



CAPÍTULO 3

Morfologia Matemática

Contribuíram:

Agnaldo Izidoro de Souza, Cleiton Almeida dos Santos

O que é Morfologia Matemática ?

O estudo morfológico concentra-se na estrutura geométrica das imagens. A morfologia pode ser aplicada em diversas áreas de processamento de imagens, como realce, filtragem, segmentação, esqueletização e outras afins. Morfologia é a forma e estrutura de um objeto ou os arranjos e inter-relacionamentos entre as partes de um objeto. Os mais antigos usos desta palavra estão relacionados com linguagem e biologia. Em lingüística, morfologia é o estudo da estrutura das palavras. Em biologia, morfologia esta relacionada mais diretamente à forma de um organismo. A forma de uma folha pode ser usada para identificar uma planta ou a forma de uma colônia de bactérias pode ser usada para identificar sua variedade.

Morfologia digital é um caminho para descrever ou analisar a forma de um objeto digital. A morfologia digital é uma ciência relativamente recente, pois só os computadores digitais permitiram seu uso na prática. Por outro lado os matemáticos a consideram uso da teoria de conjuntos que é uma area bem estudada.

A idéia de morfologia digital é que uma imagem consiste de um conjunto de "picture elements" (pixels) que são reunidos em grupos tendo uma estrutura bidimensional (forma). Certas operações matemáticas em conjuntos de pixels podem ser usadas para ressaltar aspectos



específicos das formas permitindo que sejam contadas ou reconhecidas. A base da morfologia consiste em extrair de uma imagem desconhecida a sua geometria através da utilização da transformação de uma outra imagem completamente definida, seja, consiste em extrair as informações relativas a geometria e a topologia de um conjunto desconhecido (no caso uma imagem) pela transformação através de outro conjunto bem-definido, chamado *elemento estruturante*. Com isso torna importante ao contexto a utilização de teoria dos conjuntos, pois esta é a base utilizada na morfologia, assim é com esta teoria que será descrita e apresentada uma imagem. Por exemplo a definição de um vetor bidimensional onde será exposta as coordenadas (x, y) para sua representação gráfica.

As operações básicas da morfologia digital são a *erosão*, em que pixels que não atendem a um dado padrão são apagados da imagem, e *dilatação*, em que uma pequena area relacionada a um pixel é alterada para um dado padrão. Todavia, dependendo do tipo de imagem sendo processada (preto e branco, tons de cinza ou colorida) a definição destas operações muda, assim cada tipo deve ser considerado separadamente.

Tipos de Operações Morfológicas

Operações morfológicas estão divididas em operações morfológicas binárias e operações morfológicas sobre imagens coloridas ou de tons de cinza. Há uma diferença de enfoque nos dois grupos.

Operações morfológicas binárias são definidas em imagens com pixels pretos e brancos somente. Um objeto é considerado ser um conjunto matemático de pixels pretos, cada pixel é identificado pelos seus índices de linha e coluna sendo chamado de um ponto no espaço bidimensional (E^2). Ex. $A = \{(1,1), (3,3), (7,3)\}$. Aqui um pixel será, quando for afetado por uma operação, substituído pelo seu valor oposto.

Nas operações morfológicas sobre imagens com tons de cinza ou cores, as operações de conjunto realizadas não implicam na retirada ou inclusão de um pixel (substituição de seu valor por preto ou branco), mas sim na modificação parcial de seu valor.



Organização deste Capítulo

Antes de conhecer as operações básicas da morfologia binária precisamos rever algumas definições básicas com conjuntos matemáticos. Imagens com níveis de cinza serão representados em vetores de 3 elementos, sendo os dois primeiros coordenadas do pixel e o terceiro o do tom de cinza. Assim será feito um estudo superficial na teoria dos conjuntos, pois não é o escopo deste capítulo fazer um amplo estudo sobre conjuntos, mas somente dar uma noção da utilização na morfologia para seu entendimento.

Após estes conceitos matemáticos serão explanados os conceitos básicos e fundamentais da morfologia que é a dilatação a erosão e seus correlatos pois entende-se que estes sejam um dos principais pontos da morfologia. Em segundo nível serão visto o funcionamento dos algoritmos de abertura e fechamento, para assim fechar os principais métodos morfológicos, pois qualquer algoritmo de processamento de imagens podem ser descrito com essas combinações e correlatas.

Após o entendimento destas definições básicas que é a base da morfologia será incorporado ao trabalho vários algoritmos morfológico como (limiarização, erro e acerto, engorda, emagrece e outros).

Concluindo será feito um estudo sobre morfologia em tom de cinza, com isso pretende-se dar uma noção geral das principais formas de morfologia e suas aplicações no contexto de processamento digital de imagens.

Teoria dos Conjuntos

Conjunto como próprio nome diz é uma coleção de objetos sem repetição. Se um conjunto não é muito grande, este pode ser descrito listando-se os seus elementos em qualquer ordem por exemplo:

$$A = \{1, 2, 3, 4\}$$

Se um conjunto é muito grande ou infinito, este pode ser descrito através de propriedades requeridas de seus membros:



$X = \{x \mid x \text{ e um } n^{\circ} \text{ inteiro, positivo e par}\}$

$x \in X$, se x esta em X , caso contrario $x \notin X$.

Exemplos:

Se $A = \{x \mid x^2 + x - 6 = 0\}$ e $B = \{2, -3\}$ então $A = B$

Se $C = \{1, 3\}$ e $A = \{1, 2, 3, 4\}$ então C e um subconjunto de A .

Combinação de Conjuntos

Sejam dois conjuntos X e Y :

UNIÃO

A união de dois conjuntos X e Y é o conjunto de elementos que pertencem ou X ou Y ou ambos:

$$X \cup Y = \{x \mid (x \in X) \text{ ou } (x \in Y)\}$$

Exemplo:

Considere dois conjuntos $A = \{1, 3, 5\}$ e $B = \{4, 5, 6\}$ então

$$A \cup B = \{1, 3, 4, 5, 6\}$$

INTERSEÇÃO

A interseção de dois conjuntos X e Y é o conjunto de elementos pertencentes a ambos X e Y :

$$X \cap Y = \{x \mid (x \in X) \text{ e } (x \in Y)\}$$

Exemplo:

Considere dois conjuntos $A = \{1, 3, 5\}$ e $B = \{4, 5, 6\}$ então

$$A \cap B = \{5\}$$

DIFERENÇA

A diferença entre os conjuntos X e Y é o conjunto de elementos que pertencem a X mas não pertencem a Y :



$$X - Y = \{ x \mid (x \in A) \text{ e } (x \notin B) \}$$

Exemplo:

Considere dois conjuntos $A = \{ 1, 3, 5 \}$ e $B = \{ 4, 5, 6 \}$ então

$$A - B = \{ 1, 3 \}$$

$$B - A = \{ 4, 6 \}$$

COMPLEMENTO

O complemento do conjunto X é o conjunto dos elementos não pertencentes ao conjunto X . Se nos pensarmos em imagem este poderia corresponder aos pixels brancos na figura ou na linguagem da teoria dos conjuntos:

$$A^c \text{ ou } X' = \{ x \mid x \notin X \}$$

Exemplo:

Seja o conjunto $V = \{ 1, 2, 3, 4, 5 \}$ e $A = \{ 1, 3, 5 \}$ então

$A' = \{ 2, 4 \}$ e o complemento de A em relação a V .

DISJUNÇÃO

São disjuntos quando não existem elementos comuns entre eles.

$X \cap Y = \emptyset$ então X e Y são disjuntos.

Exemplo:

Seja o conjunto $A = \{ 1, 4, 5 \}$ e $B = \{ 2, 6 \}$ então

A e B são disjuntos.

TRANSLAÇÃO

A translação do conjunto X pelo ponto x é definida, em notação de conjuntos, como:

$$X' = \{ y \mid y = a + x, x \in X \}$$

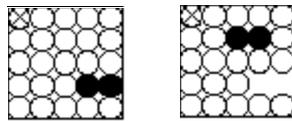
Exemplo:



Seja o conjunto $X = \{1, 2\}$

então todos os pixels da imagem serão movidos uma coluna para a direita e duas linhas para baixo conforme exemplificado abaixo com uma imagem:

Figura 3.1. **Translação**

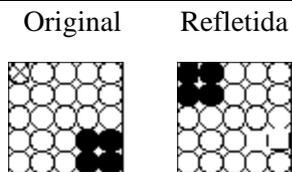


REFLEXÃO

A reflexão do conjunto X é definida como:

$$X' = \{ y \mid y = -x, x \in X \}$$

Figura 3.2. **Esta é uma rotação de 180° sobre a origem**



Operações Básicas da Morfologia Matemática

A Morfologia Matemática se caracteriza por um conjunto específico de operações sobre imagens enquanto conjuntos de pixels. Podemos dividi-las em operações realizadas sobre imagens binárias e operações realizadas em imagens em tons de cinza e coloridas. Começaremos examinando as operações sobre imagens binárias. Para cada operação binária existe uma correspondente para tons de cinza e imagens em cores.



Dilatação Binária (\oplus)

A dilatação, também as vezes chamada de *dilatação*, é uma transformação morfológica que combina dois conjuntos usando adição vetorial. Seu símbolo é \oplus . Como o nome diz, o resultado será uma imagem “engordada”.

A dilatação de um conjunto A pelo conjunto B é definida por:

$$A \oplus B = \{ c \mid c = a + b, a \in A, b \in B \}$$

Onde A representa a imagem sendo operada e B é um segundo conjunto onde é chamado *elemento estrutural* e sua composição define a natureza específica da dilatação, sendo assim a dilatação expande uma imagem. Ela pode ser representada pela união $A \oplus B = \cup B$.

Seja o conjunto $A = \{ (0,1), (1,1), (2,1), (2,2), (3,0) \}$ e $B = \{ (0,0), (0,1) \}$ então o resultante da dilatação é:

$$A \oplus B = \{ A + \{ (x_1 \in B) \} \cup A + \{ (x_2 \in B) \} \}$$

$$A \oplus B = \{ (0,1), (1,1), (2,1), (3,0), (0,2), (1,2), (2,2), (2,3), (3,1) \}$$

Figura 3.3.

Exemplo: $A \oplus B$

$$A = \{ (0,1), (1,1), (2,1), (2,2), (3,0) \} =$$


$$B = \{ (0,0), (0,1) \} =$$


$$A \oplus B = \{ (0,1), (1,1), (2,1), (3,0), (0,2), (1,2), (2,2), (2,3), (3,1) \} =$$


logo:

$$A \oplus B = [A + \{ (0,0) \}] \cup [A + \{ (1,0) \}] \text{ resulta em:}$$

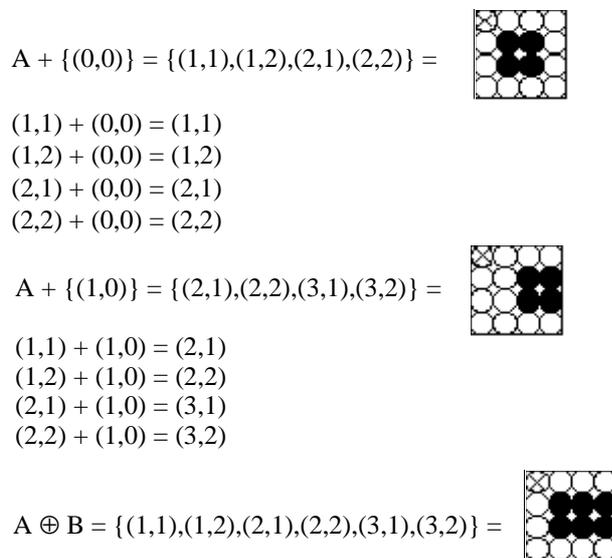
$$A \oplus B = \{ (1,1), (1,2), (2,1), (2,2), (3,1), (3,2) \}$$



O pixel marcado com um “x” representa a origem (0,0) de cada imagem. A localização da origem é muito importante; no exemplo anterior se a origem do conjunto B fosse o pixel da direita, $\{(-1,0),(0,0)\}$, a dilatação acrescentaria pixels a esquerda na imagem A. Pode-se visualizar todo o processo da melhor forma graficamente:

Figura 3.4.

Seqüência de passos na Dilatação de A por B



A Figura 3.5. mostra mais dois exemplos de dilatação binária que devem esclarecer quaisquer dúvidas que ainda persistam.

Erosão Binária (\ominus)

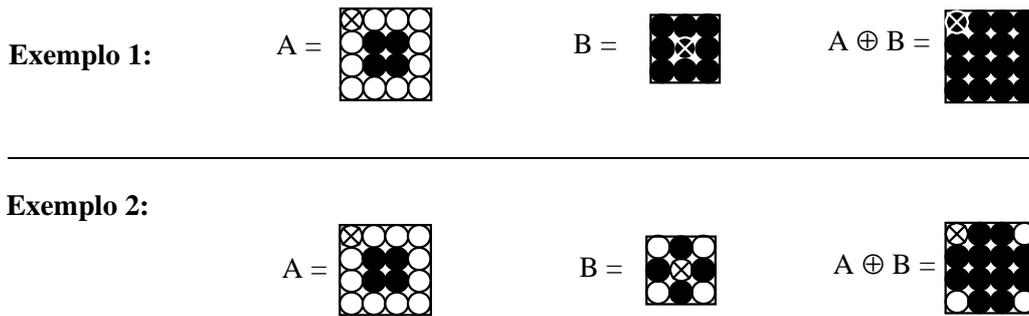
A erosão basicamente encolhe uma imagem e pode ser vista como uma transformação morfológica que combina dois conjuntos usando vetores de subtração. Ela é expressa como a interseção de A e B. Assim e definido $A \ominus B = B \cap A$.

A erosão da imagem A pelo elemento estrutural B pode ser definida como:

$$A \ominus B = \{ x / x + b \in A \text{ para todo } b \in B \}$$



Figura 3.5. Mais dois exemplos de dilatação binária



$$A = \{(1,1), (1,2), (2,1), (2,2)\}$$

$$B = \{(-1,0), (0,-1), (0,1), (1,0)\}$$

$$A \oplus B = [A + \{(-1,0)\}] \cup [A + \{(0,-1)\}] \cup [A + \{(0,1)\}] \cup [A + \{(1,0)\}]$$

$$A + \{(-1,0)\} = \{(0,1), (0,2), (1,1), (1,2)\}$$

$$A + \{(0,-1)\} = \{(1,0), (1,1), (2,0), (2,1)\}$$

$$A + \{(0,1)\} = \{(1,2), (1,3), (2,2), (2,3)\}$$

$$A + \{(1,0)\} = \{(2,1), (2,2), (3,1), (3,2)\}$$

$$A \oplus B = \{(0,1), (0,2), (1,0), (1,1), (1,2), (1,3), (2,0), (2,1), (2,2), (2,3), (3,1), (3,2)\}$$

ou

$$A \ominus B = \{c \mid B' \subseteq A\} \text{ definição de Matheron (1967).}$$

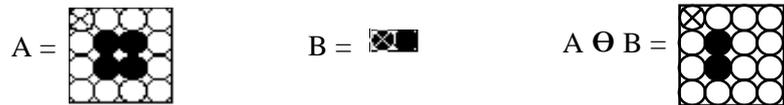
Assim define-se que a erosão é o conjunto de todos os pixels, e o elemento estruturante B e transladado pelo c corresponde a um conjunto de pixel em A.

Podemos visualizar a erosão através do exemplo dado a seguir. Assim verifica-se que cada elemento de B e um elemento de A ou seja B esta contido em A.

A Figura 3.7. mostra mais alguns exemplos de erosão binária.



Figura 3.6. Exemplo de erosão binária



O conjunto $A \ominus B$ é o conjunto de translações de B que alinham B sobre o conjunto de pixels pretos em A . Isso Significa que nem todas as translações necessitam ser consideradas, mas somente aquelas que inicialmente localizam sua origem de B em um membro de A . Existem quatro dessas translações:

$B_{(1,1)} = \{(1,1), (2,1)\}$
 Como os pixels (1,1) e (2,1) são pretos na imagem A .
 O pixel (1,1) no resultado será preto.



$B_{(1,2)} = \{(1,2), (2,2)\}$
 Como os pixels (1,2) e (2,2) são pretos na imagem A .
 O pixel (1,2) no resultado será preto.



$B_{(2,1)} = \{(2,1), (3,1)\}$
 Como o pixel (3,1) não é preto na imagem A .
 O pixel (2,1) no resultado não será preto.



$B_{(2,2)} = \{(4,4), (5,4)\}$
 Como o pixel (5,4) não é preto na imagem A .
 O pixel (4,4) no resultado não será pretor



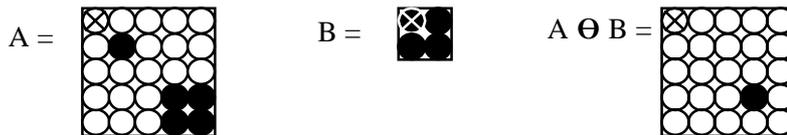
$A \ominus B = \{ (1,1) \mid B_{(1,1)} \subseteq A \} \cup \{ (1,2) \mid B_{(1,2)} \subseteq A \} =$





Figura 3.7. Outros exemplos de erosão binária

Exemplo 1:



$$A = \{(1,1), (3,3), (3,4), (4,3), (4,4)\}$$

$$B = \{(0,0), (0,1), (1,0), (1,1)\}$$

$$B_{(1,1)} = \{(1,1), (1,2), (2,1), (2,2)\}$$

$$B_{(3,3)} = \{(3,3), (3,4), (4,3), (4,4)\}$$

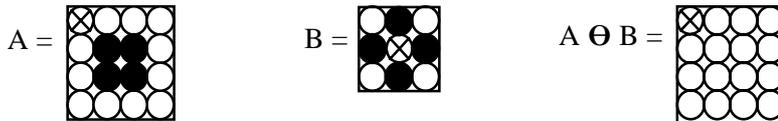
$$B_{(3,4)} = \{(3,4), (3,5), (4,4), (4,5)\}$$

$$B_{(4,3)} = \{(4,3), (4,4), (5,3), (5,4)\}$$

$$B_{(4,4)} = \{(4,4), (4,5), (5,4), (5,5)\}$$

$$A \ominus B = \{(3,3) \mid B_{(3,3)} \subseteq A\}$$

Exemplo 2:



$$A = \{(3,3), (3,4), (4,3), (4,4)\}$$

$$B = \{(-1,0), (0,-1), (0,1), (1,0)\}$$

$$B_{(3,3)} = \{(2,3), (3,2), (3,4), (4,3)\}$$

$$B_{(3,4)} = \{(2,4), (3,3), (3,5), (4,4)\}$$

$$B_{(4,3)} = \{(3,3), (4,2), (4,4), (5,3)\}$$

$$B_{(4,4)} = \{(3,4), (4,3), (4,5), (5,4)\}$$

$$A \ominus B = \emptyset$$

Exemplo 3:





Dualidade entre Dilatação e Erosão

A erosão não é o inverso da dilatação, somente em alguns casos de erosão isso ocorre. Erosão e dilatação são duais no seguinte sentido:

$$(A \ominus E)^c = A^c \oplus \hat{E}$$

Isso significa que o complemento de uma erosão é o mesmo que uma dilatação do complemento da imagem pelo elemento estrutural refletido.

Quando os operadores são dual de outros, estes meios que um pode ser escrito em termos de um outro. Dilatação e erosão são relatados em seguida. Se B denota a reflexão de B.

$$(A \ominus E)' = A' \oplus E$$

$$(A \oplus E)' = A' \ominus E$$

Em outras palavras, dilatação a primeiro plano esta a mesmo como erosão a plano fundo, mas o elemento estruturante reflete entre estas elas. Igualmente, erosão a plano fundo e dilatação a primeiro plano.

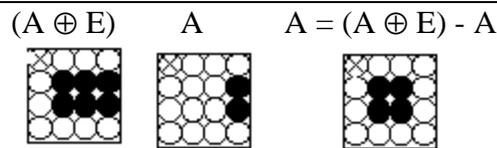
Usando Dilatação e Erosão

Suponha que você queira achar todos pixels do um limite de um objeto. Nos poderíamos executar a seguinte operação:

$$\text{Bound}_{\text{EXT}} A = (A \oplus E) - A$$

Figura 3.8.

Exemplo de um detetor de bordas externas usando morfologia matemática binária



Onde B tem uma matriz 3X3 como elemento estruturante contando todos 1s. Isto iria dar todos pixel que limitam o objeto. Ou se quisermos todos os pixels que limitam o pano fundo, nos usariamos,



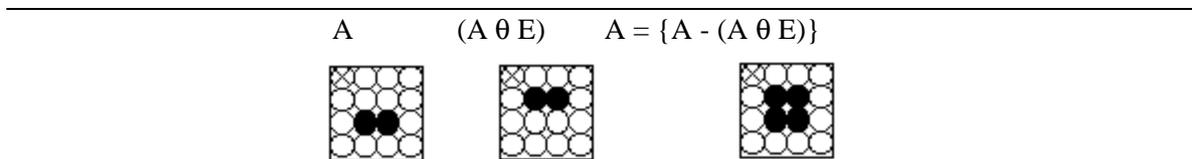
com isso é possível extrair contorno de um conjunto A, executando uma erosão de A por E, então calculando a diferença entre A e sua erosão:

$$\text{Bound}_{\text{INT}} A = \{A - (A \ominus E)\}$$

Onde E é um elemento estruturante adequado.

Figura 3.9.

Exemplo de um detetor de bordas internas usando morfologia matemática binária



Dilatação Condicional ($\oplus | I$)

Condicional de Dilatação essencialmente envolve a dilatação de uma imagem seguida por uma interseção com alguma outra condição da imagem. Em outras palavras novos pixels são adicionados ao original.

Deste modo a imagem viria a ser uma máscara sobre a dilatação. Isto é escrito como:

$$A \oplus | I B = (A \oplus B) \cap I$$

Aqui isto é um exemplo de como podemos usar condicional de dilatação. Suponha que antes de aplicar certo operador morfológico para achar um ponto específico na imagem nos queremos achar o componente conectado original dentro deste ponto. Na essência, nos queremos crescer este ponto fora para encher o componente conectado original. Um óbvio modo iria ser para combinar o componente conectado rotulando com operador morfológico, entretanto você pode fazer isto com o condicional de dilatação.

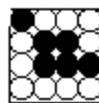
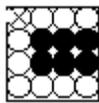


$$A \oplus |_I B = (A \oplus B) \cap I$$

Figura 3.10.

Exemplo de dilatação condicional.

$$(A \oplus B) \quad I = \text{Outra condição imagem} \quad A \oplus |_I B = (A \oplus B) \cap I$$



Gradiente Morfológico

Esta operação é composta de três outras operações básicas da morfologia: a dilatação, erosão e a subtração e é definida da seguinte forma:

$$X = (A \oplus B) - (A \ominus B)$$

onde:

A é a imagem original e

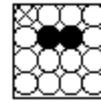
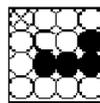
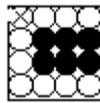
B o elemento estruturante

Uma importante aplicação do gradiente é para achar a fronteira.

Figura 3.11.

Deteção de bordas com gradiente morfológico binário

$$(A \oplus B) \quad (A \ominus B) \quad X = (A \oplus B) - (A \ominus B)$$



Abertura (Opening)

A abertura em geral suaviza o contorno de uma imagem, quebra estreitos e elimina proeminências delgadas, a operação de abertura é usada também para remover ruídos da imagem. A abertura de um conjunto A por elemento estruturante B é denotado $A \circ B$ e definida como:

$$A \circ B = (A \ominus B) \oplus B$$



A aplicação de uma erosão imediatamente seguida de uma dilatação usando o mesmo elemento estrutural é uma operação de abertura, ela tende a abrir pequenos vazios ou espaços entre objetos próximos numa imagem. Ou seja em outras palavras uma abertura e uma erosão seguida de uma dilatação usando um mesmo elemento estruturante, lembrando que a erosão acha todos os lugares onde o ajuste do elemento estruturante esta dentro a imagem, mas isto somente marca esta posição a origem de um elemento.

Porém seguinte uma erosão por uma dilatação, nos enchemos de brancos os lugares onde o ajuste do elemento estruturante estiver dentro do objeto. Na abertura podem se considerar para ser a união de todos copias traduzidos de um ajuste elemento estruturante que possam estar dentro do objeto.

$$A \circ B = (A \ominus B) \oplus B$$

Figura 3.12.

Exemplos de uma abertura. Observe-se que as “sujeiras” ao redor de A foram eliminadas.



Fechamento (Closing)

O fechamento funde pequenos quebras e alargas golfos estreitos elimina pequenos orifícios, se uma abertura cria pequenos vazios na imagem, um fechamento irá preencher ou fechar os vazios, estas operações podem remover muitos dos pixels brancos com ruídos, ou seja basicamente ele é igual a abertura só que primeiramente é feita a dilatação e após é feita a erosão assim ela se define como:

$$A \bullet B = (A \oplus B) \ominus B$$

Em outras palavras o fechamento trabalha de um modo oposto ao método abertura, onde ela remove todos os pixels onde o ajuste do elemento estruturante não esta dentro da imagem (primeiro pano), fechamento



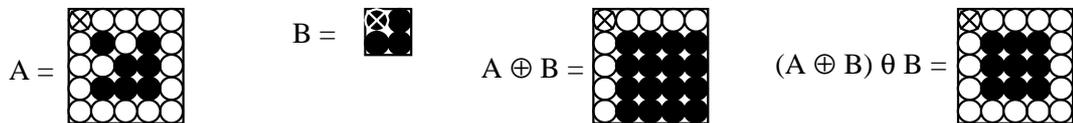
enche todos os lugares onde o elemento estruturante não iria ajustar na imagem (pano fundo).

Entretanto operações inversas, abertura e fechamento não irão estabelecer a imagem original.

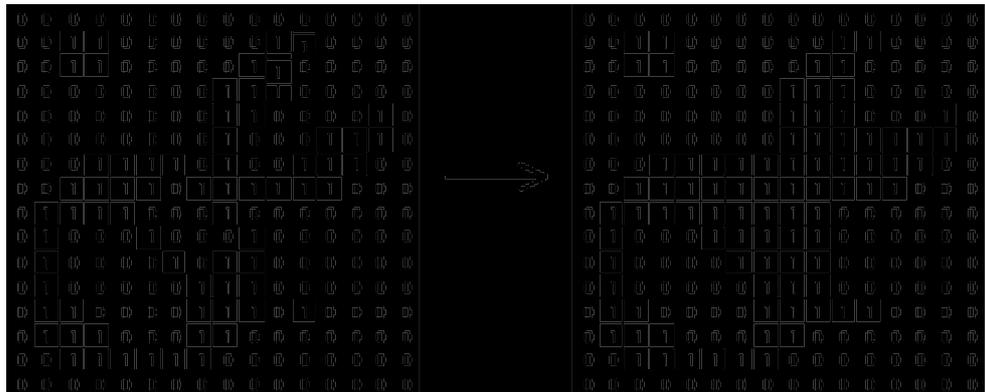
$$A \bullet B = (A \oplus B) \ominus B$$

Figura 3.13.

Dois exemplos de fechamento. Observe-se que o “furo” em diagonal em A no exemplo 1 foi eliminado.



Exemplo 2:



Propriedades da Aplicação Conjunta de Abertura e Fechamento

Dualidade entre Abertura e Fechamento

Abertura e dual a fechamento.

$$(A \circ B)' = A' \bullet B$$



Uma erosão pode ser implementada usando uma dilatação e vice versa, abertura pode ser implementada usando um fechamento e vice-versa.

Idempotência entre Abertura e Fechamento

Uma importante propriedade de operadores abertura e fechamento é que eles são idempotente, ou seja se você aplicar mas de uma vez, nada muda antes da primeira aplicação, ou seja:

$$A \circ B \circ B = A \circ B$$

$$A \bullet B \bullet B = A \bullet B$$

Acerto e Erro (Hit and Miss)

Nos podemos combinar erosão e dilatação para produzir um operador que tem uma ação como esta: acerto e erro . O operador leva dois elementos o primeiro e o acerto e o outro e o erro. O operador é definido como o segue, se o elemento estruturante J(acerto) e K(erro) são aplicados a imagem A:

$$A \otimes (J, K) = (A \ominus J) \cap (A' \ominus K)$$

$$A \otimes (J, K) = (A \ominus J) \cap (A' \oplus K)$$

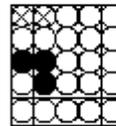
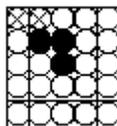
Em outras palavras o elemento estruturante J deveria ajustar dentro do objeto e o elemento K deveria ajustar fora do objeto nesta posição. Este forma de modelo binário e inicializado. Por exemplo o seguinte elemento estruturante é dado, para detectar os cantos:

Figura 3.14.

Dois elementos estruturantes para utilização com hit & miss para encontrar a borda superior direita de um objeto.

J – elemento estruturante

K – elemento estruturante



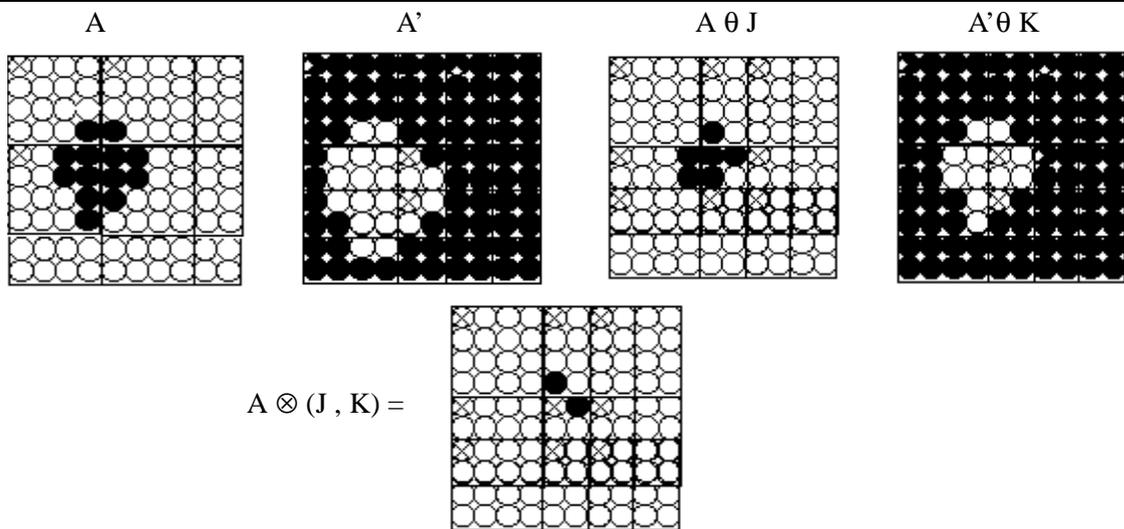


O elemento J acha os pontos com correção a esquerda e abaixo do vizinho, e o K acha os pontos superior a direita do vizinho. Esta forma talvez seja a mais usada para visualizar qual o elemento estruturante e designado. Aplica-se diferentes operadores de acerto e outro de erro. Exemplo:

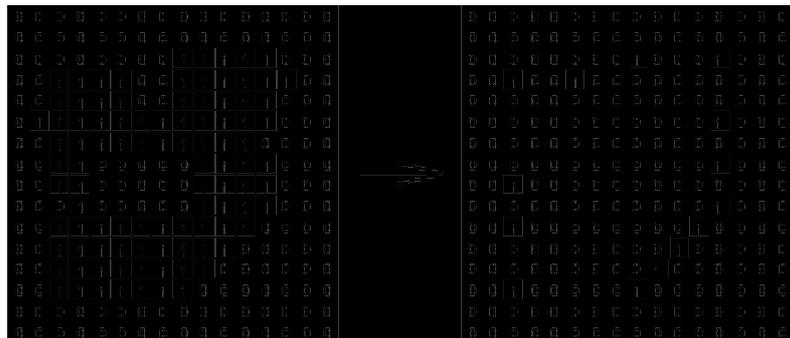
$$A \otimes (J, K) = (A \ominus J) \cap (A' \ominus K)$$

Figura 3.15.

Resultado da aplicação de hit & miss para determinar a borda superior de um objeto. O exemplo 2 mostra a detecção de cantos.



Exemplo 2:



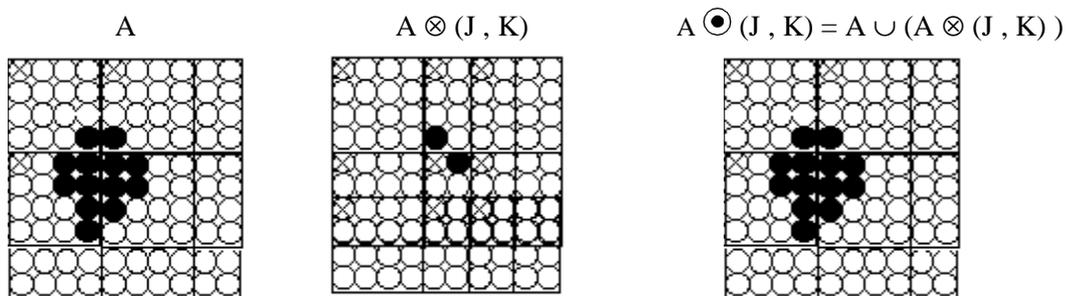


Engorda e Emagrecimento (thickening/thinning)

Nos podemos usar operador erro e acerto para achar pontos específicos, somando ou removendo para o objeto. Se nos achamos tal pixel e adicionarmos para o objeto então constituímos um operador de engorda (thickening):

$$A \odot (J, K) = A \cup (A \otimes (J, K))$$

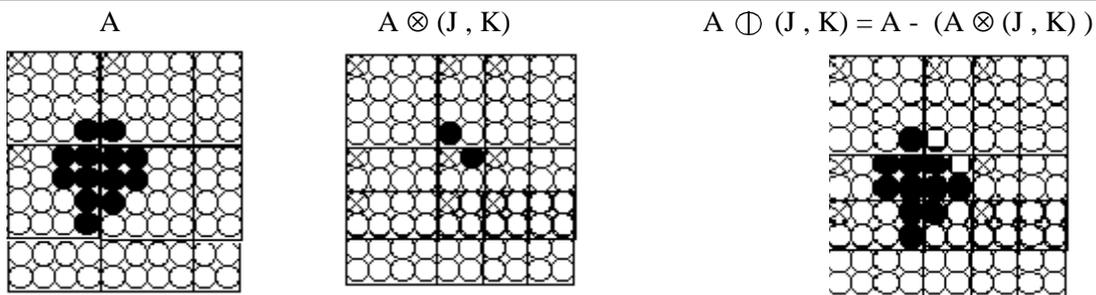
Figura 3.16. **Thickening**



Se removermos Similarmente nos achamos um operador de emagrecimento (thinning):

$$A \oplus (J, K) = A - (A \otimes (J, K))$$

Figura 3.17. **Thining**



Outra expressão para o afinamento ou emagrecimento de A é baseada em uma sequência de elementos estruturantes: $B = \{B^1, B^2, B^3, B^4, \dots B^N\}$



Em outras palavras, o processo consiste em afinar A por um passo com B^1 , então afinar o resultado com o passo B^2 , e assim sucessivamente, até que A seja o afinado com um passo de B^n . O processo todo é repetido até que não ocorram outras mudanças.

Esqueletonização

Uma abordagem importante para representar a forma estrutural de uma região plana e reduzi-la a um grafo. Esta redução pode ser implementada obtendo a esqueletonização da região através de um algoritmo. Existem vários algoritmos para essas transformações um deles é o MAT (Medial Axis Transformation) proposto por Blum e definida como "A MAT de uma regra R com fronteira B é obtida da seguinte forma: Para cada ponto p em R, encontra-se seu vizinho mais próximo em B. Se p tem mais de um vizinho a mesma distância mínima, diz-se que p pertence ao eixo médio de R."

Outra forma de obter a esqueletonização é através de técnicas morfológicas. Através de erosão e aberturas ou seja imagine a região A através desta técnica obtém $S(A)$ que é a esqueletonização de A. Isto é definida como:

$$S(A) = \cup S_K(A)$$

Com

$$S_K(A) = \cup \{(A \ominus kB) - [A \ominus kB] \circ B\}$$

Onde B é um elemento estruturante ($A \ominus kB$) indica k erosão sucessivas de A; ou seja:

$$(A \ominus kB) = ((\dots(A \ominus B) \ominus B)\dots) \ominus B$$

k vezes e K é o último passo iterativo antes de A resultar, por erosão em um conjunto vazio. Em outras palavras:

$$K = \max \{k \mid A \ominus kB \neq \emptyset\}.$$

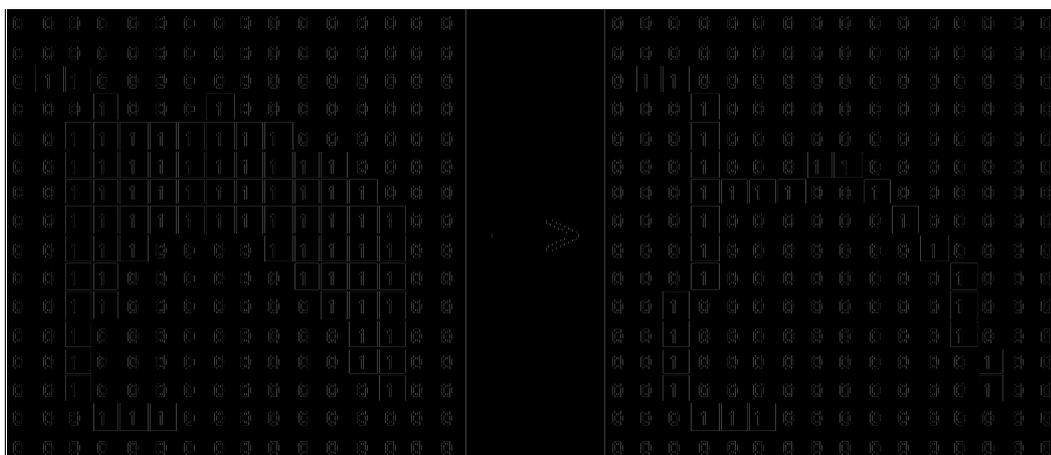
Essas equações indicam que $S(A)$ é o esqueleto de A, pode ser obtido pelo subconjuntos de esqueletonização de $S_K(A)$. Pode-se mostrar, além disso, que A pode ser reconstruído a partir destes subconjuntos utilizando a equação:



$$A = \cup (S_K(A) \oplus KB)$$

Onde $(S_K(A) \oplus KB)$ denota k dilatações sucessivas de $S_K(A)$.

Figura 3.18. Exemplo de esqueletonização através de morfologia binária



Algoritmos com buscas heurísticas muitas vezes produzem resultados melhores, pois os com formulações matemáticas removem a conectividade dos elementos da imagem original.

Uma esqueletonização de um objeto e um simples pixel, de um conjunto magro (mínimo) de pixel produzido por repetidamente emagrecimento de um objeto. Nos não podemos fazer isto usando somente erosão. Porque nos temos que ser capaz detectar quando a delação de um pixel iria fazer o objeto desconcertar. Nos teríamos para uso o erro/acerto.

Limiarização (Thresholding)

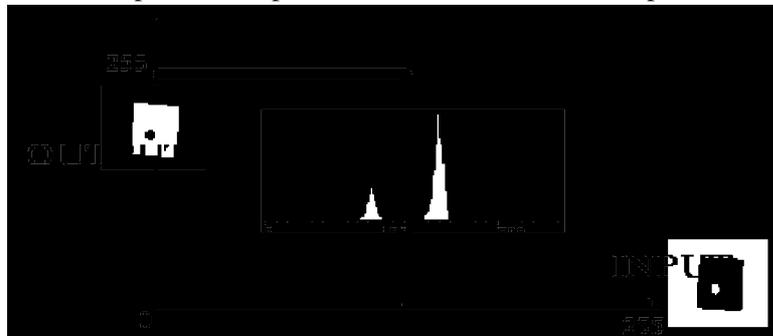
O princípio básico da limiarização consiste em separar as regiões de uma imagem quando esta apresenta duas classes (fundo e objeto). A forma mas simples da limiarização consiste na bipartição da imagem, convertendo os pixels cujo tom de cinza e maior ou igual a um curto valor (T) em brancos e os demais em pretos.



Esta operação pode ser descrita como uma técnica de processamento de imagem na qual uma imagem de entrada $f(x,y)$ de N níveis de cinza produz uma saída $g(x,y)$, chamado de imagem limiarizada, cujo numero de níveis de cinza e menor que N , normalmente $g(x,y)$ apresenta dois níveis de cinza.

$$g(x,y) = \begin{cases} 1 & \text{se } f(x,y) \geq T \\ 0 & \text{se } f(x,y) < T \end{cases}$$

onde os pixels rotulados com 1 correspondem aos objetos e os pixels com 0 correspondem ao pano de fundo, conforme exemplo:



Morfologia de Tons de Cinza

O uso de tons de cinza introduz uma enorme complicação tanto no conceito como na implementação. Um pixel pode agora ter qualquer valor inteiro, assim a facilidade de considerar a imagem como um conjunto desaparece.

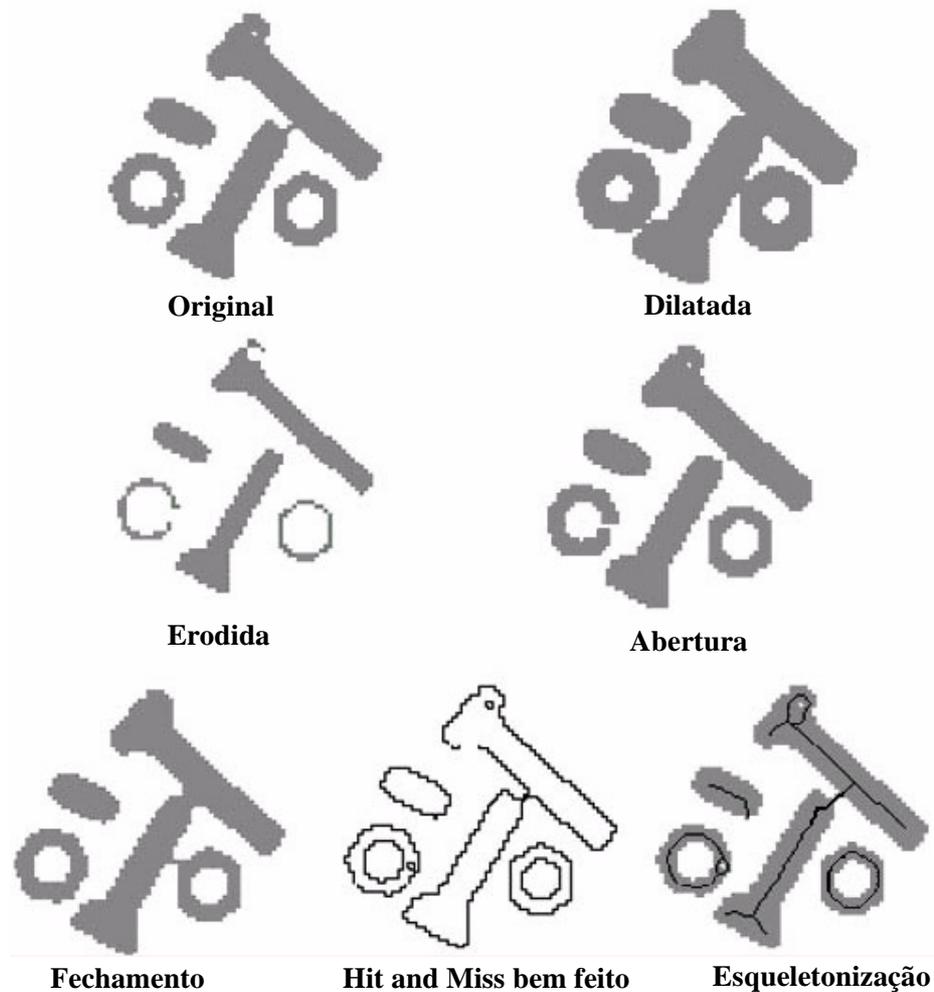
A idéia básica de Morfologia binaria estende-se para tom de cinza, mas operações lógicas simulam a conversão aritméticas: Uniões se tornam máximos e interseções se tornam mínimo etc.

Relembrando que imagens são funções do tipo espacial: $I(x,y)$.

A sombra (umbra) e uma função de um conjunto de todas as posições (x,v) tal que o valor e menor que ou igual a $f(x)$.



Figura 3.19. Visão geral das operações da morfologia matemática binária



$$\{(x,v' \mid v \leq f(x) \}$$

Esta construção permite que consideremos também escala de cinza, da mesma forma que usávamos para imagens binárias.

Nos podemos igualmente considerar uma dimensão como elemento estruturante g para ser um conjunto considerado sombra:



$$(x,v) | v \leq g(x).$$

Nos podemos estender estas idéias para imagem duas dimensões $I(x,v)$ por considerar o conjunto de posições (x,v) e possível intensidade I .

$$(x,v,y) | v \leq I(x,y).$$

Dilatação de Tons de Cinza

A dilatação em imagens de tons de cinza pode ser calculada como:

$$(A \oplus B) = \max \{A(i-x, j-y) + B(x, y) | (i-x, j-y) \in A, (x, y) \in B\}$$

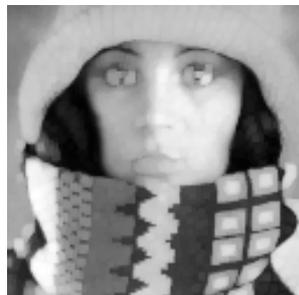
onde B é um elemento estrutural e A é a imagem de tons de cinza. Para calcular a dilatação faz-se o seguinte:

1. Posiciona-se a origem do elemento estrutural sobre o primeiro pixel da imagem a ser dilatada.
2. Calcula-se a soma de cada par correspondente de valores de pixels do elemento estrutural e da imagem.
3. Acha-se o valor máximo de todas essas somas, e armazena-se o pixel correspondente na imagem de saída para este valor.
4. Repete-se este processo para cada pixel da imagem a ser dilatada.

Os valores dos pixels do elemento estrutural também são tons de cinza e podem ser negativos. Uma vez que pixel de valores negativos não podem ser mostrados, existem dois possíveis modos de tratar com pixels negativos na imagem resultado:

Figura 3.20.

Resultado de uma dilatação em tons de cinza. Observe que as partes claras da imagem estão exacerbadas.



Morfologia de Tons de Cinza



Valores negativos podem ser alterados para zero (underflow).

A imagem inteira poderia ter seus valores aumentados para que o menor valor de pixel fosse zero mantendo os valores relativos entre os pixels.

Erosão de Tons de Cinza

A erosão em imagens de tons de cinza pode ser calculada como:

$$(A \ominus B) = \min \{A(i - x, j - y) - B(x, y) \mid (i - x, j - y) \in A, (x, y) \in B\}$$

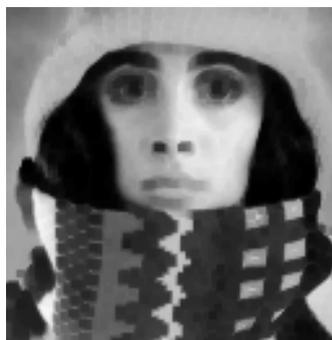
Onde B é um elemento estrutural e A é a imagem de tons de cinza. Para calcular a erosão faz-se o seguinte:

1. Posiciona-se a origem do elemento estrutural sobre o primeiro pixel da imagem que sofre erosão.
2. Calcula-se a diferença de cada par correspondente de valores de pixels do elemento estrutural e da imagem.
3. Acha-se o valor mínimo de todas essas diferenças, e armazena-se o pixel correspondente na imagem de saída para este valor.
4. Repete-se este processo para cada pixel da imagem que sofre erosão.

A mesma dualidade entre erosão e dilatação apresentada para imagens preto e branco vale para imagens tons de cinza. A Figura 3.21. mostra o resultado de uma erosão em tons de cinza, ressaltando as partes escuras da imagem, que ficam mais “gordas”.

Figura 3.21.

Resultado de uma erosão em tons de cinza. Observe que as partes escuras da imagem estão exacerbadas.





Abertura de Tons de Cinza

O processo de abertura funciona da mesma forma que definido anteriormente só que usa dilatação e erosão de tons de cinza. Isto é, a abertura usa uma erosão de tons de cinza seguida de uma dilatação de tons de cinza usando o mesmo elemento estrutural.

Suponha que nos primeiramente erodimos, então dilatamos uma função 1-d por algum elemento estruturante. Isto é uma abertura em tons de cinza.

$$f \circ_g (G) = (f \ominus_g G) \oplus_g G$$

ou para um constante elemento estruturante:

$$f \circ_g (G) = \max_{(a \in g)} \min_{(b \in g)} f(x-a+b)$$

considerar que aconteça a abertura de tons de cinza:

1. Refletir a posição do elemento estruturante na posição x.
2. pointwise adicionar o elemento estruturante em cima do neighborhood.
3. levar o máximo da qual resultado acima do neighborhood
4. não refletir a posição do elemento estruturante na posição x.
5. pointwise subtrair o elemento estruturante acima do neighborhood e
6. levar o mínimo deste resultado acima do neighborhood.

Fechamento de Tons de Cinza

O processo de fechamento funciona da mesma forma que definido anteriormente só que usa a dilatação e erosão de tons de cinza. Isto é, o fechamento usa uma dilatação de tons de cinza seguida de uma erosão de tons de cinza usando o mesmo elemento estrutural ou seja e uma dilatação seguida de uma erosão:

$$f \bullet_g (G) = (f \oplus_g G) \ominus_g G$$

ou para uma constante elemento estruturante:

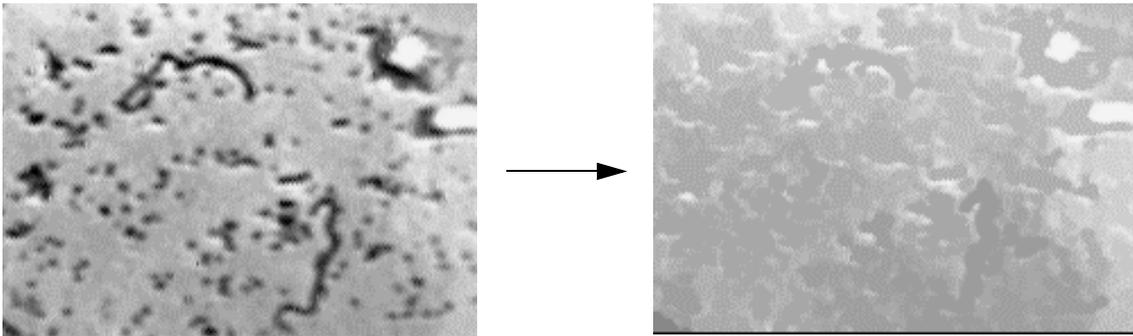
Morfologia de Tons de Cinza



$$f \bullet_g (G) = \min_{(a \in g)} \max_{(b \in g)} f(x+a+b)$$

Figura 3.22.

Exemplo de um fechamento em tom de cinza



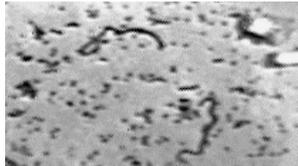
Morfologia em Imagens Coloridas

É basicamente a mesma coisa que em imagens em tons de cinza. Trabalhamos no espaço RGB ou outro espaço de cores ao invés de trabalharmos no espaço vetorial unidimensional dos tons de cinza. Uma outra opção simplificante, mas eficaz, é a de trabalharmos com a *luminância* do pixel colorido, ao invés de o representarmos no espaço tridimensional de sua cor. Se fizermos isto, podemos aplicar diretamente os algoritmos para tons de cinza sobre o valor da luminância do pixel como sendo seu valor para fins de morfologia.

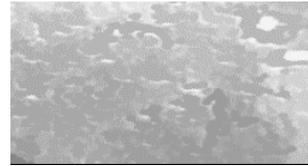
Na Figura 3.24. pode-se observar alguns exemplos de operações realizadas com uma imagem da Ponte Hercílio Luz em Florianópolis. Observe um fenômeno interessante: como o “objeto”, no caso a Ponte, é escuro e portanto responde por “ausência” de sinal para o algoritmo, neste exemplo ocorre exatamente o contrário do que esperamos. Na erosão a Ponte “aumenta” e na dilatação ela diminui. Na verdade na dilatação é o céu **claro** que aumenta, pois como o céu possui valor de pixel mais alto, ele é o “objeto” na imagem que de fato é dilatado.



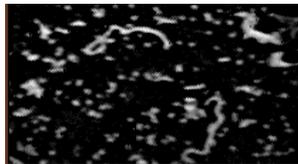
Figura 3.23. Exemplo de uma seqüência de operações realizada sobre uma imagem para extrair os dois sulcos maiores.



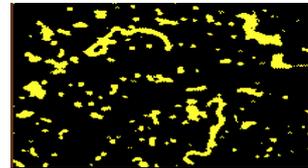
original



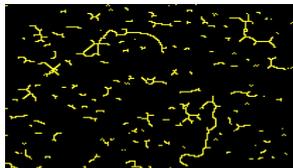
fechamento



subtract



limiarização simples



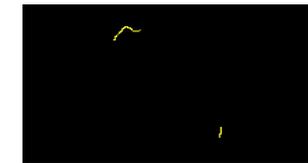
esqueletonização 1



thinning



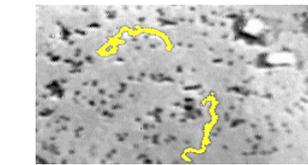
esqueletonização 2



subtração de imagens



abertura



projeção do resultado sobre o original

Morfologia de Tons de Cinza



Figura 3.24. Exemplo de morfologia sobre imagens a cores. Nesta foto onde o objeto é uma silueta escura, ocorre o contrário do que esperávamos.

Original



Erodida



Dilatada



Ferramentas de Software para Morfologia Matemática

Implementações de algoritmos de morfologia matemática são bastante difundidas, inclusive pela sua simplicidade de implementação.

Até a maioria dos programas de “desenho” possui alguma rotina de morfologia, nem sempre das melhores, mas muitas vezes suficiente para um processamento manual de uma imagem. Geralmente estas implementações em programas comerciais para uso doméstico não possuem a possibilidade de que se edite o elemento estruturante, vindo com operações do tipo “dilatar” ou “erodir” prontas, com elementos estruturantes pré-definidos. Há pouco o autor conseguiu constatar até em um software para operação de um tomógrafo computadorizado um ponto de menu indicando a possibilidade de se submeter as imagens do paciente a operações morfológicas.

Uma das melhores implementações de morfologia, tanto para operações binárias como para tons de cinza é a que se encontra no pacote Khoros [3.5][3.7]. Este pacote está descrito no capítulo de ferramentas deste trabalho.

Referências

- [3 . 1] **Gonzalez, Rafael C. & Woods, Richard E.** *Digital Image Processing*. Reading, Addison-Wesley ,1992.
- [3 . 2] **Parker, J. R.** *Algorithms for Image Processing and Computer Vision*. New York, John wiley & Sons, 1997.
- [3 . 3] **Russ, John C. .** *The image processing handbook*. 2 ed., Boca Raton, CRC Press, 1995.
- [3 . 4] **Marques Filho, Ogê & Vieira Neto, Hugo.** *Processamento digital de Imagens*. Rio de Janeiro, Brasport, 1999.
- [3 . 5] www.ime.usp.br/mac/khoros/mmach.old/

Referências



- [3 . 6] www.ph.tn.tudelft.nl/Courses/
- [3 . 7] www.khoral.com/
- [3 . 8] www.hig.no/bibliotec/

CAPÍTULO 3. Morfología Matemática

