Symbolic and structural models for image understanding

Part III: graphs, grammars and constraint satisfaction problems

Jamal Atif, Isabelle Bloch, Céline Hudelot

PSL RESEARCH UNIVERSITY DAUPHINE UNIVERSITÉ PARIS LAMSADE UMR CNRS 7243 laboratoire d'analyse et modélisation de systèmes pour l'aide à la décision
cnrs TELECOM ParisTech université PARIS-SACLAY CentraleSupélec

IJCAI 2016

Agenda

- Graphs: representations of spatial entities and spatial relations, graph-based reasoning.
- Conceptual graphs, constraint satisfaction problems, applications in scene understanding.
- Stochastic grammars and image parsing.

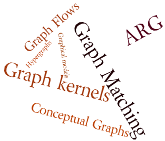
You are here!

1. Preliminaries
2. Graph-based reasoning for joint segmentation and recognition
3. CSP-based approaches
4. Stochastic grammars and image parsing
5. Conclusion and open problems

Graphs for image processing and understanding

Interest

- Structural information.
- Compact representations.
- “Efficient" manipulation and reasoning tools.
- Theoretical guarantees.



Some applications

- Low-level processing: denoising, segmentation, registration, etc.
  e.g. Graph flows.
- Mid-level analysis: object and action recognition, image classification, indexing, etc.
  e.g. Matching, Graph kernels, Graphical models.
- High-level interpretation: semantic interpretation
  e.g. Conceptual graphs.

Graphs for high-level image understanding

Attributed Relational Graph

$$G = (X, E, \mu, \nu)$$

- $\mu : X \rightarrow L_X$: vertex interpreter (L_X=vertices attributes). E.g. objects attributes (color, shape, texture, etc.).
- $\nu : E \rightarrow L_E$: edge interpreter (L_E= edges attributes). E.g. spatial and/or contrast constraints.

$\Rightarrow$ allows to associate structural, numeric or symbolic information with the elements of the graph (nodes and edges)

Fuzzy attributed Relational Graph

$$G = (X, E, \mu_f, \nu_f)$$

- $\mu_f : X \rightarrow [0, 1]$
- $\nu_f : E \rightarrow [0, 1]$

with $\forall x, y \in X \times X, \nu_f(x, y) \leq \mu_f(x)\mu_f(y)$ or $\nu_f(x, y) \leq \min(\mu_f(x), \mu_f(y))$

ARG: illustration

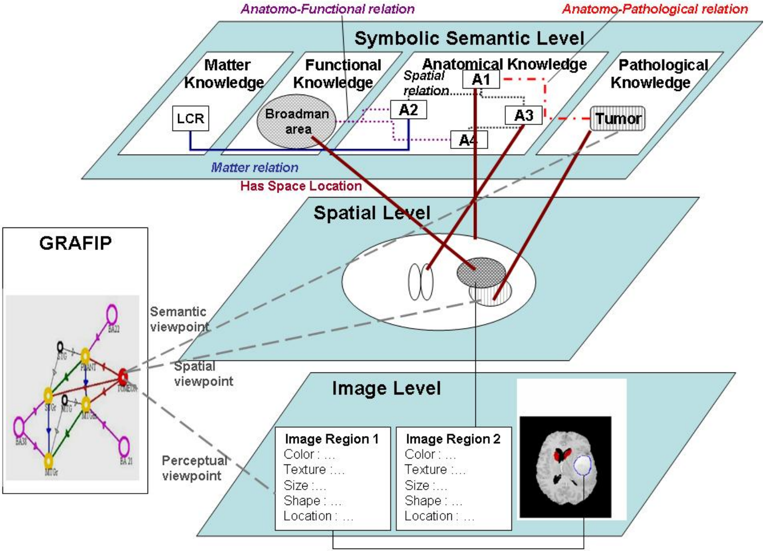


Image Region 1
Color : ...
Texture :...
Size :...
Shape : ...
Location : ...

Image Region 2
Color : ...
Texture :...
Size :...
Shape : ...
Location : ...

Conceptual graphs

Vocabulary

$V = (T_C, T_R, I)$

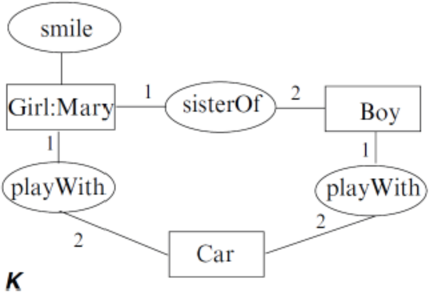
- T_C and T_R: the ontologies representing the set of relations and concepts in the domain, respectively.
- I: a set of names, called individual markers (used for denoting specific objects or entities).

CG

A bipartite graph $G = (N_C, N_R, E, l)$ built over V.

- N_C and N_R are the concept node and relation node sets, respectively. The set of nodes of G is equal to $N_C \cup N_R$,
- E is the family of edges,
- Relation and concept nodes are labeled by types from T_C and T_R using the function l.

Conceptual graphs



FOL correspondance

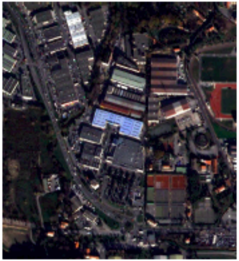
Conceptual graphs correspond to a fragment of FOL without functions.

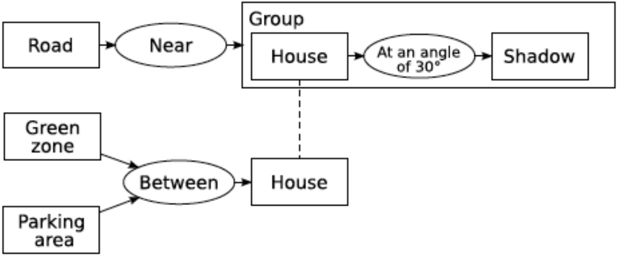
$$\Phi(G) = \exists x \exists y (Girl(Mary) \wedge Boy(x) \wedge Car(y) \wedge smile(Mary) \wedge \\ sisterOf(Mary, x) \wedge playWith(Mary, y) \wedge playWith(x, y))$$

Core reasoning service: the subsumption relation between graphs (graph homomorphism). Exponential complexity in some applications.

Nested conceptual graphs

- The concept nodes can have a conceptual graph contained in them.
- Useful for hierarchically structured knowledge.





- - →: coreference concepts representing the relations between objects inside a complex concept node and concept nodes outside it.

You are here!

1 Preliminaries

2 Graph-based reasoning for joint segmentation and recognition

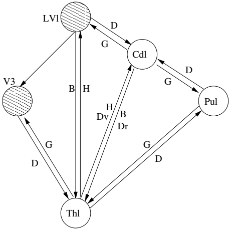
3 CSP-based approaches

4 Stochastic grammars and image parsing

5 Conclusion and open problems

Image labeling as an exploration process

- Sequential segmentation and recognition of structures [Colliot et al.].
- Start from structures “easy to segment" and proceed sequentially towards the “difficult” structures by constraining the search space.
- Graph-based optimization procedures [Fouquier et al.].
- Exploitation of pre-attentional mechanisms (saliency maps) to guide the exploration process [Fouquier et al.].



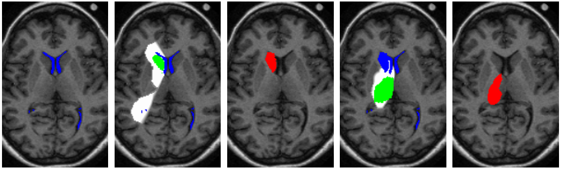
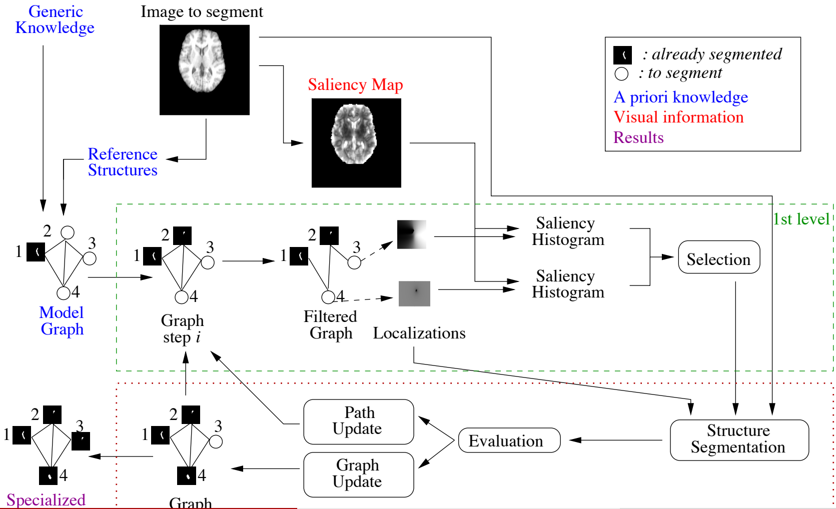
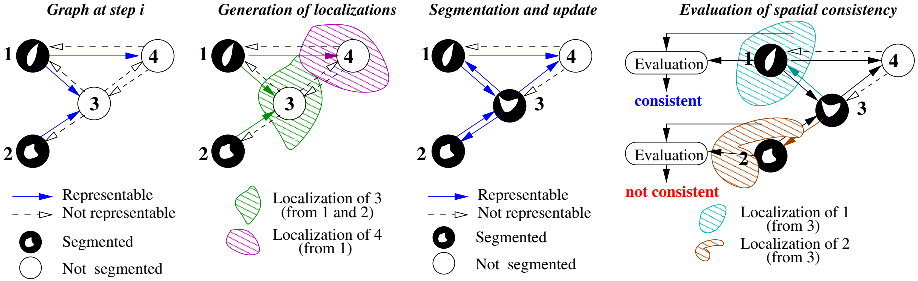


Image labeling as an exploration process

G. Fouquier, 2010



Segmentation assessment



Backtracking

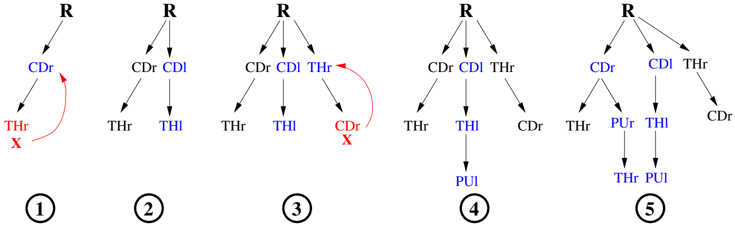
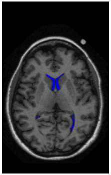


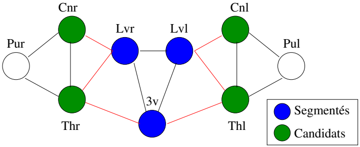
Figure: Structure of control of the segmentation results and configuration of the process. This structure keeps information about past segmentations of structures with different configurations to prevent the process of trying an already known configuration and to easily find remaining not-tested configurations.

Optimisation of the recognition sequence

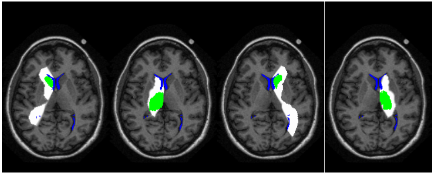
1st step:



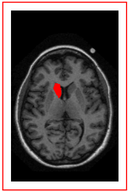
Localisation computing
Saliency based selection



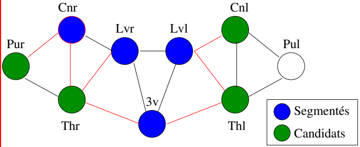
Segmentation CDr



Cdr Thr CDl THl



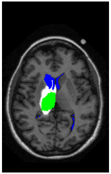
Segm.



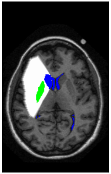
$i = 1$

Optimisation of the recognition sequence

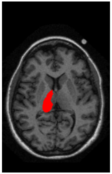
2nd step:



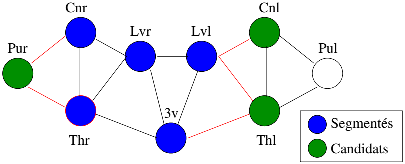
Thr



Pur

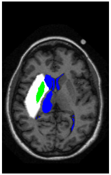


Segm.

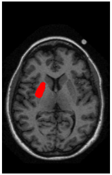


Graph $i = 2$

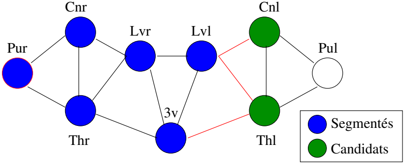
3rd step:



Pur



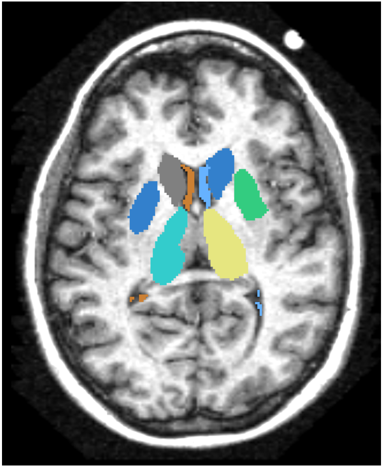
Segm.

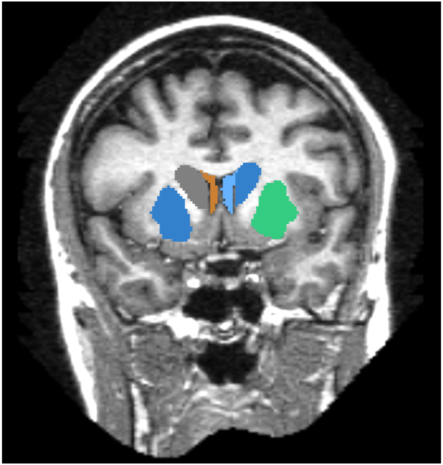


Graph $i = 3$

Optimisation of the recognition sequence

Final results





You are here!

1 Preliminaries

2 Graph-based reasoning for joint segmentation and recognition

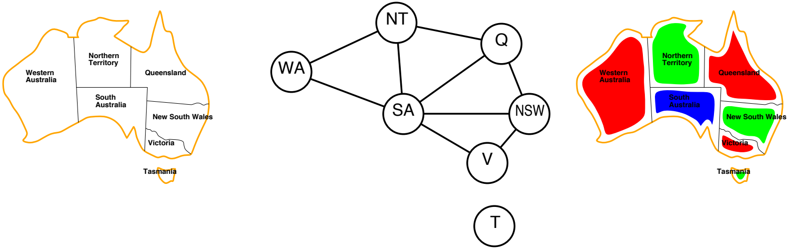
3 CSP-based approaches

4 Stochastic grammars and image parsing

5 Conclusion and open problems

Constraint Satisfaction Problems

- A generic framework for expressing and solving problems. whose aim is to find one or all solutions to a set of constraints.
- A constraint represents a relation, and a constraint satisfaction problem states which relations should hold among a given set of decision variables.
- A solution of a CSP is an assignment of values to all the variables that satisfy all the constraints.



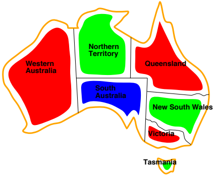
From Russel and Norvig book

Constraint Satisfaction Problems

Constraint network $\mathcal{P} = (\mathcal{X}, \mathcal{D}, \mathcal{C})$

- $\mathcal{X}$: a set of variables.
- $\mathcal{D}$: the domain of variables ($D_i \in \mathcal{D}$ is the domain of variable $x_i \in \mathcal{X}$).
- $\mathcal{C}$: a set of constraints. Each constraint $C \in \mathcal{C}$ is defined through a pair $\langle vars(C), rel(C) \rangle$ where $rel(C)$ is a subset of the Cartesian product of the domain of the variables in $vars(C \subseteq \mathcal{X})$.
- $A = \{a_1, a_2, \cdots, a_n\}$ is a solution of $\mathcal{P}$ if every $a_i \in D_i$ and for each $C_j \in \mathcal{C}$ its corresponding relation $rel(C)$ holds on the projection of A onto $var(C)$.

Australia example:
Variables: WA, NT, Q, NSW, V, SA, T
Domains: $D_i = \{red, green, blue\}$
Constraints: adjacent regions must have different colors
e.g., $WA \neq NT$ (if the language allows this), or
$(WA, NT) \in \{(red, green), (red, blue), (green, red), (green, blue), \ldots\}$



CSP related problems

- Determine whether $\mathcal{P}$ is consistent.
- Search a solution A to $\mathcal{P}$.
- Find the number of solutions.
- Find the set of solutions.

Procedures for finding a solution

- Search procedures: explore one by one every combination of the domain of each variable and reject those combinations which do not satisfy one of the constraint. Solved by backtracking or branch and bound.
- Inference or filtering: reducing the domain of the variables by applying local consistency algorithms.

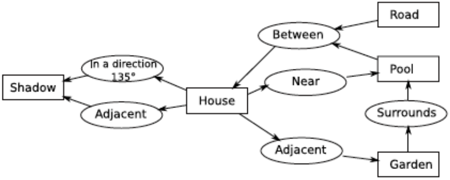
Local consistencies

- Arc consistency: $X \to Y$ is consistent iff for *every* value x of X there is *some* allowed y
- k-consistency: consistency over k variables.
- Node consistency: 1-consistency.

CSP for scene labeling

Basic idea

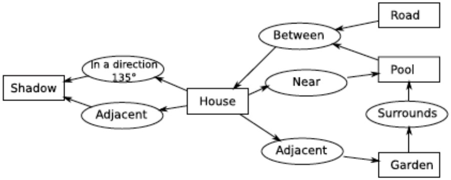
- Knowledge graph model as a constraint network.
- Image of segmented regions.
- Label the image by satisfying the constraints in the knowledge graph.

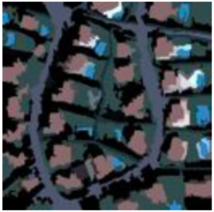


CSP for scene labeling

Basic idea

- Knowledge graph model as a constraint network.
- Image of segmented regions.
- Label the image by satisfying the constraints in the knowledge graph.

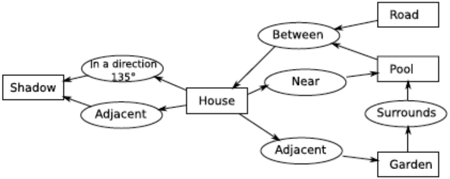


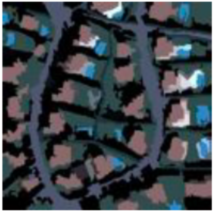


CSP for scene labeling

Basic idea

- Knowledge graph model as a constraint network.
- Image of segmented regions.
- Label the image by satisfying the constraints in the knowledge graph.





CSP-based approaches, early work

Waltz 72

- Understanding of line drawings with shadows.
- logical reasoning in worlds with simple geometrical objects (the worlds of blocks).
- Arc-consistency like algorithm to remove inconsistent interpretations.

Rosenfeld et al. 76

- Labeling of grey-level scenes.
- Parallel version of Waltz filtering approach.
- Fuzzy and probabilistic models allowing each interpretation to have a weight between $[0, 1]$.

Limitations

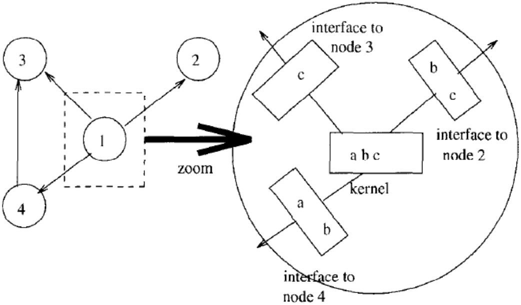
- Each label (node of the constraint network) is associated with a region in the segmented image.
- Bijection between the image and the knowledge graphs.
- Image segmentation is not perfect!

CSP with bi-level constraints

Labeling of oversegmented images

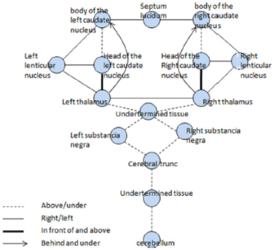
Deruyver et al. 97

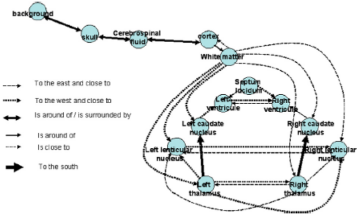
- A node in the constraint network is composed of:
  - a kernel K_i,
  - a set of interfaces X_i.
- Constraints inside the node and between nodes.
- A new Arc-consistency algorithm (AC4bc) with theoretical guarantees.

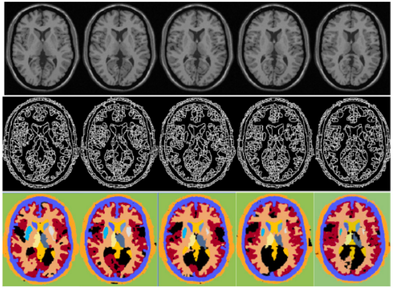


CSP with bi-level constraints

Interpretation of MR brain images







Images from Deruyver et al. AI 09.

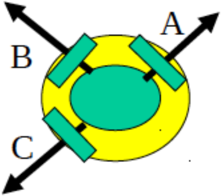
Dealing with unexpected objects

Deruyver et al. 09

- Bi-level constraints are adequate for a matching corresponding to a surjective function.
- Not appropriate for undersegmented images (surjection does not hold), or presence of unexpected objects (e.g. tumors). Two cases:
  1. Presence of extra data which cannot be associated with any node.
  2. No datum can be associated with a graph node.
- Move from strict arc-consistency to quasi arc-consistency.

Degree of relaxation

- 0 → A and B and C must be satisfied.
- 1 → (A and B) or (A and C) or (B and C) must be satisfied.
- 2 → A or B or C must be satisfied.



From Deruyver et al AI 09.

Application to pathological brain understanding

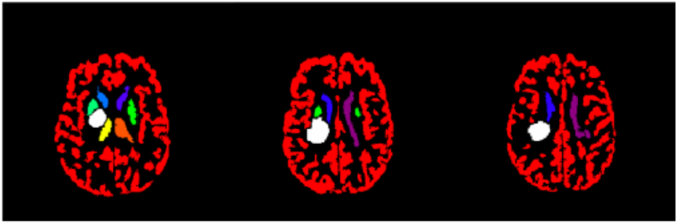


Image from Deruyver et al AI 09.

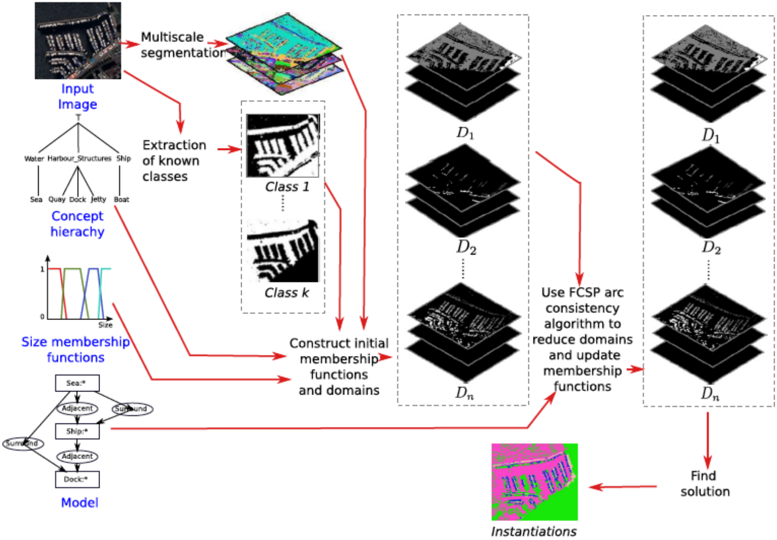
Fuzzy constraint satisfaction approach

Vanegas et al. 16

Main ideas

- Use of a nested CG.
- Deal with multiple instantiations through fuzzy CSP.
- Extend FCP3 to deal with groups of objects (e.g. alignments).

Outline of the FCSP-based image interpretation approach



Input Image

Multiscale segmentation

Extraction of known classes

Class 1

Class k

Concept hierachy

Size membership functions

Model

Construct initial membership functions and domains

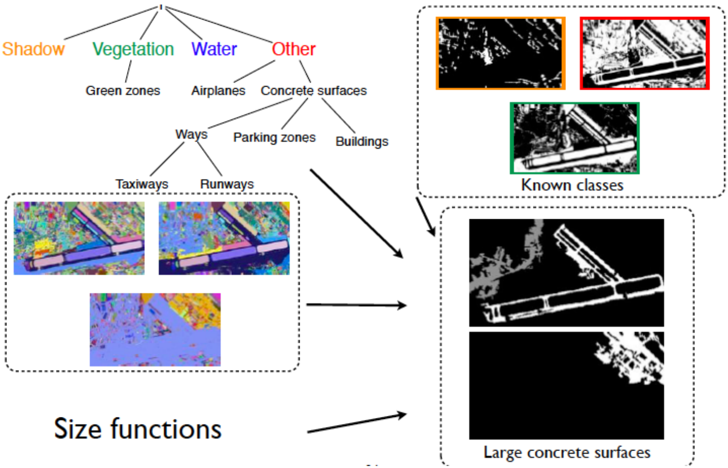
Use FCSP arc consistency algorithm to reduce domains and update membership functions

Find solution

Instantiations

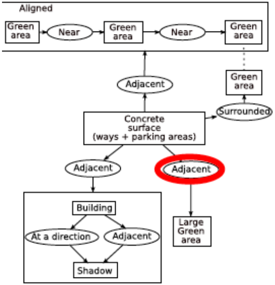
Illustration

Interpretation of satellite images

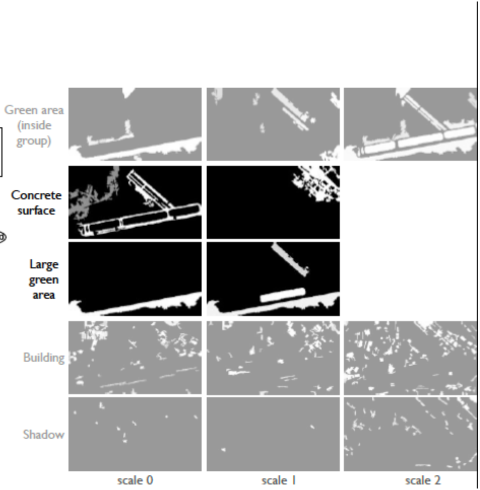


Illustration

Interpretation of satellite images

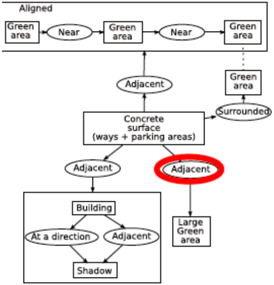


Before evaluating arc-consistency

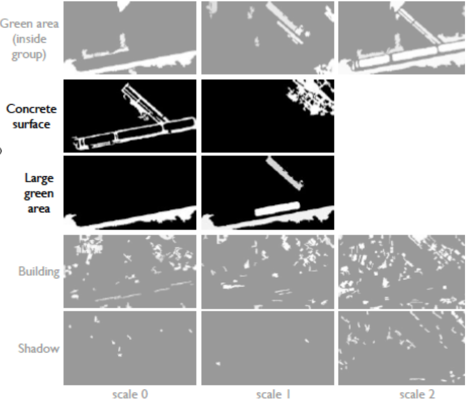


Illustration

Interpretation of satellite images

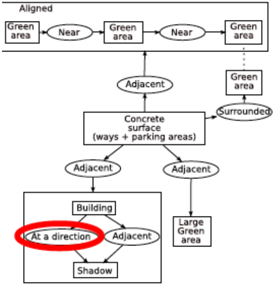


After evaluating arc-consistency

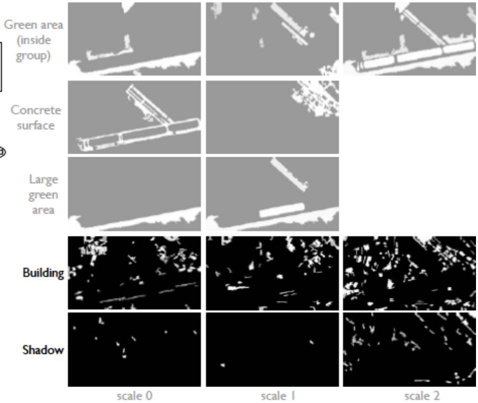


Illustration

Interpretation of satellite images

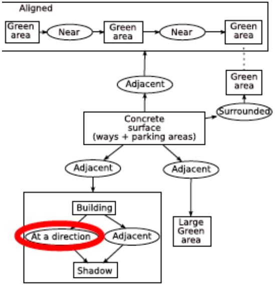


Before evaluating arc-consistency

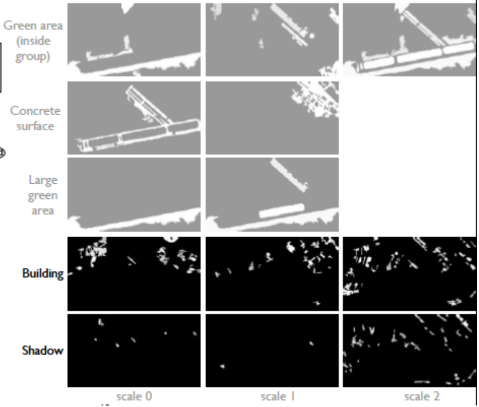


Illustration

Interpretation of satellite images

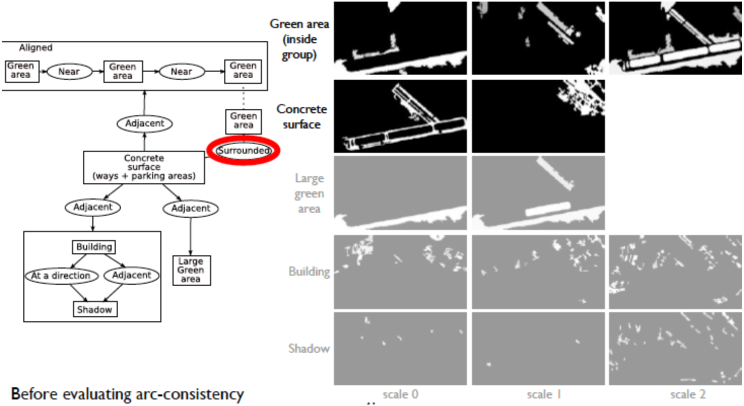


After evaluating arc-consistency



Illustration

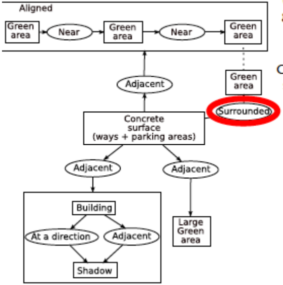
Interpretation of satellite images



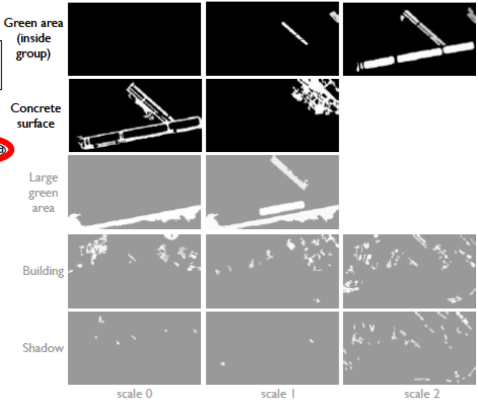
Before evaluating arc-consistency

Illustration

Interpretation of satellite images

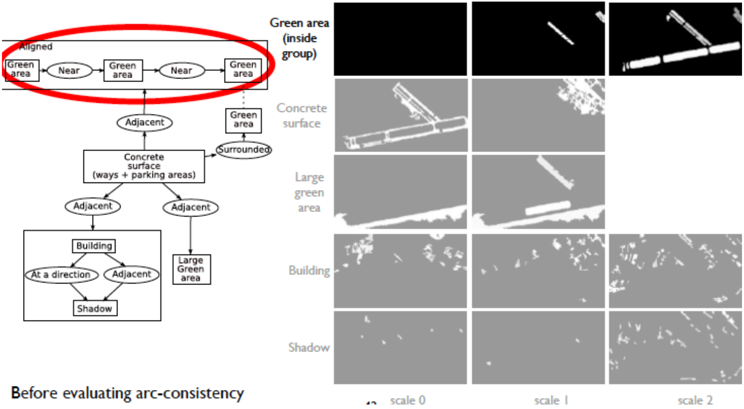


After evaluating arc-consistency



Illustration

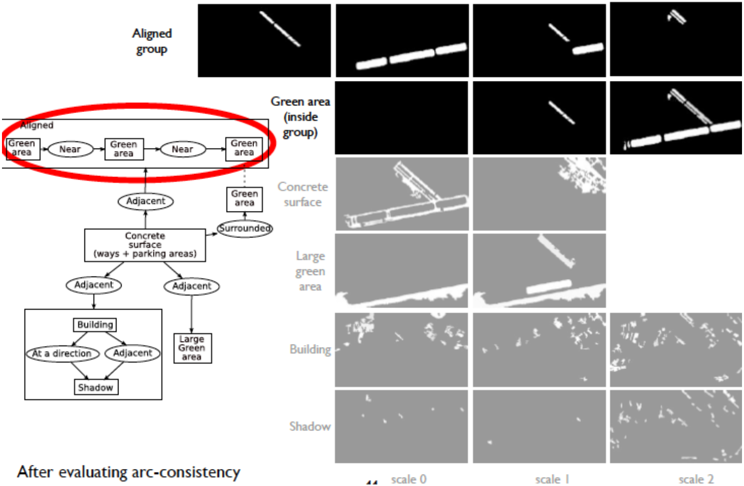
Interpretation of satellite images



Before evaluating arc-consistency

Illustration

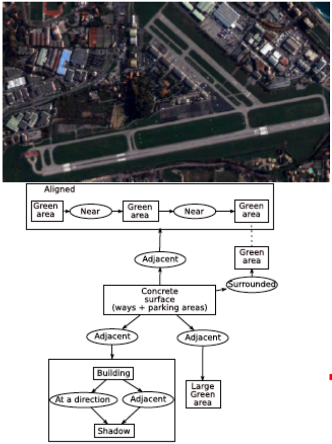
Interpretation of satellite images

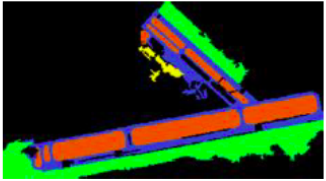


After evaluating arc-consistency

Illustration

Interpretation of satellite images





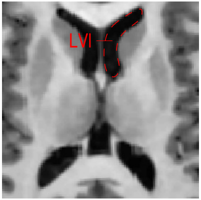
Instantiations

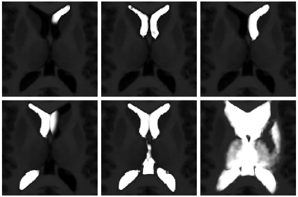
- 🟦 Concrete surface
- 🟧 Aligned green areas
- 🟩 Large green area
- 🟨 Building + shadow
- ⬛ Other

Joint segmentation and recognition as the resolution of a constraint network

Nempont et al. 13

- Brain structure A = fuzzy subset of the space μ_A ($\in \mathcal{F}$)
  = variable of the problem.
- μ_A takes values in a domain D_A (D_A = subset of $\mathcal{F}$).
- Structural priors = constraints between variables (subset of $D_A \times D_B$ for a constraint that involves two structures A and B).





D_{LVl}

Joint segmentation and recognition as the resolution of a constraint network

Definition of a constraint network $\langle \chi, \mathcal{D}, \mathcal{C} \rangle$ with:

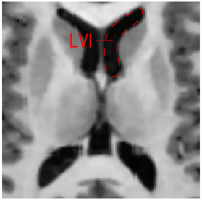
- χ = set of variables representing the brain structures to be recognized,
- $\mathcal{D}$ = set of associated domains,
- $\mathcal{C}$ = set of constraints involving variables in χ.

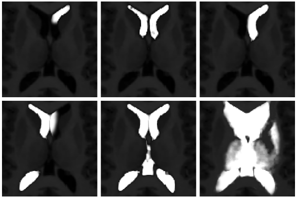
Segmentation and recognition = find a solution of the constraint network

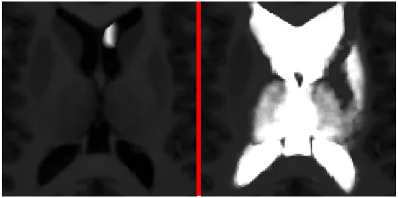
- Exhaustive search algorithms are untractable.
- $\Rightarrow$ Propagation algorithm (progressive reduction of the domains) and extraction of an acceptable solution.

Domains representation

- Domains are subsets of $\mathcal{F}$: cardinal = 10^{131072} on a 256×256 grid!
- The set of fuzzy sets $\mathcal{F}$ with the usual partial order $\leq$ is a complete lattice:
  Upper bound: $\overline{A} = \vee\{\nu \in D_A\}$
  Lower bound: $\underline{A} = \wedge\{\nu \in D_A\}$
- Representation of a domain by its bounds: $(\underline{A}, \overline{A}) = \{\nu \in \mathcal{F} | \underline{A} \leq \nu \leq \overline{A}\}$.
- $\Rightarrow$ Reduction of domains by updating their bounds.
- Complexity $\Rightarrow$ definition of contracting operators that reduce domain bounds and are computed on domain bounds.




A domain for LVl


$\underline{LVl}$ and $\overline{LVl}$

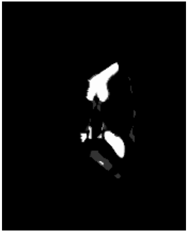
Constraint example: directional relative position

- Constraint:

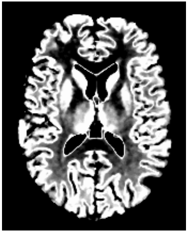
$$C^{dir}_{A,B}(\mu_1, \mu_2) = \begin{cases} 1 & \text{if } \mu_2 \leq \delta_\nu(\mu_1) \\ 0 & \text{otherwise} \end{cases}$$

- $\exists \mu_1 \in (\underline{A}, \overline{A}), C^{dir}_{A,B}(\mu_1, \mu_2) = 1 \Leftrightarrow \mu_2 \leq \delta_\nu(\overline{A})$
- Domain reduction:

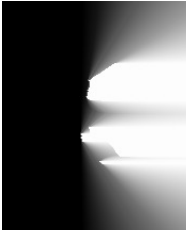
$$\frac{\langle A, B; (\underline{A}, \overline{A}), (\underline{B}, \overline{B}); C^{dir}_{A,B} \rangle}{\langle A, B; (\underline{A}, \overline{A}), (\underline{B}, \overline{B} \wedge \delta_\nu(\overline{A}); C^{dir}_{A,B} \rangle}$$



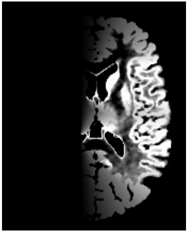
$\overline{LVl}$



$\overline{CNl}$



right of $\overline{LVl}$



$\overline{CNl} \wedge \delta_\nu(\overline{LVl})$

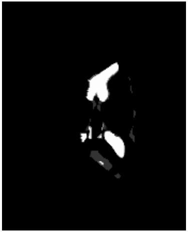
Constraint example: directional relative position

- Constraint:

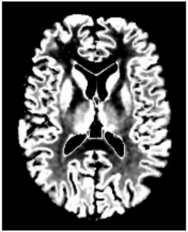
$$C^{dir}_{A,B}(\mu_1,\mu_2) = \begin{cases} 1 & \text{if } \mu_2 \leq \delta_\nu(\mu_1) \\ 0 & \text{otherwise} \end{cases}$$

- $\exists \mu_1 \in (\underline{A}, \overline{A}), C^{dir}_{A,B}(\mu_1,\mu_2) = 1 \Leftrightarrow \mu_2 \leq \delta_\nu(\overline{A})$
- Domain reduction:

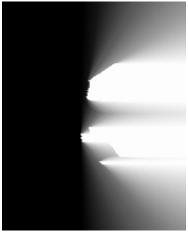
$$\frac{\langle A, B; (\underline{A}, \overline{A}), (\underline{B}, \overline{B}); C^{dir}_{A,B}\rangle}{\langle A, B; (\underline{A}, \overline{A}), (\underline{B}, \overline{B} \wedge \delta_\nu(\overline{A}); C^{dir}_{A,B}\rangle}$$



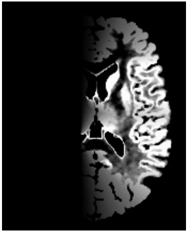
$\overline{LVl}$



$\overline{CNl}$



right of $\overline{LVl}$



$\overline{CNl} \wedge \delta_\nu(\overline{LVl})$

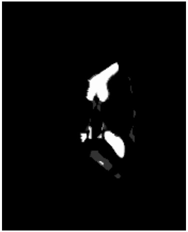
Constraint example: directional relative position

- Constraint:

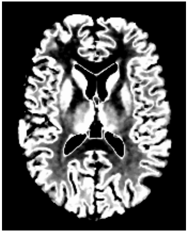
$$C^{dir}_{A,B}(\mu_1, \mu_2) = \begin{cases} 1 & \text{if } \mu_2 \leq \delta_\nu(\mu_1) \\ 0 & \text{otherwise} \end{cases}$$

- $\exists \mu_1 \in (\underline{A}, \overline{A}), C^{dir}_{A,B}(\mu_1, \mu_2) = 1 \Leftrightarrow \mu_2 \leq \delta_\nu(\overline{A})$
- Domain reduction:

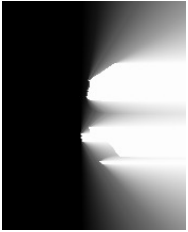
$$\frac{\langle A, B; (\underline{A}, \overline{A}), (\underline{B}, \overline{B}); C^{dir}_{A,B} \rangle}{\langle A, B; (\underline{A}, \overline{A}), (\underline{B}, \overline{B} \wedge \delta_\nu(\overline{A}); C^{dir}_{A,B} \rangle}$$



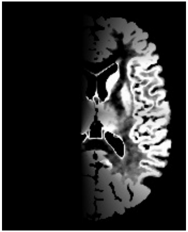
$\overline{LVl}$



$\overline{CNl}$

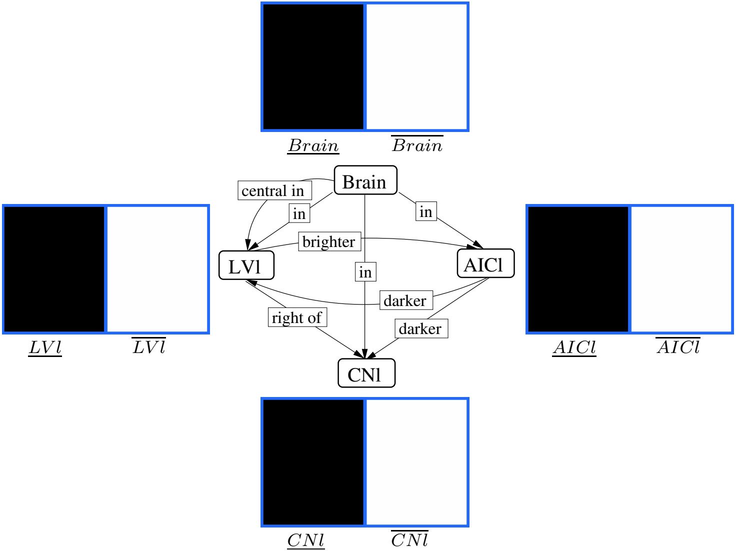


right of $\overline{LVl}$

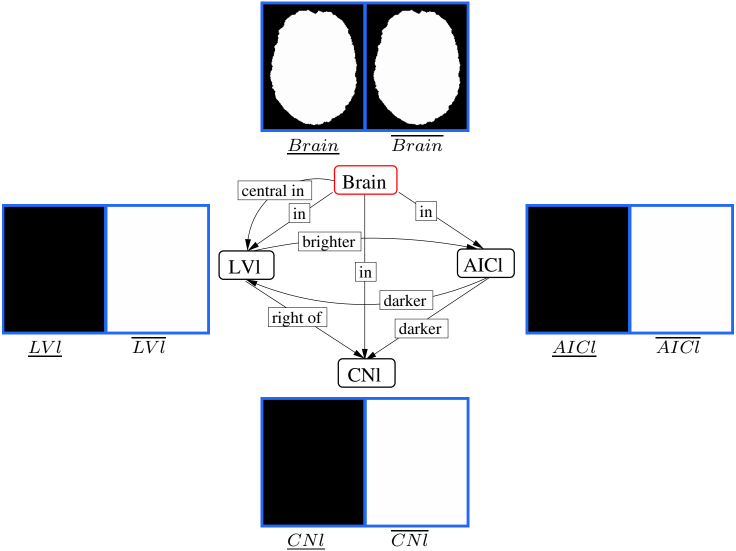


$\overline{CNl} \wedge \delta_\nu(\overline{LVl})$

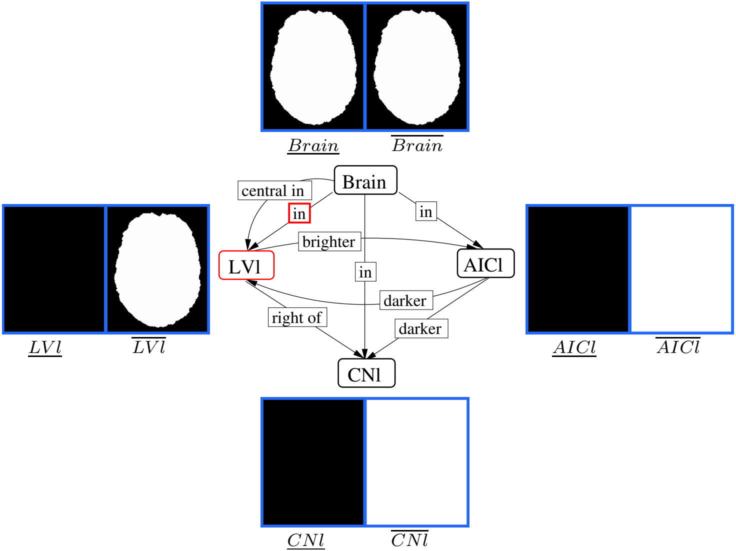
Propagation



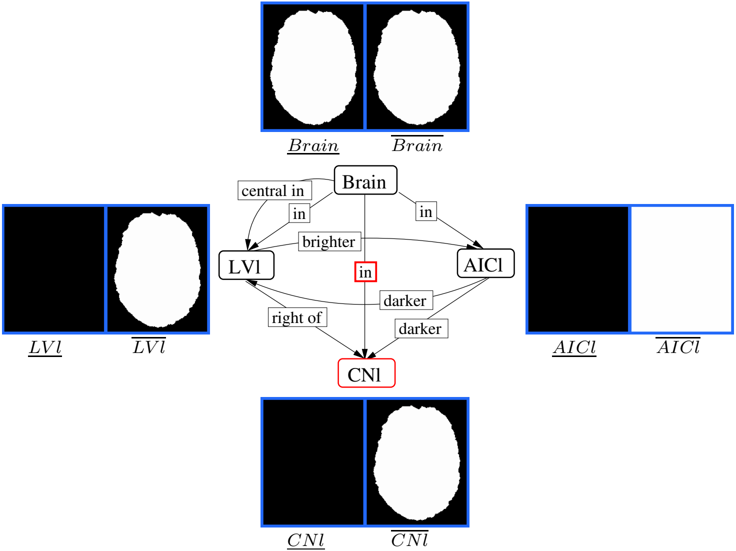
Propagation



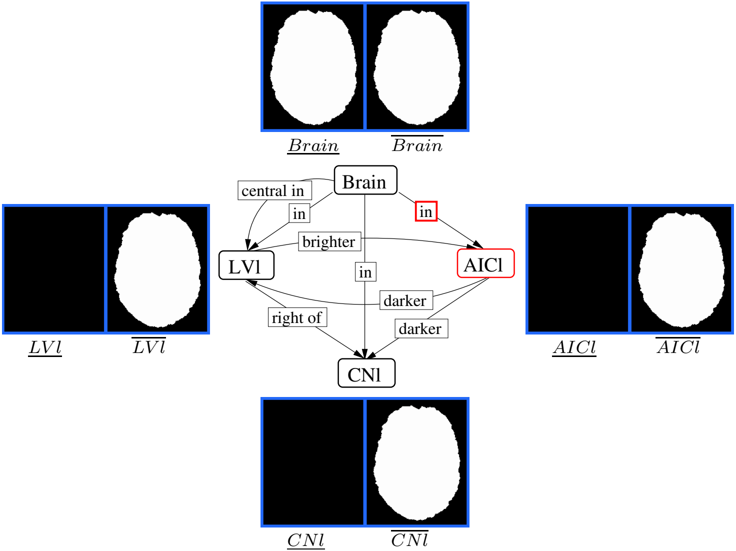
Propagation



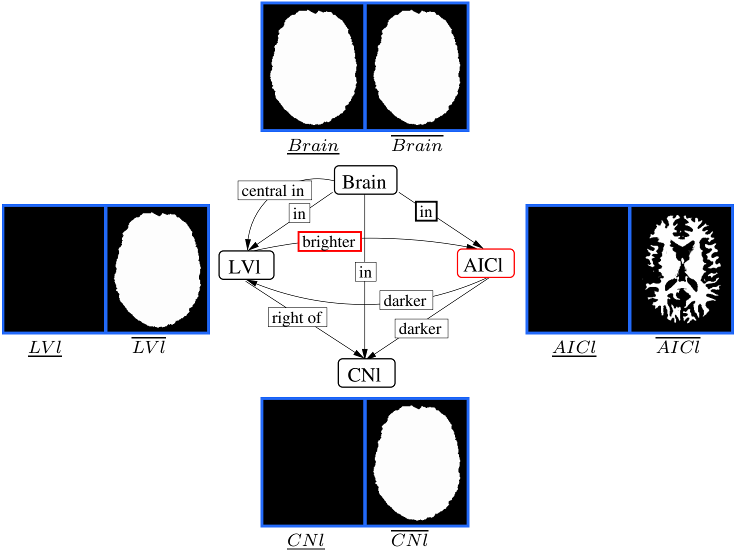
Propagation



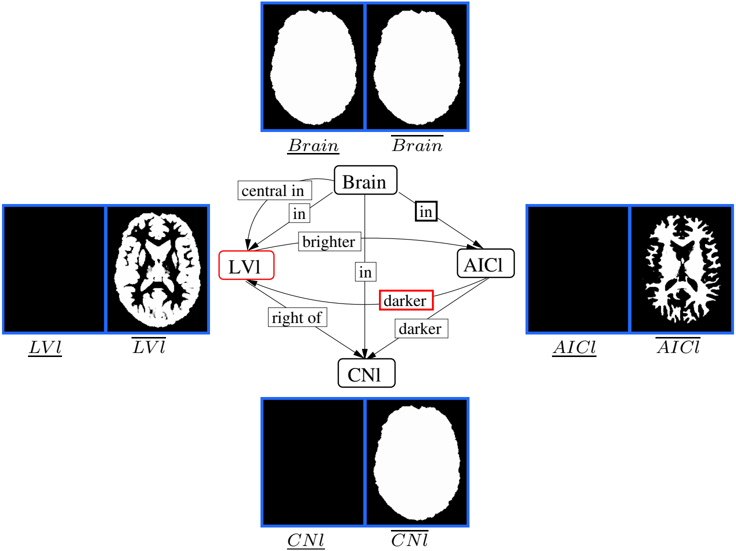
Propagation



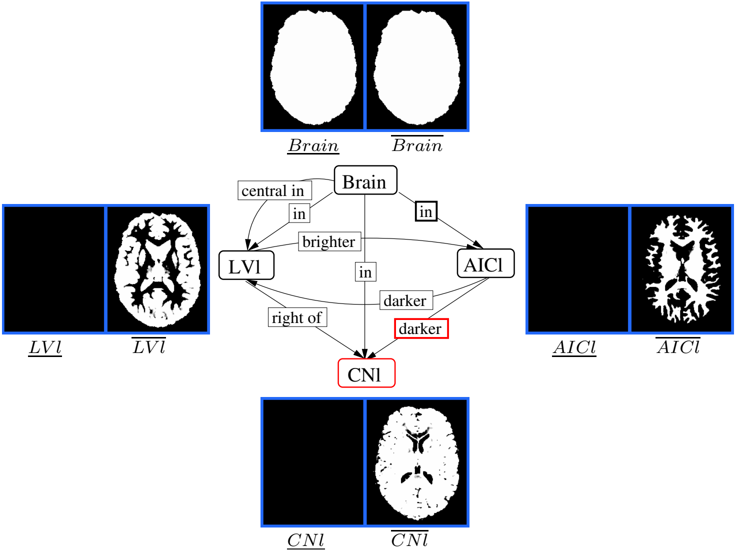
Propagation



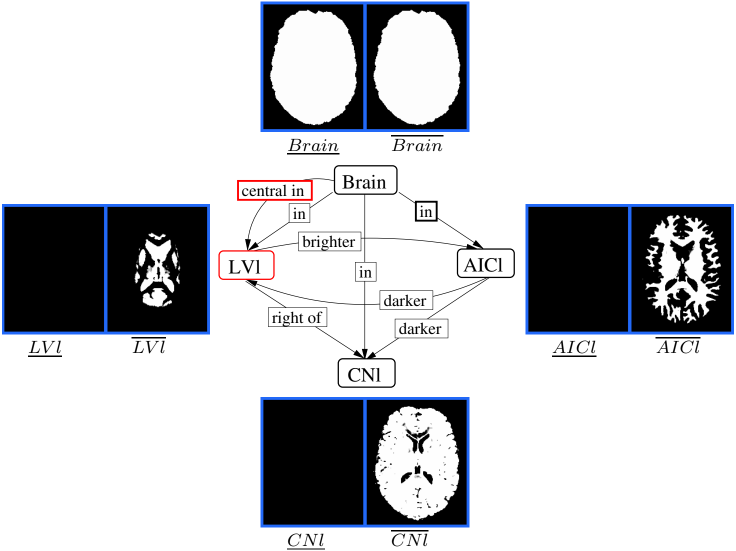
Propagation



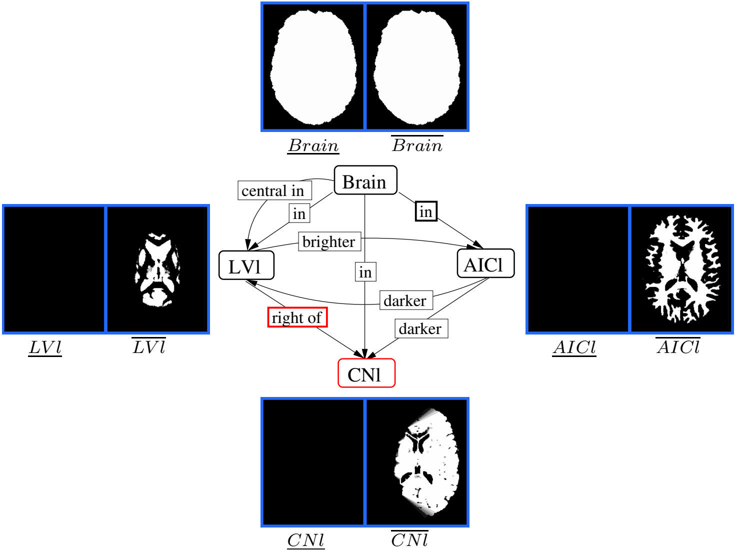
Propagation



Propagation



Propagation



$\underline{Brain}$ $\overline{Brain}$

Brain

central in

in

in

brighter

LVl

in

AICl

darker

right of

darker

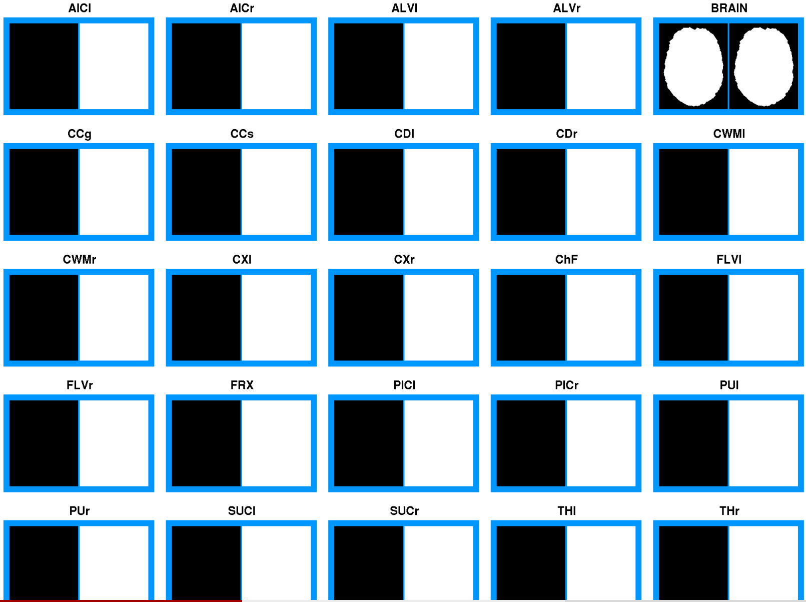
CNl

$\underline{LVl}$ $\overline{LVl}$

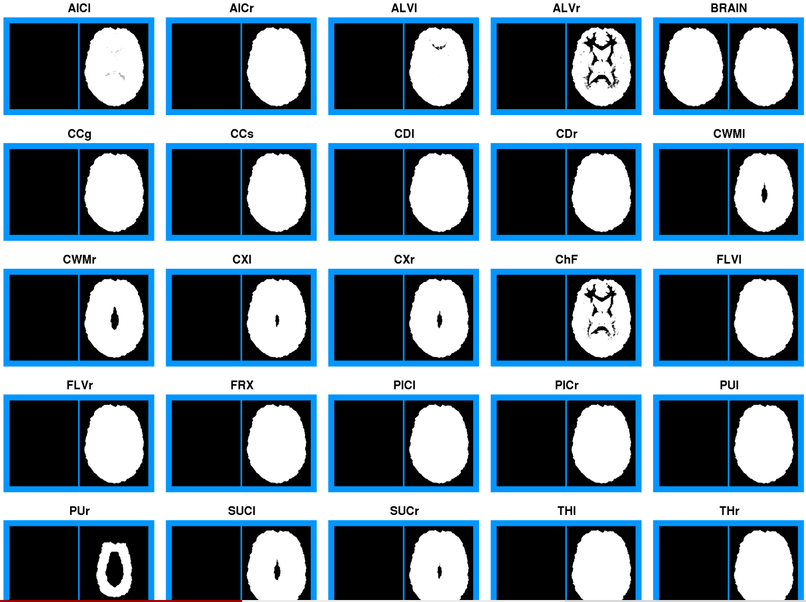
$\underline{AICl}$ $\overline{AICl}$

$\underline{CNl}$ $\overline{CNl}$

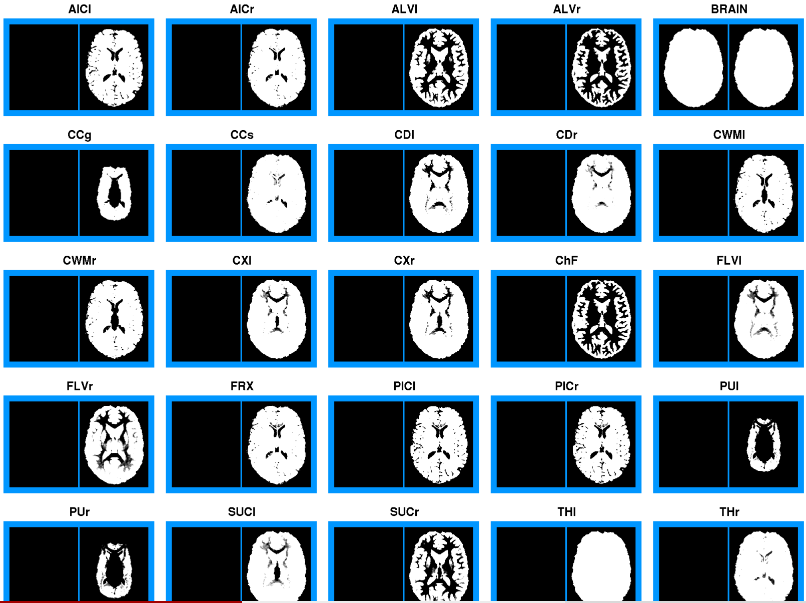
Propagation: iteration 0



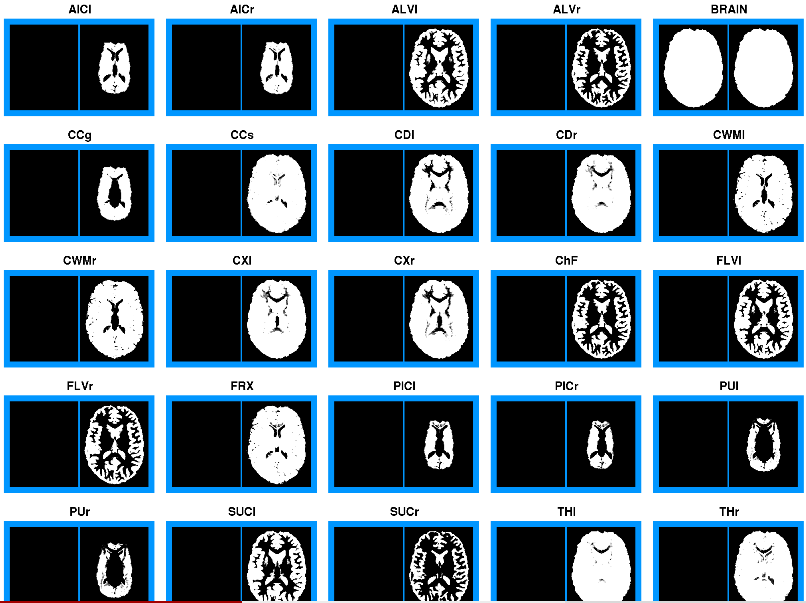
Propagation: iteration 200



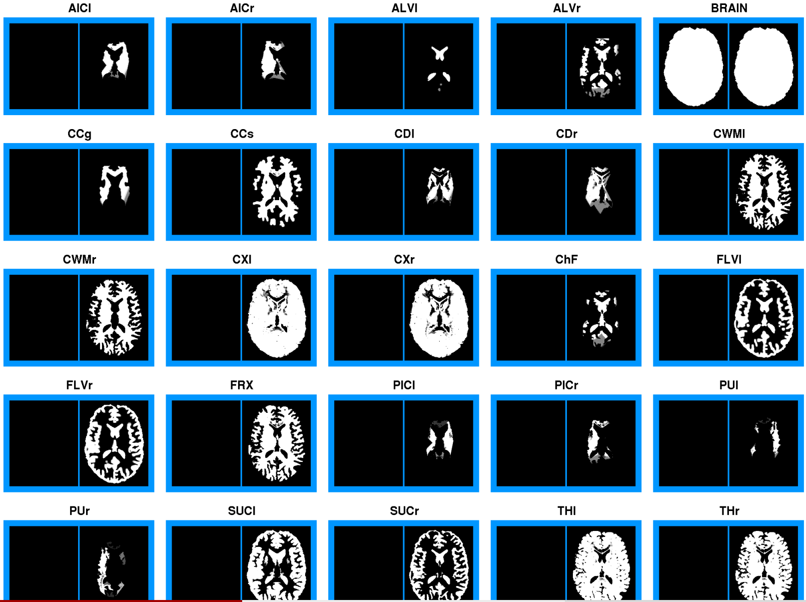
Propagation: iteration 500



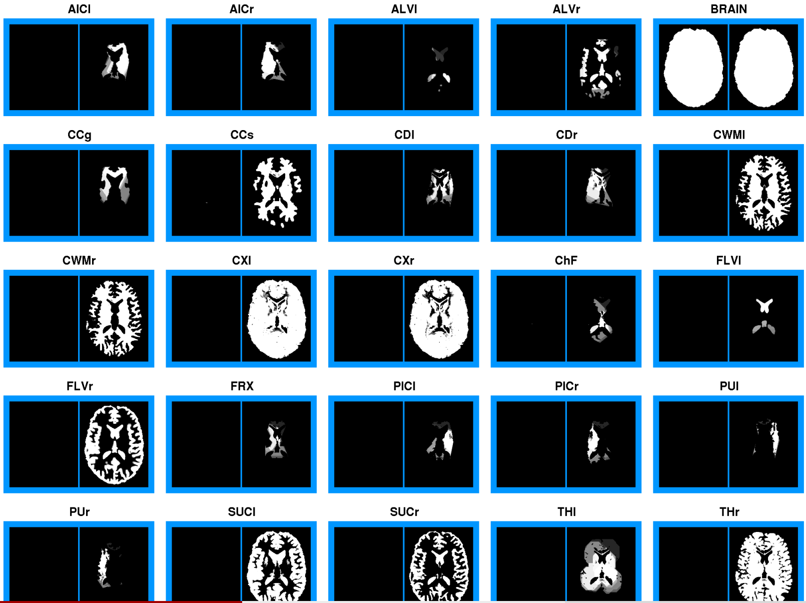
Propagation: iteration 1000



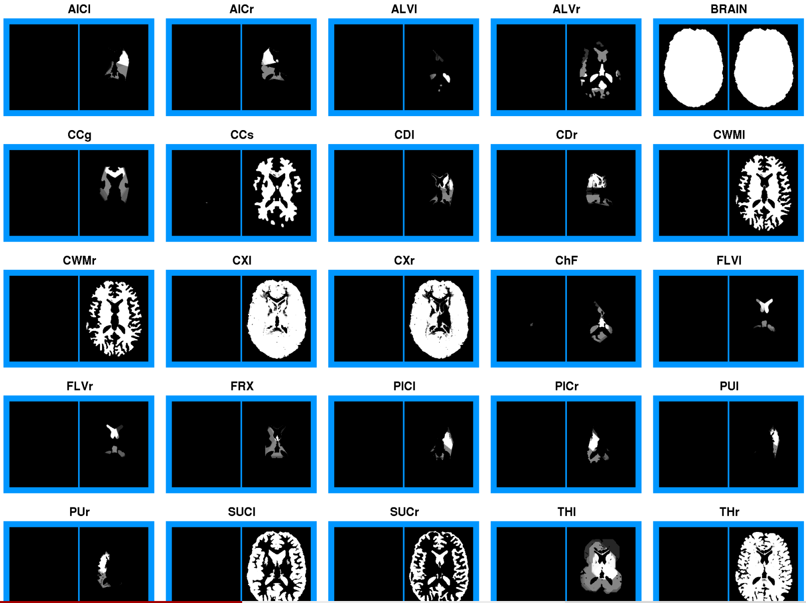
Propagation: iteration 2000



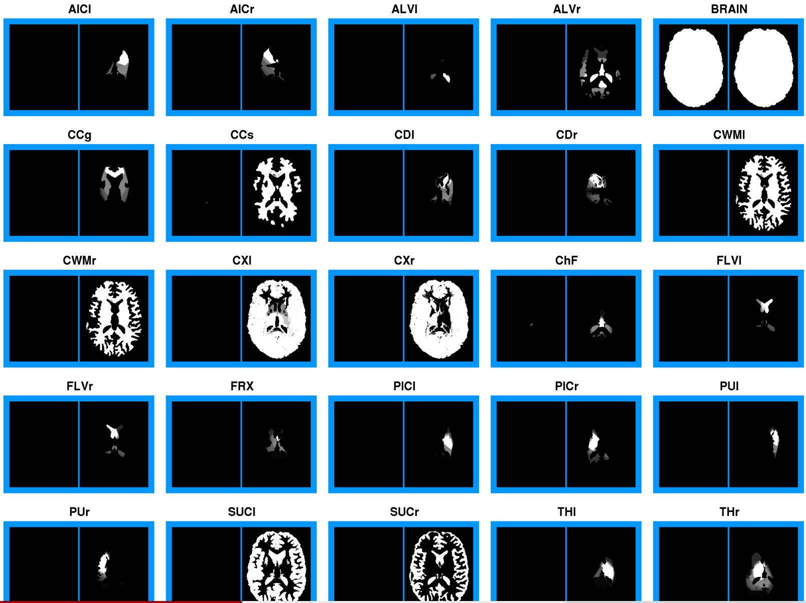
Propagation: iteration 3000



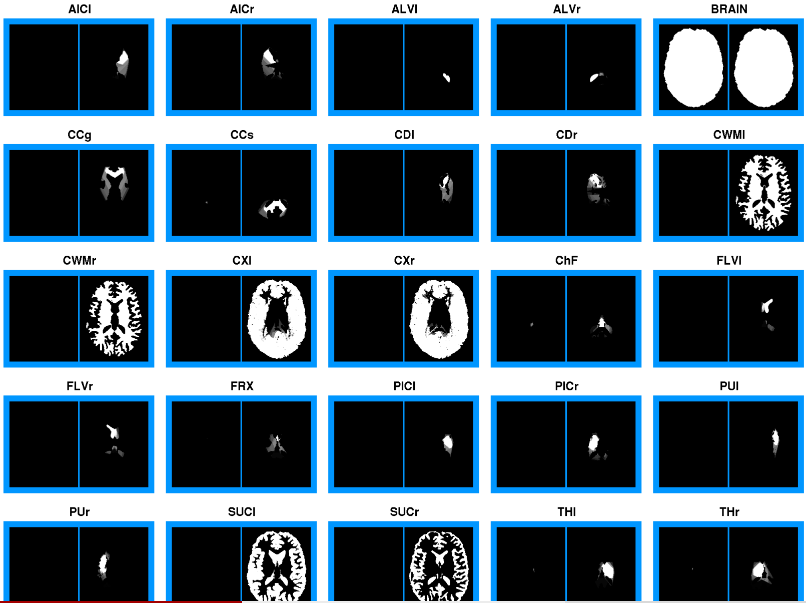
Propagation: iteration 4000



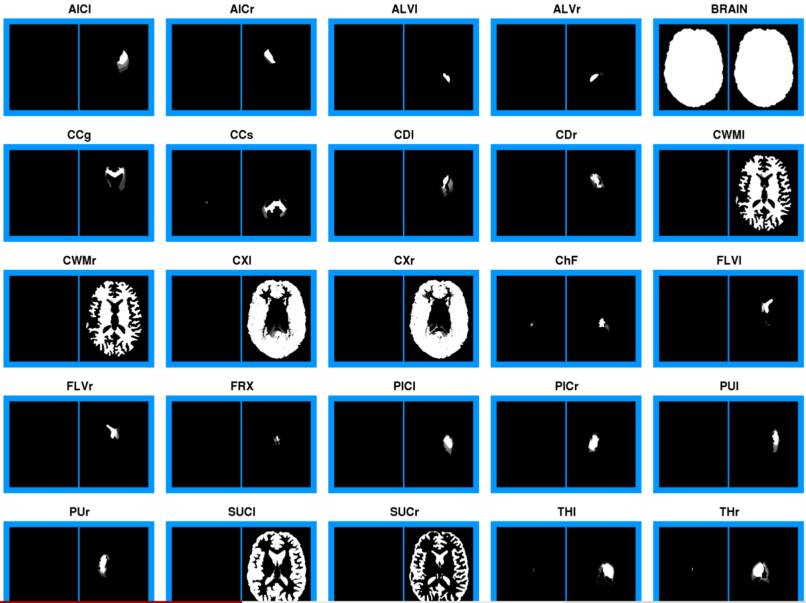
Propagation: iteration 5000



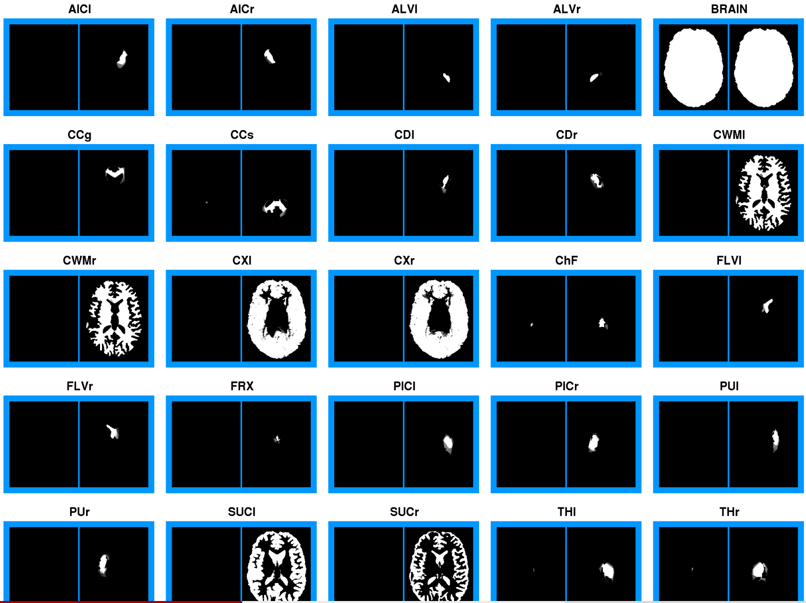
Propagation: iteration 6000



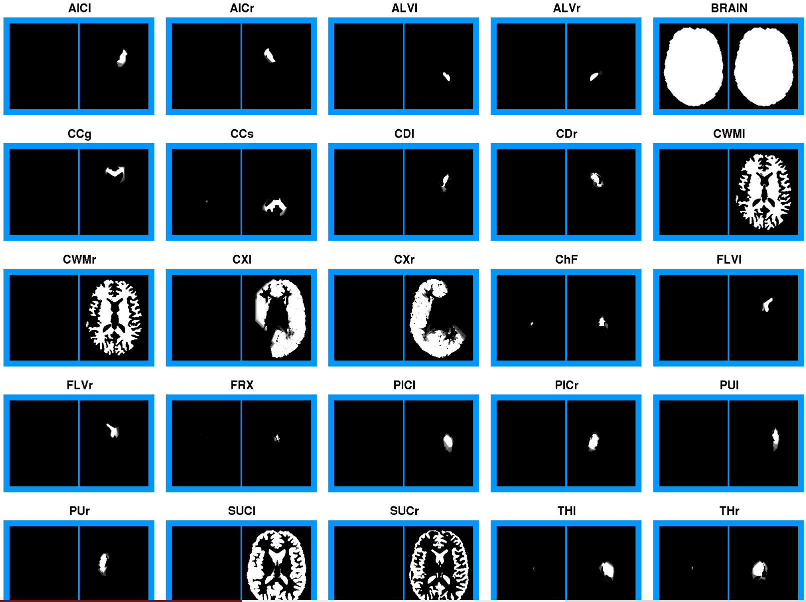
Propagation: iteration 7000



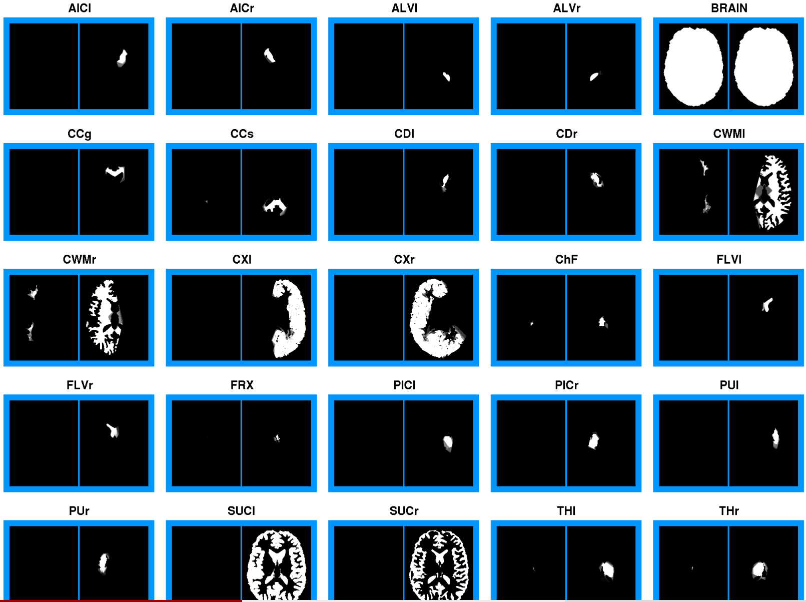
Propagation: iteration 8000



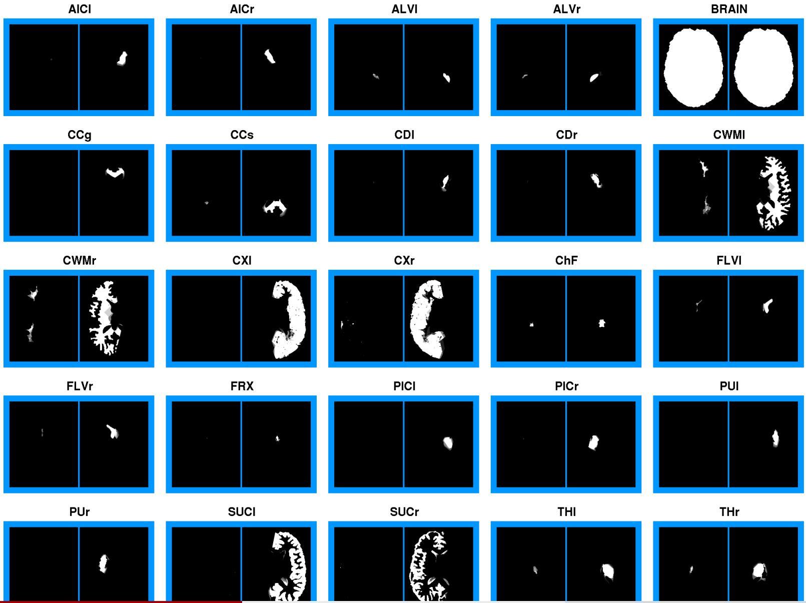
Propagation: iteration 9000



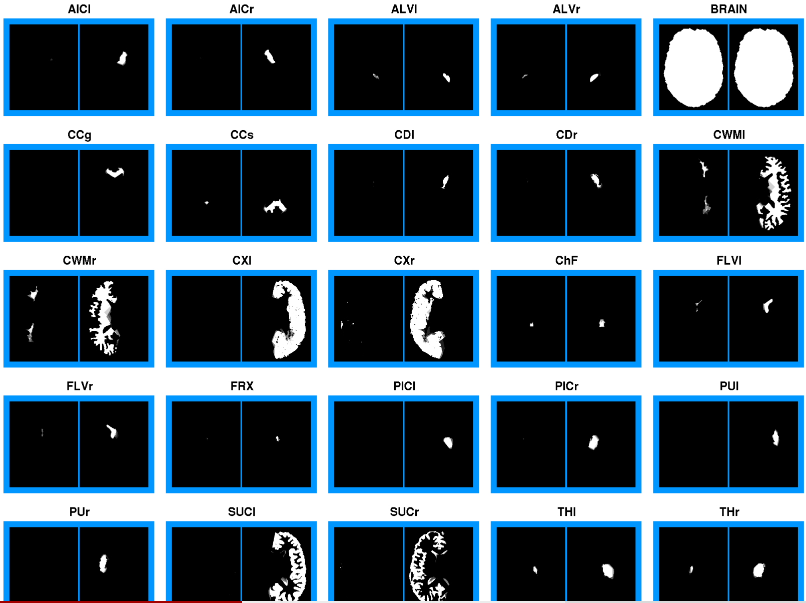
Propagation: iteration 10000



Propagation: iteration 13000



Propagation: iteration 15000



AICl AICr ALVl ALVr BRAIN

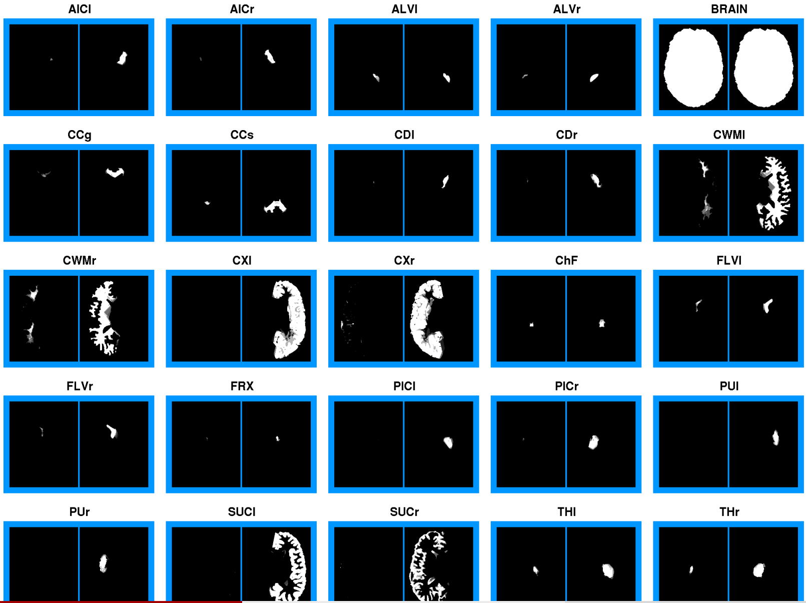
CCg CCs CDl CDr CWMl

CWMr CXl CXr ChF FLVl

FLVr FRX PICl PICr PUl

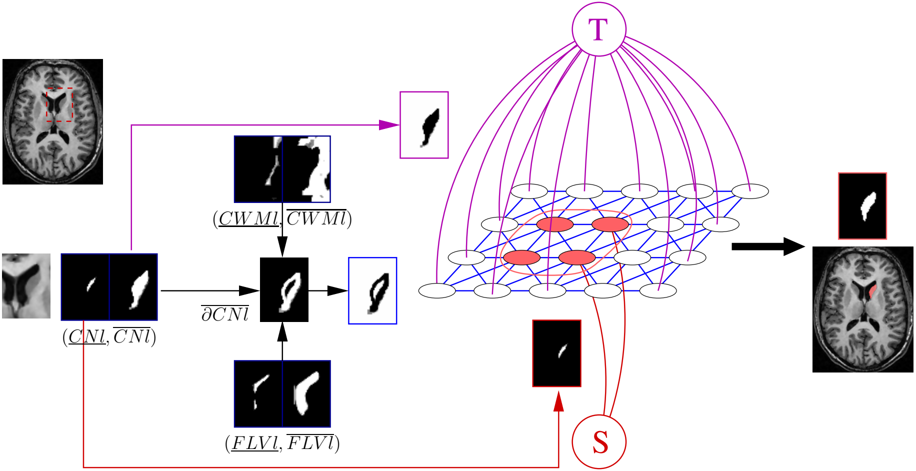
PUr SUCl SUCr THl THr

Propagation: iteration 18000

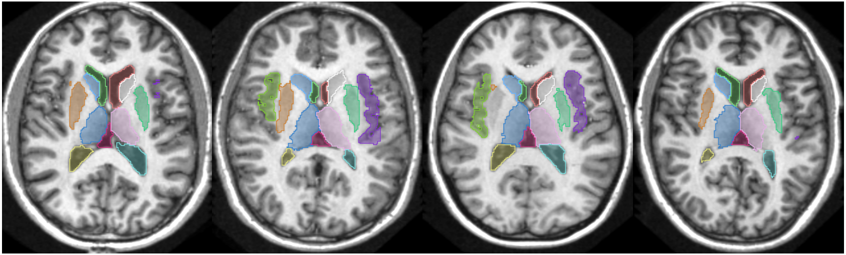


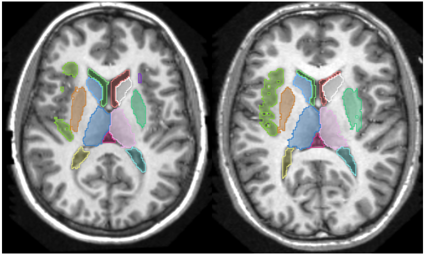
Extraction of a solution

Nempont et al. 13


T
$(\underline{CWMl}, \overline{CWMl})$
$(\underline{CNl}, \overline{CNl})$
$\overline{\partial CNl}$
$(\underline{FLVl}, \overline{FLVl})$
S

Results





Structure	kappa	D_M
CDl	0.94	0.3
CDr	0.94	0.3
FLVl	0.91	0.4
FLVr	0.89	0.8
THl	0.91	1.0
THr	0.92	0.8
PUl	0.86	1.0
PUr	0.73	3.0

You are here!

1 Preliminaries

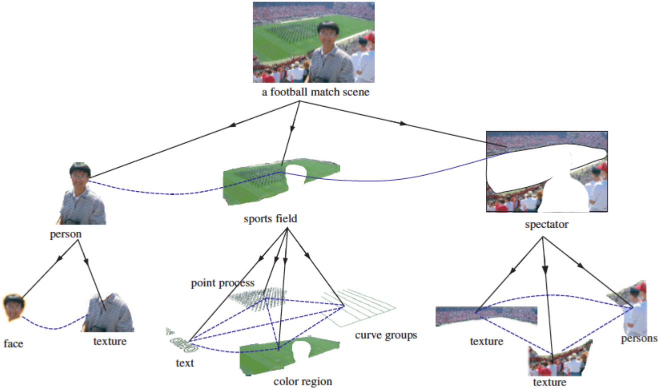
2 Graph-based reasoning for joint segmentation and recognition

3 CSP-based approaches

4 Stochastic grammars and image parsing

5 Conclusion and open problems

Grammar for IU: a foretaste



From Zhu and Mumford

Grammar in a nutshell

Chomsky 56

Syntax

A grammar $\mathcal{G}$ =

- A finite set N of nonterminal symbols,
- A finite set T of terminal symbols that is disjoint from N,
- A finite set R of production rules,
- A distinguished symbol $S \in N$ that is the start symbol, also called the sentence symbol.

Semantics

A sentence w is valid if it can be derived from in a finite number of steps from the start symbol S:

$$\left\{ w \in (T \cup N)^* \mid S \underset{G}{\overset{*}{\Rightarrow}} w \right\}$$

where $\cdot^*$ is the Kleene star operator: $V^* = \cup_{n \in \mathbb{N}} V^n$, $V^n = \underbrace{V \times \cdots \times V}_{n}$.

The set of all valid sentences define the language of $\mathcal{G}$, denoted $L(\mathcal{G})$.

Grammar in a nutshell

Reasoning

- Generative mode: *generate* a set of sentences from a given grammar.
- Analysis mode: check if a sentence has been generated from a given grammar.
- Inference mode: find a grammar that would have created a set of sentences.

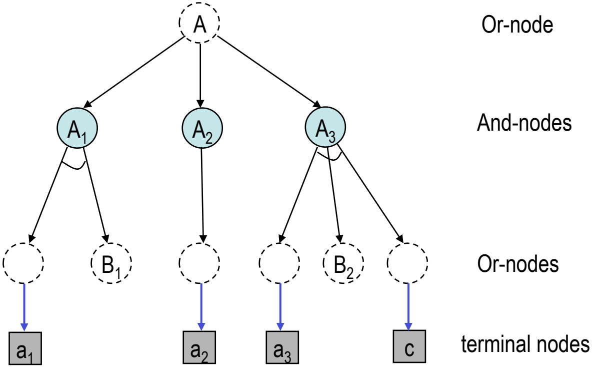
Types of grammars

- Type 0: Free or Unrestricted.
- Type 1: Context-Sensitive $\alpha A \beta \rightarrow \alpha B \beta$ with $A \in N$ and $\alpha, \beta, B \in (N \cup T)^*$
- Type 2: Context-Free $N \rightarrow (N \cup T)^*$.
- Type 3: Finite State or Regular. CFG with RHS restricted to an empty string, or single terminated symbol, or a terminal symbol followed by a non-terminal one.

Grammar and AND-OR tree

A ::= aB | a | aBc

A production rule can be represented by an And-Or tree



From Zhu and Mumford.

Grammars for image understanding

1960-80

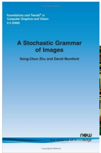
- Inspiration from Natural Language Understanding.
- Intensive work of K.S. Fu, and others (e.g. Riseman, Ohta and Kanade).
- Problems: Knowledge bottleneck, Computation complexity, Semantic gap.

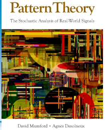
90's: Hibernation but with some resistants

- Zhu and Yuille 96,
- Mangin et al. 94: Brain cortical sulcal anatomy labelling.

mid 00's–: resurgence

- Geman, Ahuja, Yuille, **Zhu and Mumford**.
- Mumford and Desolneux's book: General Pattern Theory.
- Reasons: advances in mathematical models (e.g. Markov graphical models, sparse coding, stochastic context free grammars), inference algorithms, real datasets.

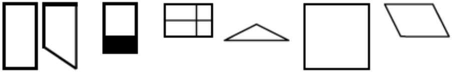

Foundations and Trends in Computer Graphics and Vision
A Stochastic Grammar of Images
Song-Chun Zhu and David Mumford
now


Pattern Theory
The Stochastic Analysis of Real-World Signals
David Mumford • Agnès Desolneux

Example: house description

[Stanchev and Green]

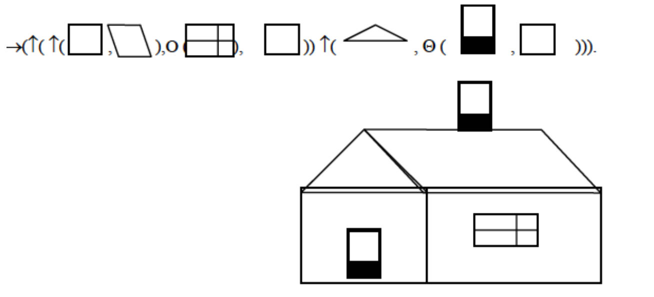
T=



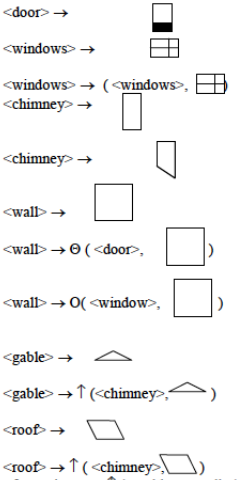
→ O Θ ↑ ( )

$$V_N = \{\langle door\rangle, \langle windows\rangle, \langle chimney\rangle, \langle wall\rangle, \langle gable\rangle, \langle roof\rangle, \langle frontview\rangle, \langle sideview\rangle, \langle house\rangle\}$$

$$S = \{\langle house\rangle\}$$



Rules



<door> →

<windows> →

<windows> → (<windows>,)

<chimney> →

<chimney> →

<wall> →

<wall> → Θ (<door>,)

<wall> → O(<window>,)

<gable> →

<gable> → ↑ (<chimney>,)

<roof> →

<roof> → ↑ (<chimney>,)

<front view> → ↑ (<gable>, <wall>)

<side view> → ↑ (<roof>, <wall>)

<house> → <front view>

<house> → (<house>, <side view>)

Modern grammatical approaches

Stochastic Context-Free Grammars

$G = (N, T, R, \{P_l\})$

- Production rules are assigned with probabilities: $P_l(R_l), \sum_{R_l} P_l(R_l)$.
- Sentences from the language are assigned with probabilities:

$$L(G) = \left\{ (w, P(w)) \mid S \underset{G}{\overset{*}{\Rightarrow}} w, w \in (T \cup N)^* \right\}$$

1. Start from root node S.
2. Apply rules sampled from a distribution.
3. Outputs a parse tree, leaf node are terminals.
4. Parse tree β is a set of nodes V, each node has label $l(v)$.
5. Prob. of the tree: $P_{tree}(\beta \mid roots) = \prod_{v \in V} P_{l(v)}(R(v))$.

SCFG for vision applications

- SCFG ignores spatial constraints between objects, and objects attributes.
- Define nodes attributes representing objects properties $\Longrightarrow$ attributed SCFG.
- Move from parse trees to parse graphs to deal with spatial constraints.
- Lost of independence property.

Attributed SCFG and AND-OR Graph

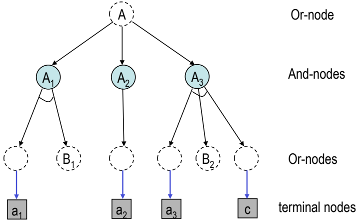
- State variables y_μ defined at nodes $\mu \in \mathcal{V}$.
- Edges $\mathcal{E}$ specify connections between nodes and define their cliques.
- Potential functions defined over the cliques.

$$P(y \mid I; \alpha) = \frac{1}{Z(I, \alpha)} exp(-\alpha.\Phi(I, \alpha))$$

$$E(y \mid I) = \sum_{\mu \in V^{AND}(t)} \alpha_\mu^{AND} \Phi^{AND}(z_\mu, z_{Ch_\mu}) + \sum_{\mu \in V^{OR}(t)} \alpha_\mu^{OR} \Phi^{OR}(z_\mu, t_\mu, z_{t_\mu}) + \sum_{\mu \in V^{LEAF}(t)} \alpha_\mu^{D} \Phi^{D}(I, z_\mu)$$

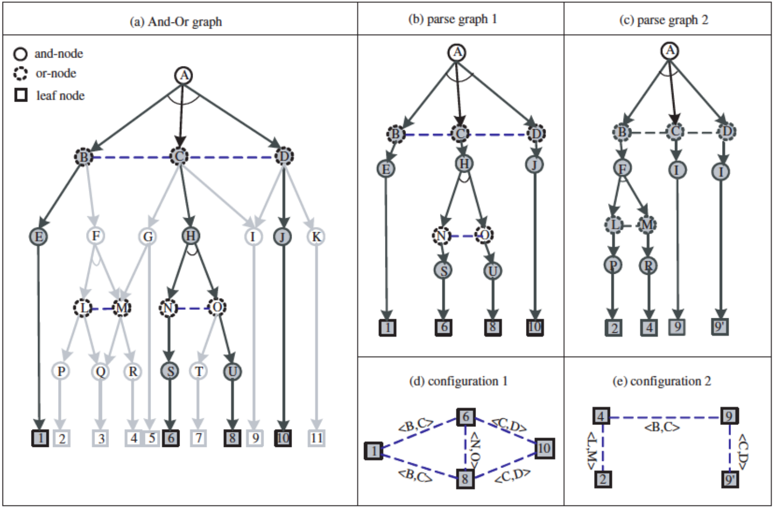
- Energy can be decomposed recursively from level to level.
- Enables dynamic programming.

$G = (\mathcal{V}, \mathcal{E})$



From Zhu and Mumford

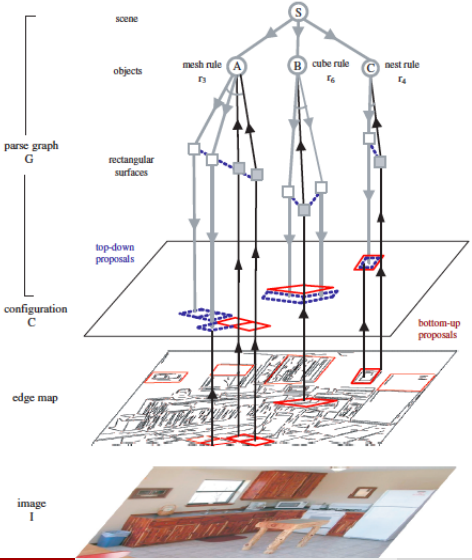
AND-OR Graph and Parse Graph



From Zhu and Mumford

Some applications

Image parsing bottom-up top-down process (Zhu and Mumford)



Demo: Natural language Query based on Joint Parsing

UCLA Center for Vision, Cognition, Learning and Art

You are here!

1 Preliminaries

2 Graph-based reasoning for joint segmentation and recognition

3 CSP-based approaches

4 Stochastic grammars and image parsing

5 Conclusion and open problems

Concluding remarks and open problems

- Image understanding is an AI task.
- Importance of spatial information and calculi.
- Focus on a subset of approaches, but there are many many others.
- Other interpretation tasks, problems, and applications.

Open problems

- Learning knowledge graphs (e.g. terminologies).
- Do all interpretation problems are learning-oriented?
- More on logical reasoning: abduction, revision, spatio-temporal reasoning.
- Combining different types of uncertainty.