Efficient optimization for Hierarchically-structured Interacting Segments (HINTS)

Hossam Isack[1] Olga Veksler[1] Ipek Oguz[2] Milan Sonka[3] Yuri Boykov[1]

University of Western Ontario[1] University of Pennsylvania[2] University of Iowa[3]

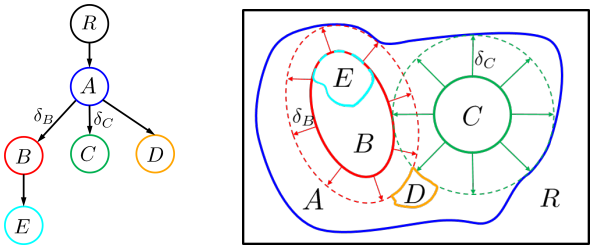
Abstract

We propose an effective optimization algorithm for a general hierarchical segmentation model with geometric interactions between segments. Any given tree can specify a partial order over object labels defining a hierarchy. It is well-established that segment interactions, such as inclusion/exclusion and margin constraints, make the model significantly more discriminant. However, existing optimization methods do not allow full use of such models. Generic a-expansion results in weak local minima, while common binary multi-layered formulations lead to non-submodularity, complex high-order potentials, or polar domain unwrapping and shape biases. In practice, applying these methods to arbitrary trees does not work except for simple cases. Our main contribution is an optimization method for the Hierarchically-structured Interacting Segments (HINTS) model with arbitrary trees. Our Path-Moves algorithm is based on multi-label MRF formulation and can be seen as a combination of well-known a-expansion and Ishikawa techniques. We show state-of-the-art biomedical segmentation for many diverse examples of complex trees.

1. Introduction

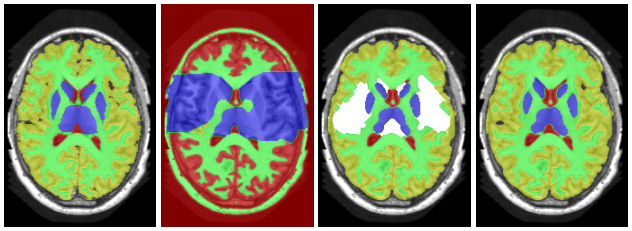
Basic cues like smooth boundaries and appearance models are often insufficient to regularize complex segmentation problems. This is particularly true in medical applications where objects have weak contrast boundaries and overlapping appearances. Thus, additional priors are needed, e.g. shape-priors [29], volumetric constraints [3], or segments interaction [31, 8]. The latter constraint is the essence of **H**ierarchically-structured **Int**eracting **S**egments (HINTS) model [8, 31], which was successfully applied to many segmentation problems, e.g. cells [23], joint cartilage [31], and cortical [24] or tubular [21] surfaces.

HINTS model overview: Any hierarchically-structured[1] segments could be represented as a label tree $\mathcal{T}$, see Fig. 1. Tree $\mathcal{T}$ defines topological relationship between segments as follows; (a) child-parent relation means the child segment is inside its parent's segment, (b) sibling relation means the corresponding segments exclude each other. For example, in Fig. 1 A is inside R, E is inside B, while B, C and D exclude each other in A. Min-margin is one form of interaction between regions. If region X has δ_X min-margin then its outside boundary pushes away the outside boundary of its parent and siblings by at least δ_X, Fig.1(b).



(a) tree $\mathcal{T}$ & margins δ (b) a feasible segmentation

Figure 1. (a) tree with 6 labels, and δ_B and δ_C are min-margins of labels B and C, respectively. None displayed margins imply zero min-margin. (b) feasible segmentation that satisfies the hierarchical-structure, i.e. partial ordering, and margins defined by $\mathcal{T}$ and δ. Notice how B's outside boundary pushes its parent's and siblings' outside boundaries to be at least δ_B pixels away from it.



ground truth a-exp[7, 8] QPBO [26, 8] ours

Figure 2. HINTS segmentation of brain using different optimization methods where white-matter (yellow), grey-matter (green) and background are nested regions, and cerebrospinal fluid (red) and sub-cortical grey-matter (blue) are mutually exclusive regions inside white-matter. Starting from a trivial solution a-exp converged to a bad local minimum unlike Path-Moves which explores more solutions. QPBO failed to label some pixels (shown in white) due to the non-submodular energy and ambiguous color models.

[1]We use hierarchically-structured and partially ordered interchangeably.

Limitations of previous algorithms: We extend [8], which introduced HINTS for arbitrary trees. In [8] a-expansion (a-exp) [7] was used to optimize the multi-label formulation of HINTS, but it often results in bad local minima due to complexities of interaction constraints, e.g. Fig. 2. The contribution of [8] is a binary multi-layered HINTS formulation. They use high-order data terms, which are not easy to convert into unary and pairwise potentials for arbitrary trees. Their algorithm's global optimality guarantee depends on the tree at hand. Only trees that do not yield *frustrated cycles* [26] have this guarantee, but this is not immediately obvious for any given tree. In [8], non-submodular binary energy implied by frustrated cycles were addressed by QPBO [26]. In practice, QPBO produces only partial solutions for most trees, see Figs. 2, 10 and 12.

As an alternative to QPBO, [28] formulated HINTS as constraint optimization. They solve the Lagrangian dual of this NP-hard problem using an iterative sequence of graph cuts. However, the duality gap may be arbitrarily large and the optimum for HINTS is not guaranteed. Their super-gradient optimization of Lagrange multiplier guesses initial solutions and time-step parameters. Also, according to Lemma 5 in [28] their super-gradient corresponds to the hard exclusion constraint of HINTS, which is $\{0, \infty\}$-valued. They do not discuss how this affects the algorithm.

In [31] the authors generalize their earlier method [21] for segmenting multiple nested surfaces, i.e. $\mathcal{T}$ is a chain. In [31] the aim was to segment multiple mutually exclusive objects each with a set of nested surfaces, i.e. $\mathcal{T}$ is a spider[2]. But, the proposed approach can handle only a single pair of mutually exclusive objects in a given image region. As such [31] requires a prior knowledge of the region of interaction for two excluded objects, or computes it using a problem specific trained classifier [31]. Such prior knowledge is not required for our method. In contrast to our approach, [31] requires a sufficiently close initial segmentation satisfying interaction constraints. In all of our experiments we started from a trivial solution. Unlike our approach, [31] implicitly imposes a star like shape prior [29] and use non-homogeneous anisotropic polar grids.

If interactivity (min-margin) constraints are dropped HINTS degenerates to *tree-metric labeling*. Certain tree-metric labeling problems are addressed in [10] using DP to find the global optima if the data terms are also a tree-metric. Recently, [1] used convex relaxation to approximate labeling problems where labels are leafs of a DAG. This problem can be reduced[3] to general *metric labeling* [18]. Such labeling problems are significantly different from HINTS due to lack of interactions between segments.

Motivation for Path-Moves: In the context of multi-label HINTS formulation, we propose an effective move-making algorithm applicable to *arbitrary* label trees $\mathcal{T}$ avoiding limitations of the previous optimization methods.

In contrast to a-exp [7], our Path-Moves are non-binary: when expanding label α any pixel can change its current label to any label in the path between its current label and α in the tree. Note that the path is specific to each pixel. Optimization uses our generalization of the well-known multi-layered Ishikawa technique [16] for convex potentials over strictly ordered labels. In essence, Path-Moves combine a-exp and Ishikawa. Indeed, in the special case of a chain-tree (nested segments) our algorithm reduces to Ishikawa-like construction in [8, 16] finding global minimum in one step. On the contrary, when $\mathcal{T}$ is a single-level star our algorithm reduces to a-exp. Note that closely related multi-label *range-moves* [30] also combine a-exp and Ishikawa for non-convex pairwise potentials over strictly ordered labels (a chain). In contrast to Path-Moves, in *range-moves* all pixels have the same set of feasible labels.

Our contributions are summarized below:

- we propose Path-Moves - approximate optimization method applicable to HINTS. Unlike [8, 31], Path-Moves work for arbitrary trees avoiding weak local minima typical of a-exp [7] in the context of HINTS.
- we show how a generalization of star shape priors, e.g. [29, 12, 14], integrate into multi-label HINTS model, if needed. Path-Moves can address this too.
- we show state-of-the-art biomedical segmentation results for complex trees.

The paper is organized as follows. Section 2 defines multi-label formulation of HINTS. Section 3 introduces our multi-label expansion move, which we call Path-Move. Separately, in Section 4 we discuss shape priors as they are not mandatory for HINTS when using Path-Moves, unlike [31, 21]. We validate and compare our approach to [26, 7, 8] on multiple medical segmentation applications in Section 5. Section 6 discusses space complexity and limitations of our method. Finally, Section 7 concludes.

2. Hierarchically-structured Interacting Segments

Given pixel set Ω, neighborhood system $\mathcal{N}$, and labels (regions) $\mathcal{L}$ the HINTS model can be formulated as

$$E(\mathbf{f}) = \overbrace{\sum_{p\in\Omega} D_p(f_p)}^{\textit{data}} + \lambda \overbrace{\sum_{pq\in\mathcal{N}} V_{pq}(f_p, f_q)}^{\textit{smoothness}} + \overbrace{T(\mathbf{f})}^{\substack{\textit{interaction}\\ \textit{constraints}}} \qquad (1)$$

where f_p is a label assigned to p and $\mathbf{f} = [f_p \in \mathcal{L} | \ \forall p \in \Omega]$ is a labeling of all pixels.

The *data* and *smoothness* terms are widely used in segmentation, e.g. [6, 4]. Data term $D_p(f_p)$ is the cost incurred when pixel p is assigned to label f_p. Usually, D_p is negative log likelihood of the label's probabilistic model, which for example is fitted using scribbles [4, 25] or known a priori.

[2] Tree with one node of degree ≥ 3 and all others with degree ≤ 2.

[3] Personal communication with the authors of [1].

The *smoothness* term regularizes segmentation discontinuities. A discontinuity occurs when two neighboring pixels $(p, q) \in \mathcal{N}$ are assigned to different labels. Parameter λ weights the importance of the smoothness term. The most commonly used *smoothness* potential is Potts model [7]. We use tree-metric smoothness [10, 11] which is more true to the physical structure of the labels in some settings, especially medical segmentation as we explain shortly.

A function V is tree-metric if there exists a tree with non-negative edge weights and $V(u, w)$ is equal to the sum of edge weights along the unique path between nodes u and w in the tree. In our setting $\mathcal{T}$ is such a tree and V_{pq} is completely defined by assigning non-negative weights to every edge in $\mathcal{T}$. Thus, for any α, β in $\mathcal{L}$

$$V_{pq}(\alpha, \beta) = \sum_{ij \in \Gamma(\alpha,\beta)} V_{pq}(i, j), \tag{2}$$

where $\Gamma(\alpha, \beta)$ is the set of ordered labels on the path between α and β in the undirected tree $\mathcal{T}$. The summation in (2) is between pairs of neighbouring labels on path $\Gamma(\alpha, \beta)$.

To motivate tree-metric smoothness consider $\mathcal{T}$ in Fig.1(a). This tree implies that regions R, A and D are nested. For example, R, A and D could be background, cell and nucleus, respectively. In the physical world boundaries of nested regions never merge into a single boundary. That is, if in the image we observe a boundary between D and R, this corresponds to two boundaries, namely D/A and A/R in the physical world. Therefore, the D/R boundary cost should be the sum of D/A and A/R boundary costs. The summation property of nested boundaries can be modeled as tree-metric smoothness. In contrast, Potts model penalizes multiple nested boundaries as a single boundary.

The *interaction* term in (1) ensures that the min-margin constraints are satisfied at every pixel, see Fig. 3,

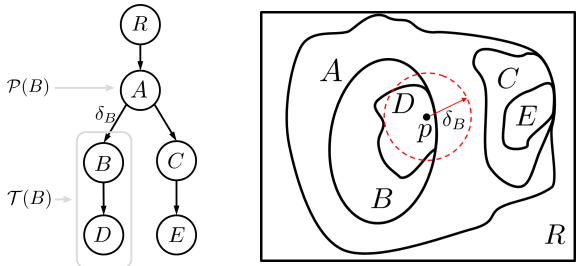
$$T(\mathbf{f}) = w_\infty \sum_{\ell \in \mathcal{L}} \sum_{\substack{p \in \Omega \\ f_p \in \mathcal{T}(\ell)}} \sum_{\substack{q \in \Omega \\ \|p-q\| < \delta_\ell}} [f_q \notin \{\mathcal{T}(\ell) \cup \mathcal{P}(\ell)\}] \tag{3}$$

where w_∞ is an infinitely large scalar, $\mathcal{T}(X)$ are the nodes of the subtree rooted at X, $\mathcal{P}(X)$ is X's parent in $\mathcal{T}$, and [ ] is the *Iverson bracket*[4]. This term guarantees that any labeling that violates min-margin constraint has infinite energy.

In general, the *interaction* term could model not only min-margin but also region attraction [8], scene parsing [22, 8], or a combination of these constraints. However, the focus of this paper is developing an effective combinatorial optimization move for energy (1). Thus, for simplicity of exposition we only cover min-margins.

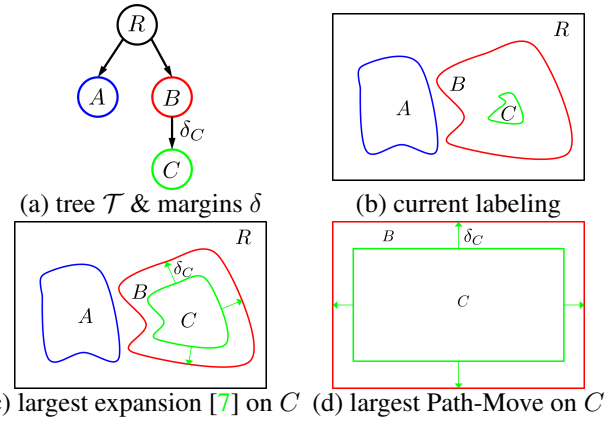
We now compare our formulation to that in [8]. *Inclusion* is an easy constraint to impose in both formulations as it reduces to using tree-metric smoothness. In our formulation *exclusion* is satisfied by definition because we use multi-label formulation and each pixel is assigned to only one label. In contrast, in [8] the label of a pixel is represented by several binary variables. Therefore, [8] needs to explicitly enforce exclusion to maintain the validity of these binary variables w.r.t. tree $\mathcal{T}$. Often this leads to non-submodular terms that are difficult to optimize.

[4][True]= 1 and [False]=0



(a) tree (b) min-margin constraint at pixel p

Figure 3. (a) tree $\mathcal{T}$ with 6 labels, $\mathcal{T}(B)$ is the subtree rooted at B and $\mathcal{P}(B)$ is B's parent. (b) visually illustrates the min-margin constraint δ_B at an arbitrary pixel p with label $f_p \in \mathcal{T}(B)$. For a labeling $\mathbf{f}$ to be valid w.r.t. min-margin δ_B; if $f_p \in \mathcal{T}(B)$ then any neighboring pixel q within δ_B pixels from p must be assigned to either B, one of its descendants or its parent, i.e. $f_q \in \{\mathcal{T}(B) \cup \mathcal{P}(B)\}$. Note that if q was assigned to either one of A's ancestors or B's siblings this means we encountered the outside boundary of A or B's siblings within the δ_B margin.



(a) tree $\mathcal{T}$ & margins δ (b) current labeling

(c) largest expansion [7] on C (d) largest Path-Move on C

Figure 4. (a) shows tree and margins. (b) shows the current labeling. (c) and (d) show the largest possible expansion of label C using binary expansion move [7] and our multi-label expansion move (Path-Move), respectively. Unlike [7], Path-Move is capable of pushing all regions' boundaries when expanding C without violating the interaction constraints.

3. Optimization

The authors in [8] showed that HINTS is non-submodular for a general tree $\mathcal{T}$ and they used either QPBO or a-exp for optimization. Unfortunately, QPBO does not guarantee to label all pixels and we observed that in our experiments, see Fig. 2. The a-exp algorithm [7] is guaranteed to label all pixels but prone to bad local minima, see Fig. 2.

We build on a-exp algorithm [7], which maintains a valid current labeling $\mathbf{f}'$ and iteratively tries to decrease the energy by switching from the current labeling to a nearby labeling via a *binary expansion* move. In a binary expansion, a label $\alpha \in \mathcal{L}$ is chosen randomly and allowed to expand. Each pixel is given a *binary* choice to either stay as f'_p or switch to α, i.e. $f_p \in \{f'_p, \alpha\}$. The algorithm stops when it cannot decrease the energy anymore.

A-EXPANSION ALGORITHM [7]

1 $\mathbf{f}' :=$ initial valid labeling
2 **repeat**
3 **for each** $\alpha \in \mathcal{L}$
4 $\mathbf{f}^\alpha := \arg\min_{\mathbf{f}} E(\mathbf{f})$ where $\mathbf{f}$ is an α-expansion of $\mathbf{f}'$
5 **if** $E(\mathbf{f}^\alpha) < E(\mathbf{f}')$
6 $\mathbf{f}' := \mathbf{f}^\alpha$
7 **until** converged

Due to the "binary" nature of the expansion move interaction constraints cause a-exp to be highly sensitive to initialization and more prone to converge to a bad local minima even for simple trees, see Fig. 4.

Instead of using a binary expansion move [7] in step 4 of the a-exp algorithm, we propose a more powerful "multi-label" move, namely, Path-Move. Figure 4(d) shows how robust a Path-Move is compared to a binary one [7].

3.1. Path-Move

In a Path-Move on α each pixel p can choose any label in the ordered set $\Gamma(f'_p, \alpha)$ where f'_p is the current label of p. Thus, the set of feasible labels for p is $\Gamma(f'_p, \alpha)$, see examples in Fig. 5.

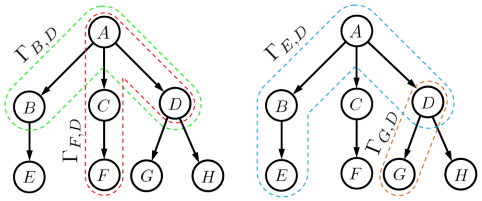
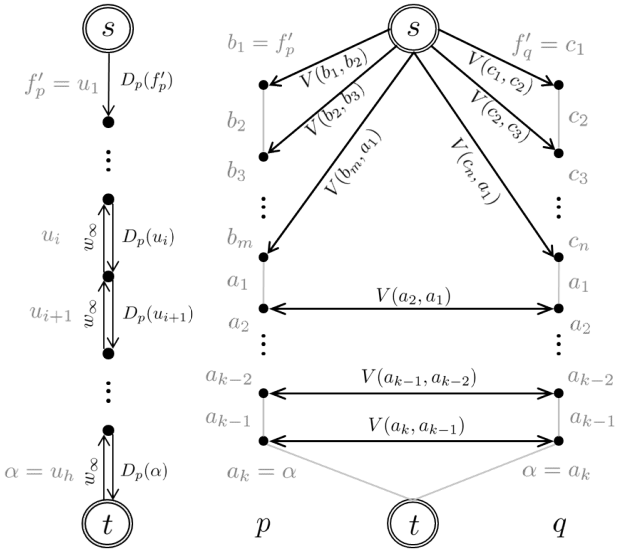


Figure 5. shows for some $\mathcal{T}$ the sets of feasible labels when expanding on D for pixels whose current labels are B (green), F (red), E (blue) and G (brown). Unlike Path-Move, in [16, 30] the feasible set of labels during an expansion is the same for all pixels.

Given an arbitrary $\mathcal{T}$, current labeling $\mathbf{f}'$ and label α, we now show how to build a graph such that the min-cut on this graph corresponds to the optimal Path-Move. We use s and t to denote source and sink nodes of the min-cut problem, respectively. Our construction is motivated by [16, 7, 30].

Data Term: For each pixel p we generate a chain of nodes $\mathcal{C}_p$ whose size is $|\Gamma(f'_p, \alpha)| - 1$. Let us rename $\Gamma(f'_p, \alpha)$ to $(u_1, u_2, \ldots, u_h)$ where $u_1 = f'_p$ and $u_h = \alpha$. Note that u_i and h depend on p but we drop explicit dependence on p from notation for clarity. Fig. 6(a) illustrates chain $\mathcal{C}_p$ and how it is linked to s and t. The edge weights along the



(a) data encoding (b) smoothness encoding

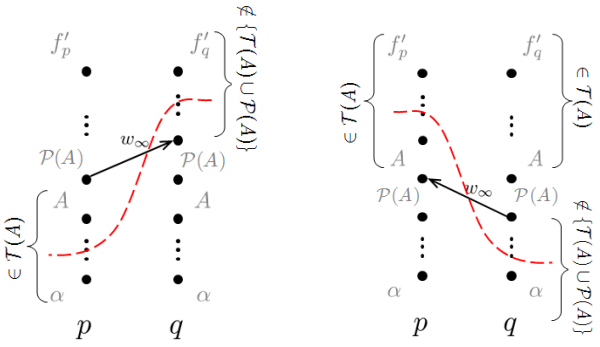
Figure 6. (a) shows the part of our graph that encodes the data term of pixel p. The black nodes represent $\mathcal{C}_p$. Next to each edge along $(s, \mathcal{C}_p, t)$ we show in light grey the label that pixel p is assigned to if that edge is cut. (b) shows the part of our graph that encodes the smoothness term for neighboring pixels p and q with current labels f'_p and f'_q, respectively. Grey edges in (b) are those ones illustrated in (a) but redrawn in (b) without their weights for clarity.

directed path $(s, \mathcal{C}_p, t)$ encode the the data terms of p while the weights along the opposite direction are w_∞. If the i^{th} edge along the $(s, \mathcal{C}_p, t)$ is cut, then pixel p is assigned to label u_i. The w_∞ edges ensure that any min-cut severs only one edge on the $(s, \mathcal{C}_p, t)$ path as proposed by [16]. Thus, the sum of severed edges on paths $(s, \mathcal{C}_p, t)$ for all pixels p adds to the data term in (1).

Smoothness: Let p and q be a pair of neighboring pixels. Note the overlap between $\Gamma(f'_p, \alpha)$ and $\Gamma(f'_q, \alpha)$ is at least one label, see Fig. 5. Our graph construction treats the sequence of overlapping labels of paths $\Gamma(f'_p, \alpha)$ and $\Gamma(f'_q, \alpha)$ differently from the non-overlapping parts. Therefore we rename $\Gamma(f'_p, \alpha) = (b_1, \ldots, b_m, a_1, \ldots, a_k)$ and $\Gamma(f'_q, \alpha) = (c_1, \ldots, c_n, a_1, \ldots, a_k)$ to emphasize the overlap. Figure 6(b) shows the newly added weighted edges that encode the smoothness penalty V_{pq}.

The overlapping part $(a_1, \ldots, a_k)$ forms a linear ordering for which the smoothness cost is encoded as proposed by [16]. The non-overlapping parts $(b_1, \ldots, b_m, a_1)$ and $(c_1, \ldots, c_m, a_1)$ each forms a linear ordering independent of the other, but extending $(a_1, \ldots, a_k)$ linear ordering. In this case the smoothness penalties are handled by additional links from the source. See [13] for proof of correctness.

Interaction Constraints: Let p and q be $\delta_A > 0$ within each other. As per energy (3), to impose the δ_A margin constraint between p and q we need to add edges to our graph to ensure that whenever $f_p \in \mathcal{T}(A)$ and $f_q \notin \{\mathcal{T}(A) \cup \mathcal{P}(A)\}$



(a) $\alpha \in \mathcal{T}(A)$ $f'_p, f'_q \notin \mathcal{T}(A)$ (b) $\alpha \notin \mathcal{T}(A)$ $f'_p, f'_q \in \mathcal{T}(A)$

Figure 7. (a) and (b) show the required w_∞ edge for the two main cases that occur when imposing δ_A. The red dashed curves illustrate prohibitively expensive cuts that violate δ_A constraint. For clarity we only show the newly added edges to our graph.

the corresponding energy is infinite. To impose such constraint there are several cases to consider depending on whether each of α, f'_p or f'_q is in $\mathcal{T}(A)$ or not as follows.

Scenario I when $\alpha \in \mathcal{T}(A)$:

Case 1, $f'_p \notin \mathcal{T}(A)$ and $f'_q \notin \mathcal{T}(A)$: in this case we can deduce that A and $\mathcal{P}(A)$ are both in $\Gamma(f'_p, \alpha)$ and $\Gamma(f'_q, \alpha)$. Thus, there is possibility of forbidden configurations and we handle them by adding a w_∞ edge as shown in Fig.7(a).

Case 2, $f'_q \in \mathcal{T}(A)$: in this case we can deduce that $\Gamma(f'_q, \alpha) \subseteq \mathcal{T}(A)$. Thus, additional edges are not needed since f_q is guaranteed to be in $\mathcal{T}(A)$.

Case 3, $f'_p \in \mathcal{T}(A)$ and $f'_q \notin \mathcal{T}(A)$: in this case we can deduce that $\mathcal{P}(A) \in \Gamma(f'_q, \alpha)$. If $f'_q = \mathcal{P}(A)$ then no additional edges needed since $f_q \in \{\mathcal{T}(A) \cup \mathcal{P}(A)\}$. The case $f'_q \neq \mathcal{P}(A)$ is not possible as the current labeling would violate the margin constraint.

Scenario II when $\alpha \notin \mathcal{T}(A)$:

Case 1, $f'_p \in \mathcal{T}(A)$ and $f'_q \in \mathcal{T}(A)$: this case follows the same reasoning as scenario I, case 1. To handle the forbidden configuration we add a w_∞ edge as shown in Fig.7(b).

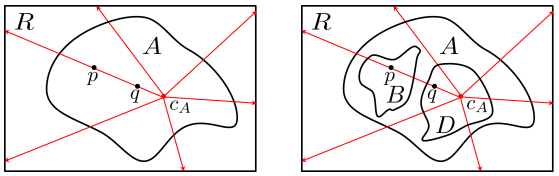
Case 2, $f'_p \notin \mathcal{T}(A)$: in this case we can deduce that $f_p \notin \mathcal{T}(A)$ and no new edges are needed, as we are only interested in the case when $f_p \in \mathcal{T}(A)$.

Case 3, $f'_p \in \mathcal{T}(A)$ and $f'_q \notin \mathcal{T}(A)$: here we can deduce that $f'_q = \mathcal{P}(A)$ otherwise the current labeling would violate δ_A. If $f'_q = \mathcal{P}(A)$ the construction is as shown in Fig.7(b) except there are no nodes above $\mathcal{P}(A)$ for q.

For a more thorough discussion and illustrations of the aforementioned cases the reader is referred to [13].

4. Shape Priors for HINTS

In this section we extend star-shape [29], Geodesic-star [12] and Hedgehogs [14] priors to the HINTS model and show how to enforce these priors during a Path-Move.



(a) star-shape prior [29] (b) star-shape prior + HINTS

Figure 8. (a) and (b) illustrate star-shape prior constraint for label A in binary and partially ordered segmentations, respectively. c_A denotes star-center. (a) and (b) show a valid star-shape for label A.

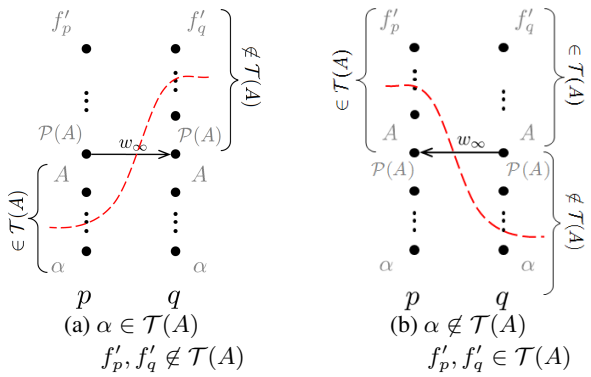
In the context of binary segmentation, star-shape prior [29] on label A with star center c_A reduces to the following constraint. If pixels p and q lie on any line originating from c_A with q in the middle and p is labeled A, then q must also be labeled A, see Fig.8(a). Geodesic-star [12] and Hedgehogs [14] differ from star-shape prior in terms of what defines the center and how lines from the center (or geodesic paths) are generated. Furthermore, Hedgehogs [14] allow control over shape constraint tightness, see [14] for details.

For partially ordered segments we generalize the star-shape prior constraint as follows. If pixels p and q lie on any line originating from c_A with q in the middle and f_p is in $\mathcal{T}(A)$, then f_q must also be in $\mathcal{T}(A)$, see Fig.8(b).

The shape prior penalty term is

$$S(\mathbf{f}) = w_\infty \sum_{\ell \in \mathcal{L}} \sum_{\substack{p \in \Omega \\ f_p \in \mathcal{T}(\ell)}} \sum_{pq \in \mathcal{S}_\ell} [f_q \notin \mathcal{T}(\ell)], \qquad (4)$$

where $\mathcal{S}_\ell$ is the set of all ordered pixel pairs[5] (p, q) along any line containing c_ℓ such that q is between p and c_ℓ. Using [12] or [14] instead of [29] results in a different $\mathcal{S}_\ell$.



(a) $\alpha \in \mathcal{T}(A)$ $f'_p, f'_q \notin \mathcal{T}(A)$ (b) $\alpha \notin \mathcal{T}(A)$ $f'_p, f'_q \in \mathcal{T}(A)$

Figure 9. Assume pixels p and q lie on a line originating from star-center c_A of label A, and that q lies between c_A and p. (a) and (b) show the two cases that require a w_∞ edge to impose the star-shape prior on Label A. The red dashed curves are prohibitively expensive cuts that violate the star-shape constraint.

Let pixels p and q lie on a line passing through c_A, and q is between c_A and p. To impose star-shape prior for label A during a Path-Move, there are multiple cases to consider depending on whether each of α, f'_p and f'_q is in $\mathcal{T}(A)$ or not.

[5] In practice, it is enough to include only consecutive pixel pairs in $\mathcal{S}_\ell$.

Scenario I: when $\alpha \in \mathcal{T}(A)$:

Case 1, $f'_p \notin \mathcal{T}(A)$ and $f'_q \notin \mathcal{T}(A)$: in this case we can deduce that A and $\mathcal{P}(A)$ are both in $\Gamma(f'_p, \alpha)$ and $\Gamma(f'_q, \alpha)$. Thus, there are possible forbidden configurations and to handle them we add an w_∞ edge as in Fig. 9(a).

Cases 2, $f'_q \in \mathcal{T}(A)$: we can deduce that $\Gamma(f'_q, \alpha) \subseteq \mathcal{T}(A)$. Thus, no additional edges are needed since f_q is guaranteed to be in $\mathcal{T}(A)$.

Case 3, $f'_p \in \mathcal{T}(A)$ and $f'_q \notin \mathcal{T}(A)$: impossible case as the current labeling would be violating the shape-prior.

Scenario II: when $\alpha \notin \mathcal{T}(A)$:

Case 1, $f'_p \in \mathcal{T}(A)$ and $f'_q \in \mathcal{T}(A)$: this case is similar to scenario I, case 1, the added edge is shown in Fig. 9(b).

Cases 2, $f'_p \notin \mathcal{T}(A)$: we can deduce that $\Gamma(f'_p, \alpha) \nsubseteq \mathcal{T}(A)$. Thus, no edge needed since f_p can not be in $\mathcal{T}(A)$.

Case 3, $f'_p \in \mathcal{T}(A)$ and $f'_q \notin \mathcal{T}(A)$: see case 3 above.

5. Experiments

Our 2D medical segmentation experiments focus on comparing Path-Moves for optimizing energy (1) or (1)+(4) to QPBO [26, 8] and a-exp [7, 8]. In all experiments λ was set to 1. To define our tree-metric, every edge (γ, β) in $\mathcal{T}$ was assigned a non-negative weight $V_{pq}(\gamma, \beta)$ computed using a non-increasing function of difference in p and q intensities similar to [2]. Also, whenever a Hedgehog [14] shape prior was used its tightness parameter was set to $\pi/9$.

The experiments evaluate the effectiveness of Path-Moves. As such we assume that color models are known a priori. One can easily integrate Path-Moves in a framework that estimates initial color models using user interaction and iteratively alternates between labeling and re-estimating color models in an EM fashion, e.g. [25, 9, 15].

Brain Segmentation: We combined the labeled regions in dataset [20] (T1W MRI) to create the tree shown in Fig. 10(a). In this setting, the data term is the sum of color model penalty and an L_2 shape prior [5] based on an automatically extracted brain mask using [17],

$$D_p(f_p) = \begin{cases} -\ln(Pr(I_p|f_p)) & \text{background} \\ -\ln(Pr(I_p|f_p)) + DT(p) & \text{otherwise,} \end{cases} \quad (5)$$

where I_p is the intensity at pixel p and DT is the Euclidean Distance Transform of the extracted brain mask. The min-margins are shown in Fig. 10(a). We also added a Hedgehog prior [14] for the sub-cortical grey-matter to help our energy differentiate between grey-matter and sub-cortical grey-matter. See [13] for results without shape prior.

In this application our method outperformed QPBO in most cases and a-exp in all cases. In fact a-exp always converged to a bad local minima. See Fig. 10 for results.

Figure 2 shows the results for Subject 1 when using min-margins and Hedgehog prior. Figure 11 (Top row) show the results for the same subject but without using min-margins. Path-Moves converged after two iterations to a lower energy than a-exp, which converged after six iterations. In this case a-exp local minimum was due to the Hedgehog prior. See Fig. 11 (Bottom row) for results without using min-margins or Hedgehog prior.

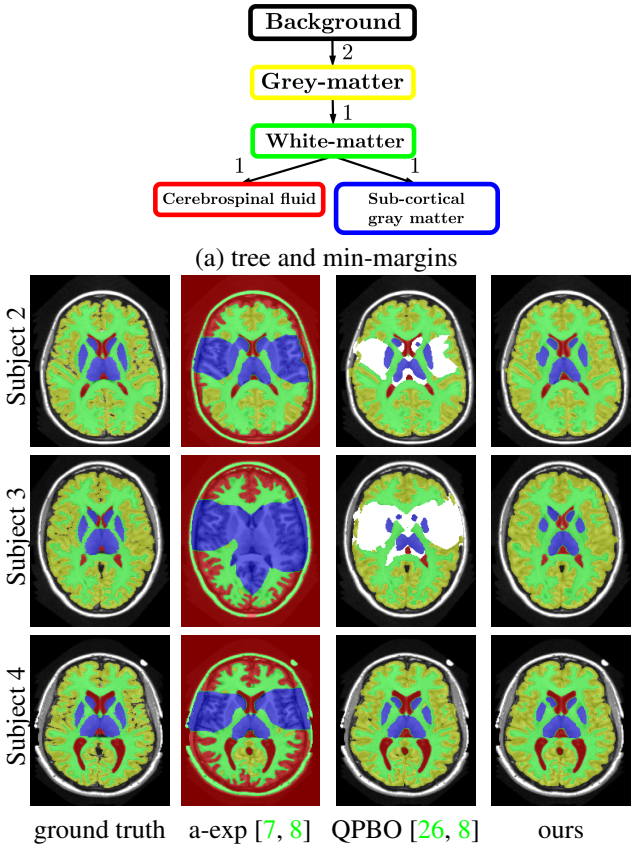


Figure 10. sample results when using tree in (a). An arc weight in (a) represent the min-margin of the head node. White pixels were unlabeled by QPBO.

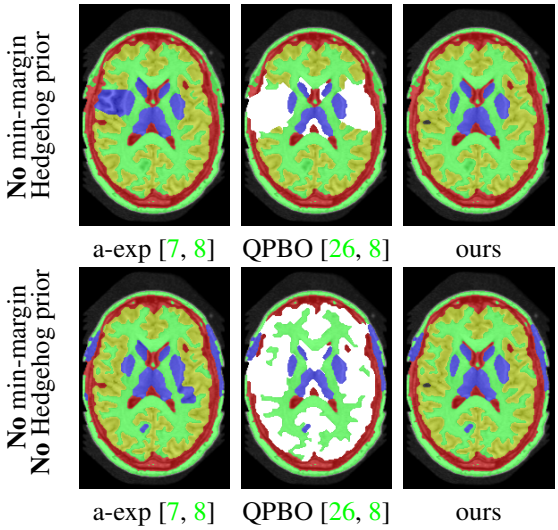


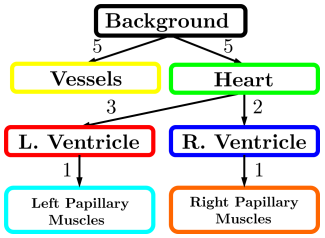
Figure 11. Subject 1 results (Top) without min-margins, (Bottom) without min-margins or Hedgehog prior.

	Grey-Matter			White-Matter			CSF			SGM		
	Ours	QPBO	a-exp	Ours	QPBO	a-exp	Ours	QPBO	a-exp	Ours	QPBO	a-exp
F_1 Score	**0.92**	0.83	0.32	**0.92**	0.9	0.56	**0.85**	0.82	0.04	**0.83**	0.81	0.37
Precision	0.87	0.87	0.88	0.92	0.92	0.46	0.78	0.83	0.02	0.92	0.93	0.23
Recall	0.97	0.80	0.19	0.93	0.88	0.74	0.93	0.82	0.56	0.76	0.71	0.92

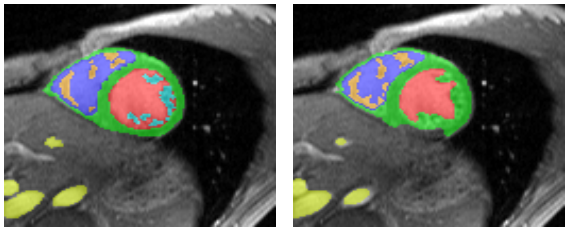
Table 1. Compares our Path-Moves optimization to QPBO [26] and a-exp [7] which were proposed by [8]. The precision and recall were averaged over 15 examples. Our method and QPBO clearly outperformed a-exp which was very sensitive to initialization and the order in which labels were expanded on. On average QPBO left 2.8% of the pixels unlabeled and in one instance 7%. These values rise significantly when not using the Hedgehog shape prior, see Fig. 11(Bottom) and [13] for more results.

Table 1 compares the precision, recall and F_1 score for each region individually, where $F_1 = 2\frac{precision \cdot recall}{precision + recall}$. The higher F_1 values corresponds to better segmentation.

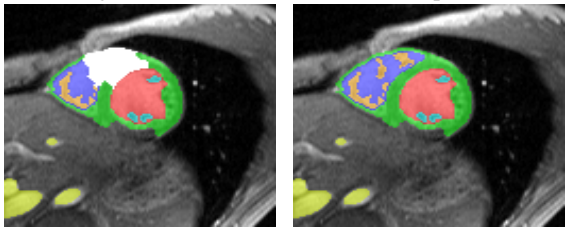
Heart Segmentation: In this setting we only used color models for the data term and no shape priors. Figure 12(a) shows the used tree. For a-exp to escape its local minimum it needs to first expand the left ventricle and then the left papillary muscles. However, expanding on left ventricle would lead to a higher energy than the current one. Path-Moves avoids this local minimum by allowing both labels to expand simultaneously when performing a Path-Move on the left papillary muscles.



(a) heart tree



(b) ground truth (c) a-exp[7, 8]



(d) QPBO [26, 8] (e) ours (Path-Moves)

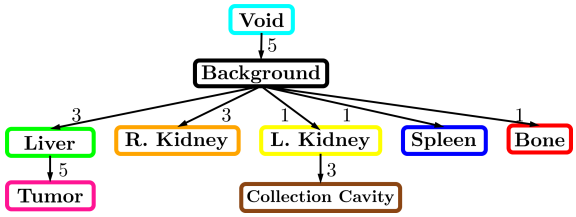
Figure 12. Heart segmentation using tree shown in (a). Using a-exp leads to a local minimum. Path-Moves avoids this minimum via multi-label expansions. QPBO leaves many pixels unlabeled.

Abdominal Organ Segmentation: We used a CT dataset and extended the work in [14] which used Hedgehogs to segment liver and kidneys. In contrast to [14], we utilized more detailed structures reaching 13 labels.

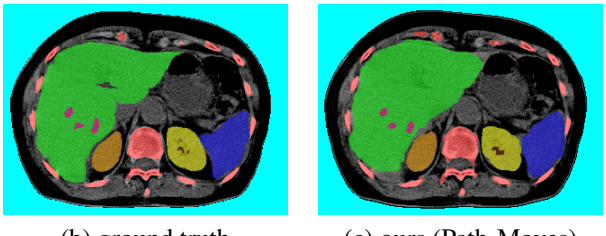
We computed for each example the weighted precision $\sum_{\ell \in \mathcal{L}} \frac{|\mathbf{f}^* = \ell|}{|\Omega|} \times \text{precision}_\ell$ where $\mathbf{f}^*$ is the ground truth labeling. The weighted recall is defined similarly. As shown in Table 2, all methods performed comparably due to the use of Hedgehog priors and the star-like tree of $\mathcal{T}$ that α-exp is well suited for. Figure 13 shows the tree and our result for one test case. Interestingly, QPBO labeled all the pixels in all 7 test cases.

	Ours	QPBO	a-exp
F_1 Score	**0.95**	**0.95**	0.93
Weighted Precision	0.95	0.95	0.94
Weighted Recall	0.95	0.95	0.92

Table 2. Weighted precision and recall were averaged over 7 test cases. When using shape priors all methods performed comparably. For results without shape priors see [13].

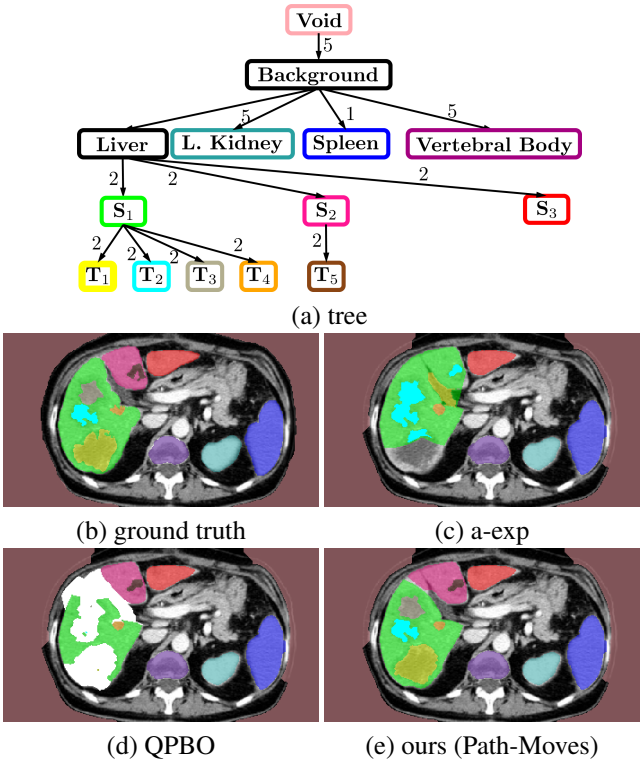


(a) tree



(b) ground truth (c) ours (Path-Moves)

Figure 13. (a) abdominal organs structure used for this example. Void is the empty space around the body. We only show our result as QPBO and a-exp results were almost identical to ours.



(a) tree

(b) ground truth (c) a-exp

(d) QPBO (e) ours (Path-Moves)

Figure 14. (a) a challenging abdominal organs structure. S_x and T_y denote liver segment x and tumor y, respectively. The liver label in (a) is a conceptual/artificial label with infinity data term penalty. Our method significantly outperformed QPBO and a-exp.

We pursued a more challenging structure, see Fig.14(a). The objective in this case was to segment the liver into three different segments and any tumors inside them separately. Due to the large overlap between color models and the complex structure, having hedgehog priors was not enough for QPBO or a-exp to converge to an adequate solution, see Fig.14(c-e). Path-Moves was able to achieve good results avoiding local minima as in Fig.14(c). Furthermore, Path-Moves always results in full labeling compared to QPBO, which left 7.4% of the pixels unlabeled, see Fig.14(d).

6. Discussion

Path-Moves is applicable to tree-metrics which could be used to approximate arbitrary metrics [19, 10]. Even in the absence of interactions, Path-Moves is a more powerful move making algorithm than a-exp [7] because of the *multi-label* nature of its moves. Thus, Path-Moves is a better fit for applications that rely on tree-metrics such as [19]. In the presence of interaction constraints the optimality bound of [7] is not valid. The proof in [7] assumes that given any labeling every pixel with ground truth label X could switch to X via a binary expansion on X. This is no longer guaranteed as interaction constraints limit [7] expansion domain, e.g. see Fig.4(c). Our experiments empirically show that Path-Moves finds optimal or near optimal solution. In the cases where QPBO found full labeling, i.e. optimal solution, Path-Moves either found the same solution or a very close one, see Table 2 and Fig.10 Subject 4.

In terms of space complexity a-exp is the most efficient as it requires building a graph with $O(|\Omega|)$ nodes while QPBO requires a significantly larger graph with $O(|\Omega||\mathcal{L}|)$ nodes. A Path-Move graph size depends on $\mathcal{T}$. When $\mathcal{T}$ is balanced it requires $O(|\Omega|\ln(|\mathcal{L}|))$ nodes and $O(|\Omega||\mathcal{L}|)$ in the worse case when $\mathcal{T}$ is a chain.

There is one limitation when using our Path-Moves to optimize (1) compared to [8]. In [8] it is possible to explicitly control the min. exclusion margin between two siblings, say A and B in $\mathcal{T}$. In our model the min. exclusion margin is implicit and it is equal to $\max(\delta_A, \delta_B)$. Because siblings such as A and B are not directly connected in the tree.

Another limitation are interaction constraints that are not Path-Move representable. An interaction constraint is not Path-Move representable if there exists α, β and $\gamma \in \mathcal{L}$ where $a < b \in \Gamma(\gamma, \alpha)$ and $c < d \in \Gamma(\beta, \alpha)$ while configuration $[a, d]$ is prohibited and $[b, c]$ is permissible [27], see [13] for illustration. In general, this could be avoided either by slightly modifying tree $\mathcal{T}$ or relaxing the interaction constraints, see [13] for an example.

7. Conclusion

The proposed multi-labeling move is effective in optimizing models with hierarchically-structured segments (partially ordered labels) and interaction constraints. In contrast to binary expansion move [7], our move avoids local minima caused by interaction constraints.

Our experiments cover various medical segmentation applications, e.g. brain and heart segmentation. Our results show that Path-Moves always perform at least as well as prior methods. Moreover, Path-Moves significantly outperform prior methods when using complex trees and/or regions with ambiguous color models.

Path-Moves is applicable to arbitrary trees. This is in contrast to [8] which is not easy to generalize for an arbitrary tree as it relies on the cumbersome process of reducing high-order data terms to unary and pairwise potentials.

We generalized star-like shape priors in the context of partially ordered labels. Extending preexisting commonly used priors to partially ordered labels is an interesting idea on its own and we leave this for future work.

Acknowledgments This work was supported by NIH grants R01-EB004640, P50-CA174521 and R01-CA167632. We thank Drs. S. O'Dorisio and Y. Menda for providing the liver data (NIH grant U01-CA140206). This work was also supported by NSERC Discovery and RTI grants (Canada) for Y. Boykov and O. Veksler.

References

[1] J. S. H. Baxter, M. Rajchl, A. J. McLeod, J. Yuan, and T. M. Peters. Directed acyclic graph continuous max-flow image segmentation for unconstrained label orderings. *International Journal of Computer Vision*, pages 1–20, 2017. 2

[2] Y. Boykov and G. Funka-Lea. Graph cuts and efficient N-D image segmentation. *International Journal of Computer Vision (IJCV)*, 70(2):109–131, 2006. 6

[3] Y. Boykov, H. Isack, C. Olsson, and I. Ben Ayed. Volumetric bias in segmentation and reconstruction: Secrets and solutions. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015. 1

[4] Y. Boykov and M.-P. Jolly. *Interactive graph cuts* for optimal boundary & region segmentation of objects in N-D images. In *ICCV*, volume I, pages 105–112, July 2001. 2

[5] Y. Boykov, V. Kolmogorov, D. Cremers, and A. Delong. An integral solution to surface evolution PDEs via geo-cuts. In *European Conference on Computer Vision (ECCV)*, Graz, Austria, May 2006. 6

[6] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. In *ICCV*, volume I, pages 377–384, 1999. 2

[7] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, November 2001. 1, 2, 3, 4, 6, 7, 8

[8] A. Delong and Y. Boykov. Globally Optimal Segmentation of Multi-Region Objects. In *International Conference on Computer Vision (ICCV)*, 2009. 1, 2, 3, 6, 7, 8

[9] A. Delong, A. Osokin, H. Isack, and Y. Boykov. Fast Approximate Energy Minization with Label Costs. *International Journal of Computer Vision (IJCV)*, 96(1):1–27, January 2012. 6

[10] P. F. Felzenszwalb, G. Pap, E. Tardos, and R. Zabih. Globally optimal pixel labeling algorithms for tree metrics. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3153–3160. IEEE, 2010. 2, 3, 8

[11] I. Gridchyn and V. Kolmogorov. Potts model, parametric maxflow and k-submodular functions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2320–2327, 2013. 3

[12] V. Gulshan, C. Rother, A. Criminisi, A. Blake, and A. Zisserman. Geodesic star convexity for interactive image segmentation. In *CVPR, 2010 IEEE Conference on*, pages 3129–3136. IEEE, 2010. 2, 5

[13] H. Isack, O. Veksler, I. Oguz, M. Sonka, and Y. Boykov. Efficient optimization for hierarchically-structured interacting segments (HINTS). *arXiv:1703.10530*, 2017. 4, 5, 6, 7, 8

[14] H. Isack, O. Veksler, M. Sonka, and Y. Boykov. Hedgehog shape priors for multi-object segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2, 5, 6, 7

[15] H. N. Isack and Y. Boykov. Energy-based Geometric Multi-Model Fitting. *International Journal of Computer Vision (IJCV)*, 97(2):123–147, April 2012. 6

[16] H. Ishikawa. Exact optimization for Markov Random Fields with convex priors. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 25(10):1333–1336, 2003. 2, 4

[17] M. Jenkinson, M. Pechaud, and S. Smith. Bet2: Mr-based estimation of brain, skull and scalp surfaces. In *Eleventh annual meeting of the organization for human brain mapping*, volume 17, page 167. Toronto, ON, 2005. 6

[18] J. Kleinberg and E. Tardos. Approximation algorithms for classification problems with pairwise relationships: Metric labeling and markov random fields. *J. ACM*, 49(5):616–639, Sept. 2002. 2

[19] M. P. Kumar and D. Koller. Map estimation of semi-metric mrfs via hierarchical graph cuts. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, pages 313–320, Arlington, Virginia, United States, 2009. AUAI Press. 8

[20] B. Landman and S. Warfield. Miccai workshop on. In *Multi-Atlas Labeling*, 2012. 6

[21] K. Li, X. Wu, D. Z. Chen, and M. Sonka. Optimal surface segmentation in volumetric images-a graph-theoretic approach. *IEEE transactions on pattern analysis and machine intelligence*, 28(1):119–134, 2006. 1, 2

[22] X. Liu, O. Veksler, and J. Samarabandu. Graph cut with ordering constraints on labels and its applications. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 3

[23] A. Lucchi, C. Becker, P. M. Neila, and P. Fua. Exploiting enclosing membranes and contextual cues for mitochondria segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 65–72. Springer, 2014. 1

[24] 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*, 33(6):1220–1235, 2014. 1

[25] C. Rother, V. Kolmogorov, and A. Blake. Grabcut - interactive foreground extraction using iterated graph cuts. In *ACM transactions on Graphics (SIGGRAPH)*, August 2004. 2, 6

[26] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer. Optimizing binary mrfs via extended roof duality. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007. 1, 2, 6, 7

[27] D. Schlesinger and B. Flach. *Transforming arbitrary minsum problem into a binary one*. TU, Fak. Informatik, 2006. 8

[28] J. Ulén, P. Strandmark, and F. Kahl. An efficient optimization framework for multi-region segmentation based on lagrangian duality. *IEEE transactions on medical imaging*, 32(2):178–188, 2013. 2

[29] O. Veksler. Star shape prior for graph-cut image segmentation. In *European Conference on Computer Vision (ECCV)*, 2008. 1, 2, 5

[30] O. Veksler. Multi-label moves for mrfs with truncated convex priors. *IJCV*, 98(1):1–14, 2012. 2, 4

[31] 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*, 29(12):2023–2037, 2010. 1, 2