Multi-Object Segmentation by Hierarchical Layered Oriented Image Foresting Transform

Leissi M. Castañeda Leon, Paulo A. V. Miranda
Institute of Mathematics and Statistics,
University of São Paulo,
São Paulo, SP, Brazil.
Email: leissicl@ime.usp.br, pmiranda@vision.ime.usp.br

***Abstract*—This paper introduces a new method for multi-object segmentation in images, named as Hierarchical Layered Oriented Image Foresting Transform (HLOIFT). As input, we have an image, a tree of relations between image objects, with the individual high-level priors of each object coded in its nodes, and the objects' seeds. Each node of the tree defines a weighted digraph, named as layer. The layers are then integrated by the geometric interactions, such as inclusion and exclusion relations, extracted from the given tree into a unique weighted digraph, named as hierarchical layered digraph. A single energy optimization is performed in the hierarchical layered weighted digraph by Oriented Image Foresting Transform (OIFT) leading to globally optimal results satisfying all the high-level priors. We evaluate our framework in the multi-object segmentation of medical and synthetic images, obtaining results comparable to the state-of-the-art methods, but with low computational complexity. Compared to multi-object segmentation by min-cut/max-flow algorithm, our approach is less restrictive, leading to globally optimal results in more general scenarios.**

I. Introduction

Image segmentation is the task of partitioning an image into objects of interest by assigning distinct labels to their composing pixels. Hence, all the pixels sharing the same label in the image form an object. This task is not only important from the computer vision point of view, but also has impact in different research areas such as Ecology, Medicine, Neurology, Artificial Intelligence [1]–[3]. For example, in medical imaging, image segmentation can be used to isolate the objects corresponding to different organs in Magnetic Resonance Images (MRI), helping to analyze their forms, volumes and textures for the diagnostic of pathologies [4].

Currently, interactive graph-based methods are commonly used in image segmentation tasks, where the image is modeled as a connected graph. The image segmentation task is posed as a graph partition problem, formulated as an optimization problem, satisfying some optimality constraints [5]–[9].

In the context of multiple object segmentation, each object may present its own distinctive features, requiring different high-level priors, such as connectivity [10], [11], shape [12]–[15] and boundary polarity [16], [17]. In order to obtain a good segmentation result, the segmentation method must attend all the individual priors of each object, as well capture the contextual or structural relations between them. However, many existing methods do not include any form of structural information or only include high-level priors for single object segmentation [10], [11], [13], [16], [17]. Consequently, they may be inappropriate in the context of multiples objects.

The priori information about topology has been shown to improve the results in different applications of graph-based methods [18]. In the context of image segmentation, most of the methods that include structural information are based on graph-cut optimization and are performed by a min-cut/max-flow algorithm [19]–[21]. These methods usually use priors, based on inclusion or exclusion interactions between objects. However, their globally optimal results are restricted only to some particular cases and they usually have a high computational cost. The methods based in LOGISMOS [22], [23] require an approximated pre-segmentation whenever the objects present complex shapes. Note that the fast segmentation obtained by our proposed method could also be used as a starting point for LOGISMOS.

In the context of segmentation by the *Image Foresting Transform* (IFT) framework, in order to incorporate structural information among objects, Fuzzy Object Models (FOMs) [24]–[32] are usually employed. However, these approaches are based on separate IFT executions per object, that do not incorporate contextual information and the high-level priors of all objects in a single energy optimization, limiting their potential.

In this work, we circumvent the aforementioned problem of IFT-based methods, by proposing a hierarchical layered graph-based approach for the multiple object segmentation by *Oriented Image Foresting Transform* (OIFT) [17], named as *Hierarchical Layered OIFT* (HLOIFT), using a single energy optimization. A hierarchical graph is defined by graph-based layers representing each image object. The hierarchy considers geometric interactions as inclusion and exclusion between objects as in [19], but we also argue that each image object must have its own set of local and global priors. Further, we formulate the integration of local and structural priors from layers, within a single energy optimization, overcoming the mentioned limitations from previous works and conserving the low computational cost of the OIFT framework.

Therefore, our main contributions are: **Theoretical**, we propose a new method for multi-object segmentation allowing high-level priors for image objects and geometric interactions between them. **Generality**, our approach is less restrictive, leading to globally optimal results in more general scenarios.

Complexity, our method has low computational complexity compared to methods based on min-cut/max-flow.

A. Paper Overview

Our paper is organized as follows: In Section II, we introduce some required definitions and review the IFT and OIFT frameworks. HLOIFT is then proposed in Section III. In Section IV, we evaluate HLOIFT and our conclusions are stated in Section V.

II. Background

In this section, we introduce some notations and definitions used in our proposed method and review the IFT and OIFT frameworks, introduced before in [7], [17].

An *image* $\mathbf{I}$ is a pair $(\mathcal{I}, I)$ where $\mathcal{I}$ is a finite set of *pixels*, i.e., $\mathcal{I} \in \mathbb{Z}^2$, and I is a mapping that assigns a pixel value $I(t)$ to each pixel $t \in \mathcal{I}$.

An *adjacency relation* $\mathcal{A}$ is a binary relation between the pixels within $\mathcal{I}$. We usually consider translation invariant relations, meaning that $\mathcal{A}$ depends only on the relative position $t - s$ of the pixels. For example, we can take $\mathcal{A}$ to consist of all pairs of distinct pixels $(s,t) \in \mathcal{I} \times \mathcal{I}$ such that $d(s,t) \leq \rho$ where $d(s,t)$ denotes the Euclidean distance and ρ is a specified constant (e.g., 4-neighborhood when $\rho = 1$ and 8-neighborhood when $\rho = \sqrt{2}$). We use the expressions $t \in \mathcal{A}(s)$ and $(s,t) \in \mathcal{A}$ to indicate that t is adjacent to s.

For a given adjacency relation $\mathcal{A}$, an image $\mathbf{I}$ can be interpreted as a weighted digraph $G = (\mathcal{I}, \mathcal{A}, \omega)$, where the image pixels in $\mathcal{I}$ are the nodes, the pixel pairs $(s,t) \in \mathcal{A}$ are directed arcs and $\omega(s,t) \geq 0$ is a fixed weight assigned to each arc (s,t), which may be obtained by a dissimilarity measure between pixels s and t. A digraph G is symmetric if $\forall (s,t) \in \mathcal{A}$, the pair (t,s) is also an arc of G.

For a given graph G, a *path* is a sequence of pixels $\pi = \langle t_1, t_2, \ldots, t_k \rangle$, where $(t_i, t_{i+1}) \in \mathcal{A}$, for $1 \leq i \leq k-1$. We use π_t to indicate that pixel t is the *terminus* of the path. To explicitly indicate the *origin* of the path, we use the notation $\pi_{s \rightsquigarrow t} = \langle s = t_1, t_2, \ldots, t_k = t \rangle$, where s indicates the origin and t the terminus. A path is *trivial* when $k = 1$. If π_s and $\tau = \langle s,t \rangle$ are both paths, we denote as $\pi_s \cdot \tau$ the concatenation of the two paths. We denote as $\Pi(G,t)$ the set of all paths in the graph G with terminus t. A *spanning forest* is a function P that assigns to each pixel t in $\mathcal{I}$ either some other adjacent pixel in $\mathcal{I}$, or a distinctive marker nil not in $\mathcal{I}$, with no containing cycles. Thus, for any pixel $t \in \mathcal{I}$, a spanning forest P defines a path π_t^P recursively as $\langle t \rangle$ if $P(t) = nil$ (root node), and $\pi_s^P \cdot \langle s,t \rangle$ if $P(t) = s \neq nil$ (predecessor node of t).

A *connectivity function* is a function f that assigns to each path π a path cost value $f(\pi)$. Usually, the cost of path depends on the arc weights ω along the path. In this paper, we use connectivity functions (f) constrained to paths starting in a given set $S \subseteq \mathcal{I}$ of pixels. We use the term *seeds* to denote S. This constraint is modeled by defining a new path-cost function $f^S(\pi)$, which is equal to $f(\pi)$ when the origin of π belongs to S, and it is equal to $+\infty$ otherwise.

A path π_t is *optimum* if $f(\pi_t) \leq f(\tau_t)$ for any other path $\tau_t \in \Pi(G,t)$. Let $\mathcal{V}$ be a set of cost values. Without loss of generality, we assume that $\mathcal{V}$ contains a maximum element, which we denote by $+\infty$. The *optimum path value* $V_{opt}(t)$ is obtained by taking to each terminus pixel t one optimum path, which is defined by

$$V_{opt}(t) = \min_{\forall \pi_t \in \Pi(G,t)} \{f(\pi_t)\}. \quad (1)$$

A. Image Foresting Transform (IFT)

The IFT algorithm solves the optimization problem, given by Equation 1, by dynamic programming, such that given an image graph G and a *smooth path-cost function* f [7], it associates one optimum path π_t^P to each pixel $t \in \mathcal{I}$. The IFT path-cost map V converges to V_{opt} being f a smooth function. Finally, a spanning forest P of optimum paths is obtained. Algorithm 1 describes the IFT procedure.

Algorithm 1 General IFT algorithm

INPUT: Image $\mathbb{I} = (\mathcal{I}, I)$, adjacency $\mathcal{A}$, set of seeds S_o (object) and S_b (background), and path-cost function f.
OUTPUT: Optimum-path forest P, the path-cost map V which converges to V_{opt} in the case of smooth functions and label map L.

1: **for all** $t \in S_o$, **do** $L(t) \leftarrow 1$
2: **for all** $t \in S_b$, **do** $L(t) \leftarrow 0$
3: **for all** $t \in \mathcal{I}$ **do**
4: Set $P(t) \leftarrow nil$, $V(t) \leftarrow f(\langle t \rangle)$
5: Set $status(t) \leftarrow 0$. ▷ $status$ is an auxiliary array
6: **if** $V(t) \neq +\infty$, **then** insert t in Q ▷ Q is an auxiliary priority queue
7: **while** $Q \neq \emptyset$ **do**
8: Remove s from Q such that $V(s)$ is minimum
9: Set $status(s) \leftarrow 1$
10: **for all** $t \in \mathcal{A}(s)$, such that $status(t) = 0$, **do**
11: Compute $tmp \leftarrow f(\pi_s^P \cdot \langle s,t \rangle)$ ▷ tmp is an auxiliary variable
12: **if** $tmp < V(t)$, **then**
13: **if** $V(t) \neq +\infty$, **then**
14: Remove t from Q
15: Set $P(t) \leftarrow s$, $V(t) \leftarrow tmp$
16: Set $L(t) \leftarrow L(s)$
17: Insert t in Q

In the case of non-smooth functions, the IFT algorithm produces a spanning forest P but the paths π_t^P may not be optimum. However, the computed spanning forest P may be optimal according to other optimality criteria, as proved for the cost function of OIFT [17].

B. Oriented Image Foresting Transform (OIFT)

The OIFT method explores the object-contour orientation, in directed graphs using non-smooth connectivity functions (NSCF), as described in [17], and some optimal criteria based on a cut measure on the graph. It improves the segmentation results because it helps to distinguish between two similar and nearby boundary segments with opposite orientations from distinct regions.

Let G be a strongly connected and symmetric digraph. The arcs weights $\omega(s,t)$, are a combination of an undirected dissimilarity measure $\psi(s,t)$ between neighboring pixel s and t, multiplied by an orientation factor, as follows:

$$\omega(s,t) = \begin{cases} \psi(s,t) \times (1+\alpha) & \text{if } I(s) > I(t), \\ \psi(s,t) \times (1-\alpha) & \text{if } I(s) < I(t), \\ \psi(s,t) & \text{otherwise,} \end{cases} \quad (2)$$

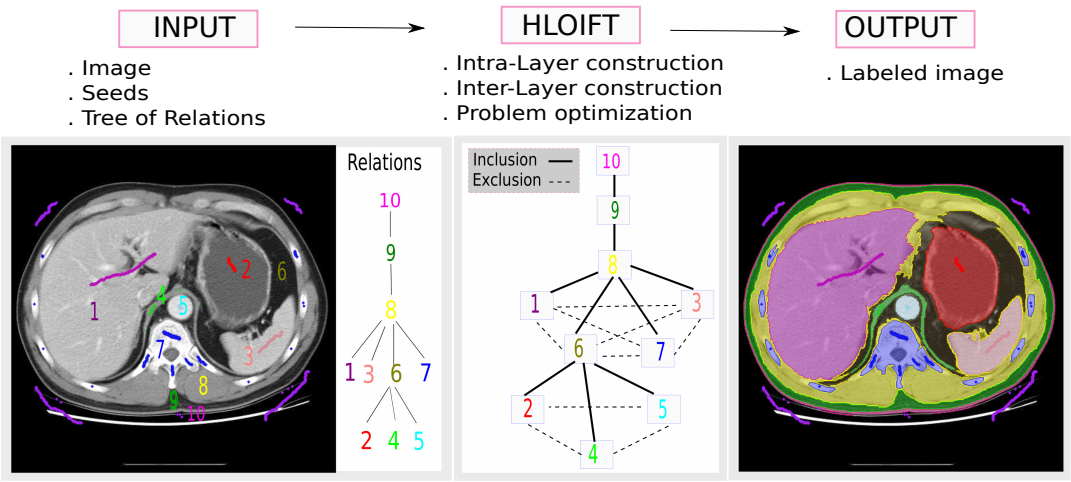

Fig. 1: Overview of our multi-object segmentation framework. Given the input parameters, HLOIFT constructs a hierarchical layered weighted digraph using the *inclusion* (solid line) and *exclusion* (dashed line) geometric interactions between objects. Finally, we have a labeled image as output.

where $\alpha \in [-1, 1]$. Note that, we usually have $\omega(s,t) \neq \omega(s,t)$ for $\alpha \neq 0$. Also, for $\alpha > 0$, the segmentation by OIFT favors transitions from bright to dark pixels, and $\alpha < 0$ favors the opposite orientation.

The OIFT is build upon the IFT framework by considering the following path-cost function in a symmetric digraph:

$$f^{\star}(\langle t \rangle) = \begin{cases} -1 & \text{if } t \in \boldsymbol{\mathcal{S}_o} \cup \boldsymbol{\mathcal{S}_b} \\ +\infty & \text{otherwise} \end{cases}$$
$$f^{\star}(\pi_{r \rightsquigarrow s} \cdot \langle s,t \rangle) = \begin{cases} \omega(s,t) & \text{if } r \in \boldsymbol{\mathcal{S}_o} \\ \omega(t,s) & \text{otherwise} \end{cases} \quad (3)$$

where the function $f^{\star}$ is a non-smooth connectivity function.

Finally, the OIFT segmentation is defined from the labeled map L computed by Algorithm 1 using the function $f^{\star}$. The optimality by OIFT is supported by the maximization of the energy $E(L)$ (Equation 5) involving the outer-cut boundary of the object in the image graph $C(L)$ (Equation 4).

$$\boldsymbol{\mathcal{C}}(L) = \{(s,t) \in \mathcal{A} \mid L(s) = 1 \text{ and } L(t) = 0\} \quad (4)$$

$$E(L) = \min_{(s,t) \in \boldsymbol{\mathcal{C}}(L)} \omega(s,t) \quad (5)$$

III. Hierarchical Layered OIFT

We propose the ***Hierarchical Layered Oriented Image Foresting*** (HLOIFT) as a new method for multi-object segmentation. Figure 1 shows an overview of our framework. For a given input image, seeds sets for some objects, and the tree of relations between objects, the HLOIFT method has the following steps: (1) HLOIFT constructs each layer as a weighted digraph representing each object with its own priors (described in Section III-B). (2) HLOIFT defines a setup for the inter-layer connections representing the geometric interactions, such as inclusion and exclusion relations, by a hierarchy of layers (described in Section III-C). (3) HLOIFT uses an extension of the OIFT algorithm to compute an optimal cut over the hierarchical layered digraph, giving as output a labeled image (described in Section III-D).

A. Notations and definitions

We have the following definitions for multiple object segmentation. We denote by $\mathcal{L}$ an index set, where each element in $\mathcal{L}$ is associated to an object. We denote by $m = |\mathcal{L}|$ the number of objects to be segmented. The binary variable x_p^i identifies the location of pixel $p \in \mathcal{I}$ with respect to the object $i \in \mathcal{L}$, i.e., $x_p^i = 1$ or $x_p^i = 0$ for pixels inside and outside the ith object from layer i, respectively.

We denote as h the *tree* of relations among objects, where each of its nodes represents an object and the arcs represent the a priori inclusion relation between two objects. Formally h is an acyclic connected graph whose nodes are indexed by $1 \leq i, j \leq m$. We use the notation $h[i].parent = j$ to indicate that node j is the parent of node i and the notation $h[i].parent = h[j].parent$, to say that the two nodes are sibling nodes. Next, we present the two resulting definitions from the relations coded in h.

Definition 1 (Parent object)**.** *For a given sequence of sets $O_1, O_2, \ldots O_m$ of pixels, which represent m objects, such that $O_1 \subseteq O_2 \subseteq \cdots \subseteq O_m$, we define the* parent object *of O_i $(1 \leq i \leq m-1)$ as the object O_{i+1}, that is, $h[i].parent = i+1$.*

Thus, notation $h[i].parent = j$ also reflects that an object O_j is a parent object of an object O_i. If $h[i].parent = -1$ we have that the image domain $\mathcal{I}$ is the parent of object i.

Definition 2 (Sibling object)**.** *We say that the following sets $O_{i1}, O_{i2}, \ldots, O_{ik}$, for $k > 1$, are sibling objects if they have the same parent object O_j (i.e., $h[i1].parent = j, \ldots, h[ik].parent = j$) and $O_{ix} \cap O_{iy} = \emptyset$ for all $1 \leq x, y \leq k$ where $x \neq y$.*

B. Layer digraph construction

The first step of HLOIFT is to create a set of m layers, where each layer $\mathcal{H}_i$ represents one corresponding object O_i to be segmented. It is formally defined as:

Definition 3 (Layer)**.** *A layer $\mathcal{H}_i = (\mathcal{I}, \mathcal{A}_{ii}, \omega_{ii})$, is a weighted digraph where $i \in \mathcal{L}$, $\mathcal{A}_{ii}$ is the intra-layer adjacency relation and ω_{ii} are the weights of the intra-layer arcs.*

An object O_i can be defined by setting its priors, for instance, we could use boundary polarity priors [17], geodesic star convexity priors [33], both constraints at the same time [14], or none of them.

These priors affect the layer construction as follows. In the case of **boundary polarity priors** to explore the object-contour orientations, we use the same scheme that was adopted by the regular OIFT method, where $\omega_{ii}(s,t)$ is defined as described by Equation 2 and $\psi(s,t)$ is usually set to be the value $|I(s) - I(t)|$, however, other settings can be used too. Each object O_i has its own α value in $[-1, 1]$ coded in the node $h[i]$ of the tree of relations h as $h[i].\alpha$, so that we can favor the segmentation of O_i with transitions from *bright to dark* pixels ($h[i].\alpha > 0$) or the opposite orientation ($h[i].\alpha < 0$). Note that $h[i].\alpha = 0$ can be used to indicate that the object O_i has no boundary polarity prior.

If the ith layer is created using the **geodesic star convexity prior** then we will be prioritizing the segmentation of O_i with more regular shape [33]. Moreover, it is still possible to simultaneously handle boundary polarity and shape priors [14]. The geodesic star convexity prior for the ith layer is obtained by setting the weights ω_{ii} of some arcs in $\mathcal{H}_i$ to $-\infty$, according to the scheme proposed in [14].

C. Setup of inter-layer connections

In this step, HLOIFT generates a weighted digraph $\mathcal{H}$ from the set of m layers $\{\mathcal{H}_1, \mathcal{H}_2, ..., \mathcal{H}_m\}$ created in the previous step, according to the template given by the *tree* h of geometric interactions among objects. The *hierarchical layered digraph* is defined as:

Definition 4 (Hierarchical layered digraph)**.** *A hierarchical layered graph $\mathcal{H}$ is the weighted digraph obtained as the union of all graphs $\mathcal{H}_i$, $i = 1, \ldots, m$, with additional* inter-layer *arcs, described by an* inter-layer *adjacency relation $\mathcal{A}_{ij}$, with ω_{ij} representing the inter-layer arcs weights between pixels of layers $\mathcal{H}_i$ and $\mathcal{H}_j$.*

We use two types of geometric interactions among objects: **inclusion**, when an object is a subset of another object, and **exclusion**, when the two objects are disjoint sets. These geometric interactions were previously studied under the context of the graph cut algorithm in [19], but globally optimal results were restricted only to limited cases involving certain submodular functions of graph cut.

Thus, HLOIFT defines the adjacency relation $\mathcal{A}_{ij}$ and the ω_{ij} between any pair of objects related as parent or sibling objects. In the case of inclusion relation, A_{ij} and ω_{ij} are defined as follows:

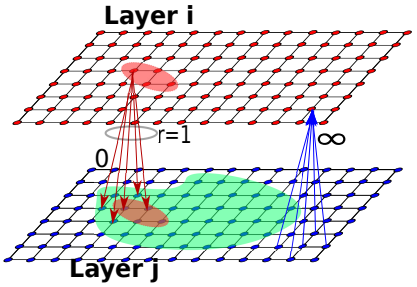

Fig. 2: Inter-layer arc construction, involving two objects i, j for inclusion (j is the parent of i), where $w_{ij} = 0$ and $w_{ji} = \infty$.

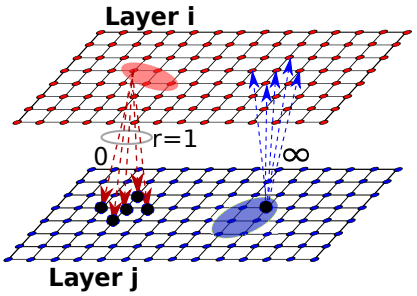

Fig. 3: Inter-layer arc construction, involving two objects i, j for the exclusion case (i, j are siblings), where $w_{ij} = 0$ and $w_{ji} = \infty$. The special arcs are represented by dashed lines and the flipped meaning of x_p^j's variables in layer j are marked with black circles.

- **Inclusion.** For each $h[i].parent = j$, where $i = 1, \ldots, m$ and $j \neq -1$, we define the arcs $(s,t) \in A_{ij}$ with $w_{ij}(s,t) = 0$ from node s in $\mathcal{H}_i$ to t in $\mathcal{H}_j$ considering a *radius* r defined by the Euclidean adjacency relation, such that $d(s',t) \leq r$, where s' is the corresponding node of pixel s in the layer $\mathcal{H}_j$ and $d(s',t)$ denotes the Euclidean distance. A reversed arc $(t,s) \in A_{ij}$ from node t in $\mathcal{H}_j$ to s in $\mathcal{H}_i$ is also considered with the worst cost $w_{ji}(t,s) = \infty$. Figure 2 shows the definition of inter-layer arcs using a 4-neighborhood adjacency ($r = 1$).
- **Exclusion.** For each pair of objects, such that $h[i].parent = h[j].parent$ $(i, j = 1, \ldots, m)$ and $i > j$, we define A_{ij} and w_{ij} similarly as for the inclusion case, but we consider *special inter-layer arcs*, which flip the meaning of x_p^j's variable values in layer j in the proposed HLOIFT algorithm, as will be discussed later. Figure 3 presents the special inter-layer arcs by dashed lines and the flipped meaning of variables x_p^j in layer j is represented by black circles.

In Figure 4, we give an example of the hierarchical layered digraph construction for a case that cannot be optimized under the graph cut framework [19]. This case combines the inclusion and exclusion geometric interactions, such that we have two mutually exclusive objects O_i and O_j, both contained within another object O_k. Globally optimal segmentation in

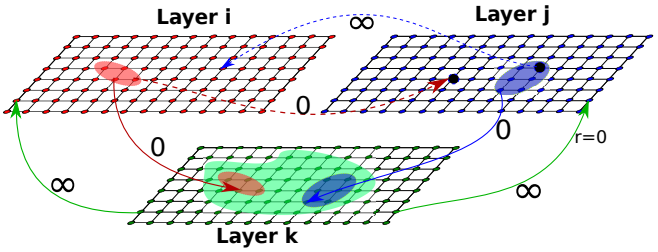

Fig. 4: Inter-layer arc construction, involving three objects i, j and k. The special arcs are represented by dashed lines and the flipped meaning of x_p^j's variables in layer j by black circles.

this case cannot be modeled with graph cuts, because it cannot be converted to a submodular energy. The proposed HLOIFT method can compute globally optimal results in this case and also in other more sophisticated cases, as depicted in Figure 1.

D. Energy optimization

In the final step, we execute the HLOIFT algorithm, which corresponds to a OIFT execution [17], [34], in the hierarchical layered graph $\mathcal{H}$, with a modified path function to properly address the special inter-layer arcs from exclusion relations, resulting in a single energy optimization, simultaneously taking into account all the constraints from all objects.

The HLOIFT algorithm uses a revised path function $f^\star$, formally defined as:

$$f^\star(\langle t \rangle) = \begin{cases} -1 & \text{if } t \in \boldsymbol{\mathcal{S}_o} \cup \boldsymbol{\mathcal{S}_b}, \\ +\infty & \text{otherwise,} \end{cases} \quad (6)$$

$$f^\star(\pi_{r \rightsquigarrow s} \cdot \langle s,t \rangle) = \begin{cases} \omega(s,t) & \text{if } L(s)=1 \text{ and } \textbf{Case 1}, \\ \omega(s,t) & \text{if } L(s)=0 \text{ and } \textbf{Case 2}, \\ \omega(t,s) & \text{otherwise,} \end{cases}$$

where r, s, t are graph nodes, $\omega(s,t) = \omega_{i,j}(s,t)$ with s in $\mathcal{H}_i$ and t in $\mathcal{H}_j$, $\boldsymbol{\mathcal{S}_b}$ and $\boldsymbol{\mathcal{S}_o}$ are seed nodes, such that $\boldsymbol{\mathcal{S}_o} = \mathcal{S}_o^1 \cup ... \cup \mathcal{S}_o^m$ and $\mathcal{S}_o^i$ is the set of seed nodes in $\mathcal{H}_i$ selected inside the ith object in layer i. The background seed pixels are marked for all the corresponding nodes in all the m layers in $\boldsymbol{\mathcal{S}_b}$. Let $l_t \in \mathcal{L}$ represent the index layer for the node t.

Case1: We have $l_s = l_t$, or $h[l_s].parent = l_t$, or $h[l_t].parent = l_s$ (inclusion), or $h[l_s].parent = h[l_t].parent$ ((exclusion, when $l_s > l_t$).

Case2: We have $h[l_t].parent = l_s$ and $l_s < l_t$.

Next, we describe how the HLOIFT algorithm generates the paths, propagating the variable values $x_s^i = L(s)$ and $x_t^j = L(t)$ from a pixel s to its adjacent t, for **inclusion** and **exclusion** cases in the modified IFT algorithm:

- For the **inclusion** relation: Line 16 of Algorithm 1 must be kept unchanged as "Set $L(t) \leftarrow L(s)$". Therefore, we will have $L(t) = x_t^j = x_s^i = L(s)$ whenever the cost $f^\star(\pi_{r \rightsquigarrow s} \cdot \langle s,t \rangle) = 0$.
  For **exclusion** relation: Line 16 of Algorithm 1 we must be changed to "Set $L(t) \leftarrow 1 - L(s)$", because $x_t^{l_t} = L(t)$ and $x_s^{l_s} = L(s)$ have flipped meaning. Therefore, we will have $L(t) = x_t^j \neq x_s^i = L(s)$ whenever the cost $f^\star(\pi_{r \rightsquigarrow s} \cdot \langle s,t \rangle) = 0$.

The optimality by HLOIFT is supported by an energy maximization of $E(L)$, on a labeled image L and its resulting cut in the graph $\mathcal{H}$, involving arcs from object to background pixels $C(L)$ (outer-cut boundary), both defined in Equation 5 and Equation 4.

Respect to the computational complexity, HLOIFT is $\mathcal{O}(M+N)$, where N is the number of vertex and M is the number of arcs in the graph $\mathcal{H}$, when $\mathcal{Q}$ is implemented using bucket sorting [7] and $\mathcal{O}(M + N \log N)$ (linearithmic time) if $\mathcal{Q}$ is a heap. The Graph cut is $\mathcal{O}(\sqrt{M} * N^2 = N^{2.5})$ when $\mathcal{H}$ is a sparse graph, which is more than quadratic-time using a push-relabel based on the highest label node selection rule [35].

IV. Experiments

This section presents an experimental evaluation of the HLOIFT method. We formulated four experiments: The first experiment analyzes the effect of high levels priors on the segmentation result. The second experiment assesses the performance of the HLOIFT against the multiple object segmentation by the IFT method. The third experiment shows an application on colored images. Finally, the fourth experiment shows how to model and segment objects on more sophisticated images such as medical images. **Reproducibility**, our code and data are available under request.

A. Different priors for objects

This experiment shows how the definition of high-level priors for each object is essential to obtain a target segmentation result. Figure 5 shows a two object segmentation problem for a given synthetic image. We set an internal seed for the object O_1, and the relation that O_2 is parent of O_1 (i.e., $h[1].parent = 2$). Our HLOIFT model with different priors for each object can provide different results allowing the easy customization of the segmentation to any desired particular result.

B. Comparison with the IFT method

The objective of this experiment is to visually compare the results obtained by HLOIFT against the IFT method [7] for multi-object segmentation. We compared with the IFT because it allows multiples objects segmentation while the original OIFT only works for binary segmentation.

Figure 6 shows a medical image segmentation task where the goal is to segment the knee with three objects. The objects O_1, O_2 and O_3 are shown in yellow, blue and red color, respectively.

The left part, from first row, of the figure contains the input image with seeds for each object (for the three objects plus background) and the output segmentation obtained by the IFT method. The second row of the figure contains the image with only seeds for some objects (for two inner objects plus background), and as output, we are given two results

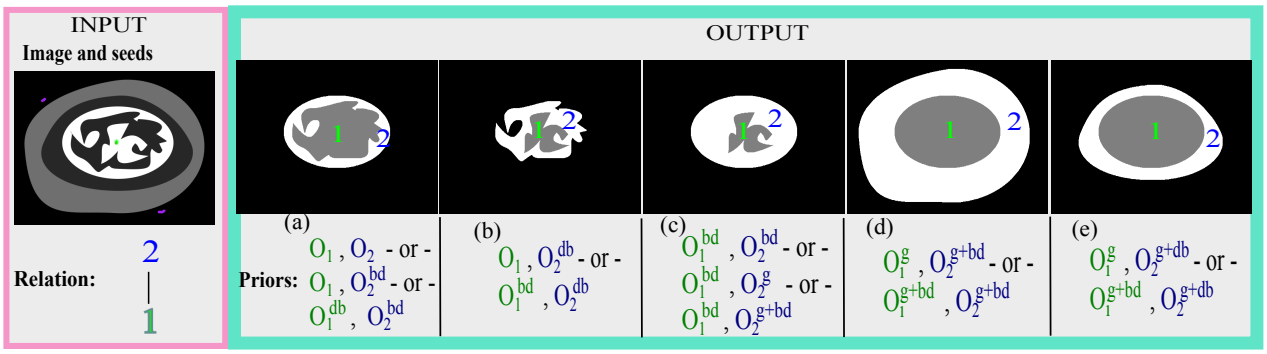

Fig. 5: Example of two object segmentation using HLOIFT where O_2 is parent of O_1. Each object has different high-level priors –db: polarity from dark to bright and bd: from bright to dark, g: geodesic star prior.

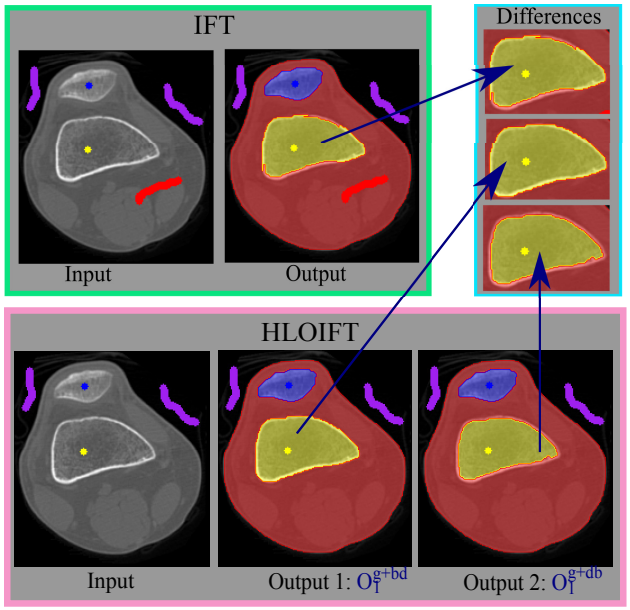

Fig. 6: Knee segmentation modeled by three objects. On the left of first row, we have the output using IFT; on the second row, the results by HLOIFT using different priors for O_1; and the main differences are highlighted in the right column of first row.

of segmentation obtained by HLOIFT. The first output result was obtained using the prior geodesic start constraint with boundary polarity from bright to dark ($g + bd$) for all objects, and for the second output the O_1 uses the prior geodesic with boundary polarity from dark to bright ($g + db$).

Now, in the right column of first row, we can observe the differences highlighted in the O_1's contour. The IFT gives poor result for O_1 mixing bright and dark boundaries because it does not allow the usage of the boundary polarity prior. On the other hand, HLOIFT can define different priors for each object, giving better results compared to IFT and also requiring fewer seeds. HLOIFT can be seen as an extension of OIFT for multiple object segmentation.

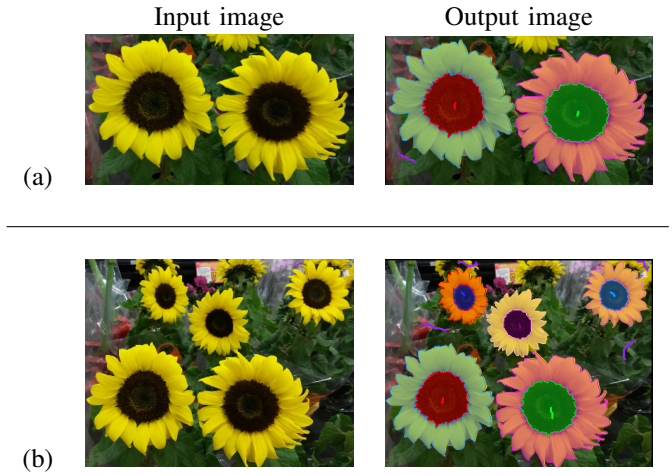

Fig. 7: Application of HLOIFT in colored images for (a) two sunflowers and (b) five sunflowers, where each sunflower is defined by two objects.

C. Application on colored images

Figure 7 shows one application example of HLOIFT in colored images. In the first row (a), we show the segmentation result of two sunflowers, and in the second row (b) we have the result for five sunflowers. In all cases, each sunflower is properly defined by two objects, the seeds (dark part) and the petals (yellow part), modelled by a inclusion relation. Also, each seeds part has a geodesic star prior.

D. Application on medical image segmentation

In medical image segmentation is common to have sophisticated multiple objects which are hard to detect with simple models without user interaction. We show that by modelling them under the HLOIFT framework we have good results with less effort. An example of a sophisticated medical image segmentation, involving ten objects, was presented in Figure 1.

Another example is given in Figure 8 for brain segmentation involving seven objects. As input we have the given image, the objects definition and the seeds. In this case we are given seeds only for the objects O_1, O_2 and background. Then, by

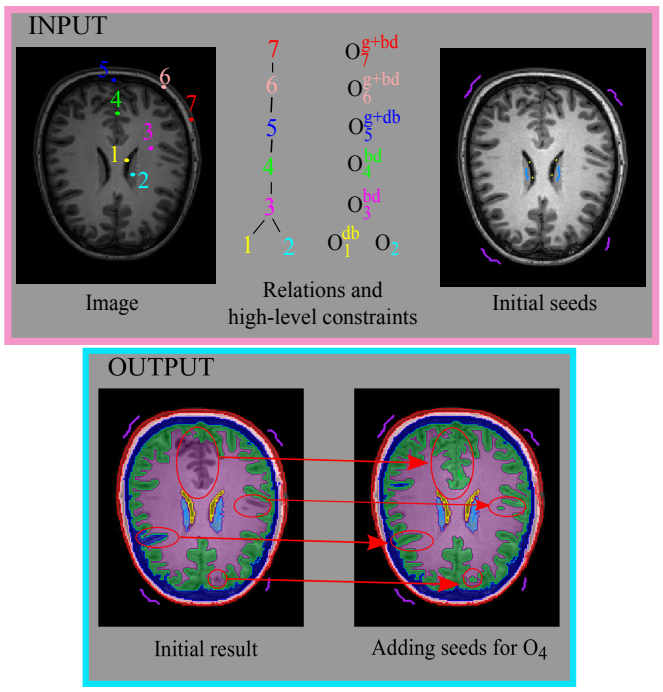

Fig. 8: Brain segmentation considering the following objects: internal capsule (O_1), putamen (O_2), white matter (O_3), gray matter (O_4), cerebral spinal fluid (O_5), skull (O_6) and scalp (O_7). For the given input parameters, we obtain an initial result. By adding more seeds (red circles), the segmentation result is improved.

using the tree of relations and the high-level priors for each object, we can automatically segment the other objects. As output, the obtained initial result still contains some small errors, however our framework allows the user to perform corrections to improve the result with a simple user interaction of adding new seeds (second row of Figure 8).

V. Conclusion

We proposed a new graph-based approach, named as HLOIFT, which extends OIFT for multi-object segmentation, allowing each object to have its own high-level priors and preserving its low time complexity. Besides the theoretical contribution in the context of the multi-object segmentation problem, our experiments show that good segmentation results can be obtained, even when considering a simple measure of intensity dissimilarity. Besides being faster than min-cut/max-flow based approaches, it is also less restrictive, allowing globally optimal results for arbitrary hierarchies. Note also that HLOIFT could be combined with LOGISMOS. Finally, as a future work we intend to compare HLOIFT with other methods from the literature in more specific applications.

Acknowledgment

The authors would like to thank CNPq (308985/2015-0, 486083/2013-6, 486988/2013-9, FINEP 1266/13), FAPESP (2011/50761-2, 2016/21591-5), CAPES, and NAP eScience - PRP - USP for funding. This research is also part of the FAPESP Thematic Research Project (proc. 2014/12236-1) and part of the INCT of the Future Internet for Smart Cities funded by CNPq, proc. 465446/2014-0, CAPES proc. 88887.136422/2017-00, and FAPESP, proc. 2014/50937-1.

References

[1] S. Gordon, I. Dolgopyat, I. Kahn, and T. R. Raviv, "Co-segmentation of multiple images into multiple regions: Application to mouse brain MRI," in *13th IEEE International Symposium on Biomedical Imaging, ISBI 2016, Prague, Czech Republic, April 13-16, 2016*. IEEE, 2016, pp. 399–402. [Online]. Available: http://dx.doi.org/10.1109/ISBI.2016.7493292

[2] K. D. Toennies, *Guide to medical image analysis: methods and algorithms*. Springer Science & Business Media, 2012.

[3] E. Visser, M. C. Keuken, G. Douaud, V. Gaura, A.-C. Bachoud-Levi, P. Remy, B. U. Forstmann, and M. Jenkinson, "Automatic segmentation of the striatum and globus pallidus using mist: Multimodal image segmentation tool," *NeuroImage*, vol. 125, pp. 479 – 497, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1053811915009143

[4] M. Jackowski and A. Goshtasby, *A Computer-Aided Design System for Segmentation of Volumetric Images*. Boston, MA: Springer US, 2005, pp. 251–272.

[5] J. Cousty, G. Bertrand, L. Najman, and M. Couprie, "Watershed cuts: Thinnings, shortest path forests, and topological watersheds," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 5, pp. 925–939, 2010.

[6] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient nd image segmentation," *International journal of computer vision*, vol. 70, no. 2, pp. 109–131, 2006.

[7] A. X. Falcão, J. Stolfi, and R. de Alencar Lotufo, "The image foresting transform: Theory, algorithms, and applications," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 1, pp. 19–29, 2004.

[8] H. H. C. Bejar and P. A. Miranda, "Oriented relative fuzzy connectedness: theory, algorithms, and its applications in hybrid image segmentation methods," *EURASIP Journal on Image and Video Processing*, vol. 2015, no. 1, pp. 1–15, 2015.

[9] J. K. Udupa and P. K. Saha, "Fuzzy connectedness and image segmentation," *Proceedings of the IEEE*, vol. 91, no. 10, pp. 1649–1669, 2003.

[10] S. Vicente, V. Kolmogorov, and C. Rother, "Graph cut based image segmentation with connectivity priors," in *Computer vision and pattern recognition, 2008. CVPR 2008. IEEE conference on*. IEEE, 2008, pp. 1–8.

[11] L. A. Mansilla and P. A. Miranda, "Oriented image foresting transform segmentation with connectivity constraints," *ICIP*, 2016.

[12] H. Isack, O. Veksler, M. Sonka, and Y. Boykov, "Hedgehog shape priors for multi-object segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2434–2442.

[13] V. Gulshan, C. Rother, A. Criminisi, A. Blake, and A. Zisserman, "Geodesic star convexity for interactive image segmentation," in *Proceedings of Computer Vision and Pattern Recognition*, 2010, pp. 3129–3136.

[14] L. Mansilla and P. Miranda, "Image segmentation by oriented image foresting transform with geodesic star convexity," in *Computer Analysis of Images and Patterns (CAIP)*, vol. 8047, York, UK, Aug 2013, pp. 572–579.

[15] O. Veksler, "Star shape prior for graph-cut image segmentation," *Computer Vision–ECCV 2008*, pp. 454–467, 2008.

[16] D. Singaraju, L. Grady, and R. Vidal, "Interactive image segmentation via minimization of quadratic energies on directed graphs," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[17] P. A. Miranda and L. A. Mansilla, "Oriented image foresting transform segmentation by seed competition," *Image Processing, IEEE Transactions on*, vol. 23, no. 1, pp. 389–398, 2014.

[18] F. M. Lopes, D. C. Martins, J. Barrera, and R. M. Cesar, "A feature selection technique for inference of graphs from their known topological properties: Revealing scale-free gene regulatory networks," *Information Sciences*, vol. 272, pp. 1 – 15, 2014.

[19] A. Delong and Y. Boykov, "Globally optimal segmentation of multi-region objects," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 285–292.

[20] A. Delong, L. Gorelick, O. Veksler, and Y. Boykov, "Minimizing energies with hierarchical costs," *International journal of computer vision*, vol. 100, no. 1, pp. 38–58, 2012.

[21] J. Ulén, P. Strandmark, and F. Kahl, "An efficient optimization framework for multi-region segmentation based on lagrangian duality," *Medical Imaging, IEEE Transactions on*, vol. 32, no. 2, pp. 178–188, 2013.

[22] Y. Yin, X. Zhang, R. Williams, X. Wu, D. D. Anderson, and M. Sonka, "LOGISMOS–layered optimal graph image segmentation of multiple objects and surfaces: cartilage segmentation in the knee joint," *IEEE transactions on medical imaging*, vol. 29, no. 12, pp. 2023–2037, 2010.

[23] I. Oguz and M. Sonka, "LOGISMOS–B: layered optimal graph image segmentation of multiple objects and surfaces for the brain," *IEEE transactions on medical imaging*, vol. 33, no. 6, pp. 1220–1235, 2014.

[24] P. A. V. Miranda, A. X. Falcao, and J. K. Udupa, "Cloud bank: A multiple clouds model and its use in mr brain image segmentation," in *2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, June 2009, pp. 506–509.

[25] J. Udupa, D. Odhner, A. Falcão, K. Ciesielski, P. Miranda, P. Vaideeswaran, S. Mishra, G. Grevera, B.Saboury, and D. Torigian, "Fuzzy object modeling," in *Proc.SPIE*, vol. 7964, 2011, pp. 7964 – 7964 – 10.

[26] J. Udupa, D. Odhner, A. Falcão, K. Ciesielski, P. Miranda, S. Mishra, G. Grevera, B. Saboury, and D. Torigian, "Automatic anatomy recognition via fuzzy object models," in *In Proceedings of SPIE on Medical Imaging: Image-Guided Procedures, Robotic Interventions, and Modeling*, vol. 8316, San Diego, California, USA, 2012.

[27] J. K. Udupa, D. Odhner, Y. Tong, M. M. S. Matsumoto, K. C. Ciesielski, P. Vaideeswaran, V. Ciesielski, B. Saboury, L. Zhao, S. Mohammadianrasanani, and D. Torigian, "Fuzzy model-based body-wide anatomy recognition in medical images," vol. 8671, 2013, pp. 86 712B–86 712B–7. [Online]. Available: http://dx.doi.org/10.1117/12.2007983

[28] L. Rittner, J. K. Udupa, and D. A. Torigian, "Multiple fuzzy object modeling improves sensitivity in automatic anatomy recognition," in *In Proceedings of SPIE on Medical Imaging: Image Processing*, vol. 9034, San Diego, California, USA, 2014.

[29] K. Sun, J. K. Udupa, D. Odhner, Y. Tong, and D. A. Torigian, "Automatic thoracic anatomy segmentation on ct images using hierarchical fuzzy models and registration," in *In Proceedings of SPIE on Medical Imaging: Image-Guided Procedures, Robotic Interventions, and Modeling*, vol. 9036, San Diego, California, USA, 2014.

[30] J. K. Udupa, D. Odhner, L. Zhao, Y. Tong, M. M. Matsumoto, K. C. Ciesielski, A. X. Falcao, P. Vaideeswaran, V. Ciesielski, B. Saboury, S. Mohammadianrasanani, S. Sin, R. Arens, and D. A. Torigian, "Body-wide hierarchical fuzzy modeling, recognition, and delineation of anatomy in medical images," *Medical Image Analysis*, vol. 18, no. 5, pp. 752 – 771, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1361841514000498

[31] S. Mohammadianrasanani, "The use of a body-wide automatic anatomy recognition system in image analysis of kidneys," Master's thesis, School of Technology and Health, Royal Institute of Technology, Flemingsberg, Sweden, 2013.

[32] Y. Tong, J. K. Udupa, D. Odhner, S. Sin, and R. Arens, "Abdominal adiposity quantification at MRI via fuzzy model-based anatomy recognition," in *In Proceedings of SPIE on Medical Imaging: Biomedical Applications in Molecular, Structural, and Functional Imaging*, vol. 8672, Orlando, Florida, USA, 2013.

[33] L. Mansilla, M. Jackowski, and P. Miranda, "Image foresting transform with geodesic star convexity for interactive image segmentation," in *IEEE International Conference on Image Processing (ICIP)*, Melbourne, Australia, Sep 2013, pp. 4054–4058.

[34] L. Mansilla and P. Miranda, "Image segmentation by oriented image foresting transform: Handling ties and colored images," in *18th International Conference on Digital Signal Processing (DSP)*. Santorini, Greece: IEEE, Jul 2013, pp. 1–6.

[35] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 9, pp. 1124–1137, 2004.