

Uma nova abordagem para as operações de Transformada da Erosão e Transformada da Dilatação através do uso da união da decomposição de elementos estruturantes

Alexandre Y. Harano

Instituto de Matemática e Estatística da Universidade de São Paulo (IME-USP)
ayharano@ime.usp.br

11 de março de 2013



1 Problema

- Arcabouço
- Motivação
- Definição do Problema

2 Método

- Desenvolvimento
- Estrutura de Dados
- Algoritmo
- Corretismo

3 Avaliação

- Método de Avaliação
- Análise computacional
- Resultados

1 Problema

■ Arcabouço

■ Motivação

■ Definição do Problema

2 Método

■ Desenvolvimento

■ Estrutura de Dados

■ Algoritmo

■ Corretismo

3 Avaliação

■ Método de Avaliação

■ Análise computacional

■ Resultados

Morfologia Matemática (MM)

- Fundamentado por G. Matheron e J. Serra em 1964.
- Aplicada a imagens binárias: $\mathbb{Z}^d \rightarrow \mathbb{Z}^d$.
- Abordagem discreta em imagens binárias através de operações em conjuntos, e.g., união, interseção, diferença, complemento.
- Operadores básicos:

$$\text{Erosão } \varepsilon_W(G) = \{h \in \mathbb{Z}^d : (h + W) \subseteq G\}$$

$$\text{Dilatação } \delta_W(G) = \{j \in \mathbb{Z}^d : (j + W^r) \cap G \neq \emptyset\}$$

- $G \subset \mathbb{Z}^d$ é uma imagem binária.
- $W \subset \mathbb{Z}^d$ é um elemento estruturante (EE).

Morfologia Matemática (MM)

- Fundamentado por G. Matheron e J. Serra em 1964.
- Aplicada a imagens binárias: $\mathbb{Z}^d \rightarrow \mathbb{Z}^d$.
- Abordagem discreta em imagens binárias através de operações em conjuntos, e.g., união, interseção, diferença, complemento.
- Operadores básicos:

$$\text{Erosão } \varepsilon_W(G) = \{h \in \mathbb{Z}^d : (h + W) \subseteq G\}$$

$$\text{Dilatação } \delta_W(G) = \{j \in \mathbb{Z}^d : (j + W^c) \cap G \neq \emptyset\}$$

- $G \subset \mathbb{Z}^d$ é uma imagem binária.
- $W \subset \mathbb{Z}^d$ é um elemento estruturante (EE).

Morfologia Matemática (MM)

- Fundamentado por G. Matheron e J. Serra em 1964.
- Aplicada a imagens binárias: $\mathbb{Z}^d \rightarrow \mathbb{Z}^d$.
- Abordagem discreta em imagens binárias através de operações em conjuntos, e.g., união, interseção, diferença, complemento.
- Operadores básicos:

$$\text{Erosão } \varepsilon_W(G) = \{h \in \mathbb{Z}^d : (h + W) \subseteq G\}$$

$$\text{Dilatação } \delta_W(G) = \{j \in \mathbb{Z}^d : (j + W) \cap G \neq \emptyset\}$$

- $G \subset \mathbb{Z}^d$ é uma imagem binária.
- $W \subset \mathbb{Z}^d$ é um elemento estruturante (EE).

Morfologia Matemática (MM)

- Fundamentado por G. Matheron e J. Serra em 1964.
- Aplicada a imagens binárias: $\mathbb{Z}^d \rightarrow \mathbb{Z}^d$.
- Abordagem discreta em imagens binárias através de operações em conjuntos, e.g., união, interseção, diferença, complemento.
- Operadores básicos:

Erosão $\varepsilon_W(G) = \{h \in \mathbb{Z}^d : (h + W) \subseteq G\}$

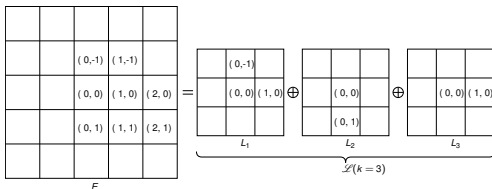
Dilatação $\delta_W(G) = \{j \in \mathbb{Z}^d : (j + W^r) \cap G \neq \emptyset\}$

- $G \subset \mathbb{Z}^d$ é uma imagem binária.
- $W \subset \mathbb{Z}^d$ é um elemento estruturante (EE).

Tópicos abordados

Decomposição de EEs

Diversos algoritmos da MM utilizam EEs de tamanho restrito, e.g., 3×3 .



Tópicos abordados

Algoritmos de Erosão e Dilatação

- Usualmente definidos para \mathbb{Z}^2 .
- Muitas propostas otimizam para EEs específicos, e.g., linha, cruz.
- Um dos principais *gargalos*: determinação das coordenadas a serem analisadas.
- **Proposta**: como minimizar as comparações para tal determinação?

Tópicos abordados

Algoritmos de Erosão e Dilatação

- Usualmente definidos para \mathbb{Z}^2 .
- Muitas propostas otimizam para EEs específicos, e.g., linha, cruz.
- Um dos principais *gargalos*: determinação das coordenadas a serem analisadas.
- **Proposta**: como minimizar as comparações para tal determinação?

Tópicos abordados

Algoritmos de Erosão e Dilatação

- Usualmente definidos para \mathbb{Z}^2 .
- Muitas propostas otimizam para EEs específicos, e.g., linha, cruz.
- Um dos principais *gargalos*: determinação das coordenadas a serem analisadas.
- **Proposta**: como minimizar as comparações para tal determinação?

Tópicos abordados

Algoritmos de Erosão e Dilatação

- Usualmente definidos para \mathbb{Z}^2 .
- Muitas propostas otimizam para EEs específicos, e.g., linha, cruz.
- Um dos principais *gargalos*: determinação das coordenadas a serem analisadas.
- **Proposta**: como minimizar as comparações para tal determinação?

Tópicos abordados

Algoritmos de Erosão e Dilatação

- Usualmente definidos para \mathbb{Z}^2 .
- Muitas propostas otimizam para EEs específicos, e.g., linha, cruz.
- Um dos principais *gargalos*: determinação das coordenadas a serem analisadas.
- **Proposta**: como minimizar as comparações para tal determinação?

1 Problema

- Arcabouço
- Motivação
- Definição do Problema

2 Método

- Desenvolvimento
- Estrutura de Dados
- Algoritmo
- Corretismo

3 Avaliação

- Método de Avaliação
- Análise computacional
- Resultados

Problema original

Sejam:

- uma imagem binária $X \subset \mathbb{Z}^d$, e
- um elemento estruturante $E \subset \mathbb{Z}^d$.

Efetue as Transformadas da Erosão e Dilatação de X e E :

$$\text{TRANSFORMADAEROSÃO}(X, E)[c] = \begin{cases} \max\{w \mid c \in \varepsilon_{wE}(X)\} & : c \in X \\ -1 & : c \notin X \end{cases}$$

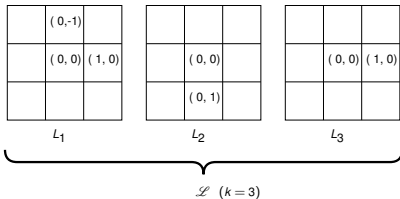
$$\text{TRANSFORMADADILATAÇÃO}(X, E)[c] = \begin{cases} \min\{w \mid c \in \delta_{wE}(X)\} \\ \quad : \text{se } \exists w, c \in \delta_{wE}(X) \\ -1 : \forall w, c \notin \delta_{wE}(X) \end{cases}$$

Definições das Transformadas adaptadas de (Chen e Haralick, 1995).

Exemplo de Transformada da Erosão

	(1, 0)	(2, 0)		
	(1, 1)	(2, 1)	(3, 1)	
(0, 2)	(1, 2)	(2, 2)	(3, 2)	
(0, 3)	(1, 3)	(2, 3)	(3, 3)	(4, 3)
(0, 4)	(1, 4)	(2, 4)	(3, 4)	(4, 4)

X



-1	1	1	-1	-1
-1	2	1	1	-1
1	2	1	1	-1
2	2	2	1	1
1	1	1	1	1

TRANSFORMADA EROSAO(X, L, k)

Problema atacado

Sejam:

- uma imagem binária $X \subset \mathbb{Z}^d$, e
- uma lista de $k + 1$ EEs $\mathcal{L} = [L_0, L_1, \dots, L_k]$, $\{0\} \subseteq L_i \subset \mathbb{Z}^d$, $\forall i \geq 0$.

Efetue as Transformadas da Erosão e Dilatação de X e \mathcal{L} :

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} \max\{w \mid c \in \varepsilon_{wL_k}(\dots(\varepsilon_{wL_2}(\varepsilon_{wL_1}(X)))\dots)\} \\ \quad : c \in X \\ -1 : c \notin X \end{cases}$$

$$\text{TRANSFORMADADILATAÇÃO}(X, \mathcal{L}, k)[c] = \begin{cases} \min\{w \mid c \in \delta_{wL_k}(\dots(\delta_{wL_2}(\delta_{wL_1}(X)))\dots)\} \\ \quad : \text{se } \exists w, c \in \delta_{wL_k}(\dots(\delta_{wL_2}(\delta_{wL_1}(X)))\dots) \\ -1 : \forall w, c \notin \delta_{wL_k}(\dots(\delta_{wL_2}(\delta_{wL_1}(X)))\dots) \end{cases}$$

- 1 Problema
 - Arcabouço
 - Motivação
 - Definição do Problema
- 2 Método
 - **Desenvolvimento**
 - Estrutura de Dados
 - Algoritmo
 - Corretismo
- 3 Avaliação
 - Método de Avaliação
 - Análise computacional
 - Resultados

Algoritmos Prévios

- Para o desenvolvimento do algoritmo proposto, baseamos nas implementações triviais (TRANSFORMADAEROSÃOSIMPLES e TRANSFORMADADILATAÇÃOSIMPLES).
- Propomos quatro refinamentos incrementais:

TRANSFORMADADUAL

Um único algoritmo dual para atender às duas operações.

TRANSFORMADADUALLINEAR

A linearização dos dados utilizados.

TRANSFORMADADUALLINEARDEBORDA

A utilização da borda de acordo com a união dos EEs de \mathcal{L} .

TRANSFORMADAPROPOSTA

O relacionamento entre as coordenadas de interesse e as coordenadas da união dos EEs.



Algoritmos Prévios

- Para o desenvolvimento do algoritmo proposto, baseamos nas implementações triviais (TRANSFORMADAEROSÃOSIMPLES e TRANSFORMADADILATAÇÃO SIMPLES).
- Propomos quatro refinamentos incrementais:

TRANSFORMADADUAL

Um único algoritmo dual para atender às duas operações.

TRANSFORMADADUALLINEAR

A linearização dos dados utilizados.

TRANSFORMADADUALLINEARDEBORDA

A utilização da borda de acordo com a união dos EEs de \mathcal{L} .

TRANSFORMADAPROPOSTA

O relacionamento entre as coordenadas de interesse e as coordenadas da união dos EEs.



Algoritmos Prévios

- Para o desenvolvimento do algoritmo proposto, baseamos nas implementações triviais (TRANSFORMADAEROSÃOSIMPLES e TRANSFORMADADILATAÇÃOSIMPLES).
- Propomos quatro refinamentos incrementais:

TRANSFORMADADUAL

Um único algoritmo dual para atender às duas operações.

TRANSFORMADADUALLINEAR

A linearização dos dados utilizados.

TRANSFORMADADUALLINEARDEBORDA

A utilização da borda de acordo com a união dos EEs de \mathcal{L} .

TRANSFORMADAPROPOSTA

O relacionamento entre as coordenadas de interesse e as coordenadas da união dos EEs.



Algoritmos Prévios

- Para o desenvolvimento do algoritmo proposto, baseamos nas implementações triviais (TRANSFORMADAEROSÃOSIMPLES e TRANSFORMADADILATAÇÃOSIMPLES).
- Propomos quatro refinamentos incrementais:

TRANSFORMADADUAL

Um único algoritmo dual para atender às duas operações.

TRANSFORMADADUALLINEAR

A linearização dos dados utilizados.

TRANSFORMADADUALLINEARDEBORDA

A utilização da borda de acordo com a união dos EEs de \mathcal{L} .

TRANSFORMADAPROPOSTA

O relacionamento entre as coordenadas de interesse e as coordenadas da união dos EEs.



Algoritmos Prévios

- Para o desenvolvimento do algoritmo proposto, baseamos nas implementações triviais (TRANSFORMADAEROSÃOSIMPLES e TRANSFORMADADILATAÇÃO SIMPLES).
- Propomos quatro refinamentos incrementais:

TRANSFORMADADUAL

Um único algoritmo dual para atender às duas operações.

TRANSFORMADADUALLINEAR

A linearização dos dados utilizados.

TRANSFORMADADUALLINEARDEBORDA

A utilização da borda de acordo com a união dos EEs de \mathcal{L} .

TRANSFORMADAPROPOSTA

O relacionamento entre as coordenadas de interesse e as coordenadas da união dos EEs.



Algoritmos Prévios

- Para o desenvolvimento do algoritmo proposto, baseamos nas implementações triviais (TRANSFORMADAEROSÃOSIMPLES e TRANSFORMADADILATAÇÃO SIMPLES).
- Propomos quatro refinamentos incrementais:

TRANSFORMADADUAL

Um único algoritmo dual para atender às duas operações.

TRANSFORMADADUALLINEAR

A linearização dos dados utilizados.

TRANSFORMADADUALLINEARDEBORDA

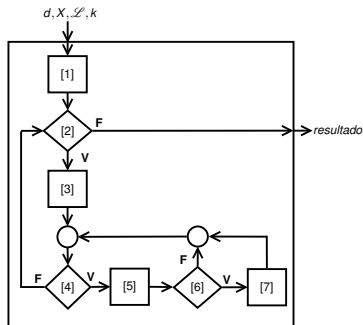
A utilização da borda de acordo com a união dos EEs de \mathcal{L} .

TRANSFORMADAPROPOSTA

O relacionamento entre as coordenadas de interesse e as coordenadas da união dos EEs.



Algoritmos Prévios



TRANSFORMADA EROÇÃO SIMPLES e TRANSFORMADA DILATAÇÃO SIMPLES

- Algoritmos mais triviais para as operações desejadas.
- Componente de maior tempo computacional: determinação dos elementos a serem analisados.
- Não requer espaço computacional adicional.

TRANSFORMADA EROÇÃO SIMPLES e TRANSFORMADA DILATAÇÃO SIMPLES

- Algoritmos mais triviais para as operações desejadas.
- Componente de maior tempo computacional: determinação dos elementos a serem analisados.
- Não requer espaço computacional adicional.

TRANSFORMADA EROSÃO SIMPLES e TRANSFORMADA DILATAÇÃO SIMPLES

- Algoritmos mais triviais para as operações desejadas.
- Componente de maior tempo computacional: determinação dos elementos a serem analisados.
- Não requer espaço computacional adicional.

TRANSFORMADADUAL

- Unifica a execução das duas transformadas de acordo com o parâmetro $ehTransformadaErosao$ e utiliza a dualidade através da relação $\delta_W(G) = (\epsilon_{Wr}(G^c))^c$.
- Introduz o conceito de *coordenadas de interesse*.
- Componente de maior tempo computacional: determinação dos elementos a serem analisados.
- Não requer espaço computacional adicional.

TRANSFORMADA DUAL

- Unifica a execução das duas transformadas de acordo com o parâmetro $ehTransformadaErosao$ e utiliza a dualidade através da relação $\delta_W(G) = (\epsilon_{Wr}(G^c))^c$.
- Introduce o conceito de *coordenadas de interesse*.
- Componente de maior tempo computacional: determinação dos elementos a serem analisados.
- Não requer espaço computacional adicional.

TRANSFORMADA DUAL

- Unifica a execução das duas transformadas de acordo com o parâmetro $ehTransformadaErosao$ e utiliza a dualidade através da relação $\delta_W(G) = (\epsilon_{Wr}(G^c))^c$.
- Introduce o conceito de *coordenadas de interesse*.
- Componente de maior tempo computacional: determinação dos elementos a serem analisados.
- Não requer espaço computacional adicional.

TRANSFORMADA DUAL

- Unifica a execução das duas transformadas de acordo com o parâmetro $ehTransformadaErosao$ e utiliza a dualidade através da relação $\delta_W(G) = (\varepsilon_{Wr}(G^c))^c$.
- Introduce o conceito de *coordenadas de interesse*.
- Componente de maior tempo computacional: determinação dos elementos a serem analisados.
- Não requer espaço computacional adicional.

TRANSFORMADA DUAL LINEAR

- Armazena em vetores os dados que não sofrerão alterações ao longo da execução e em listas, caso contrário.
- Não é mais necessário verificar todas as coordenadas da caixa delimitadora da imagem para cada valor de x .
- Componente de maior tempo computacional: linearização dos dados e verificação de cada coordenada de interesse contra cada elemento da união dos EEs de \mathcal{L} .

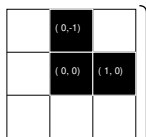
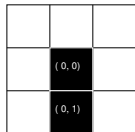
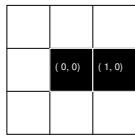
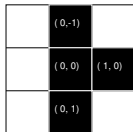
TRANSFORMADA DUAL LINEAR

- Armazena em vetores os dados que não sofrerão alterações ao longo da execução e em listas, caso contrário.
- Não é mais necessário verificar todas as coordenadas da caixa delimitadora da imagem para cada valor de x .
- Componente de maior tempo computacional: linearização dos dados e verificação de cada coordenada de interesse contra cada elemento da união dos EEs de \mathcal{L} .

TRANSFORMADA DUAL LINEAR

- Armazena em vetores os dados que não sofrerão alterações ao longo da execução e em listas, caso contrário.
- Não é mais necessário verificar todas as coordenadas da caixa delimitadora da imagem para cada valor de x .
- Componente de maior tempo computacional: linearização dos dados e verificação de cada coordenada de interesse contra cada elemento da união dos EEs de \mathcal{L} .

Exemplo de inicialização de dados de \mathcal{L}

 L_1  L_2  L_3 (a) Elementos estruturantes de \mathcal{L} .

$$unio = \bigcup_{i=1}^k L_i$$

(b) União dos elementos estruturantes de \mathcal{L} .

cardinalidadeUnio

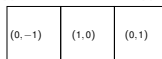
3

(c) Cardinalidade da união dos elementos estruturantes de \mathcal{L} .

elementosUnio[1]

elementosUnio[3]

elementosUnio



elementosUnio[2]

(d) Elementos da união dos elementos estruturantes de \mathcal{L} .

cardinalidadeL

cardinalidadeL[1]

2

cardinalidadeL[2]

1

cardinalidadeL[3]

1

(e) Vetor com a cardinalidade dos elementos estruturantes de \mathcal{L} .

elementosL

elementosL[1]

1 2

elementosL[2]

3

elementosL[3]

2

TRANSFORMADA DUAL LINEAR DE BORDA

- Ao invés de armazenar todas as coordenadas de interesse, armazena somente os elementos da borda da imagem a ser determinada, de acordo com a união dos EEs de \mathcal{L} .
- Componente de maior tempo computacional: linearização dos dados e manutenção da invariante proposta.

TRANSFORMADA DUAL LINEAR DE BORDA

- Ao invés de armazenar todas as coordenadas de interesse, armazena somente os elementos da borda da imagem a ser determinada, de acordo com a união dos EEs de \mathcal{L} .
- Componente de maior tempo computacional: linearização dos dados e manutenção da invariante proposta.

- 1 Problema
 - Arcabouço
 - Motivação
 - Definição do Problema

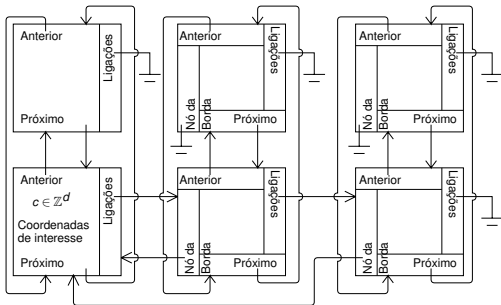
- 2 Método
 - Desenvolvimento
 - **Estrutura de Dados**
 - Algoritmo
 - Corretismo

- 3 Avaliação
 - Método de Avaliação
 - Análise computacional
 - Resultados

Estrutura de Dados

Nós da Borda Dado, armazenado em uma matriz de mesmo tamanho da imagem de entrada, que enfileira nós NL de uma posição candidata a ser processada.

Nós de Ligação Relaciona uma posição a ser processada ao motivo de seu processamento, i.e. um elemento da união dos EEs de \mathcal{L} .

(a) *listaNsBorda.*(b) *vetorElementosUnio[3].*(c) *vetorElementosUnio[2].*

(d) Nó da borda.

(e) Nó de ligação.

(f) Outro nó de ligação.

- 1 Problema
 - Arcabouço
 - Motivação
 - Definição do Problema
- 2 Método
 - Desenvolvimento
 - Estrutura de Dados
 - **Algoritmo**
 - Corretismo
- 3 Avaliação
 - Método de Avaliação
 - Análise computacional
 - Resultados

Visão Geral do Algoritmo

- $U = \bigcup_i L_i$.
- Usa a restrição $\{o\} \subseteq L_i, \forall i \geq 0$, de modo a armazenar as informações estritamente necessárias da borda para a próxima iteração.
- Matriz esparsa bidimensional retém as informações relativas ao gradiente morfológico interno ou externo da imagem de trabalho por U .
- O algoritmo é dividido em: (i) inicialização das estruturas de dados, e (ii) operação escolhida propriamente dita.

Visão Geral do Algoritmo

LINEARIZEELEMENTOESTRUTURANTE

Extrai informações dos EE de \mathcal{L} através da união de seus elementos e armazena em vetores.

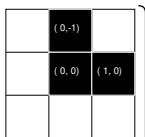
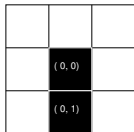
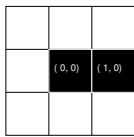
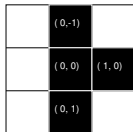
INICIALIZEMATRIZESPARGA

Popula as estruturas de dados relacionadas à matriz esparsa (listas e matriz baseada em X).

ALGORITMOPROPOSTO

Efetua a Transformada solicitada através do consumo das listas de nós de ligação e da manutenção da matriz esparsa.

Visão Geral do Algoritmo

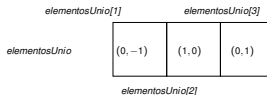
 L_1  L_2  L_3 (a) Elementos estruturantes de \mathcal{L} .

$$unio = \bigcup_{i=1}^k L_i$$

(b) União dos elementos estruturantes de \mathcal{L} .

cardinalidadeUnio

3

(c) Cardinalidade da união dos elementos estruturantes de \mathcal{L} .(d) Elementos da união dos elementos estruturantes de \mathcal{L} .

cardinalidadeL

cardinalidadeL[1]

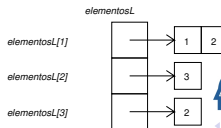
2

cardinalidadeL[2]

1

cardinalidadeL[3]

1

(e) Vetor com a cardinalidade dos elementos estruturantes de \mathcal{L} .

Visão Geral do Algoritmo

LINEARIZEELEMENTOESTRUTURANTE

Extrai informações dos EE de \mathcal{L} através da união de seus elementos e armazena em vetores.

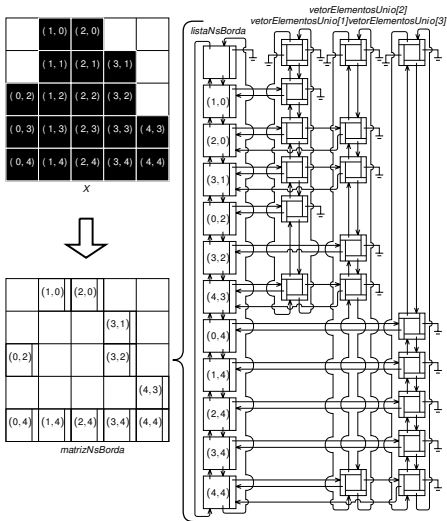
INICIALIZEMATRIZESPARGA

Popula as estruturas de dados relacionadas à matriz esparsa (listas e matriz baseada em X).

ALGORITMOPROPOSTO

Efetua a Transformada solicitada através do consumo das listas de nós de ligação e da manutenção da matriz esparsa.

Visão Geral do Algoritmo



Visão Geral do Algoritmo

LINEARIZEELEMENTOESTRUTURANTE

Extrai informações dos EE de \mathcal{L} através da união de seus elementos e armazena em vetores.

INICIALIZEMATRIZESPARGA

Popula as estruturas de dados relacionadas à matriz esparsa (listas e matriz baseada em X).

ALGORITMOPROPOSTO

Efetua a Transformada solicitada através do consumo das listas de nós de ligação e da manutenção da matriz esparsa.



ALGORITMO PROPOSTO

No início da combinação x de *iteracao* e *indiceElementoEstruturante*, valem as seguintes invariantes:

- 1 *imagemTrabalho* contém a imagem binária I_{x-1} ;
- 2 *vetorListasUnicao* contém as listas duplamente encadeada com os nós de ligação separado por elemento da união dos EEs de \mathcal{L} .
- 3 *listaNosBorda* e *matrizNosBorda* possui somente os elementos de

$$I_x \setminus \varepsilon_U(I_x) .$$

- 4 *imagemCandidato* marca os elementos que já foram inseridos em *listaNosBorda*.
- 5 *resultado* armazena os valores corretos para a Transformada até a combinação x .

ALGORITMO PROPOSTO

Inicialização Trivialmente verificada, através das estruturas inicializadas por INICIALIZEMATRIZESPARSA e a inicialização da estrutura que armazenará o resultado da Transformada desejada.

ALGORITMO PROPOSTO

Manutenção Consome as listas para compor a lista da borda como $I_{x-1} \setminus \varepsilon_{L_x}(I_{x-1})$, removendo os nós de ligação e nós de borda relacionados. ATUALIZAÇÃO ALGORITMO PROPOSTO efetua a inserção de novos nós de ligação e nós de borda, completando a etapa de manutenção.

ALGORITMO PROPOSTO

Término Trivialmente verificada, uma vez que *listaNosBorda* esteja vazia.

- 1 Problema
 - Arcabouço
 - Motivação
 - Definição do Problema
- 2 Método
 - Desenvolvimento
 - Estrutura de Dados
 - Algoritmo
 - **Corretismo**
- 3 Avaliação
 - Método de Avaliação
 - Análise computacional
 - Resultados

Prova do Corretismo de TRANSFORMADA PROPOSTA

Em linhas gerais, é verificada em três partes:

$$1 \quad \underbrace{\bigcup_{\Delta \in P} (A \setminus \varepsilon_{\{\Delta, 0\}}(A))}_R = \underbrace{A \setminus \varepsilon_P(A)}_S .$$

$$\quad \quad \quad \blacksquare \quad \varepsilon_P(A) = A \setminus (A \setminus \varepsilon_P(A)) .$$

- 2 Através da garantia das etapas de inicialização, manutenção e término das cinco invariantes adotadas ao longo do algoritmo (previamente apresentadas).
- 3 Mapeamento das propriedades de dualidade da Transformada da Dilatação nas propriedades obtidas para a Transformada da Erosão.

Evolução de resultado

Conforme apresentado no Teorema 5.11 (e no Teorema 5.14):

■ Inicialização

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} 0 & : c \in X \\ -1 & : c \notin X \end{cases} .$$

■ Manutenção

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} \max\{w \mid c \in \varepsilon_{wL_k}(\dots(\varepsilon_{wL_2}(\varepsilon_{wL_1}(X)))\dots)\} \\ \quad : c \in \bigcup_{y=1}^{x-1} B_y \\ 0 \\ \quad : c \in X \setminus \bigcup_{y=1}^{x-1} B_y \\ -1 : c \notin X \end{cases}$$

■ Término

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} \max\{w \mid c \in \varepsilon_{wL_k}(\dots(\varepsilon_{wL_2}(\varepsilon_{wL_1}(X)))\dots)\} \\ \quad : c \in X \\ -1 : c \notin X \end{cases}$$

Evolução de resultado

Conforme apresentado no Teorema 5.11 (e no Teorema 5.14):

■ Inicialização

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} 0 & : c \in X \\ -1 & : c \notin X \end{cases} .$$

■ Manutenção

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} \max\{w \mid c \in \varepsilon_{wL_k}(\dots(\varepsilon_{wL_2}(\varepsilon_{wL_1}(X)))\dots)\} \\ \quad : c \in \bigcup_{y=1}^{x-1} B_y \\ 0 \\ \quad : c \in X \setminus \bigcup_{y=1}^{x-1} B_y \\ -1 : c \notin X \end{cases}$$

■ Término

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} \max\{w \mid c \in \varepsilon_{wL_k}(\dots(\varepsilon_{wL_2}(\varepsilon_{wL_1}(X)))\dots)\} \\ \quad : c \in X \\ -1 : c \notin X \end{cases}$$

Evolução de resultado

Conforme apresentado no Teorema 5.11 (e no Teorema 5.14):

■ Inicialização

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} 0 & : c \in X \\ -1 & : c \notin X \end{cases}$$

■ Manutenção

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} \max\{w \mid c \in \varepsilon_{wL_k}(\dots(\varepsilon_{wL_2}(\varepsilon_{wL_1}(X)))\dots)\} \\ \quad : c \in \bigcup_{y=1}^{x-1} B_y \\ 0 \\ \quad : c \in X \setminus \bigcup_{y=1}^{x-1} B_y \\ -1 : c \notin X \end{cases}$$

■ Término

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} \max\{w \mid c \in \varepsilon_{wL_k}(\dots(\varepsilon_{wL_2}(\varepsilon_{wL_1}(X)))\dots)\} \\ \quad : c \in X \\ -1 : c \notin X \end{cases}$$

Evolução de resultado

Conforme apresentado no Teorema 5.11 (e no Teorema 5.14):

■ Inicialização

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} 0 & : c \in X \\ -1 & : c \notin X \end{cases} .$$

■ Manutenção

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} \max\{w \mid c \in \varepsilon_{wL_k}(\dots(\varepsilon_{wL_2}(\varepsilon_{wL_1}(X)))\dots)\} \\ \quad : c \in \bigcup_{y=1}^{x-1} B_y \\ 0 \\ \quad : c \in X \setminus \bigcup_{y=1}^{x-1} B_y \\ -1 : c \notin X \end{cases}$$

■ Término

$$\text{TRANSFORMADAEROSÃO}(X, \mathcal{L}, k)[c] = \begin{cases} \max\{w \mid c \in \varepsilon_{wL_k}(\dots(\varepsilon_{wL_2}(\varepsilon_{wL_1}(X)))\dots)\} \\ \quad : c \in X \\ -1 : c \notin X \end{cases}$$

- 1 Problema
 - Arcabouço
 - Motivação
 - Definição do Problema

- 2 Método
 - Desenvolvimento
 - Estrutura de Dados
 - Algoritmo
 - Corretismo

- 3 Avaliação
 - Método de Avaliação
 - Análise computacional
 - Resultados

Composição dos Testes

Para a realização de testes, d foi fixado em 2. Foram realizados testes para todas as combinações dos seguintes parâmetros:

X Cinco imagens: Figura 6.1, 6.2a, 6.2b, 6.2c e 6.2d.

$k(\mathcal{L})$ Variação unitária de 1 a 15.

EE 3×3 ou 5×5 .

Operação Duas operações: Transformada da Erosão e Transformada da Dilatação.

Algoritmo Três tipos de algoritmos: TRANSFORMADADUALLINEAR, TRANSFORMADADUALLINEARDEBORDA e TRANSFORMADAPROPOSTA.

Imagens de Teste X



Figura 6.1



Figura 6.2a

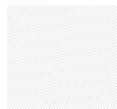


Figura 6.2b

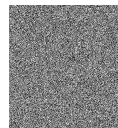


Figura 6.2c



Figura 6.2d

Sequências $L_{1,3 \times 3} \cdots L_{15,3 \times 3}$

(-1,-1)	(0,-1)	(1,-1)
	(0,0)	
(-1,1)	(0,1)	

 $L_{1,3 \times 3}$

(-1,-1)	(0,-1)	(1,-1)
(-1,0)	(0,0)	(1,0)

 $L_{2,3 \times 3}$

	(0,0)	(1,0)
	(0,1)	(1,1)

 $L_{3,3 \times 3}$

	(0,-1)	(1,-1)
	(0,0)	(1,0)
	(0,1)	

 $L_{4,3 \times 3}$

	(0,-1)	
	(0,0)	(1,0)
(-1,1)	(0,1)	

 $L_{5,3 \times 3}$

(-1,-1)	(0,-1)	(1,-1)
(-1,0)	(0,0)	
(-1,1)		

 $L_{6,3 \times 3}$

	(0,0)	
(-1,1)	(0,1)	(1,1)

 $L_{7,3 \times 3}$

	(0,-1)	(1,-1)
	(0,0)	
	(0,1)	(1,1)

 $L_{8,3 \times 3}$

	(0,-1)	(1,-1)
(-1,0)	(0,0)	
	(0,1)	(1,1)

 $L_{9,3 \times 3}$

	(0,-1)	
(-1,0)	(0,0)	(1,0)

 $L_{10,3 \times 3}$

(-1,-1)		
	(0,0)	(1,0)
		(1,1)

 $L_{11,3 \times 3}$

(-1,0)	(0,0)	(1,0)

 $L_{12,3 \times 3}$

	(0,-1)	
	(0,0)	
	(0,1)	

 $L_{13,3 \times 3}$

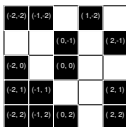
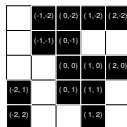
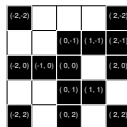
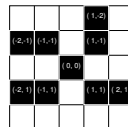
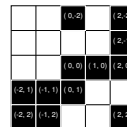
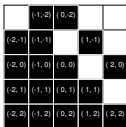
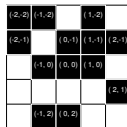
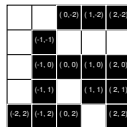
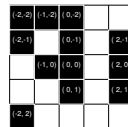
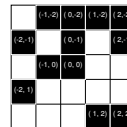
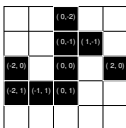
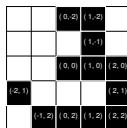
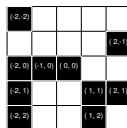
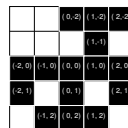
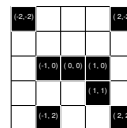
(-1,-1)		
(-1,0)	(0,0)	
(-1,1)	(0,1)	

 $L_{14,3 \times 3}$

(-1,-1)	(0,-1)	(1,-1)
	(0,0)	
	(0,1)	(1,1)

 $L_{15,3 \times 3}$

Sequências $L_{1,5 \times 5} \cdots L_{15,5 \times 5}$

 $L_{1,5 \times 5}$  $L_{2,5 \times 5}$  $L_{3,5 \times 5}$  $L_{4,5 \times 5}$  $L_{5,5 \times 5}$  $L_{6,5 \times 5}$  $L_{7,5 \times 5}$  $L_{8,5 \times 5}$  $L_{9,5 \times 5}$  $L_{10,5 \times 5}$  $L_{11,5 \times 5}$  $L_{12,5 \times 5}$  $L_{13,5 \times 5}$  $L_{14,5 \times 5}$  $L_{15,5 \times 5}$

Tempo representativo da combinação

X Cinco imagens: Figura 6.1, 6.2a, 6.2b, 6.2c e 6.2d.

k (\mathcal{L}) 1 (\mathcal{L}_1) ou .. ou 15 (\mathcal{L}_{15}).

EE 3×3 ou 5×5 .

Operação Duas operações: Transformada da Erosão e Transformada da Dilatação.

Algoritmo Três tipos de algoritmos: TRANSFORMADADUALLINEAR, TRANSFORMADADUALLINEARDEBORDA e TRANSFORMADAPROPOSTA.

- **Total de combinações:** 900.
- **Cada combinação:** 1001 execuções.
- **Tempo representativo da combinação:** média das 1001 medidas de tempo.

- 1 Problema
 - Arcabouço
 - Motivação
 - Definição do Problema

- 2 Método
 - Desenvolvimento
 - Estrutura de Dados
 - Algoritmo
 - Corretismo

- 3 Avaliação
 - Método de Avaliação
 - **Análise computacional**
 - Resultados

Tabela comparativa de análise computacional

	Tempo Computacional	Espaço Computacional
TRANSFORMADA DUAL LINEAR	$o\left([X] + \left[\bigcup_{i=1}^k L_i\right] + \sum_{i=1}^k L_i \right) + k \cdot \left \bigcup_{i=1}^k L_i\right + \sum_{x=1}^{m-k} t_{x-1} \cdot Lx $	$o\left([X] + \left[\bigcup_{i=1}^k L_i\right] + \sum_{i=1}^k L_i \right)$
TRANSFORMADA DUAL LINEAR DE BORDA	$o\left([X] + \left[\bigcup_{i=1}^k L_i\right] + \sum_{i=1}^k L_i \right) + (k + b_0) \cdot \left \bigcup_{i=1}^k L_i\right + \sum_{x=1}^{m-k} \left \text{borda}_{\bigcup_{i=1}^k L_i} (t_{x-1}) \right \cdot Lx + \sum_{x=1}^{m-k} \left \text{borda}_{Lx} (t_{x-1}) \right \cdot \left \bigcup_{i=1}^k L_i\right $	$o\left([X] + \left[\bigcup_{i=1}^k L_i\right] + \sum_{i=1}^k L_i \right)$
TRANSFORMADA PROPOSTA	$o\left([X] + \left[\bigcup_{i=1}^k L_i\right] + \sum_{i=1}^k L_i \right) + (k + b_0) \cdot \left \bigcup_{i=1}^k L_i\right + \left \text{borda}_{\bigcup_{i=1}^k L_i} (b) \right \cdot \left \bigcup_{i=1}^k L_i\right + \sum_{x=1}^{m-k} \left \text{borda}_{Lx} (t_{x-1}) \right \cdot \left \bigcup_{i=1}^k L_i\right $	$o\left([X] + \max_x \left\{ \left \text{borda}_{\bigcup_{i=1}^k L_i} (t_x) \right \cdot \left \bigcup_{i=1}^k L_i\right \right\}\right)$

- 1 Problema
 - Arcabouço
 - Motivação
 - Definição do Problema

- 2 Método
 - Desenvolvimento
 - Estrutura de Dados
 - Algoritmo
 - Corretismo

- 3 Avaliação
 - Método de Avaliação
 - Análise computacional
 - Resultados

Figura 6.1

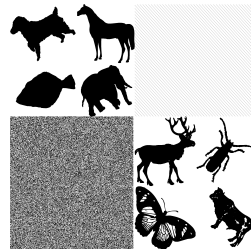
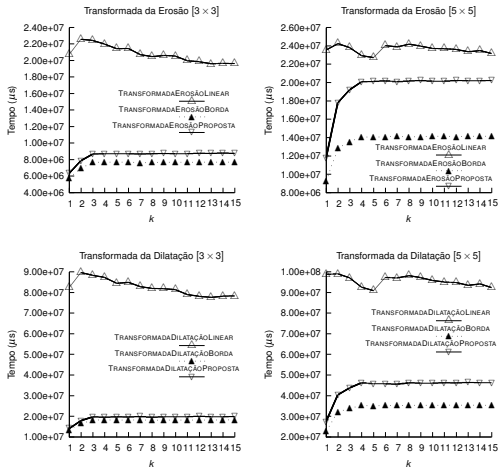


Figura 6.2a

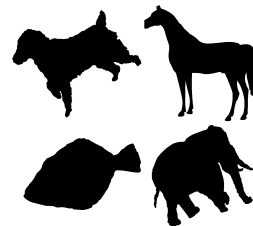
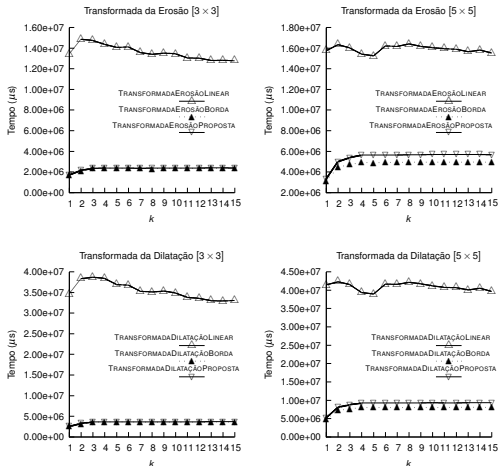


Figura 6.2b

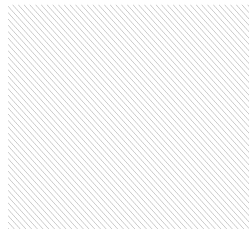
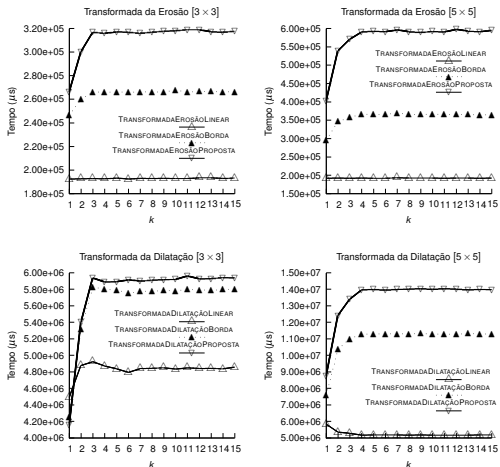


Figura 6.2c

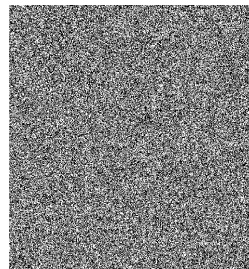
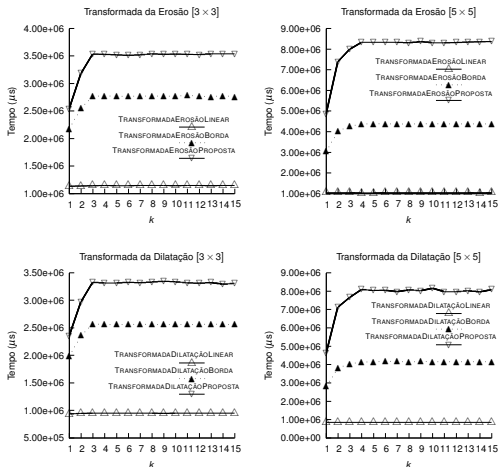
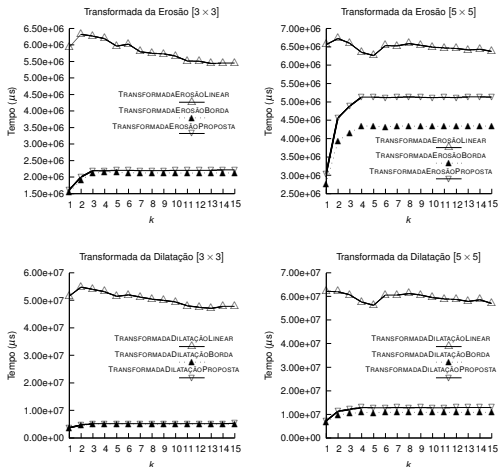


Figura 6.2d



Resultados

- Tempo computacional:
 $O(\text{TRANSFORMADADUALLINEARDEBORDA}) < O(\text{TRANSFORMADAPROPOSTA})$.
- TRANSFORMADADUALLINEARDEBORDA:
 caso médio difere do pior caso.
- TRANSFORMADAPROPOSTA:
 caso médio equivale ao pior caso.
- TRANSFORMADAPROPOSTA e TRANSFORMADADUALLINEARDEBORDA
 dependem do custo de manutenção da borda.

Resumo

- Para imagens cuja manutenção da borda consuma menos tempo que a própria iteração dos elementos a serem analisados, e.g., imagens naturais, TRANSFORMADADUALLINEARDEBORDA e TRANSFORMADAPROPOSTA apresentam bom desempenho.
- TRANSFORMADAPROPOSTA minimiza o número de comparações para a determinação dos elementos a serem analisados. Entretanto, a manutenção da estrutura que garante tal propriedade é onerosa.
- Sugestões para pesquisas futuras
 - Iteração em conjuntos, não em sequências pré-definidas.
 - Efetuar testes para d superior a 2.

Referências

- Chen e Haralick(1995)** Su Chen e R. M. Haralick. Recursive erosion, dilation, opening, and closing transforms. *Trans. Img. Proc.*, 4(3):335–345. ISSN 1057-7149. doi: 10.1109/83.366481. URL <http://dx.doi.org/10.1109/83.366481>.
- Hirata(1997)** Roberto Hirata, Jr. Segmentação de imagens por morfologia matemática. Dissertação de Mestrado, Instituto de Matemática e Estatística, Universidade de São Paulo, Brasil.
- Machado(2008)** Anderson F. Machado. Erosões e dilatações morfológicas binárias sequenciais rápidas. Dissertação de Mestrado, Instituto de Matemática e Estatística, Universidade de São Paulo, Brasil.
- Zampirolli(2003)** Francisco de Assis Zampirolli. *Transformada de Distância por Morfologia Matemática*. Tese de Doutorado, Faculdade de Engenharia Eletrica e de Computação, Universidade Estadual de Campinas, Brasil.

Obrigado!

Auxílio financeiro

■ CNPq

■ FAPESP

Alexandre Y. Harano

ayharano@ime.usp.br

<http://www.ime.usp.br/~ayharano/>

