Fundamentals of Image Processing (part IV)

Alexandre Xavier Falcão

Institute of Computing - UNICAMP

afalcao@ic.unicamp.br

Image transformations based on optimum connectivity

- Some image transformations are based on optimum connectivity (optimum paths).

Image transformations based on optimum connectivity

- Some image transformations are based on optimum connectivity (optimum paths).
- These transformations can be reduced to an optimum-path forest in the image graph followed by a local processing of its attributes.

Image transformations based on optimum connectivity

- Some image transformations are based on optimum connectivity (optimum paths).
- These transformations can be reduced to an optimum-path forest in the image graph followed by a local processing of its attributes.
- The framework to design such transformations is named Image Foresting Transform [1].

Image transformations based on optimum connectivity

- Some image transformations are based on optimum connectivity (optimum paths).

- These transformations can be reduced to an optimum-path forest in the image graph followed by a local processing of its attributes.

- The framework to design such transformations is named Image Foresting Transform [1].

- Applications involve segmentation [2, 3, 4, 5, 6, 7, 8], clustering [9, 10], classification [11, 12, 13, 14], distance transforms [15, 16], morphological reconstructions [17], multiscale skeletons [15, 16], shape saliences [18], etc.

Agenda

- Image Foresting Transform (IFT).

Agenda

- Image Foresting Transform (IFT).
- General IFT algorithm.

Agenda

- Image Foresting Transform (IFT).
- General IFT algorithm.
- Examples and main properties.

Image Foresting Transform

For a given image graph $(\mathcal{N}, \mathcal{A}, \mathrm{I})$, $\mathcal{N} \subseteq D_I$,

- a connectivity (path-cost) function $f(\pi_p)$ assigns a cost to any path in the set Π of paths in the graph.

For a given image graph $(\mathcal{N}, \mathcal{A}, \mathrm{I})$, $\mathcal{N} \subseteq D_I$,

- a connectivity (path-cost) function $f(\pi_p)$ assigns a cost to any path in the set Π of paths in the graph.
- The set Π contains trivial paths $\pi_p = \langle p \rangle$ and paths $\pi_p \cdot \langle p, q \rangle$ that represent the extension of π_p by an arc $(p, q) \in \mathcal{A}$.

Image Foresting Transform

For a given image graph $(\mathcal{N}, \mathcal{A}, \mathrm{I})$, $\mathcal{N} \subseteq D_I$,

- a connectivity (path-cost) function $f(\pi_p)$ assigns a cost to any path in the set Π of paths in the graph.
- The set Π contains trivial paths $\pi_p = \langle p \rangle$ and paths $\pi_p \cdot \langle p, q \rangle$ that represent the extension of π_p by an arc $(p, q) \in \mathcal{A}$.
- The IFT algorithm minimizes a path-cost map V,

$$V(p) \quad = \quad \min_{\pi_p \in \Pi} \{f(\pi_p)\},$$

for all $p \in \mathcal{N}$, irrespective to its root node.

- The IFT algorithm can always output a rooted spanning forest in a predecessor map P (see lecture 4).

Image Foresting Transform

- The IFT algorithm can always output a rooted spanning forest in a predecessor map P (see lecture 4).
- The map P will be optimal (i.e., an optimum-path forest) whenever f satisfies the conditions stated in [19].

- The IFT algorithm can always output a rooted spanning forest in a predecessor map P (see lecture 4).
- The map P will be optimal (i.e., an optimum-path forest) whenever f satisfies the conditions stated in [19].
- However, there are applications for the case P is just a rooted spanning forest [3, 4, 16].

Image Foresting Transform

- The IFT algorithm can always output a rooted spanning forest in a predecessor map P (see lecture 4).
- The map P will be optimal (i.e., an optimum-path forest) whenever f satisfies the conditions stated in [19].
- However, there are applications for the case P is just a rooted spanning forest [3, 4, 16].
- The image transformations derive from attributes of the forest: paths, costs, root labels, etc.

The IFT algorithm

- It starts from all nodes $p \in \mathcal{N}$ as trivial paths with values $V(p) \leftarrow f(\pi_p)$ in a priority queue Q. The roots will derive from the minima of this initial path-cost map.

The IFT algorithm

- It starts from all nodes $p \in \mathcal{N}$ as trivial paths with values $V(p) \leftarrow f(\pi_p)$ in a priority queue Q. The roots will derive from the minima of this initial path-cost map.

- By removing the nodes p in a non-decreasing order of path values from Q, it verifies for each adjacent $q \in \mathcal{A}(p)$

$$\begin{aligned} \textit{if} \quad f(\pi_p \cdot \langle p, q \rangle) < V(q), & \text{ then} \\ \pi_q \leftarrow \pi_p \cdot \langle p, q \rangle & \text{ and} \\ V(q) \leftarrow f(\pi_p \cdot \langle p, q \rangle). & \end{aligned}$$

The IFT algorithm

- It starts from all nodes $p \in \mathcal{N}$ as trivial paths with values $V(p) \leftarrow f(\pi_p)$ in a priority queue Q. The roots will derive from the minima of this initial path-cost map.
- By removing the nodes p in a non-decreasing order of path values from Q, it verifies for each adjacent $q \in \mathcal{A}(p)$

$$\begin{array}{ll} \textit{if} & f(\pi_p \cdot \langle p,q\rangle) < V(q), \text{ then} \\ & \pi_q \leftarrow \pi_p \cdot \langle p,q\rangle \text{ and} \\ & V(q) \leftarrow f(\pi_p \cdot \langle p,q\rangle). \end{array}$$

- It stops when Q is empty and the optimum paths π^*_p for all $p \in \mathcal{N}$ can be retrieved from P.

Input: Image graph $(\mathcal{N}, \mathcal{A}, \mathrm{I})$ and connectivity function f.
Output: Cost map V and predecessor map P.

1 For each $q \in \mathcal{N}$, set $V(q) \leftarrow f(\langle q \rangle)$ and $P(q) \leftarrow nil$, and insert q in Q.
2 While $Q \neq \emptyset$ do
3 Remove from Q the node $p = arg \min_{q \in Q}\{V(q)\}$.
4 For each $q \in \mathcal{A}(p)$, $q \in Q$, do
5 If $V(q) > f(\pi_p \cdot \langle p, q \rangle)$, then
6 update $V(q) \leftarrow f(\pi_p \cdot \langle p, q \rangle)$ and $P(q) \leftarrow p$.

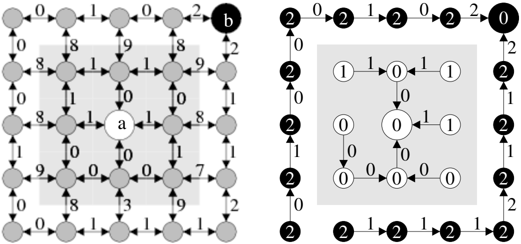
Consider, for example, a max-arc-weight function f_{max} that forces optimum paths to start in a seed set $\mathcal{S} = \{a, b\} \subset \mathcal{N} = D_I$.

$$
\begin{aligned}
f_{\text{max}}(\langle p \rangle) &= \begin{cases} 0 & \text{if } p \in \mathcal{S}, \\ +\infty & \text{otherwise.} \end{cases} \\
f_{\text{max}}(\pi_p \cdot \langle p, q \rangle) &= \max\{f_{\text{max}}(\pi_p), \|\mathbf{I}(q), \mathbf{I}(p)\|_2\}.
\end{aligned}
$$

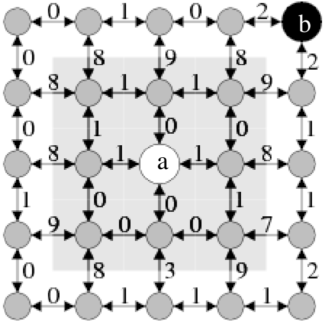
Consider, for example, a max-arc-weight function f_{max} that forces optimum paths to start in a seed set $\mathcal{S} = \{a, b\} \subset \mathcal{N} = D_I$.

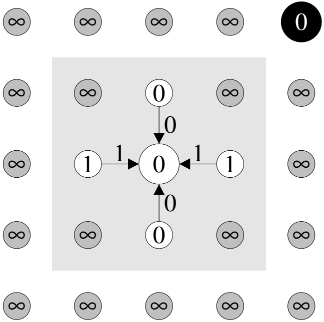
$$f_{max}(\langle p \rangle) = \begin{cases} 0 & \text{if } p \in \mathcal{S}, \\ +\infty & \text{otherwise.} \end{cases}$$
$$f_{max}(\pi_p \cdot \langle p, q \rangle) = \max\{f_{max}(\pi_p), \|\mathbf{I}(q), \mathbf{I}(p)\|_2\}.$$

For the graph on the left, the output is the forest on the right.

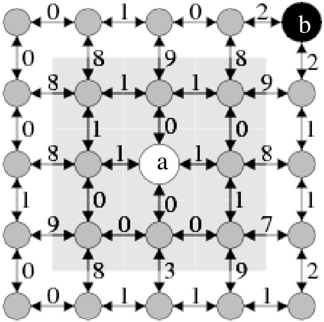


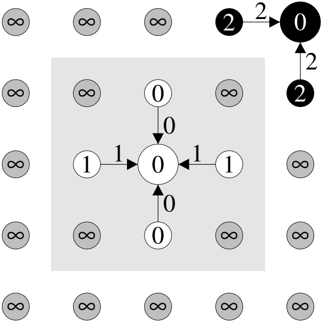
From iteration 1 to 5, iteration 12, 20, and 25.





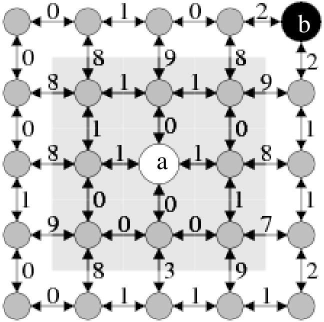
From iteration 1 to 5, iteration 12, 20, and 25.

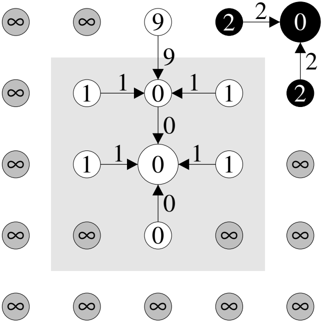




Example of optimum path propagation

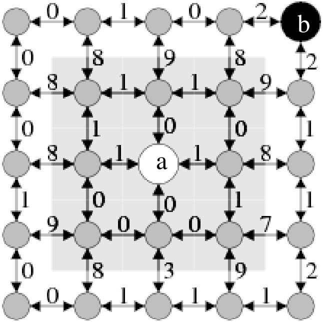
From iteration 1 to 5, iteration 12, 20, and 25.

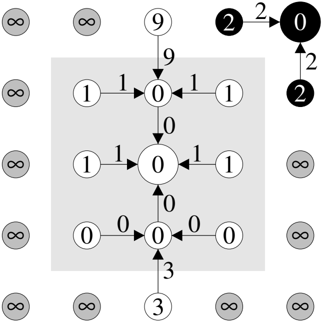




Example of optimum path propagation

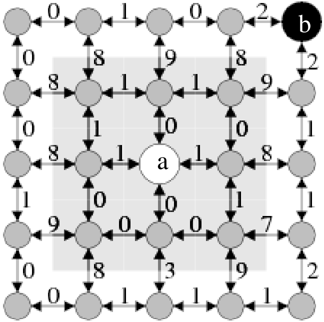
From iteration 1 to 5, iteration 12, 20, and 25.

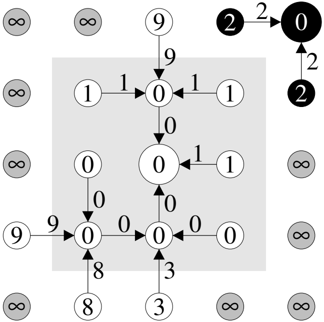




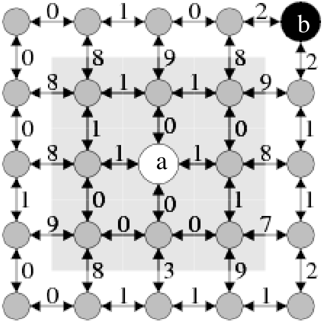
Example of optimum path propagation

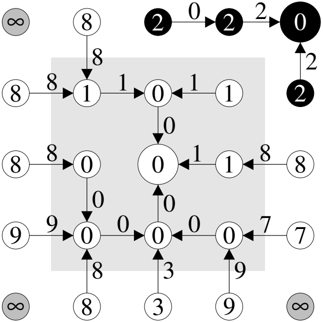
From iteration 1 to 5, iteration 12, 20, and 25.





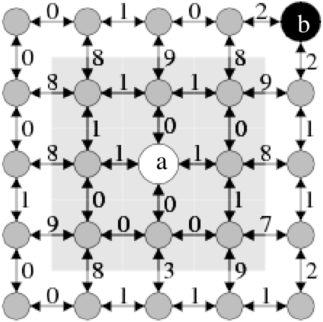
From iteration 1 to 5, iteration 12, 20, and 25.

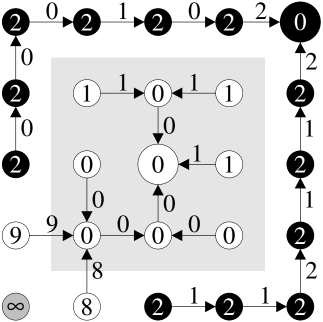




Example of optimum path propagation

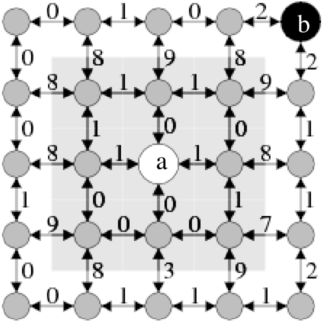
From iteration 1 to 5, iteration 12, 20, and 25.

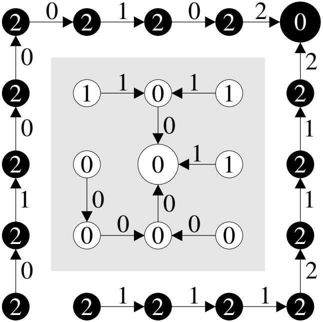




Example of optimum path propagation

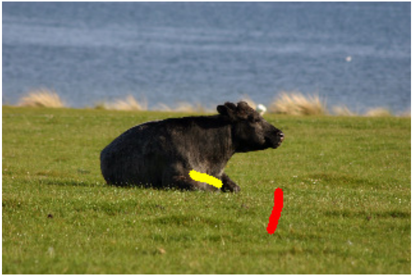
From iteration 1 to 5, iteration 12, 20, and 25.





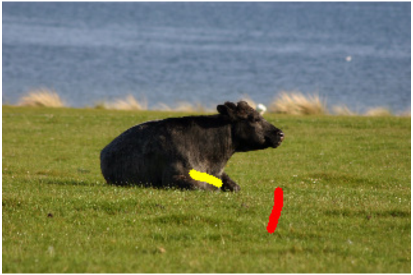
Example of seed-based segmentation

Using the same function $f_{\max}$, the roots may be forced to start from internal and external markers (location) for object delineation.



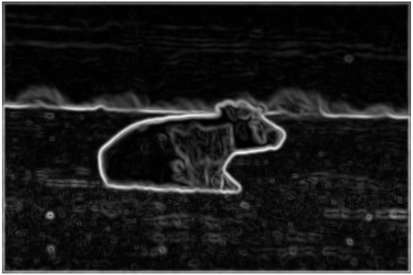
The object is defined by the optimum-path forest rooted at its internal markers.

Example of seed-based segmentation



- Image with internal and external markers.

Example of seed-based segmentation

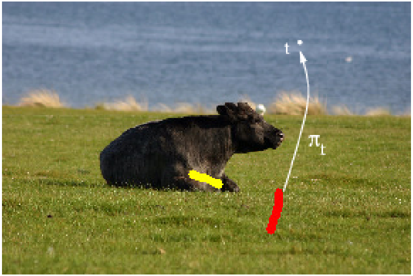


- Image with internal and external markers.
- Arc-weight image.

Example of seed-based segmentation

- Image with internal and external markers.
- Arc-weight image.
- Optimum-paths to foreground pixels.

Example of seed-based segmentation



- Image with internal and external markers.
- Arc-weight image.
- Optimum-paths to foreground pixels.
- Optimum-paths to background pixels.

Example of seed-based segmentation

- Image with internal and external markers.
- Arc-weight image.
- Optimum-paths to foreground pixels.
- Optimum-paths to background pixels.
- Segmentation result.

Exercise

In the last example, let seeds $s \in \mathcal{S}$ have labels $\lambda(s) \in \{0, 1\}$ to indicate background and object seeds, respectively.

Exercise

In the last example, let seeds $s \in \mathcal{S}$ have labels $\lambda(s) \in \{0, 1\}$ to indicate background and object seeds, respectively.

- Change the general IFT algorithm to receive $(\mathcal{S}, \lambda)$ and return in a label map L with the label $L(p) \in \{0, 1\}$ of background and object pixels.

Exercise

In the last example, let seeds $s \in \mathcal{S}$ have labels $\lambda(s) \in \{0, 1\}$ to indicate background and object seeds, respectively.

- Change the general IFT algorithm to receive $(\mathcal{S}, \lambda)$ and return in a label map L with the label $L(p) \in \{0, 1\}$ of background and object pixels.
- Change it now to output a root map R that assigns to every $p \in \mathcal{N}$ the root node $R(p)$ in the optium path π^*_p.

Main properties

- When p is removed from Q,
  - If $P(p) = nil$, then p is a root of the forest and can be saved in a root set $\mathcal{R}$.

Main properties

- When p is removed from Q,
  - If $P(p) = nil$, then p is a root of the forest and can be saved in a root set $\mathcal{R}$.
  - The cost $V(p)$ and path π_p^* in P are optimal.

Main properties

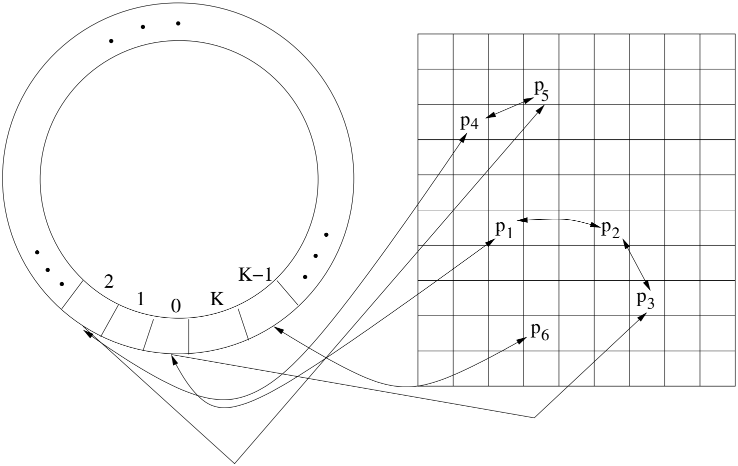
- When p is removed from Q,
  - If $P(p) = nil$, then p is a root of the forest and can be saved in a root set $\mathcal{R}$.
  - The cost $V(p)$ and path π^*_p in P are optimal.
- The algorithm takes $O(|\mathcal{A}| + |\mathcal{N}|^2)$.

Main properties

- When p is removed from Q,
  - If $P(p) = nil$, then p is a root of the forest and can be saved in a root set $\mathcal{R}$.
  - The cost $V(p)$ and path π_p^* in P are optimal.
- The algorithm takes $O(|\mathcal{A}| + |\mathcal{N}|^2)$.
- If $|\mathcal{A}| \ll |\mathcal{N}|^2$ and Q is a binary heap, it takes $O(|\mathcal{N}| \log |\mathcal{N}|)$.

Main properties

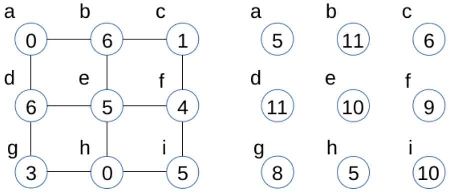
- When p is removed from Q,
  - If $P(p) = nil$, then p is a root of the forest and can be saved in a root set $\mathcal{R}$.
  - The cost $V(p)$ and path π_p^* in P are optimal.
- The algorithm takes $O(|\mathcal{A}| + |\mathcal{N}|^2)$.
- If $|\mathcal{A}| \ll |\mathcal{N}|^2$ and Q is a binary heap, it takes $O(|\mathcal{N}| \log |\mathcal{N}|)$.
- If $|\mathcal{A}| \ll |\mathcal{N}|^2$ and $f(\pi_p \cdot \langle p, q \rangle) - f(\pi_p) \in [0, K]$, $K \ll |\mathcal{N}|$, it can take $O(|\mathcal{N}|)$ using bucket sort.



Nodes p are inserted in and removed from bucket $V(p)$ mod $K+1$ in $O(1)$.

Exercise

Let $\hat{I} = (D_I, I)$ be the image on the left, where the numbers indicate $I(t)$, $\mathcal{N} = D_I$ and $\mathcal{A}$ is defined by the four neighbors.



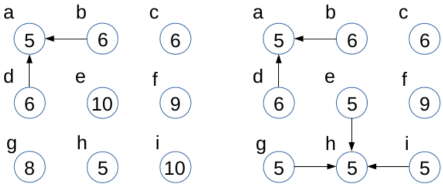
On the right, the trivial forest of the connectivity function

$$
\begin{aligned}
f(\langle t \rangle) &= I(t) + 5, \\
f(\pi_s \cdot \langle s, t \rangle) &= \max\{f(\pi_s), I(t)\}.
\end{aligned}
$$

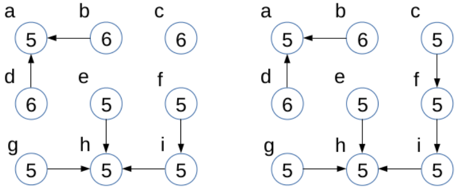
Can you tell which nodes will be in the root set $\mathcal{R}$? Change the algorithm to propagate a distinct label per optimum-path tree.

Subsequent iterations

After the first iteration (left) and second (right).



After the third (left) and from 4–9 (right).



Let's see Watershed.ipynb in notebooks.tar.gz

[1] A.X. Falcão, J. Stolfi, and R.A. Lotufo.
The image foresting transform: Theory, algorithms, and applications.
IEEE Trans. on Pattern Analysis and Machine Intelligence, 26(1):19–29, 2004.

[2] T.V. Spina, P.A.V. Miranda, and A.X. Falcão.
Hybrid approaches for interactive image segmentation using the live markers paradigm.
IEEE Transactions on Image Processing, 23(12):5756–5769, 2014.

[3] P.A.V. Miranda and L.A.C. Mansilla.
Oriented image foresting transform segmentation by seed competition.
IEEE Transactions on Image Processing, 23(1):389–398, Jan 2014.

[4] J.E. Vargas-Muñoz, A.S. Chowdhury, E.B. Alexandre, F.L. Galvão, P.A.V. Miranda, and A.X. Falcão.
An iterative spanning forest framework for superpixel segmentation.
IEEE Transactions on Image Processing, 28(7):3477–3489, 2019.

[5] S.B. Martins, J. Bragantini, A.X. Falcão, and C.L. Yasuda.
An adaptive probabilistic atlas for anomalous brain segmentation in mr images.
Medical Physics, 46(11):4940–4950, 2019.

[6] A.M. Sousa, S.B. Martins, A.X. Falcão, F. Reis, E. Bagatin, and K. Irion.
Altis: A fast and automatic lung and trachea ct-image segmentation method.
Medical Physics, 46(11):4970–4982, 2019.

[7] M.A.T. Condori, F.A.M. Cappabianco, A.X. Falcão, and P.A.V. Miranda.
An extension of the differential image foresting transform and its application to superpixel generation.
Journal of Visual Communication and Image Representation, 71:102748, 2020.

[8] F.L. Galvão, S.J.F. Guimarães, and A.X. Falcão.
Image segmentation using dense and sparse hierarchies of superpixels.
Pattern Recognition, 108:107532, 2020.

[9] L.M. Rocha, F.A.M. Cappabianco, and A.X. Falcão.
Data clustering as an optimum-path forest problem with applications in image analysis.
Int. J. Imaging Syst. Technol., 19(2):50–68, June 2009.

[10] F.A.M. Cappabianco, A.X. Falcão, C.L. Yasuda, and J.K. Udupa.
Brain tissue mr-image segmentation via optimum-path forest clustering.

Computer Vision and Image Understanding, 116(10):1047 – 1059, 2012.

[11] J.P. Papa, A.X. Falcão, V.H.C. de Albuquerque, and J.M.R.S. Tavares.
Efficient supervised optimum-path forest classification for large datasets.
Pattern Recognition, 45(1):512 – 520, 2012.

[12] W.P. Amorim, A.X. Falcão, J.P. Papa, and M.H. Carvalho.
Improving semi-supervised learning through optimum connectivity.
Pattern Recognition, 60:72 – 85, 2016.

[13] J.P. Papa, S.E.N. Fernandes, and A.X. Falcão.
Optimum-path forest based on k-connectivity: Theory and applications.
Pattern Recognition Letters, 87:117 – 126, 2017.
Advances in Graph-based Pattern Recognition.

[14] W.P. Amorim, A.X. Falcão, and J.P. Papa.
Multi-label semi-supervised classification through optimum-path forest.
Information Sciences, 465:86 – 104, 2018.

[15] A.X. Falcão, L.F. Costa, and B.S. Cunha.
Multiscale skeletons by image foresting transform and its application to neuromorphometry.
Pattern Recognition, 35(7):1571 – 1582, 2002.

[16] A.X. Falcão, C. Feng, J. Kustra, and A.C. Telea.
Chapter 2 - multiscale 2d medial axes and 3d surface skeletons by the image foresting transform.
In P.K. Saha, G. Borgefors, and G.S. di Baja, editors, *Skeletonization*, pages 43 – 70. Academic Press, 2017.

[17] A.X. Falcão, B.S. Cunha, and R.A. Lotufo.
Design of connected operators using the image foresting transform.
In Milan Sonka and Kenneth M. Hanson, editors, *Medical Imaging 2001: Image Processing*, volume 4322, pages 468 – 479. International Society for Optics and Photonics, SPIE, 2001.

[18] R. da S. Torres and A.X. Falcão.
Contour salience descriptors for effective image retrieval and analysis.
Image and Vision Computing, 25(1):3 – 13, 2007.

[19] K.C. Ciesielski, A.X. Falcão, and P.A.V. Miranda.
Path-value functions for which dijkstra's algorithm returns optimal mapping.
J Math Imaging Vis, 60:1025–1036, 2018.