The Image Foresting Transform: Recent Advances and Perspectives

Alexandre Xavier Falcão
afalcao@ic.unicamp.br

Laboratory of Visual Informatics (LIV), Institute of Computing (IC), University of Campinas (UNICAMP)
A brief history

This work stems from the interactive boundary delineation problem by optimum paths in an image graph (15 years ago).
A brief history

This work stems from the interactive boundary delineation problem by optimum paths in an image graph (15 years ago).

The most popular solution was the live-wire segmentation (show video 1).
A brief history

This work stems from the interactive boundary delineation problem by optimum paths in an image graph (15 years ago).

The most popular solution was the live-wire segmentation (*show video 1*).

The optimum path requires the computation of an optimum-path tree rooted at its starting seed point.
A brief history

For a set of seed points, similar algorithm provides an optimum-path forest by seed competition.
A brief history

For a set of seed points, similar algorithm provides an optimum-path forest by seed competition.

So, can we use this more general formulation and reduce other image problems to a local processing of the forest’s attributes?
A brief history

For a set of seed points, similar algorithm provides an optimum-path forest by seed competition.

So, can we use this more general formulation and reduce other image problems to a local processing of the forest’s attributes?

Yes, the method is named Image Foresting Transform (IFT).
What is the IFT?

The IFT is a tool for the design of image processing and analysis operators by choice of an **adjacency relation** and a **connectivity function** between image elements (pixels, vertices, edges, regions, or contour segments) [1].
Contributors

Several colleagues and students who appear in the list of references at the end of this talk. The pdf file of the talk will be available for downloading at

www.ic.unicamp.br/~afalcao.
Outline

- A short review on the IFT.
Outline

- A short review on the IFT.
- How to use it in image processing and analysis (a few examples).
Outline

- A short review on the IFT.
- How to use it in image processing and analysis (a few examples).
- Its recent extension to the feature space.
Outline

- A short review on the IFT.
- How to use it in image processing and analysis (a few examples).
- Its recent extension to the feature space.
- Some recent applications.
Outline

- A short review on the IFT.
- How to use it in image processing and analysis (a few examples).
- Its recent extension to the feature space.
- Some recent applications.
- Open problems, on-going works and their perspectives.
Motivation

**Unification:** Several image operators are derived from a general algorithm. This favors
Unification: Several image operators are derived from a general algorithm. This favors hardware-based implementations [2, 3],
Motivation

**Unification:** Several image operators are derived from a general algorithm. This favors

- hardware-based implementations [2, 3],
- to understand the relation among some image operators [4, 5, 6], and
Unification: Several image operators are derived from a general algorithm. This favors hardware-based implementations [2, 3], to understand the relation among some image operators [4, 5, 6], and possible extensions [7, 8, 9, 10, 11].
Motivation

**Efficiency:** Most image operators can be implemented in linear time and further optimizations are possible with differential [12] and parallel [13] computation, and for some specific applications [14, 15].
Motivation

- **Efficiency**: Most image operators can be implemented in linear time and further optimizations are possible with differential [12] and parallel [13] computation, and for some specific applications [14, 15].

- **Simplicity**: The image operators are reduced to the choice of two parameters in the IFT algorithm and a local processing of its output.
What kind of problems can be solved?

Problems which are either directly or indirectly related to an optimal image (set) partition.

- **Distance transforms and related operators:**
  - Multiscale skeletonization [16], fractal dimensions [17], shape filtering [14, 16], shape description by saliences and tensor scale [17, 18, 19, 20], geodesic paths [1].
What kind of problems can be solved?

- Filtering and Segmentation: Morphological reconstructions [4] and image segmentation based on watershed transforms [5, 12, 7, 21], pixel clustering [10], contour tracking [22, 15, 23], tree pruning [9, 24], graph-cut measures [8], and fuzzy-connected components [25, 26].

- Pattern recognition: Fast supervised and unsupervised learning approaches [10, 11, 27, 28].
Images as graphs

The image is interpreted as a graph whose nodes are the pixels and the arcs are defined by an adjacency relation $\mathcal{A} : (s, t) \in \mathcal{A}$ if $\|t - s\| \leq d_i$.

(a) $d_i = 1$ and (b) $d_i = 2$. 
A path $\pi_t$ is a sequence of adjacent nodes with terminus at some node $t$ and $\pi_t = \langle t \rangle$ is said trivial.
Paths in the graph

- A path $\pi_t$ is a sequence of adjacent nodes with terminus at some node $t$ and $\pi_t = \langle t \rangle$ is said trivial.

- The predecessor $P(s)$ of each node $s \in \pi_t$ leads to a root node $R(t)$ and $P(R(t)) = nil$. 
Optimum paths

A connectivity function $f(\pi_t)$ assigns a value to any path $\pi_t$. 
A connectivity function \( f(\pi_t) \) assigns a value to any path \( \pi_t \).

A path \( \pi_t \) is optimum if \( f(\pi_t) \leq f(\tau_t) \) for any other \( \tau_t \), irrespective to its root.
Optimum paths

- A connectivity function $f(\pi_t)$ assigns a value to any path $\pi_t$.
- A path $\pi_t$ is optimum if $f(\pi_t) \leq f(\tau_t)$ for any other $\tau_t$, irrespective to its root.
- The dual definition $f(\pi_t) \geq f(\tau_t)$ is also valid.
The IFT

- From an initial path-value map \( V_0(t) = f(\langle t \rangle) \) where all nodes are trivial paths.
The IFT

- From an initial path-value map $V_0(t) = f(\langle t \rangle)$ where all nodes are trivial paths.

- The IFT computes a minimal (maximal) path-value map $V(t) = \min_{\forall \pi_t} \{ f(\pi_t) \}$ for every node $t$. 
The IFT

From an initial path-value map \( V_0(t) = f(\langle t \rangle) \) where all nodes are trivial paths.

The IFT computes a minimal (maximal) path-value map \( V(t) = \min_{\forall \pi_t} \{ f(\pi_t) \} \) for every node \( t \).

The result is an optimum-path forest \( P \) — an acyclic graph where all paths are optimum according to \( f \).
The first roots are identified as the global minima (maxima) of $V_0$. 

A.X. Falcão, DSP 2009 – p. 15
The first roots are identified as the global minima (maxima) of $V_0$.

They may conquer their adjacent nodes by offering them better paths.
The IFT computation

- The first roots are identified as the global minima (maxima) of $V_0$.

- They may conquer their adjacent nodes by offering them better paths.

- The process continues from the adjacent nodes in a non-decreasing (non-increasing) order of path value.

$$\text{if } f(\pi_s \cdot \langle s, t \rangle) < f(\pi_t) \text{ then } \pi_t \leftarrow \pi_s \cdot \langle s, t \rangle.$$
Segmentation by seed competition (IFTSC)
An optimum-path forest is computed by assigning each pixel to its most strongly connected marker pixel.
Segmentation by seed competition (IFTSC)

An optimum-path forest is computed by assigning each pixel to its most strongly connected marker pixel.

The object is defined by the union of optimum-path trees rooted in the object markers (show videos 2 and 3).
Segmentation by seed competition (IFTSC)

An image-graph where arc weights $w(s, t)$ indicate the dissimilarity between adjacent nodes and we wish to compute $V$ such that the roots of $P$ belong to a seed set $S = \{(3, 3), (5, 1)\}$.
We need to define a path-initialization rule and a path-extension rule. The path-value map $V$ is minimized for function $f_1$.

$$f_1(\langle t \rangle) = \begin{cases} 0 & \text{if } t \in S \\ +\infty & \text{otherwise} \end{cases}$$

$$f_1(\pi_s \cdot \langle s, t \rangle) = \max\{f_1(\pi_s), w(s, t)\}$$
Path propagation
Path propagation
Path propagation
Path propagation
Path propagation
Path propagation

after 12 iterations.
Path propagation

after 20 iterations.
Path propagation
Path propagation
Path propagation

after 25 iterations.
Information propagation

The IFT requires a priority queue $Q$ for path propagation (modified Dijkstra’s algorithm).
Information propagation

- The IFT requires a priority queue \( Q \) for path propagation (modified Dijkstra’s algorithm).
- A root \( s \) is an optimum trivial path \( \langle s \rangle \), such that \( P(s) = nil \) when \( s \) is removed from \( Q \).
Information propagation

- The IFT requires a **priority queue** $Q$ for path propagation (modified Dijkstra’s algorithm).

- A root $s$ is an optimum trivial path $\langle s \rangle$, such that $P(s) = nil$ when $s$ is removed from $Q$.

- It can propagate other informations to each node: a root label $[12, 21]$, its propagation order $[26]$, a graph-cut measure $[8]$, etc.
Information propagation

The IFT requires a priority queue $Q$ for path propagation (modified Dijkstra’s algorithm).

A root $s$ is an optimum trivial path $\langle s \rangle$, such that $P(s) = \text{nil}$ when $s$ is removed from $Q$.

It can propagate other informations to each node: a root label $[12, 21]$, its propagation order $[26]$, a graph-cut measure $[8]$, etc.

The operators result from a local processing of one or more of these informations.
For every node $t$, there must exist at least one optimum path $\pi_t$ which is either trivial or has the form $\pi_s \cdot \langle s, t \rangle$ where:

1. $f(\pi_s) \leq f(\pi_t)$.
2. $\pi_s$ is optimum.
3. For any optimum path $\tau_s$, $f(\tau_s \cdot \langle s, t \rangle) = f(\pi_t)$.

These conditions are applied to only optimum paths.
Euclidean distance transform (EDT)

The EDT of a pixel set $S$ uses Euclidean adjacency $A$ and minimizes $V$ for $f_2$ [16].

$$f_2(\langle t \rangle) = \begin{cases} 0 & \text{if } t \in S \\ +\infty & \text{otherwise} \end{cases}$$

$$f_2(\pi_s \cdot \langle s, t \rangle) = \|t - R(s)\|$$

where $R(s)$ is the root of $s$. 

A.X. Falcão, DSP 2009 – p. 22
A consecutive integer number can be assigned to each contour pixel in $S$ and propagated to the rest of the image.
Multiscale skeletons

A difference image is obtained from the labeled image. Increasing thresholds create more simplified one-pixel-wide and connected skeletons.
Skeletons of multiples contours

The method is easily extended to incorporate the SKIZ in the case of multiple contours.
Shape saliences

Skeleton saliences are detected from the aperture angles of their influence zones within a small dilation radius, leading to contour saliences [17].
Live-wire-on-the-fly solves the boundary delineation problem with real time response to the user's actions, independently of the image size [15].
its path function

The minimization of $V$ for path function $f_3$ allows longer and oriented segments.

$$f_3(\langle t \rangle) = \begin{cases} 0 & \text{if } t \in S \\ +\infty & \text{otherwise} \end{cases}$$

$$f_3(\pi_s \cdot \langle s, t \rangle) = \begin{cases} f_3(\pi_s) + w^a(s, t) & \text{if } L(s, t) < R(s, t) \\ f_3(\pi_s) + K & \text{otherwise} \end{cases}$$

for $0 \leq w^a(s, t) \leq K$, $a > 0$, $L(s, t)$ and $R(s, t)$ are intensities on the left and right sides of arc $(s, t)$. 

A.X. Falcão, DSP 2009 – p. 28
Multiple object segmentation

Multiple objects are segmented by using a gradient image $I$ and initial path values $V_0(t) = H(t) > I(t)$ [4, 7].

$H$ is the closing of $I$ plus 1, using adjacency radius $d_i = 2.5$. The IFT uses adjacency radius $d_i = 3.5$ and path function $f_4$ for segmentation.
A label map $L$ is obtained by minimizing $V$ for $f_4$.

$$f_4(\langle t \rangle) = \begin{cases} I(t) & \text{if } t \in \mathcal{R} \\ H(t) & \text{otherwise} \end{cases}$$

$$f_4(\pi_s \cdot \langle s, t \rangle) = \max\{f_4(\pi_s), I(t)\}$$

where $\mathcal{R}$ is the root set whose pixels come from some minima of $H$. Initially, $f_4(\langle t \rangle) = H(t)$ for all $t$, then the roots are identified on-the-fly: if $P(t) = \text{nil}$ when $t$ is removed from $Q$ then $t \in \mathcal{R}$ and $f_4(\langle t \rangle) \leftarrow I(t)$ for label propagation.
Extension to the feature space

The graph nodes may be images, shapes, regions or other entities of a dataset.
Extension to the feature space

The graph nodes may be images, shapes, regions or other entities of a dataset.

Each node $s$ is represented by a feature vector $\vec{v}(s)$ of $n$ dimensions.
Extension to the feature space

- The graph nodes may be images, shapes, regions or other entities of a dataset.
- Each node $s$ is represented by a feature vector $\vec{v}(s)$ of $n$ dimensions.
- The similarity between adjacent nodes is given by a distance function $d(s, t)$ (e.g., $d(s, t) = \|\vec{v}(t) - \vec{v}(s)\|$).
Extension to the feature space

- The graph nodes may be images, shapes, regions or other entities of a dataset.
- Each node \( s \) is represented by a feature vector \( \vec{v}(s) \) of \( n \) dimensions.
- The similarity between adjacent nodes is given by a distance function \( d(s, t) \) (e.g., \( d(s, t) = ||\vec{v}(t) - \vec{v}(s)|| \)).
- The graph is usually weighted on the arcs by \( d(s, t) \) and may also be weighted on the nodes by a probability density function \( \rho(s) \).
How do we create arcs in this case?

The adjacency relation may be complete, may consider maximum distances around the nodes or may take into account the \( k \)-nearest nodes (\( k \)-nn graph). For example:

\[
A_k : \quad (s, t) \in A_k \quad (\text{or } t \in A_k(s)) \quad \text{if } t \text{ is } k \text{ nearest neighbor of } s \text{ in the feature space.}
\]

The best value of \( k \) can be computed by several different ways, depending on the learning approach: supervised or unsupervised.
Clustering by OPF

The graph is weighted on the arcs $(s, t) \in A_k$ by $d(s, t)$ and on the nodes by the pdf $\rho(s)$ [10].

$$\rho(s) = \frac{1}{\sqrt{2\pi\sigma^2}|A_k(s)|} \sum_{\forall t \in A_k(s)} \exp\left(\frac{-d^2(s, t)}{2\sigma^2}\right)$$

where $\sigma = \frac{d_f}{3}$ and $d_f = \max_{\forall(s,t)\in A_k} \{d(s, t)\}$. The best value of $k$ is found as the one whose clustering produces a minimum normalized cut in the $k$-nn graph.
We maximize $V$ for path function $f_5$.

\[
\begin{align*}
    f_5(\langle t \rangle) &= \begin{cases} 
    \rho(t) & \text{if } t \in \mathcal{R} \\
    H(t) & \text{otherwise}
    \end{cases} \\
    f_5(\pi_s \cdot \langle s, t \rangle) &= \min\{f_5(\pi_s), \rho(t)\}
\end{align*}
\]

where $\rho(t) > H(t)$, $H(t)$ can be the result of some anti-extensive operation on $\rho$, and $\mathcal{R}$ is the root set found on-the-fly ($P(t) = \text{nil}$ when $t$ is removed from $Q$).
Each cluster in (c) is an optimum-path tree rooted at one maximum of the pdf shown in (b). Irrelevant clusters can be eliminated by choice of $H$, as shown in (d). The method is shape independent.
Some recent applications

- A graph-based framework for fast interactive segmentation of natural images [29].
Some recent applications

- A graph-based framework for fast interactive segmentation of natural images [29].
- Texture image recognition [30].
Some recent applications

- A graph-based framework for fast interactive segmentation of natural images [29].
- Texture image recognition [30].
- Spoken emotion recognition [31].
Some recent applications

- A graph-based framework for fast interactive segmentation of natural images [29].
- Texture image recognition [30].
- Spoken emotion recognition [31].
- Fingerprint and face recognition [32, 33, 34].
Some recent applications

- A graph-based framework for fast interactive segmentation of natural images [29].
- Texture image recognition [30].
- Spoken emotion recognition [31].
- Fingerprint and face recognition [32, 33, 34].
- Remote sensing: land use classification [35] and rainfall estimation [36].
Some recent applications

- Object tracking in video [37].
Some recent applications

- **Object tracking** in video [37].
- **Cloud bank model** for automatic structure segmentation [38, 39].
Some recent applications

- **Object tracking** in video [37].
- **Cloud bank model** for automatic structure segmentation [38, 39].
- Automatic brain tissue classification in MR images [40, 10] using clustering by OPF.
Some recent applications

- **Object tracking** in video [37].
- **Cloud bank model** for automatic structure segmentation [38, 39].
- Automatic brain tissue classification in MR images [40, 10] using clustering by OPF.
- Oropharyngeal dysphagia identification and laryngeal pathology detection [41, 42].
Object tracking in video

1. Starting from the interactive segmentation by IFTSC in a first frame.
Object tracking in video

1. Starting from the interactive segmentation by IFTSC in a first frame.

2. Motion estimation is constrained within the object mask.
Object tracking in video

1. Starting from the interactive segmentation by IFTSC in a first frame.

2. Motion estimation is constrained within the object mask.

3. The average motion is applied to an external marker around the object.
Object tracking in video

1. Starting from the interactive segmentation by IFTSC in a first frame.
2. Motion estimation is constrained within the object mask.
3. The average motion is applied to an external marker around the object.
4. New internal and external markers are selected for automatic segmentation by IFTSC in the next frame, and the process is repeated from 2 (show video 4).
Cloud bank segmentation

The method is divided in two steps: training and execution.
Cloud bank segmentation

- The method is divided in two steps: training and execution.
- In training, a discrete shape model is created to guide the delineation process.
Cloud bank segmentation

The method is divided in two steps: training and execution.

In training, a discrete shape model is created to guide the delineation process.

In execution, delineation by IFT provides a candidate object for each position of the shape model on the image and a criterion function assigns a score to that candidate.
Cloud bank segmentation

- The method is divided in two steps: training and execution.
- In training, a discrete shape model is created to guide the delineation process.
- In execution, delineation by IFT provides a candidate object for each position of the shape model on the image and a criterion function assigns a score to that candidate.
- Segmentation is defined by the object with the highest score.
Cloud bank training

A set of training objects is provided to capture the shape variations of a given structure.
Cloud bank training

- A set of training objects is provided to capture the shape variations of a given structure.
- The objects are translated to a common reference point and their 0/1 values are averaged.
Cloud bank training

- A set of training objects is provided to capture the shape variations of a given structure.
- The objects are translated to a common reference point and their 0/1 values are averaged.
- The idea is to avoid registration in training and execution.
Cloud bank training

The training creates a cloud image, with an uncertainty region, which should include the object’s boundary for some image location.
To encode shape variations with narrower uncertainty regions, the objects are grouped by maximal cliques on a complete graph weighted by their similarity values, forming a cloud bank.
Cloud execution: brain segmentation

For brain segmentation, the image is first aligned by the mid-sagittal plane (MSP) and the search is performed in multiple scales.
For brain segmentation, the image is first aligned by the mid-sagital plane (MSP) and the search is performed in multiple scales. Each cloud moves over the MSP and, for each position, delineation is done by IFTSC inside the uncertainty region.
Cloud execution: brain segmentation

For brain segmentation, the image is first aligned by the mid-sagittal plane (MSP) and the search is performed in multiple scales.

Each cloud moves over the MSP and, for each position, delineation is done by IFTSC inside the uncertainty region.

The mean graph cut is used to assign scores to the candidate objects.
Cloud execution: brain segmentation

For brain segmentation, the image is first aligned by the mid-sagittal plane (MSP) and the search is performed in multiple scales.

Each cloud moves over the MSP and, for each position, delineation is done by IFTSC inside the uncertainty region.

The mean graph cut is used to assign scores to the candidate objects.

Execution takes about 1 min for 4 clouds in a modern PC.
2D example in brain segmentation

(a) Input image. (b) The recognition score for all positions. (c) The best location. (d) Final segmentation.
3D results in brain segmentation

<table>
<thead>
<tr>
<th>Cloud Bank:</th>
<th>Interactive:</th>
<th>Cloud Bank:</th>
<th>Interactive:</th>
<th>Cloud Bank:</th>
<th>Interactive:</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Cloud Bank Image" /></td>
<td><img src="image2" alt="Interactive Image" /></td>
<td><img src="image3" alt="Cloud Bank Image" /></td>
<td><img src="image4" alt="Interactive Image" /></td>
<td><img src="image5" alt="Cloud Bank Image" /></td>
<td><img src="image6" alt="Interactive Image" /></td>
</tr>
<tr>
<td><img src="image7" alt="Cloud Bank Image" /></td>
<td><img src="image8" alt="Interactive Image" /></td>
<td><img src="image9" alt="Cloud Bank Image" /></td>
<td><img src="image10" alt="Interactive Image" /></td>
<td><img src="image11" alt="Cloud Bank Image" /></td>
<td><img src="image12" alt="Interactive Image" /></td>
</tr>
<tr>
<td><img src="image13" alt="Cloud Bank Image" /></td>
<td><img src="image14" alt="Interactive Image" /></td>
<td><img src="image15" alt="Cloud Bank Image" /></td>
<td><img src="image16" alt="Interactive Image" /></td>
<td><img src="image17" alt="Cloud Bank Image" /></td>
<td><img src="image18" alt="Interactive Image" /></td>
</tr>
</tbody>
</table>
3D results in brain segmentation

Evaluation on a dataset with 40 MR images of normal subjects from both genders and ages from 9 to 49 years (Dice similarities).

<table>
<thead>
<tr>
<th>Object</th>
<th>Mean</th>
<th>S.dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>brain (GM+WM)</td>
<td>97.37</td>
<td>0.46</td>
</tr>
<tr>
<td>hemispheres</td>
<td>97.59</td>
<td>0.48</td>
</tr>
<tr>
<td>cerebellum</td>
<td>94.97</td>
<td>0.84</td>
</tr>
<tr>
<td>right hemisphere</td>
<td>97.16</td>
<td>0.48</td>
</tr>
<tr>
<td>left hemisphere</td>
<td>97.05</td>
<td>0.55</td>
</tr>
</tbody>
</table>
Some open problems

- Parallel IFT in multi-core platforms.
Some open problems

- Parallel IFT in multi-core platforms.
- Cloud-based image segmentation for different applications and using other IFT-based operators for delineation.
Some open problems

- Parallel IFT in multi-core platforms.
- Cloud-based image segmentation for different applications and using other IFT-based operators for delineation.
- OPF-based learning algorithms for large datasets and their applications in segmentation, content-based image retrieval, super-resolution and image compression problems.
On-going works and their perspectives

An IFT-based hardware architecture for image processing.
On-going works and their perspectives

- An IFT-based hardware architecture for image processing.
- IFT-based tracking of multiple objects with occlusion.
On-going works and their perspectives

- An IFT-based hardware architecture for image processing.
- IFT-based tracking of multiple objects with occlusion.
- MR-image segmentation of brain structures by cloud bank.
On-going works and their perspectives

- An IFT-based hardware architecture for image processing.
- IFT-based tracking of multiple objects with occlusion.
- MR-image segmentation of brain structures by cloud bank.
- Brain tissue classification of segmented structures by OPF clustering.
Acknowledgments

Thank you!!!

UNICAMP, FAPESP and CNPq.
References


49-2


[40] F. Cappabianco, A.X. Falcão, and L.M. Rocha. Clustering by optimum path forest and its application to automatic GM/WM
