Efficient Inference Algorithms for Scene Understanding Problems

Karteek Alahari

Inria Grenoble – Rhône-Alpes

Joint work with Anoop Cherian, Pushmeet Kohli, Lubor Ladicky, Julien Mairal, Chris Russell, Cordelia Schmid, Paul Sturgess, Philip Torr

informatiques mathématiques Inria

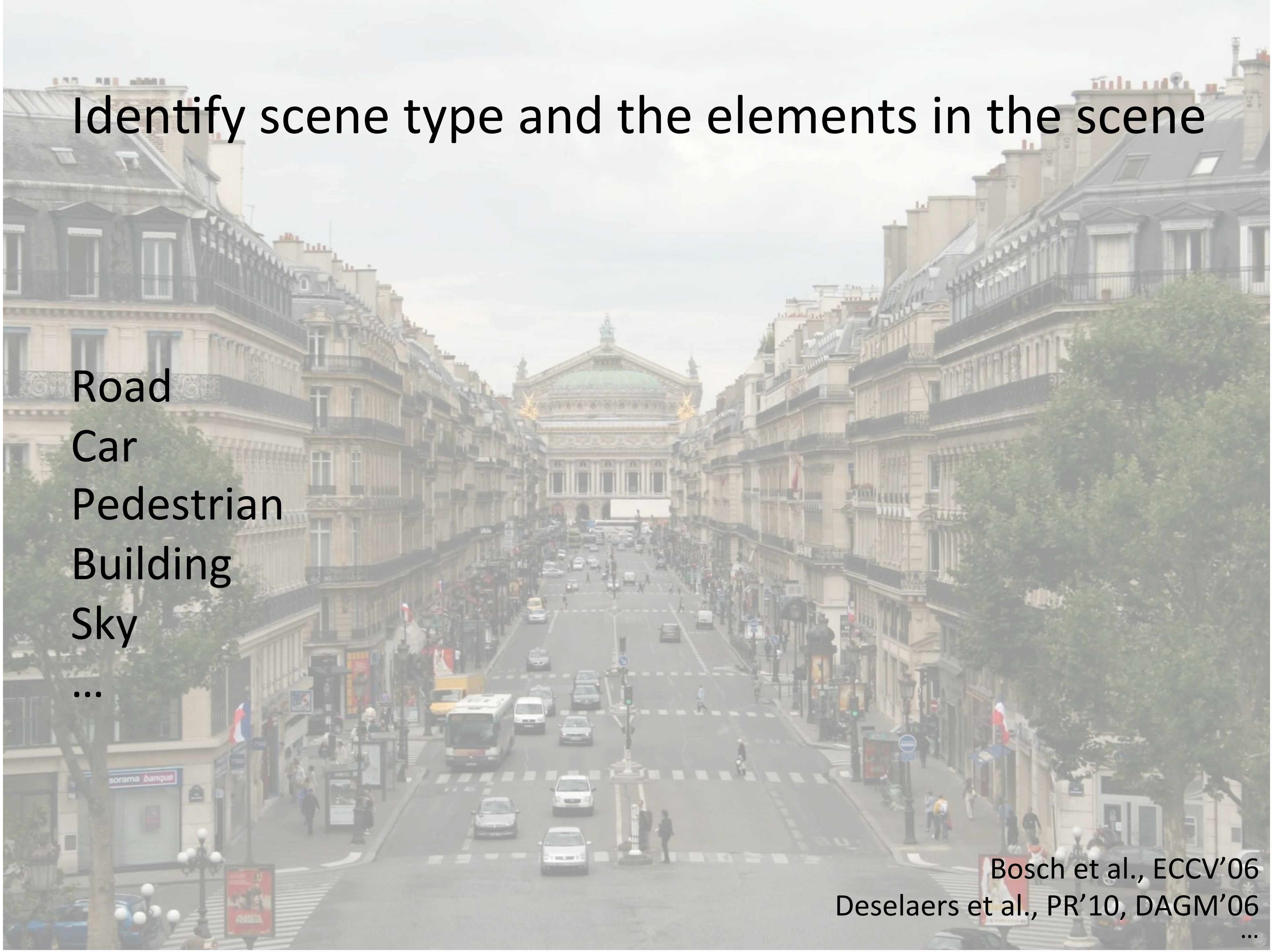
LABORATOIRE JEAN KUNTZMANN MATHÉMATIQUES APPLIQUÉES - INFORMATIQUE

cnrs

UNIVERSITÉ GRENOBLE ALPES

What is scene understanding?

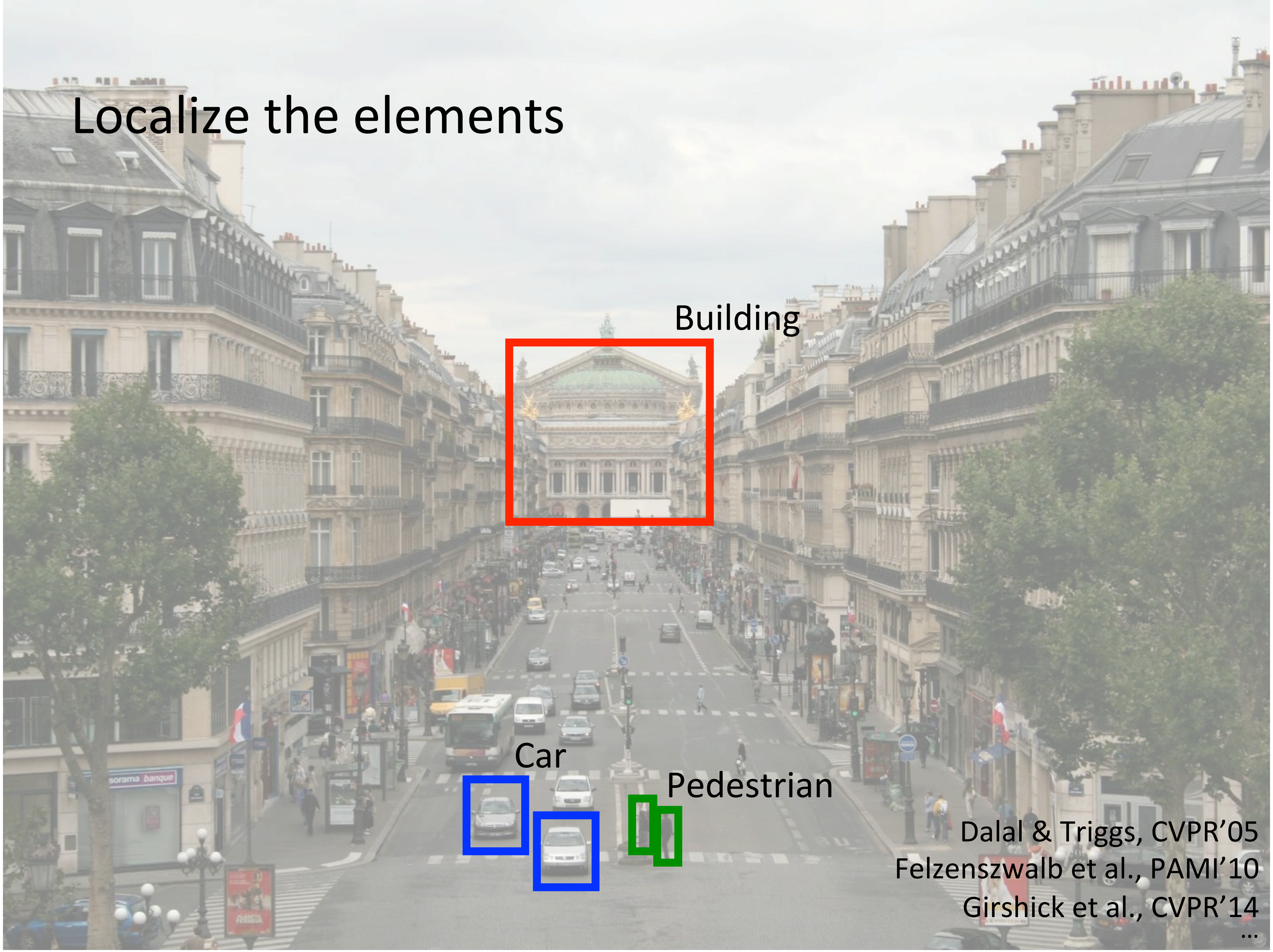
Identify scene type and the elements in the scene



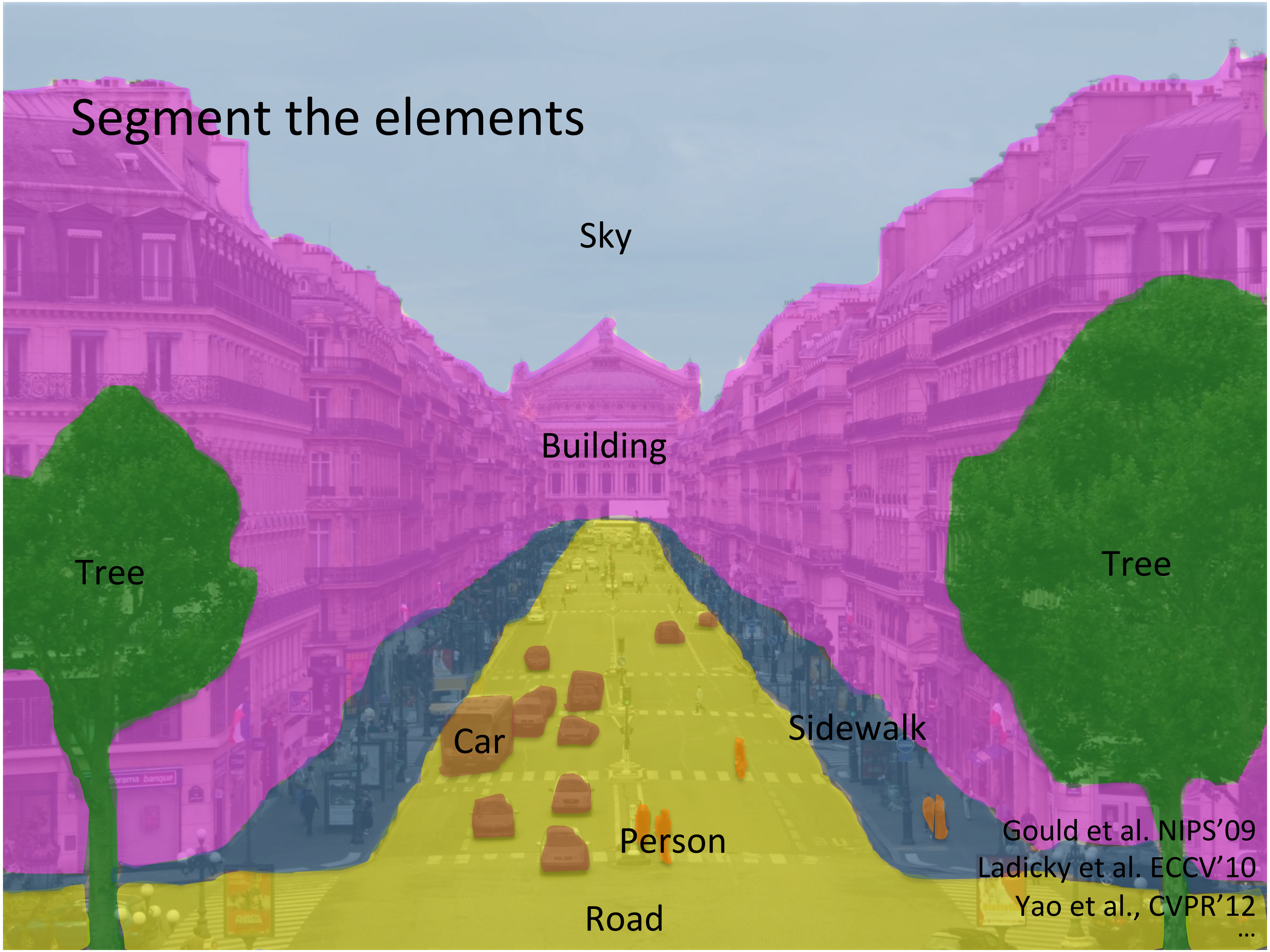
Road
Car
Pedestrian
Building
Sky
…

Bosch et al., ECCV’06
Deselaers et al., PR’10, DAGM’06
…

Localize the elements



Dalal & Triggs, CVPR’05
Felzenszwalb et al., PAMI’10
Girshick et al., CVPR’14
…


Segment the elements
Sky
Building
Tree
Tree
Sidewalk
Car
Person
Road
Gould et al. NIPS'09
Ladicky et al. ECCV'10
Yao et al., CVPR'12
…

Not all images are the same!

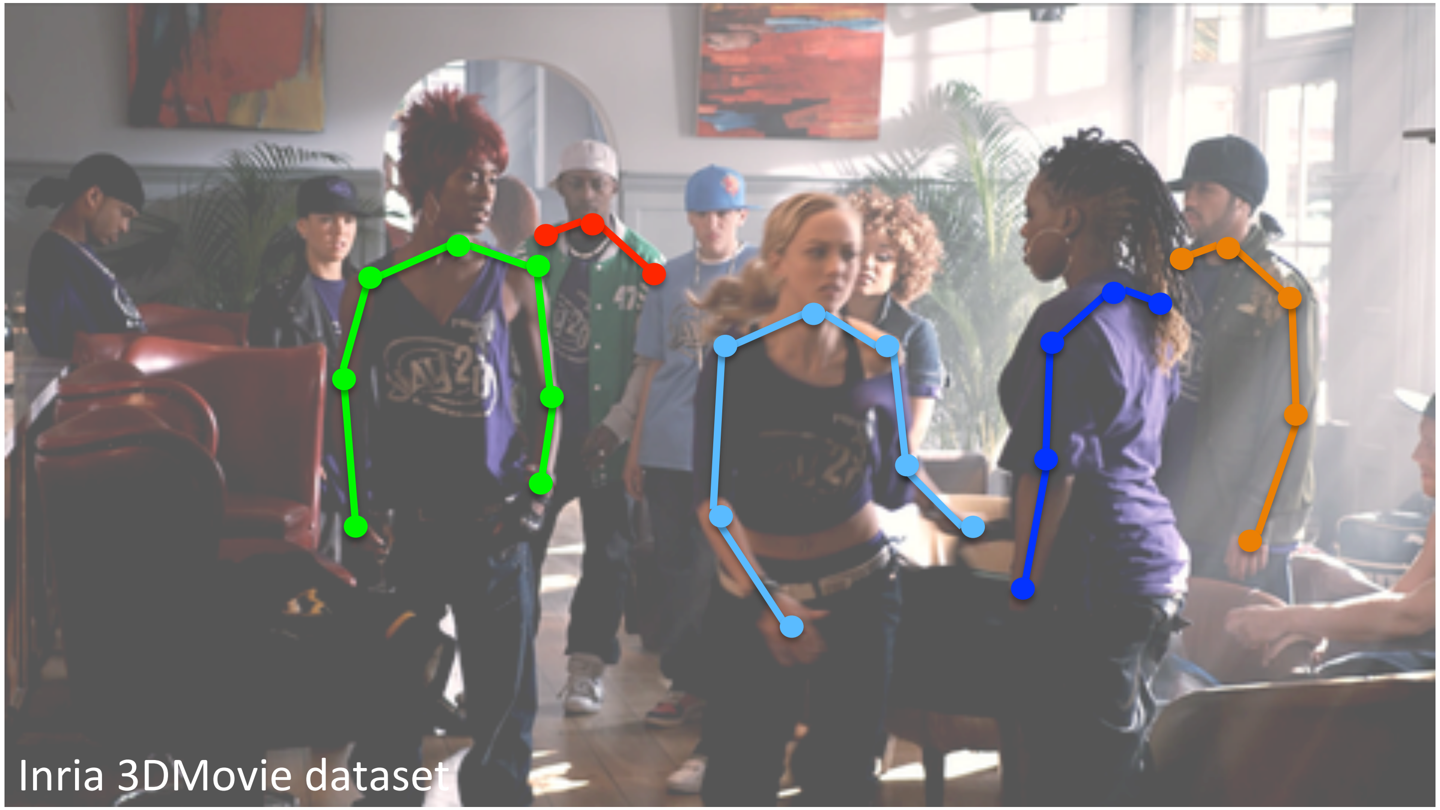
Image courtesy: L'Obs

Analyze people in a scene

Inria 3DMovie dataset

Analyze people in a scene

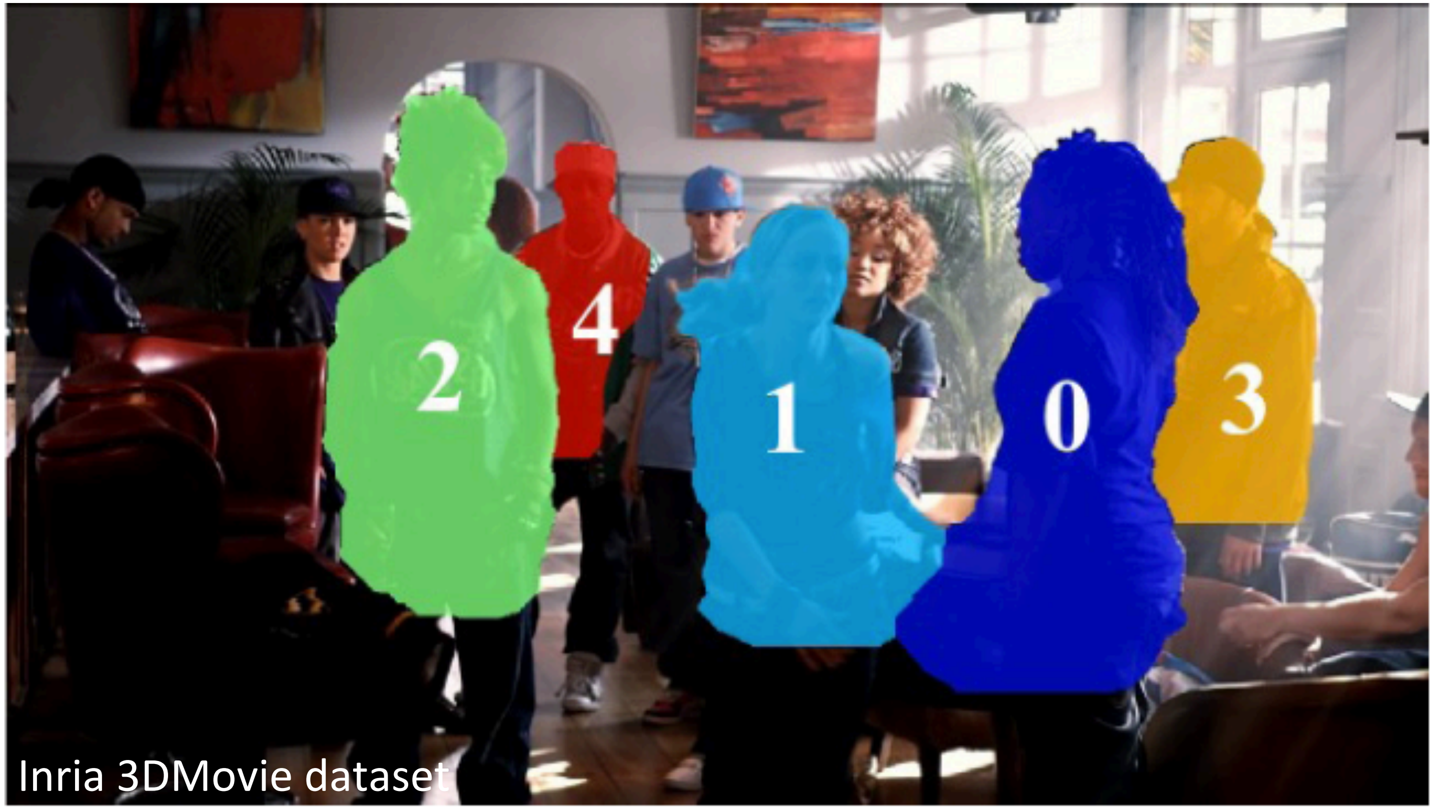
Estimate poses (body-joint locations)



Yang & Ramanan, PAMI'12
Tompson et al., CVPR'14
...

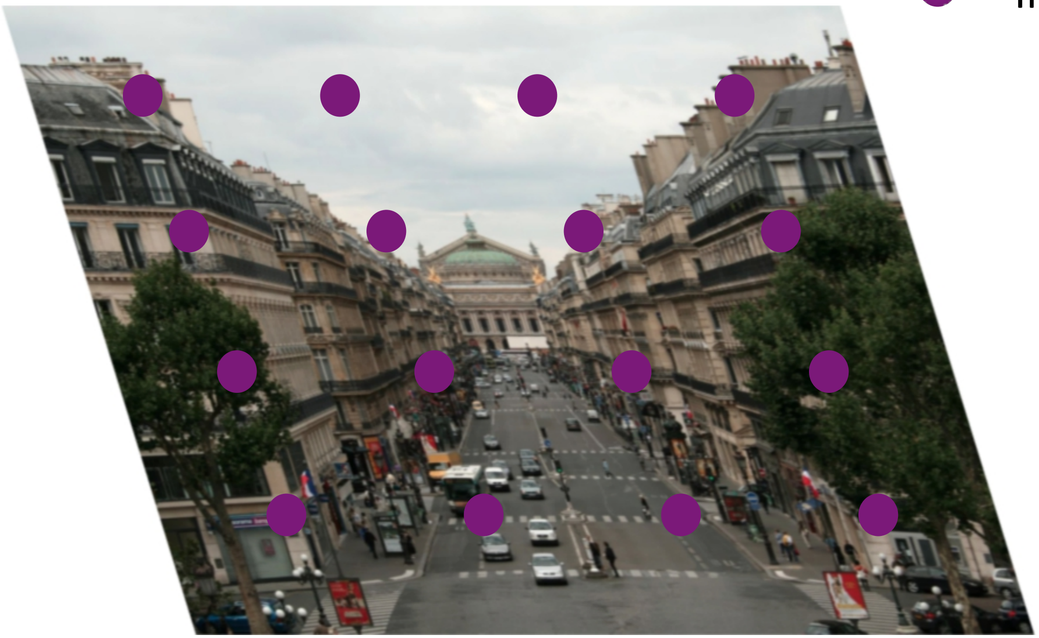
Analyze people in a scene

Segment the individuals



Yang et al., PAMI’11
Shotton et al., CVPR’11
…

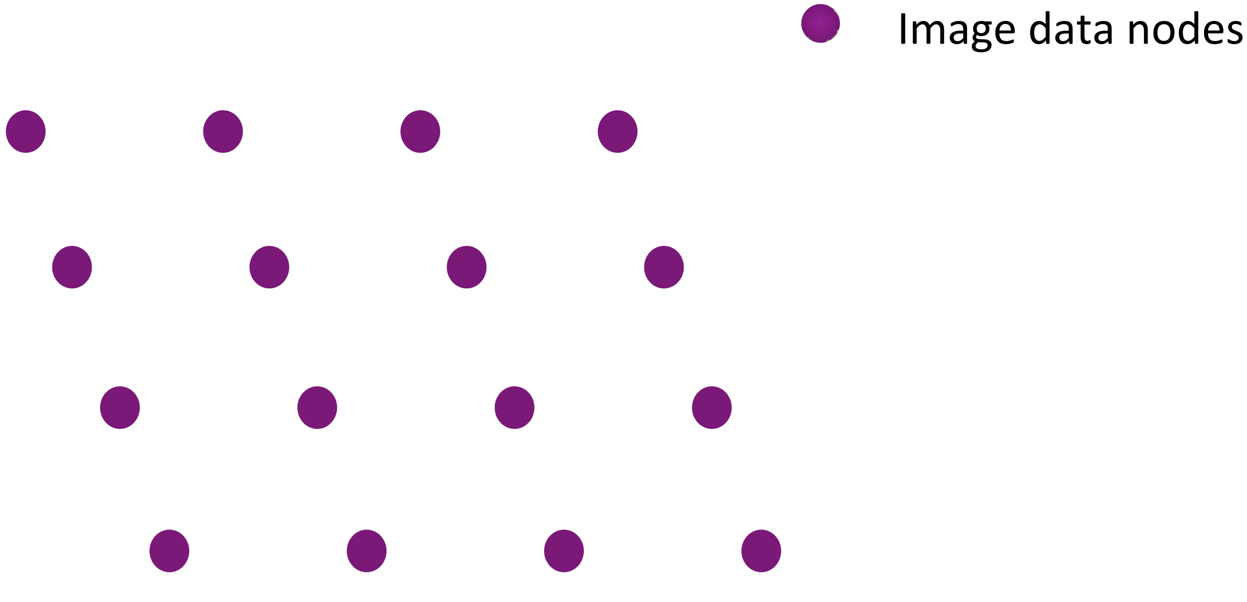
Scene Understanding



● Image data nodes

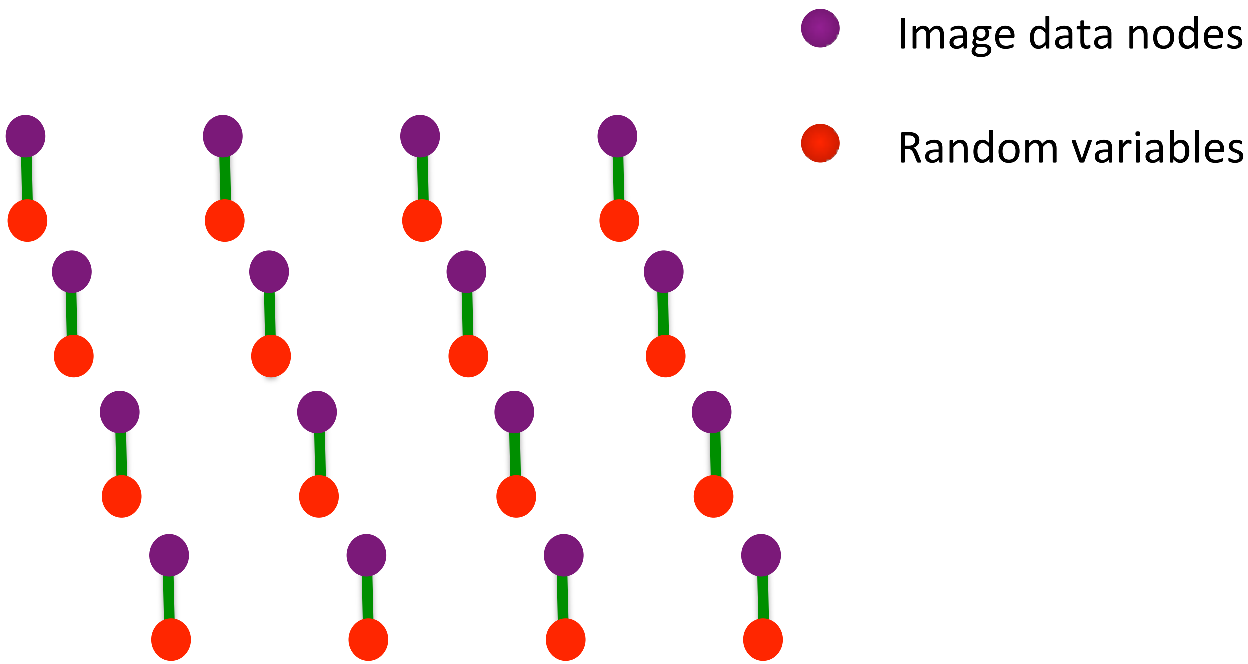
Markov Random Fields [Preston ’74]

Scene Understanding



Markov Random Fields [Preston '74]

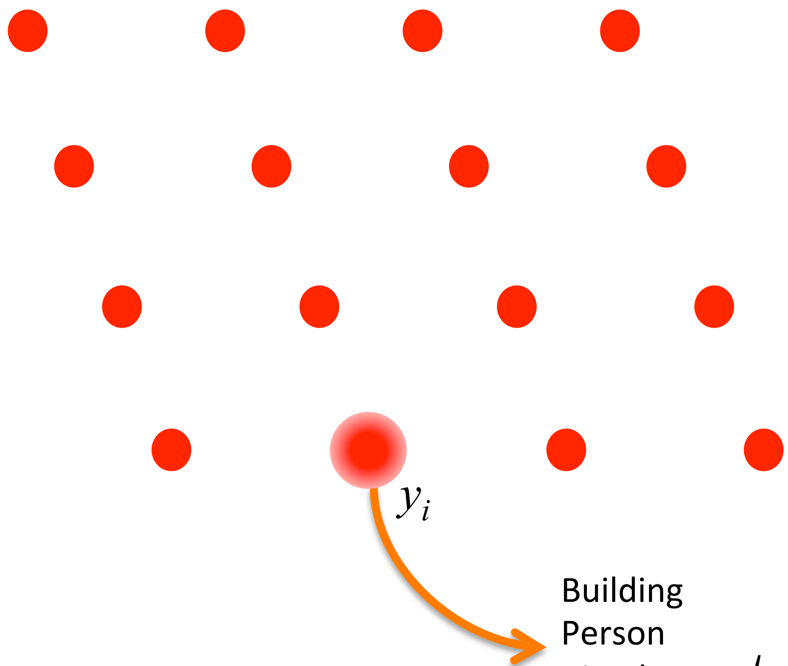
Scene Understanding



Markov Random Fields [Preston ’74]

Scene Understanding

● Random variables



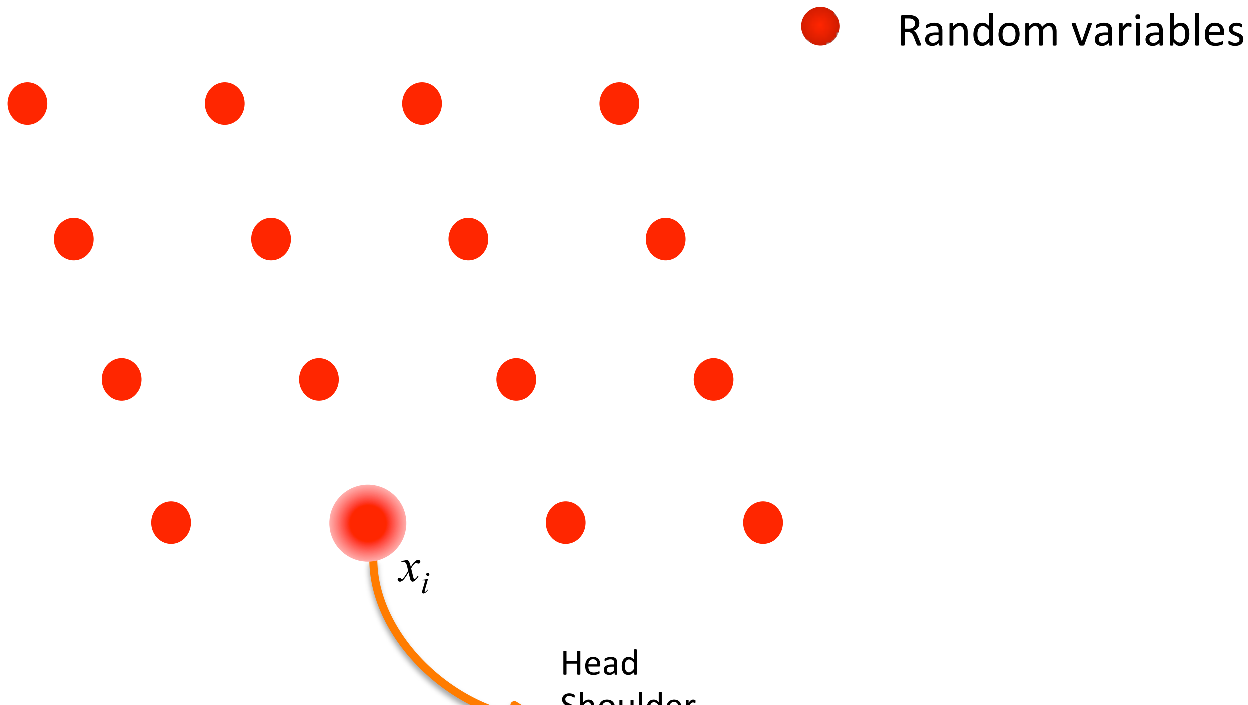
Building
Person
Bicycle
Road
Sky
…

$\psi_i(y_i)$: **Unary cost**

e.g, object recognition

Markov Random Fields [Preston '74]

Scene Understanding



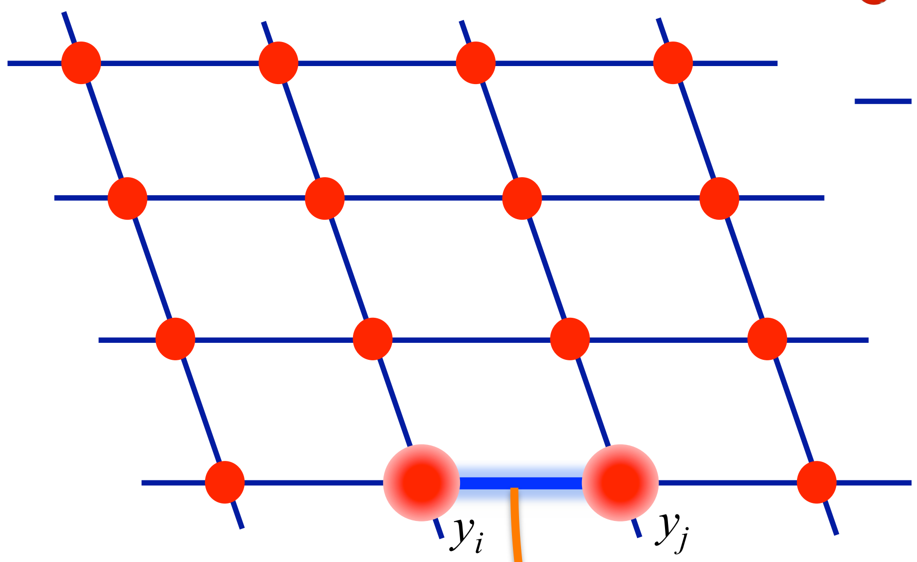
Head
Shoulder
Elbow
Wrist
…

$\psi_i(y_i)$ **: Unary cost**

e.g, pose estimation

Markov Random Fields [Preston '74]

Scene Understanding



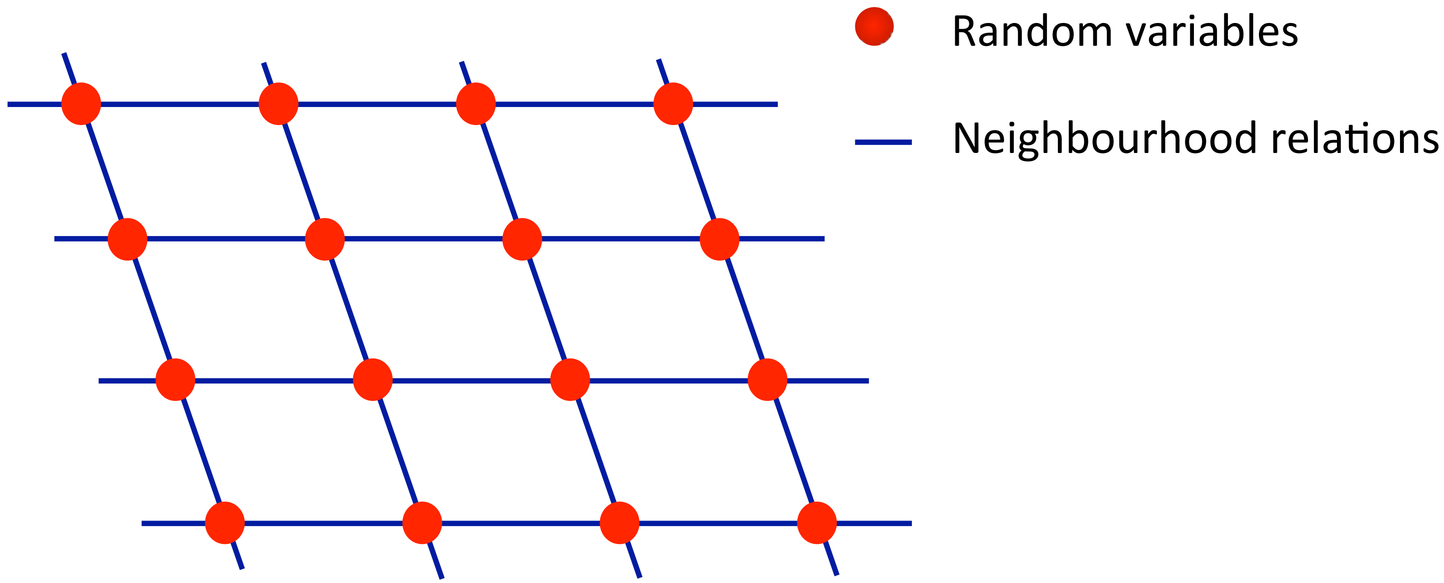
Random variables

Neighbourhood relations

$\psi_{ij}(y_i, y_j)$: **Pairwise cost**

Markov Random Fields [Preston '74]

Scene Understanding



$$E(\mathbf{y}) = \sum_{i \in \mathcal{V}} \psi_i(y_i) + \sum_{(i,j) \in \mathcal{E}} \psi_{ij}(y_i, y_j)$$ **: Cost of labelling**

Scene Understanding

- In the context of such labelling problems
  - Learning parameters of the function
  - Modelling image priors & temporal constraints
  - Performing inference

Scene Understanding

- In the context of such labelling problems
  - Learning parameters of the function
  - Modelling image priors & temporal constraints
  - Performing inference

Parameter learning

- Consider a 2-label problem (for now)

- e.g., binary image segmentation

- Also, consider a pairwise random field

Parameter learning

- Conditional probability of a labelling **y**:

$$\Pr(\mathbf{y}|\mathbf{x},\boldsymbol{\theta}) \quad = \quad \frac{1}{Z(\boldsymbol{\theta})}\prod_{i\in\mathcal{V}}\exp(y_i\boldsymbol{\theta}_u^\top h_i(\mathbf{x}))\prod_{(i,j)\in\mathcal{E}}\exp(y_i y_j\boldsymbol{\theta}_p^\top \nu_{ij}(\mathbf{x}))$$

Parameter learning

- Conditional probability of a labelling **y**:

$$\Pr(\mathbf{y}|\mathbf{x},\boldsymbol{\theta}) \quad = \quad \frac{1}{Z(\boldsymbol{\theta})}\prod_{i\in\mathcal{V}}\exp(y_i\boldsymbol{\theta}_u^\top h_i(\mathbf{x}))\prod_{(i,j)\in\mathcal{E}}\exp(y_iy_j\boldsymbol{\theta}_p^\top \nu_{ij}(\mathbf{x}))$$

The label taken by pixel x_i = -1 or 1

Parameter learning

- Conditional probability of a labelling **y**:

$$\Pr(\mathbf{y}|\mathbf{x},\boldsymbol{\theta}) \quad = \quad \frac{1}{Z(\boldsymbol{\theta})} \prod_{i \in \mathcal{V}} \exp(y_i \boldsymbol{\theta}_u^\top \underbrace{h_i(\mathbf{x})}_{\text{Unary features}}) \prod_{(i,j) \in \mathcal{E}} \exp(y_i y_j \boldsymbol{\theta}_p^\top \underbrace{\nu_{ij}(\mathbf{x})}_{\text{Pairwise features}})$$

Parameter learning

- Conditional probability of a labelling **y**:

$$\Pr(\mathbf{y}|\mathbf{x},\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})}\prod_{i\in\mathcal{V}}\exp(y_i\boldsymbol{\theta}_u^\top h_i(\mathbf{x}))\prod_{(i,j)\in\mathcal{E}}\exp(y_i y_j\boldsymbol{\theta}_p^\top \nu_{ij}(\mathbf{x}))$$

Unary parameters

Pairwise parameters

Learning Approaches

- Maximum likelihood-based
- Large margin based
- Other iterative methods

Maximum likelihood-based

- Compute $\hat{\boldsymbol{\theta}}$ which maximizes likelihood

$$\hat{\boldsymbol{\theta}} \quad = \quad \underset{\boldsymbol{\theta}}{\arg\max} \; Pr(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta})$$

- Hard to solve for vision problems

Maximum likelihood-based

- Why is it hard?

$$\Pr(\mathbf{y}|\mathbf{x},\boldsymbol{\theta}) \quad = \quad \frac{1}{Z(\boldsymbol{\theta})} \prod_{i \in \mathcal{V}} \exp(y_i \boldsymbol{\theta}_u^\top h_i(\mathbf{x})) \prod_{(i,j) \in \mathcal{E}} \exp(y_i y_j \boldsymbol{\theta}_p^\top \nu_{ij}(\mathbf{x}))$$

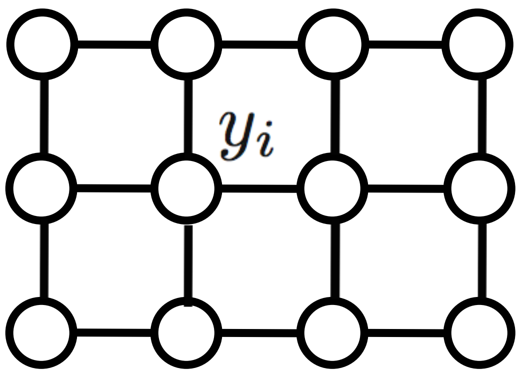
Partition function – Infeasible to compute
for vision problems (2^N values)

- Approximate methods: Sampling, mode of the distribution, pseudo-likelihood

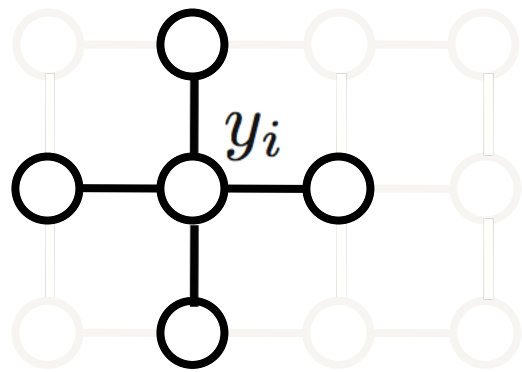
e.g., Kumar and Hebert, NIPS'03

Pseudo-likelihood

$$\Pr(\mathbf{y}|\mathbf{x},\boldsymbol{\theta}) = \prod_{\substack{i\in\mathcal{V}\\(i,j)\in\mathcal{E}}} \Pr(y_i|\mathbf{x}, y_j, \boldsymbol{\theta})$$



Likelihood



Pseudo-likelihood

Besag, The Statistician, 1975

Pseudo-likelihood

$$\Pr(\mathbf{y}|\mathbf{x},\boldsymbol{\theta}) = \prod_{\substack{i\in\mathcal{V}\\(i,j)\in\mathcal{E}}} \Pr(y_i|\mathbf{x}, y_j, \boldsymbol{\theta})$$

$$\frac{1}{z_i(\boldsymbol{\theta})}\exp(y_i\boldsymbol{\theta}_u^\top h_i(\mathbf{x}))\cdots$$

✓ Easy to compute

✗ But, can lead to poor accuracy

Max-Margin Learning

- Consider

$$\log \Pr(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta}) = \boldsymbol{\theta}\mathbf{X}\mathbf{y} - \log Z(\boldsymbol{\theta})$$

- Maximize confidence margin in true label assignment
- In other words, maximize

$$\log \Pr(\hat{\mathbf{y}}|\mathbf{x}, \boldsymbol{\theta}) - \log \Pr(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta}) = \boldsymbol{\theta}\mathbf{X}(\hat{\mathbf{y}} - \mathbf{y})$$

True label

All other label assignments

Taskar et al. NIPS'04, Tsochantaridis et al. ICML'04

Max-Margin Learning

- Formulated as SVM learning problem

$$\min \quad \frac{1}{2}||\boldsymbol{\theta}||^2 + C\xi$$

$$\text{subject to} \quad \boldsymbol{\theta}\mathbf{F}(\hat{\mathbf{y}} - \mathbf{y}) \geq N - \hat{\mathbf{y}}_i^\top \mathbf{y}_i - \xi, \forall \mathbf{y} \in \mathcal{Y}$$

Max-Margin Learning

- Formulated as SVM learning problem

$$\min \quad \frac{1}{2}||\boldsymbol{\theta}||^2 + C\xi$$

$$\text{subject to} \quad \boldsymbol{\theta}\mathbf{F}(\hat{\mathbf{y}} - \mathbf{y}) \geq N - \hat{\mathbf{y}}_i^\top \mathbf{y}_i - \xi, \forall \mathbf{y} \in \mathcal{Y}$$

Exponential no. of constraints!!
(**$2^N - 1$**)

Max-Margin Learning

- Formulated as SVM learning problem

$$\min \quad \frac{1}{2}||\boldsymbol{\theta}||^2 + C\xi$$

~~subject to~~ $\cancel{\boldsymbol{\theta}\mathbf{F}(\hat{\mathbf{y}} - \mathbf{y}) \geq N - \hat{\mathbf{y}}_i^\top \mathbf{y}_i - \xi, \forall \mathbf{y} \in \mathcal{Y}}$

$$\text{subject to} \quad \boldsymbol{\theta}\mathbf{F}\hat{\mathbf{y}} - N + \xi \geq \max_{\mathbf{y} \in \mathcal{Y}} \; \boldsymbol{\theta}\mathbf{F}\mathbf{y} - \hat{\mathbf{y}}_i^\top \mathbf{y}_i$$

Inference problem to find the **most violated constraint**
e.g., graph cuts

Max-Margin Learning

- Formulated as SVM learning problem

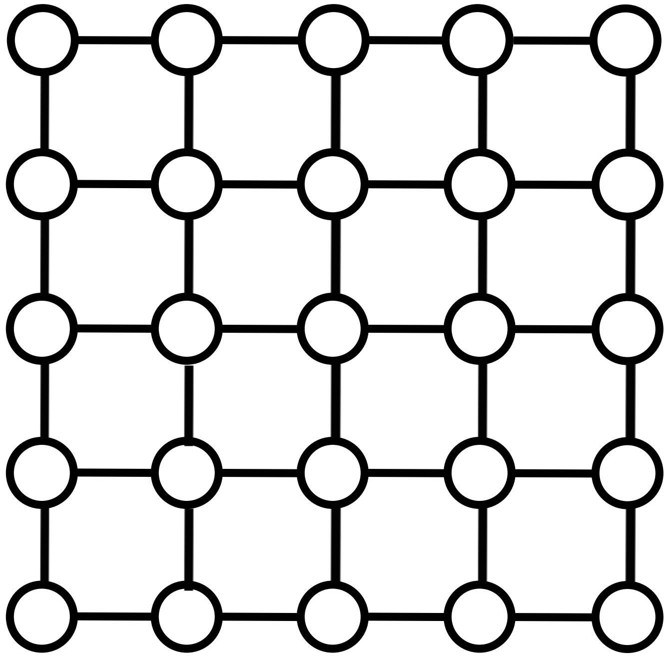
$$\min \quad \frac{1}{2}||\boldsymbol{\theta}||^2 + C\xi$$

$$\text{subject to} \quad \boldsymbol{\theta}\mathbf{F}\hat{\mathbf{y}} - N + \xi \geq \max_{\mathbf{y}\in\mathcal{Y}} \quad \boldsymbol{\theta}\mathbf{F}\mathbf{y} - \hat{\mathbf{y}}_i^\top \mathbf{y}_i$$

✓ Eliminates the partition function

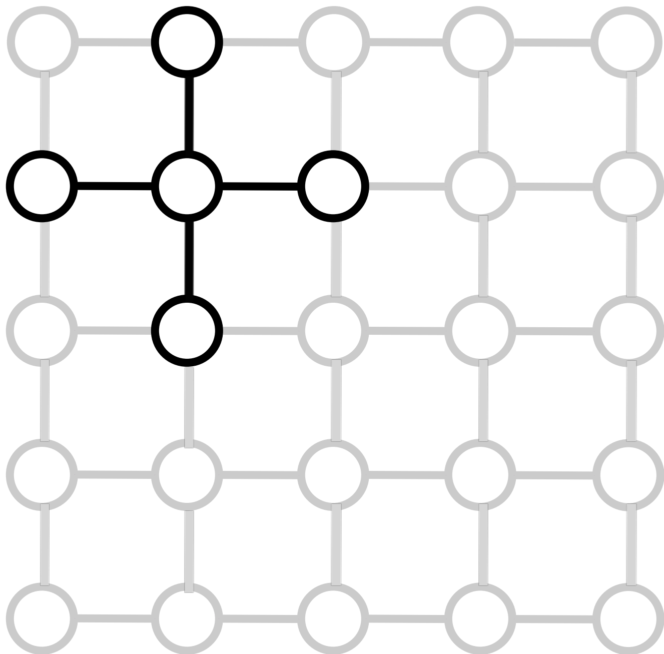
✗ Limited by the inference step

Our Piecewise Model



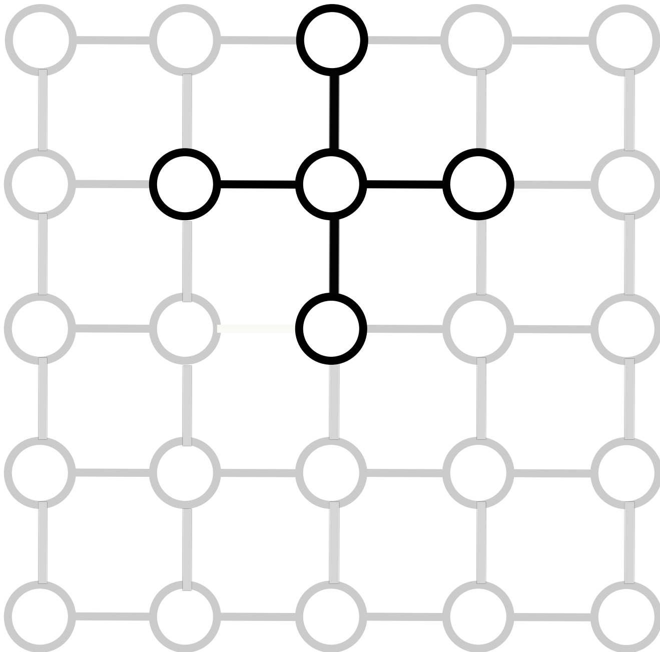
Alahari et al., CVPR 2010

Our Piecewise Model



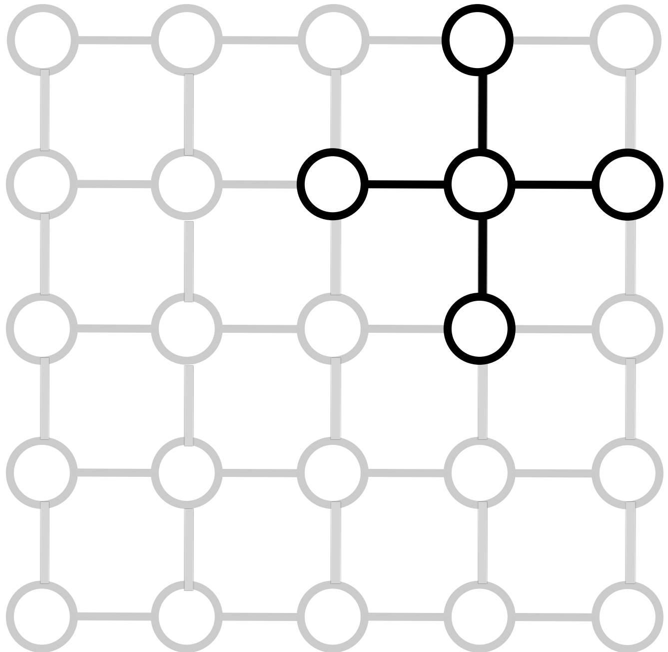
Training Exemplar 1

Our Piecewise Model



Training Exemplar 2

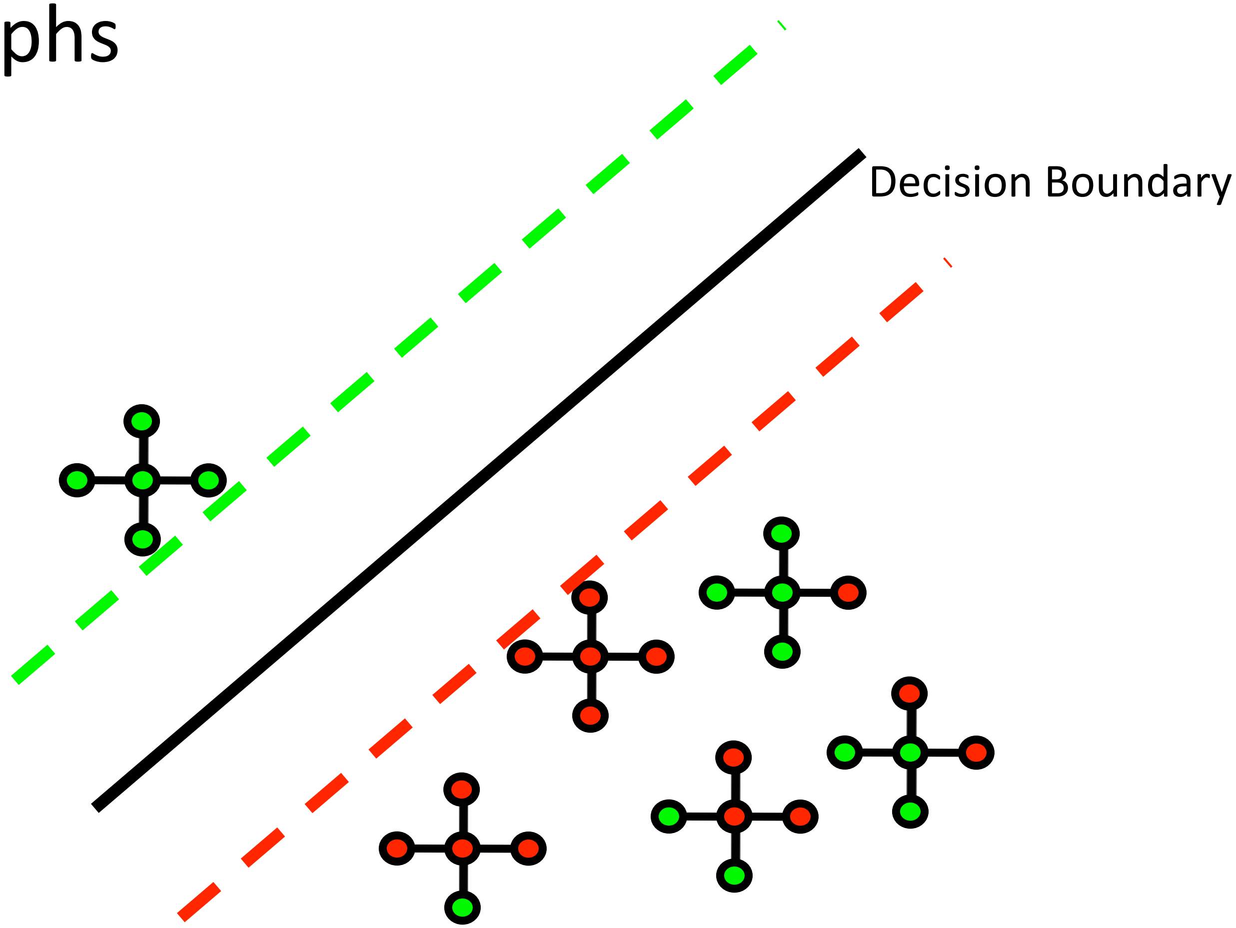
Our Piecewise Model



Training Exemplar 3

Our Piecewise Model

- Formulate a max-margin problem on these sub-graphs



Our Piecewise Model

- Formulate a max-margin problem on these sub-graphs

$$\min \quad \frac{1}{2}||\boldsymbol{\theta}||^2 + C\xi_s$$

$$\text{subject to} \quad \boldsymbol{\theta}\mathbf{F}_s(\hat{\mathbf{y}}_s - \mathbf{y}_s) \geq N - \hat{\mathbf{y}}_{s i}^\top \mathbf{y}_{s i} - \xi_s, \forall \mathbf{y}_s \in \mathcal{Y}_s$$

Our Piecewise Model

- Formulate a max-margin problem on these sub-graphs

Slack variable

$$\min \quad \frac{1}{2}||\boldsymbol{\theta}||^2 + C\xi_s$$

$$\text{subject to} \quad \boldsymbol{\theta}\mathbf{F}_s(\hat{\mathbf{y}}_s - \mathbf{y}_s) \geq N - \hat{\mathbf{y}}_{s\,i}^{\top}\mathbf{y}_{s\,i} - \xi_s, \forall \mathbf{y}_s \in \mathcal{Y}_s$$

Our Piecewise Model

- Formulate a max-margin problem on these sub-graphs

$$\min \quad \frac{1}{2}||\boldsymbol{\theta}||^2 + C\xi_s$$

$$\text{subject to} \quad \boldsymbol{\theta}\mathbf{F}_s(\hat{\mathbf{y}}_s - \mathbf{y}_s) \geq N - \hat{\mathbf{y}}_{s\,i}^{\top}\mathbf{y}_{s\,i} - \xi_s, \forall \mathbf{y}_s \in \mathcal{Y}_s$$

Exponential # constraints again!

Our Piecewise Model

- Formulate a max-margin problem on these sub-graphs

$$\min \quad \frac{1}{2}||\boldsymbol{\theta}||^2 + C\xi_s$$

$$\text{subject to} \quad \boldsymbol{\theta}\mathbf{F}_s(\hat{\mathbf{y}}_s - \mathbf{y}_s) \geq N - \hat{\mathbf{y}}_{si}^{\top}\mathbf{y}_{si} - \xi_s, \forall \mathbf{y}_s \in \mathcal{Y}_s$$

- Can be solved efficiently using BP messages

Piecewise: Results

- Middlebury Stereo

Method	Art	Books	Dolls	Laundry	Moebius	Reindeer	*Average*
Li and Huttenlocher	14.66	19.12	12.70	19.16	10.88	11.72	14.71
Our method	**12.94**	**16.24**	**12.21**	**16.72**	**10.82**	**11.10**	**13.34**

- Man-made Structures

Method	FP per image	DR %
MRF	2.36	57.20
DRF1	**1.37**	70.50
DRF2	1.76	72.54
Our method	1.40	**72.60**

So far …

- In the context of labelling problems
  - Learning parameters of the function

  Next:
  - Modelling image priors & performing inference

Modelling image priors & inference

- Energy function: $E(\mathbf{y}) = \sum_{i \in \mathcal{V}} \psi_i(y_i) + \sum_{(i,j) \in \mathcal{E}} \psi_{ij}(y_i, y_j)$

- How to minimize this efficiently?

- New image priors, e.g., higher-order term: $\sum_{\mathbf{c} \in \mathcal{S}} \psi_{\mathbf{c}}(y_{\mathbf{c}})$

Example: Image Segmentation

$$E(\mathbf{y}) = \sum_{i \in \mathcal{V}} \psi_i(y_i) + \sum_{(i,j) \in \mathcal{E}} \psi_{ij}(y_i, y_j)$$

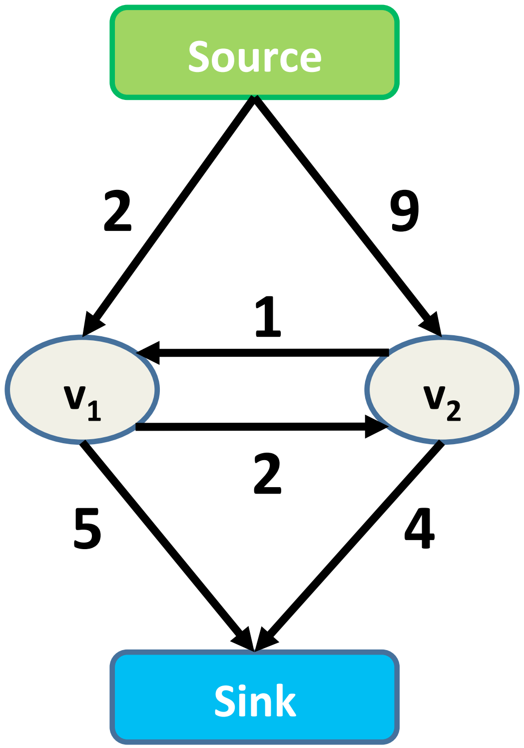
E: {0, 1}^{N} → $\mathcal{R}$
0 → *fg*
1 → *bg*

N: number of pixels

How to minimize $E(y)$?

st-mincut algorithm, belief propagation, α -expansion

The st-Mincut Problem



Graph (V, E, C)

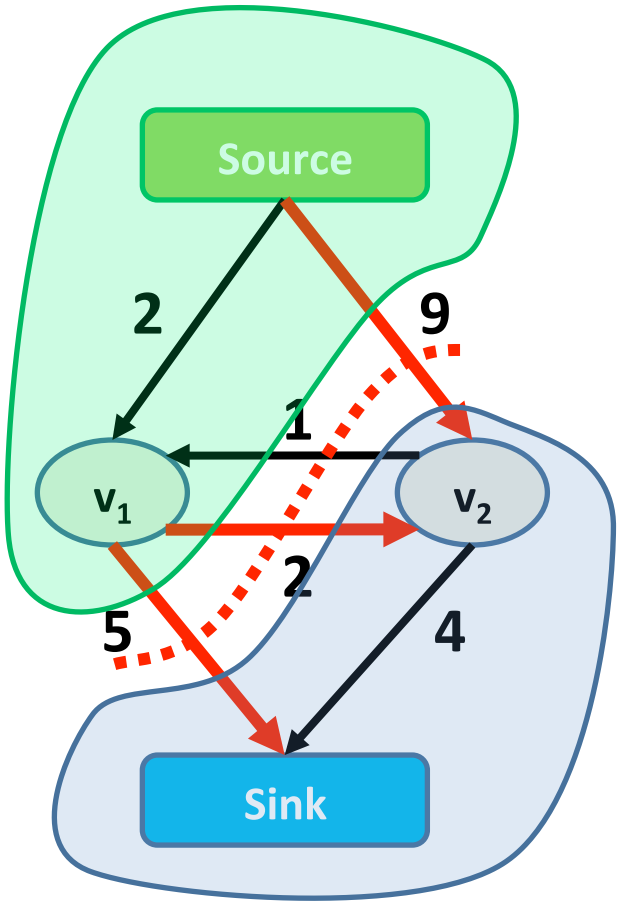
Vertices V = $\{v_1, v_2 \ldots v_n\}$

Edges E = $\{(v_1, v_2) \ldots.\}$

Costs C = $\{c_{(1, 2)} \ldots.\}$

Slide courtesy: Pushmeet Kohli

The st-Mincut Problem



5 + 2 + 9 = 16

Slide courtesy: Pushmeet Kohli

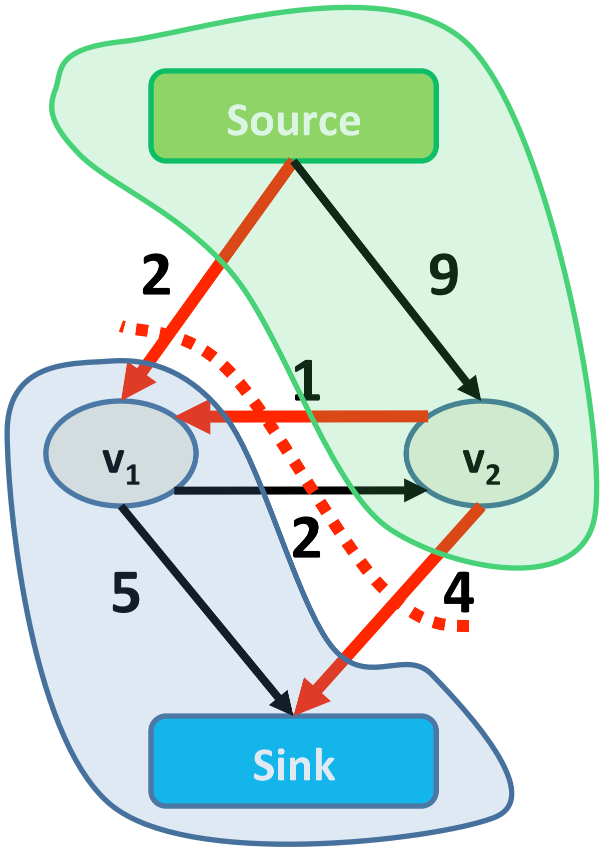
What is an st-cut?

An st-cut (**S**,**T**) divides the nodes between source and sink.

What is the cost of an st-cut?

Sum of cost of all edges going from S to T

The st-Mincut Problem



2 + 1 + 4 = 7

Slide courtesy: Pushmeet Kohli

What is an st-cut?

An st-cut (**S**,**T**) divides the nodes between source and sink.

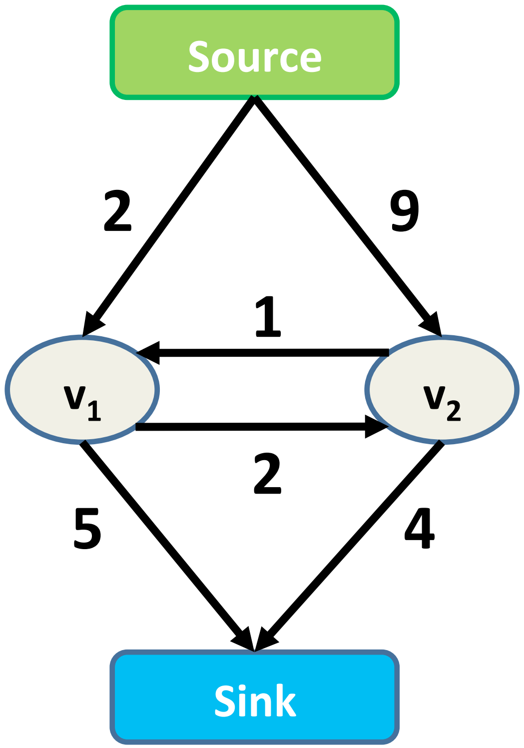
What is the cost of an st-cut?

Sum of cost of all edges going from S to T

What is the st-mincut?

st-cut with the minimum cost

How to compute the st-mincut?



Solve the dual **maximum flow** problem

Compute the maximum flow between Source and Sink

Constraints

Edges: Flow < Capacity

Nodes: Flow in = Flow out

Min-cut/Maxflow Theorem

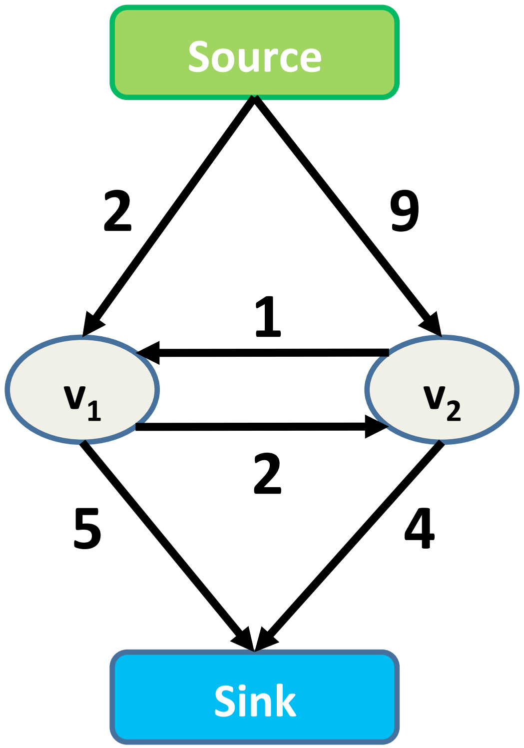
In every network, the maximum flow equals the cost of the st-mincut

Dantzig, Dinitz, Edmonds & Karp, Karzanov, Ford & Fulkerson, Goldberg & Tarjan, …

Slide courtesy: Pushmeet Kohli

Maxflow Algorithms

Flow = 0



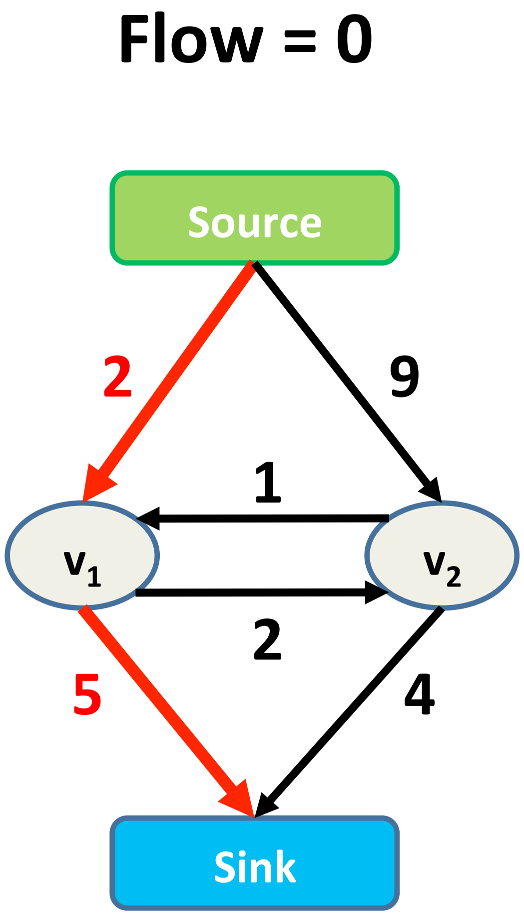
Augmenting Path Based Algorithms

1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path
3. Repeat until no path can be found

Algorithms assume non-negative capacity

Slide courtesy: Pushmeet Kohli

Maxflow Algorithms



Augmenting Path Based Algorithms

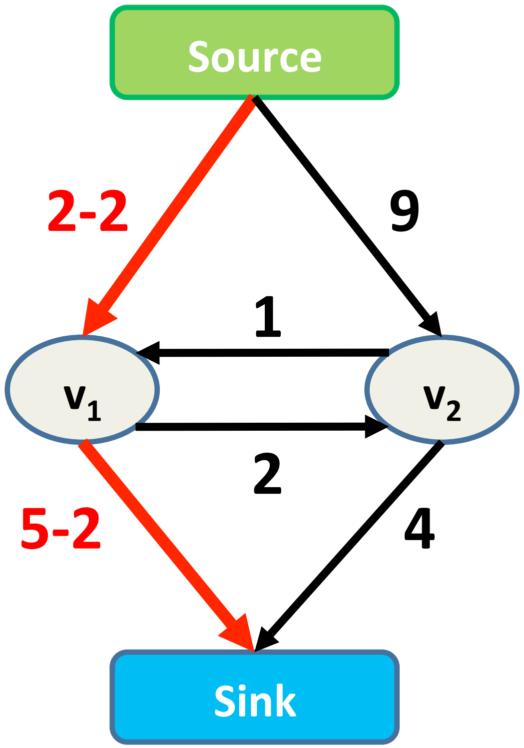
1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path
3. Repeat until no path can be found

Algorithms assume non-negative capacity

Slide courtesy: Pushmeet Kohli

Maxflow Algorithms

Flow = 0 + 2



Augmenting Path Based Algorithms

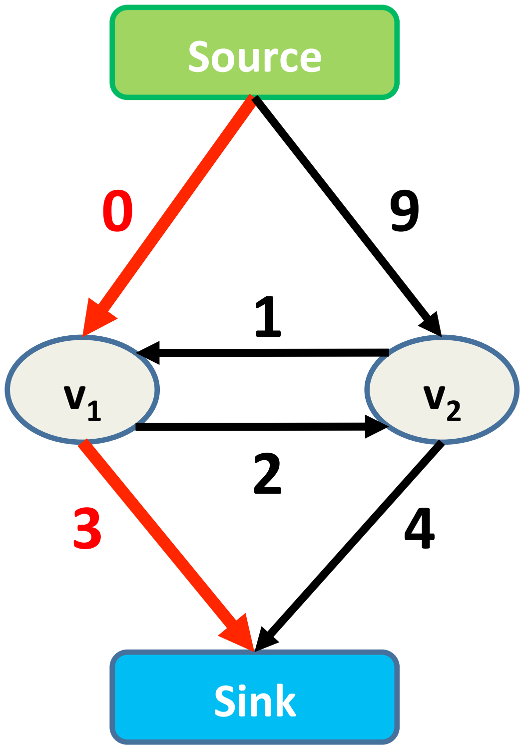
1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path
3. Repeat until no path can be found

Algorithms assume non-negative capacity

Slide courtesy: Pushmeet Kohli

Maxflow Algorithms

Flow = 2



Augmenting Path Based Algorithms

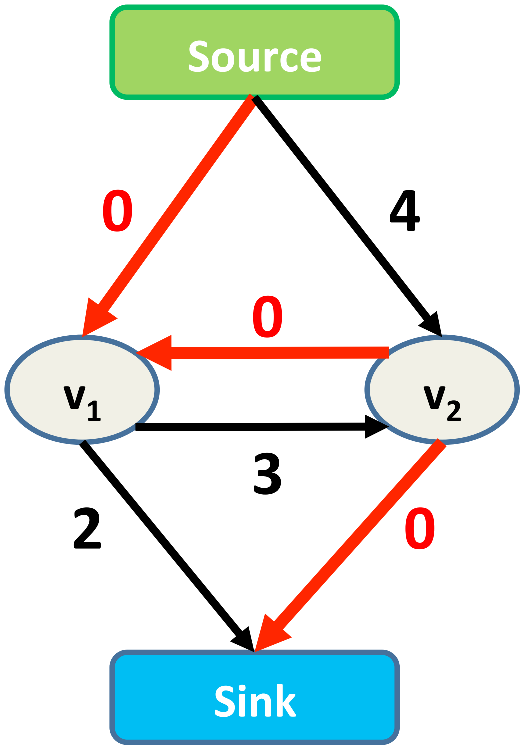
1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path
3. Repeat until no path can be found

Algorithms assume non-negative capacity

Slide courtesy: Pushmeet Kohli

Maxflow Algorithms

Flow = 7



Augmenting Path Based Algorithms

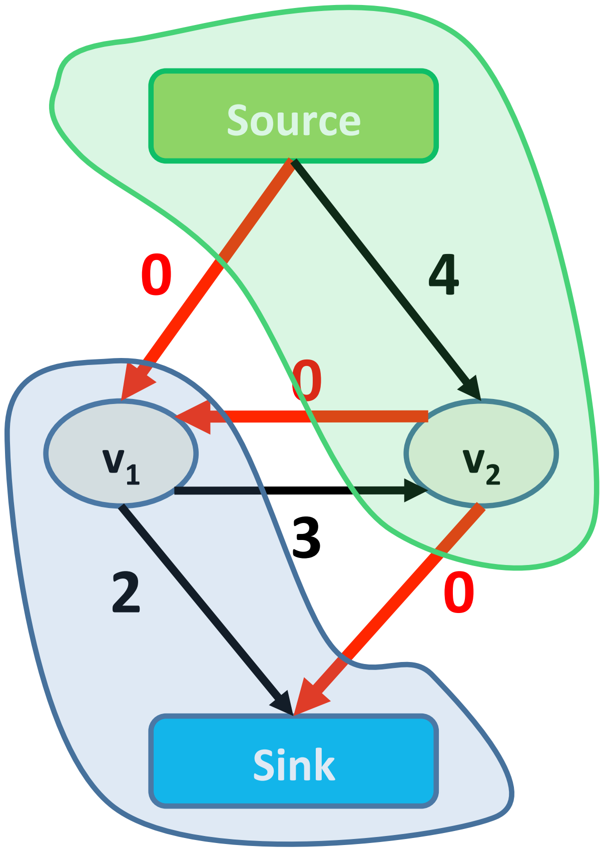
1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path
3. Repeat until no path can be found

Algorithms assume non-negative capacity

Slide courtesy: Pushmeet Kohli

Maxflow Algorithms

Flow = 7



Augmenting Path Based Algorithms

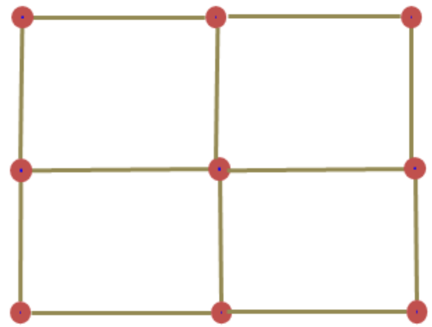
1. Find path from source to sink with positive capacity
2. Push maximum possible flow through this path
3. Repeat until no path can be found

Algorithms assume non-negative capacity

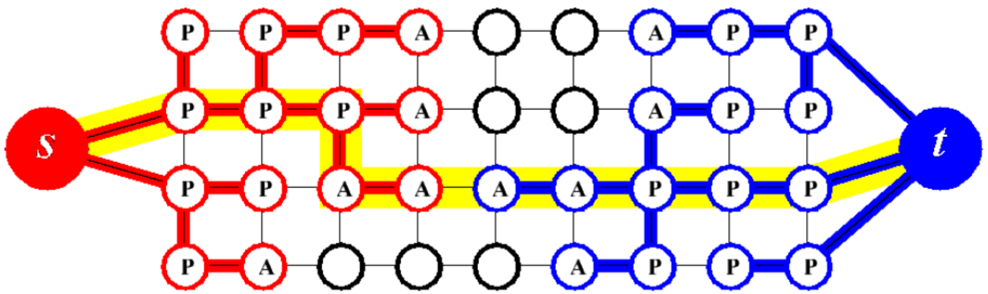
Slide courtesy: Pushmeet Kohli

Maxflow in Computer Vision

- Specialized algorithms for vision problems
  - Grid graphs
  - Low connectivity (m ~ O(n))



- Dual search tree augmenting path algorithm [Boykov & Kolmogorov PAMI'04]



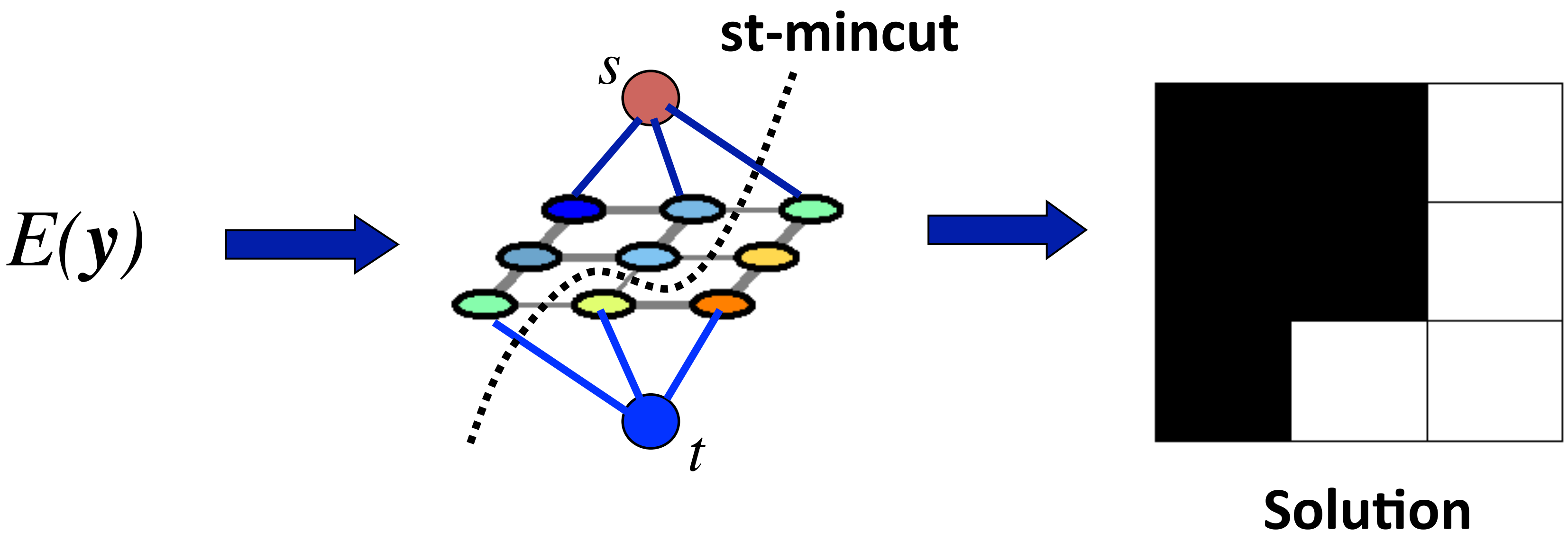
  - Finds approximate shortest augmenting paths efficiently
  - High worst-case time complexity
  - Empirically outperforms other algorithms on vision problems

Slide courtesy: Pushmeet Kohli

st-mincut ⇔ Labelling

Construct a graph such that:

1. Any st-cut corresponds to an assignment of $\mathbf{y}$
2. The cost of the cut is equal to the energy of $\mathbf{y}$: $E(\mathbf{y})$



Slide courtesy: Pushmeet Kohli

St-mincut based Move algorithms

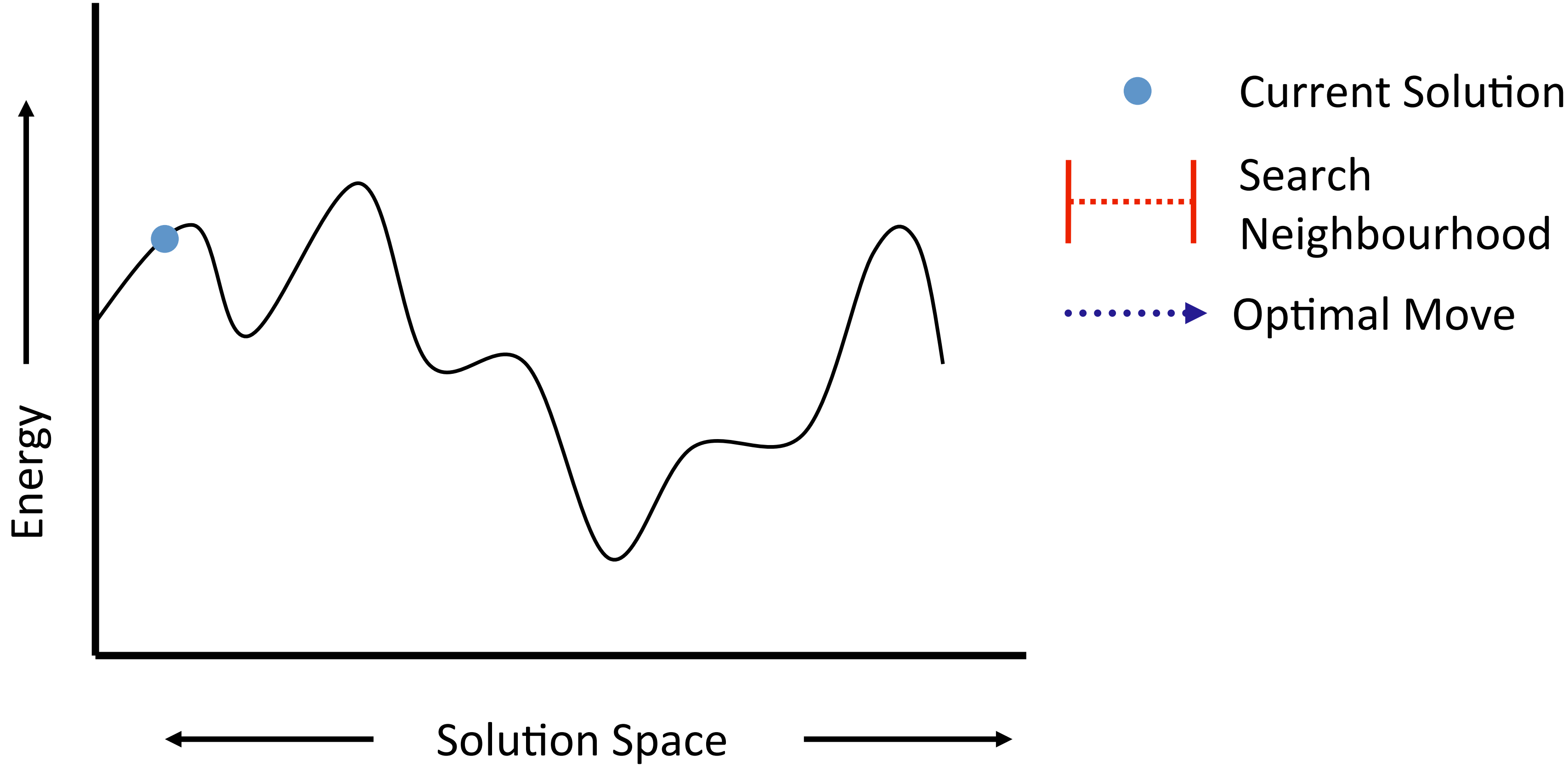
$$E(\mathbf{y}) = \sum_{i \in \mathcal{V}} \psi_i(y_i) + \sum_{(i,j) \in \mathcal{E}} \psi_{ij}(y_i, y_j)$$

y ϵ Labels $\mathcal{L} = \{l_1, l_2, \ldots, l_k\}$

- Commonly used for solving multi-label problems
- Extremely efficient and produce good solutions
- Not Exact: Produce local optima

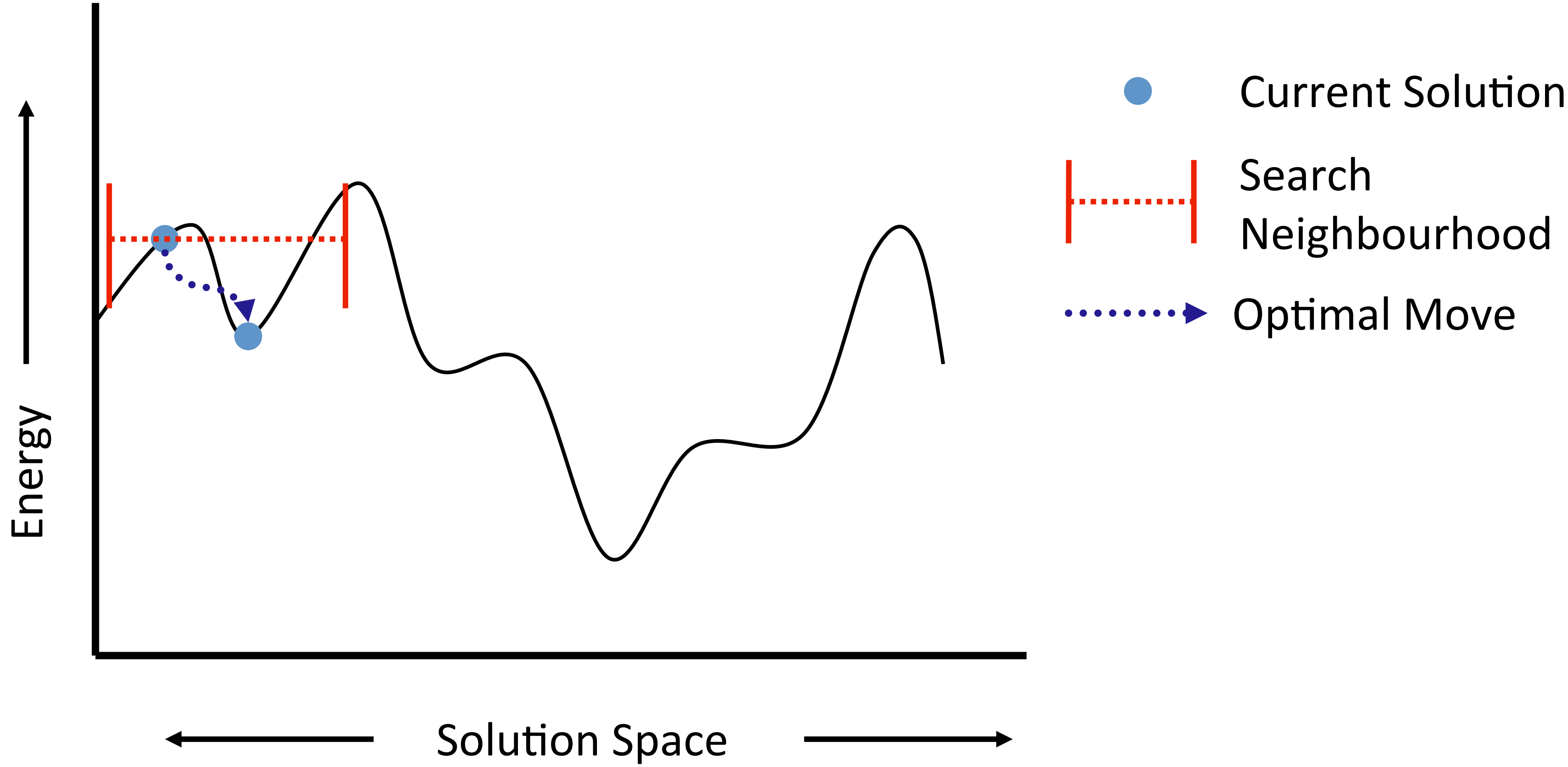
Slide courtesy: Pushmeet Kohli

Move Making Algorithms



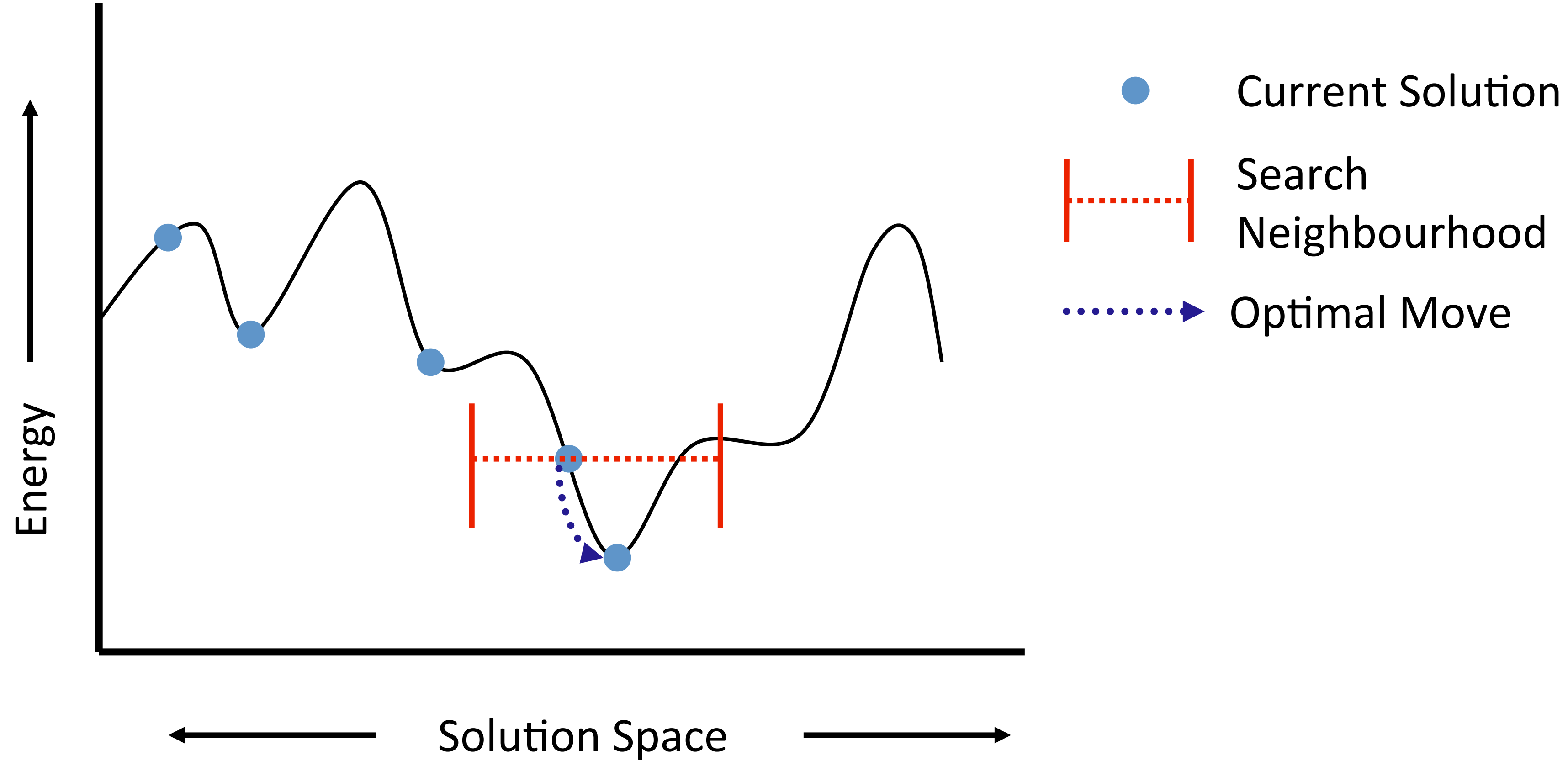
Slide courtesy: Pushmeet Kohli

Move Making Algorithms



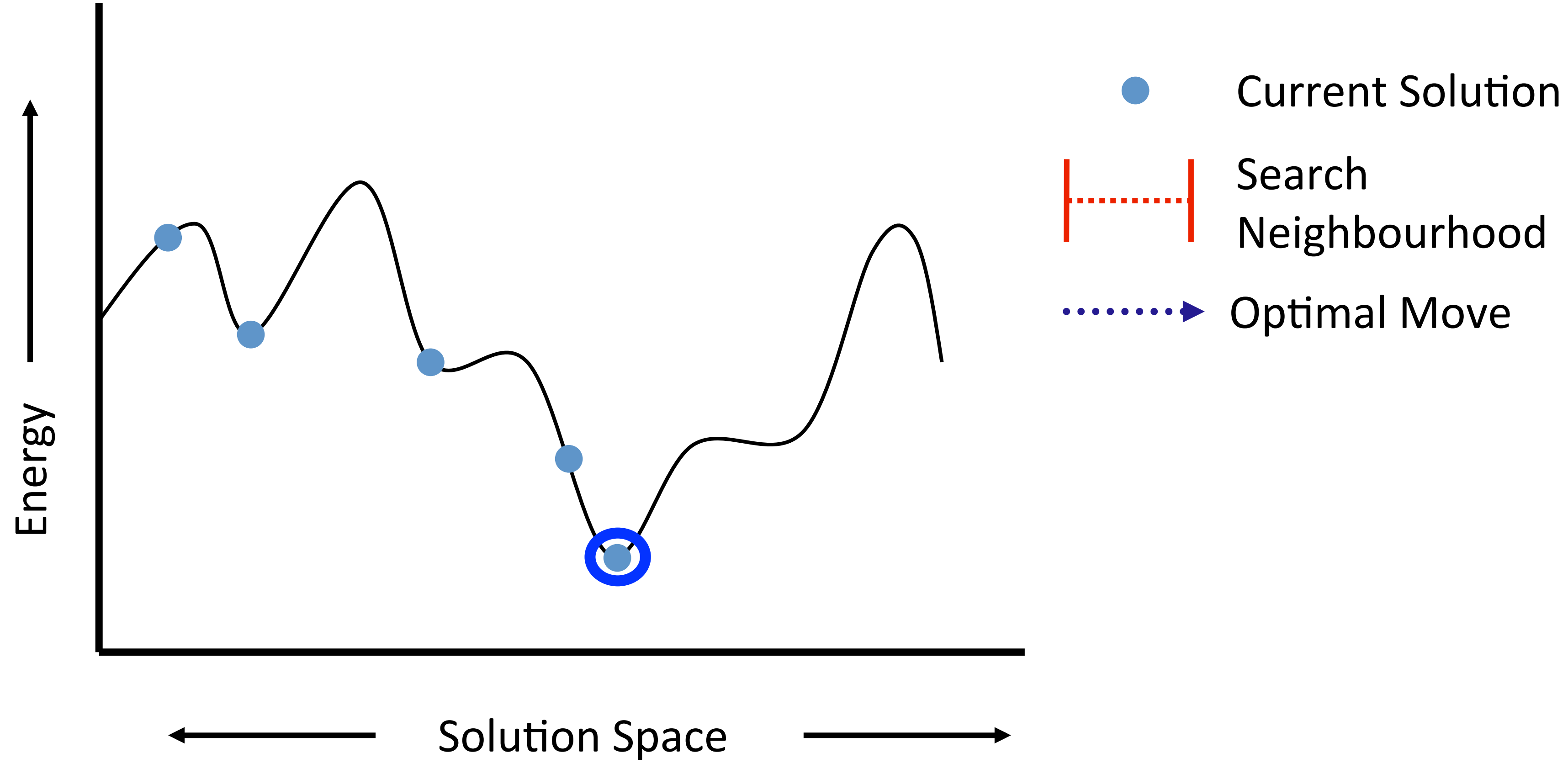
Slide courtesy: Pushmeet Kohli

Move Making Algorithms



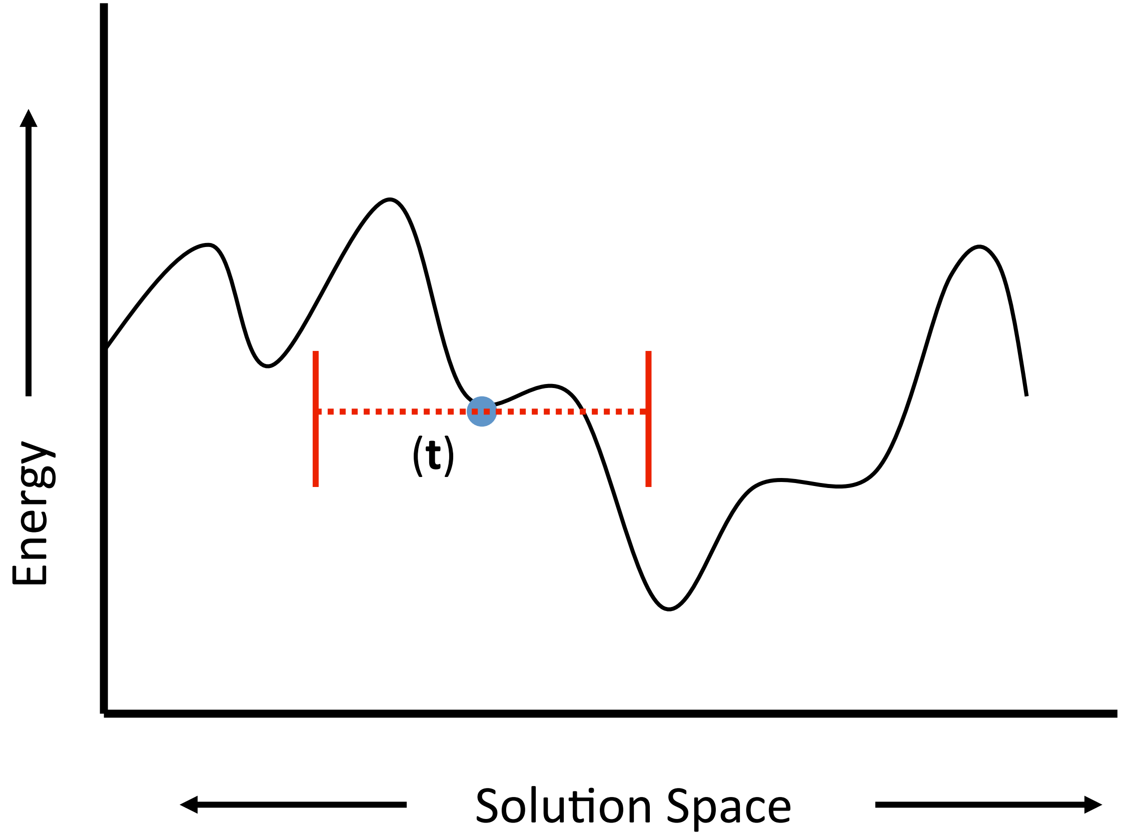
Slide courtesy: Pushmeet Kohli

Move Making Algorithms



Slide courtesy: Pushmeet Kohli

Computing the Optimal Move



Current Solution

Search Neighbourhood

Optimal Move

Key Property

Move Space

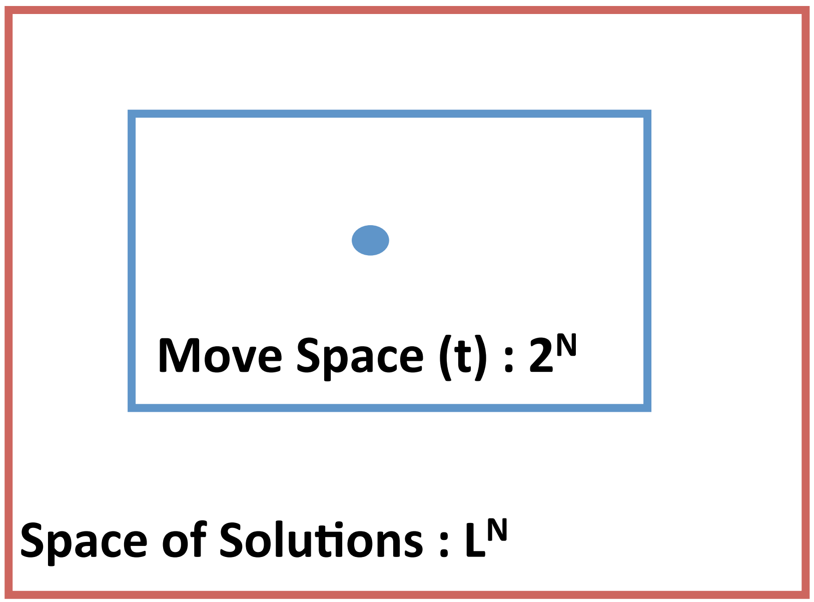
Bigger move space →
- **Better solutions**
- **Finding the optimal move hard**

Slide courtesy: Pushmeet Kohli

Moves using Graph Cuts

Expansion and Swap move algorithms [Boykov et al., PAMI'01]

- Make a series of changes to the solution (moves)
- Each move results in a solution with smaller energy



Current solution

Search neighbourhood

N Number of variables

L Number of labels

How to minimize move functions?

Slide courtesy: Pushmeet Kohli

Expansion Move

- Variables take label α or retain current label

Tree
Ground
House
Sky

Status:

Slide courtesy: Pushmeet Kohli

Boykov, Veksler, Zabih, PAMI'01

Expansion Move

- Variables take label α or retain current label

Tree
Ground
House
Sky

Status: Initialize with Tree

Slide courtesy: Pushmeet Kohli

Boykov, Veksler, Zabih, PAMI'01

Expansion Move

- Variables take label α or retain current label

Tree
Ground
House
Sky

Status: Expand Ground

Slide courtesy: Pushmeet Kohli

Boykov, Veksler, Zabih, PAMI'01

Expansion Move

- Variables take label α or retain current label

Tree
Ground
House
Sky

Status: Expand House

Slide courtesy: Pushmeet Kohli

Boykov, Veksler, Zabih, PAMI'01

Expansion Move

- Variables take label α or retain current label

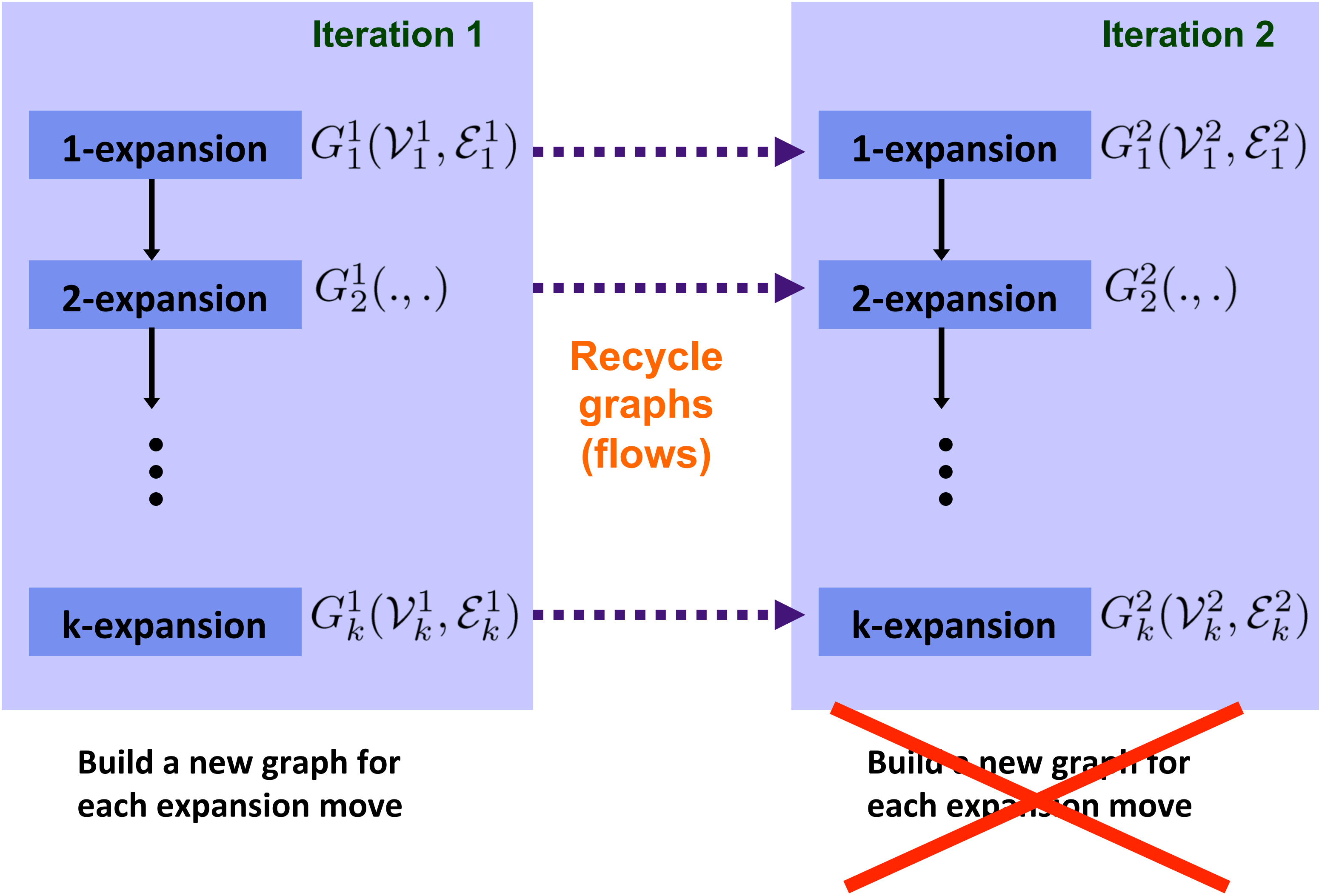
Tree
Ground
House
Sky

Status: Expand Sky

Slide courtesy: Pushmeet Kohli

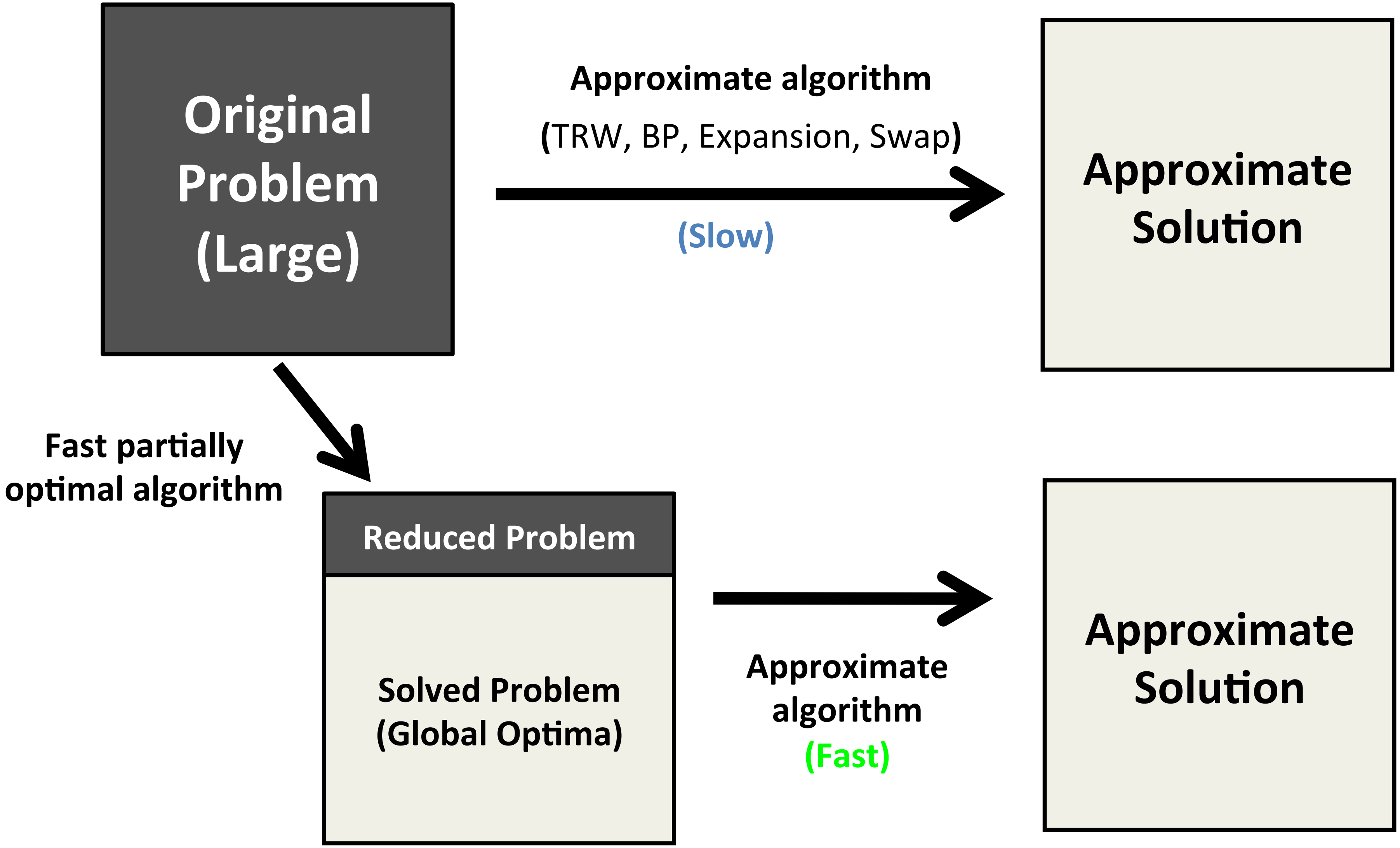
Boykov, Veksler, Zabih, PAMI'01

Efficient Inference: Recycling



Alahari et al., PAMI'10

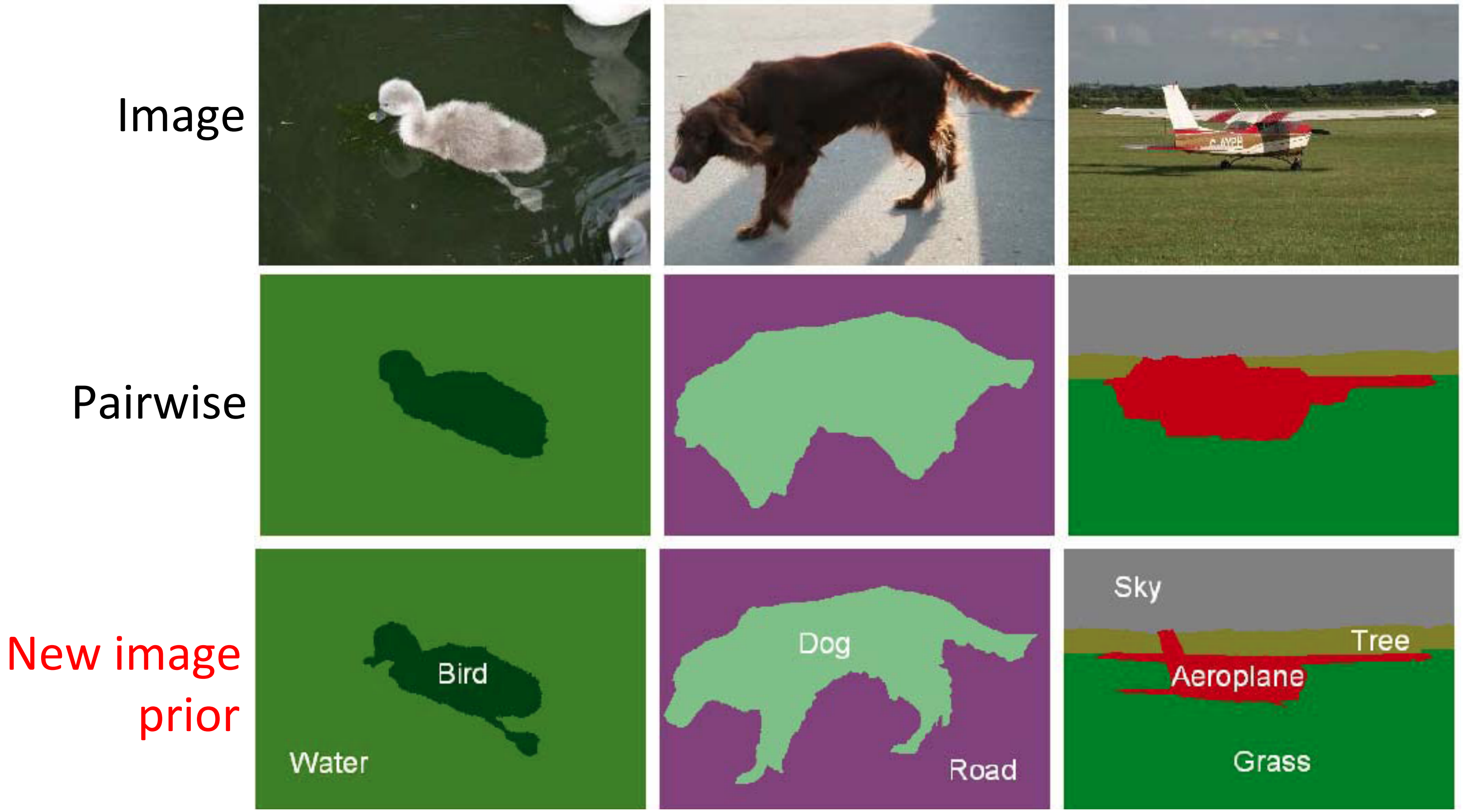
Efficient Inference: Reduction



Alahari et al., PAMI'10

Modelling image priors & inference

- Pairwise energy functions are limited



Kohli et al., IJCV’09, Ladicky et al., ECCV’10

Modelling image priors & inference

- Pairwise energy functions are limited

Image

Pairwise

- Need new energy terms to capture fine contours

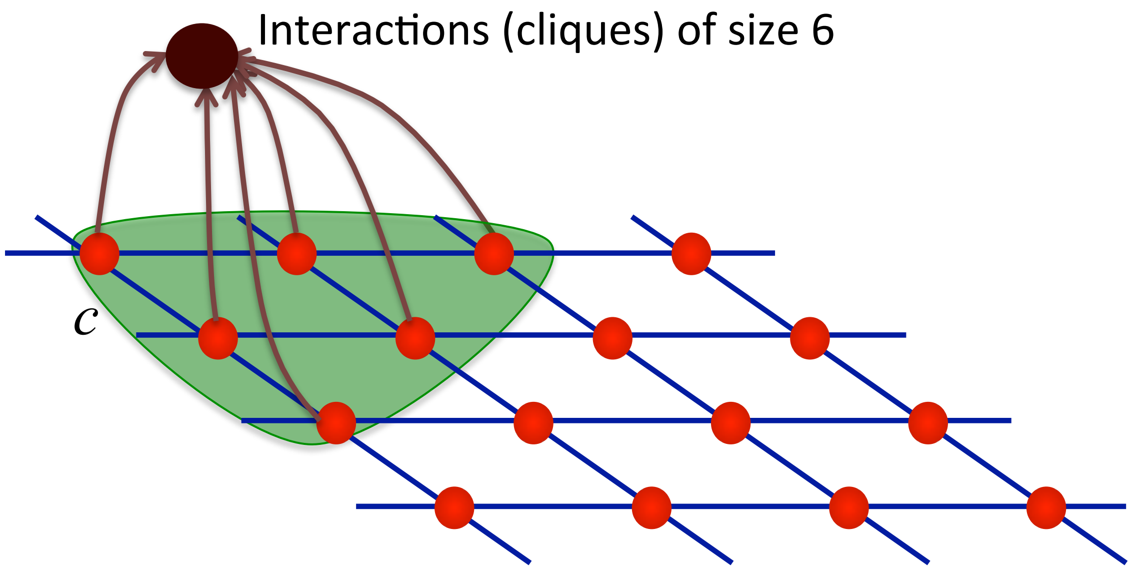
Kohli et al., IJCV’09, Ladicky et al., ECCV’10

Modelling image priors & inference

- New energy terms to capture fine contours

$$E(\mathbf{y}) = \sum_{i \in \mathcal{V}} \psi_i(y_i) + \sum_{(i,j) \in \mathcal{E}} \psi_{ij}(y_i, y_j) + \boxed{\sum_{\mathbf{c} \in \mathcal{S}} \psi_{\mathbf{c}}(y_{\mathbf{c}})}$$

- Higher-order terms, involving several nodes, e.g.,

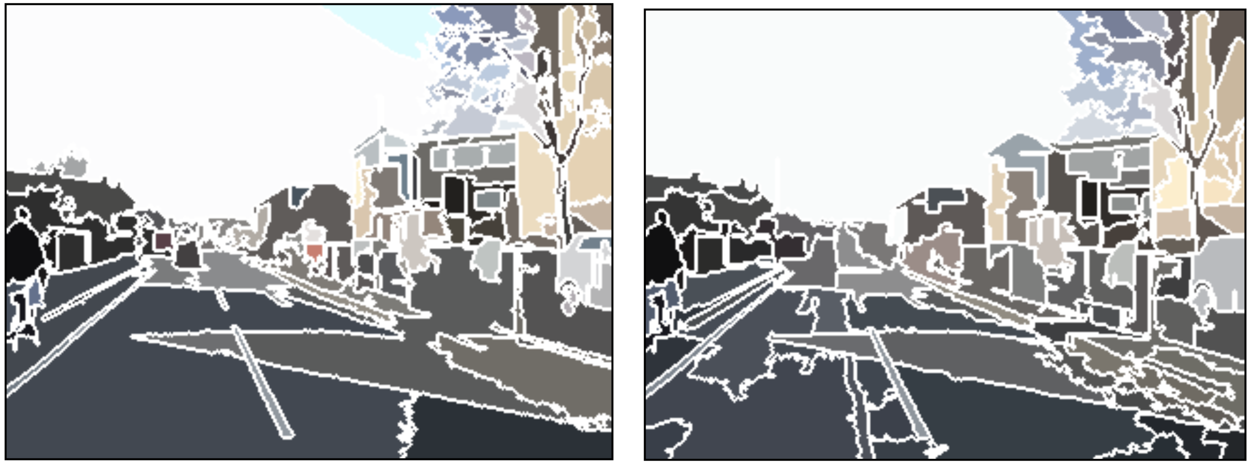


Kohli et al., IJCV'09, Ladicky et al., ECCV'10, Alahari PAMI'10

Modelling image priors & inference

- Higher-order terms, e.g.,
  - grouping "similar" pixels

Input Image

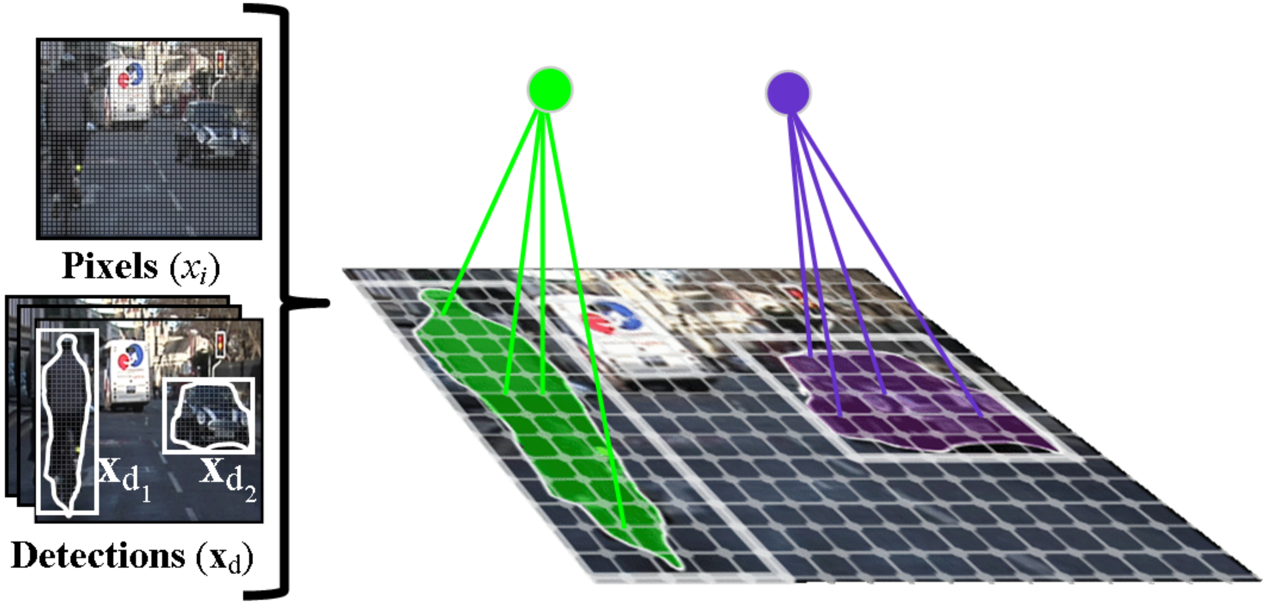


Meanshift segments [Comaniciu & Meer '02]

$$\psi_c(\mathbf{y}_c) = \min\left\{\min_{k\in\mathcal{L}}((|c| - n_k(\mathbf{y}_c))\theta_k + \gamma_k),\ \gamma_{\max}\right\}$$

Modelling image priors & inference

- Higher-order terms, e.g.,
  - Using "detected" objects



$$\psi_h(\mathbf{y}) = \min(\gamma_{max}, \min_l(\gamma_l + k_l \sum_{i \in \mathbf{y}} \delta(y_i \neq l)))$$

Modelling image priors & inference

- Higher-order terms of the form:

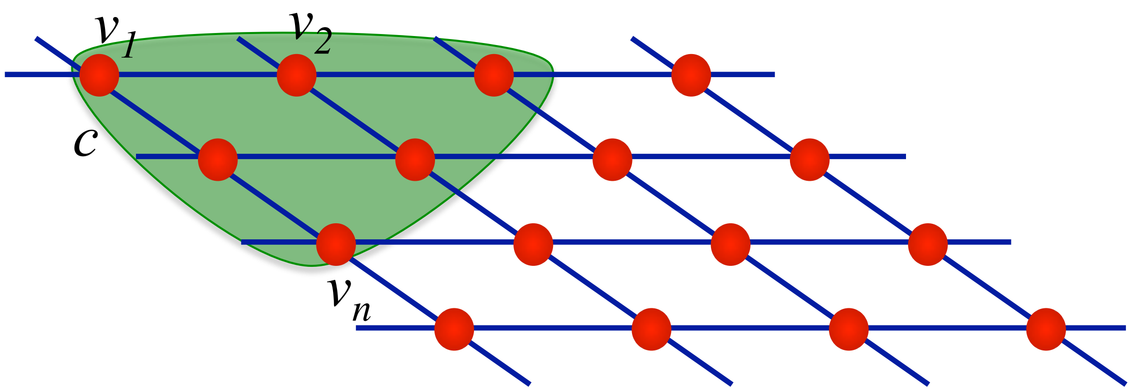
$$\psi_h(\mathbf{y}) = \min(\gamma_{max}, \min_l(\gamma_l + k_l \sum_{i \in \mathbf{y}} \delta(y_i \neq l)))$$

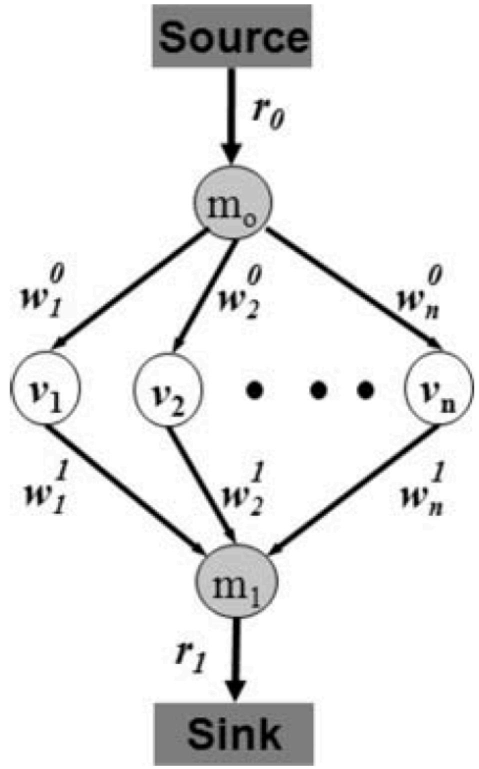
- Can be minimized efficiently with move-making algorithms

Kohli et al., IJCV'09, Ladicky et al., ECCV'10, Alahari PAMI'10

Modelling image priors & inference

- Graph construction with auxiliary nodes





Kohli et al., IJCV'09, Ladicky et al., ECCV'10, Alahari PAMI'10

Modelling image priors & inference

- Evaluation: CamVid dataset [Brostow et al. '08]

	Building	Tree	Sky	Car	Sign-Symbol	Road	Pedestrian	Fence	Column-Pole	Sidewalk	Bicyclist	**Average**
Recall[6]												
Brostow et al. '08	46.2	61.9	89.7	68.6	42.9	89.5	**53.6**	46.6	0.7	60.5	22.5	53.0
Sturgess et al. '09	**84.5**	72.6	**97.5**	72.7	34.1	**95.3**	34.2	45.7	8.1	77.6	28.5	59.2
Without detectors	79.3	76.0	96.2	74.6	**43.2**	94.0	40.4	47.0	**14.6**	81.2	31.1	61.6
Our method	81.5	**76.6**	96.2	**78.7**	40.2	93.9	43.0	**47.6**	14.3	**81.5**	**33.9**	**62.5**
Intersection *vs* Union[7]												
Sturgess et al. '09	**71.6**	60.4	**89.5**	58.3	19.4	86.6	**26.1**	35.0	7.2	63.8	22.6	49.2
Without detectors	70.0	**63.7**	**89.5**	58.9	17.1	86.3	20.0	**35.8**	9.2	**64.6**	23.1	48.9
Our method	71.5	**63.7**	89.4	**64.8**	**19.8**	**86.8**	23.7	35.6	**9.3**	**64.6**	**26.5**	**50.5**

Ladicky et al., ECCV'10

So far …

- In the context of labelling problems
  - Learning parameters of the function
  - Modelling image priors & inference

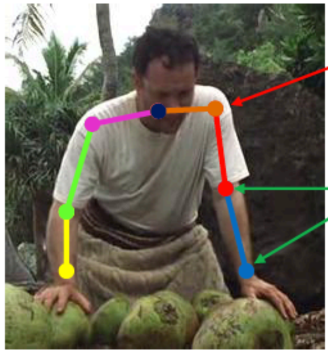
  Next:
  - Modelling temporal constraints & inference

Human Pose Estimation

Poses in the Wild dataset, CVPR 2014

Human Pose Estimation (in an image)

- Formulated as a graph optimization problem



ϕ_u : unary potential

$\psi_{u,v}$: pairwise potential

For an image *I*, pose model $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, and

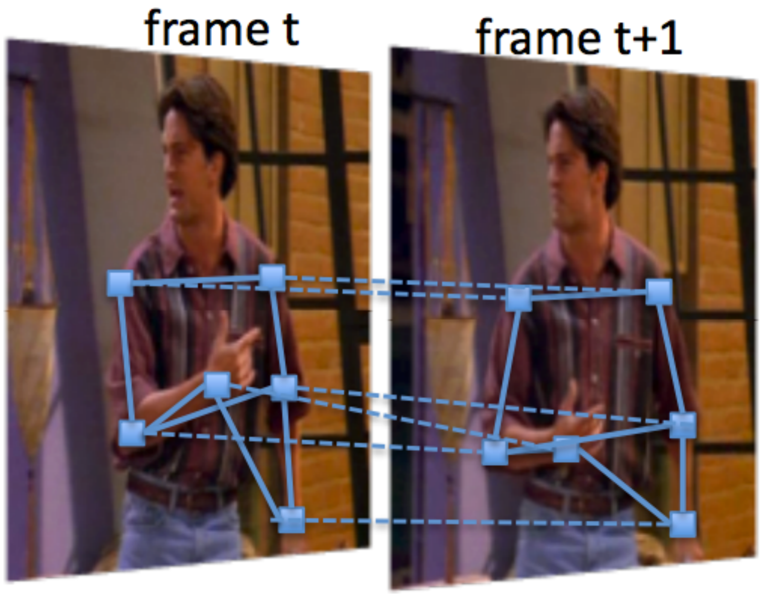
$$p = \{p^u = (x^u, y^u) \in \mathbb{R}^2 : \forall u \in \mathcal{V}\}$$

$$\min C(I, p) := \sum_{u \in \mathcal{V}} \phi_u(I, p^u) + \sum_{(u,v) \in \mathcal{E}} \psi_{u,v}(p^u - p^v)$$

Yang and Ramanan, CVPR 2011

Human Pose Estimation

- Extension to videos: introduce temporal links
- Inference is now computationally expensive – requires approximate methods



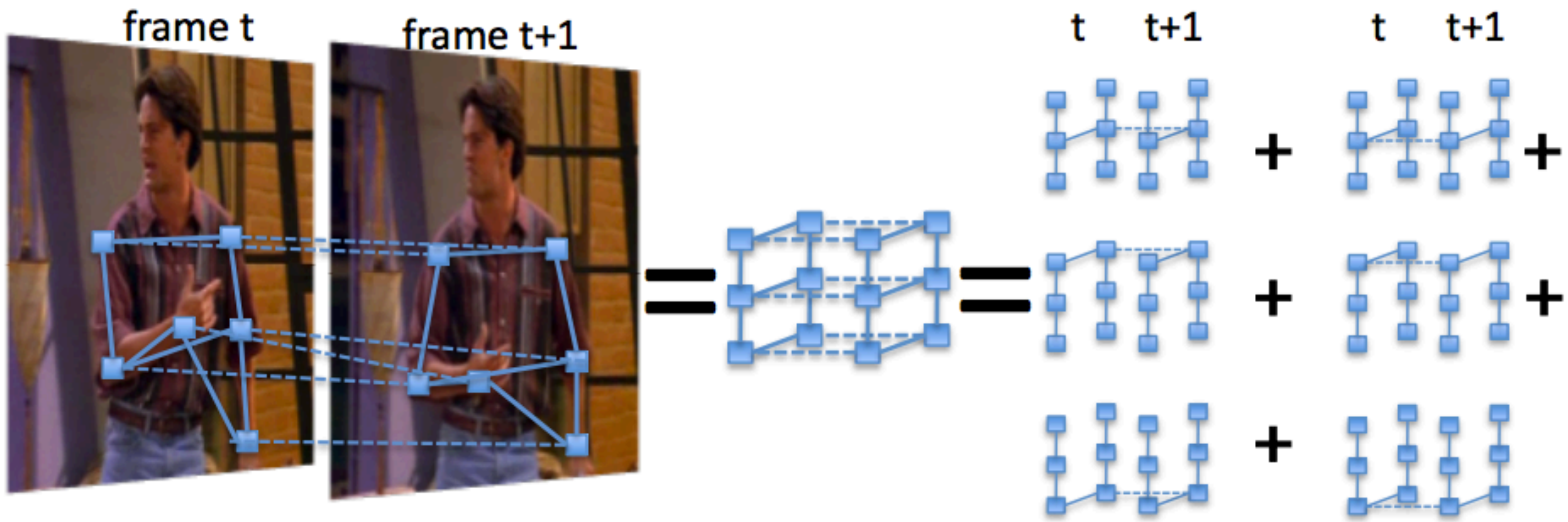
e.g., Sapp et al., ’11, Tokola et al., ’13

Human Pose Estimation

- Extension to videos: introduce temporal links
- Inference is now computationally expensive – requires approximate methods
- e.g.,
  – Change graph structure [Sapp et al. '11, Weiss et al. '11]
  – Use approximate inference [Ferrari et al. '08, Wang et al. '08, Park & Ramanan '11, Tokola et al. '13]

Human Pose Estimation (Video)

- Approximate the graph as combination of trees

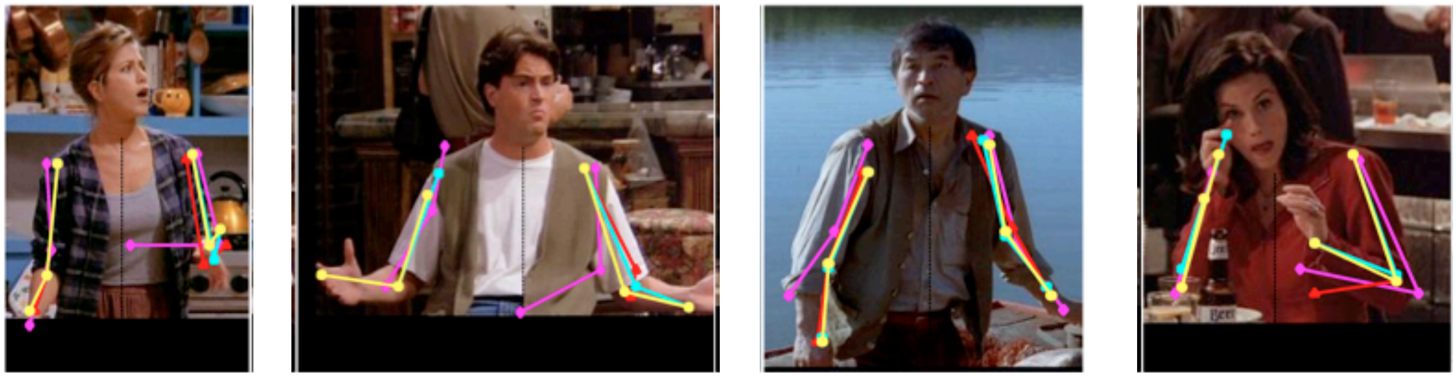


- Several useful interactions are discarded

Sapp, Weiss, Taskar, CVPR 2011

Human Pose Estimation (Video)

- Compute a candidate set of poses in each frame
- Then, track (entire pose or pose-parts) over time



- Limited by the no. of candidates or regularization

Park and Ramanan, ICCV 2011; Ramakrishna, Kanade, Sheikh, CVPR 2013

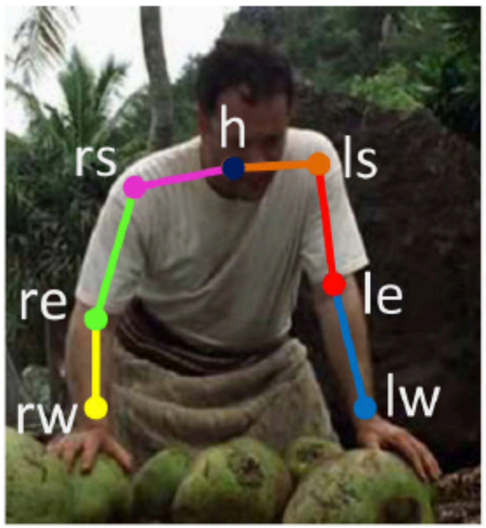
Our Pose Estimation Approach

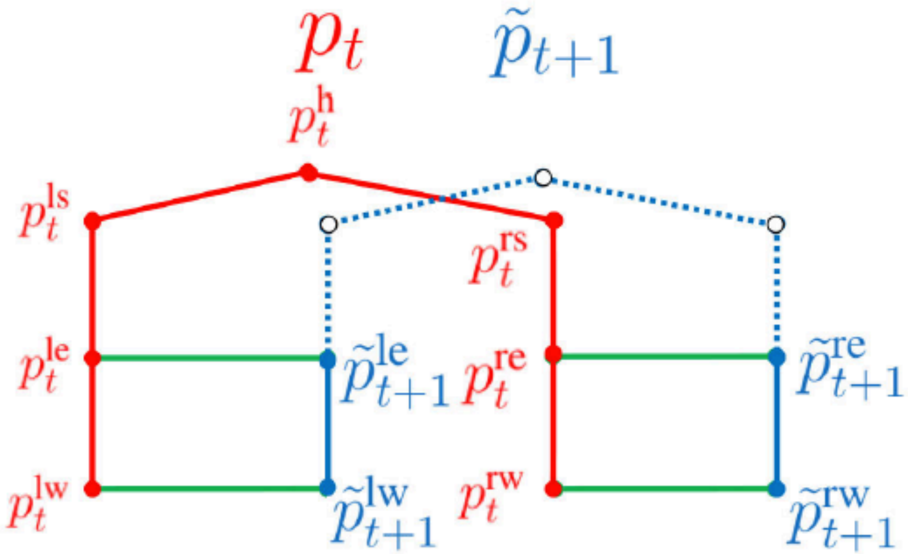
- Combines
  1. Candidate pose set
     - Generate better candidates
  2. Decomposition strategy
     - Generate limb sequences and recompose the pose

Cherian, Mairal, Alahari, Schmid, CVPR 2014

Better Candidate Poses

- Stabilize the lower-limb pose estimates

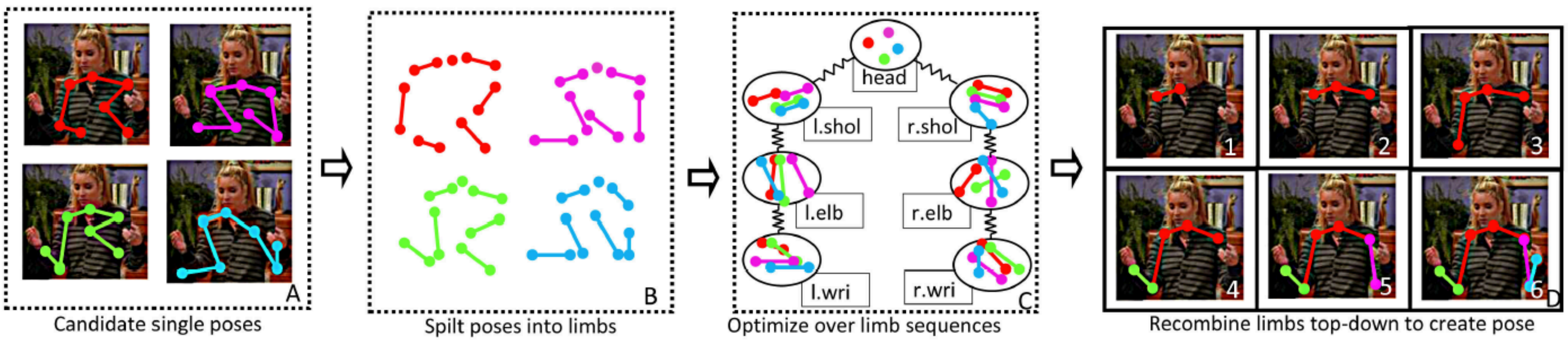




$$C(I_t, p_t) + \tilde{C}(I_{t+1}, \tilde{p}_{t+1}) + \tilde{\lambda}_1 \sum_{u \in \mathcal{W}} \|\tilde{p}_{t+1}^u - p_t^u - f_t(p_t^u)\|_2^2$$

De/Re- composition

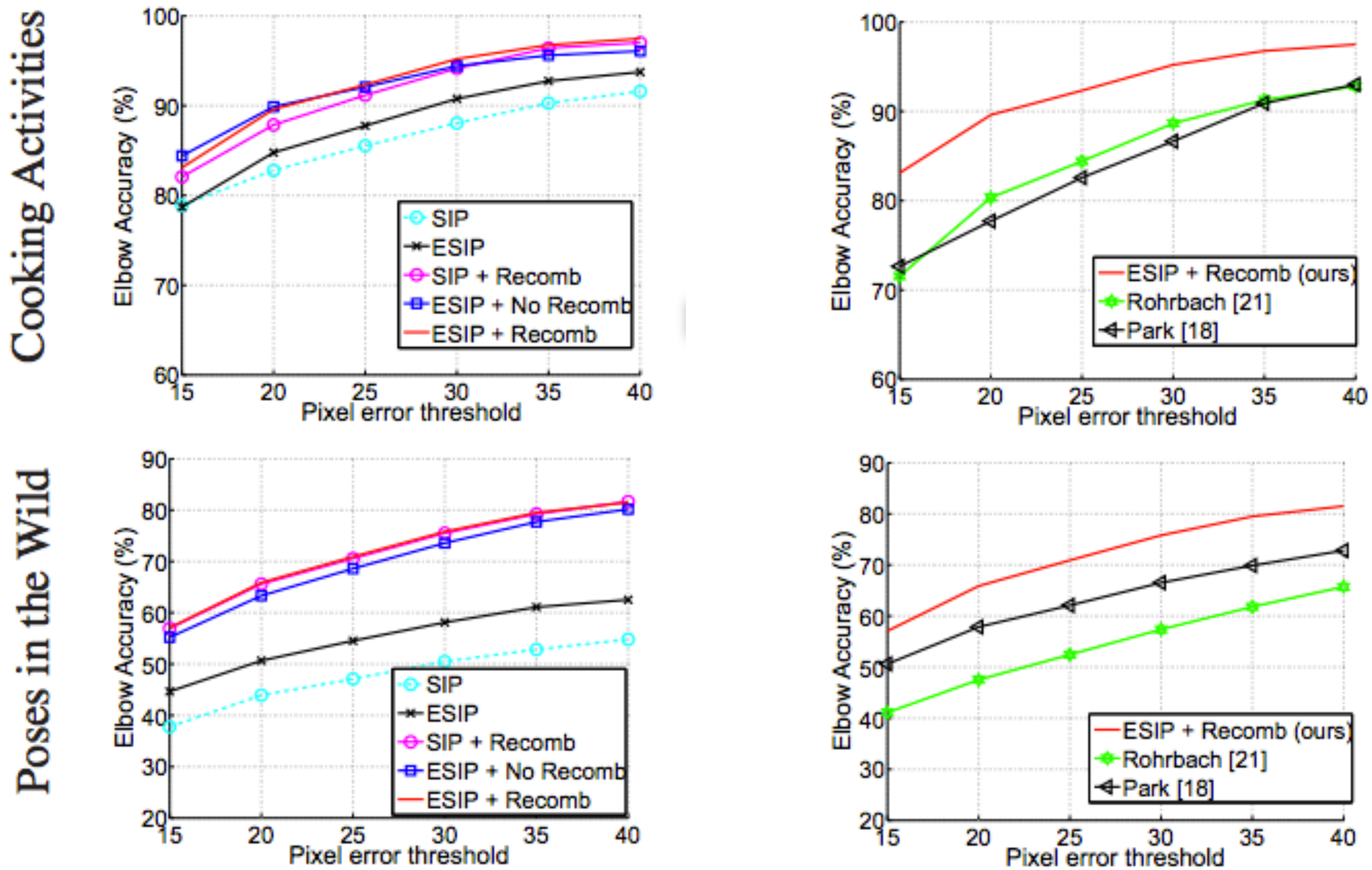
- Decompose poses and perform limb-tracking



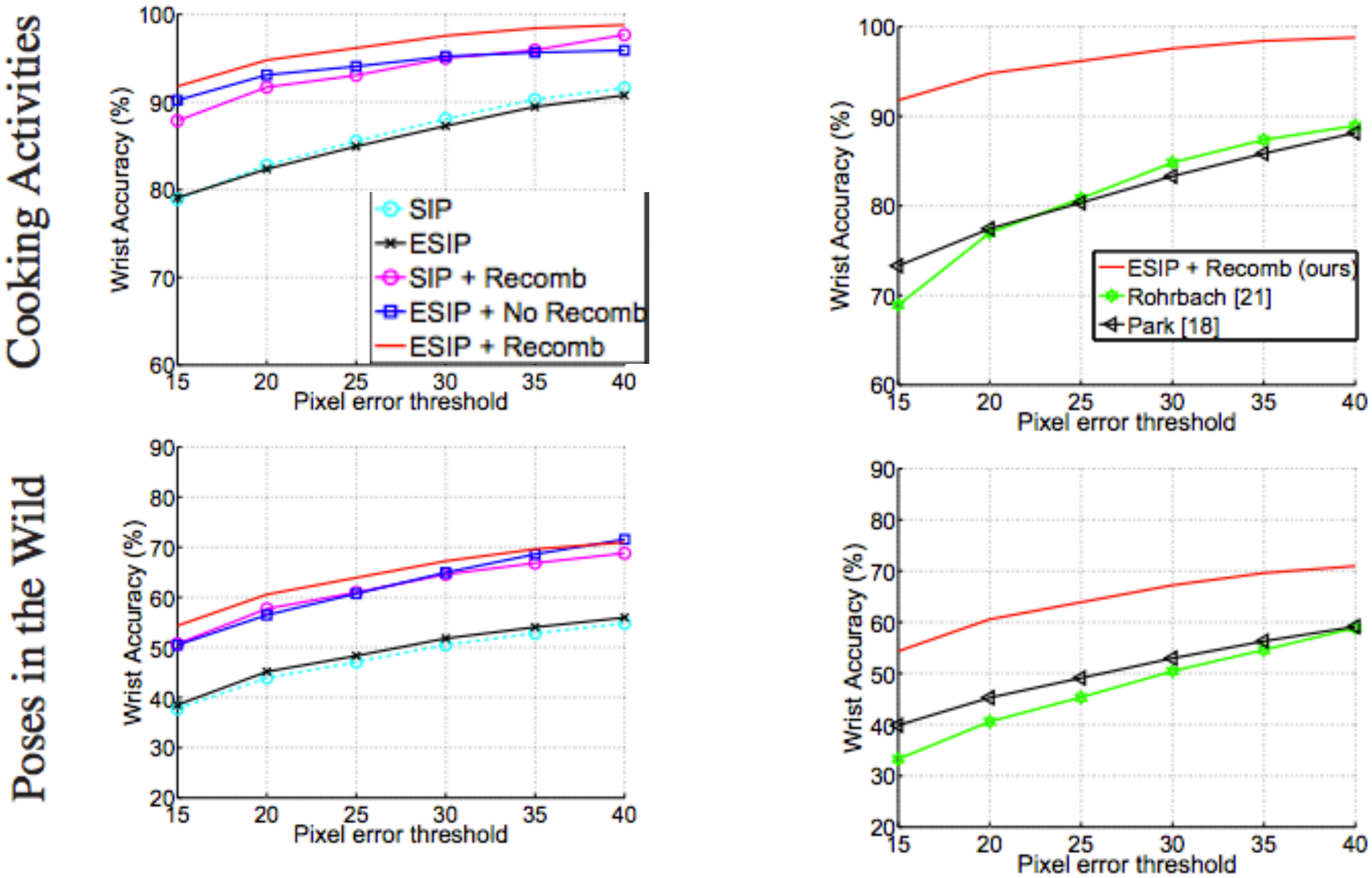
Evaluation: Poses in the Wild Dataset

Available at: http://lear.inrialpes.fr/research/posesinthewild/

Human Pose Estimation: Elbows

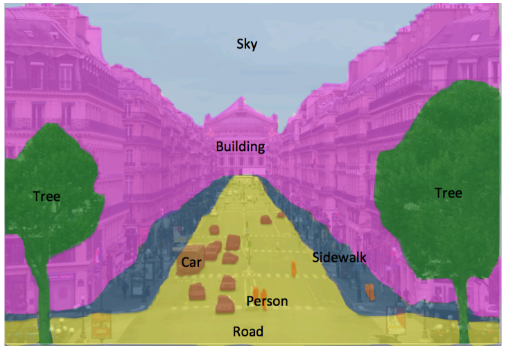


Human Pose Estimation: Wrists

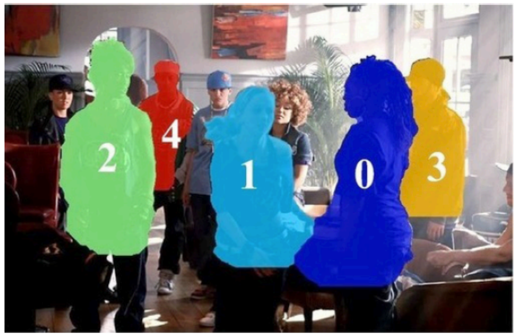


Summary

- In the context of labelling problems



Semantic segmentation



Person recognition

Pose estimation

  - Learning parameters of the function
  - Modelling image priors & inference
  - Modelling temporal constraints & inference

Code + Datasets at: http://lear.inrialpes.fr/~alahari