Implicit Component-Graph: A Discussion

Nicolas Passat[1] Benoît Naegel [2] Camille Kurtz[3]

[1]Université de Reims Champagne-Ardenne, CReSTIC (Reims, France)
[2]Université de Strasbourg, ICube (Strasbourg, France)
[3]Université Paris Descartes, LIPADE (Paris, France)

Motivations

Context

- Component-graphs (CGs) generalize component-trees to images with values in partially ordered sets.
- CGs allow for the development of various image processing approaches.
- CGs are not trees, but directed acyclic graphs: this induces a structural complexity associated to a higher combinatorial cost.

Purpose

- New way of building and manipulating CGs: purpose of reaching reasonable space and time costs.
- Tackling complexity issues is required for involving CGS in efficient image processing approaches.

Basic notions on Component-Graphs

Image I defined on a the vertices of a non-directed graph $(\Omega, \frown)$, taking values in an ordered set $(V, \leqslant)$.

$$\left| \begin{array}{cccc} I & : & \Omega & \longrightarrow & V \\ & & x & \longmapsto & v \end{array} \right. \tag{1}$$

For any $v \in V$, the thresholding function at value v is defined by

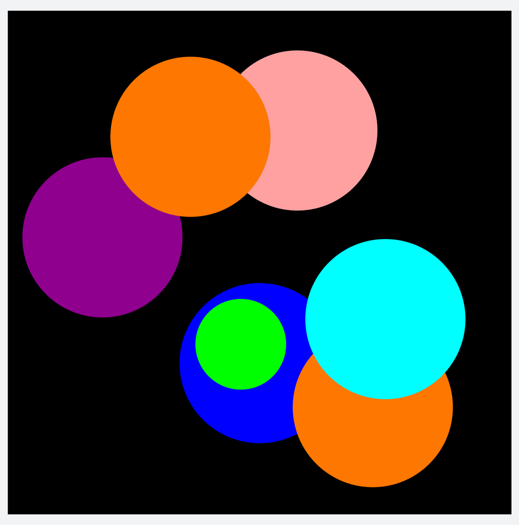
$$\left| \begin{array}{cccc} \lambda_v & : & V^\Omega & \longrightarrow & 2^\Omega \\ & & I & \longmapsto & \{x \in \Omega \mid v \leqslant I(x)\} \end{array} \right. \tag{2}$$

For any $X \subseteq \Omega$, the set of the connected components of the graph $(X, \frown)$ is noted $\mathcal{C}[X]$. Let $v \in V$ and $X \in \mathcal{C}[\lambda_v(I)]$. The couple $K = (X, v)$ is a valued connected component. The set Θ of all the valued connected components of I is

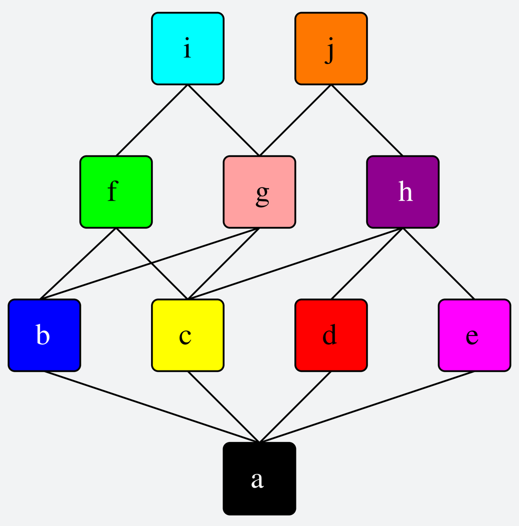
$$\Theta = \bigcup_{v \in V} \mathcal{C}[\lambda_v(I)] \times \{v\} \tag{3}$$

The component-graph of I is the Hasse diagram $\mathfrak{G} = (\Theta, \blacktriangleleft)$ of the ordered set $(\Theta, \trianglelefteq)$.

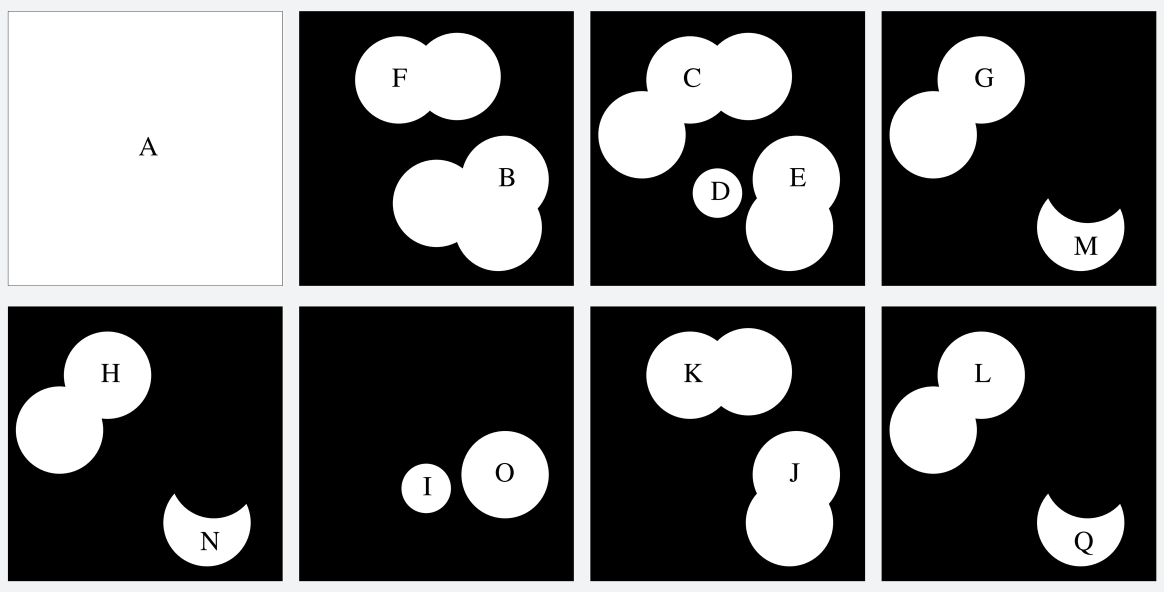
An example of Component-Graph



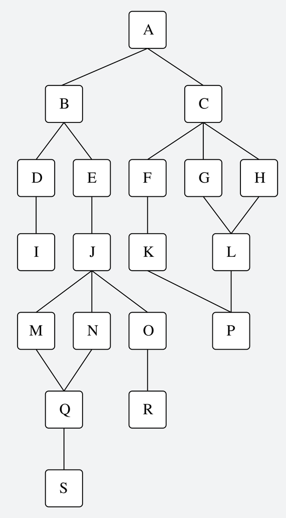
$(V, \leqslant)$ – Hasse



$I : \Omega \to V$



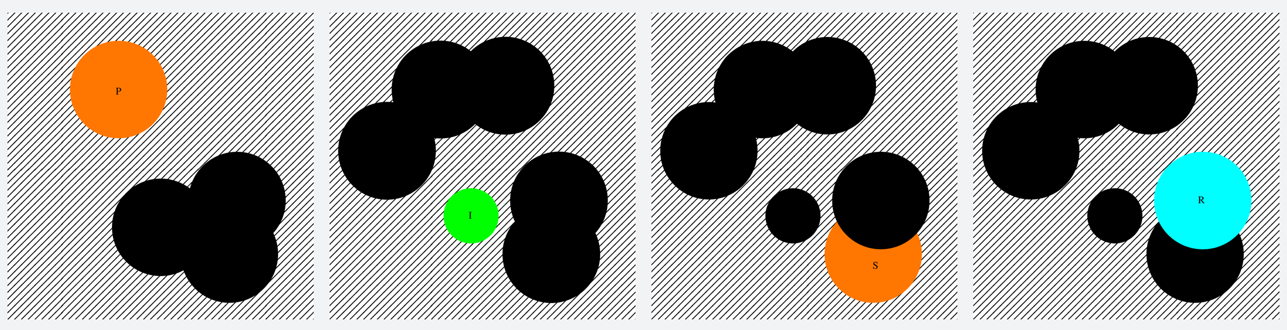
From top left to bottom right: $\lambda_a(I)$ to $\lambda_j(I)$



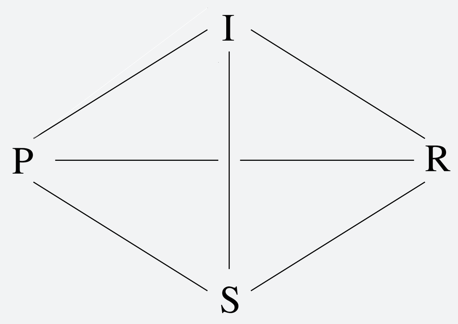
$\mathfrak{G} = (\Theta, \blacktriangleleft)$

Construction strategy

- Building the set of leaves $\Lambda \subseteq \Omega$.
- Building the function $\rho : \Lambda \to 2^\Omega$ that maps each leaf to its reachable zone.
- Building the reachable zone graph $\mathfrak{R} = (\Lambda, \frown_\Lambda)$.
- Computing the valuation function $\sigma : \Lambda \times \Lambda \to 2^V$ of $\mathfrak{R}$ and the associated reachable zone graph $(\mathfrak{R}, \sigma_\triangledown)$.



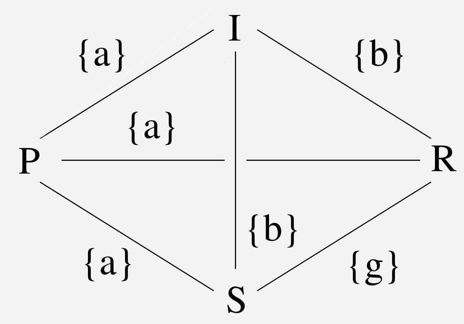
ρ(P) ρ(I) ρ(S) ρ(R)



$\mathfrak{R}$

	P	I	S	R
P	-	{a}	{a}	{a}
I	{a}	-	{a,b}	{a,b}
S	{a}	{a,b}	-	{a,b,c,g}
R	{a}	{c,b}	{a,b,c,g}	-

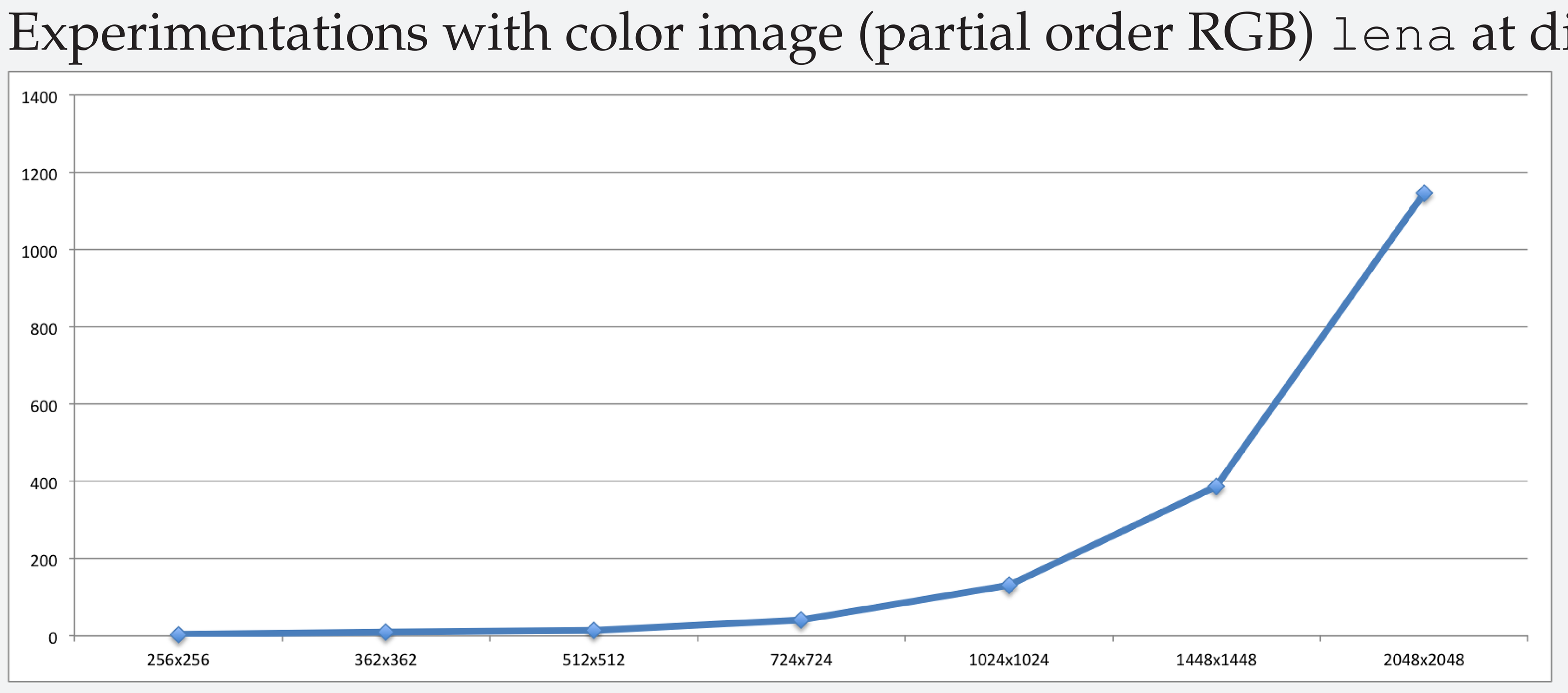
σ



$(\mathfrak{R}, \sigma_\triangledown)$

Computational cost

- Theoretical time cost: $\mathcal{O}((k.|\Omega|^\delta)^2/|\Lambda|^\beta)$
- Experimentations with color image (partial order RGB) `lena` at different size:



- Measured complexity: $\mathcal{O}(|\Omega|^{\log_2(3)}) \approx \mathcal{O}(|\Omega|^{1.58})$.

Contributions and perspectives

With these data-sructures, we can manipulate an implicit model of CG, and answer the following questions:

- Which are the nodes of $\mathfrak{G}$?
- What is a node of $\mathfrak{G}$?
- Is a node of $\mathfrak{G}$ lower, greater, or non-comparable to another, with respect to $\trianglelefteq$?

Next steps

- Interactive segmentation.
- Distributed algorithmics.
- Cache data-structure.

This work was partially funded by the French program Investissement d'Avenir (Agence Nationale pour la Recherche, grant ANR-11-INBS-0006).

Université de Reims Champagne-Ardenne Université de Strasbourg Université Paris Descartes USPC Université Sorbonne Paris Cité