Optimizing Partition Trees for Multi-Object Segmentation with Shape Prior

Emmanuel Maggiori[1]
emmanuel.maggiori@inria.fr
Yuliya Tarabalka[1]
yuliya.tarabalka@inria.fr
Guillaume Charpiat[2]
guillaume.charpiat@inria.fr

[1] Inria Sophia Antipolis Méditerranée
Titane team
Sophia Antipolis, France

[2] Inria Saclay
TAO team
Orsay, France

In this supplementary material we include:

- The **proofs** of the different properties referred to in the main manuscript.
- A **video file** showing the evolution of the segmentation map during the proposed optimization procedure for the image over New York City ($k = 30$). Each frame reflects the application of one move. Different nuances of the same color have been randomly assigned to the objects of every class.

We first generalize and develop the proofs of the energy formulation referred to in Section 2 of the manuscript. We then elaborate the proofs of the propositions regarding the space of moves in the optimization approach. Propositions 1-2 coincide with Propositions 1-2 in the main manuscript. In addition, Proposition 3 proves an additional property of the moves (referred to in Section 5.1), and Proposition 4 establishes a bound on the total number of possible moves on a BPT (mentioned in Section 5). The proofs of the space and computational complexities of including convex hulls in Binary Partition Trees (BPTs) (mentioned in Section 3) are here added as propositions 5 and 6, respectively.

1 Energy formulation

1.1 Total energy

Let us call $P(L|I,S)$ the probability of observing a label L, given color and shape features (I and $\mathbf{S}$, respectively):

$$P(L|I,\mathbf{S}) = \frac{P(L,I,\mathbf{S})}{P(I,\mathbf{S})} = \frac{P(I,\mathbf{S}|L)P(L)}{P(I,\mathbf{S})}. \quad (1)$$

Assuming conditional independence of the shape and color observations with respect to the label ($P(I,\mathbf{S}|L) = P(I|L)P(\mathbf{S}|L)$), and independence between color and shape ($P(I,\mathbf{S}) = P(I)P(\mathbf{S})$), we have:

$$P(L|I,\mathbf{S}) = \frac{P(I|L)P(\mathbf{S}|L)P(L)}{P(I,\mathbf{S})} = \frac{P(I|L)P(\mathbf{S}|L)P(L)}{P(I)P(\mathbf{S})}, \quad (2)$$

which in turn leads to the following formulation:

$$P(L|I,\mathbf{S}) = \frac{P(L|I)P(L|\mathbf{S})}{P(L)}. \tag{3}$$

We derive our energy as the negative log-likelihood of (3), adding the contribution of all pixels in the image. If we express the total energy in a per-pixel basis, we obtain:

$$E(\mathcal{R},L) = -\sum_{j=1}^{n} \Big(\log P(L_{R(j)}|I_j) + \log P(L_{R(j)}|\mathbf{S}_{R(j)}) - \log P(L_{R(j)}) \Big). \tag{4}$$

Alternatively, observing that all pixels in a segmented region share the same label, we can group them and sum in a per-region basis. This leads to the following equivalent formulation:

$$E(\mathcal{R},L) = -\sum_{i=1}^{|\mathcal{R}|} \Big(\sum_{j \in R_i} \log P(L_i|I_j) + |R_i| \log P(L_i|\mathbf{S}_i) - |R_i| \log P(L_i) \Big). \tag{5}$$

Let us remark that there are three terms in (4)/(5): a color prior, a shape prior and a label likelihood. In the main manuscript (Equations 1 and 5) we considered equal class probabilities and ignored the third term in the energy.

1.2 Shape prior

Let us call $P(L|\mathbf{S})$ the probability of observing label L given the vector of shape features $\mathbf{S}$. Using the same derivation as to go from (1) to (3), and assuming the appropriate independences, we can express:

$$P(L|\mathbf{S}) = P(L)^{1-m} \prod_{k=1}^{m} P(L|s_k) = P(L)^{1-m} \prod_{k=1}^{m} \frac{p(s_k|L)P(L)}{\sum_{L_j \in \mathcal{L}} p(s_k|L_j)P(L_j)}. \tag{6}$$

Considering equal class probabilities ($P(L_j) = \frac{1}{m}, \forall j$), the previous equation simplifies to the expression in the main manuscript (Equation 4).

2 Properties of *prune-and-paste* moves

Proposition 1. *Given a tree τ, suppose a node R_m is pasted at $\tau_i < \tau_1$ leading to a new tree φ. Let us consider an alternative move that pastes R_m at τ_j, with $\tau_i < \tau_j < \tau_1$, producing a tree ψ. In the cases where either $C(\varphi_1) - C(\tau_1) \leqslant 0$ or $C(R_m) \geqslant C(\varphi_1) - C(\tau_1)$, then $C(\psi_1) \geqslant C(\varphi_1)$.*

Proof. Let us abbreviate $\mathcal{E}(\tau_i)$ as e_i^τ and $C(\tau_i)$ as c_i^τ. Following (9) in the main manuscript,

$$c_1^\tau = min(e_1^\tau, \overline{c_2^\tau} + min(e_2^\tau, \overline{c_3^\tau} + min(...min(e_{i-2}^\tau, \overline{c_{i-1}^\tau} + min(e_{i-1}^\tau, \overline{c_i^\tau} + c_i^\tau))...))), \tag{7}$$

where $\overline{c_i^\tau}$ denotes the sibling of c_i^τ (see Fig. 2b in the main manuscript). This implies:

$$\underbrace{\overset{i-1}{\underset{k=1}{\forall}} \left(e_k^\tau \geqslant c_1^\tau - \Sigma_{j=2}^{k} \overline{c_j^\tau} \right)}_{\alpha} \text{ and } \left(c_i^\tau \geqslant c_1^\tau - \Sigma_{j=2}^{i} \overline{c_j^\tau} \right). \tag{8}$$

Let us now suppose that a node R_m is pasted at τ_i. Then

$$c_1^\varphi = min(e_1^\varphi, \overline{c_2^\tau} + min(e_2^\varphi, \overline{c_3^\tau} + min(...min(e_{i-2}^\varphi, \overline{c_{i-1}^\tau} + min(e_{i-1}^\varphi, \overline{c_i^\tau} + min(e_x^\varphi, c_i^\tau + c_m^R)))...))), \tag{9}$$

which implies:

$$\underbrace{\overset{i-1}{\underset{k=1}{\forall}} \left(e_k^\varphi \geqslant c_1^\varphi - \Sigma_{j=2}^k \overline{c_j^\tau}\right)}_{\beta} \text{ and } \left(e_x^\varphi \geqslant c_1^\varphi - \Sigma_{j=2}^i \overline{c_j^\tau}\right) \text{ and } \underbrace{\left(c_m^R \geqslant c_1^\varphi - \Sigma_{j=2}^i \overline{c_j^\tau} - c_i^\tau\right)}_{\gamma}. \tag{10}$$

Let us now paste R_m one position upper than before. We wish to check if it is possible that this move will be better that the previous one ($c_1^\psi < c_1^\varphi$):

$$\begin{aligned} c_1^\psi = & min(e_1^\varphi, \overline{c_2^\tau} + min(e_2^\varphi, \overline{c_3^\tau} + min(...min(e_{i-2}^\varphi, \overline{c_{i-1}^\tau} + min(e_y^\psi, \\ & c_m^R + min(e_{i-1}^\tau, \overline{c_i^\tau} + c_i^\tau)))...)) < c_1^\varphi. \end{aligned} \tag{11}$$

This can be true if and only if:

$$\begin{aligned} & \underbrace{\overset{i-2}{\underset{k=1}{\exists}} \left(e_k^\varphi < c_1^\varphi - \Sigma_{j=2}^k \overline{c_j^\tau}\right)}_{I} \text{ or } \underbrace{\left(e_y^\psi < c_1^\varphi - \Sigma_{j=2}^{i-1} \overline{c_j^\tau}\right)}_{II} \\ & \text{or } \underbrace{\left(e_{i-1}^\tau < c_1^\varphi - \Sigma_{j=2}^{i-1} \overline{c_j^\tau} - c_m^R\right)}_{III} \text{ or } \underbrace{\left(c_m^\tau < c_1^\varphi - \Sigma_{j=2}^i \overline{c_j^\tau} - c_i^\tau\right)}_{IV}. \end{aligned} \tag{12}$$

In this expression, *I* contradicts β. Considering that $e_y^\psi = e_{i-1}^\varphi$ (see Fig. 2b), the term *II* also contradicts β. The term *IV* contradicts γ. We must now analyze *III*. By combining *III* and α:

$$c_1^\tau - \Sigma_{j=2}^{i-1} \overline{c_j^\tau} \leqslant e_{i-1}^\tau < c_1^\varphi - \Sigma_{j=2}^{i-1} - c_m^R \Rightarrow c_m^R < c_1^\varphi - c_1^\tau. \tag{13}$$

If $c_1^\varphi - c_1^\tau \leqslant 0$, then c_m^R must be non-positive, which contradicts our hypothesis. If $c_1^\varphi - c_1^\tau > 0$: then it must be $c_m^R < c_1^\varphi - c_1^\tau$. As a conclusion, if the first move decreases C, *III* is contradicted, hence it must be $c_1^\psi \geqslant c_1^\varphi$. For a positive gain, *III* is contradicted unless $c_m^R < c_1^\varphi - c_1^\tau$. □

Proposition 2. *Let us consider a case where Prop. 1 hypotheses do not apply. There might then exist a higher paste place τ_α so that $C(\psi_1) < C(\varphi_1)$. Let us suppose that instead of pasting at τ_α we paste at τ_β, with $\tau_\alpha < \tau_\beta < \tau_1$, leading to a tree ρ. Then $C(\rho_1)$ would monotonously decrease as the paste place τ_β is located higher.*

Proof. If Prop. 1 hypotheses do not apply, then the term *III* in its proof must be true. This term implies that when pasting at τ_j, the cut on the tree will be located at or below R_m. The cost c_1^ψ associated with this move will then be

$$c_1^\psi = \Sigma_{j=2}^i \overline{c_j^\tau} + c_i^\tau + c_m^R, \tag{14}$$

considering the location of the new cut. Analogously, the cost when pasting R_m k units up of i is:

$$c_1^\rho = \Sigma_{j=2}^{i-k} \overline{c_j^\tau} + c_{i-k}^\tau + c_m^R. \tag{15}$$

Notice in the previous expression that we consider that the cut is still as low as R_m. The cut could not be higher, because if it were the case, then it would have already been cut there before.

Let us now see if the cost c_1^ρ could increase as k advances:

$$\begin{aligned} &\Sigma_{j=2}^{i-k}\overline{c_j^\tau} + c_{i-k}^\tau + c_m^R - (\Sigma_{j=2}^{i-k+1}\overline{c_j^\tau} + c_{i-k+1}^\tau + c_m^R) \\ =& -\overline{c_{i-k+1}^\tau} + c_{i-k}^\tau - c_{i-k+1}^\tau > 0 \\ \Leftrightarrow& c_{i-k}^\tau > c_{i-k+1}^\tau + \overline{c_{i-k+1}^\tau}, \end{aligned} \tag{16}$$

contradicting the algorithm to compute the cuts, hence the cost at R_1 must monotonously decrease. □

Proposition 3. *Let us suppose we paste R_m at or over the initial cut of tree τ, leading to tree φ. Let us consider we paste higher instead, producing tree ψ. It must then be $c_1^\psi \geqslant c_1^\varphi$.*

Proof. Let us resume the proof of Proposition 2 in the paper. It was shown that $c_1^\psi < c_1^\varphi$ if and only if *III* was false. At that point we could not contradict *III* but show that under certain conditions it would be contradicted. Now we will show that the fact that we know the first cut was at or below τ_i will contradict *III*.

After *III* and γ we have:

$$\begin{aligned} &e_{i-1}^\tau + c_1^\varphi - \Sigma_{j=2}^{i}\overline{c_j^\tau} - c_i^\tau \leqslant e_{i-1}^\tau + c_m^R < c_1^\varphi - \Sigma_{j=2}^{i-1}\overline{c_j^\tau} \\ \Leftrightarrow& e_{i-1}\tau - \overline{c_i^\tau} - c_i^\tau \leqslant e_{i-1}^\tau + c_m^R < 0. \end{aligned} \tag{17}$$

If we now add the knowledge about the cut being below τ_{i-1}, it must be $e_{i-1}^\tau > c_i^\tau + \overline{c_i^\tau}$, then

$$c_i^\tau + \overline{c_i^\tau} - \overline{c_i^\tau} - c_i^\tau < e_{i-1}^\tau - \overline{c_i}^\tau - c_i^\tau \leqslant e_{i-1}^\tau + c_m^R < 0 \Leftrightarrow 0 < 0. \tag{18}$$

Therefore, *III* cannot be true, which proves the proposition.

□

Corollary: Combining Proposition 2 in the paper, where C monotonously decreases ($\leqslant$) as the paste location gets higher, and Proposition 3, where C cannot decrease ($\geqslant$), it becomes evident that pasting a node anywhere between the initial cut and the lowest common ancestor produces the same effect on the energy.

Proposition 4. *The amount of spatially adjacent regions in a balanced BPT is bounded by $O(n\log(n))$.*

Proof. Let us call $\mathcal{N}_R$ the number of neighbors of the region R. At the lowest scale and in a discrete environment we can suppose that the number of neighbors is equal to its boundary length (δR). We are interested in knowing the number of neighbors at all scales. In a balanced tree it can be assumed that the number of neighbors at every scale is half the number at the following one. As a result:

$$\mathcal{N}_R = \delta R + \frac{1}{2}\delta R + \frac{1}{2^2}\delta R + \frac{1}{2^3}\delta R + ... < 2\delta R. \tag{19}$$

In a discrete implementation (assuming 4-connectivity):

$$\mathcal{N}_R < 2\delta R \leqslant 2 \cdot 4|R| = 8|R|. \tag{20}$$

The summation of the neighbors of *all* regions in a tree $\mathcal{T}$ is then

$$\sum_{R_i \in \mathcal{T}} \mathcal{N}_{R_i} < 8 \sum_{R_i \in \mathcal{T}} |R_i|. \tag{21}$$

Following (24), the total number of neighbors (the possible cut/paste moves) is a factor of $n\log(n)$. □

3 Complexity of incorporating convex hulls

Proposition 5. *The storage space required to add the convex hull to every node of a balanced BPT in a discrete environment is bounded by $O(n\log(n))$.*

Proof. Let us call $CH(R)$ the convex hull of a region R. In the extreme case (the most compact region), $CH(R)$ can be as large as the perimeter δR of R which, in a discrete implementation (assuming 4-connectivity) does not contain more points than four times the area of the region:

$$CH(R) \leqslant \delta R \leqslant 4|R|. \tag{22}$$

As a consequence, the points of the convex hull of *all* regions in a tree $\mathcal{T}$ must be

$$\sum_{R_i \in \mathcal{T}} |CH(R_i)| \leqslant 4 \sum_{R_i \in \mathcal{T}} |R_i|. \tag{23}$$

If we observe that in a balanced tree

$$\sum_{R_i \in \mathcal{T}} |R_i| = \sum_{l=1}^{\#levels} \sum_{R_j \in l \in \mathcal{T}} |R_j| = \sum_{l=1}^{\#levels} n = n \sum_{l=1}^{\#levels} 1 = n \cdot \#levels = n\log(n), \tag{24}$$

then (23) is bounded by a factor of $n\log(n)$.

□

Proposition 6. *The complexity of computing the convex hull of every region represented in a balanced BPT in a discrete environment, is bounded by $O(n\log(n))$.*

Proof. Let us call $CH(R)$ the convex hull of a region R. In the extreme case (the most compact region), $CH(R)$ can be as large as the perimeter δR of R which, in a discrete implementation (assuming 4-connectivity) does not contain more points than four times the area of the region:

$$CH(R) \leqslant \delta R \leqslant 4|R|. \tag{25}$$

The time to compute $CH(R_i)$ is linear on the number of points in the polygons of the children (see [35] in the article):

$$O(|\delta LeftChild(R_i)| + |\delta RightChild(R_i)|). \tag{26}$$

The time to compute the convex hull of every node in the tree is then bounded by a factor of:

$$\sum_{R_i \in \mathcal{T}} (|\delta LeftChild(R_i)| + |\delta RightChild(R_i)|) \leqslant 4 \sum_{R_i \in \mathcal{T}} (|LeftChild(R_i)| + |RightChild(R_i)|) = 4 \tag{27}$$

Following (24), the execution time is then a factor of $n\log(n)$. □