Selection of relevant nodes from component-trees in linear time

Nicolas Passat [(a)] and Benoît Naegel [(b)]

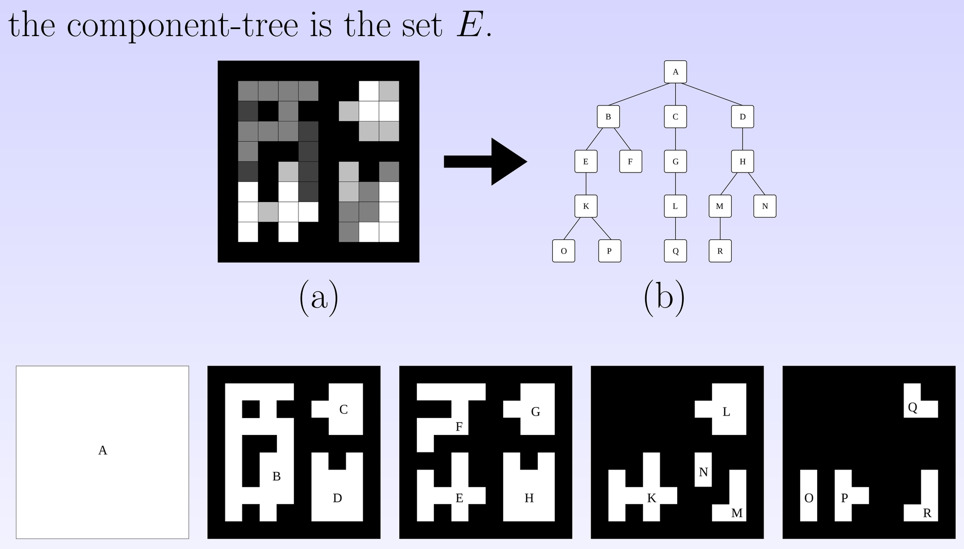
(a) Université de Strasbourg, LSIIT, UMR CNRS 7005, France
(b) Université Nancy 1, LORIA, UMR CNRS 7503, France

Abstract

Component-trees associate to a discrete grey-level image a descriptive data structure induced by the inclusion relation between the binary components obtained at successive level-sets. In this work we propose a method to extract a subset of the component-tree of an image enabling to fit at best a given binary target selected beforehand in the image. A proof of the algorithmic efficiency of this method is proposed. Application examples related to the extraction of drop caps from ancient documents emphasise the usefulness of this technique in the context of assisted segmentation.

Component-tree

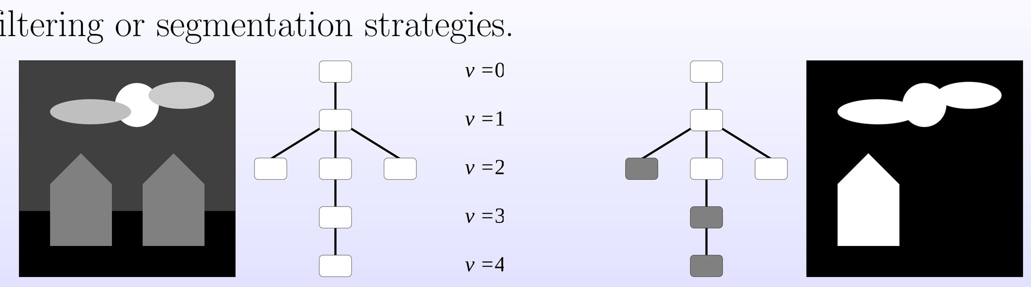
- A component-tree associates to a (discrete) grey-level image a descriptive data structure induced by the inclusion relation between the binary components obtained at successive level-sets.
- A **threshold set** of an image $I : E \to V$ at level v is defined as: $X_v(I) = \{p \in E \mid I(p) \geq v\}$.
- A **node** of the component-tree is a connected component of $X_v(F)$.
- A node N_1 is an **ancestor** of N_2 if $N_2 \subset N_1$.
- The **root** of the component-tree is the set E.



(a) (b) (c) (d) (e) (f) (g)

FIGURE 1: A grey-level image I (a), its successive threshold sets $X_v(I)$ for v from 0 to 4 (c-g), and its component-tree T (b).

- By selecting relevant nodes, component-trees can be used to develop image processing operators based on filtering or segmentation strategies.



(a) (b) (c) (d)

FIGURE 2: (a) Example image and (b) its component-tree T. (c) In grey: subset $\mathcal{K}'$ of nodes selected from T. (d) The associated binary image $\bigcup_{N \in \mathcal{K}'} N$.

Problem to solve

- Let I a grey-level image, T its component-tree T and G a binary marker (for instance a rough segmentation of the image).
- Which is the subset of nodes of T enabling to generate a binary object being as similar as possible to the target G?

- This problem can be summarised as a minimisation problem, consisting of determining:

$$\widehat{\mathcal{K}} = \arg \min_{\mathcal{K}' \in \mathcal{P}(\mathcal{K})} \{d(\bigcup_{N \in \mathcal{K}'} N, G)\},$$

where:

- $\mathcal{K}$ is the set of nodes of T (connected-components of the threshold sets of I);
- Given a parameter $\alpha \in [0,1]$, d is a pseudo-distance that takes account of the amount of false-positives/negatives between G and $\bigcup_{N \in \mathcal{K}'} N$:

$$d^{\alpha}(X,Y) = \alpha.|X \setminus Y| + (1-\alpha).|Y \setminus X|.$$

- Let $\mathcal{F}^{\alpha}$ and c^{α} be the functions recursively cross-defined, for all $N \in \mathcal{K}$, by:

$$(\mathcal{F}^{\alpha}(N), c^{\alpha}(N)) = \begin{cases} (\{N\}, \alpha.n(N,G)) & \text{if } \alpha.n(N,G) \prec (1-\alpha).p^*(N,G) + \sum_{N' \in ch(N)} c^{\alpha}(N') \\ (\bigcup_{N' \in ch(N)} \mathcal{F}^{\alpha}(N'), (1-\alpha).p^*(N,G) + \sum_{N' \in ch(N)} c^{\alpha}(N')) & \text{otherwise} \end{cases}$$

where:

- $ch(N)$ is the set of children of N;
- $n(N,G) = |N \setminus G|$ (the number of points of N which do not belong to G);
- $p^*(N,G) = |(N \setminus \bigcup_{N' \in ch(N)} N') \cap G|$ (the number of points of N which belong to G and which do not belong to any children of N).

- The set of nodes $F^{\alpha}(E)$ enables to minimize $d^{\alpha}(.,G)$;
- $\mathcal{F}^{\alpha}(E)$ can be computed with an algorithmic complexity $\mathcal{O}(\max\{|\mathcal{K}|, |E|\})$, linear with respect to the number of nodes of the tree or the size of the image.

Interactive segmentation method

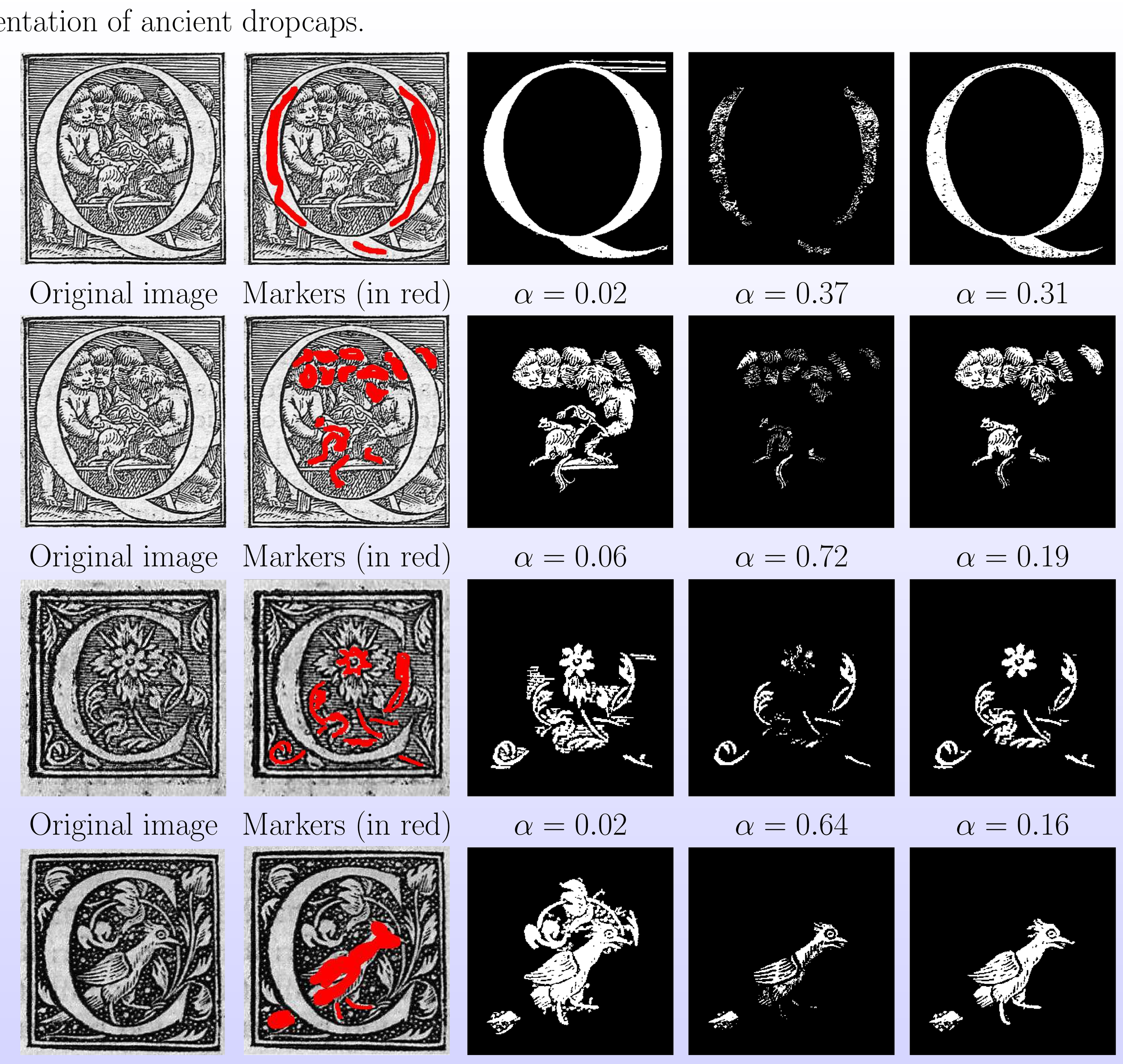
- An interactive segmentation method has been designed based on this concept[a].
- The segmentation method is based on an iterative processus:

1. Interactive drawing of the marker set (the binary image G);
2. Automatic computation of component-tree;
3. Interactive choice of α parameter, which defines the distance between marker set and component-tree nodes;
4. To refine the result, the marker can be updated.

[a] Software freely available at `http://webloria.loria.fr/~naegelbe/index.php/software`

Results

- Segmentation of ancient dropcaps.

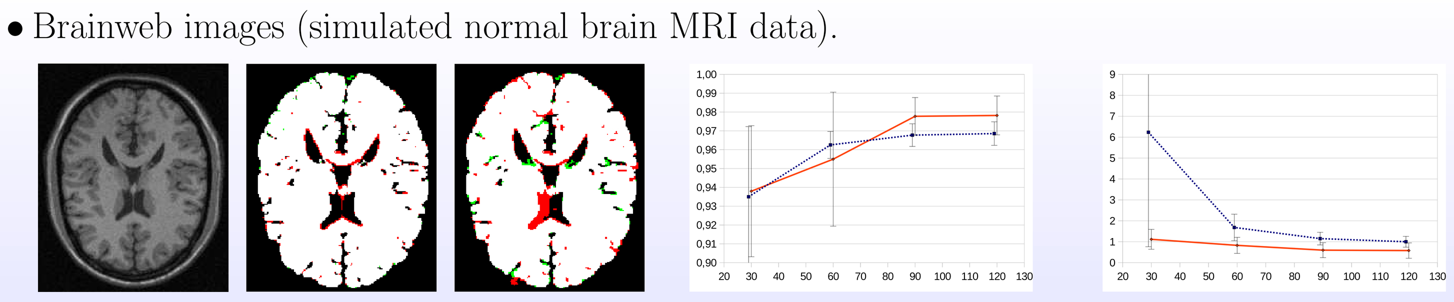


Original image | Markers (in red) | $\alpha = 0.02$ | $\alpha = 0.37$ | $\alpha = 0.31$
Original image | Markers (in red) | $\alpha = 0.06$ | $\alpha = 0.72$ | $\alpha = 0.19$
Original image | Markers (in red) | $\alpha = 0.02$ | $\alpha = 0.64$ | $\alpha = 0.16$
Original image | Markers (in red) | $\alpha = 0.06$ | $\alpha = 0.73$ | $\alpha = 0.41$

FIGURE 3: Interactive segmentation process: given an image and a set of manually delineated markers (in red), the parameter α can be set interactively in order to choose the "best" segmentation, i.e. the best compromise between the rate of false-positives and false-negatives.

- Medical images: our segmentation method has been compared with graph-cuts based method, leading to similar (and slightly better) results.

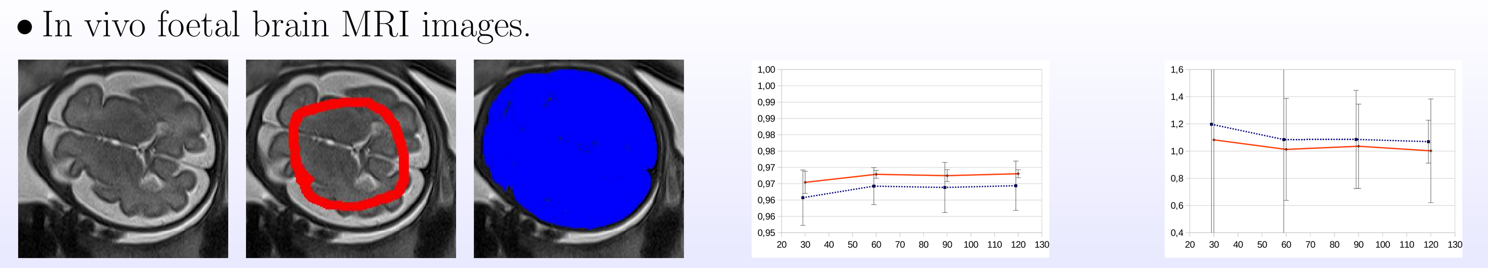
- Brainweb images (simulated normal brain MRI data).



(a) (b) (c) (d) Time of interaction vs. κ index. (e) Time of interaction vs. point-to-set distance.

FIGURE 4: (a) Original image. (b) Segmentation using the proposed method. (c) Segmentation using graph-cuts. False-positives are in red, false-negatives in green, and true-positives in white. (d-e) κ index and point-to-set distance between segmented image and ground-truth for our method (in blue) and method based on graph-cuts (in red).

- In vivo foetal brain MRI images.



(a) (b) (c) (d) Time of interaction vs. κ index. (e) Time of interaction vs. point-to-set distance.

FIGURE 5: (a) Original image. (b) Manual marker (in red). (c) Segmentation result (in blue). (d-e) κ index and point-to-set distance between segmented image and ground-truth for our method (in blue) and method based on graph-cuts (in red).