Gerações	TFP	Tempo (h)	Gerações	TFP	Tempo (h)	Gerações	TFP
<i>Char_s</i>	9 × 7	3.1	200	50	1.7	143	37	3.2	171	58
	11 × 9	1.3	128	48	1.8	149	40	3.2	196	61
<i>Mapa</i>	9 × 9	5.8	200	64	3.7	65	22	3.4	144	47
<i>CircCirc</i>	9 × 9	0.3	84	45	0.8	200	48	0.5	119	58
<i>TexCirc</i>	9 × 9	2.8	200	54	2.3	170	48	1.3	111	34
<i>TexRev</i>	9 × 7	0.9	104	30	1.2	120	33	2.0	136	50
<i>RuidoBool</i>	9 × 9	1.8	137	38	1.8	157	37	2.1	147	47
<i>TexRev</i>	9 × 7	0.9	104	30	1.2	120	33	2.0	136	50

Os melhores cromossomos obtidos pelos testes são exibidos na Figura 6.2, bem como as janelas manuais e os MAEs para as imagens  $\mathbb{S}, \mathbb{I}$  que foram usados para selecionar as combinações.



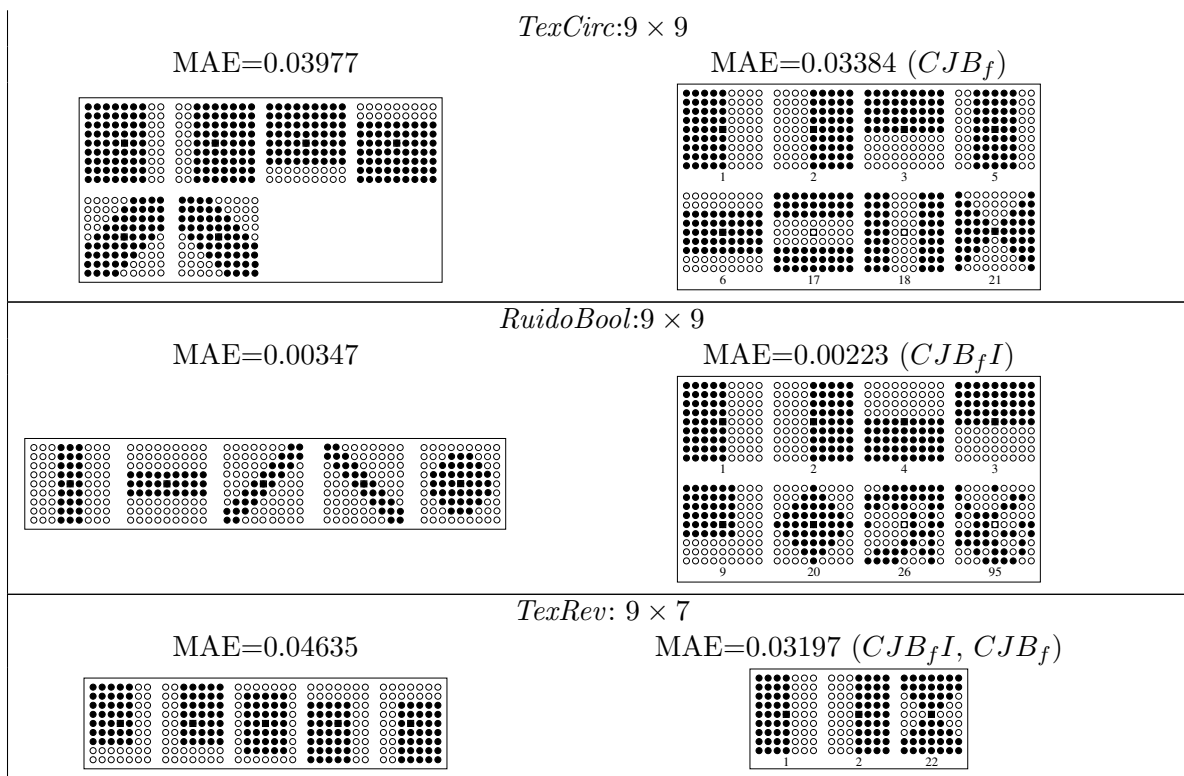


Figura 6.2: Janelas usadas nos testes de Hirata (2009) e as melhores janelas obtidas pelo WGA.

Fragmentos de imagens para cada conjunto de dados, resultantes da aplicação dos melhores operadores WGA sobre imagens em  $\mathbb{S}_t$  são apresentadas na Figura 6.3. As imagens completas podem ser visualizadas no Apêndice A.

Diante das discussões e resultados expostos, o WGA apresenta-se como uma metodologia viável para a seleção de combinações de janelas. Sua única desvantagem é o tempo de processamento, que pode ser reduzido com uso de processamento paralelo. Os resultados também mostraram que as coleções de janelas propostas são razoáveis e abrangentes para problemas de diferentes contextos.

Os resultados destes experimentos geraram um artigo (Dornelles e Hirata, 2012) aceito para o 21st International Conference on Pattern Recognition (ICPR) que ocorreu em novembro de 2012 no Japão.



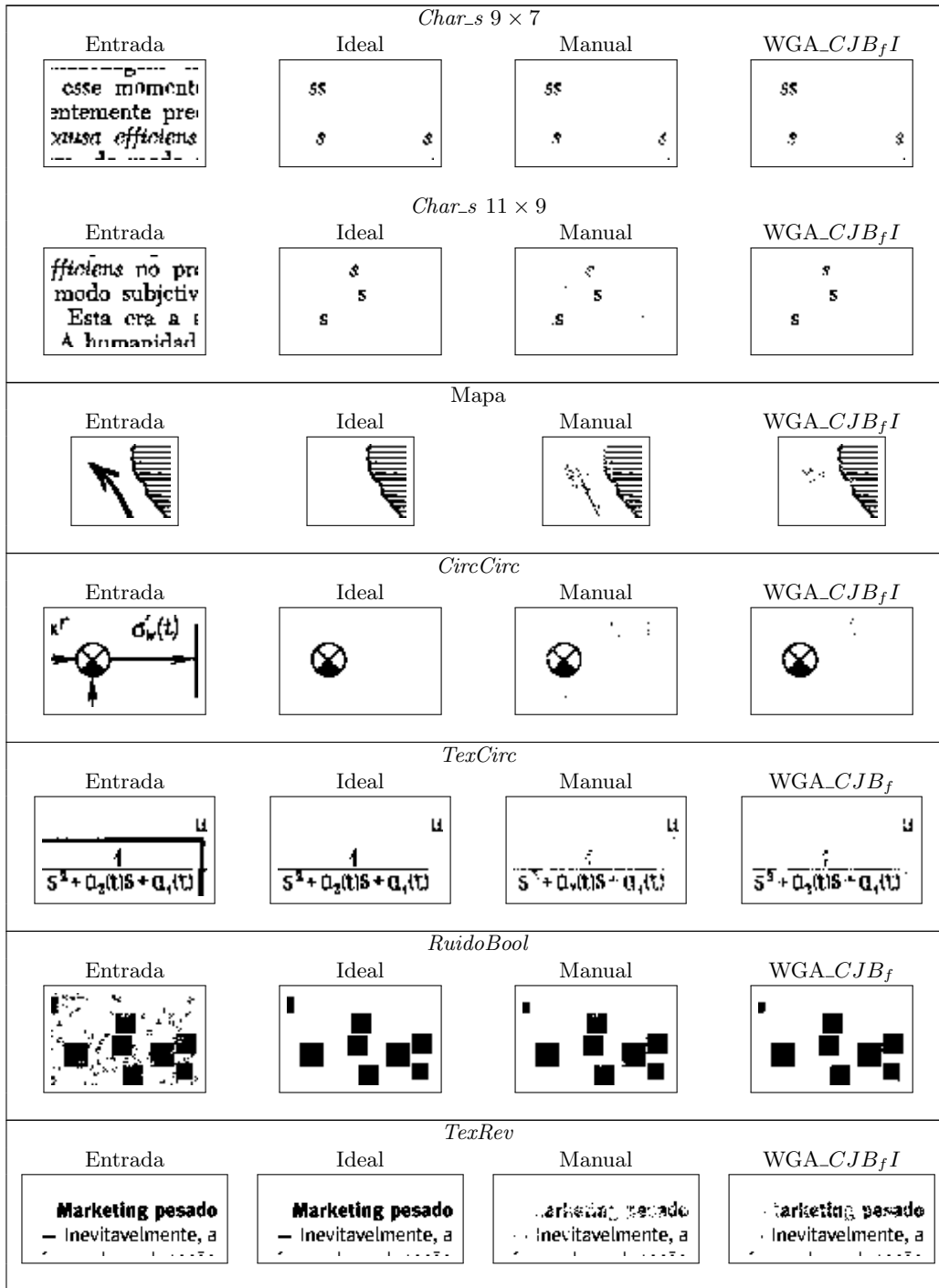


Figura 6.3: Fragmentos de uma imagem de entrada em  $S_t$  para cada conjunto de dados, suas respectivas imagens ideais, os resultados da aplicação dos operadores manuais e os resultados da aplicação dos operadores obtidos pela combinação do WGA.

### 6.3 Experimentos com a WES

O objetivo desta seção é realizar comparações da WES com as seleções manuais de Hirata (2009), bem como avaliar o desempenho da WES sobre as coleções de janelas propostas no Capítulo 3. Aqui foram utilizados os mesmos conjuntos de dados usados na seção anterior.

#### 6.3.1 WES versus seleção manual

Nestes experimentos a coleção de janelas básicas adotada foi a fixa em que o domínio foi igual ao utilizado na seleção manual. As coleções foram as mesmas utilizadas no WGA. Três baterias de testes foram realizadas:

**Teste WES\_CJI** : executou-se a WES com CJI.

**Teste WES\_CJB<sub>f</sub>** : executou-se a WES com o CJB<sub>f</sub>.

**Teste WES\_CJB<sub>f</sub>I**: executou-se a WES com o CJB<sub>f</sub>I = CJB<sub>f</sub> ∪ CJI.

Os parâmetros do algoritmo foram: a CJ de um dos testes, os conjuntos de imagens ( $\mathbb{S}^{(1)}, \mathbb{I}^{(1)}$ ), ( $\mathbb{S}^{(2)}, \mathbb{I}^{(2)}$ ) e ( $\mathbb{S}, \mathbb{I}$ ) e  $J = 15$ .

#### Resultados e discussões

Na Tabela 6.5 estão dispostos os MAEs das melhores combinações selecionadas pela WES. O cálculo ocorreu para as imagens obtidas pela aplicação dos operadores resultantes das combinações WES sobre ( $\mathbb{S}$ ). Os resultados mostram que a WES utilizando a CJB<sub>f</sub> sempre forneceu uma combinação melhor que à manual. A WES\_CJB<sub>f</sub> em 80% dos resultados apresentou uma seleção superior às outras duas coleções de janelas.

**Tabela 6.5:** Comparação dos MAEs sobre ( $\mathbb{S}, \mathbb{I}$ ) obtidos pelos operadores das combinações manual e WES\_CJI, WES\_CJB<sub>f</sub> e WES\_CJB<sub>f</sub>I.

Dados	Domínio de uma janela	MAE sobre ( $\mathbb{S}, \mathbb{I}$ )			
		Combinação manual	Combinação WES_CJI	Combinação WES_CJB <sub>f</sub>	Combinação WES_CJB <sub>f</sub> I
<i>Char_s</i>	9 × 7	0.00552 <sub>5</sub>	0.00717 <sub>8</sub>	0.00446 <sub>8</sub>	0.00435 <sub>6<sub>b</sub>,1<sub>i</sub></sub>
<i>CircCirc</i>	9 × 9	0.00756 <sub>6</sub>	0.00679 <sub>15</sub>	0.00628 <sub>13</sub>	0.00658 <sub>3<sub>b</sub>,7<sub>i</sub></sub>
<i>TexCirc</i>	9 × 9	0.03977 <sub>6</sub>	0.04245 <sub>8</sub>	0.03947 <sub>7</sub>	0.03979 <sub>6<sub>b</sub>,2<sub>i</sub></sub>
<i>RuidoBool</i>	9 × 9	0.00347 <sub>5</sub>	0.00346 <sub>7</sub>	0.00242 <sub>9</sub>	0.00251 <sub>6<sub>b</sub>,2<sub>i</sub></sub>
<i>TexRev</i>	9 × 7	0.04635 <sub>5</sub>	0.04320 <sub>3</sub>	0.03739 <sub>10</sub>	0.03996 <sub>4<sub>b</sub>,5<sub>i</sub></sub>

Com o objetivo de verificar se a WES apresenta uma eficácia tão boa quanto a seleção manual, uma comparação dos MAEs sobre as imagens de teste foi realizada. Os valores podem ser visualizados na Tabela 6.6. Com exceção de *TextCirc*, a metodologia de ordenação sempre forneceu uma combinação melhor que a manual. No caso de *TextCirc* não foi melhor, mas ficou bem próximo.

O tempo de execução da metodologia é influenciado pelo número de janelas em uma coleção, pela forma e tamanho das janelas e pela complexidade do problema. Na Tabela 6.7 estão expostos os tempos para os testes realizados.

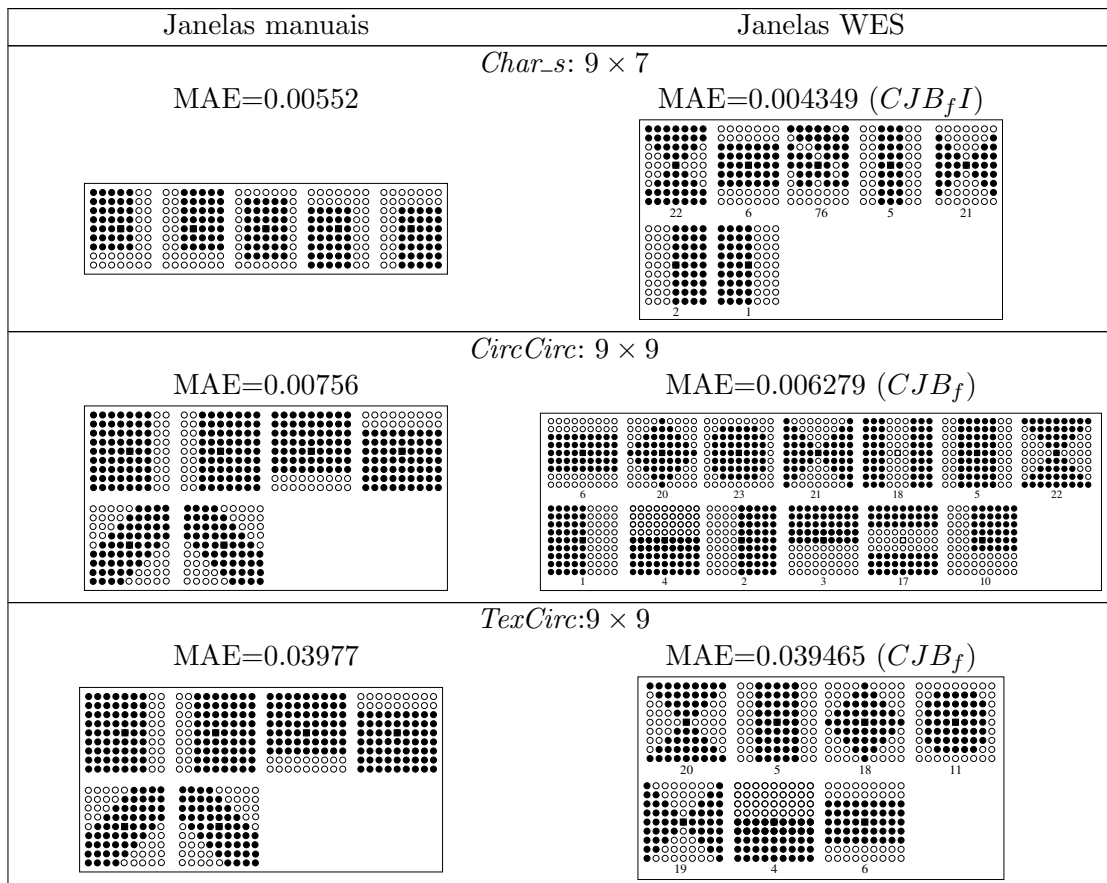
**Tabela 6.6:** Comparação dos MAEs sobre  $(S_t, I_t)$  obtidos pelos operadores das combinações de janelas manual,  $WES\_CJI$ ,  $WES\_CJB_f$  e  $WES\_CJB_fI$ .

Dados	Domínio de uma janela	Combinação manual	MAE sobre $(S_t, I_t)$		
			Combinação $WES\_CJI$	Combinação $WES\_CJB_f$	Combinação $WES\_CJB_fI$
<i>Char_s</i>	$9 \times 7$	0.00561 <sub>5</sub>	0.00734 <sub>8</sub>	0.00460 <sub>8</sub>	0.00466 <sub>6_b, 1_i</sub>
<i>CircCirc</i>	$9 \times 9$	0.00651 <sub>6</sub>	0.00562 <sub>15</sub>	0.00470 <sub>13</sub>	0.00571 <sub>3_b, 7_i</sub>
<i>TexCirc</i>	$9 \times 9$	0.04196 <sub>6</sub>	0.04794 <sub>8</sub>	0.04535 <sub>7</sub>	0.04648 <sub>6_b, 2_i</sub>
<i>RuidoBool</i>	$9 \times 9$	0.00331 <sub>5</sub>	0.00337 <sub>7</sub>	0.00225 <sub>9</sub>	0.00237 <sub>6_b, 2_i</sub>
<i>TexRev</i>	$9 \times 7$	0.02600 <sub>5</sub>	0.03580 <sub>3</sub>	0.02198 <sub>10</sub>	0.02490 <sub>4_b, 5_i</sub>

**Tabela 6.7:** Tempo de execução da WES com as coleções  $CJI$ ,  $CJB_f$  e  $CJB_fI$  em uma máquina com o sistema operacional Ubuntu 11.04/Linux e arquitetura com quatro processadores Intel i5-2410M de 2.30GHz cada um.

Dados	Domínio de uma janela	Tempo em minutos		
		Combinação $WES\_CJI$	Combinação $WES\_CJB_f$	Combinação $WES\_CJB_fI$
<i>Char_s</i>	$9 \times 7$	4, 95	4, 34	6, 56
<i>CircCirc</i>	$9 \times 9$	33, 34	17, 04	35, 41
<i>TexCirc</i>	$9 \times 9$	13, 59	60, 22	40, 47
<i>RuidoBool</i>	$9 \times 9$	142, 66	123, 70	127, 18
<i>TexRev</i>	$9 \times 7$	131, 36	238, 27	242, 83

As janelas do primeiro nível encontradas pela WES para os conjuntos de dados podem ser visualizadas na Figura 6.4. Os MAEs expostos são aqueles utilizados para a seleção das combinações de janelas  $(S, I)$ . As imagens mostram que, nos testes realizados, o número de janelas selecionado pela WES sempre foi superior à escolha manual.



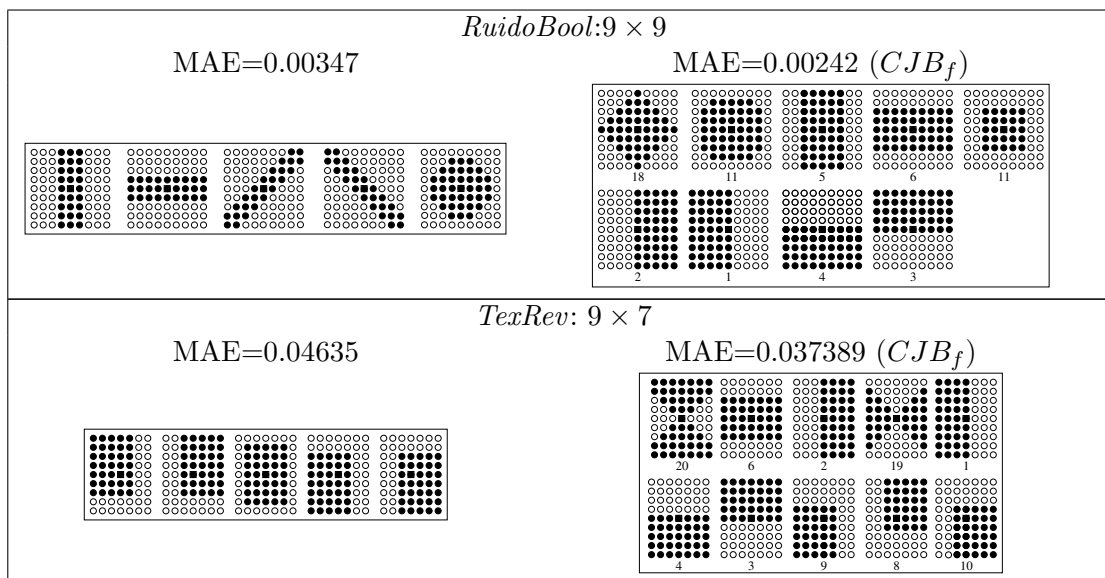
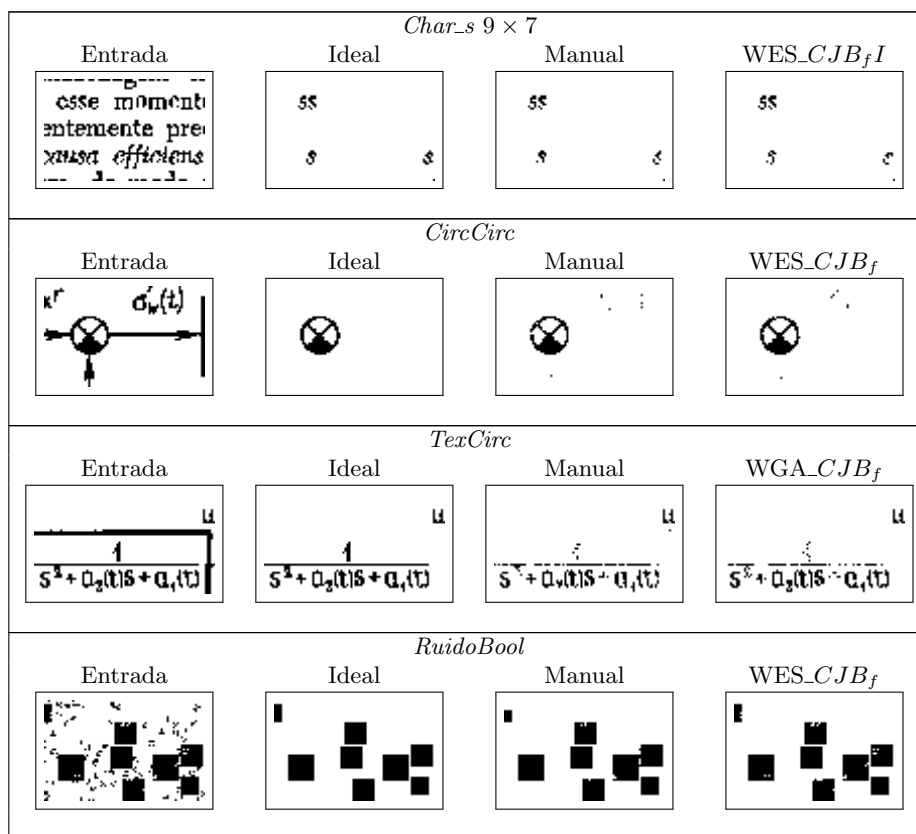
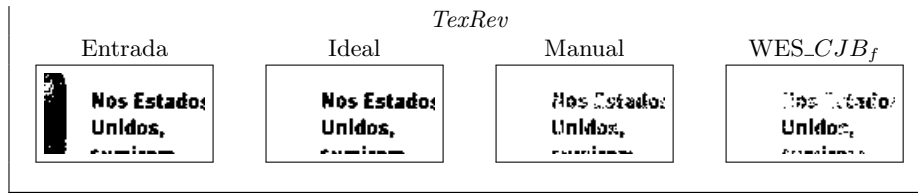


Figura 6.4: Janelas usadas nos testes de Hirata (2009) e as melhores janelas obtidas pela WES.

Na Figura 6.5 estão dispostos alguns fragmentos de imagens resultantes da aplicação dos melhores operadores obtidos pela WES e os correspondentes dos operadores manuais. Esses fragmentos mostram que os resultados são similares. As imagens de entrada e ideal pertencem a  $(\mathbb{S}_t, \mathbb{I}_t)$  e podem ser visualizadas na íntegra no Apêndice A.





**Figura 6.5:** Fragmentos de uma imagem de entrada em  $S_t$  para cada conjunto de dados, suas respectivas imagens ideais, os resultados da aplicação dos operadores manuais e os resultados da aplicação dos operadores obtidos pela combinação selecionada pela WES.

Diante dos resultados alcançados, a seleção de janelas ordenadas pela entropia condicional corrigida sobre uma coleção de janelas básicas se apresenta como uma alternativa à seleção manual. Sua utilização com uma  $CJB_f I$  também é uma alternativa, porém o uso com  $CJI$  pode não alcançar resultados satisfatórios. Provavelmente a coleção de janelas básicas contempla melhor a diversidade na combinação, já que as janelas são originalmente distintas entre si, o que pode não ser o caso de  $CJI$ .

### 6.3.2 Avaliação das coleções de janelas e domínio automático com a WES

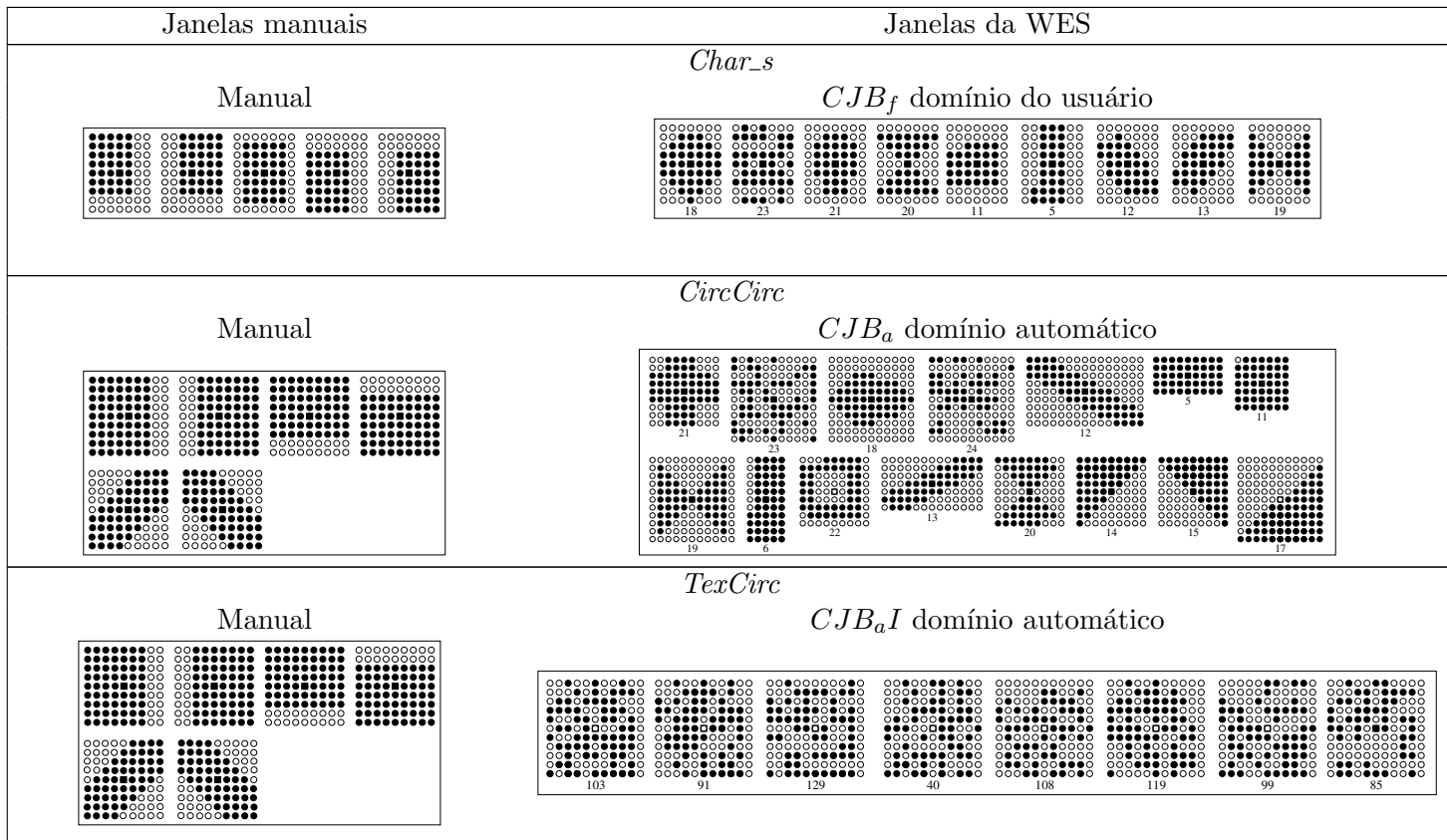
Nas seções anteriores os experimentos realizados com o WGA e a WES levaram em consideração o domínio definido pelo usuário. Esta seção procura avaliar o comportamento da WES com as coleções em que o tamanho das janelas é selecionado de um intervalo  $I_W$  e o domínio é automático, como definidos no Capítulo 3.

Nos experimentos foram criadas coleções de janelas baseadas nas três metodologias propostas: janelas básicas fixas  $CJB_f$  com domínio definido pelo usuário; janelas básicas com tamanho de janelas selecionado aleatoriamente em  $I_W$  ( $CJB_a$ ) e domínio definido pelo usuário; janelas básicas com tamanho de janelas selecionado aleatoriamente em  $I_W$  ( $CJB_a$ ) e domínio automático. Também foi utilizada a coleção  $CJI$  e a combinação dessa com as outras três descritas anteriormente.

Uma comparação dos MAEs sobre  $(S_t, I_t)$  obtidos pela aplicação dos operadores resultantes das combinações de janelas dadas pela WES, podem ser visualizados na Tabela 6.8. As melhores combinações de primeiro nível selecionadas estão na Figura 6.6. A metodologia de ordenação sempre forneceu uma combinação melhor que a manual. Em 60% dos testes, a ordenação utilizando o domínio automático e a seleção de tamanho em  $I_W$  forneceu os melhores resultados.

**Tabela 6.8:** Comparação dos MAEs sobre  $(S_t, I_t)$  obtidos pelos operadores das combinações manual e WES para as coleções de janelas obtidas com domínio definido pelo usuário e domínio automático.

Dados	DOM USU	Combinação manual	MAE sobre $(S_t, I_t)$							
			Domínio do usuário				Domínio automático			
			$CJB_f$	$CJB_f I$	$CJI$	$CJB_a$	$CJB_a I$	$CJI$	$CJB_a$	$CJB_a I$
<i>Char_s</i>	$9 \times 7$	0.00561 <sub>5</sub>	0.00460 <sub>8</sub>	0.00466 <sub>6,1_i</sub>	0.00734 <sub>8</sub>	0.00455 <sub>9</sub>	0.00479 <sub>9,1_i</sub>	0.00457 <sub>7</sub>	0.00544 <sub>12</sub>	0.00476 <sub>5,2_i</sub>
<i>CircCirc</i>	$9 \times 9$	0.00651 <sub>6</sub>	0.00470 <sub>13</sub>	0.00573 <sub>3,7_i</sub>	0.00562 <sub>15</sub>	0.00508 <sub>9</sub>	0.00497 <sub>5,8_i</sub>	0.00599 <sub>15</sub>	0.00411 <sub>15</sub>	0.00422 <sub>7,7_i</sub>
<i>TexCirc</i>	$9 \times 9$	0.04196 <sub>6</sub>	0.04535 <sub>7</sub>	0.04648 <sub>6,2_i</sub>	0.04794 <sub>8</sub>	0.04394 <sub>8</sub>	0.04394 <sub>8,0_i</sub>	0.03897 <sub>8</sub>	0.03973 <sub>8</sub>	0.03897 <sub>0,8_i</sub>
<i>RuidoBool</i>	$9 \times 9$	0.00331 <sub>5</sub>	0.00225 <sub>9</sub>	0.00237 <sub>6,2_i</sub>	0.00337 <sub>7</sub>	0.00268 <sub>78</sub>	0.00246 <sub>4,3_i</sub>	0.00337 <sub>7</sub>	0.00290 <sub>37</sub>	0.00264 <sub>6,3_i</sub>
<i>TexRev</i>	$9 \times 7$	0.02600 <sub>5</sub>	0.02198 <sub>10</sub>	0.02490 <sub>4,5_i</sub>	0.03580 <sub>3</sub>	0.02483 <sub>9</sub>	0.02525 <sub>8,3_i</sub>	0.04318 <sub>3</sub>	0.02026 <sub>13</sub>	0.02455 <sub>2,1_i</sub>



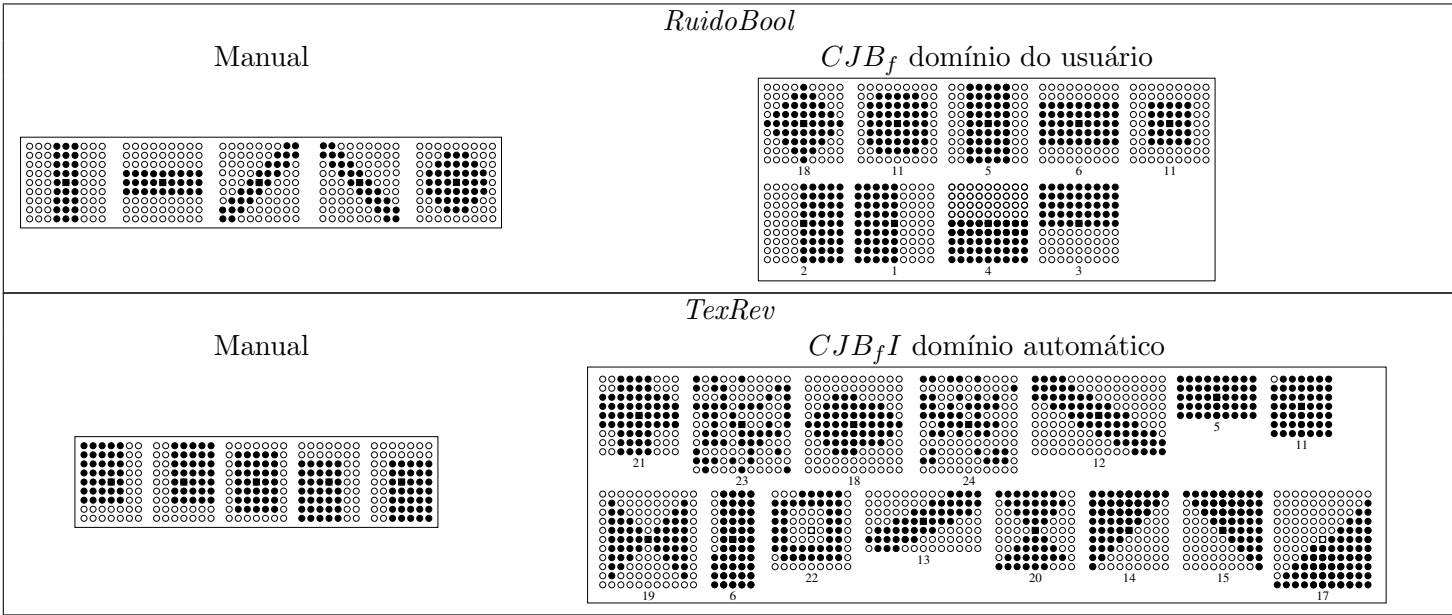


Figura 6.6: Janelas usadas nos testes de Hirata (2009) e as melhores combinações de janelas obtidas pela WES.

### 6.3.3 Testes complementares

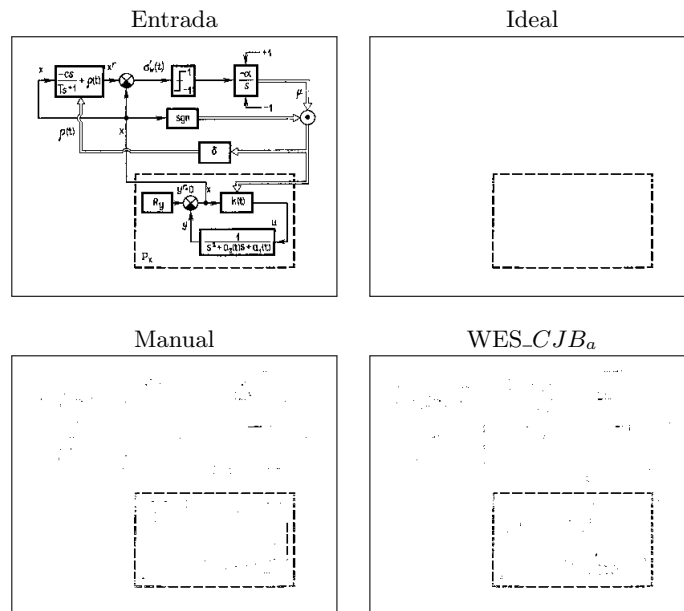
Dois conjuntos de dados não foram usados na comparação do WGA com a combinação de janelas manual devido ao tamanho do domínio ser grande ou não existirem imagens suficientes para  $\mathbb{S}_t$ . Esses conjuntos foram deixados para avaliar a WES com a  $CJB_a$  e domínio automático. São eles o *BoxCirc* e o *Mapa*. No caso do *Mapa*, como não haviam disponíveis imagens  $(\mathbb{S}_t, \mathbb{I}_t)$ , apenas comparações com as imagens  $(\mathbb{S}, \mathbb{I})$  foram realizadas.

Na Tabela 6.9 estão os MAEs e, em subscrito, o número de janelas encontrado pela WES. Nestes testes a metodologia também se mostrou eficaz. O interessante é que o procedimento automático definiu janelas maiores para o *BoxCirc* e janelas menores para o *Mapa*, acompanhando as janelas definidas pelo especialista. Esse resultado e os da seção anterior apontam que o intervalo de tamanho de janelas está funcionando bem. Além disso, eles sugerem que a WES com janelas básicas e domínio automático ( $CJB_a$ ) e uso de  $I_W$  é uma proposta promissora na formulação de uma metodologia automática para o projeto de W-operadores.

As janelas encontradas pela WES estão apresentadas na Figura 6.9 e imagens resultantes da aplicação dos operadores projetados para *BoxCirc* e *Mapa* a partir da combinação manual e WES estão nas Figuras 6.7 e 6.8.

**Tabela 6.9:** MAEs obtidos pelos operadores das combinações manual e WES.

Dados	Domínio manual de uma janela	MAE sobre $(\mathbb{S}, \mathbb{I})$		MAE sobre $(\mathbb{S}_t, \mathbb{I}_t)$	
		Combinação manual	WES $CJB_a$	Combinação manual	WES $CJB_a$
<i>BoxCirc</i>	$25 \times 25$	0.024345 <sub>6</sub>	0.028471 <sub>15</sub>	0.026003	0.026485
<i>Mapa</i>	$9 \times 9$	0.043178 <sub>8</sub>	0.037041 <sub>7</sub>	—	—



**Figura 6.7:** Uma imagem de entrada em  $\mathbb{S}_t$ , sua respectiva imagem ideal, o resultado obtido pela janelas manuais e o resultado obtido pelas janelas da  $WES.CJB_a$  com domínio automático para *BoxCirc*.



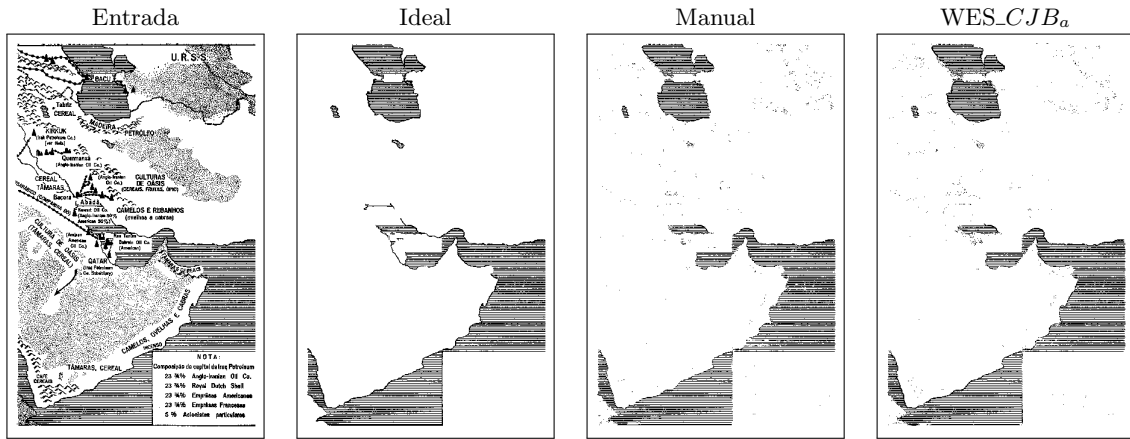


Figura 6.8: Uma imagem de entrada em S, sua respectiva imagem ideal, o resultado obtido pela janelas manuais e o resultado obtido pelas janelas da WES\_CJB<sub>a</sub> com domínio automático para Mapa.

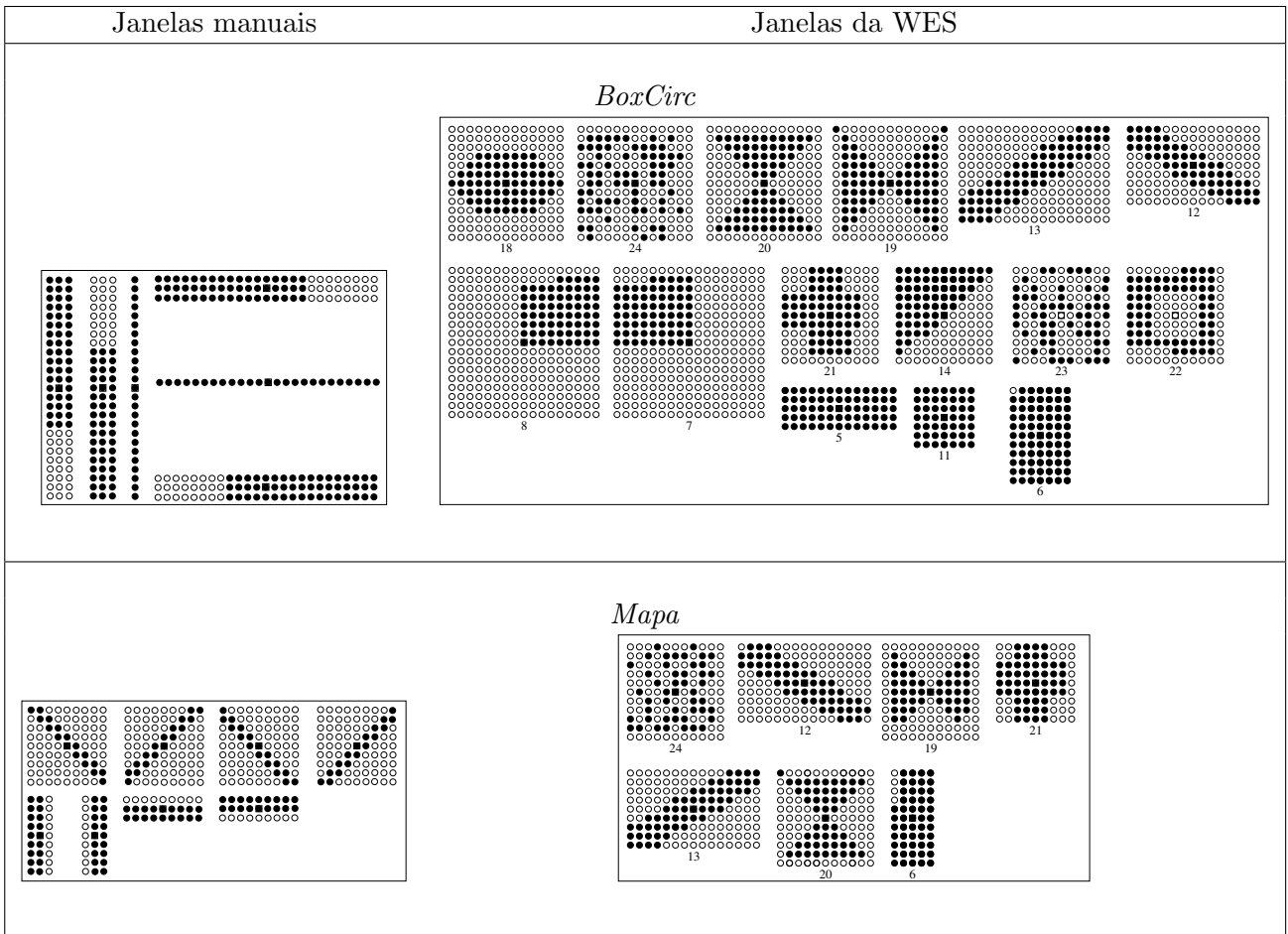


Figura 6.9: Janelas manuais e as janelas selecionadas pela WES\_CJB<sub>a</sub> para BoxCirc e Mapa.

## 6.4 WGA versus WES

Esta seção realiza uma comparação entre o WGA e a WES, agrupando os resultados das seções 6.2 e 6.3, a fim de verificar qual é a melhor metodologia de seleção de janelas. Na Tabela 6.10 estão os MAEs sobre as imagens  $(\mathbb{S}_t, \mathbb{I}_t)$ .

**Tabela 6.10:** MAEs sobre  $(\mathbb{S}_t, \mathbb{I}_t)$  obtidos pelos operadores das combinações manual, WGA e WES. O subscrito representa o número de janelas do primeiro nível, em que  $b$ =básica e  $i$ = janela obtida pela informação de interação.

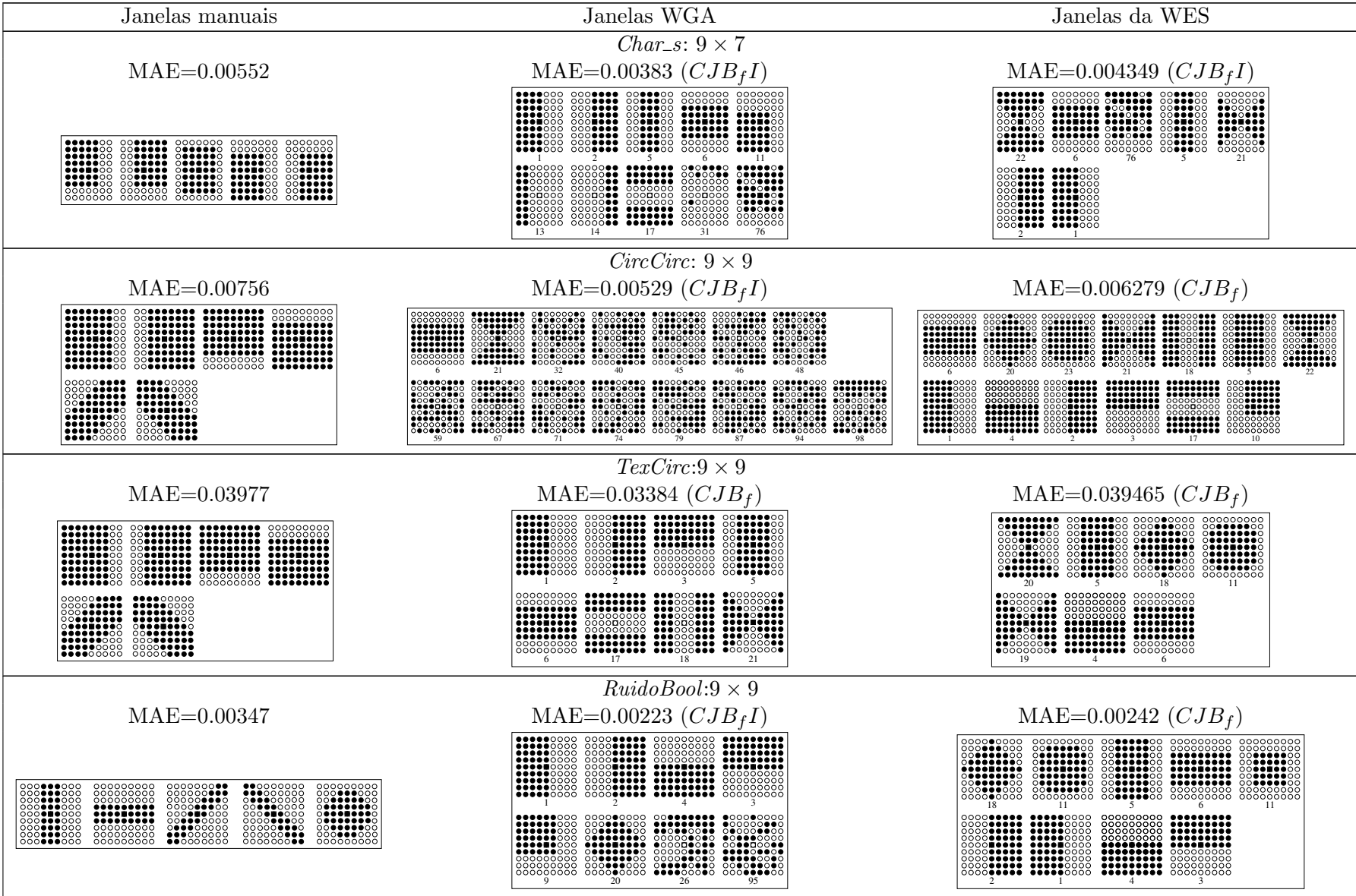
Dados	Domínio de uma janela	Combinação manual	MAE sobre $(\mathbb{S}_t, \mathbb{I}_t)$					
			CJI		$CJB_f$		$CJB_fI$	
			WGA	WES	WGA	WES	WGA	WES
<i>Char_s</i>	$9 \times 7$	0.00561 <sub>5</sub>	0.00672 <sub>8</sub>	0.007344 <sub>8</sub>	0.00461 <sub>8</sub>	0.004604 <sub>8</sub>	0.00433 <sub>8b,2i</sub>	0.004662 <sub>6b,1i</sub>
<i>CircCirc</i>	$9 \times 9$	0.00651 <sub>6</sub>	0.00461 <sub>15</sub>	0.005621 <sub>15</sub>	0.00510 <sub>15</sub>	0.004702 <sub>13</sub>	0.00396 <sub>2b,13i</sub>	0.005729 <sub>3b,7i</sub>
<i>TexCirc</i>	$9 \times 9$	0.04196 <sub>6</sub>	0.04685 <sub>8</sub>	0.047941 <sub>8</sub>	0.03784 <sub>8</sub>	0.045346 <sub>7</sub>	0.04147 <sub>5b,3i</sub>	0.046481 <sub>6b,2i</sub>
<i>RuidoBool</i>	$9 \times 9$	0.00331 <sub>5</sub>	0.21672 <sub>8</sub>	0.003371 <sub>7</sub>	0.00224 <sub>8</sub>	0.002246 <sub>9</sub>	0.00215 <sub>6b,2i</sub>	0.002367 <sub>4b,2i</sub>
<i>TexRev</i>	$9 \times 7$	0.02600 <sub>5</sub>	0.03280 <sub>3</sub>	0.035802 <sub>3</sub>	0.01248 <sub>3</sub>	0.021978 <sub>10</sub>	0.01248 <sub>3b,0i</sub>	0.024903 <sub>4b,5i</sub>

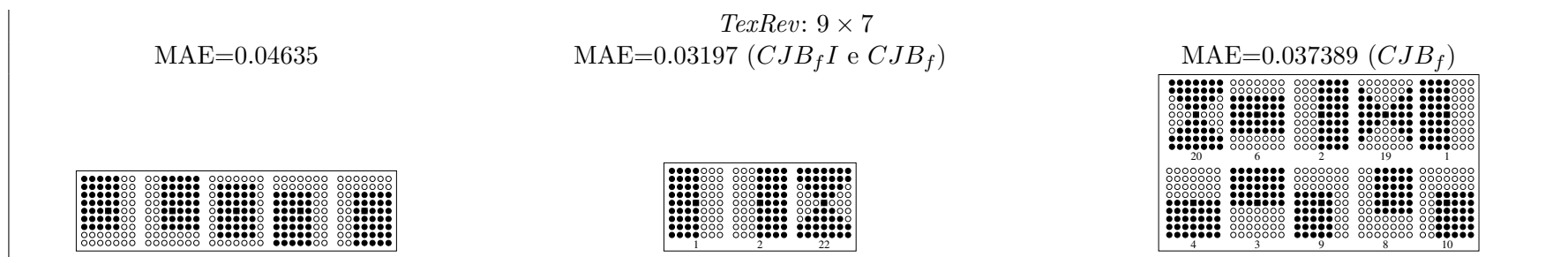
Os resultados mostram que o WGA fornece uma combinação melhor que a WES, mas os valores alcançados pela WES se aproximam daqueles obtidos pelo WGA. A desvantagem do WGA em relação a WES é o seu tempo de execução. Supondo uma coleção de janelas com 25 janelas, a população do WGA com 20 cromossomos fixos e 200 gerações, o treinamento do primeiro nível seria executado 25 vezes para as janelas da coleção, mais  $4000 = 20 * 200$  vezes pela função de avaliação, totalizando 4025 vezes.

A WES executaria o treinamento um nível 15 vezes para as janelas de primeiro nível e mais 14 vezes para o segundo nível, totalizando 29 execuções. Assim, dada uma coleção de janelas, esses valores demonstram que a WES, nas condições propostas neste trabalho, é mais rápida computacionalmente que o WGA e seu resultado, no geral, é tão bom quanto, ou melhor, que uma combinação manual.

As janelas selecionadas manualmente pelo WGA e pela WES podem ser vistas na Figura 6.10. Os MAEs expostos são os calculados sobre  $(\mathbb{S}, \mathbb{I})$ . O que se percebe é que o WGA consegue avaliar melhor as janelas em  $CJB_fI$ . Isto sugere que o critério de ordenação da WES pode ser aprimorado com o acréscimo de uma parcela que reflita a diversidade em um grupo de janelas. Dessa forma, acreditamos que se pode melhorar sua atuação sobre  $CJB_fI$ .

Na Figura 6.11 são exibidos fragmentos de imagens resultantes da aplicação dos melhores operadores do WGA, WES e do operador manual. No Apêndice A podem ser vistas as imagens completas.





**Figura 6.10:** Janelas usadas nos testes de *Hirata (2009)*, as melhores janelas obtidas pelo WGA e pela WES.

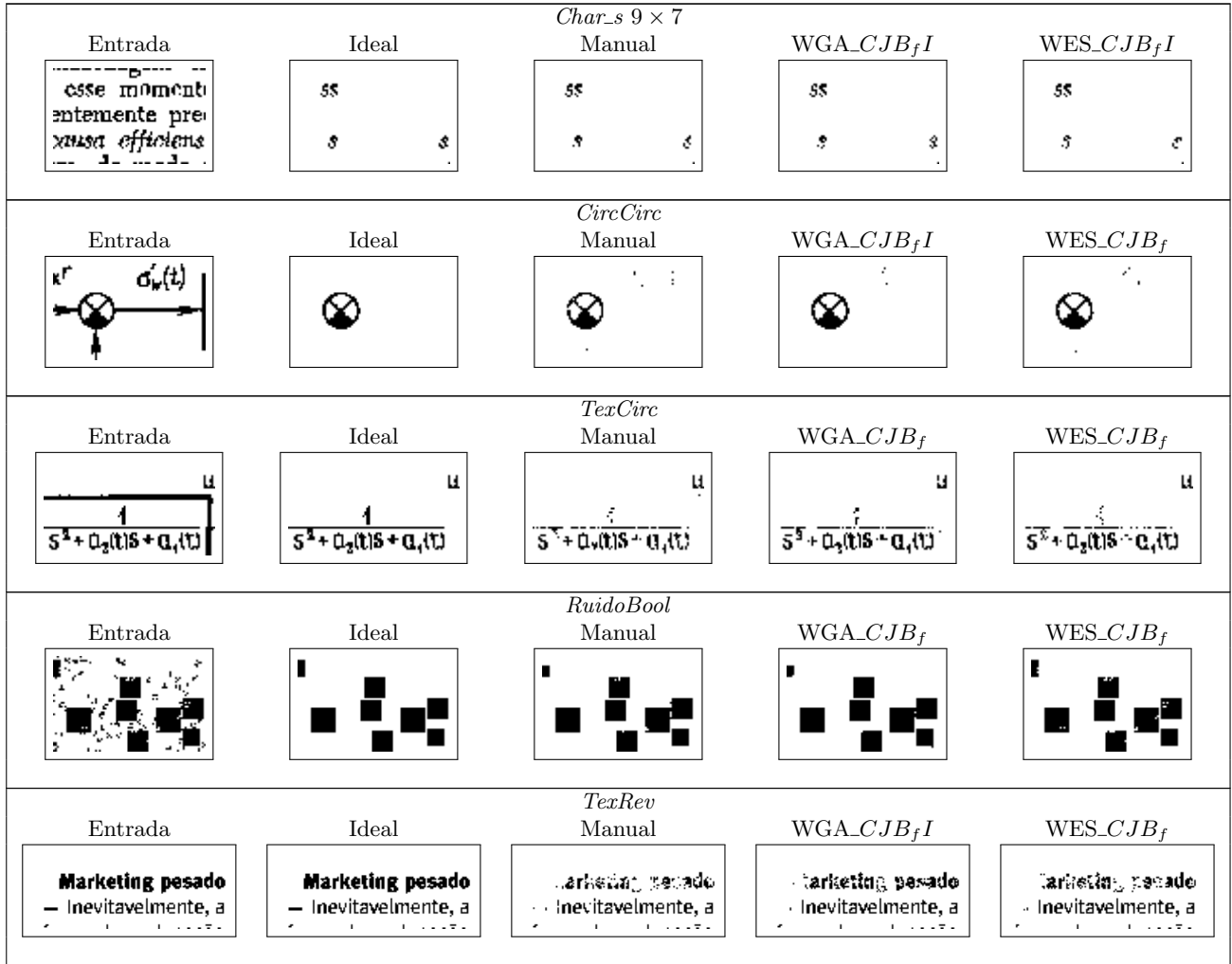


Figura 6.11: Fragmentos de imagens de entrada para cada conjunto de dados, suas respectivas imagens ideais, os resultados da aplicação dos operadores manuais e os resultados da aplicação dos operadores obtidos pela combinação WGA e pela WES.

## 6.5 Operadores um nível versus combinações do WGA e WES

Esta pesquisa apoiou-se na observação empírica do trabalho de Hirata (2009) de que uma combinação de operadores em dois níveis resulta em um operador final com melhor desempenho do que um operador um nível. Além disso, não foram encontrados relatos que comparam o MAE de um operador dois níveis com o MAE de seus operadores individuais.

Na intenção de complementar o nosso trabalho e atestar a eficácia de combinações dois níveis no projeto de W-operadores, foi calculado o MAE para os operadores individuais referentes a cada janela utilizada no conjunto de janelas  $CJB_fI$ . Para cálculo do MAE foi utilizado o conjunto de imagens de teste  $(S_t, I_t)$ . Os operadores foram projetados a partir das imagens em  $(S^1, I^1) \cup (S^2, I^2)$ . O menor  $CJB_fI$  tem 86 janelas e o maior 105 janelas.

Na Tabela 6.11 estão dispostos os MAEs das melhores combinações obtidas pelo WGA e pela WES e os melhores MAEs dos operadores individuais. Analisando estes resultados, verifica-se que uma combinação de operadores, mesmo que manual, oferece um operador superior ao projetado com uma janela individual.

**Tabela 6.11:** MAEs sobre  $(S_t, I_t)$  obtidos pelos operadores individuais, WGA e WES. O subscrito representa o número de janelas do primeiro nível, em que  $b$ =básica e  $i$ = janela obtida pela informação de interação.

Dados	Domínio de uma janela	Combinação manual	MAE sobre $(S_t, I_t)$		
			WGA	WES	Operador Individual
<i>Char_s</i>	$9 \times 7$	0.00561 <sub>5</sub>	0.00433 <sub>8b,2i</sub>	0.004604 <sub>8</sub>	0.00998
<i>CircCirc</i>	$9 \times 9$	0.00651 <sub>6</sub>	0.00396 <sub>2b,13i</sub>	0.004702 <sub>13</sub>	0.01567
<i>TexCirc</i>	$9 \times 9$	0.04196 <sub>6</sub>	0.03784 <sub>8</sub>	0.045346 <sub>7</sub>	0.06594
<i>RuidoBool</i>	$9 \times 9$	0.00331 <sub>5</sub>	0.00215 <sub>6b,2i</sub>	0.002246 <sub>9</sub>	0.00421
<i>TexRev</i>	$9 \times 7$	0.02600 <sub>5</sub>	0.01248 <sub>3b,0i</sub>	0.021978 <sub>10</sub>	0.03150

# Capítulo 7

## Discussões finais

Esta pesquisa teve por objetivo principal investigar e propor um arcabouço que selecione uma combinação de janelas de primeiro nível para projetar um operador dois níveis. O objetivo foi alcançado através de metodologias apresentadas neste texto.

O Capítulo 3 descreveu um método que estabelece um intervalo  $I_W$  de tamanhos de janela que pode ser usado em procedimentos para construção de janelas. Também foram apresentadas três propostas para a construção de coleções de janelas: a partir de janelas básicas (CJB) com tamanho selecionado em  $I_W$  ou prefixado pelo algoritmo de construção da janela; usando as janelas obtidas pelo algoritmo de minimização da magnitude da informação de interação (CJI); e a união dessas duas técnicas (CJBI). O domínio das janelas pode ser estabelecido manualmente ou automaticamente pelo algoritmo que constrói cada janela. As janelas básicas foram definidas de tal forma que atendam a problemas de diferentes domínios. As janelas da CJI são criadas para atender um problema específico.

Dois mecanismos de seleção de janelas, tendo como entrada uma coleção de janelas predefinida, foram apresentados nos Capítulos 4 e 5. O primeiro é um algoritmo genético, denominado neste texto de WGA e seus parâmetros foram definidos após a realização de vários testes. No segundo, um procedimento para seleção de janelas mediante a ordenação de uma coleção de janelas foi estabelecido. Um critério de ordenação baseado em uma versão corrigida da entropia condicional foi proposto. O procedimento de ordenação utilizando a entropia condicional corrigida é chamado de WES.

Experimentos foram realizados com as duas metodologias de seleção de janelas e com as diferentes coleções de janelas. Os resultados sugerem que:

- As coleções de janelas propostas foram adequadas para os conjuntos de dados, que eram de contexto diferentes, atestando sua abrangência;
- O WGA mostrou-se uma metodologia eficaz na seleção de combinações de janelas;
- Os testes com o WGA mostraram que a CJBI fornece combinações de janelas melhores que a utilização isolada da CJB ou CJI;
- A WES trabalhou melhor com as CJBs. Acreditamos que isso seja devido a diversidade garantida nestas coleções;
- A WES mostrou-se uma alternativa razoável para a seleção de janelas;
- A WES com CJB, domínio automático e seleção de tamanho de janelas em  $I_W$  é uma metodologia promissora para a implementação de uma ferramenta automática para o projeto de W-operadores;

- O WGA apresentou resultados melhores que a WES, mas em contrapartida a WES é mais eficiente.

Claramente, se for agregado à entropia condicional corrigida  $\hat{H}_c(Y|X)$  uma parcela que reflita a diversidade de um grupo de janelas, o novo critério fará com que a WES seja mais abrangente, melhorando sua aplicação com a CJBI e ampliando sua utilização para qualquer coleção de janelas. Esse critério também poderá ser adotado pelo WGA como função de avaliação, tornando-o mais eficiente.

As coleções de janelas propostas são flexíveis, no sentido que um especialista pode adicionar novos procedimentos para construção de janelas ou inserir diretamente uma nova janela à coleção construída. Assim, ele pode aprimorar uma determinada coleção. Além disso, as combinações selecionadas dessas coleções podem ser um ponto de partida para o especialista, que pode trabalhar sobre elas, adicionando ou alterando janelas em uma combinação.

A investigação de uma função que mensure a diversidade de um conjunto de janelas é objeto de trabalhos futuros. Além disso, pretende-se:

- Desenvolver uma ferramenta amigável que permita a utilização da WES por usuários com pouco conhecimento na área de W-operadores;
- Disponibilizar uma biblioteca com as metodologias propostas, permitindo a utilização individual de cada uma;
- Desenvolver uma interface que facilite especialistas manipularem as coleções de janelas ou adicionarem novos algoritmos para a construção delas.

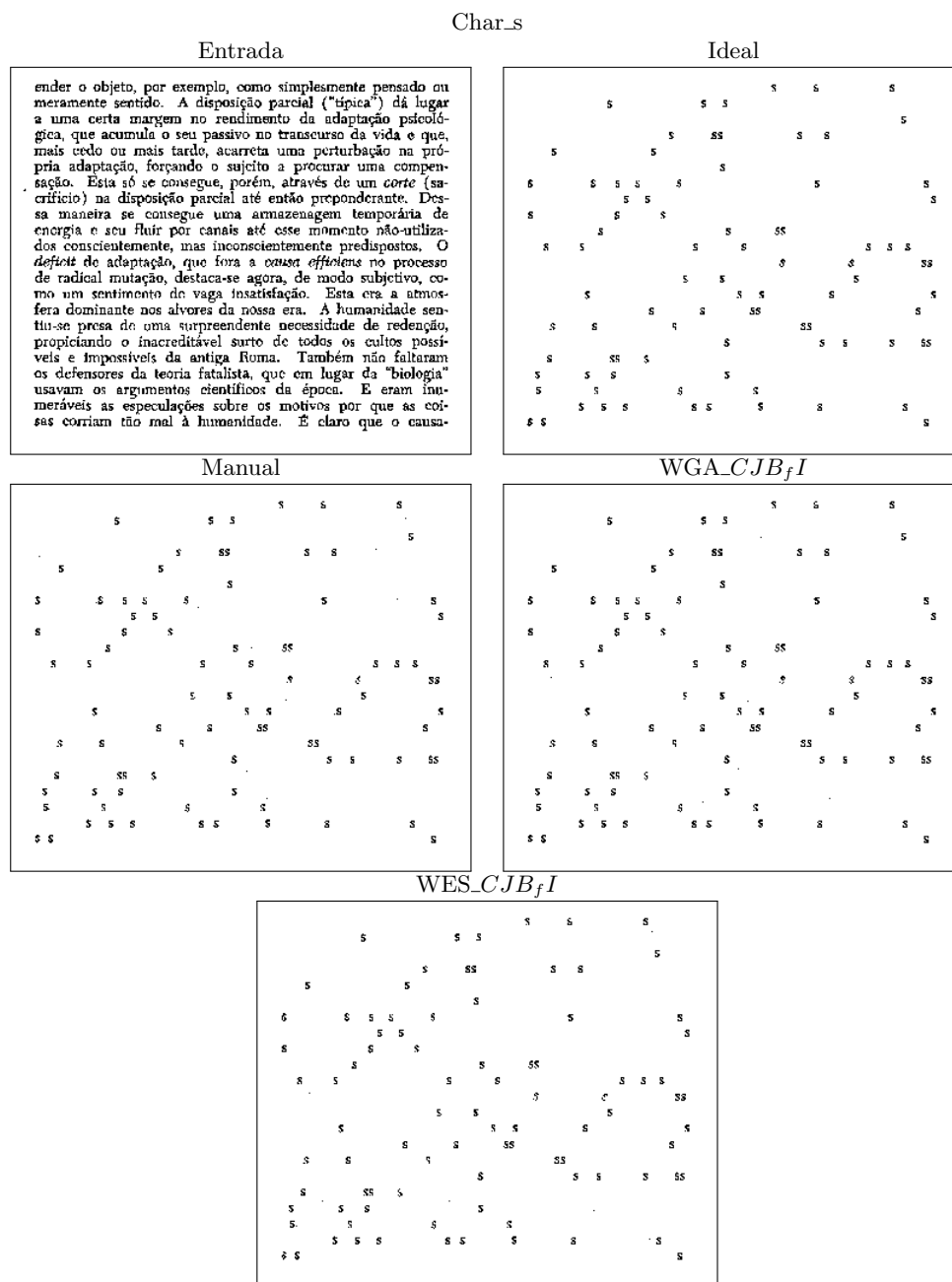
De forma geral concluímos que tanto o WGA e a WES, com coleções de janelas coerentes, são metodologias eficazes na seleção de janelas para um projeto dois níveis de W-operadores, colaborando no trabalho de um especialista na área. Acreditamos ainda que as metodologias: domínio automático, seleção de tamanho de janelas por  $I_W$ , CJB, WES com a entropia condicional corrigida, compõem uma ferramenta promissora para o projeto automático de W-operadores. Assim, esta pesquisa apresentou contribuições relevantes para o projeto de W-operadores e abriu novos caminhos para trabalhos futuros.



# Apêndice A

## Imagens

Este apêndice traz alguns exemplos de imagens de teste dos conjuntos de dados utilizados nos experimentos dos Capítulos 4, 5 e 6. Traz também exemplos de imagens resultantes da aplicação dos melhores operadores do WGA, da WES e os respectivos operadores definidos em [Hirata \(2009\)](#).



**Figura A.1:** Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual de janelas, o resultado obtido pela combinação do WGA\_CJB<sub>f</sub>I e pela combinação da WES\_CJB<sub>f</sub>I para Char\_s 9 × 7.

Char\_s

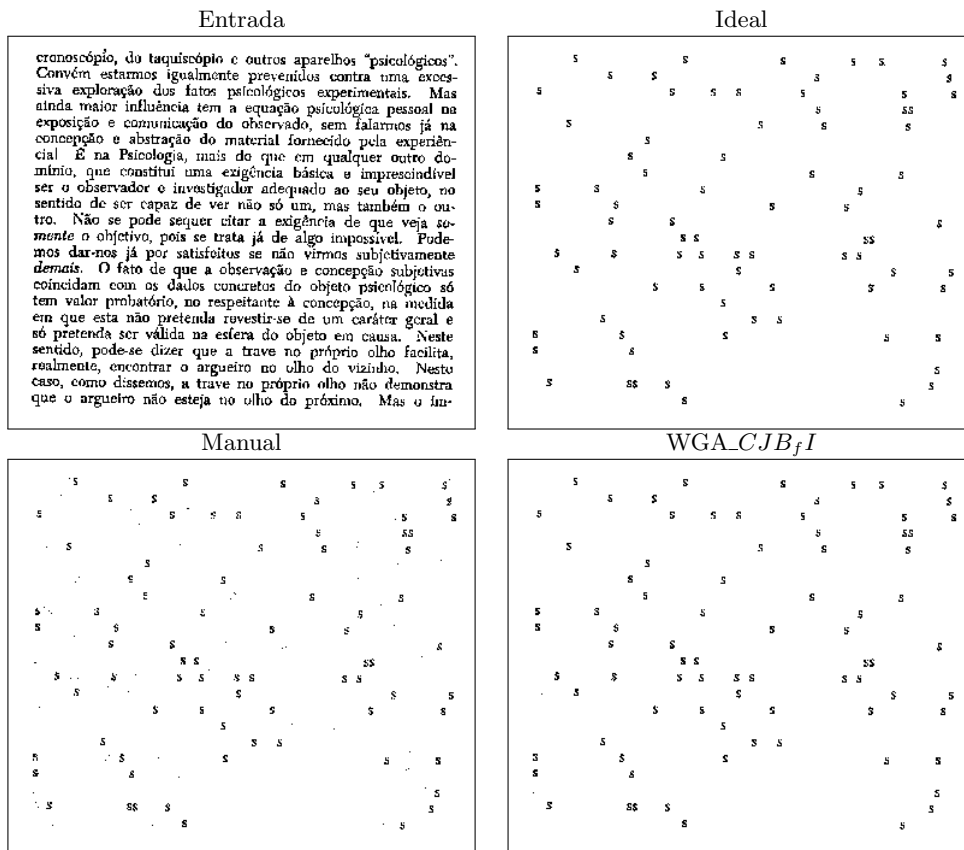


Figura A.2: Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual e o resultado obtido pela combinação do WGA\_CJB<sub>fI</sub> para Char\_s 11 × 9.

Mapa

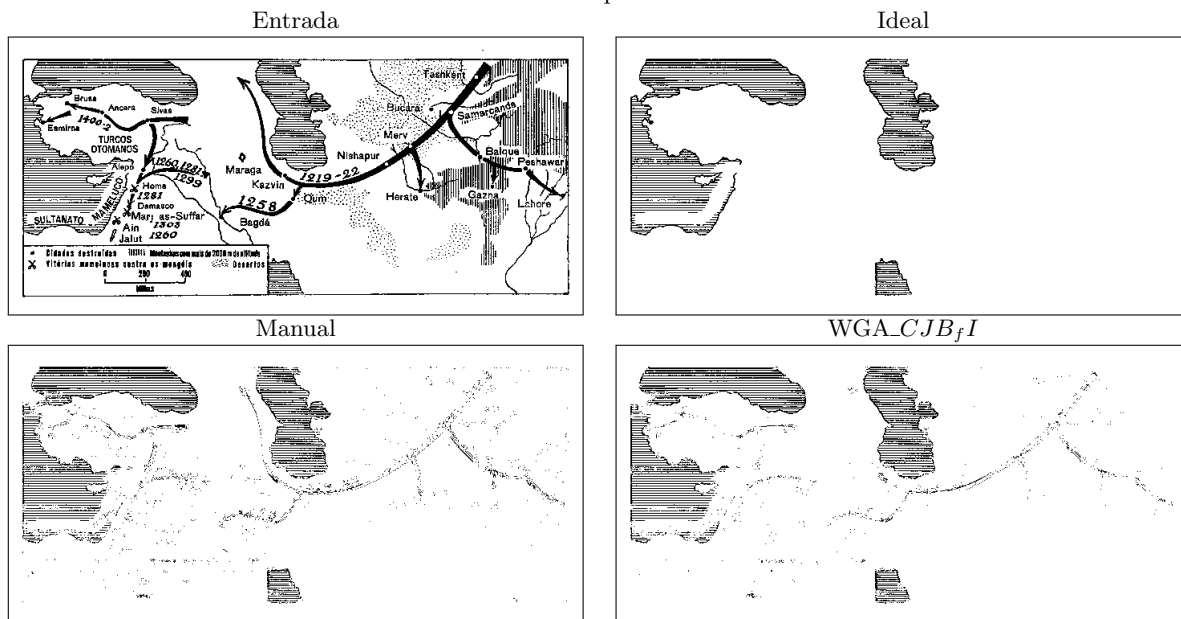
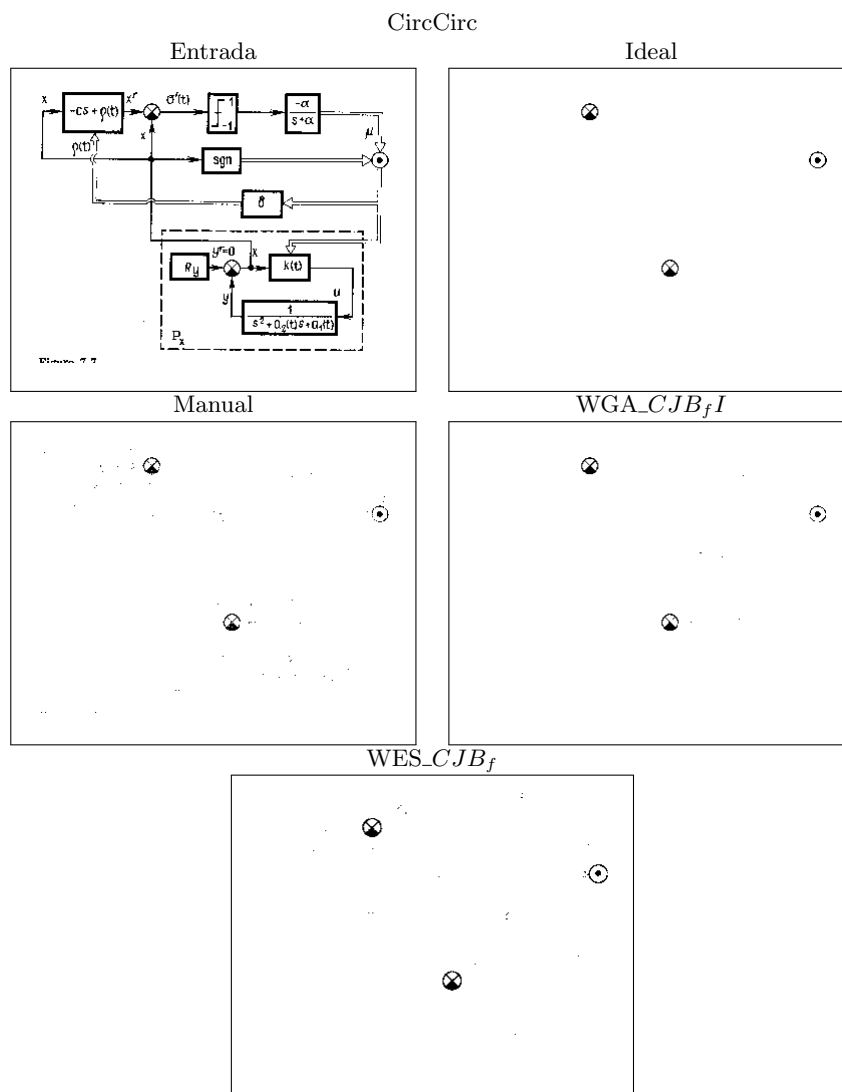


Figura A.3: Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual e o resultado obtido pela combinação do WGA\_CJB<sub>fI</sub> para Mapa.



**Figura A.4:** Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual e o resultado obtido pela combinação do WGA\_CJB\_fI e pela combinação da WES\_CJB\_f para CircCirc.

TexCirc

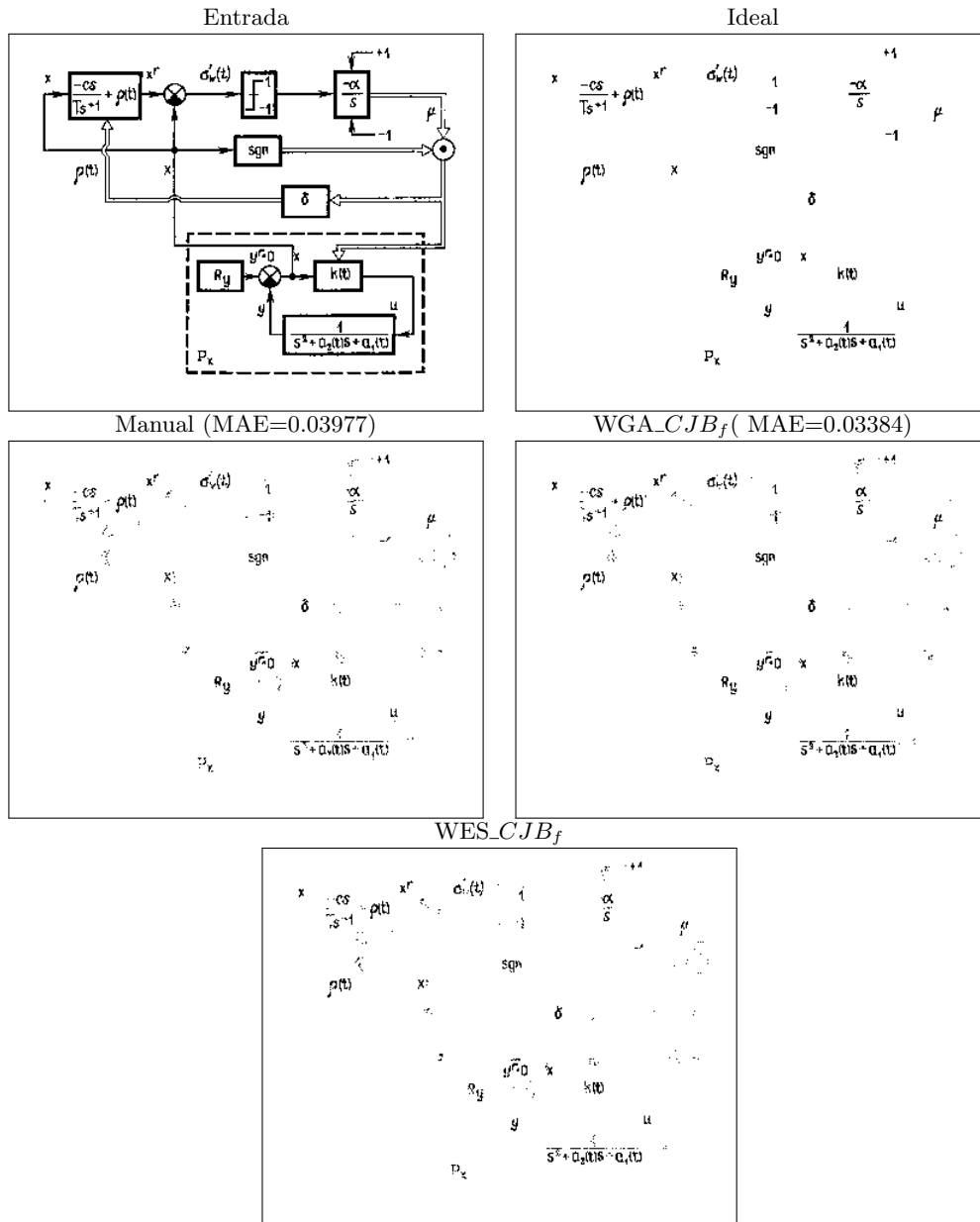
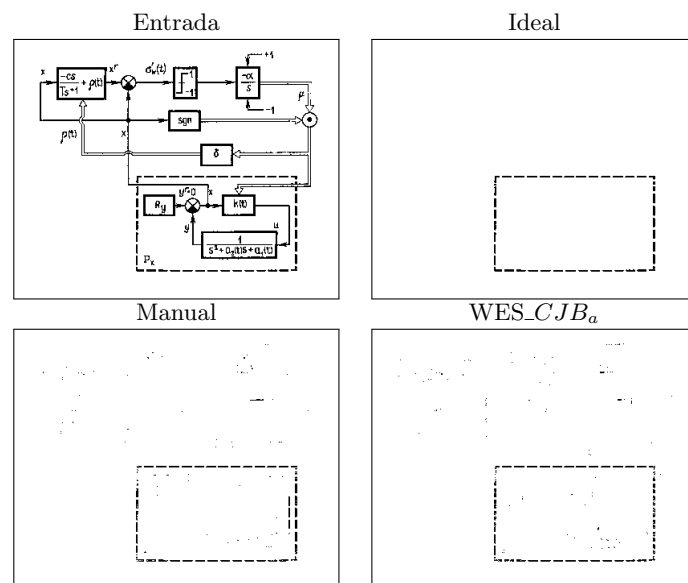


Figura A.5: Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual e o resultado obtido pela combinação do WGA\_CJB\_f e pela combinação da WES\_CJB\_f para TexCirc.



**Figura A.6:** Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual e o resultado obtido pela combinação da WES\_CJB<sub>a</sub> com domínio automático para BoxCirc.



**Distância Social**  
 Na portaria das grandes cidades as escovas de dactilografia ainda proliferam. Ali, a máquina de escrever serve de "trabalho" para o computador.

Scotlin, executivo de uma empresa de consultoria internacional. Quanto ao pessoal de suporte administrativo de um escritório, já foi totalmente reduzido ou eliminado. "O quadro de carreiras na administração da Editora Abril prevê 730 cargos", diz Luís Angelo Fellet, diretor de recursos humanos da empresa. "Uma recente redução reduziu-os para 95. E para ocupar qualquer um deles exigem-se conhecimentos de informática."

**Impacto** - Com a automação, o que muda mais não é a técnica, e sim a própria função. "Os trabalhadores de operação estão assumindo agora funções de controle", diz Norton Pereira, coordenador do núcleo de automação industrial do Senai de São Paulo. Um técnico mecânico ainda é a pessoa mais indicada para operar um torno, seja o equipamento mecânico ou de controle numérico. "O grande problema é que a substituição da máquina não é programada para coincidir com a substituição da mão-de-obra", diz Norton Pereira. Segundo um estudo de economia Márcia Helena de Lima intitulado "Perfil das atividades econômicas contribuintes do Senai", a automação teve impacto maior na qualificação da mão-de-obra do que no nível de emprego. O caso do eletroeletrônico Luciano Domingues é exemplar. Formou-se técnico de dois anos atrás e, desempregado, já teve de voltar aos bancos de aprendizagem para ter alguma chance. "O que aprendeu na escola e usava até o último emprego não serve para uma máquina de controle computadorizado. Tem como colegas de turma Hilson, de 23 anos e Ismael, de 30, ambos técnicos do Playcentro, o maior parque de diversões do país. Alguns brinquedos de parque, como o "Tornado" e o Superjet, são controlados eletronicamente. "Temos de estudar de novo senão ficamos para trás", diz Ismael.

**Nos Estados Unidos, sempre 14% dos empregos de secretária nos últimos dez anos. No Brasil, essa tendência ainda está restrita.**



Manual



**Distância Social**  
 Na portaria das grandes cidades as escovas de dactilografia ainda proliferam. Ali, a máquina de escrever serve de "trabalho" para o computador.

Scotlin, executivo de uma empresa de consultoria internacional. Quanto ao pessoal de suporte administrativo de um escritório, já foi totalmente reduzido ou eliminado. "O quadro de carreiras na administração da Editora Abril prevê 730 cargos", diz Luís Angelo Fellet, diretor de recursos humanos da empresa. "Uma recente redução reduziu-os para 95. E para ocupar qualquer um deles exigem-se conhecimentos de informática."

**Impacto** - Com a automação, o que muda mais não é a técnica, e sim a própria função. "Os trabalhadores de operação estão assumindo agora funções de controle", diz Norton Pereira, coordenador do núcleo de automação industrial do Senai de São Paulo. Um técnico mecânico ainda é a pessoa mais indicada para operar um torno, seja o equipamento mecânico ou de controle numérico. "O grande problema é que a substituição da máquina não é programada para coincidir com a substituição da mão-de-obra", diz Norton Pereira. Segundo um estudo de economia Márcia Helena de Lima intitulado "Perfil das atividades econômicas contribuintes do Senai", a automação teve impacto maior na qualificação da mão-de-obra do que no nível de emprego. O caso do eletroeletrônico Luciano Domingues é exemplar. Formou-se técnico de dois anos atrás e, desempregado, já teve de voltar aos bancos de aprendizagem para ter alguma chance. "O que aprendeu na escola e usava até o último emprego não serve para uma máquina de controle computadorizado. Tem como colegas de turma Hilson, de 23 anos e Ismael, de 30, ambos técnicos do Playcentro, o maior parque de diversões do país. Alguns brinquedos de parque, como o "Tornado" e o Superjet, são controlados eletronicamente. "Temos de estudar de novo senão ficamos para trás", diz Ismael.

**Nos Estados Unidos, sempre 14% dos empregos de secretária nos últimos dez anos. No Brasil, essa tendência ainda está restrita.**



WGA\_CJB\_fI

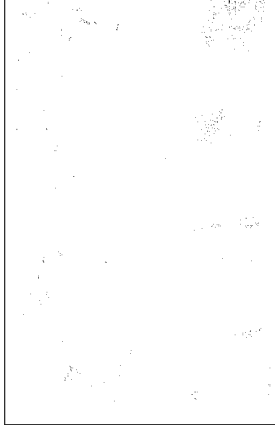


**Distância Social**  
 Na portaria das grandes cidades as escovas de dactilografia ainda proliferam. Ali, a máquina de escrever serve de "trabalho" para o computador.

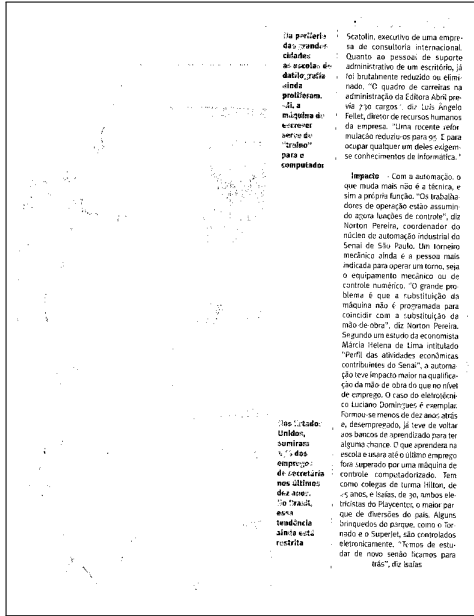
Scotlin, executivo de uma empresa de consultoria internacional. Quanto ao pessoal de suporte administrativo de um escritório, já foi totalmente reduzido ou eliminado. "O quadro de carreiras na administração da Editora Abril prevê 730 cargos", diz Luís Angelo Fellet, diretor de recursos humanos da empresa. "Uma recente redução reduziu-os para 95. E para ocupar qualquer um deles exigem-se conhecimentos de informática."

**Impacto** - Com a automação, o que muda mais não é a técnica, e sim a própria função. "Os trabalhadores de operação estão assumindo agora funções de controle", diz Norton Pereira, coordenador do núcleo de automação industrial do Senai de São Paulo. Um técnico mecânico ainda é a pessoa mais indicada para operar um torno, seja o equipamento mecânico ou de controle numérico. "O grande problema é que a substituição da máquina não é programada para coincidir com a substituição da mão-de-obra", diz Norton Pereira. Segundo um estudo de economia Márcia Helena de Lima intitulado "Perfil das atividades econômicas contribuintes do Senai", a automação teve impacto maior na qualificação da mão-de-obra do que no nível de emprego. O caso do eletroeletrônico Luciano Domingues é exemplar. Formou-se técnico de dois anos atrás e, desempregado, já teve de voltar aos bancos de aprendizagem para ter alguma chance. "O que aprendeu na escola e usava até o último emprego não serve para uma máquina de controle computadorizado. Tem como colegas de turma Hilson, de 23 anos e Ismael, de 30, ambos técnicos do Playcentro, o maior parque de diversões do país. Alguns brinquedos de parque, como o "Tornado" e o Superjet, são controlados eletronicamente. "Temos de estudar de novo senão ficamos para trás", diz Ismael.

**Nos Estados Unidos, sempre 14% dos empregos de secretária nos últimos dez anos. No Brasil, essa tendência ainda está restrita.**



WES\_CJB\_f



**Distância Social**  
 Na portaria das grandes cidades as escovas de dactilografia ainda proliferam. Ali, a máquina de escrever serve de "trabalho" para o computador.

Scotlin, executivo de uma empresa de consultoria internacional. Quanto ao pessoal de suporte administrativo de um escritório, já foi totalmente reduzido ou eliminado. "O quadro de carreiras na administração da Editora Abril prevê 730 cargos", diz Luís Angelo Fellet, diretor de recursos humanos da empresa. "Uma recente redução reduziu-os para 95. E para ocupar qualquer um deles exigem-se conhecimentos de informática."

**Impacto** - Com a automação, o que muda mais não é a técnica, e sim a própria função. "Os trabalhadores de operação estão assumindo agora funções de controle", diz Norton Pereira, coordenador do núcleo de automação industrial do Senai de São Paulo. Um técnico mecânico ainda é a pessoa mais indicada para operar um torno, seja o equipamento mecânico ou de controle numérico. "O grande problema é que a substituição da máquina não é programada para coincidir com a substituição da mão-de-obra", diz Norton Pereira. Segundo um estudo de economia Márcia Helena de Lima intitulado "Perfil das atividades econômicas contribuintes do Senai", a automação teve impacto maior na qualificação da mão-de-obra do que no nível de emprego. O caso do eletroeletrônico Luciano Domingues é exemplar. Formou-se técnico de dois anos atrás e, desempregado, já teve de voltar aos bancos de aprendizagem para ter alguma chance. "O que aprendeu na escola e usava até o último emprego não serve para uma máquina de controle computadorizado. Tem como colegas de turma Hilson, de 23 anos e Ismael, de 30, ambos técnicos do Playcentro, o maior parque de diversões do país. Alguns brinquedos de parque, como o "Tornado" e o Superjet, são controlados eletronicamente. "Temos de estudar de novo senão ficamos para trás", diz Ismael.

**Nos Estados Unidos, sempre 14% dos empregos de secretária nos últimos dez anos. No Brasil, essa tendência ainda está restrita.**

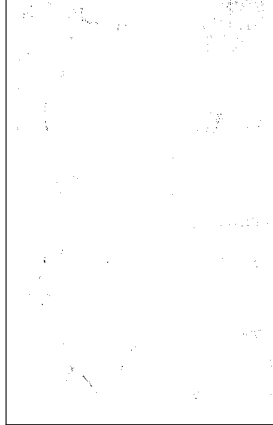
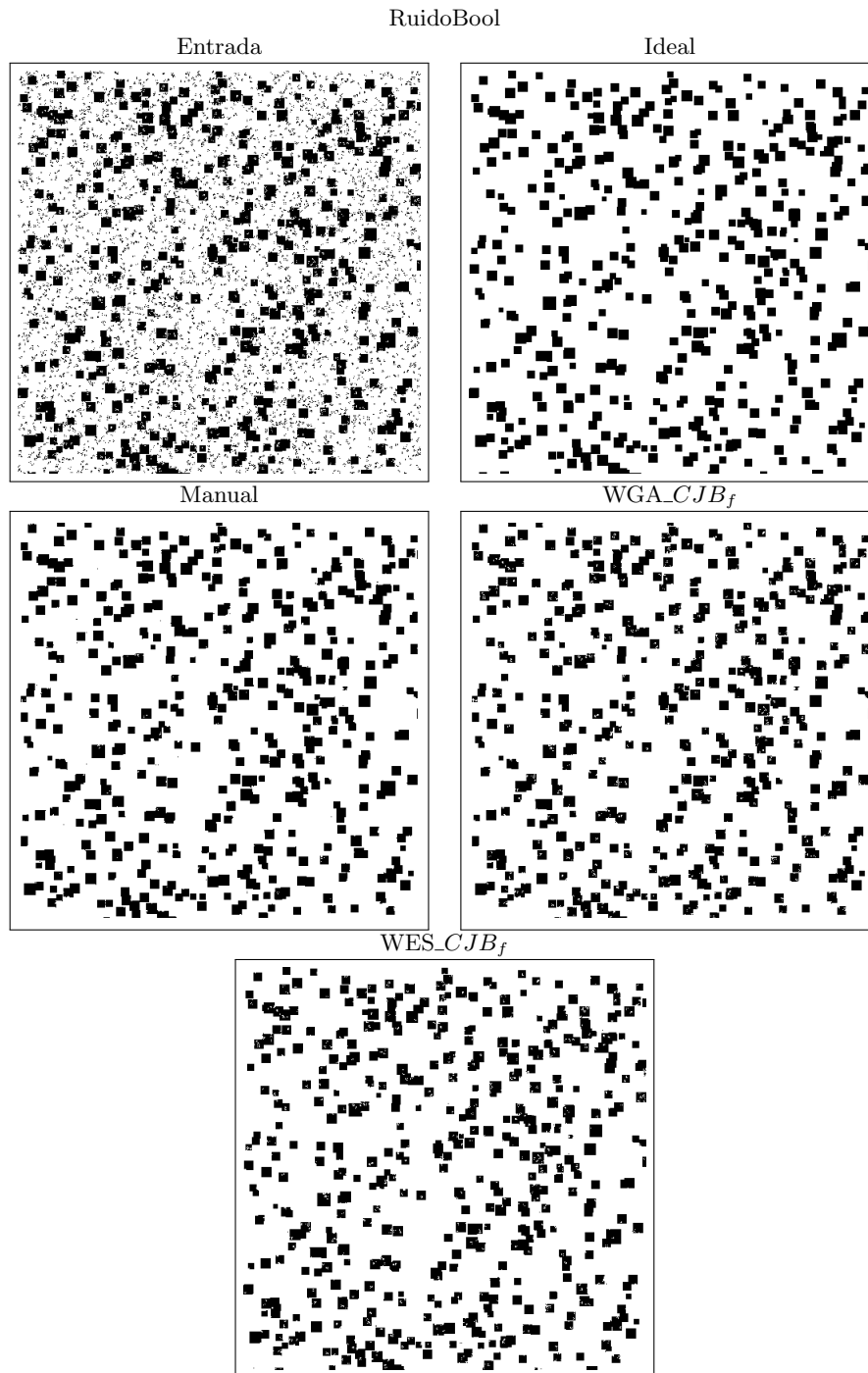


Figura A.7: Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela combinação manual e o resultado obtido pela combinação do WGA\_CJB\_fI e pela combinação da WES\_CJB\_f para TexRev.



**Figura A.8:** Uma imagem de entrada, sua respectiva imagem ideal, o resultado obtido pela respectiva combinação manual e o resultado obtido pela combinação do WGA\_CJB<sub>f</sub> e pela combinação da WES\_CJB<sub>f</sub> para RuidoBool.



## Apêndice B

# Procedimentos de construção de janelas para $CJB_\alpha$

Este apêndice descreve os procedimentos que constroem as janelas para a  $CJB_\alpha$ .

### B.1 Procedimentos padrões

Nesta seção estão descritos os procedimentos que constroem as janelas quando o domínio definido pelo usuário não é suficiente para a construção de uma janela com o tamanho sorteado em  $I_W$ . Nessa situação o domínio do usuário  $l \times c$  é mantido e o tamanho da respectiva janela é dado pelo procedimento que a constrói. Abaixo segue a descrição de cada procedimento.

$W_1$ : janela vertical ocupando o lado esquerdo do domínio com espessura de  $\lfloor \frac{c}{3} \rfloor$ .

$W_2$ : janela vertical ocupando o lado direito do domínio com espessura de  $\lfloor \frac{c}{3} \rfloor$ .

$W_3$ : janela horizontal ocupando o lado superior do domínio com espessura de  $\lfloor \frac{l}{3} \rfloor$ .

$W_4$ : janela horizontal ocupando o lado inferior do domínio com espessura de  $\lfloor \frac{l}{3} \rfloor$ .

$W_5$ : janela vertical central (coluna). Sua espessura é dada por  $2\lfloor \frac{c}{4} \rfloor + 1$ .

$W_6$ : janela horizontal central (linha). Sua espessura é dada por  $2\lfloor \frac{l}{4} \rfloor + 1$ .

$W_7$ : janela ocupando o quadrante esquerdo superior do domínio.

$W_8$ : janela ocupando o quadrante direito superior do domínio.

$W_9$ : janela ocupando o quadrante esquerdo inferior do domínio.

$W_{10}$ : janela ocupando o quadrante direito inferior do domínio.

$W_{11}$ : janela retangular ocupando o centro do domínio com base igual a  $2\lfloor \frac{c}{4} \rfloor + 1$  e altura igual a  $2\lfloor \frac{l}{4} \rfloor + 1$ .

$W_{12}$ : janela diagonal principal do domínio com espessura dada pelo maior inteiro ímpar menor ou igual a  $(\lceil \frac{c}{3} \rceil + 1)$ .

$W_{13}$ : janela diagonal secundária do domínio com espessura dada pelo maior inteiro ímpar menor ou igual a  $(\lceil \frac{c}{3} \rceil + 1)$ .

$W_{14}$ : janela triângulo com o ângulo reto no canto superior esquerdo do domínio até completar a diagonal secundária do domínio.

- $W_{15}$ : janela triângulo com o ângulo reto no canto superior direito do domínio até completar a diagonal primária do domínio.
- $W_{16}$ : janela triângulo com o ângulo reto no canto inferior esquerdo do domínio até completar a diagonal primária do domínio.
- $W_{17}$ : janela triângulo com o ângulo reto no canto inferior direito do domínio até completar a diagonal secundária do domínio.
- $W_{18}$ : janela losango centrada no domínio com as extremidades nos lados do domínio.
- $W_{19}$ : janela ampulheta horizontal centrada no domínio com os lados coincidindo com os lados do domínio.
- $W_{20}$ : janela ampulheta vertical centrada no domínio com os lados coincidindo com os lados do domínio.
- $W_{21}$ : janela cruz centrada no domínio com os lados externos coincidindo com os lados do domínio.
- $W_{22}$ : janela anel centrada no domínio. O lado externo do anel coincide com os lados do domínio. O lado interno horizontal do anel é dado por  $2\lfloor \frac{c}{4} \rfloor + 1$  e o lado interno vertical do anel é dado por  $2\lfloor \frac{l}{4} \rfloor + 1$ .
- $W_{23}$ : Janela definida com pontos aleatórios sobre o domínio.
- $W_{24}$ : Janela definida com pontos aleatórios sobre o domínio.

## B.2 Cálculo automático do domínio de definição das janelas

De acordo com a forma e o número de pontos  $np$  sorteado em  $I_W$  para cada janela da CJ, seu respectivo domínio é definido. O domínio sempre será definido com valores ímpares. Caso o valor não seja ímpar, pega-se o próximo inteiro ímpar. Abaixo segue a descrição para a definição de cada janela e seu domínio.

- $W_1$ : janela vertical ocupando o lado esquerdo do domínio. O lado maior (vertical) da janela é duas vezes o lado menor. O lado menor é dado por  $\lceil \sqrt{\frac{np}{2}} \rceil$ . O número de linhas do domínio é definido pelo lado maior (ímpar) e o número de colunas é três vezes o lado menor (ímpar).
- $W_2$ : janela vertical ocupando o lado direito do domínio. Idem como para  $W_1$  só que a janela agora é construída da direita para o centro.
- $W_3$ : janela horizontal ocupando o lado superior do domínio com espessura de  $\lfloor \frac{l}{3} \rfloor$ . O lado maior (horizontal) da janela é duas vezes o lado menor. O lado menor é dado por  $\lceil \sqrt{\frac{np}{2}} \rceil$ . O número de colunas do domínio é definido pelo lado maior (ímpar) e o número de linhas é três vezes o lado menor (ímpar).
- $W_4$ : janela horizontal ocupando o lado inferior do domínio. Idem como para  $W_3$  só que a janela agora é construída da borda inferior para o centro.
- $W_5$ : janela vertical central (coluna). O lado maior (vertical) da janela é duas vezes o lado menor. O lado menor é dado por  $\lceil \sqrt{\frac{np}{2}} \rceil$ . O número de colunas do domínio é definido pelo lado menor (ímpar) e o número de linhas pelo lado maior (ímpar).

- $W_6$ : janela horizontal central (linha). Sua espessura é dada por  $2\lfloor \frac{l}{4} \rfloor + 1$ . O lado maior (horizontal) da janela é duas vezes o lado menor. O lado menor é dado por  $\lceil \sqrt{\frac{np}{2}} \rceil$ . O número de linhas do domínio é definido pelo lado menor (ímpar) e o número de colunas pelo lado maior (ímpar).
- $W_7$ : janela ocupando o quadrante esquerdo superior do domínio. O número de linhas do domínio é igual ao número de colunas e é dado pelo maior inteiro ímpar menor ou igual a  $2 * \lceil \sqrt{np} \rceil$ .
- $W_8$ : janela ocupando o quadrante direito superior do domínio. Domínio como em  $W_7$ .
- $W_9$ : janela ocupando o quadrante esquerdo inferior do domínio. Domínio como em  $W_7$ .
- $W_{10}$ : janela ocupando o quadrante direito inferior do domínio. Domínio como em  $W_7$ .
- $W_{11}$ : janela retangular ocupando o centro do domínio. O número de linhas e colunas do domínio é dado pelo menor inteiro ímpar maior ou igual a  $\lceil \sqrt{np} \rceil$ .
- $W_{12}$ : janela diagonal principal do domínio. A janela a partir da diagonal principal levando em consideração o domínio definido como segue: Encontra-se  $le$  como o menor inteiro ímpar maior ou igual a  $\lceil \sqrt{\frac{np}{1.5}} \rceil$ . O número de linhas do domínio é dado pelo menor inteiro ímpar maior ou igual a  $\lceil \frac{np}{le} \rceil$ . O número de colunas é dado pelo menor inteiro ímpar maior ou igual a  $(l + le - 1)$ .
- $W_{13}$ : janela diagonal secundária do domínio. Domínio como em  $W_{12}$ .
- $W_{14}$ : janela triângulo com o angulo reto no canto superior esquerdo do domínio. O número de linhas do domínio é igual ao número de colunas e é o menor inteiro ímpar maior ou igual a  $\lfloor \sqrt{2 * np} \rfloor$ .
- $W_{15}$ : janela triângulo com o angulo reto no canto superior direito do domínio. Domínio como em  $W_{14}$ .
- $W_{16}$ : janela triângulo com o angulo reto no canto inferior esquerdo do domínio. Domínio como em  $W_{14}$ .
- $W_{17}$ : janela triângulo com o angulo reto no canto inferior direito do domínio. Domínio como em  $W_{14}$ .
- $W_{18}$ : janela losango centrada no domínio com as extremidades nos lados do domínio. O número de linhas do domínio é igual ao número de colunas e é o menor inteiro ímpar maior ou igual a  $\lfloor \sqrt{2 * np} \rfloor$ .
- $W_{19}$ : janela ampulheta horizontal centrada no domínio com os lados coincidindo com os lados do domínio. O número de linhas do domínio é igual ao número de colunas e é o menor inteiro ímpar maior ou igual a  $\lceil \sqrt{2 * np + 0,25} - 0,25 \rceil$ .
- $W_{20}$ : janela ampulheta vertical centrada no domínio com os lados coincidindo com os lados do domínio. Domínio como em  $W_{19}$ .
- $W_{21}$ : janela que é a aproximação de uma cruz formada por dois retângulos, um na horizontal e outro na vertical centralizados no centro domínio. É assumido que o lado maior do retângulo é igual a 2 vezes o lado menor e que a área desse retângulo é dada por  $np/2$ . Assim, o lado menor é  $\lceil \sqrt{\frac{np}{4}} \rceil$ . O lado maior é dado por  $\lceil \frac{np}{2 * \text{lado menor}} \rceil + \frac{\text{lado menor}}{2}$ . O número de linhas do domínio é igual ao número de colunas e é o menor inteiro ímpar maior ou igual ao lado maior.

$W_{22}$ : janela anel centrada no domínio. O lado externo do anel é aproximadamente duas vezes o lado interno do anel. O lado interno do anel é definido como sendo o menor inteiro ímpar maior ou igual a  $\lceil \sqrt{\frac{np}{3}} \rceil$ . O número de linhas do domínio é igual ao número de colunas e é dado por  $2 * \text{lado interno} + 1$ .

$W_{23}$ : Janela definida com pontos aleatórios sobre o domínio.

$W_{24}$ : Janela definida com pontos aleatórios sobre o domínio.

## Apêndice C

# Publicações

As publicações oriundas desta pesquisa foram:

- M. M. Dornelles e Nina S.T. Hirata. A genetic algorithm based approach for combining binary image operators. Em Proceedings of 21st International Conference on Pattern Recognition, volume 1 of ICPR 2012, páginas 3184-3187, Tsukuba, Japan.
- M. M. Dornelles e Nina S.T. Hirata. Selection of windows for W-operator combination from entropy based ranking. In: CONFERENCE ON GRAPHICS, PATTERNS AND IMAGES, 28. (SIBGRAPI), 2015, Salvador. Proceedings... Los Alamitos: IEEE Computer Societys Conference Publishing Services, 2015. On-line.



# Referências Bibliográficas

- MAT(2012)** *MATLAB*. The MathWorks Inc., Natick, Massachusetts. URL <http://www.mathworks.com/help/toolbox/images/f18-9530.html>. Citado na pág. 16
- Pyt(2015)** Python. Disponível em: <http://www.python.org>. Acesso em: 09 de fevereiro, 2015. Citado na pág. 34, 51
- Araújo et al.(2007)** R. A. Araújo, R. P. Sousa e T. A. Ferreira. An intelligent hybrid approach for designing increasing translation invariant morphological operators for time series forecasting. Em *Proceedings of the 4th international symposium on Neural Networks: Part II-Advances in Neural Networks*, ISNN '07, páginas 602–611, Berlin, Heidelberg. Springer-Verlag. ISBN 978-3-540-72392-9. doi: [http://dx.doi.org/10.1007/978-3-540-72393-6\\_72](http://dx.doi.org/10.1007/978-3-540-72393-6_72). URL [http://dx.doi.org/10.1007/978-3-540-72393-6\\_72](http://dx.doi.org/10.1007/978-3-540-72393-6_72). Citado na pág. 2
- Arifovic e Gencay(2001)** J. Arifovic e R. Gencay. Using genetic algorithms to select architecture of a feedforward artificial neural network. *Physica A: Statistical Mechanics and its Applications*, 289(3-4):574–594. URL <http://linkinghub.elsevier.com/retrieve/pii/S0378437100004799>. Citado na pág. 5
- Baker(1985)** J. E. Baker. Adaptive selection methods for genetic algorithms. Em *Proceedings of the 1st International Conference on Genetic Algorithms*, páginas 101–111, Hillsdale, NJ, USA. L. Erlbaum Associates Inc. ISBN 0-8058-0426-9. URL <http://portal.acm.org/citation.cfm?id=645511.657075>. Citado na pág. 39
- Banon e Barrera(1991)** G. J. F. Banon e J. Barrera. Minimal representations for translation-invariant set mappings by mathematical morphology. *SIAM J. Appl. Math.*, 51:1782–1798. ISSN 0036-1399. doi: <http://dx.doi.org/10.1137/0151091>. URL <http://dx.doi.org/10.1137/0151091>. Citado na pág. 1, 8
- Barrera et al.(1997)** J. Barrera, E. R. Dougherty e N. S. Tomita. Automatic programming of binary morphological machines by design of statistically optimal operators in the context of computational learning theory. *J. Electronic Imaging*, 6(1):54–67. Citado na pág. xiii, 2
- Barrera et al.(2000)** J. Barrera, R. Terada, R. Hirata-Jr e N. S. T. Hirata. Automatic programming of morphological machines by PAC learning. *Fundam. Inform.*, páginas 229–258. Citado na pág. xiii, 1, 2
- Barthes et al.(2001)** A. Barthes, J. Conrath, M. Rassigni, M. Adel e J.P. Petrakian. Mathematical morphology in computerized analysis of angiograms in age-related macular degeneration. *Med Phys*, 28(12):2410–9. Citado na pág. 1
- Basha e Prasad(2009)** S. S. Basha e K. S. Prasad. Automatic detection of breast cancer mass in mammograms using morphological operators and fuzzy c-means clustering. *Journal of Theoretical and Applied Information Technology.*, páginas 704–709. Citado na pág. 1
- Béliz-Osorio et al.(2011)** N. Béliz-Osorio, J. Crespo, M. García-Rojo, A. Muñoz e J. Azpiazu. Cytology imaging segmentation using the locally constrained watershed transform. Em *Proceedings of the 10th international conference on Mathematical morphology and its applications*

- to image and signal processing*, ISMM'11, páginas 429–438, Berlin, Heidelberg. Springer-Verlag. ISBN 978-3-642-21568-1. URL <http://dl.acm.org/citation.cfm?id=2023043.2023089>. Citado na pág. 16
- Bishop(1996)** C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, USA, 1 edição. ISBN 9780198538646. URL <http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0198538642>. Citado na pág. 10
- Box e Muller(1958)** G. E. P. Box e Mervin E. Muller. A note on the generation of random normal deviates. *Ann. Math. Statist.*, 29(2):610–611. doi: 10.1214/aoms/1177706645. URL <http://dx.doi.org/10.1214/aoms/1177706645>. Citado na pág. 29
- Bradski(2000)** G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*. Citado na pág. 16
- Breiman et al.( )** L. Breiman, J. Friedman, R. Olshen e C. Stone. *Classification and Regression Trees*. Wadsworth and Brooks. Citado na pág. 10
- Brindle(1981)** A. Brindle. *Genetic Algorithms for Function Optimization*. Phd thesis, University of Alberta. Citado na pág. 39
- Coley(1998)** D. A. Coley. *An Introduction to Genetic Algorithms for Scientists and Engineers*. World Scientific Publishing Co., Inc., River Edge, NJ, USA. ISBN 9810236026. Citado na pág. 5
- Cormen(2002)** C. E.; Rivest R. L.; Stein C Cormen, T. H.; Leiserson. *Algoritmos: Teoria e Prática*. Editora Campus, Rio de Janeiro. ISBN 8535209263. Citado na pág. 14
- Cover e Thomas(1991)** T. M. Cover e J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience, New York, NY, USA. ISBN 0-471-06259-6. URL <http://portal.acm.org/citation.cfm?id=129837>. Citado na pág. 17
- Dornelles e Hirata(2012)** M. M. Dornelles e Nina S.T. Hirata. A genetic algorithm based approach for combining binary image operators. Em *Proceedings of 21st International Conference on Pattern Recognition*, volume 1 of *ICPR 2012*, páginas 3184–3187, Tsukuba, Japan. Citado na pág. 60
- Dougherty(1992a)** E. R. Dougherty. Optimal mean-square N-observation digital morphological filters : I. optimal binary filters. *CVGIP: Image Understanding*, 55(1):36–54. ISSN 1049-9660. doi: DOI:10.1016/1049-9660(92)90005-N. URL <http://www.sciencedirect.com/science/article/pii/104996609290005N>. Citado na pág. 2, 10
- Dougherty(1992b)** E. R. Dougherty. Optimal mean-square N-observation digital morphological filters : Ii. optimal gray-scale filters. *CVGIP: Image Understanding*, 55(1):55–72. ISSN 1049-9660. doi: DOI:10.1016/1049-9660(92)90006-O. URL <http://www.sciencedirect.com/science/article/pii/104996609290006O>. Citado na pág. 2
- Dougherty(1998)** E. R. Dougherty. *Random Processes for Image and Signal Processing*. SPIE and IEEE Press. Citado na pág. 10
- Facon(1996)** J. Facon. *Morfologia Matemática: Teoria e Exemplos*. Editora Universitária Champagnat da Pontifícia Universidade Católica do Paraná. Citado na pág. 16
- Fiszelew et al.(2007)** A. Fiszelew, P. Britos, A. Ochoa, H. Merlino, E. Fernández e R. Garcia-Martinez. Finding optimal neural network architecture using genetic algorithms. *Research in Computing Science Journal*, 27:15–24. URL <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.72.8699&rep=rep1&type=pdf>. Citado na pág. 5



- Fleuret e Guyon(2004)** F. Fleuret e I. Guyon. Fast binary feature selection with conditional mutual information. *Journal of Machine Learning Research*, 5:1531–1555. Citado na pág. 4, 18
- Frénay et al.(2013)** Benoît Frénay, Gauthier Doquire e Michel Verleysen. Is mutual information adequate for feature selection in regression? *Neural Networks*, 48(0):1 – 7. ISSN 0893-6080. doi: <http://dx.doi.org/10.1016/j.neunet.2013.07.003>. URL <http://www.sciencedirect.com/science/article/pii/S0893608013001883>. Citado na pág. 18
- Glover e Laguna(1997)** F. Glover e M. Laguna. *Tabu Search*. Kluwer Academic Publishers, Norwell, MA, USA. ISBN 079239965X. Citado na pág. 5
- Goldberg e Deb(1991)** D. E. Goldberg e K. Deb. A comparative analysis of selection schemes used in genetic algorithms. Em *Foundations of Genetic Algorithms*, páginas 69–93. Morgan Kaufmann. Citado na pág. 39
- Grünwald(2007)** P. D. Grünwald. *The Minimum Description Length Principle*, volume 1 of MIT Press Books. The MIT Press. URL <http://ideas.repec.org/b/mtp/titles/0262072815.html>. Citado na pág. 4, 18
- Hall e Smith(1999)** M. A. Hall e L. A. Smith. Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper. Em *Proceedings of the Twelfth International Florida Artificial Intelligence Research Society Conference*, páginas 235–239. AAAI Press. ISBN 1-57735-080-4. URL <http://portal.acm.org/citation.cfm?id=646812.707499>. Citado na pág. 17
- Hirata(2015)** N. S. T. Hirata. Página oficial: Multilevel training of binary morphological operators. Disponível em: [http://www.vision.ime.usp.br/nina/research/multilevel\\_supplementary](http://www.vision.ime.usp.br/nina/research/multilevel_supplementary). Acesso em: 09 de fevereiro, 2015. Citado na pág. 55
- Hirata(2000)** N. S. T. Hirata. *Projeto Automático de Operadores: Explorando Conhecimento a Priori*. Tese de Doutorado, Instituto de Matemática e Estatística-Universidade de São Paulo, São Paulo, SP - Brasil. Citado na pág. 1, 10, 15
- Hirata(2011)** N. S. T. Hirata. Morphological operator design from training data. Em Halina Kwasnicka e Lakhmi Jain, editors, *Innovations in Intelligent Image Analysis*, volume 339 of *Studies in Computational Intelligence*, páginas 31–58. Springer Berlin / Heidelberg. ISBN 978-3-642-17933-4. URL [http://dx.doi.org/10.1007/978-3-642-17934-1\\_3](http://dx.doi.org/10.1007/978-3-642-17934-1_3). 10.1007/978-3-642-17934-1\_3. Citado na pág. 10, 11, 13
- Hirata et al.(2002)** N. S. T. Hirata, J. Barrera, R. Terada e E. R. Dougherty. The incremental splitting of intervals algorithm for the design of binary image operators. Em *Proceedings of the 6th International Symposium: ISMM 2002*, páginas 219–228. 6th International Symposium on Mathematical Morphology (H. Talbot and R. Beare, ed.). Citado na pág. 14, 15
- Hirata(2009)** Nina S. T. Hirata. Multilevel training of binary morphological operators. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31:707–720. ISSN 0162-8828. doi: <http://dx.doi.org/10.1109/TPAMI.2008.118>. URL <http://dx.doi.org/10.1109/TPAMI.2008.118>. Citado na pág. xiv, xv, xvii, 3, 5, 13, 15, 18, 30, 33, 44, 45, 47, 51, 52, 53, 55, 56, 60, 62, 64, 67, 72, 73, 77
- Hirata-Jr et al.(2002)** R. Hirata-Jr, J. Barrera, R. F. Hashimoto, D. O. Dantas e G. H. Esteves. Segmentation of microarray images by mathematical morphology. *Real-Time Imaging*, 8(6):491–505. ISSN 1077-2014. doi: 10.1006/rtim.2002.0291. URL <http://www.sciencedirect.com/science/article/pii/S1077201402902913>. Citado na pág. 1
- Holland(1975)** J. H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, MI, USA. Citado na pág. 5, 35

- Ileanã et al.(2004)** I. Ileanã, C. Rotar e A. Incze. The optimization of feed forward neural networks structure using genetic algorithms. Em *Proceedings of the International Conference on Theory and Applications of Mathematics and Informatics - ICTAMI*, Thessaloniki, Greece. Citado na pág. 5
- Ishikawa et al.(2010)** A. S. Ishikawa, E. A. da Silva e R. A. de A. Nóbrega. Extração de rodovias em imagens digitais de alta resolução com o uso da teoria de morfologia matemática. *Revista Brasileira de Cartografia*, (63/01):131–140. ISSN 0560-4613. Citado na pág. 1
- Jakulin e Bratko(2003)** A. Jakulin e I. Bratko. Analyzing attribute dependencies. Em *PKDD 2003, volume 2838 of LNAI*, páginas 229–240. Springer-Verlag. Citado na pág. 4, 18
- Jelodar et al.(2006)** M. S. Jelodar, S. M. Fakhraie e M. N. Ahmadabadi. Two-stage morphological filter design using genetic algorithm. Em *Engineering of Intelligent Systems, 2006 IEEE International Conference on*, páginas 1–5. doi: 10.1109/ICEIS.2006.1703152. Citado na pág. 5
- Ji et al.(1989)** L. Ji, J. Piper e J.Y. Tang. Erosion and dilation of binary images by arbitrary structuring elements using interval coding. 9(3):201–209. Citado na pág. 16
- Kirkpatrick et al.(1983)** S. Kirkpatrick, C. D. Gelatt e M. P. Vecchi. Optimization by simulated annealing. *Science*, 220:671–680. Citado na pág. 5
- Knuth(1981)** D.E. Knuth. *The Art of Computer Programming: Seminumerical algorithms*. Addison-Wesley series in computer science and information processing. Addison-Wesley. ISBN 9780201038224. URL <http://books.google.co.uk/books?id=L4dQAAAAMAAJ>. Citado na pág. 29
- Kraft et al.(1995)** P. Kraft, S. Marshall, J. J. Soraghan e N. R. Harvey. Parallel genetic algorithms for optimizing morphological filters. Em *Proceedings of the Fifth International Conference on Image Processing and Its Applications*, páginas 504–516. Citado na pág. 5
- Kuncheva(2004)** Ludmila I. Kuncheva. *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience. ISBN 0471210781. Citado na pág. 47, 49
- Linden(2006)** R. Linden. *Algoritmos Genéticos*. Brasport, Rio de Janeiro. Citado na pág. 5, 35, 39
- Lipschutz e Lipson(2004)** S. Lipschutz e M. Lipson. *Matemática Discreta: Coleção Schaum*. Bookman. ISBN 9788536303611. URL <http://books.google.com.br/books?id=2S9bwDmD1P0C>. Citado na pág. 4
- Liu et al.(2008)** Y. Liu, M. Ye, J. Peng e H. Wu. Finding the optimal number of clusters using genetic algorithms. *Science And Technology*, páginas 1325–1330. URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4670864>. Citado na pág. 5
- Martins-Jr(2004)** D. C. Martins-Jr. Redução de dimensionalidade utilizando entropia condicional média aplicada a problemas de bioinformática e de processamento de imagens. Dissertação de Mestrado, Instituto de Matemática e Estatística-Universidade de São Paulo, São Paulo, SP - Brasil. Citado na pág. 1, 30
- Martins-Jr et al.(2004)** D. C. Martins-Jr, R. M. Cesar-Jr e J. Barrera. W-operator window design by maximization of training data information. Em *Proceedings of 17th Proc. Brazilian Symposium on Computer Graphics, Image Processing - SIBGRAPI*, páginas 162–169. IEEE Computer Society Press. Citado na pág. xi, xiii, 17, 23, 30, 31, 48, 49
- Martins-Jr et al.(2005)** D. C. Martins-Jr, R. M. Cesar-Jr e J. Barrera. Automatic window design for gray-scale image processing based on entropy minimization. Em Manuel Lazo Cortes e Alberto Sanfeliu, editors, *X Iberoamerican Congress on Pattern Recognition*, Lecture Notes in Computer Science, Havana, Cuba. Springer-Verlag. Citado na pág. xi, 3, 5, 17, 23, 49

- Matheron(1975)** G. Matheron. *Random Sets and Integral Geometry*. John Wiley and Sons. Citado na pág. 7
- Montagner(2015)** I. S. Montagner. Trios: Training image operators from samples. Disponível em: <http://trioslib.sourceforge.net/>. Acesso em: 09 de fevereiro, 2015. Citado na pág. 51
- Oliveira(1998)** J. R. de F. Oliveira. *O Uso de Algoritmos Genéticos na Decomposição Morfológica de Operadores Invariantes em Translação Aplicados a Imagens Digitais*. Tese de Doutorado, São José dos Campos - Brasil. Citado na pág. 35
- Oracle(2015)** Oracle. Java. Disponível em: <https://www.oracle.com/java/index.html>. Acesso em: 09 de fevereiro, 2015. Citado na pág. 34, 43, 51
- Porta et al.(1998)** A. Porta, G. Baselli, D. Liberati, N. Montano, C. Cogliati, T. Gnecci-Ruscione, A. Malliani e Sergio Cerutti. Measuring regularity by means of a corrected conditional entropy in sympathetic outflow. *Biological Cybernetics*, 78(1):71–78. doi: <http://dx.doi.org/10.1007/s004220050414>. Citado na pág. xi, 49, 50
- Rasure et al.(1990)** J. Rasure, D. Argiro, T. Sauer e C. Williams. Visual Language and Software Development Environment for Image Processing. *International Journal of Imaging Systems and Technology*, 2:183–199. Citado na pág. 43
- Reis(2012)** M. da S. Reis. *Minimização de funções decomponíveis em curvas em U definidas sobre cadeias de posets - algoritmos e aplicações*. Tese de Doutorado, Instituto de Matemática e Estatística-Universidade de São Paulo, São Paulo, SP - Brasil. Citado na pág. xi, 50, 53
- Salcedo-Sanz et al.(2008)** S. Salcedo-Sanz, J. A. Portilla-Figueras, E. G. Ortiz-García, A. M. Pérez-Bellido, C. Thraves, A. Fernández-Anta e X. Yao. Optimal switch location in mobile communication networks using hybrid genetic algorithms. *Appl. Soft Comput.*, 8:1486–1497. ISSN 1568-4946. doi: 10.1016/j.asoc.2007.10.021. URL <http://dl.acm.org/citation.cfm?id=1410470.1410631>. Citado na pág. 5
- Salembier(1992)** P. Salembier. Structuring element adaptation for morphological filters. *J. Vis. Comun. Image Represent.*, 3:115–136. ISSN 1047-3203. doi: [http://dx.doi.org/10.1016/1047-3203\(92\)90010-Q](http://dx.doi.org/10.1016/1047-3203(92)90010-Q). URL [http://dx.doi.org/10.1016/1047-3203\(92\)90010-Q](http://dx.doi.org/10.1016/1047-3203(92)90010-Q). Citado na pág. 16
- Santos(2010)** C. S. Santos. Construção de atributos binários baseada em análise de interações. Dissertação de Mestrado, Instituto de Matemática e Estatística-Universidade de São Paulo, São Paulo, SP - Brasil. Citado na pág. xiii, xvii, 4, 6, 18, 30, 33, 34, 48, 56, 57
- Serra(1982)** J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press. Citado na pág. 7
- Serra(1988)** J. Serra. *Image Analysis and Mathematical Morphology: Theoretical Advances*. Image Analysis and Mathematical Morphology. Academic Press. ISBN 9780126372410. Citado na pág. 7
- Shannon(1948)** C. E. Shannon. A Mathematical Theory of Communication. *Bell system technical journal*, 27. Citado na pág. 4, 17
- Soille(2003)** P. Soille. *Morphological Image Analysis*. Springer-Verlag, Berlin, 2nd edição. Citado na pág. 1, 16
- Tomita(1996)** N. S. Tomita. Programação automática de máquinas morfológicas binárias baseada em aprendizado PAC. Dissertação de Mestrado, Instituto de Matemática e Estatística-Universidade de São Paulo, São Paulo, SP - Brasil. Citado na pág. 1, 8, 10, 14, 15

- Tuzikov et al.(1992)** A. Tuzikov, P. Soille, D. Jeulin, H. Bruneel e M. Vermeulen. Extraction of grid lines on stamped metal pieces using mathematical morphology. Em *Proceedings of Eleventh IAPR International Conference on Pattern Recognition, Conference A: Computer Vision and Applications*, volume 1, páginas 425–428. Citado na pág. [1](#)
- Vaquero(2006)** D. A. Vaquero. Projeto multiresolução de operadores morfológicos a partir de exemplos. Dissertação de Mestrado, Instituto de Matemática e Estatística-Universidade de São Paulo, São Paulo, SP - Brasil. Citado na pág. [3](#), [5](#)
- Wolpert(1992)** D. H. Wolpert. Stacked generalization. *Neural Networks*, 5:214–259. Citado na pág. [3](#), [13](#)
- Yoda et al.(1994)** I. Yoda, K. Yamamoto e H. Yamada. An automatic acquisition of hierarchical mathematical morphology procedures by GA. Em *Pattern Recognition, 1994. Vol. 2 - Conference B: Computer Vision Image Processing., Proceedings of the 12th IAPR International. Conference on*, volume 2, páginas 421 –425 vol.2. doi: 10.1109/ICPR.1994.576971. Citado na pág. [5](#)