

# Transformada Imagem-Floresta

Alexandre Xavier Falcão

Instituto de Computação - UNICAMP

afalcao@ic.unicamp.br

## O que é?

A transformada imagem-floresta (IFT) é uma metodologia para o desenvolvimento de técnicas de processamento de imagem usando **caminhos ótimos** em grafos derivados da imagem.

A IFT teve como motivação os seguintes fatos:

A IFT teve como motivação os seguintes fatos:

- Processamento de imagem utiliza conceitos de diversas disciplinas com formalismos matemáticos distintos e muitas vezes distantes da implementação.

A IFT teve como motivação os seguintes fatos:

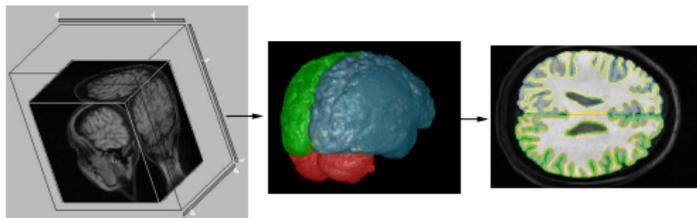
- Processamento de imagem utiliza conceitos de diversas disciplinas com formalismos matemáticos distintos e muitas vezes distantes da implementação.
- O formalismo matemático deveria unir a explicação de um método no universo físico com a sua implementação no universo computacional.

A IFT teve como motivação os seguintes fatos:

- Processamento de imagem utiliza conceitos de diversas disciplinas com formalismos matemáticos distintos e muitas vezes distantes da implementação.
- O formalismo matemático deveria unir a explicação de um método no universo físico com a sua implementação no universo computacional.
- Teoria de grafos possibilita esta ligação de forma intuitiva e elegante, oferecendo vários algoritmos com prova de corretude.

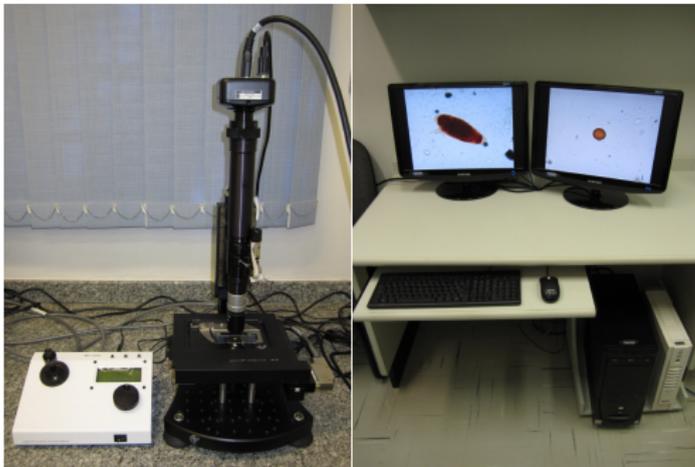
Como metodologia de pesquisa, a IFT tem sido usada com sucesso em diversos projetos.

- Análise de imagens 3D do cérebro humano.



# Introdução

Como metodologia de pesquisa, a IFT tem sido usada com sucesso em diversos projetos.



- Análise de imagens 3D do cérebro humano.
- Diagnóstico automatizado de enteroparasitoses.

# Introdução

Como metodologia de pesquisa, a IFT tem sido usada com sucesso em diversos projetos.



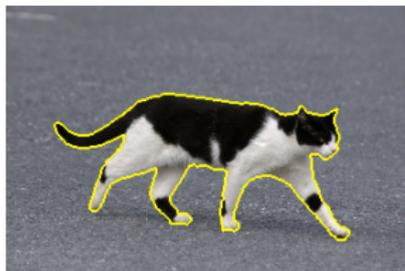
consulta



- Análise de imagens 3D do cérebro humano.
- Diagnóstico automatizado de enteroparasitoses.
- Busca de imagens por conteúdo em bases de dados.

# Introdução

Como metodologia de pesquisa, a IFT tem sido usada com sucesso em diversos projetos.



- Análise de imagens 3D do cérebro humano.
- Diagnóstico automatizado de enteroparasitoses.
- Busca de imagens por conteúdo em bases de dados.
- Edição de fotos e rastreamento de objetos em vídeo.

- Esta aula apresenta a IFT nos universos:

- Esta aula apresenta a IFT nos universos:
  - **Físico** - como uma teoria de formação ordenada de comunidades.

# Objetivo e organização

- Esta aula apresenta a IFT nos universos:
  - **Físico** - como uma teoria de formação ordenada de comunidades.
  - **Matemático** - como uma transformação que leva a imagem em um grafo e este em uma floresta de caminhos ótimos.

# Objetivo e organização

- Esta aula apresenta a IFT nos universos:
  - **Físico** - como uma teoria de formação ordenada de comunidades.
  - **Matemático** - como uma transformação que leva a imagem em um grafo e este em uma floresta de caminhos ótimos.
  - **Computacional** - como a extensão do algoritmo de Dijkstra para funções de conectividade mais gerais e com múltiplas fontes.

# Objetivo e organização

- Esta aula apresenta a IFT nos universos:
  - **Físico** - como uma teoria de formação ordenada de comunidades.
  - **Matemático** - como uma transformação que leva a imagem em um grafo e este em uma floresta de caminhos ótimos.
  - **Computacional** - como a extensão do algoritmo de Dijkstra para funções de conectividade mais gerais e com múltiplas fontes.
- Exemplos de como usar a IFT em processamento de imagem.

# Objetivo e organização

- Esta aula apresenta a IFT nos universos:
  - **Físico** - como uma teoria de formação ordenada de comunidades.
  - **Matemático** - como uma transformação que leva a imagem em um grafo e este em uma floresta de caminhos ótimos.
  - **Computacional** - como a extensão do algoritmo de Dijkstra para funções de conectividade mais gerais e com múltiplas fontes.
- Exemplos de como usar a IFT em processamento de imagem.
- Comentários conclusivos.

# Formação ordenada de comunidades

- Em uma população, cada indivíduo possui um desejo intrínseco para liderar comunidades.

# Formação ordenada de comunidades

- Em uma população, cada indivíduo possui um desejo intrínseco para liderar comunidades.
- Indivíduos com desejo maior oferecem recompensas aos seus conhecidos para que esses façam parte de suas comunidades.

# Formação ordenada de comunidades

- Em uma população, cada indivíduo possui um desejo intrínseco para liderar comunidades.
- Indivíduos com desejo maior oferecem recompensas aos seus conhecidos para que esses façam parte de suas comunidades.
- O conhecido **troca de comunidade** sempre que a recompensa que lhe é oferecida for maior do que a sua atual.

# Formação ordenada de comunidades

- Em uma população, cada indivíduo possui um desejo intrínseco para liderar comunidades.
- Indivíduos com desejo maior oferecem recompensas aos seus conhecidos para que esses façam parte de suas comunidades.
- O conhecido **troca de comunidade** sempre que a recompensa que lhe é oferecida for maior do que a sua atual.
- Um líder é aquele cujo desejo intrínseco é maior do que qualquer recompensa que lhe é oferecida.

- Um indivíduo nunca oferece recompensa maior do que a sua.

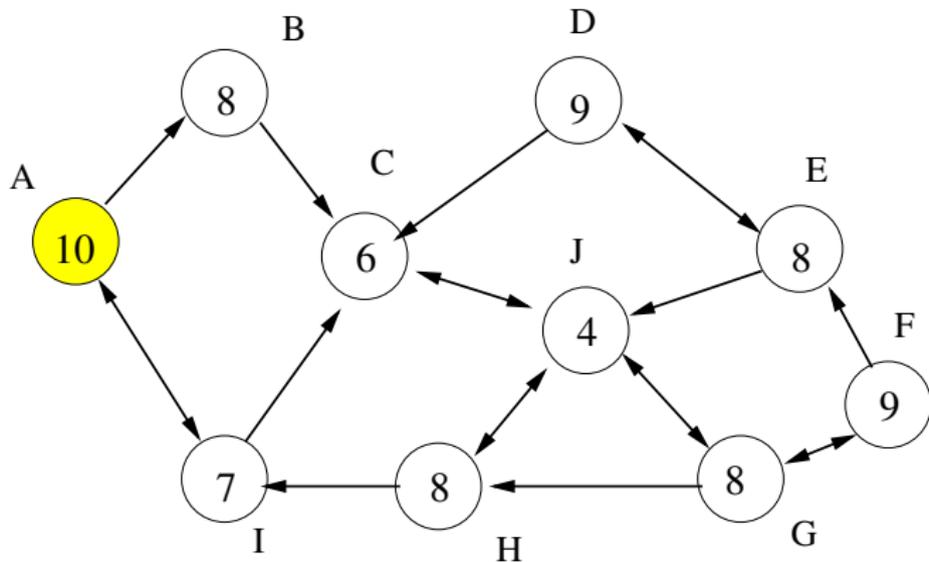
# Formação ordenada de comunidades

- Um indivíduo nunca oferece recompensa maior do que a sua.
- O processo segue a ordem dos indivíduos **de maior** recompensa para os **de menor** recompensa.

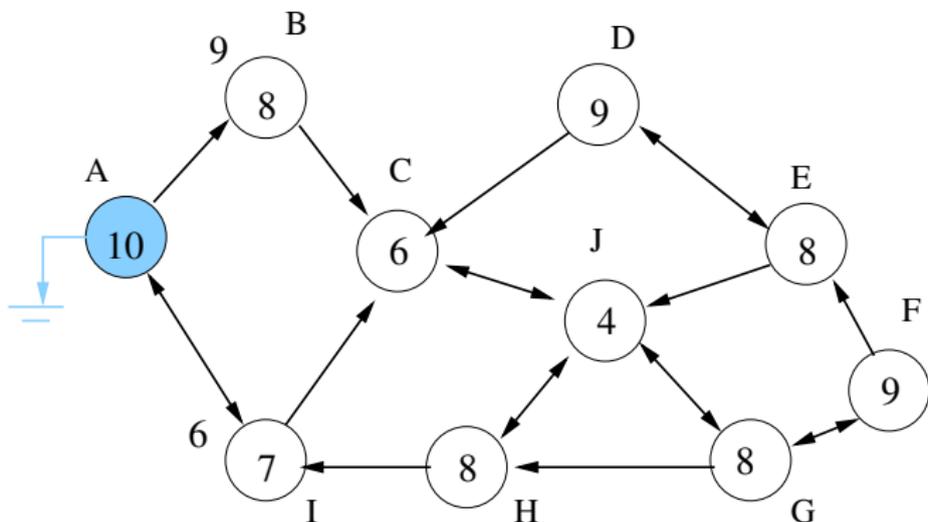
# Formação ordenada de comunidades

- Um indivíduo nunca oferece recompensa maior do que a sua.
- O processo segue a ordem dos indivíduos **de maior** recompensa para os **de menor** recompensa.
- A população é particionada em comunidades, onde cada indivíduo pertencerá ao grupo que lhe ofereceu **recompensa máxima**.

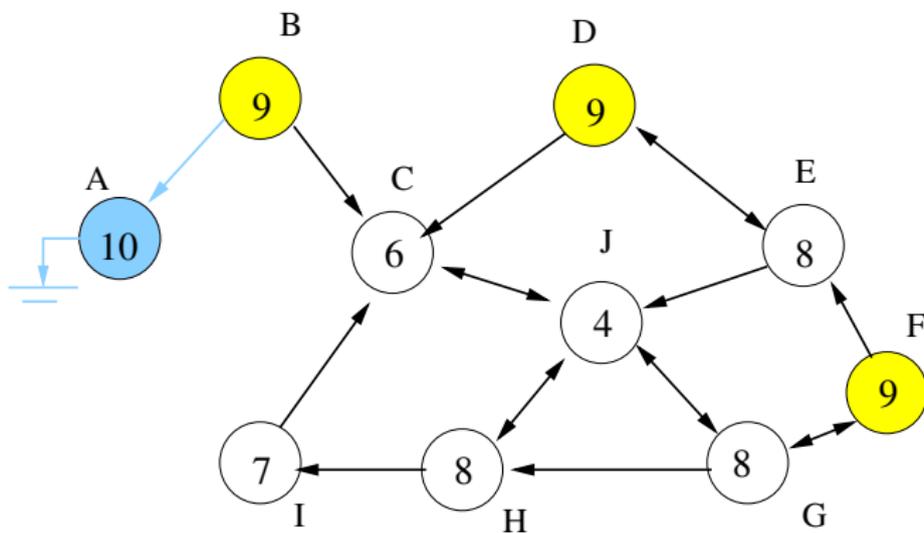
# Formação ordenada de comunidades



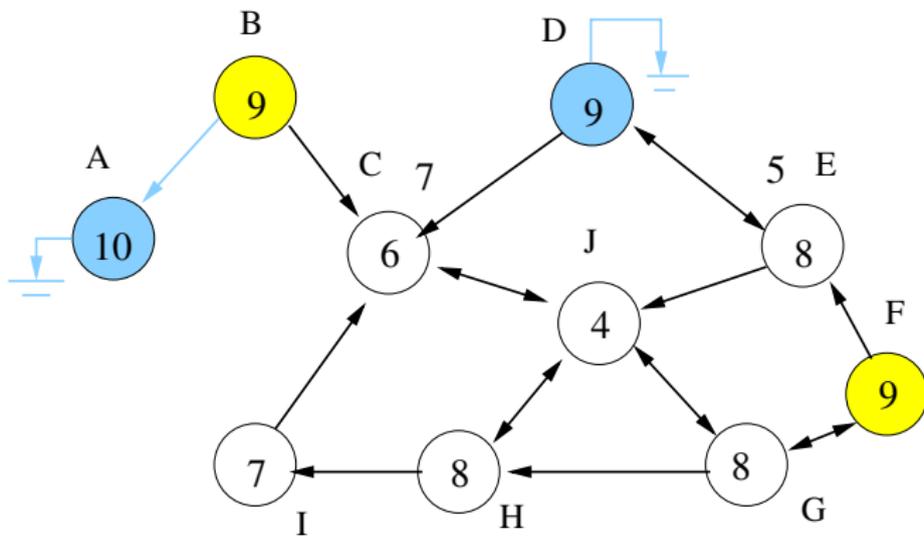
# Formação ordenada de comunidades



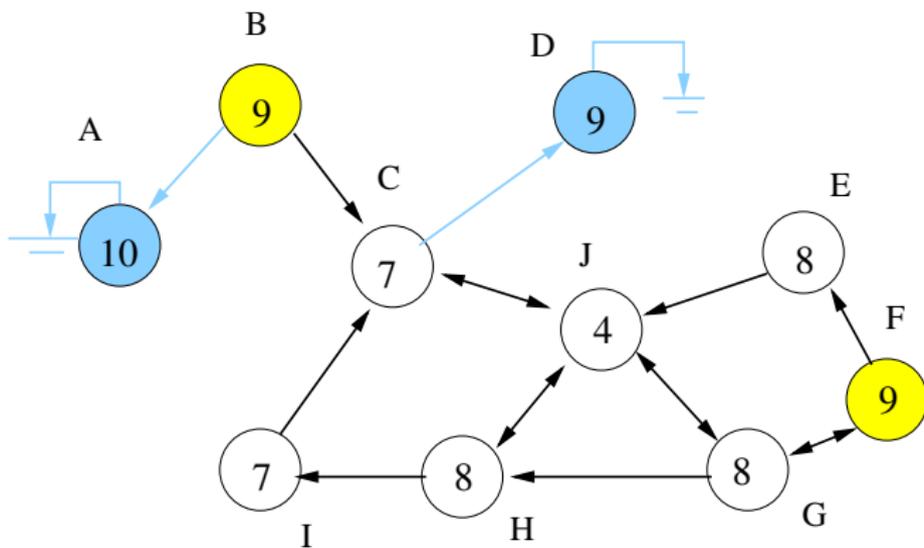
# Formação ordenada de comunidades



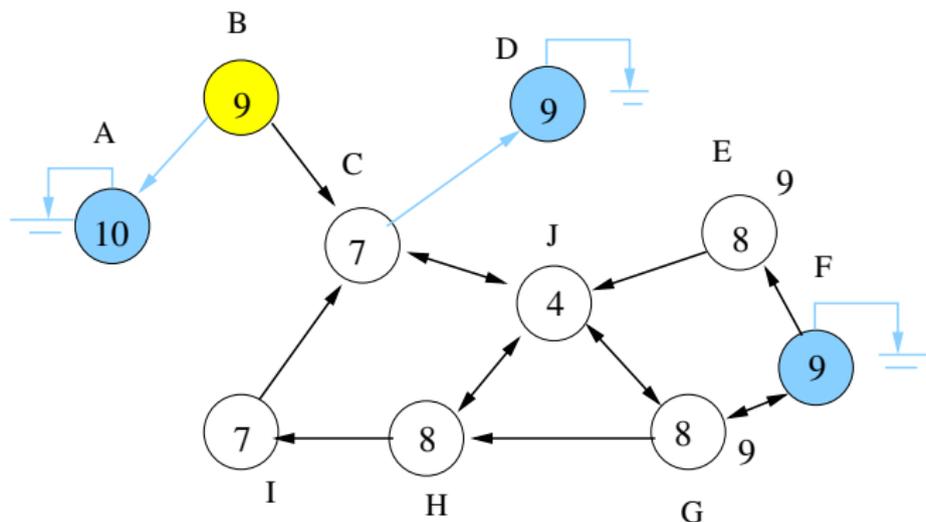
# Formação ordenada de comunidades



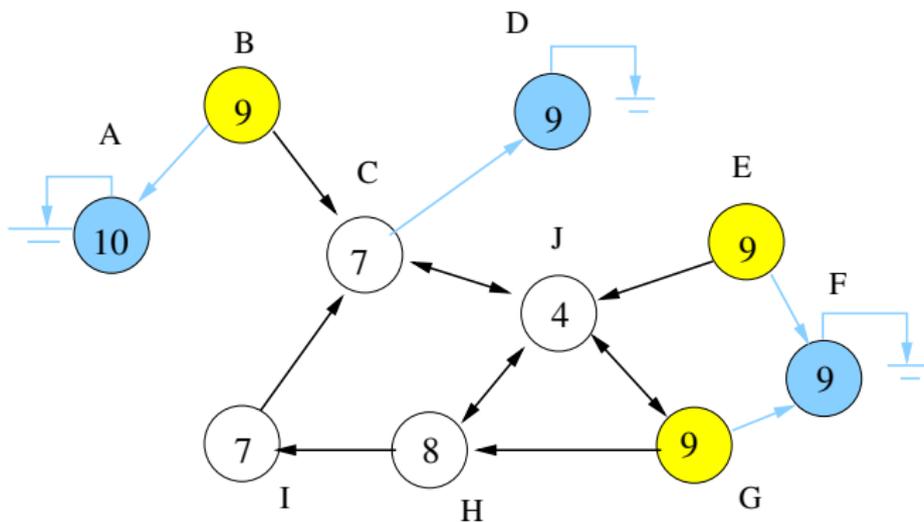
# Formação ordenada de comunidades



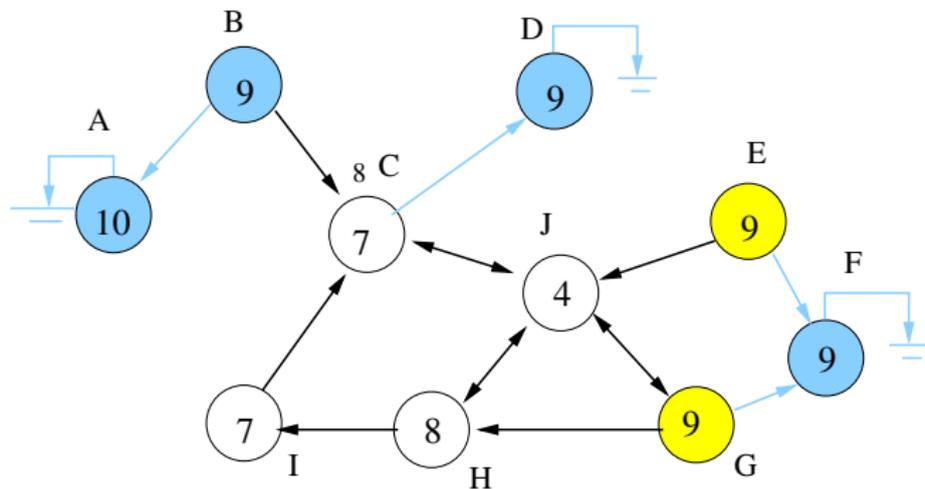
# Formação ordenada de comunidades



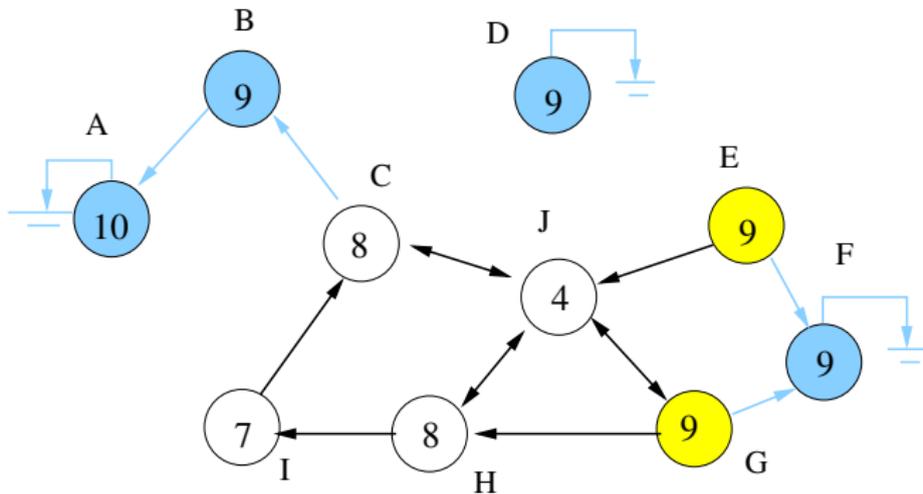
# Formação ordenada de comunidades



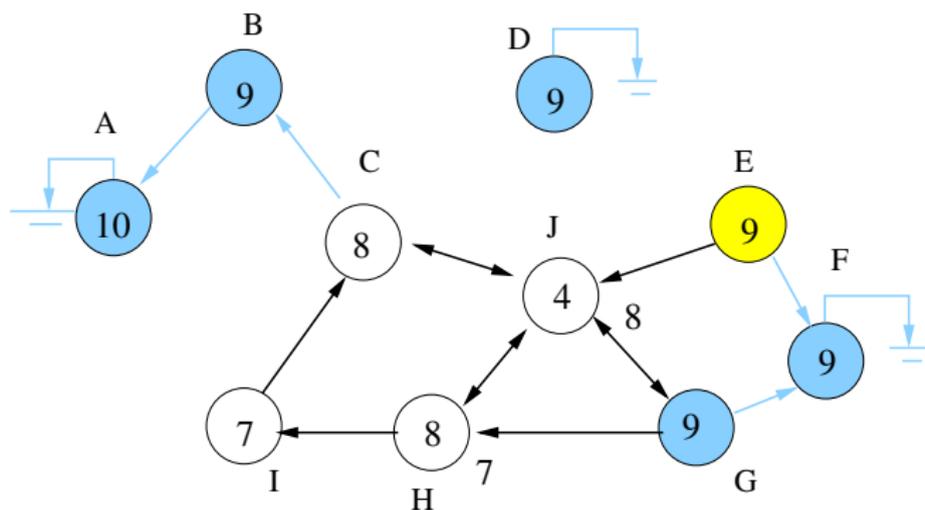
# Formação ordenada de comunidades



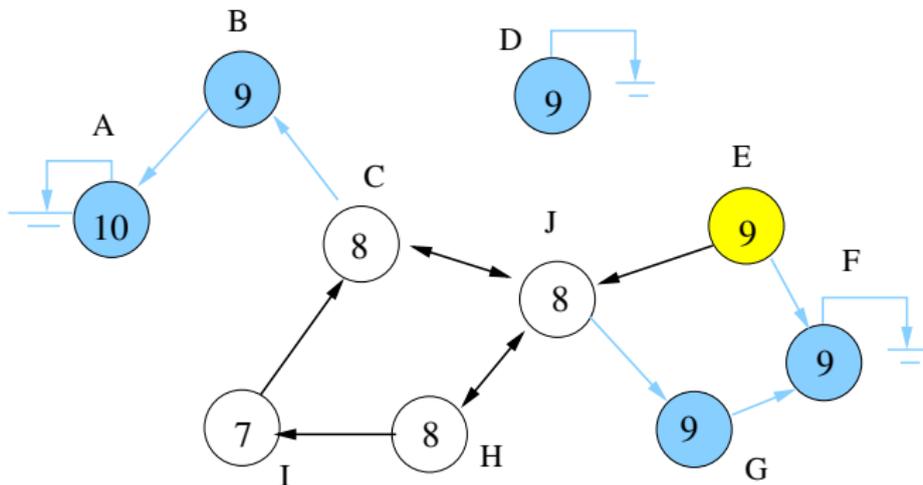
# Formação ordenada de comunidades



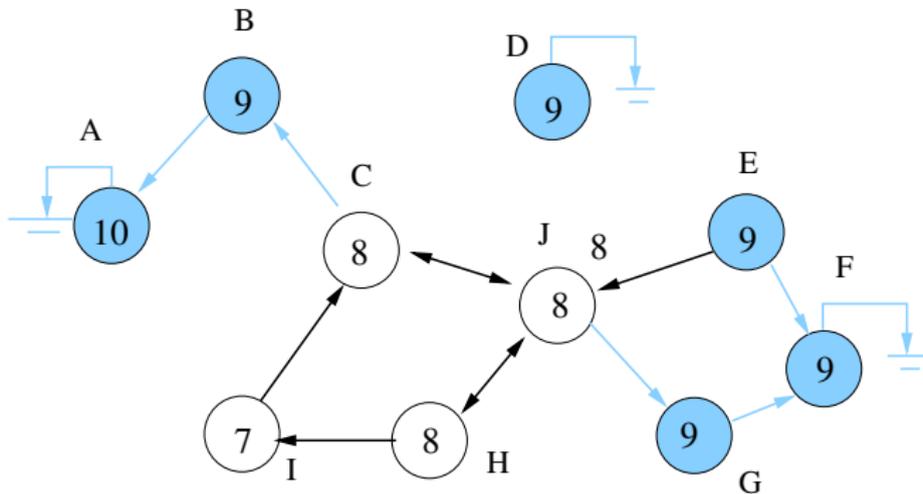
# Formação ordenada de comunidades



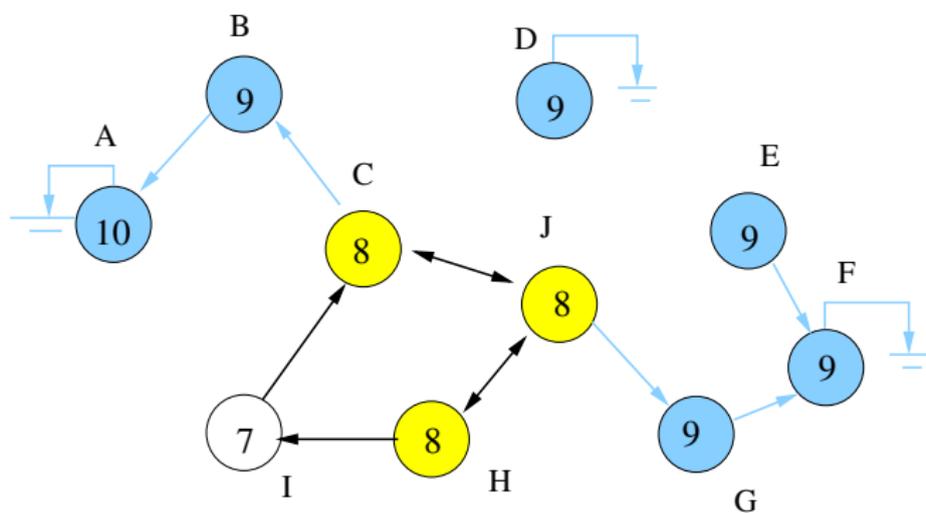
# Formação ordenada de comunidades



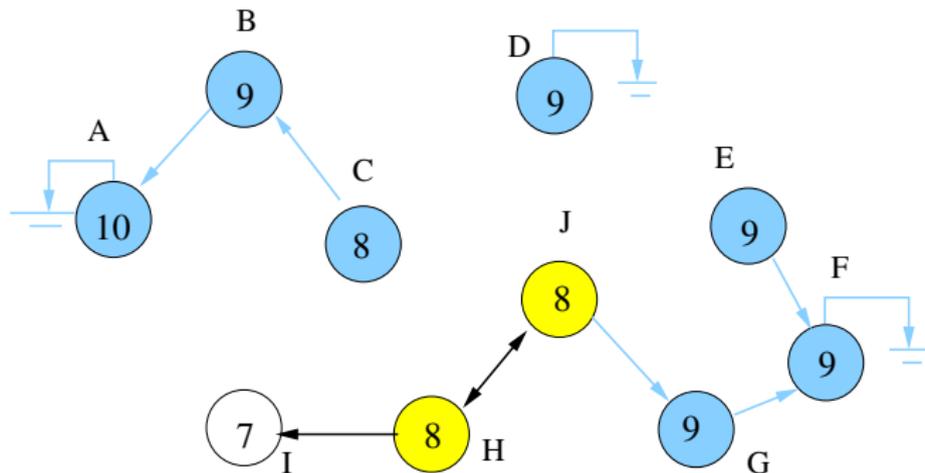
# Formação ordenada de comunidades



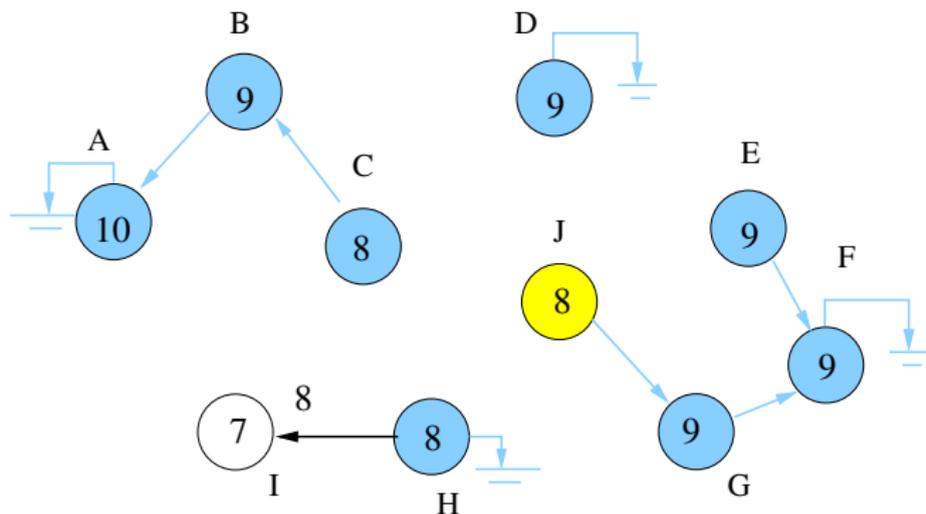
# Formação ordenada de comunidades



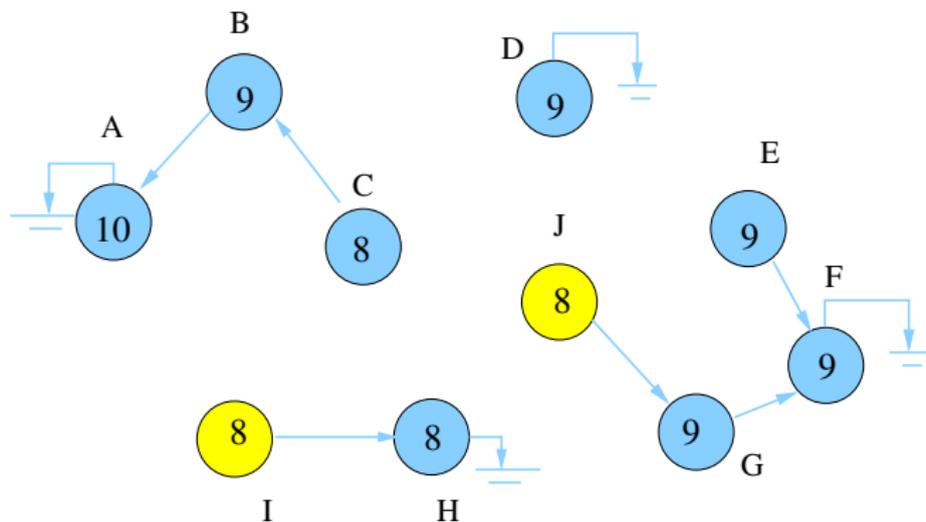
# Formação ordenada de comunidades



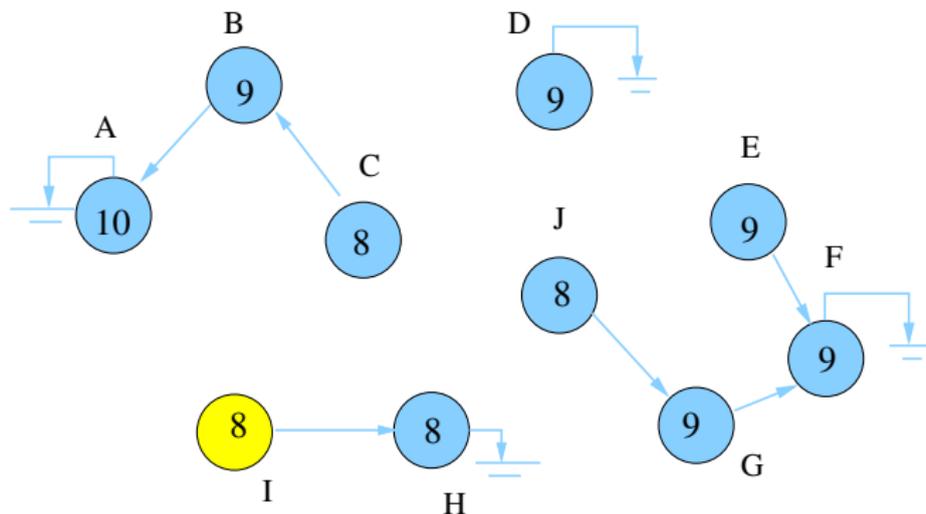
# Formação ordenada de comunidades



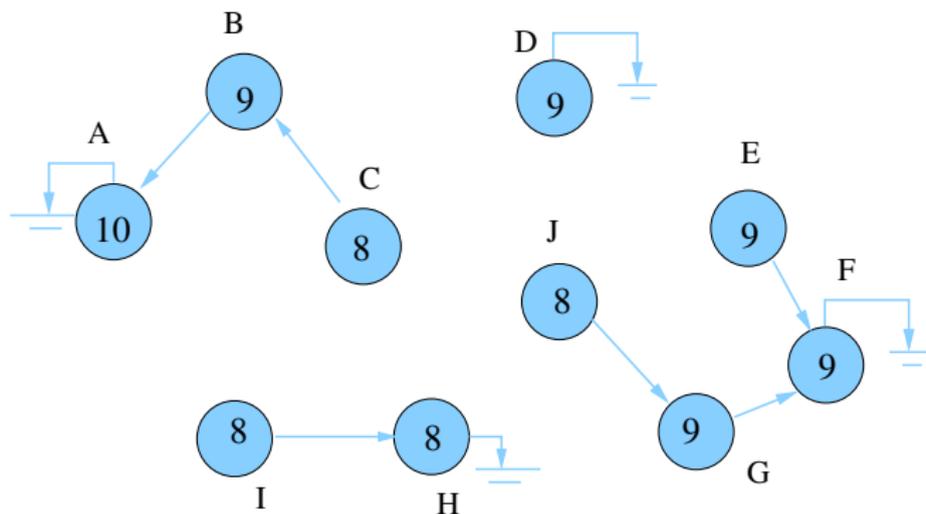
# Formação ordenada de comunidades



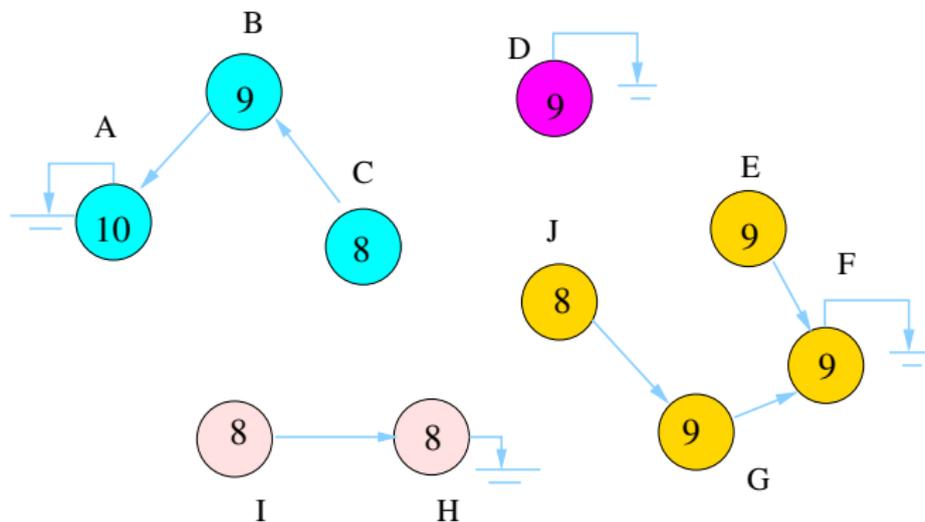
# Formação ordenada de comunidades



# Formação ordenada de comunidades



# Formação ordenada de comunidades



# Transformada imagem-floresta

- Uma imagem digital  $\hat{I}$  é um par  $(D_I, \vec{I})$ , onde  $\vec{I}(t) \in Z^m$  associa  $m$  atributos a cada pixel  $t$  do domínio  $D_I \subset Z^n$ .

# Transformada imagem-floresta

- Uma imagem digital  $\hat{I}$  é um par  $(D_I, \vec{I})$ , onde  $\vec{I}(t) \in Z^m$  associa  $m$  atributos a cada pixel  $t$  do domínio  $D_I \subseteq Z^n$ .
- Os pixels são os indivíduos de uma população  $\mathcal{N} \subseteq D_I$ .

# Transformada imagem-floresta

- Uma imagem digital  $\hat{I}$  é um par  $(D_I, \vec{I})$ , onde  $\vec{I}(t) \in Z^m$  associa  $m$  atributos a cada pixel  $t$  do domínio  $D_I \subseteq Z^n$ .
- Os pixels são os indivíduos de uma população  $\mathcal{N} \subseteq D_I$ .
- Um indivíduo ser conhecido por outro é uma **relação de adjacência**  $\mathcal{A}$  entre pixels, tal que  $t \in \mathcal{A}(s)$  define um arco  $(s, t) \in \mathcal{A}$ ,  $s \neq t$ , no grafo  $(\mathcal{N}, \mathcal{A})$ .

# Transformada imagem-floresta

- Uma imagem digital  $\hat{I}$  é um par  $(D_I, \vec{I})$ , onde  $\vec{I}(t) \in Z^m$  associa  $m$  atributos a cada pixel  $t$  do domínio  $D_I \subseteq Z^n$ .
- Os pixels são os indivíduos de uma população  $\mathcal{N} \subseteq D_I$ .
- Um indivíduo ser conhecido por outro é uma **relação de adjacência**  $\mathcal{A}$  entre pixels, tal que  $t \in \mathcal{A}(s)$  define um arco  $(s, t) \in \mathcal{A}$ ,  $s \neq t$ , no grafo  $(\mathcal{N}, \mathcal{A})$ .
- A sequência de convites que parte de um líder  $s_1$  até um membro  $s_n = t$  passando por outros é um **caminho simples**  $\pi_t = \langle s_1, s_2, \dots, s_n \rangle$ , onde  $(s_i, s_{i+1}) \in \mathcal{A}$ , sendo  $\langle t \rangle$  um **caminho trivial**.

# Transformada imagem-floresta

- Uma **função de conexidade**  $f$  aplicada a caminhos triviais,  $f(\langle t \rangle)$ , define o desejo intrínseco de  $t$  ser um líder.

# Transformada imagem-floresta

- Uma **função de conexidade**  $f$  aplicada a caminhos triviais,  $f(\langle t \rangle)$ , define o desejo intrínseco de  $t$  ser um líder.
- Sua aplicação para caminhos simples,  $f(\pi_s \cdot \langle s, t \rangle)$ , representa a recompensa que  $s$  oferece a  $t$ .

# Transformada imagem-floresta

- Uma **função de conexidade**  $f$  aplicada a caminhos triviais,  $f(\langle t \rangle)$ , define o desejo intrínseco de  $t$  ser um líder.
- Sua aplicação para caminhos simples,  $f(\pi_s \cdot \langle s, t \rangle)$ , representa a recompensa que  $s$  oferece a  $t$ .
- As comunidades são definidas pela **maximização** de um mapa  $V(t)$  de conexidade.

$$V(t) = \max_{\forall \pi_t \in \Pi(\mathcal{N}, \mathcal{A}, t)} \{f(\pi_t)\},$$

onde  $\Pi(\mathcal{N}, \mathcal{A}, t)$  é o conjunto de todos os caminhos com término  $t$ .

# Transformada imagem-floresta

- Uma **função de conexidade**  $f$  aplicada a caminhos triviais,  $f(\langle t \rangle)$ , define o desejo intrínseco de  $t$  ser um líder.
- Sua aplicação para caminhos simples,  $f(\pi_s \cdot \langle s, t \rangle)$ , representa a recompensa que  $s$  oferece a  $t$ .
- As comunidades são definidas pela **maximização** de um mapa  $V(t)$  de conexidade.

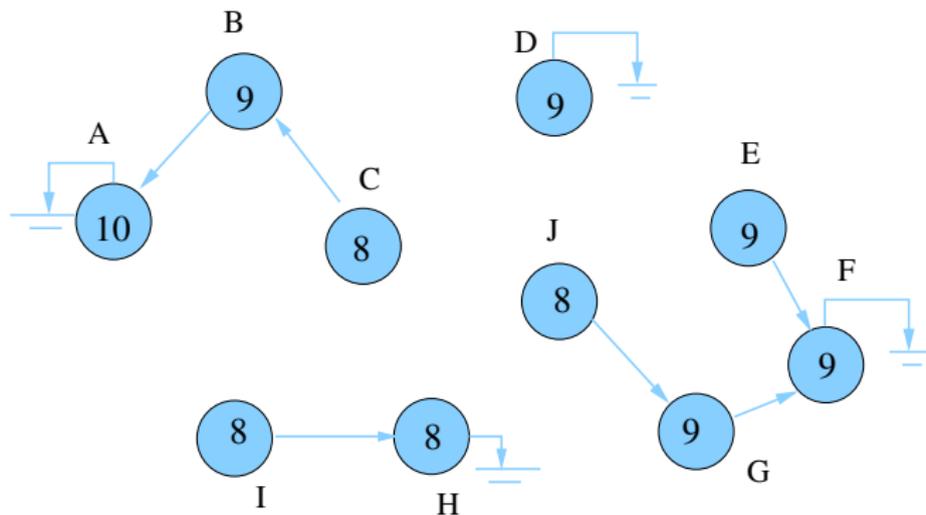
$$V(t) = \max_{\forall \pi_t \in \Pi(\mathcal{N}, \mathcal{A}, t)} \{f(\pi_t)\},$$

onde  $\Pi(\mathcal{N}, \mathcal{A}, t)$  é o conjunto de todos os caminhos com término  $t$ .

Porém, seguindo a **ordem não-crescente** de conexidade ótima.

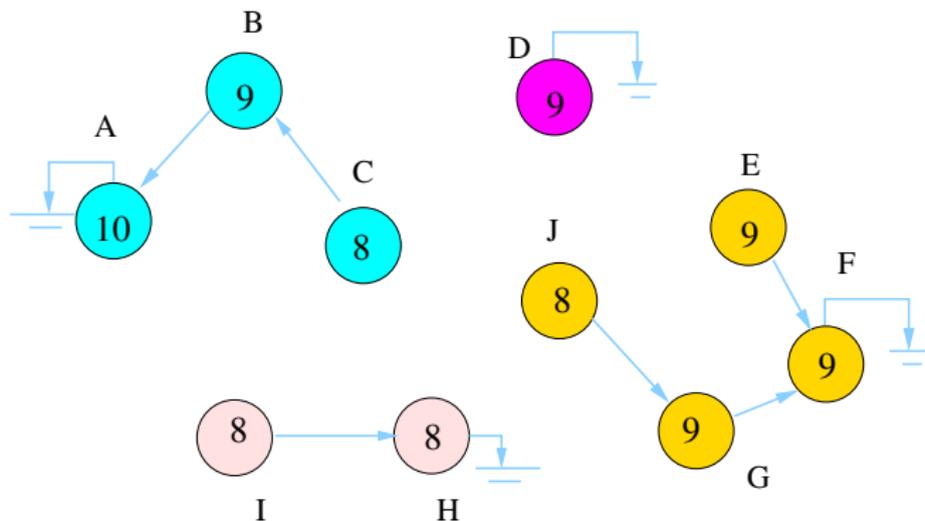
# Algoritmo geral da IFT

O algoritmo da IFT resolve o problema gerando uma **floresta de caminhos ótimos**  $P$  — mapeamento acíclico que associa  $P(t) = nil \notin \mathcal{N}$ , quando  $t$  é um líder (raiz da floresta), ou um predecessor  $P(t) = s \in \mathcal{N}$  no caminho ótimo  $P^*(t)$ .



# Algoritmo geral da IFT

Como subprodutos, o algoritmo da IFT também gera o mapa  $V(t)$  de conectividade máxima e a **partição ótima**  $R(t)$ , que associa a cada pixel  $t$  a raiz  $R(t)$  de maior conectividade ou qualquer outro rótulo  $L(t)$  associado à  $R(t)$ .

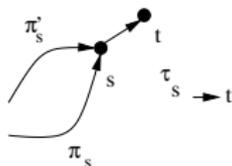


# Corretude

A corretude do algoritmo da IFT requer que para todo pixel  $t \in \mathcal{N}$ , exista ao menos um **caminho ótimo**  $\pi_t$ , de forma trivial ou simples

$\pi_t = \pi_s \cdot \tau_{s \rightarrow t}$ , onde:

- 1  $f(\pi_s) \geq f(\pi_t)$ .
- 2  $\pi_s$  é ótimo.
- 3 Para qualquer prefixo ótimo  $\pi'_s$ ,  $f(\pi'_s \cdot \tau_{s \rightarrow t}) = f(\pi_t)$ .

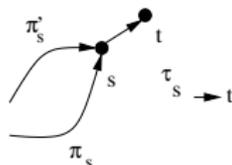


# Corretude

A corretude do algoritmo da IFT requer que para todo pixel  $t \in \mathcal{N}$ , exista ao menos um **caminho ótimo**  $\pi_t$ , de forma trivial ou simples

$\pi_t = \pi_s \cdot \tau_{s \rightarrow t}$ , onde:

- 1  $f(\pi_s) \geq f(\pi_t)$ .
- 2  $\pi_s$  é ótimo.
- 3 Para qualquer prefixo ótimo  $\pi'_s$ ,  $f(\pi'_s \cdot \tau_{s \rightarrow t}) = f(\pi_t)$ .



Estas condições são aplicadas apenas a caminhos ótimos (extensão do **algoritmo de Dijkstra**).

## Algorithm

– ALGORITMO GERAL DA IFT

1. Para todo  $t \in \mathcal{N}$ , faça
2.      $P(t) \leftarrow \text{nil}$ ,  $R(t) \leftarrow t$  e  $V(t) \leftarrow f(\langle t \rangle)$ .
3.     Se  $V(t) \neq -\infty$ , então insira  $t$  em  $Q$ .
4. Enquanto  $Q \neq \emptyset$ , faça
5.     Remova  $s$  de  $Q$  com  $V(s)$  máxima.
6.     Para cada  $t \in \mathcal{A}(s)$  tal que  $V(t) < V(s)$ , faça
7.          $\text{tmp} \leftarrow f(\pi_s \cdot \langle s, t \rangle)$ .
8.         Se  $\text{tmp} > V(t)$ , então
9.             Se  $V(t) \neq -\infty$ , remova  $t$  de  $Q$ .
10.              $P(t) \leftarrow s$ ,  $V(t) \leftarrow \text{tmp}$ ,  $R(t) \leftarrow R(s)$ .
11.             Insira  $t$  em  $Q$ .

# Como usamos a IFT?

- Identificamos um problema com associação direta ou indireta com um problema de **partição ótima** ou **conexidade ótima**.

# Como usamos a IFT?

- Identificamos um problema com associação direta ou indireta com um problema de **partição ótima** ou **conexidade ótima**.
- Note que esta partição pode ser obtida também via **minimização** de um mapa de custos  $V(t)$ , usando caminhos de custo mínimo.

# Como usamos a IFT?

- Identificamos um problema com associação direta ou indireta com um problema de **partição ótima** ou **conexidade ótima**.
- Note que esta partição pode ser obtida também via **minimização** de um mapa de custos  $V(t)$ , usando caminhos de custo mínimo.
- Especificamos então uma **relação de adjacência** e uma **função de conexidade** adequadas ao problema.

# Como usamos a IFT?

- Identificamos um problema com associação direta ou indireta com um problema de **partição ótima** ou **conexidade ótima**.
- Note que esta partição pode ser obtida também via **minimização** de um mapa de custos  $V(t)$ , usando caminhos de custo mínimo.
- Especificamos então uma **relação de adjacência** e uma **função de conexidade** adequadas ao problema.
- A IFT reduz o problema ao cálculo de uma floresta de caminhos ótimos seguido de um processamento local sobre seus atributos.

Relações de adjacência podem ser definidas no domínio da imagem, espaço de atributos, ou ambos.

Relações de adjacência podem ser definidas no domínio da imagem, espaço de atributos, ou ambos.

- Exemplo 1:  $(s, t) \in \mathcal{A}$  quando  $\|t - s\| \leq r$ , para  $r \geq 1$ .

Relações de adjacência podem ser definidas no domínio da imagem, espaço de atributos, ou ambos.

- Exemplo 1:  $(s, t) \in \mathcal{A}$  quando  $\|t - s\| \leq r$ , para  $r \geq 1$ .
- Exemplo 2:  $(s, t) \in \mathcal{A}$  quando  $t$  é  $k$ -vizinho mais próximo de  $s$  no espaço de atributos, para  $k \geq 1$ .

Relações de adjacência podem ser definidas no domínio da imagem, espaço de atributos, ou ambos.

- Exemplo 1:  $(s, t) \in \mathcal{A}$  quando  $\|t - s\| \leq r$ , para  $r \geq 1$ .
- Exemplo 2:  $(s, t) \in \mathcal{A}$  quando  $t$  é  $k$ -vizinho mais próximo de  $s$  no espaço de atributos, para  $k \geq 1$ .
- Exemplo 3:  $(s, t) \in \mathcal{A}$  quando  $\|t - s\| \leq r_1$  e  $\|\vec{I}(t) - \vec{I}(s)\| \leq r_2$ , para  $r_1 \geq 1$  e  $r_2 > 0$ .

# Funções de conexidade

- Menor peso de arco — maximização de  $V(t)$ .

$$f_{\min}(\langle t \rangle) = H(t)$$

$$f_{\min}(\pi_s \cdot \langle s, t \rangle) = \min\{f_{\min}(\pi_s), w(s, t)\}$$

ou sua função dual  $f_{\max}$  — minimização de  $V(t)$ .

# Funções de conexidade

- Menor peso de arco — maximização de  $V(t)$ .

$$f_{\min}(\langle t \rangle) = H(t)$$

$$f_{\min}(\pi_s \cdot \langle s, t \rangle) = \min\{f_{\min}(\pi_s), w(s, t)\}$$

ou sua função dual  $f_{\max}$  — minimização de  $V(t)$ .

- Soma dos pesos dos arcos — minimização de  $V(t)$ .

$$f_{\text{sum}}(\langle t \rangle) = H(t)$$

$$f_{\text{sum}}(\pi_s \cdot \langle s, t \rangle) = f_{\text{sum}}(\pi_s) + w(s, t)$$

# Funções de conexidade

- Menor peso de arco — maximização de  $V(t)$ .

$$f_{\min}(\langle t \rangle) = H(t)$$

$$f_{\min}(\pi_s \cdot \langle s, t \rangle) = \min\{f_{\min}(\pi_s), w(s, t)\}$$

ou sua função dual  $f_{\max}$  — minimização de  $V(t)$ .

- Soma dos pesos dos arcos — minimização de  $V(t)$ .

$$f_{\text{sum}}(\langle t \rangle) = H(t)$$

$$f_{\text{sum}}(\pi_s \cdot \langle s, t \rangle) = f_{\text{sum}}(\pi_s) + w(s, t)$$

- Distância Euclideana entre extremos — minimização de  $V(t)$ .

$$f_{\text{euc}}(\langle t \rangle) = \begin{cases} 0 & \text{se } t \in \mathcal{S} \\ +\infty & \text{no caso contrário} \end{cases}$$

$$f_{\text{euc}}(\pi_s \cdot \langle s, t \rangle) = \|t - R(s)\|^2$$

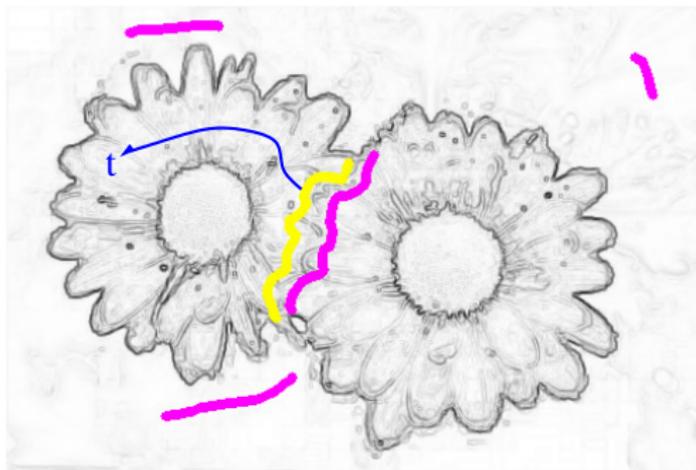
# Segmentação de imagens naturais



Sementes internas  $\mathcal{S}_i$  e externas  $\mathcal{S}_e$  competem entre si pelos pixels mais fortemente conexos e o objeto é definido pelas árvores de caminhos ótimos com raízes em  $\mathcal{S}_i$ . Estas árvores recebem o mesmo rótulo em  $L(t)$ .

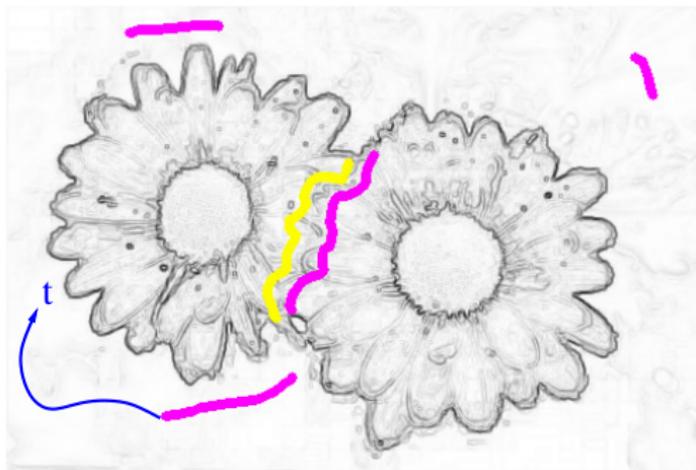


# Segmentação de imagens naturais



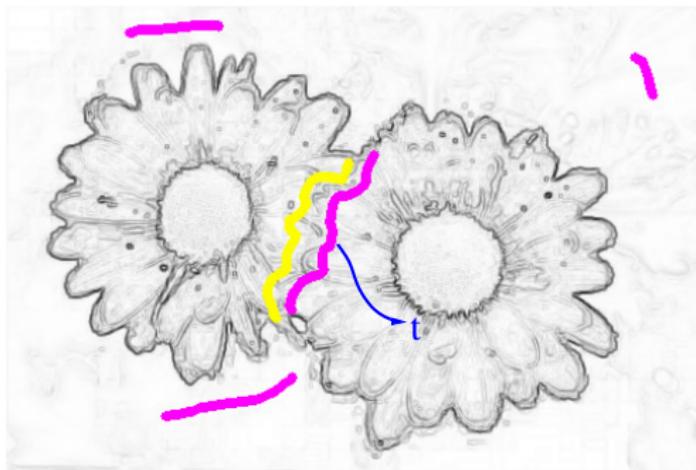
Os caminhos ótimos devem ter suas conexões **enfraquecidas** ao cruzar a fronteira entre objeto e fundo. A **ordenação** tem papel fundamental nos casos de empate entre líderes.

# Segmentação de imagens naturais



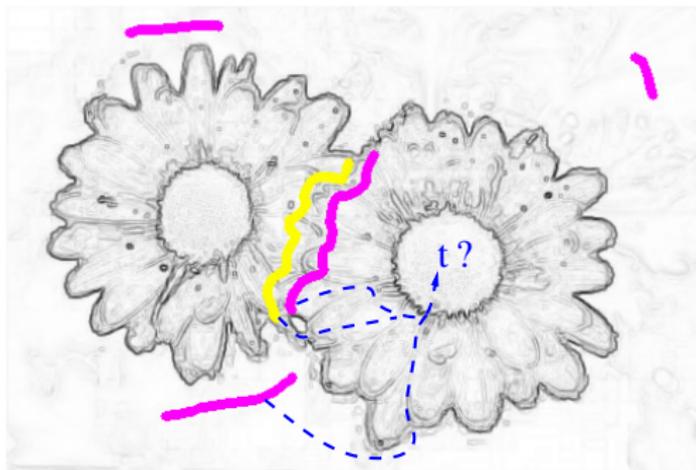
Os caminhos ótimos devem ter suas conexões **enfraquecidas** ao cruzar a fronteira entre objeto e fundo. A **ordenação** tem papel fundamental nos casos de empate entre líderes.

# Segmentação de imagens naturais



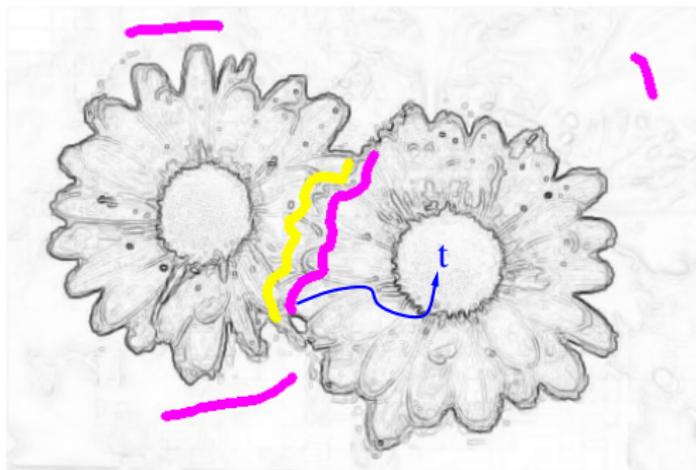
Os caminhos ótimos devem ter suas conexões **enfraqüecidas** ao cruzar a fronteira entre objeto e fundo. A **ordenação** tem papel fundamental nos casos de empate entre líderes.

# Segmentação de imagens naturais



Os caminhos ótimos devem ter suas conexões **enfraquecidas** ao cruzar a fronteira entre objeto e fundo. A **ordenação** tem papel fundamental nos casos de empate entre líderes.

# Segmentação de imagens naturais



Os caminhos ótimos devem ter suas conexões **enfraquecidas** ao cruzar a fronteira entre objeto e fundo. A **ordenação** tem papel fundamental nos casos de empate entre líderes.

- Nós:  $\mathcal{N} = D_I$  é definido por todos os pixels da imagem.

# Segmentação de imagens naturais

- Nós:  $\mathcal{N} = D_I$  é definido por todos os pixels da imagem.
- Arcos:  $t \in \mathcal{A}(s)$  se  $\|t - s\| \leq \sqrt{2}$ .

# Segmentação de imagens naturais

- Nós:  $\mathcal{N} = D_I$  é definido por todos os pixels da imagem.
- Arcos:  $t \in \mathcal{A}(s)$  se  $\|t - s\| \leq \sqrt{2}$ .
- Função de conexidade:

$$f_{\min}(\langle t \rangle) = \begin{cases} w(s, t) & \text{se } t \in (\mathcal{S}_i \cup \mathcal{S}_e) \subset \mathcal{N} \\ -\infty & \text{no caso contrário} \end{cases}$$
$$f_{\min}(\pi_s \cdot \langle s, t \rangle) = \min\{f_{\min}(\pi_s), w(s, t)\},$$

onde  $w(s, t) \geq 0$  é a similaridade entre pixels adjacentes.

# Segmentação de imagens naturais

- Nós:  $\mathcal{N} = D_I$  é definido por todos os pixels da imagem.
- Arcos:  $t \in \mathcal{A}(s)$  se  $\|t - s\| \leq \sqrt{2}$ .
- Função de conexidade:

$$f_{\min}(\langle t \rangle) = \begin{cases} w(s, t) & \text{se } t \in (\mathcal{S}_i \cup \mathcal{S}_e) \subset \mathcal{N} \\ -\infty & \text{no caso contrário} \end{cases}$$
$$f_{\min}(\pi_s \cdot \langle s, t \rangle) = \min\{f_{\min}(\pi_s), w(s, t)\},$$

onde  $w(s, t) \geq 0$  é a similaridade entre pixels adjacentes.

- O resultado é obtido em  $L(t)$  por maximização de  $V(t)$ .

# Segmentação de imagens naturais

Outras IFTs são usadas para calcular a similaridade  $w(s, t)$ .

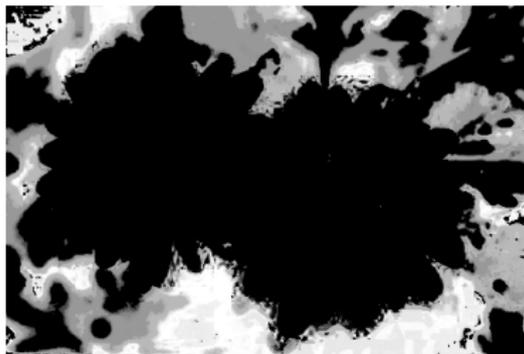


Outras IFTs são usadas para calcular a similaridade  $w(s, t)$ .

- Uma IFT estima a densidade de probabilidade  $\rho(s \setminus \mathcal{S}_i)$ .

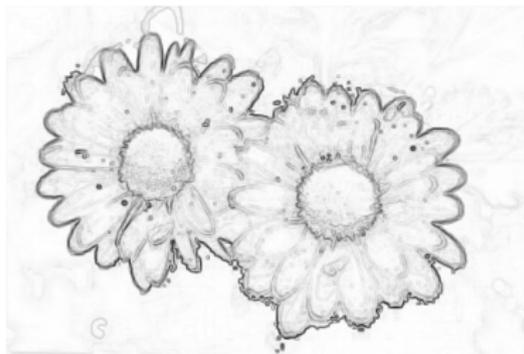


Outras IFTs são usadas para calcular a similaridade  $w(s, t)$ .



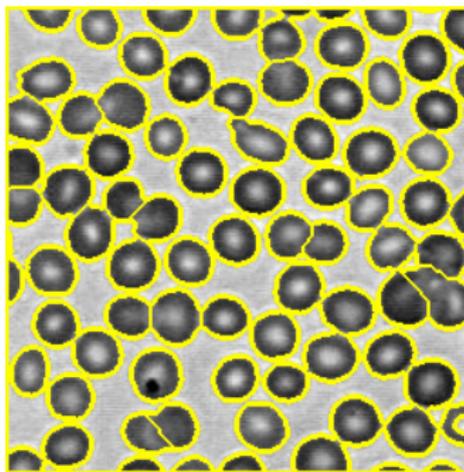
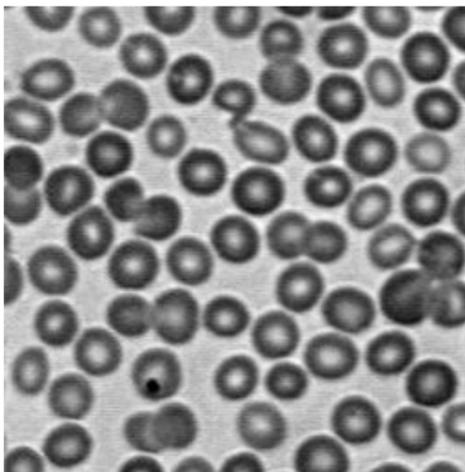
- Uma IFT estima a densidade de probabilidade  $\rho(s \setminus \mathcal{S}_i)$ .
- Outra IFT estima a densidade de probabilidade  $\rho(s \setminus \mathcal{O}_e)$ , onde  $\mathcal{O}_e$  consiste de pixels de  $\mathcal{S}_e$  que são **outliers** em  $\rho(s \setminus \mathcal{S}_i)$ .

Outras IFTs são usadas para calcular a similaridade  $w(s, t)$ .



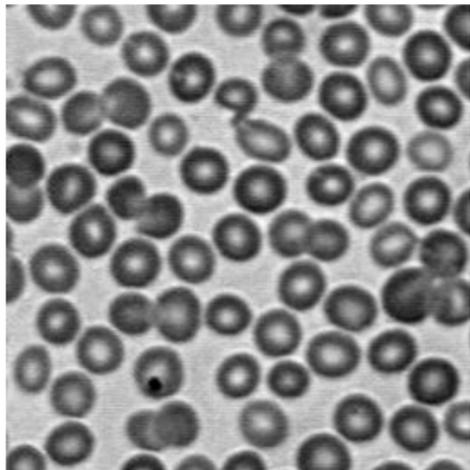
- Uma IFT estima a densidade de probabilidade  $\rho(s \setminus \mathcal{S}_i)$ .
- Outra IFT estima a densidade de probabilidade  $\rho(s \setminus \mathcal{O}_e)$ , onde  $\mathcal{O}_e$  consiste de pixels de  $\mathcal{S}_e$  que são outliers em  $\rho(s \setminus \mathcal{S}_i)$ .
- A similaridade final é obtida dessas densidades e das diferenças locais de cor entre pixels.

# Segmentação de células do sangue



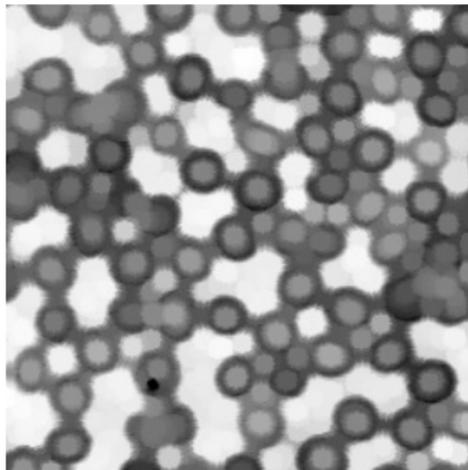
Células do sangue podem ser separadas usando uma sequência de operadores com **quatro IFTs**.

# Segmentação de células do sangue



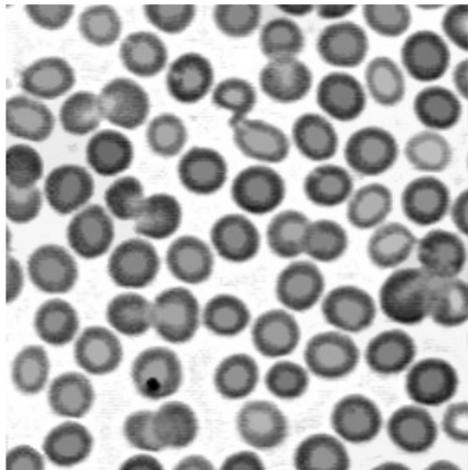
- Imagem original.

# Segmentação de células do sangue



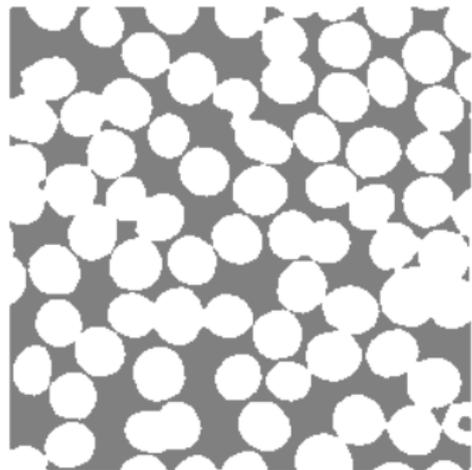
- Imagem original.
- Uma abertura morfológica reduz contraste entre núcleo e citoplasma, mas conecta ainda mais as células.

# Segmentação de células do sangue



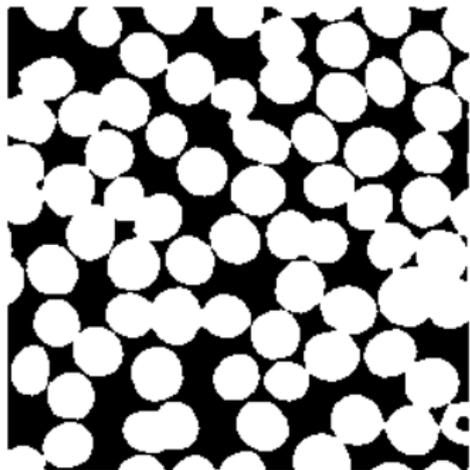
- Imagem original.
- Uma abertura morfológica reduz contraste entre núcleo e citoplasma, mas conecta ainda mais as células.
- Uma primeira IFT obtém em  $V(t)$  a reconstrução da imagem a partir de sua abertura.

# Segmentação de células do sangue



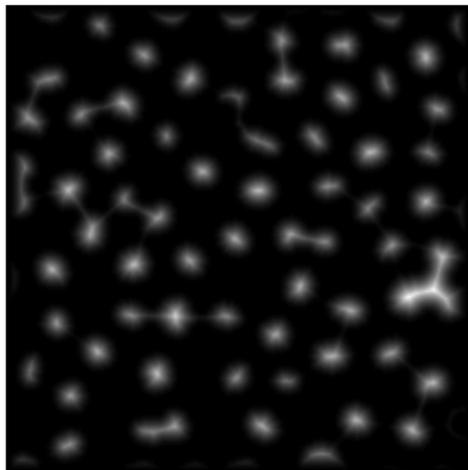
- Uma segunda IFT obtém em  $L(t)$  uma segmentação por *clustering* a partir de amostras da imagem.

# Segmentação de células do sangue



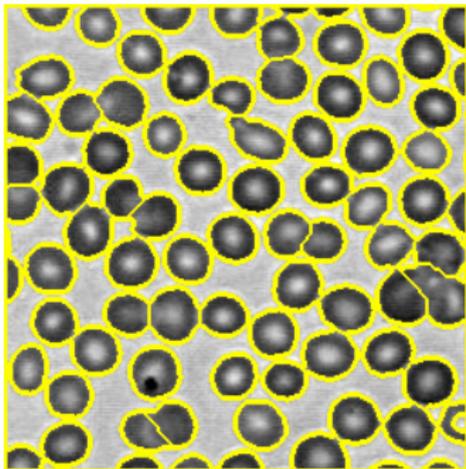
- Uma segunda IFT obtém em  $L(t)$  uma segmentação por *clustering* a partir de amostras da imagem.
- As células são representadas pelo *cluster* mais escuro.

# Segmentação de células do sangue



- Uma segunda IFT obtém em  $L(t)$  uma segmentação por *clustering* a partir de amostras da imagem.
- As células são representadas pelo *cluster* mais escuro.
- Uma terceira IFT obtém em  $V(t)$  as distâncias mínimas em relação aos pixels de borda das células.

# Segmentação de células do sangue



- Uma segunda IFT obtém em  $L(t)$  uma segmentação por *clustering* a partir de amostras da imagem.
- As células são representadas pelo *cluster* mais escuro.
- Uma terceira IFT obtém em  $V(t)$  as distâncias mínimas em relação aos pixels de borda das células.
- Uma quarta IFT obtém em  $L(t)$  uma segmentação com raízes nos máximos do mapa de distâncias.



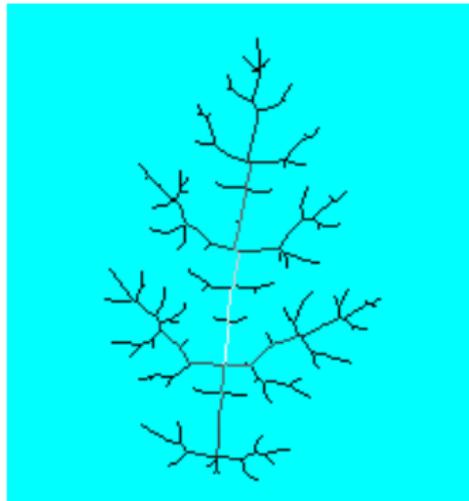
- A máscara de um objeto após a segmentação.



- A máscara de um objeto após a segmentação.
- Uma IFT obtém em  $V(t)$  as distâncias mínimas em relação aos pixels de borda e



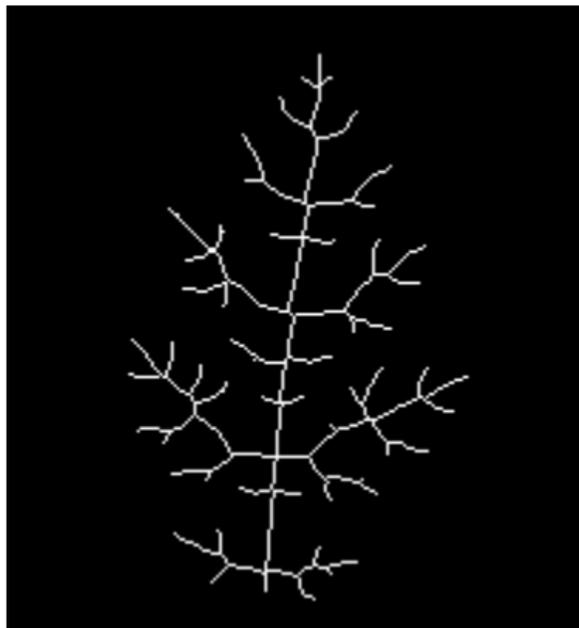
- A máscara de um objeto após a segmentação.
- Uma IFT obtém em  $V(t)$  as distâncias mínimas em relação aos pixels de borda e em  $R(t)$  a propagação das raízes da floresta para todos os pixels.



- A máscara de um objeto após a segmentação.
- Uma IFT obtém em  $V(t)$  as distâncias mínimas em relação aos pixels de borda e em  $R(t)$  a propagação das raízes da floresta para todos os pixels.
- Esqueletos multi-escala são obtidos a partir de  $R(t)$  com base nos **caminhos geodésicos** entre raízes de pixels adjacentes-4.

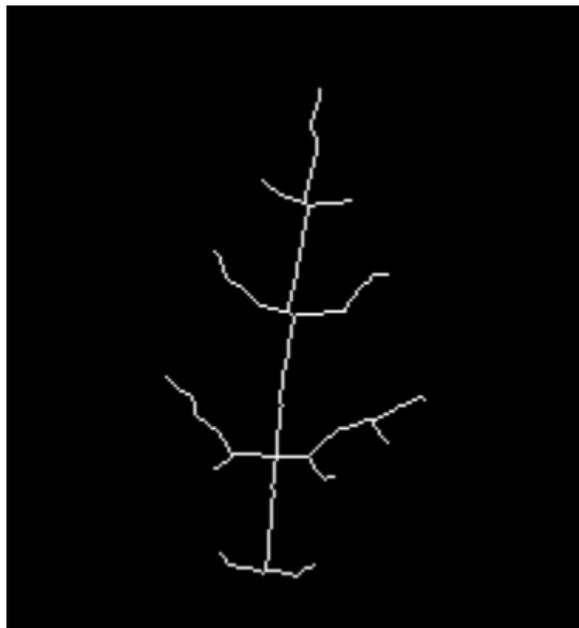
# Esqueletonização

Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.



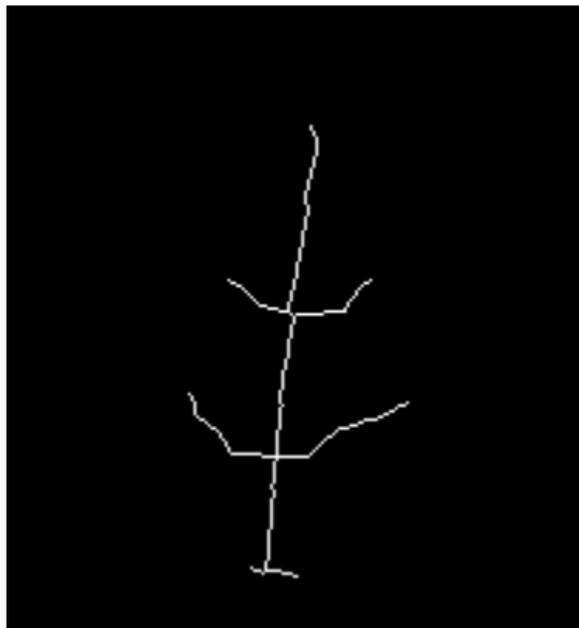
# Esqueletonização

Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.



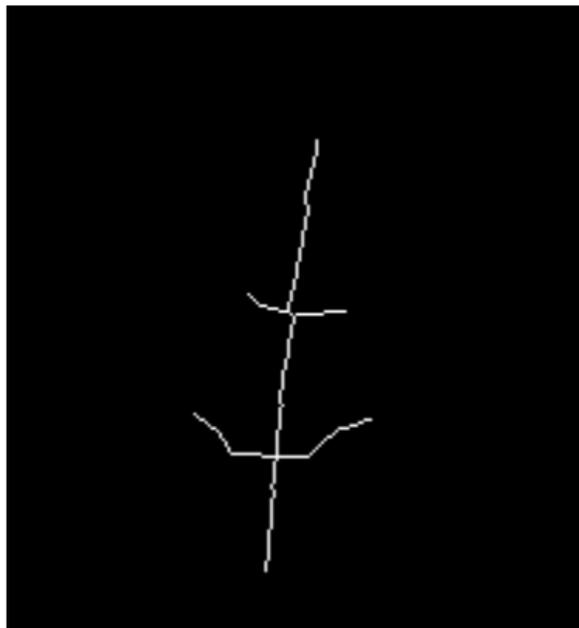
# Esqueletonização

Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.



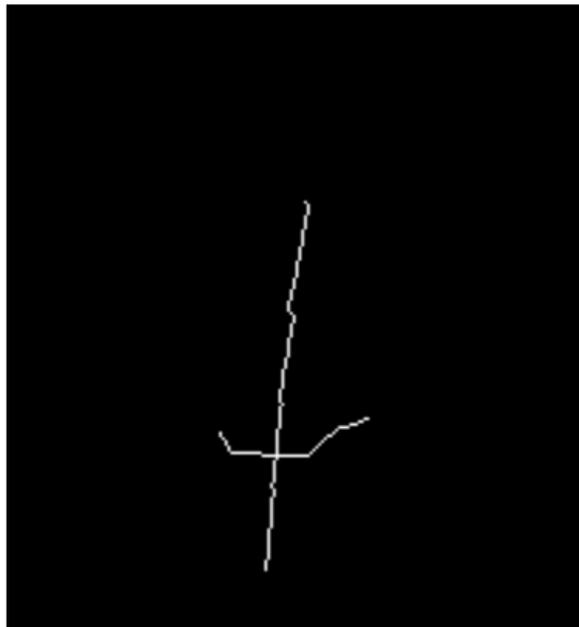
# Esqueletonização

Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.



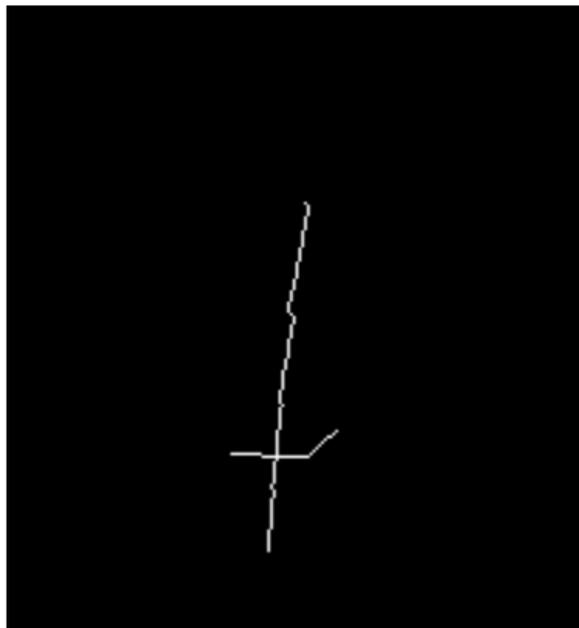
# Esqueletonização

Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.



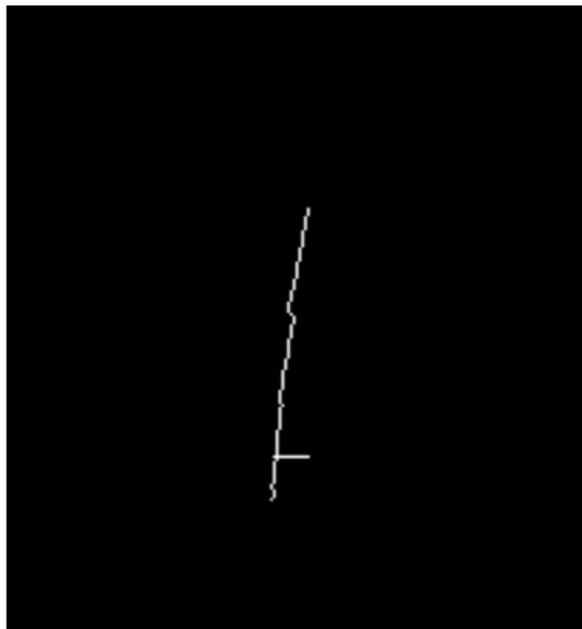
# Esqueletonização

Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.



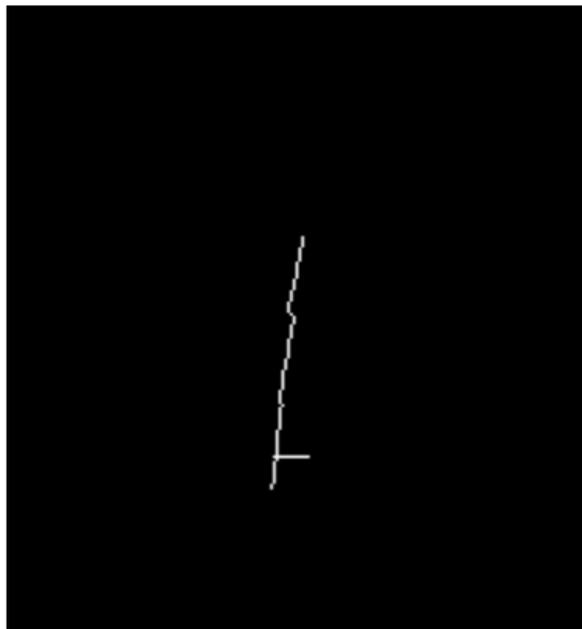
# Esqueletonização

Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.



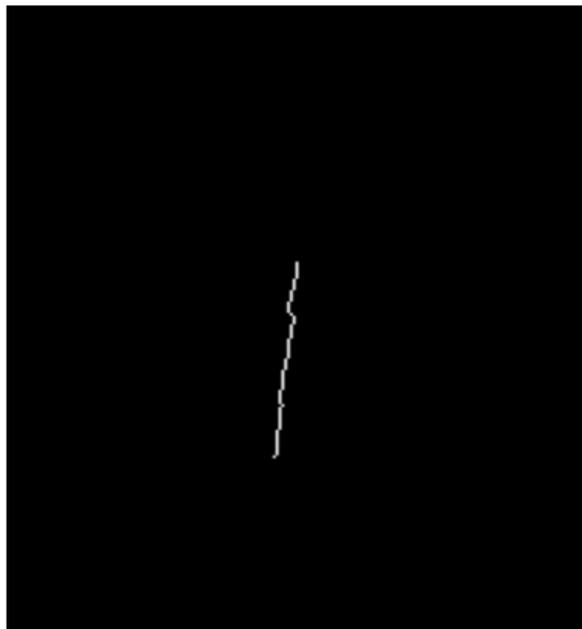
# Esqueletonização

Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.



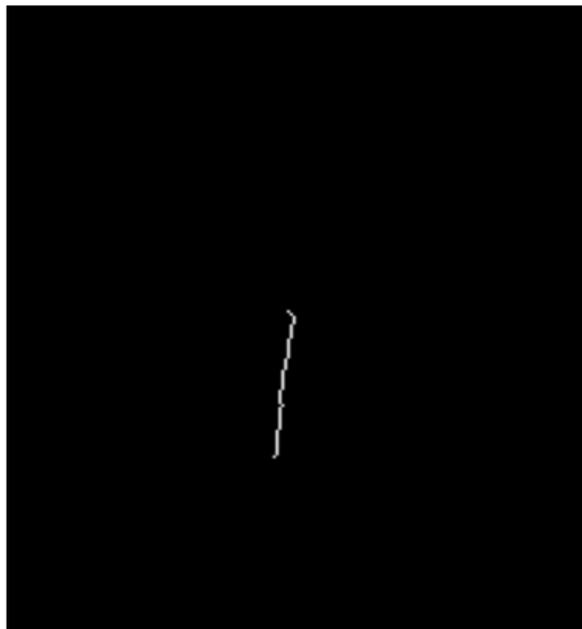
# Esqueletonização

Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.



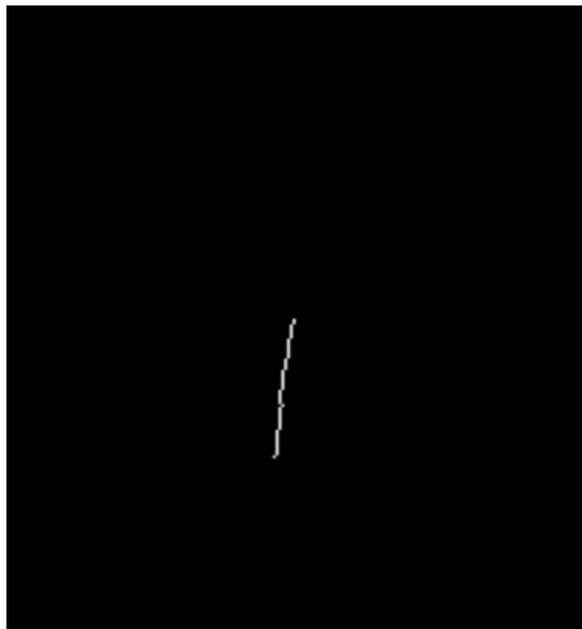
# Esqueletonização

Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.



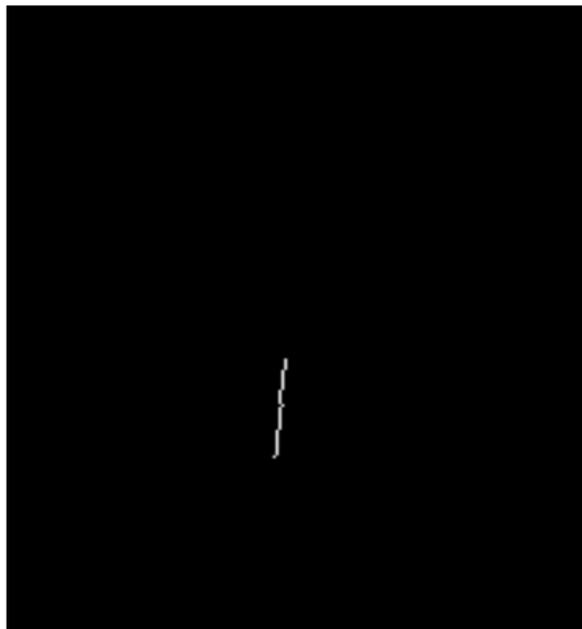
# Esqueletonização

Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.



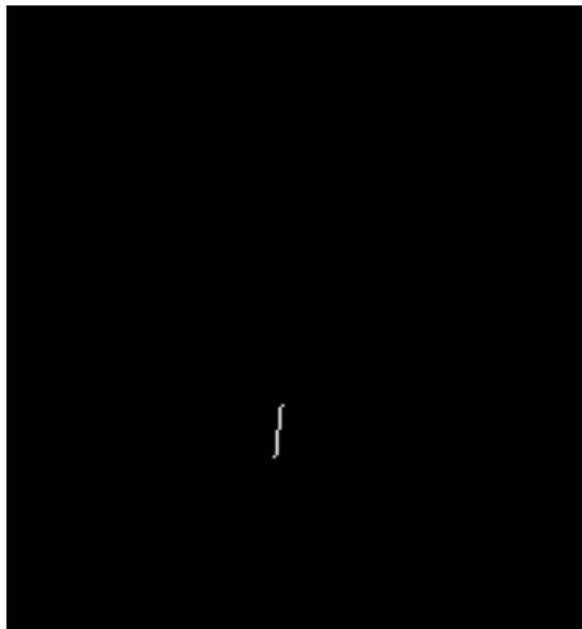
# Esqueletonização

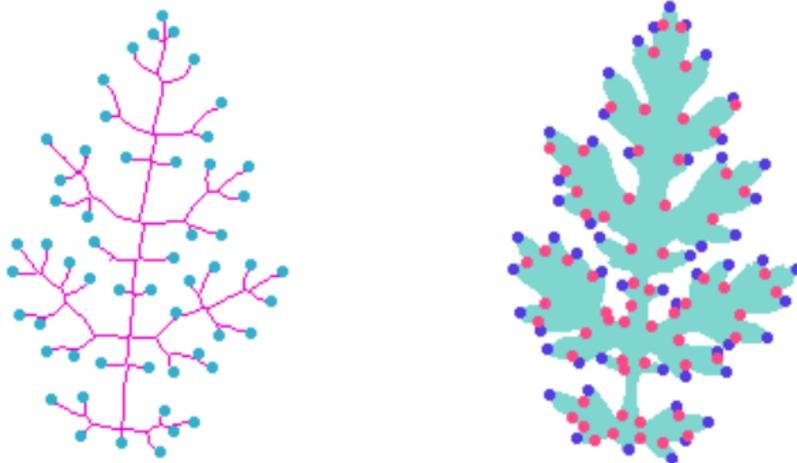
Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.



# Esqueletonização

Os esqueletos são conexos em todas as escalas e com 1 pixel de largura.





IFTs dos esqueletos interno e externo detectam suas saliências e as relacionam com as saliências convexas e côncavas do contorno, respectivamente.

# Implementação linear da fila de prioridade

- O algoritmo da IFT executa em  $O(|\mathcal{A}| + |\mathcal{N}|^2)$ .

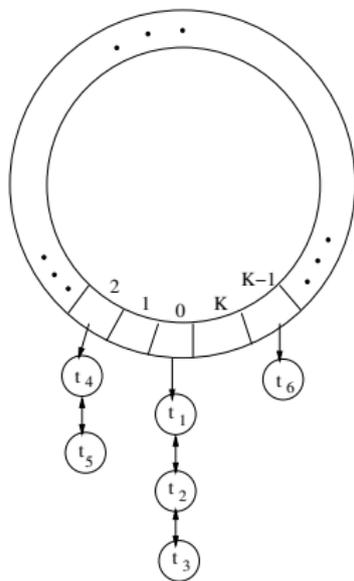
# Implementação linear da fila de prioridade

- O algoritmo da IFT executa em  $O(|\mathcal{A}| + |\mathcal{N}|^2)$ .
- Sua execução cai para  $O(|\mathcal{N}| \log(|\mathcal{N}|))$ , quando  $Q$  é um **heap binário** e o grafo é esparso ( $|\mathcal{A}| \ll |\mathcal{N}|^2$ ).

# Implementação linear da fila de prioridade

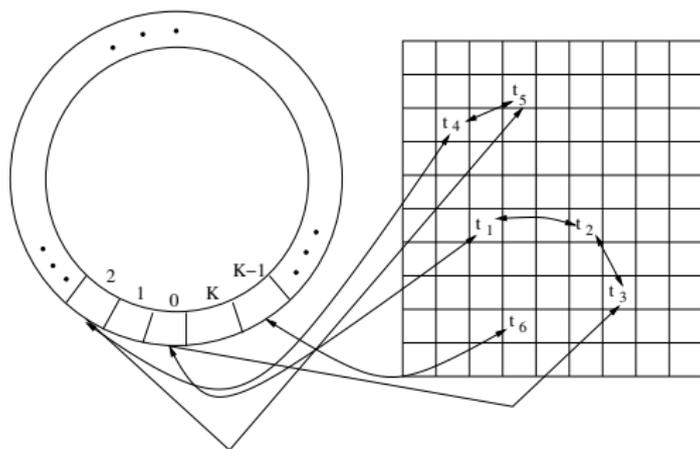
- O algoritmo da IFT executa em  $O(|\mathcal{A}| + |\mathcal{N}|^2)$ .
- Sua execução cai para  $O(|\mathcal{N}| \log(|\mathcal{N}|))$ , quando  $Q$  é um **heap binário** e o grafo é esparso ( $|\mathcal{A}| \ll |\mathcal{N}|^2$ ).
- O tempo de execução cai ainda para  $O(|\mathcal{N}|)$ , quando  $f(\pi_s) - f(\pi_s \cdot \langle s, t \rangle) \in [0..K]$ ,  $K \ll |\mathcal{N}|$ , são inteiros e o grafo é esparso.

# Fila de prioridade linear



Nós  $t$  são inseridos no *bucket*  $V(t) \% (K + 1)$  (left), formando  $K + 1$  listas (right). A propriedade  $f(\pi_s) - f(\pi_s \cdot \langle s, t \rangle) \in [0..K]$  garante que nós com valores diferentes nunca são inseridos no mesmo *bucket*.

# Fila de prioridade linear



Nós  $t$  são inseridos no *bucket*  $V(t) \% (K + 1)$  (left), formando  $K + 1$  listas (right). A propriedade  $f(\pi_s) - f(\pi_s \cdot \langle s, t \rangle) \in [0..K]$  garante que nós com valores diferentes nunca são inseridos no mesmo *bucket*.

- A IFT **unifica** diversos métodos com maior eficiência, facilitando o entendimento sobre eles, e favorecendo implementações em *hardware*.

# Comentários conclusivos

- A IFT **unifica** diversos métodos com maior eficiência, facilitando o entendimento sobre eles, e favorecendo implementações em *hardware*.
- Seu algoritmo geral executa em **tempo linear** na maioria dos casos e alguns variantes executam em **tempo sublinear**.

- A IFT **unifica** diversos métodos com maior eficiência, facilitando o entendimento sobre eles, e favorecendo implementações em *hardware*.
- Seu algoritmo geral executa em **tempo linear** na maioria dos casos e alguns variantes executam em **tempo sublinear**.
- Sua extensão usando grafos com outros tipos de nós (sinais, imagens, objetos) tem possibilitado novas técnicas de **aprendizado de máquina** e **reconhecimento de padrões**.

Suas aplicações estão nas principais etapas da análise de imagem:

Suas aplicações estão nas principais etapas da análise de imagem:

- **Filtragem**: reconstruções morfológicas, dilatações e erosões em tempo sublinear.

Suas aplicações estão nas principais etapas da análise de imagem:

- **Filtragem**: reconstruções morfológicas, dilatações e erosões em tempo sublinear.
- **Segmentação**: interativa, automática, baseada em borda, região, e modelos de objeto.

Suas aplicações estão nas principais etapas da análise de imagem:

- **Filtragem**: reconstruções morfológicas, dilatações e erosões em tempo sublinear.
- **Segmentação**: interativa, automática, baseada em borda, região, e modelos de objeto.
- **Representação**: transformadas de distâncias e geodésicas, esqueletos, escala tensorial, e saliências.

Suas aplicações estão nas principais etapas da análise de imagem:

- **Filtragem**: reconstruções morfológicas, dilatações e erosões em tempo sublinear.
- **Segmentação**: interativa, automática, baseada em borda, região, e modelos de objeto.
- **Representação**: transformadas de distâncias e geodésicas, esqueletos, escala tensorial, e saliências.
- **Descrição**: diversos descritores de forma baseados nas representações acima.

Suas aplicações estão nas principais etapas da análise de imagem:

- **Filtragem**: reconstruções morfológicas, dilatações e erosões em tempo sublinear.
- **Segmentação**: interativa, automática, baseada em borda, região, e modelos de objeto.
- **Representação**: transformadas de distâncias e geodésicas, esqueletos, escala tensorial, e saliências.
- **Descrição**: diversos descritores de forma baseados nas representações acima.
- **Classificação**: supervisionada e não-supervisionada de padrões (biomedicina, biometria, geologia, engenharia elétrica, metalurgia, sensoriamento remoto, etc).