

## Representação de contornos arbitrários com o método dos pontos de transição

Armando J. Pinho

**Resumo** – Neste artigo é apresentado um método completamente novo para a representação de contornos arbitrários. A sua eficiência de compressão é consideravelmente superior à do conhecido método da codificação de cadeias, em especial no caso de contornos de elevada complexidade.

**Abstract** – In this paper we describe a new method for the representation of arbitrary contours. Its compression efficiency is superior to that of the well known chain coding technique, specially in the case of high complexity contours.

### I. INTRODUÇÃO

A necessidade de codificar eficientemente contornos surge em várias aplicações. Uma das mais conhecidas e importantes é a codificação de imagem e vídeo baseada na decomposição das cenas visuais em regiões (objectos) de forma e tamanho variável [1]-[4]. Neste paradigma, a codificação eficiente da forma das regiões desempenha um papel extremamente importante. Contudo, a relevância do tópico que aqui abordamos não se resume somente a esta aplicação, sendo também fundamental noutras aplicações, tais como a representação de mapas de linhas de nível, de linhas de costa ou mapas meteorológicos.

Ao longo das últimas quatro décadas têm sido propostos vários métodos para a codificação de contornos, sendo provavelmente a codificação de cadeias (*chain coding*) a técnica mais popular [5]. Algumas variantes desta técnica têm sido sugeridas, algumas delas com vista à codificação sem perdas [6]-[8], outras permitindo perda de informação [9]. A simplificação de contornos [10] e aproximações baseadas em splines, polígonos e descritores de Fourier constituem também outras abordagens à codificação não reversível de contornos [11], [12].

O trabalho que aqui apresentamos enquadra-se neste tópico, isto é, na questão da representação de contornos de forma eficiente. Descrevemos um método completamente novo, capaz de codificar contornos arbitrários (abertos ou fechados) com ou sem perda de informação.

Este artigo está organizado da seguinte forma. Na Secção II apresentamos uma descrição detalhada do método da codificação de cadeias, o qual usámos como termo de comparação em relação ao método que propomos. A Secção III é dedicada à apresentação e discussão do método proposto, o qual designamos por “método dos pontos de transição”, sendo abordadas várias estratégias de codificação e apresentados resultados experimentais. Em particular, é mostrado que o método dos pontos de transição apresenta uma eficiência bastante superior ao método da codificação de cadeias, em especial no caso de contornos de elevada complexidade. Finalmente, a Secção IV apresenta conclusões e algumas perspectivas de trabalho futuro.

### II. O MÉTODO DA CODIFICAÇÃO DE CADEIAS

A técnica de representação de contornos com base na codificação de cadeias (*chain coding*) é um método bastante conhecido que permite a representação eficiente de linhas sobrepostas numa grelha regular. Desde a sua apresentação em 1961 por Freeman [5], várias aplicações têm motivado um desenvolvimento continuado desta técnica. Entre elas, destacamos a representação eficiente de: mapas geográficos contendo curvas de nível e linhas de costa [6]; desenhos baseados em linhas [9]; imagens binárias (isto é, de duas cores) [13]; imagens de níveis de cinzento [14]; formas de objectos [7], [4], [15], [16]; partições de imagens [8].

Uma das grandes vantagens do método da codificação de cadeias é a sua capacidade de permitir a representação tanto de contornos exactos como de contornos aproximados ou simplificados. Esta característica destaca esta técnica de outras que estão limitadas ou fortemente vocacionadas para a representação aproximada de contornos. Neste grupo encontramos métodos baseados, por exemplo, em splines, polígonos ou descritores de Fourier [11].

Tendo em conta o atrás apontado e também tendo em consideração que o método que propomos neste trabalho tem a propriedade de poder representar contornos arbitrários tanto de forma exacta como de forma simplificada, decidimos escolher a técnica da codificação de cadeias como termo de comparação. Apresentamos de seguida uma descrição resumida do seu funcionamento, para que o seu desempenho possa ser entendido e avaliado de uma forma mais clara.

Tipicamente, uma sequência de símbolos resultante de um codificador de cadeias tem a seguinte estrutura:

$$r_1, c_1, s_1^1, s_1^2, \dots, s_1^{N_1}, \dots, r_K, c_K, s_K^1, s_K^2, \dots, s_K^{N_K} \quad (1)$$

em que  $(r_i, c_i)$  denota o par de coordenadas que marca o início de uma cadeia (as linhas,  $r_i$ , crescem de cima para baixo na imagem; as colunas,  $c_i$ , aumentam da esquerda para a direita). Os  $s_i^j$  representam instruções relativas à construção da cadeia de elementos de contorno. No caso de contornos fechados, são necessários quatro tipos diferentes de instruções: *segue em frente* (“A”), *vira à esquerda* (“L”), *vira à direita* (“R”) e *ramificação* (“+”). O número de cadeias não ligadas é dado por  $K$ , enquanto que  $N_i$  denota o número de instruções associadas à cadeia  $i$ .

Para melhor ilustração do processo de formação da sequência descrita em (1) vamos apresentar um exemplo baseado no pequeno mapa de contornos da Fig. 1. Para estes contornos, a sequência de símbolos correspondente ao código de cadeia é:

0, 1, L, R, +, A, A, R, +, A, L, L, +, L, A, A, L, R,  
3, 3, +, A, A, L, R, A

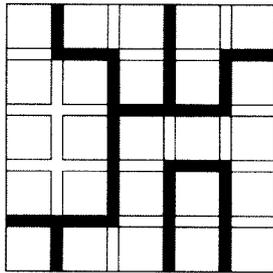


Figura 1 - Exemplo de um pequeno mapa de contornos contendo somente contornos fechados.

Repare-se que nos casos em que o ponto de início da cadeia coincide com uma das bordas da imagem então a primeira instrução é sempre "A", pelo que pode ser omitida. No exemplo apresentado, isto acontece no ponto de início de cadeia situada em (0, 1). Por outro lado, no caso dos contornos serem fechados, todos os pontos de "fim de cadeia" podem ser deduzidos durante a descodificação, uma vez que eles só podem ocorrer nas bordas da imagem ou quando um elemento de contorno previamente descodificado é encontrado.

Quando ocorrem ramificações, a direcção de continuação é escolhida de acordo com uma ordem pré-definida, a qual é também seguida pelo descodificador. No nosso caso, utilizámos uma ordem de acordo com o movimento dos ponteiros do relógio, sendo o elemento de cima (12 horas) o primeiro a ser visitado.

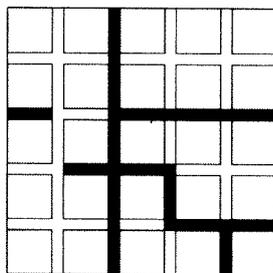


Figura 2 - Exemplo de um pequeno mapa de contornos contendo contornos abertos e fechados.

No caso dos contornos não serem fechados, é necessário adicionar mais um tipo de instrução ao conjunto das quatro já referidas, sendo sua função a indicação de *fim de cadeia* ("–"). Tomando como exemplo o mapa de contornos apresentado na Fig. 2, temos a seguinte sequência de símbolos:

0, 2, A, +, A, +, R, –, +, A, A, L, R, L, +, R, A,  
L, A, A, 2, 0, –

É importante notar que os cruzamentos de contornos (aos quais correspondem ramificações duplas) contidos no mapa apresentado na Fig. 2 geram dois símbolos do tipo "+" (segundo e terceiro "+" da sequência). Contudo, isto nem sempre acontece, já que, por vezes, durante o seguimento de um dos ramos, podemos ir ao encontro de outro ramo da

ramificação. Neste caso, a ramificação dupla é representada por um único símbolo "+".

Como vimos, a sequência de símbolos gerada pelo codificador de cadeias é composta por duas componentes distintas. Por um lado, as coordenadas dos pontos de início das cadeias, isto é, os  $(r_j, c_j)$ , e, por outro lado, as instruções de movimento ao longo da cadeia, os  $s_j^l$ . Com vista à obtenção de uma elevada eficiência de compressão, estas duas componentes são codificadas de forma separada e distinta.

O número total de bits necessário para representar um mapa de contornos usando a técnica da codificação de cadeias é calculado através da expressão:

$$n_{ch} = S + K(\bar{H}_r + \bar{H}_c) \quad (\text{bits}) \quad (2)$$

onde  $K$  denota o número de cadeias não ligadas,  $H_r$  e  $H_c$  denotam a entropia de uma fonte de informação  $\Delta x_l$ , onde

$$\Delta x_l = \begin{cases} x_l & \text{for } l = 1 \\ x_l - x_{l-1} & \text{for } l = 2, \dots, K-1 \end{cases} \quad (3)$$

com  $x = r, c$ , em que  $r$  denota linhas e  $c$  colunas correspondentes aos pontos de início das cadeias. A equação (3) representa uma codificação DPCM (*Differential Pulse Code Modulation*), destinando-se a explorar, especialmente, a variação monotónica e lenta dos  $r_j$ .  $S$  representa o número de bits gerado por um codificador aritmético adaptativo aplicado à fonte de informação  $s_j^l$ . As probabilidades do codificador aritmético são calculadas com base em contextos que podem variar entre um e seis símbolos de profundidade, sendo sempre escolhida a profundidade de contexto mais favorável.

### III. O MÉTODO DOS PONTOS DE TRANSIÇÃO

Nesta secção descrevemos um método novo capaz de representar contornos arbitrários, de forma exacta ou não, apresentando uma maior eficiência e simplicidade que o método da codificação de cadeias. Esta nova técnica baseia-se numa abordagem completamente distinta daquela que caracteriza a codificação de cadeias, ou seja, o seguimento de contornos. Em vez disso, o método que propomos tem como base a identificação e codificação de um conjunto de pontos críticos do mapa de contornos, os quais são suficientes para o representar de forma exacta. Esses pontos ocorrem sempre que os contornos mudam de estado, isto é, quando transitam do estado activo para o estado inactivo ou *vice-versa*, razão pela qual os designamos "pontos de transição" [17], [18].

Consideremos os pequenos quadrados pretos que aparecem na Fig. 3, aos quais chamamos "nós". Uma imagem com  $N_R$  linhas e  $N_C$  colunas contém  $N_R(N_C - 1) + N_C(N_R - 1)$  elementos de contorno (os rectângulos brancos na Fig. 3) e  $N_R N_C - 1$  nós. Ao conjunto de todos os elementos de contorno chamamos "mapa de contornos". Cada elemento de contorno pode estar activo, caso defina uma separação entre os dois elementos de imagem vizinhos, ou inactivo, caso contrário. Aos nós apontados por setas chamamos "nós de partida" (os quais são  $N_R + N_C - 2$ ).

Consideremos o varrimento de uma linha do mapa de contornos, da esquerda para a direita (ou de uma coluna, de

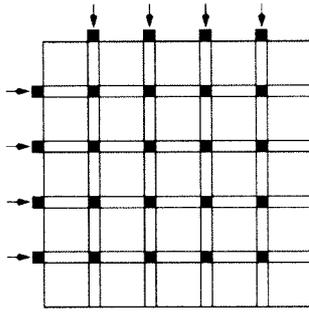


Figura 3 - Localização dos elementos de contorno (rectângulos brancos) e dos nós (quadrados pretos pequenos). Os nós marcados com uma seta são os "nós de partida". Os quadrado cinzentos indicam a posição dos elementos de imagem.

cima para baixo), começando no respectivo nó de partida. Para que o conteúdo da referida linha (ou coluna) possa ser reconstruído, basta registar a posição dos nós onde ocorrem transições no estado dos elementos de contorno, isto é, passam do estado activo para o estado inactivo ou *vice-versa*. Aos nós onde ocorrem estas transições chamamos "pontos de transição" do mapa de contornos. É imediato verificar que, no máximo, podem ocorrer  $N_R N_C - 1$  pontos de transição, ou seja, tantos quantos o número de nós do mapa de contornos.

Se usarmos este algoritmo nas  $N_R - 1$  linhas de elementos de contorno horizontais e nas  $N_C - 1$  colunas de elementos de contorno verticais, então, com base nessa informação, podemos reconstruir completamente o mapa de contornos. A Fig. 4 mostra um exemplo de um pequeno mapa de contornos, contendo contornos fechados e abertos, no qual os quadrados brancos assinalam todos os pontos de transição.

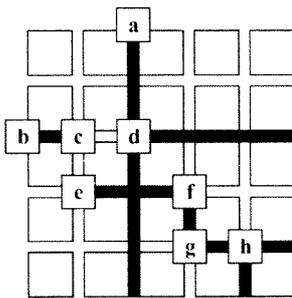


Figura 4 - Exemplo de um mapa de contornos e dos correspondentes pontos de transição (quadrados brancos com letras).

Tendo em conta o referido exemplo, não é difícil de verificar que existem pontos de transição devidos a mudanças de estado somente na direcção vertical (**a** e **h**), somente na direcção horizontal (**b**, **c**, **d** e **e**) assim como em ambas as direcções (**f** e **g**). Chamamos-lhes, respectivamente, T's verticais (**V**), T's horizontais (**H**) e cantos (**C**). A Fig. 5 ilustra a classificação de cada ponto de transição do mapa de contornos da Fig. 4 num destes três tipos.

Tendo como base a localização e tipo de todos os pontos de transição, a reconstrução do mapa de contornos é uma operação trivial. De facto, o processo de reconstrução pode ser visto como sendo gerado por uma caneta imaginária que

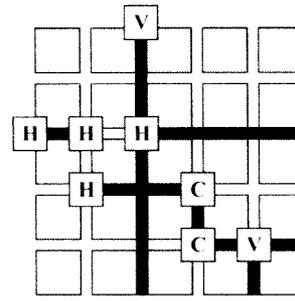


Figura 5 - Atribuição do tipo de ponto de transição: **C** indica um canto, **H** indica um T horizontal e **V** indica um T vertical.

se movimenta ao longo de cada linha de nós (da esquerda para a direita), e ao longo de cada coluna de nós (de cima para baixo) de acordo com as seguintes regras:

- No início de cada linha ou coluna a caneta está levantada.
- Ao movimentar-se ao longo de uma linha, a caneta muda de estado (isto é, levanta se estiver em baixo ou baixa se estiver levantada) quando encontra pontos de transição do tipo **H** ou **C**.
- Ao movimentar-se ao longo de uma coluna, a caneta muda de estado quando encontra pontos de transição do tipo **V** ou **C**.

A. Codificação dos pontos de transição

Uma das possíveis estratégias para a codificação dos pontos de transição baseia-se na decomposição da informação em duas sequências de símbolos: (1) distâncias,  $d_i$ , entre pontos de transição; (2) tipo de ponto de transição,  $t_i$ .

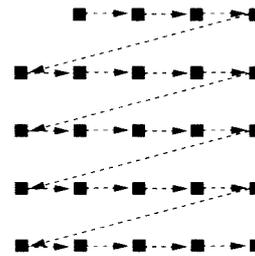


Figura 6 - Percurso utilizado para a construção da sequência de símbolos.

A sequência é formada efectuando um varrimento dos nós segundo o movimento mostrado na Fig. 6, o qual gera uma estrutura do tipo:

$$d_1, t_1, d_2, t_2, \dots, d_N, t_N, d_{N+1}$$

em que  $d_i \in \mathbb{N}_0$  e  $t_i \in \{C, H, V\}$ , sendo  $N$  o número de pontos de transição.

Alguns da informação pode ser omitida da sequência, caso possa ser deduzida pelo contexto. Este é o caso, por exemplo, dos pontos de transição que ocorrem nos nós de partida da primeira linha, os quais são sempre do tipo "V". O mesmo sucede com os nós de partida que ocorrem na primeira coluna, os quais são sempre do tipo "H". Em ambos os casos, os respectivos símbolos podem ser omitidos da sequência.

Deste modo, o mapa de contornos apresentado na Fig. 5 pode ser representado pela seguinte sequência:

1, (V), 7, (H), 0, H, 0, H, 3, H, 1, C, 4, C, 0, V, 0

onde os parêntesis indicam que o símbolo pode ser eliminado sem perda de informação.

A quantidade de informação que pode ser eliminada é ainda maior quando se consideram mapas de contornos formados somente por contornos fechados. Neste caso, o número de configurações de elementos de contorno que podem originar pontos de transição é menor, como pode ser visto na Fig. 7.

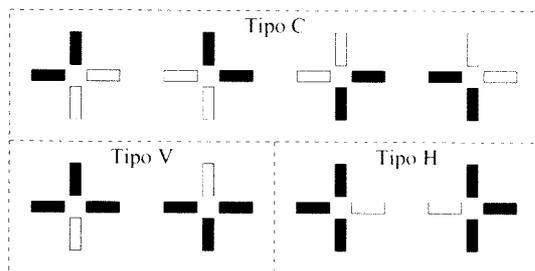


Figura 7 - Configurações dos elementos de contorno que originam pontos de transição no caso dos contornos serem fechados.

Por outro lado, esta restrição também reduz o número de possibilidades do tipo de ponto de transição, para um dado nó. De facto, devido à forma como os nós são visitados (de cima para baixo e da esquerda para a direita) é possível conhecer a natureza (activo ou inactivo) dos elementos de contorno situados acima e à esquerda do nó em análise. Dependendo desses dois elementos de contorno, o tipo de ponto de transição é limitado às possibilidades mostradas na Fig. 8.

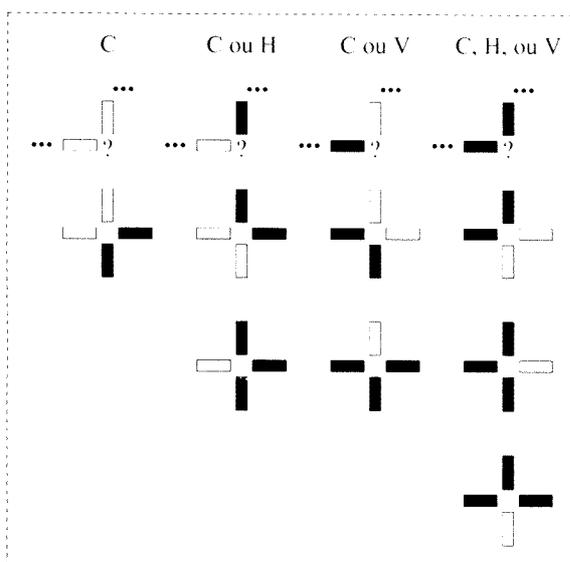


Figura 8 - No caso dos contornos serem fechados, o conjunto de tipos de pontos de transição depende do estado dos elementos de contorno situados acima e à esquerda do nó sujeito a avaliação. O ponto de interrogação indica a presença de um ponto de transição (desconhecido). Cada coluna mostra as alternativas possíveis, em cada caso.

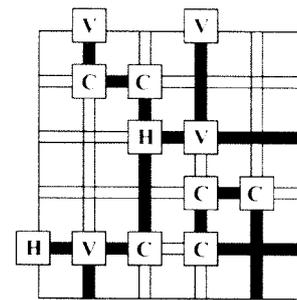


Figura 9 - Exemplo de um mapa de contornos contendo somente contornos fechados, e dos respectivos pontos de transição.

	(V)	(V)	C	C	H	V	C	C	(H)	V	C	C
# $T_G$	1	1	3	3	3	3	3	3	1	3	3	3
# $T_C$	1	1	2	2	2	3	1	2	1	2	3	2

Tabela I  
COMPARAÇÃO DA CARDINALIDADE DO CONJUNTO DE TIPOS POSSÍVEIS DOS PONTOS DE TRANSIÇÃO DO EXEMPLO DA FIG. 9, CONSIDERANDO: UM MAPA DE CONTORNOS GÊNICO ( $\#T_G$ ); UM MAPA DE CONTORNOS CONTENDO SOMENTE CONTORNOS FECHADOS ( $\#T_C$ ).

A Fig. 9 mostra um pequeno mapa de contornos formado só por contornos fechados, o qual pode ser representado pela seguinte sequência de símbolos:

0, (V), 1, (V), 2, C, 0, C, 4, H, 0, V, 4, C, 0, C, 0, (H), 0, V, 0, C, 0, C, 1

Repare-se que, para um mapa de contornos genérico (isto é, contendo qualquer tipo de contornos), a cardinalidade do conjunto de tipos possíveis para um dado ponto de transição é sempre três (excepto para os pontos de transição que ocorrem nos nós de partida, como já tínhamos visto). Contudo, para contornos fechados, essa cardinalidade varia entre um e três, dependendo do contexto em torno do ponto de transição em causa (ver Fig. 8).

A Tabela I compara a cardinalidade do conjunto de tipos possíveis de cada ponto de transição do mapa de contornos da Fig. 9, considerando: um mapa de contornos genérico ( $\#T_G$ ); um mapa só com contornos fechados ( $\#T_C$ ). Como pode ser observado, na maioria dos casos a cardinalidade é reduzida. Repare-se que quando a cardinalidade dos pontos de transição é um, podemos omitir o respectivo símbolo na sequência sem que exista perda de informação.

Uma das formas mais simples de representar eficientemente este tipo de sequências consiste na atribuição de códigos de comprimento variável a cada uma das duas componentes ( $d_i$  e  $t_i$ ). Os primeiros resultados que obtivemos usando o método dos pontos de transição foram conseguidos com base nesta abordagem. Para os símbolos  $t_i$ , foi utilizado o seguinte código (que mais não é que um código de Huffman): “0” para o símbolo C, “10” para o símbolo V e “11” para o símbolo H. A razão para a atribuição do código mais curto ao símbolo C é devido a, em geral, ele ser o mais

frequente.

Relativamente aos símbolos  $d_i$ , foi utilizado um código de comprimento variável do tipo do apresentado na Tabela II, cujo princípio de funcionamento é o seguinte. Começemos por definir uma base,  $B$ , e um incremento,  $I$ . O primeiro nível de códigos (nível 1) é constituído por todas as palavras de  $B$  bits, excepto a palavra formada só por 1's, a qual é usada como prefixo do próximo nível. O nível seguinte (nível 2) é formado por todas as palavras de  $B + I$  bits (excepto a palavra constituída só por 1's), concatenadas ao prefixo. Generalizando, o nível  $L$  é constituído por todas as palavras de  $B + I(L - 1)$  bits (excepto a palavra constituída só por 1's), concatenadas ao respectivo prefixo.

$d_i$	$B = 1, I = 1$		$B = 2, I = 1$		$B = 1, I = 2$	
	$L$	Código	$L$	Código	$L$	Código
0	1	0	1	00	1	0
1	2	100	1	01	2	1000
2	2	101	1	10	2	1001
3	2	110	2	11000	2	1010
4	3	111000	2	11001	2	1011
5	3	111001	2	11010	2	1100
6	3	111010	2	11011	2	1101
7	3	111011	2	11100	2	1110
8	3	111100	2	11101	3	111100000
9	3	111101	2	11110	3	111100001
10	3	111110	3	111110000	3	111100010
11		...		...		...

Tabela II

EXEMPLO DA CONSTRUÇÃO DO CÓDIGO DE PREFIXO EXPLICADO NO TEXTO.

**B. O mapa dos pontos de transição**

A estratégia de codificação que apresentámos atrás apresenta algumas vantagens, de entre as quais destacamos a simplicidade dos conceitos envolvidos, incluindo a possibilidade de entender e explorar de forma simples a interdependência entre os símbolos. Contudo, a sua eficiência em termos de compressão pode ser melhorada consideravelmente.

A estratégia de codificação que designamos por "mapa dos pontos de transição" baseia-se na construção de uma imagem com dois bits por elemento de imagem, de dimensões iguais às da imagem original (isto é,  $N_R \times N_C$ ), em que cada elemento de imagem contém informação respeitante a um nó. De facto, é fácil de constatar a partir da Fig. 3 que, a menos do canto superior esquerdo, o conjunto de nós pode ser considerado como uma imagem com estas características. Dos quatro valores que cada elemento de imagem pode tomar três correspondem aos tipos de pontos de transição, enquanto que o quarto é destinado a assinalar que um dado nó não é ponto de transição (símbolo "0").

A Fig. 10 mostra o mapa (simbólico) dos pontos de transição correspondentes ao mapa de contornos da Fig. 5 (os símbolos "0" não são mostrados por uma questão de simplicidade, correspondendo aos elementos de imagem

que aparecem em branco). A Fig. 11 apresenta um exemplo de um mapa de contornos de um segmento de  $64 \times 64$  da imagem "couple" (Fig. 15), assim como a imagem do respectivo mapa de pontos de transição.

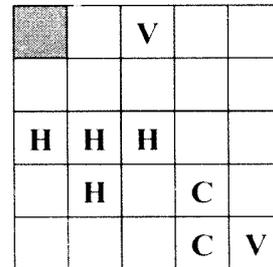


Figura 10 - Mapa de pontos de transição correspondente ao mapa de contornos da Fig. 5. O quadrado cinzento indica que o valor desse elemento é arbitrário. Os quadrados em branco correspondem ao símbolo "0", ou seja, à não existência de ponto de transição.

**B.1 Codificação usando a norma JBIG**

Uma das vantagens óbvias da utilização do mapa de pontos de transição para a representação de contornos é a possibilidade de este poder ser codificado usando uma (qualquer) técnica já existente para a compressão sem perdas de imagens. De facto, como vimos, o mapa dos pontos de transição é uma imagem com dois bits por elemento de imagem, pelo que a sua compressão pode ser encarada dessa forma simples. Repare-se que o esforço necessário para a construção do mapa de pontos de transição a partir do mapa de contornos é extremamente reduzido (para implementar essa transformação são necessárias menos de cem linhas de programa). Por outro lado, encontram-se disponíveis variadíssimas técnicas para a compressão sem perdas de imagem, pelo que o esforço necessário para a implementação dessa operação pode ser considerado aproximadamente zero. De entre essas técnicas, escolhemos o JBIG, devido a ser uma norma e também devido a ser bastante eficiente.

A norma JBIG (Joint Bi-level Image Experts Group) foi publicada em 1993 pelo ISO/IEC<sup>1</sup> e ITU-T<sup>2</sup>, com vista à compressão progressiva de imagens binárias e de níveis de cinzento, tipicamente com menos de seis bits por elemento de imagem. As maiores vantagens desta técnica, face a outras existentes tais como as de FAX Grupo 3/4 e JPEG sem perdas, são a sua capacidade de codificação progressiva e a sua maior eficiência de compressão [19]-[21].

O núcleo do JBIG é formado por um codificador aritmético binário adaptativo, o qual utiliza 1024 contextos (correspondentes a 10 elementos de imagem) no modo de operação sequencial ou na camada de mais baixa resolução no modo progressivo, ou 4096 contextos durante a codificação de camadas de alta resolução no modo progressivo. A Fig. 12 mostra as duas configurações usadas para o cálculo dos contextos de 10 elementos de imagem. O "X" indica o elemento de imagem a ser codificado, enquanto que o "A" in-

<sup>1</sup>International Organization for Standardization : International Electrotechnical Commission.

<sup>2</sup>Telecommunication Standardization Sector of the International Telecommunication Union.

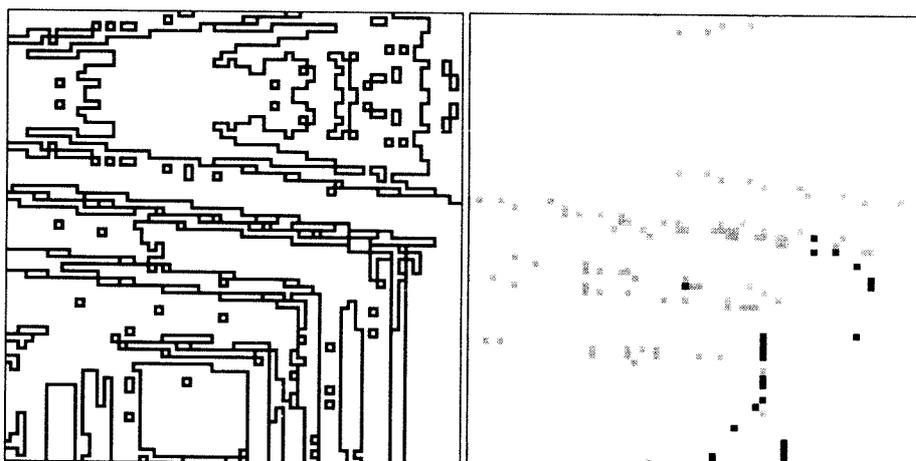


Figura 11 - À esquerda, o mapa de contornos (para uma dada segmentação) de um segmento de  $64 \times 64$  da imagem "couple". À direita, a imagem do mapa de pontos de transição correspondente.

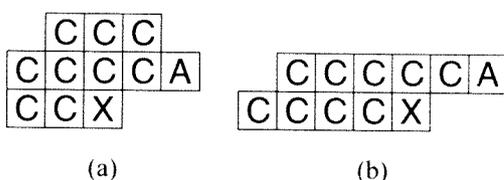


Figura 12 - As configurações de (a) três e (b) duas linhas usadas pelo JBIG para cálculo dos contextos e das respectivas probabilidades. O "X" marca o elemento de imagem a ser codificado. Os elementos marcados com um "C" correspondem à parte fixa da configuração. O elemento marcado com um "A" pode mudar a sua posição, adaptativamente.

dica um elemento da configuração que pode mudar a sua posição adaptativamente, por exemplo, para permitir explorar regularidades em padrões de *half-toning*. Os elementos marcados com um "C" correspondem à parte fixa da configuração.

Antes da compressão poder ser efectuada usando o JBIG, é necessário transformar o mapa de pontos de transição de uma representação simbólica, como a apresentada na Fig. 10, para uma representação binária, isto é, estabelecer uma correspondência entre o conjunto de símbolos  $\{C, H, V, \emptyset\}$  e o conjunto de palavras binárias  $\{00, 01, 10, 11\}$ . Em geral, a eficiência de compressão depende desta correspondência, uma vez que o JBIG codifica as imagens de níveis de cinzento (como é o caso dos mapas de pontos de transição) através de uma decomposição prévia nos vários planos de imagem, os quais são posteriormente codificados independentemente uns dos outros. A determinação da melhor correspondência pode ser realizada, por exemplo, através de uma análise das probabilidades de ocorrência dos vários símbolos. Por exemplo, se o código "00" é atribuído ao símbolo mais provável, então o código "11" deve ser atribuído ao símbolo menos provável, por forma a minimizar o número de transições  $0 \rightarrow 1$  e  $1 \rightarrow 0$  em cada plano de imagem.

## B.2 Codificação aritmética adaptativa quaternária com contextos no domínio dos pontos de transição

O método descrito na secção anterior, embora seja bastante atractivo dada a sua simplicidade de implementação (obviamente, desde que se disponha de um codificador JBIG), apresenta uma lacuna no que respeita à eficiência de compressão. De facto, ao ser efectuada uma decomposição plano a plano das imagens correspondentes aos mapas de pontos de transição, sendo esses planos codificados independentemente uns dos outros, alguma da redundância existente entre os quatro símbolos não é devidamente explorada.

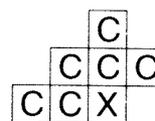


Figura 13 - Configuração utilizada para o cálculo dos contextos usados pelo codificador aritmético adaptativo quaternário. Neste caso, os contextos são calculados no mesmo domínio, isto é, no domínio dos pontos de transição. O "X" marca o a ser codificado, enquanto que o "C" indica nós usados para cálculo dos contextos.

Por esta razão, desenvolvemos um codificador aritmético adaptativo de quatro símbolos, associado a 4096 contextos cuja configuração é apresentada na Fig. 13. Repare-se que a diferença fundamental entre esta abordagem e a anterior reside na utilização de um alfabeto quaternário em vez do alfabeto binário utilizado pelo codificador aritmético do JBIG. Note-se também que o número de elementos utilizados para calcular os contextos é agora menor do que o utilizado pelo JBIG (seis, em vez de dez). No entanto, o número de contextos é superior ( $4^6 = 4096$ , em vez de  $2^{10} = 1024$ ). Esta configuração foi escolhida de forma empírica, através de experiências cujos resultados não são aqui incluídos. Verificou-se, na prática, que um maior número de contextos não melhora ou mesmo piora o desempenho, devido ao conhecido fenómeno da "diluição dos contextos".

### B.3 Codificação aritmética adaptativa quaternária com contextos no domínio dos elementos de contorno

É sabido que, para um dado modelo probabilístico da fonte de informação, a codificação aritmética consegue gerar taxas de compressão muito próximas da entropia desse modelo. Portanto, o problema da procura de um método de compressão eficiente reside, fundamentalmente, na procura de um modelo probabilístico apropriado para a fonte de informação em causa.

Com base no codificador aritmético quaternário que utilizámos atrás, procurámos encontrar um modelo probabilístico mais eficiente, tendo em conta a relação natural existente entre os elementos do mapa de contornos e os pontos de transição desse mesmo mapa. Assim, embora a codificação continue a ser realizada no domínio dos pontos de transição (quaternário), os contextos são agora calculados no domínio dos elementos de contorno (binário). A configuração utilizada para calcular os contextos encontra-se representada na Fig. 14.

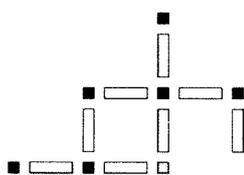


Figura 14 - Configuração utilizada para o cálculo dos contextos usados pelo codificador aritmético adaptativo quaternário. Neste caso, os contextos são calculados no domínio dos elementos de contorno (domínio binário), embora os símbolos a codificar pertençam a outro domínio, isto é, ao domínio dos pontos de transição (domínio quaternário). Os rectângulos brancos indicam os elementos de contorno usados para o cálculo dos contextos. O quadrado cinzento marca a posição do nó a ser codificado, enquanto que os quadrados pretos marcam a posição de outros nós já codificados (indicados aqui só para mais fácil orientação do leitor).

Como se pode ver, o número de contextos que resulta da configuração da Fig. 14 é bastante inferior aos que foram usados até agora. Uma vez que se trata de um domínio binário com oito elementos, temos um total de  $2^8 = 256$  contextos. No entanto, este modelo é consideravelmente mais eficiente que os outros que descrevemos, como poderá ser confirmado pelos resultados experimentais que apresentamos em seguida.

### C. Resultados experimentais

A Fig. 15 apresenta as quatro imagens ( $256 \times 256$  elementos de imagem, 8 bits por elemento de imagem) que foram usadas para a obtenção de resultados experimentais relativos à compressão de mapas de contornos: “Lena”, “fish”, “camera” e “couple”. Foram gerados vários mapas de contornos a partir destas imagens, os quais foram codificados usando o método da codificação de cadeias e as estratégias de codificação dos pontos de transição que descrevemos atrás<sup>3</sup>.

Para obter os mapas de contornos, as imagens foram uniformemente quantificadas com base na seguinte expressão,

<sup>3</sup>Na realidade, não apresentamos resultados relativamente à primeira estratégia que foi aqui descrita em virtude de esta ser bastante menos eficiente que as outras. No entanto, esses resultados podem ser consultados em [17].

a qual foi aplicada a todos os elementos de imagem:

$$\hat{g}(r, c) = \left\lfloor \frac{g(r, c)}{2q + 1} \right\rfloor (2q + 1) + q \quad (4)$$

em que  $g(r, c)$  e  $\hat{g}(r, c)$  representam, respectivamente, os elementos da imagem original e os elementos da imagem quantificada, e  $q$  denota o erro máximo absoluto de quantificação, ou seja,

$$|g(r, c) - \hat{g}(r, c)| \leq q, \quad \forall (r, c) \quad (5)$$

Após a quantificação ter sido realizada, as fronteiras das regiões foram detectadas e codificadas.

Como se pode observar nos gráficos da Fig. 16, a eficiência de compressão do método baseado nos pontos de transição é bastante superior à conseguida pelo método da codificação de cadeias. Esta superioridade é particularmente evidente quando os mapas de contornos apresentam um elevado número de elementos de contorno activos, ou seja, quando são mais complexos.

De entre as várias estratégias estudadas para a codificação dos pontos de transição, aquela que se fundamenta na codificação aritmética adaptativa quaternária com contextos no domínio dos elementos de contorno é, claramente, a melhor. Repare-se também que, como seria de esperar, a estratégia baseada na codificação aritmética adaptativa quaternária com contextos no domínio dos pontos de transição apresenta resultados ligeiramente melhores que o JBIG. No entanto, o grande salto em termos de compressão só é conseguido quando o domínio dos contextos passa a ser o domínio dos elementos de contorno, o que evidencia a importância fulcral da escolha de um modelo apropriado de geração de probabilidades.

## IV. CONCLUSÕES

Neste trabalho apresentámos uma abordagem completamente nova ao problema da representação eficiente de contornos arbitrários. Mostrámos que o método proposto, o qual designámos de “pontos de transição” é claramente superior ao conhecido método da codificação de cadeias. Na realidade, de acordo com o nosso conhecimento, o método dos pontos de transição é, actualmente, o método mais eficiente para a representação de contornos arbitrários.

Tal como foi referido na Introdução, uma das principais aplicações da codificação de contornos é a codificação de imagem baseada na decomposição das cenas visuais em regiões de forma e tamanho variáveis. Por conseguinte, uma das linhas de trabalho que pretendemos desenvolver envolve precisamente essa área, sendo dada uma atenção especial aos casos em que são exigidas segmentações bastante complexas. De facto, e como foi mostrado neste artigo, é nestes casos que a eficiência do método dos pontos de transição mais se destaca.

## AGRADECIMENTOS

O autor deseja expressar o seu agradecimento às seguintes pessoas:

- Markus Kuhn por colocar em domínio público o software JBIG-KIT (<ftp://ftp.informatik>).

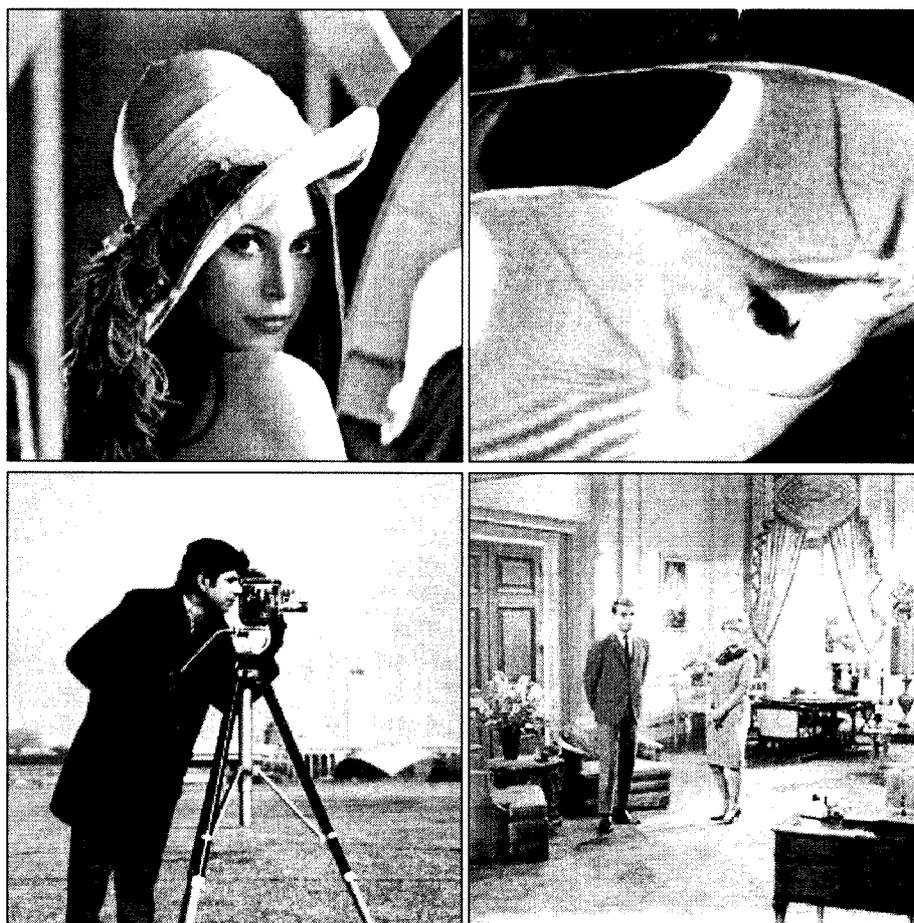


Figura 15 - Imagens utilizadas para realizar os testes de codificação de contornos com o método da codificação de cadeias e com o método baseado nos pontos de transição: (da esquerda para a direita, de cima para baixo): "Lena", "fish", "camera" e "couple".

uni-erlangen.de/pub/doc/ISO/JBIG), o qual foi usado pelo autor para a codificação / descodificação de imagens em formato JBIG.

- A Alistair Moffat *et al.* por colocarem em domínio público o software relativo a um codificador aritmético (<http://www.cs.mu.oz.au/~alistair>), o qual foi adaptado e utilizado pelo autor para efectuar codificação / descodificação aritmética.

#### REFERÊNCIAS

- [1] M. Kunt, A. Ikonopoulou, e M. Koehler, "Second-generation image-coding techniques", *Proceedings of the IEEE*, vol. 73, no. 4, pp. 549-574, Abril de 1985.
- [2] M. Kunt, M. Benard, e R. Leonardi, "Recent results in high-compression image coding", *IEEE Trans. on Circuits and Systems*, vol. 34, no. 11, pp. 1306-1336, November de 1987.
- [3] M. J. Biggar, O. J. Morris, e A. G. Constantinides, "Segmented-image coding: performance comparison with the discrete cosine transform", *IEE Proc. - F, Communications, Radar and Signal Processing*, vol. 135, no. 2, pp. 121-132, Abril de 1988.
- [4] O. Egger, F. Bossen, e T. Ebrahimi, "Region based coding scheme with scalability features", em *Signal Processing VIII - Theories and Applications, Proc. of the 8th European Signal Processing Conf., EUSIPCO'96*, G. Ramponi, G. L. Sicuranza, S. Carrato, e S. Marsi, Eds., Trieste, Italy, 1996, vol. II, pp. 747-750, Edizioni LINT Trieste.
- [5] H. Freeman, "On the encoding of arbitrary geometric configurations", *IRE Trans. on Electronic Computers*, vol. 10, pp. 260-268, Jun. de 1961.
- [6] T. Kaneko e M. Okudaira, "Encoding of arbitrary curves based on the chain code representation", *IEEE Trans. on Communications*, vol. 33, no. 7, pp. 697-707, Jul. de 1985.
- [7] M. Eden e M. Koehler, "On the performance of a contour coding algorithm in the context of image coding. Part I: contour segment coding", *Signal Processing*, vol. 8, no. 4, pp. 381-386, Jul. de 1985.
- [8] F. Marqués, J. Saulea, e A. Gasull, "Shape and location coding for contour images", em *Picture Coding Symposium*, Lausanne, Switzerland, Mar. de 1993, pp. 18.6.1-18.6.2.
- [9] T. Minami e K. Shinohara, "Encoding of line drawings with a multiple grid chain code", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 8, no. 2, pp. 269-276, Mar. de 1986.
- [10] V. A. Christopoulos, C. A. Christopoulos, J. Cornelis, e A. N. Skodras, "A new contour simplification filter for region-based coding", em *Signal Processing VIII - Theories and Applications, Proc. of the 8th European Signal Processing Conf., EUSIPCO'96*, G. Ramponi, G. L. Sicuranza, S. Carrato, e S. Marsi, Eds., Trieste, Italy, 1996, vol. I, pp. 336-339, Edizioni LINT Trieste.
- [11] A. K. Jain, *Fundamentals of digital image processing*, Prentice-Hall, Inc., 1989.

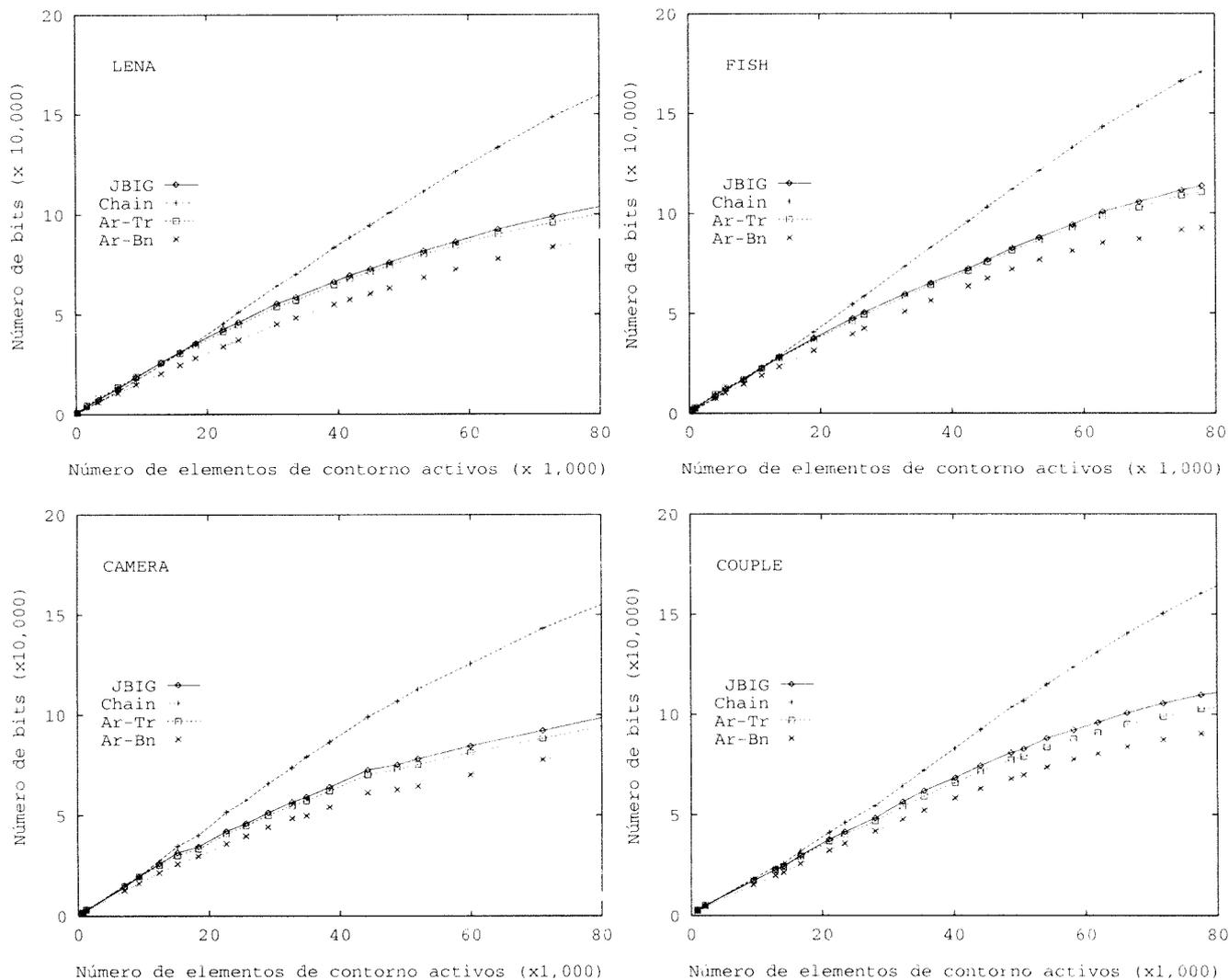


Figura 16 - Número de bits, em função do número de elementos de contorno activos, necessário para codificar vários mapas de contornos das imagens de teste ("Lena", "fish", "camera" e "couple") usando: JBIG ("JBIG"); codificação de cadeias ("Chain"); codificação aritmética adaptativa quaternária com contextos no domínio dos pontos de transição ("Ar-Tr"); codificação aritmética adaptativa quaternária com contextos no domínio dos elementos de contorno ("Ar-Bn").

- [12] P. Salembier, F. Marqués, e A. Gasull, "Coding of partition sequences", em *Video Coding: The second generation approach*, L. Torres e M. Kunt, Eds., pp. 125-170. Kluwer Academic Publishers, 1996.
- [13] R. R. Estes Jr. e V. R. Algazi, "Efficient error free chain coding of binary documents", em *Proc. of the Data Compression Conf., DCC'95*, Snowbird, Utah, Mar. de 1995, pp. 122-131.
- [14] J. A. Robinson, "Image coding with ridge and valley primitives", *IEEE Trans. on Communications*, vol. 43, no. 6, pp. 2095-2102, Jun. de 1995.
- [15] J.-W. Chung, J.-H. Moon, e J.-K. Kim, "Conditional differential chain coding for lossless representation of object contour", *Electronics Letters*, vol. 34, no. 1, pp. 55-56, Jan. de 1998.
- [16] A. K. Katsaggelos, L. P. Kondi, F. W. Meier, J. Ostermann, e G. M. Schuster, "MPEG-4 and rate-distortion-based shape-coding techniques", *Proceedings of the IEEE*, vol. 86, no. 6, pp. 1126-1154, Jun. de 1998.
- [17] A. J. Pinho, "A method for encoding region boundaries based on transition points", *Image and Vision Computing*, vol. 16, no. 3, pp. 213-218, March de 1998.
- [18] A. J. Pinho, "A JBIG-based approach to the encoding of contour maps", *IEEE Trans. on Image Processing*, vol. 9, no. 5, pp. 936-941, May de 2000.
- [19] H. Hampel, R. B. Arps, C. Chamzas, D. Dellert, D. L. Duttweiler, T. Endoh, W. Equitz, F. Ono, R. Pasco, I. Sebestyen, C. J. Starkey, S. J. Urban, Y. Yamazaki, e T. Yoshida, "Technical features of the JBIG standard for progressive bi-level image compression", *Signal Processing: Image Communication*, vol. 4, no. 2, pp. 103-111, Abr. de 1992.
- [20] International Standard ISO/IEC 11544:1993 and ITU-T Recommendation T.82(1993), *Information technology - Coded representation of picture and audio information - progressive bi-level image compression*, Mar. de 1993.
- [21] A. N. Netravali e B. G. Haskell, *Digital pictures: representation, compression and standards*, Plenum, New York, 2nd edição, 1995.