Probabilistic Graphical Models

CVFX

2015.04.23

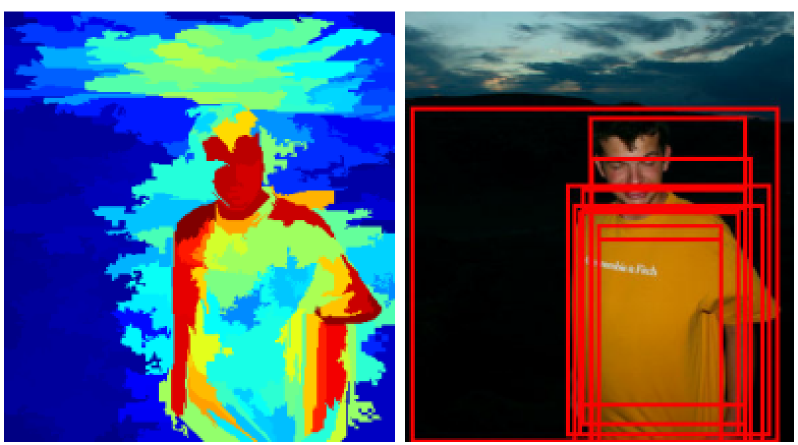
- 內容
  - representation
  - inference
  - learning

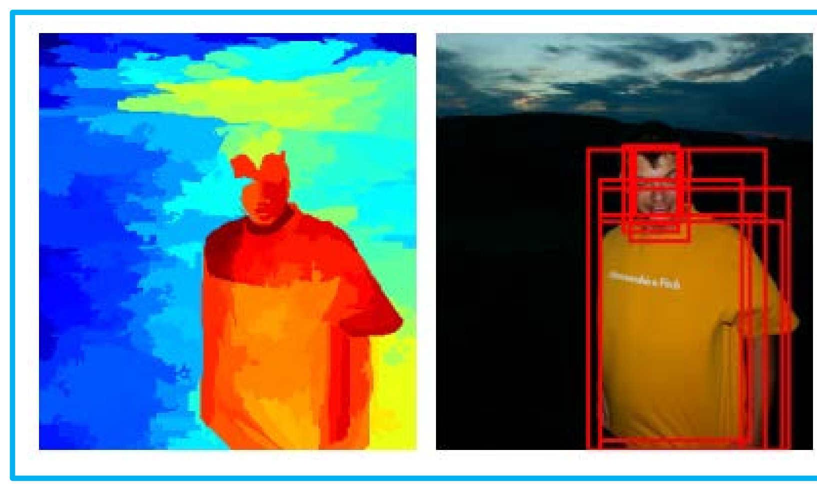
- 實例
  - 電腦視覺、影像處理

實例一

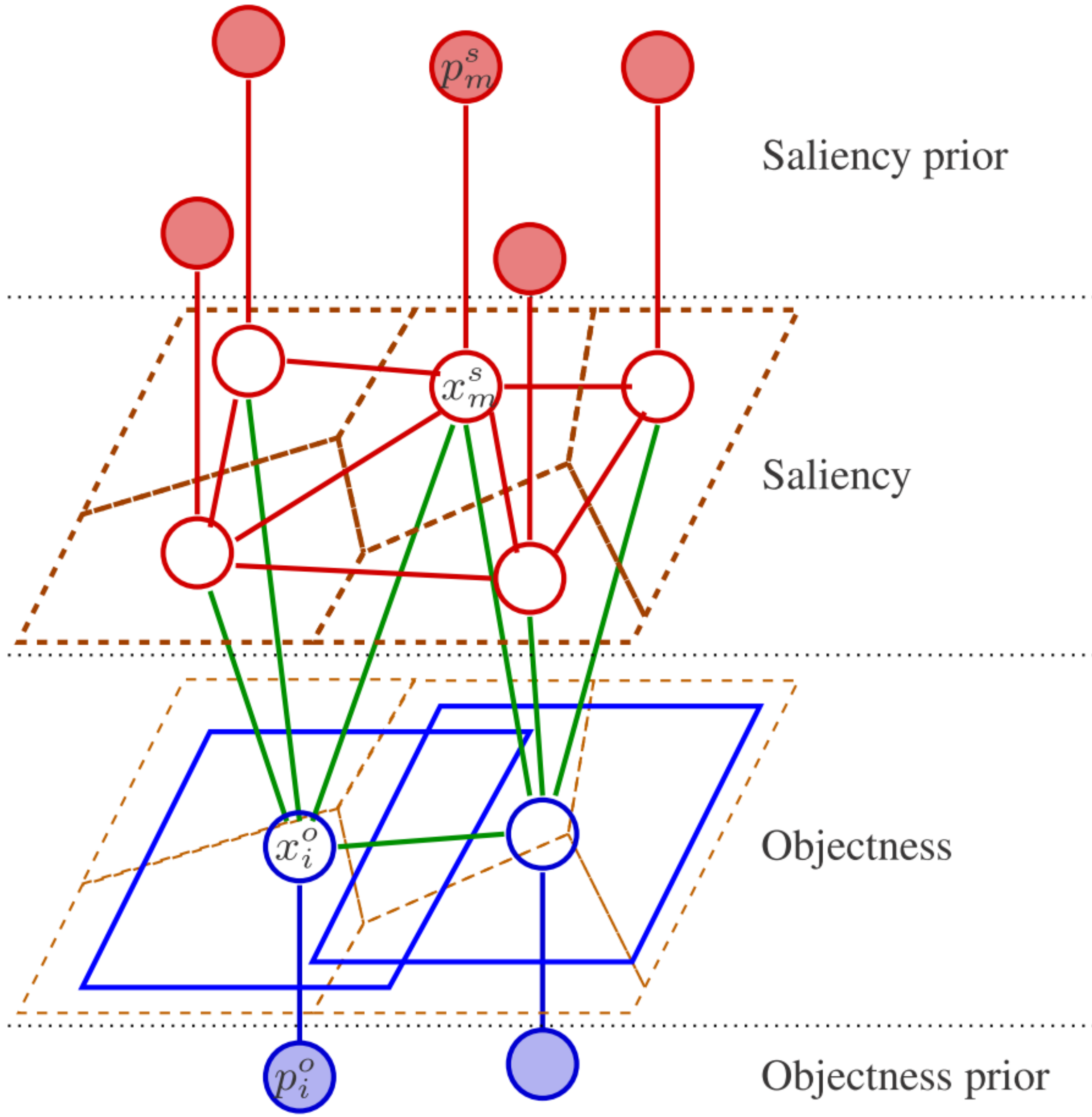
- *Fusing Generic Objectness and Visual Saliency for Salient Object Detection*
  - Chang et al., ICCV 2011

分別計算的 saliency 和 objectness





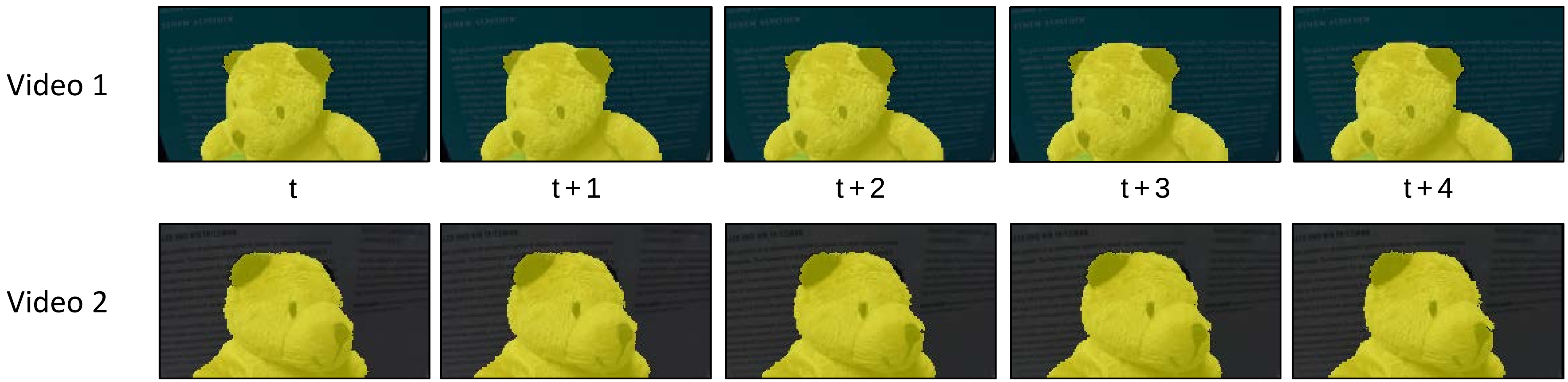
$$F(\mathbf{x}^s, \mathbf{x}^o) = F_s(\mathbf{x}^s) + F_o(\mathbf{x}^o) + \Delta(\mathbf{x}^s, \mathbf{x}^o)$$



實例二

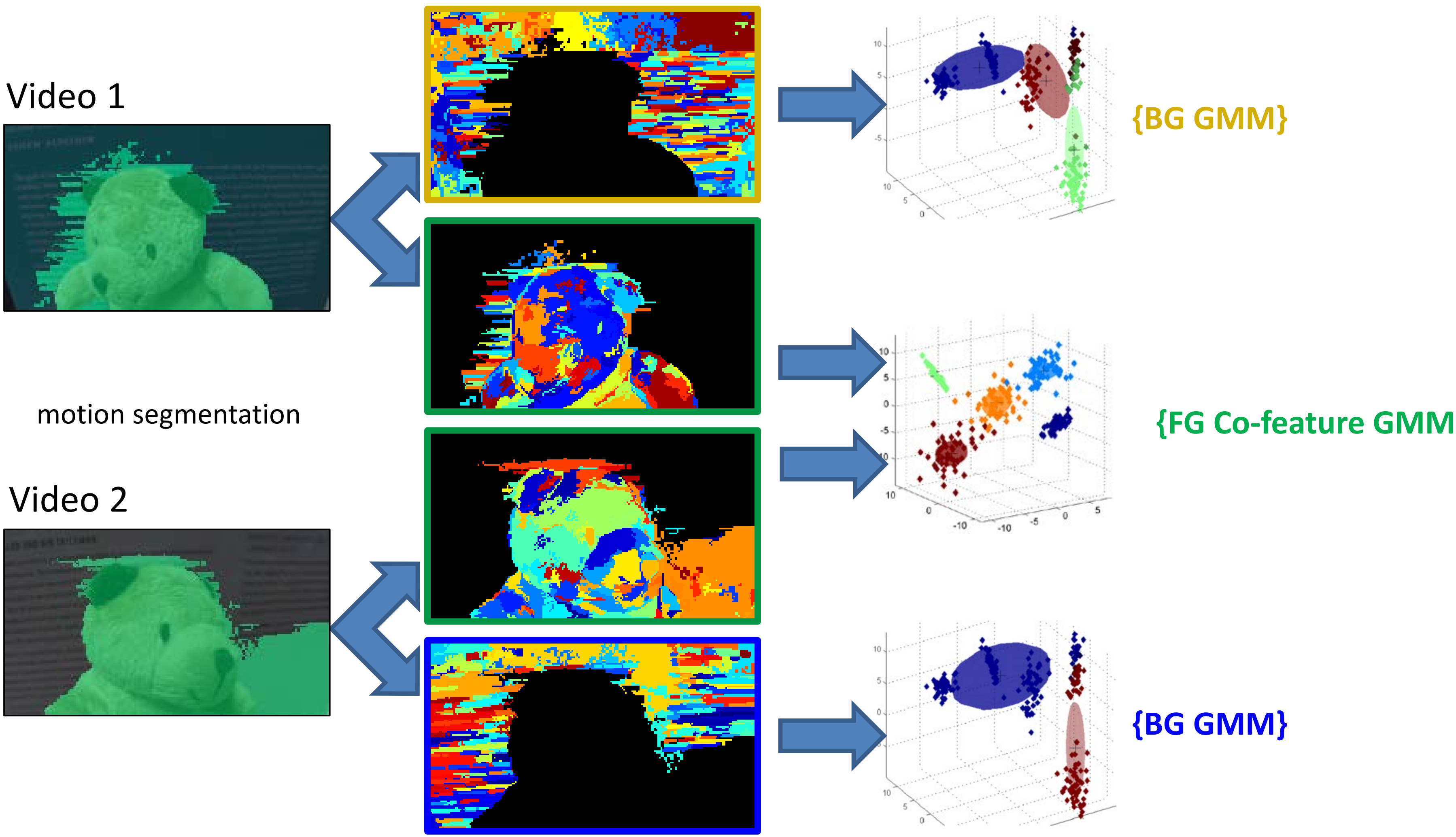
- *Video Object Co-segmentation*
  - Chen et al., ACM Multimedia 2012

- Segmenting out the regions of the common object in two videos



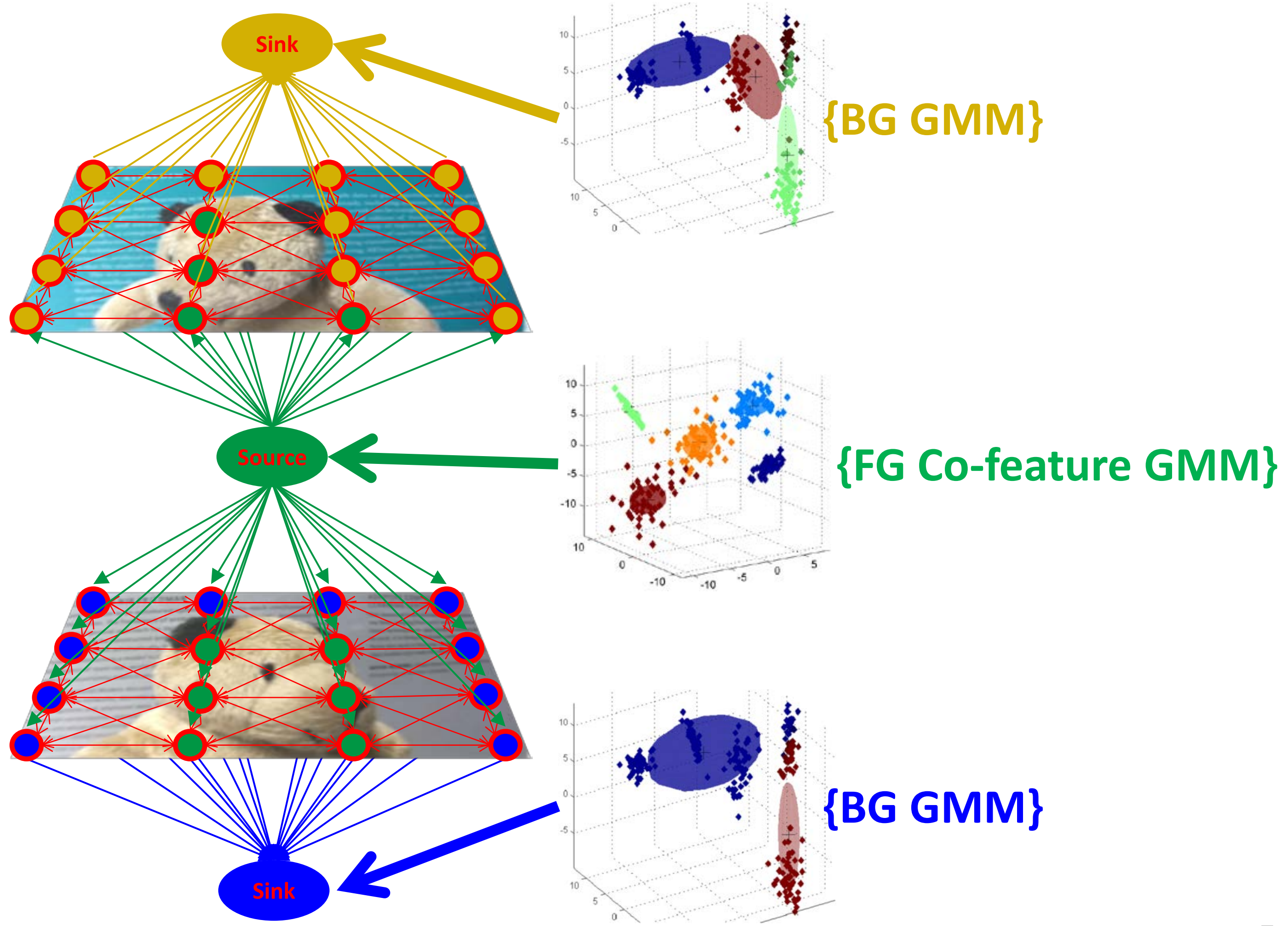
Graph cut$_{1/3}$

GMM from motion segmentation



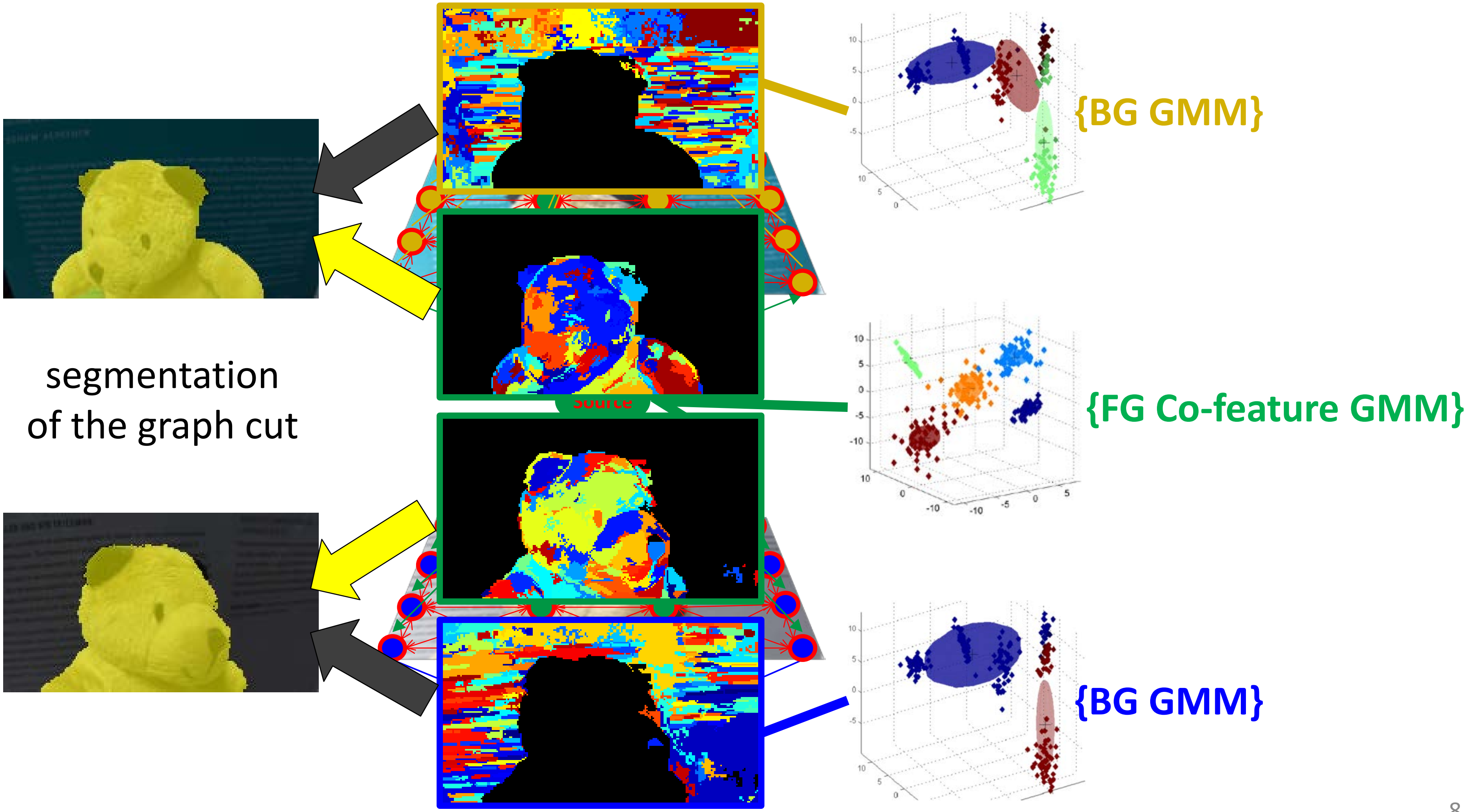
Graph cut$_{2/3}$

Graph cuts w.r.t. GMM



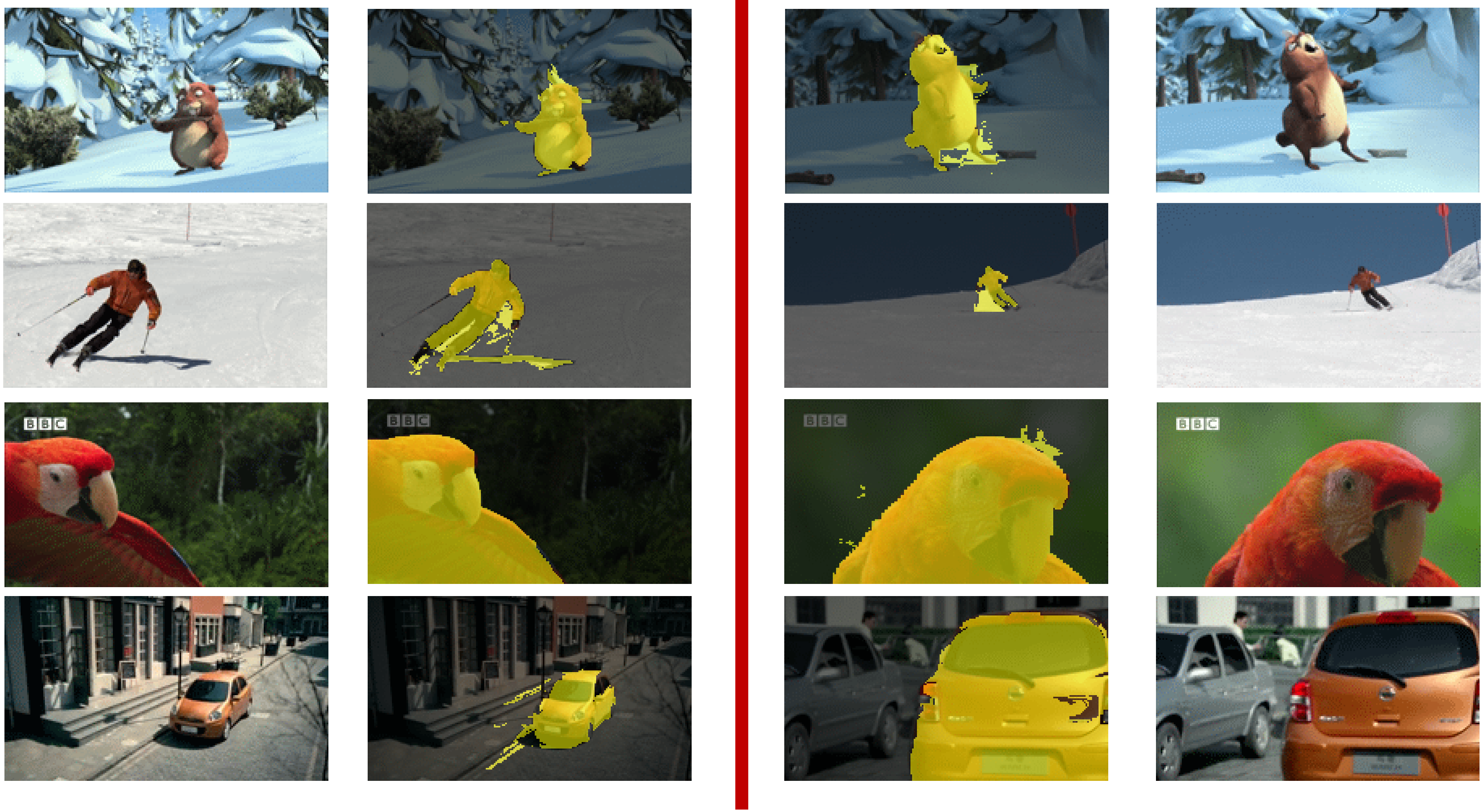
Graph cut$_{3/3}$

Derive the segmentation from the graph cuts



Video object cosegmentation

Video 1 | Cosegmentation | Cosegmentation | Video 2



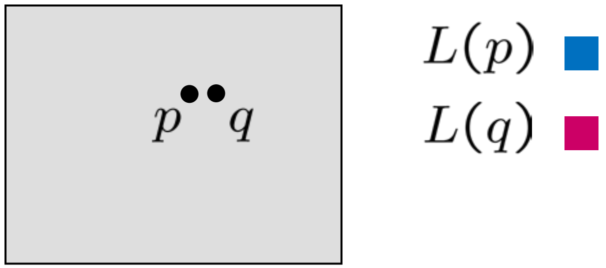
實例三

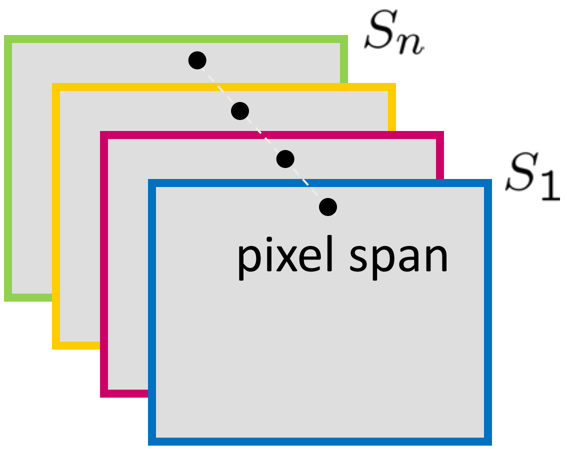
- *Interactive digital photomontage*
  - Agarwala et al., SIGGRAPH 2004

Graph-cut optimization

Suppose we have n source images $S_1, \ldots, S_n$. To form a composite, we need to choose a source image S_i for each pixel p. We call the mapping between pixels and source images a *labeling* and denote the label for each pixel $L(p)$. We say that a seam exists between two neighboring pixels p, q in the composite if $L(p) \neq L(q)$.

$$C(L) = \sum_p C_d(p, L(p)) + \sum_{p,q} C_i(p, q, L(p), L(q))$$





Data terms

Designated color (most or least similar): Euclidean distance in RGB space of the source image pixel $S_{L(p)}(p)$ from a user-specified target color in the pixel span.

Minimum (maximum) luminance: distance in luminance from the minimum (maximum) luminance pixel in a pixel span.

Minimum (maximum) likelihood: the probability (or 1-probability) of the color at $S_{L(p)}(p)$, given a distribution formed from the color histogram of all pixels in the span.

Minimum (maximum) difference: the Euclidean distance in RGB space of the source image pixel $S_{L(p)}(p)$ from a user-specified image pixel $S_u(p)$, where S_u is a user-specified source image.

Designated image : 0 if $L(p) = u$, where S_u is a user-specified source image, and a large penalty otherwise.

Contrast : a measure created by center-surround convolution.

Smoothness terms

If $L(p) = L(q)$, the smoothness term is 0. Otherwise,

$$C_i(p, q, L(p), L(q)) = \begin{cases} X & \text{if matching "colors"} \\ Y & \text{if matching "gradients"} \\ X + Y & \text{if matching "colors \& gradients"} \\ X/Z & \text{if matching "colors \& edges"} \end{cases}$$

$$\begin{aligned} X &= \|S_{L(p)}(p) - S_{L(q)}(p)\| + \|S_{L(p)}(q) - S_{L(q)}(q)\| \\ Y &= \|\nabla S_{L(p)}(p) - \nabla S_{L(q)}(p)\| + \|\nabla S_{L(p)}(q) - \nabla S_{L(q)}(q)\| \\ Z &= E_{L(p)}(p,q) + E_{L(q)}(p,q)) \end{aligned}$$

$\nabla S_z(p)$ is a 6-component color gradient of image z at pixel p.

$E_z(p,q)$ is the scalar edge potential between pixels p and q.

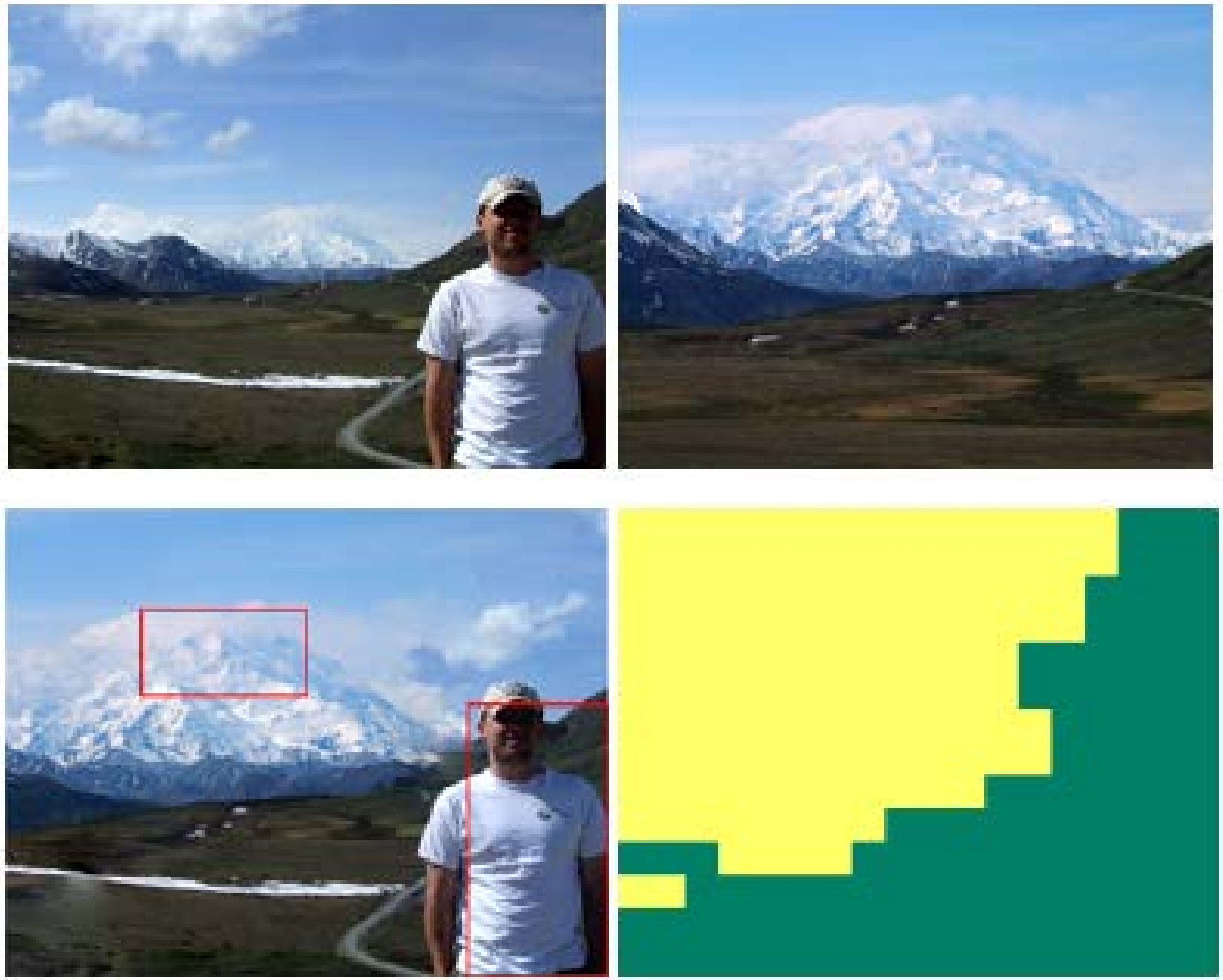
Photomontage: composite portraits

graph cuts + gradient domain blending

實例四

- *The Patch Transform and Its Applications to Image Editing*
  - Cho *et al.*, CVPR 2008

Adding two images in the patch domain



Patch transform

- the user may modify the patch statistics of the original image or constrain some patch positions

- the goal is to piece together the patches to form a plausible image

- define a probability for all possible combinations of patches and try to find a combination that yields the highest probability

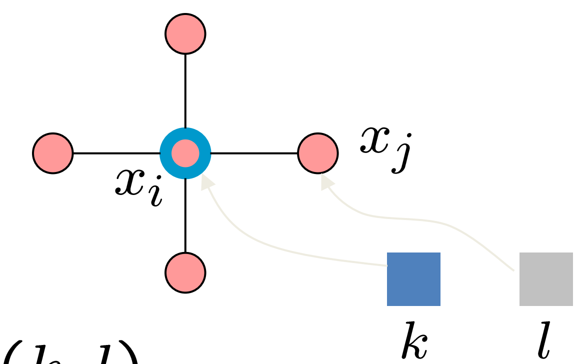
What makes a good placement of patches?

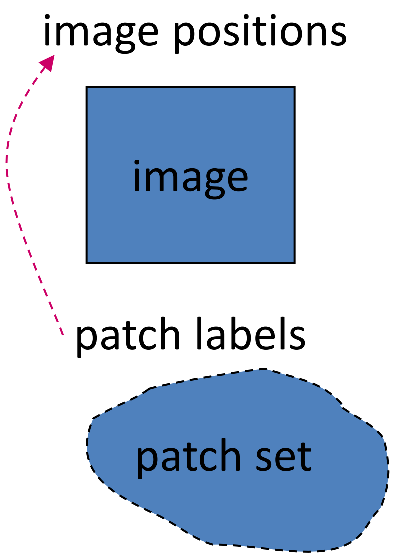
1. Adjacent patches should all plausibly fit next to each other
   – Compatibility
2. Each patch should not be used more than once
   – Or at least seldom being used more than once
   – Patch exclusion
3. The user's constraints on patch positions should be maintained
   – Local evidence

- These requirements can be enforced by terms in a Markov Random Field probability

MRF

- a node represents a spatial position where we will place a patch
  - the unknown state x_i at the ith node is the index of the patch to be placed there
  - each patch has four neighbors





  - compatibility $\psi_{i,j}(k,l)$

    the compatibility of patch k with patch l, placed at neighboring image positions i and j

MRF

- The probability of an assignment, $\mathbf{x}$, of patches to image positions

$$P(\mathbf{x}) = \frac{1}{Z} \prod_i \phi_i(x_i) \prod_{i,j \in N(i)} \psi_{ij}(x_i, x_j) E(\mathbf{x})$$

local evidence

compatibility

exclusion

user' constraints
on patch positions

zero if any two
elements of $\mathbf{x}$
are the same

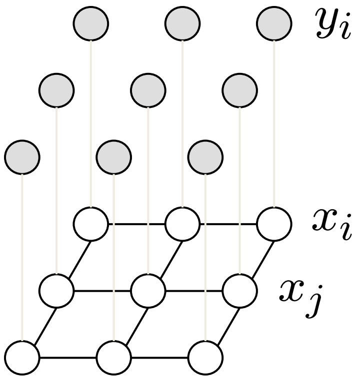
$$\mathbf{x} = \begin{bmatrix} 8 \\ 125 \\ \vdots \\ 30 \end{bmatrix}$$

patch #8 at position #1, patch #125 at position #2, ...

Factorization

$$P(\mathbf{x}) = \frac{1}{Z}\prod_i \phi_i(x_i) \prod_{i,j \in N(i)} \psi_{ij}(x_i, x_j) E(\mathbf{x})$$

$$P(\mathbf{x}) = \frac{1}{Z'}\prod_i \prod_{i,j \in N(i)} p(y_i|x_i) p_{i,j}(x_j|x_i) p(x_i) E(\mathbf{x})$$



$$p(y_i|x_i) = \phi_i(x_i)$$

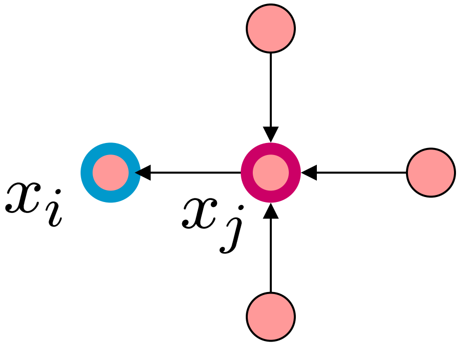
$$p(x_j|x_i) = \frac{\psi_{i,j}(x_i, x_j)}{\sum_{j=1}^{M} \psi_{i,j}(x_i, x_j)}$$

y_i is the original patch at location i

$p(x_i)$ as a uniform distribution (in most cases)

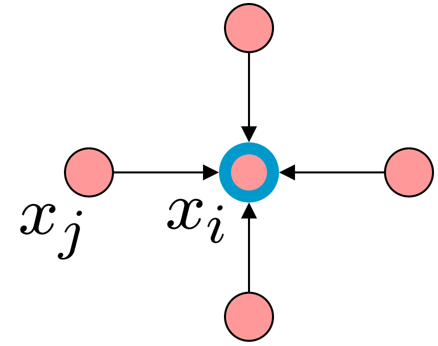
Message passing

$$m_{ji}(x_i) \propto \sum_{x_j} p_{i,j}(x_i|x_j) p(y_j|x_j) \prod_{l \in N(j)\backslash i} m_{lj}(x_j)$$



$$\widehat{x}_i = \arg\max_l b_i(x_i = l)$$

$$b_i(x_i) = p(y_i|x_i) \prod_{j \in N(i)} m_{ji}(x_i)$$



Avoiding self-repetitions

- Add a factor node that is connected to every image node x_i

$$m_{fi}(x_i) = \sum_{\{x_1,\ldots,x_N\}\backslash x_i} \psi_F(x_1,\ldots,x_N|x_i) \prod_{t\in S\backslash i} m_{tf}(x_t)$$

S is the set of all node

$\psi_F(\cdot) = 0$ if any of the two nodes $(x_l, x_m) \in S$ share the same patch;
$\psi_F(\cdot) = 1$ otherwise.

User-specified constraints

If the user has constrained patch k to be at image position i, then $p(y_i|x_i = k) = 1$ and $p(y_i|x_i = l) = 0$ for $l \neq k$.

At unconstrained nodes:

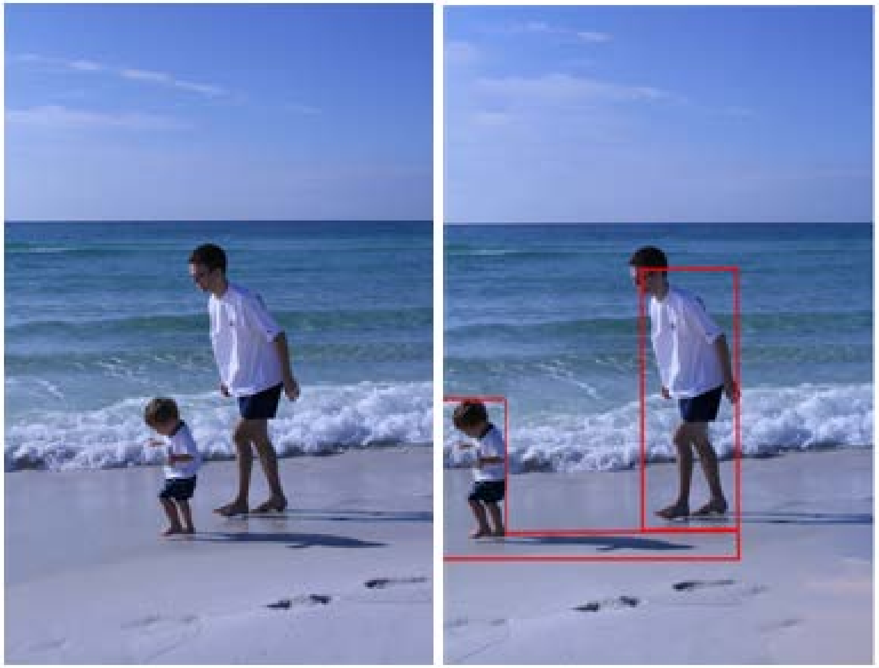
the low-resolution version of the original image can serve as a noisy observation y_i:

$$p(y_i|x_i = l) \propto \exp\left(-\frac{(y_i - m(l))^2}{\sigma_{evid}^2}\right)$$

$m(l)$ is the mean color of patch l

To keep the scene structure correct

Re-centering the region of interest



Manipulating the patch statistics of an image

$p(x_i)$

For example, if a user specified that sky should be reduced (by clicking on a sky patch x_s)

$$p(x_i|x_s) \propto \exp\left(\frac{(f(x_i)-f(x_s))^2}{\sigma_{sp}^2}\right)$$

$f(\cdot)$ extracts the feature of a patch, such as the mean color

影像處理和電腦視覺常見的應用

- MRF
  - Belief propagation
  - Graph cuts
  - Stereo matching
  - Super-resolution
  - Texture synthesis
  - …

Issues on using MRFs

- Given observations **y**, and the parameters of the MRF, how to *infer* the hidden variables **x**?

- How to *learn* the parameters of the MRF?

- Is the model (energy function) correct at all?
  – Global minimum vs. ground truth
  – Two sources of approximation
    - the model and the optimization method

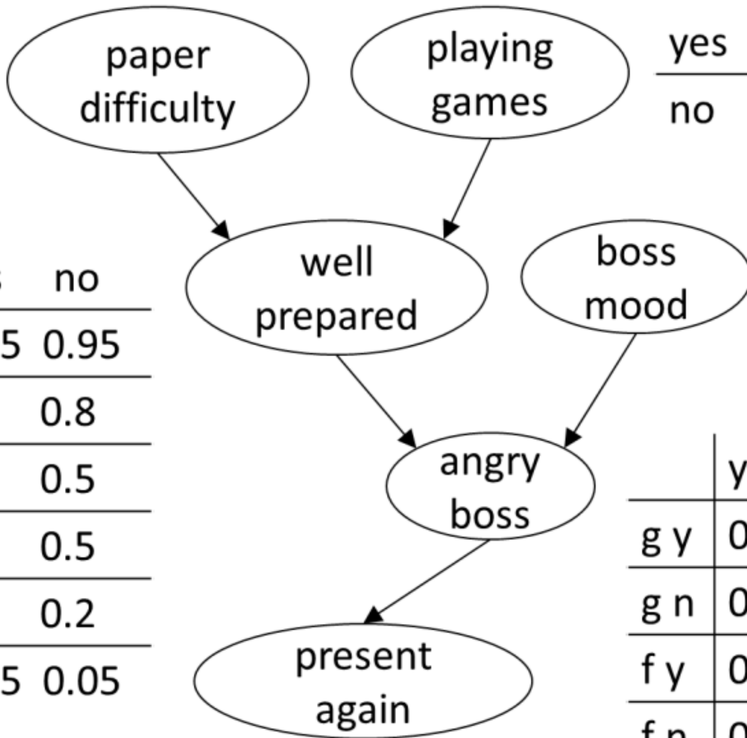
- 內容
  - representation
  - inference
  - learning
- 實例
  - 電腦視覺、影像處理

Learning

- Parameter estimation in Bayesian networks (for the CPDs)
  - maximum likelihood estimation
  - Bayesian parameter estimation: prior distribution over the parameters

- Parameter estimation in Markov networks
  - the normalization constant is responsible for coupling the estimation parameters and effectively ruling out a closed-form solution

第一節課的例子: Bayesian network



hard	0.25
fair	0.5
easy	0.25

yes	0.7
no	0.3

good	0.2
fair	0.6
bad	0.2

	yes	no
y h	0.05	0.95
y f	0.2	0.8
y e	0.5	0.5
n h	0.5	0.5
n f	0.8	0.2
n e	0.95	0.05

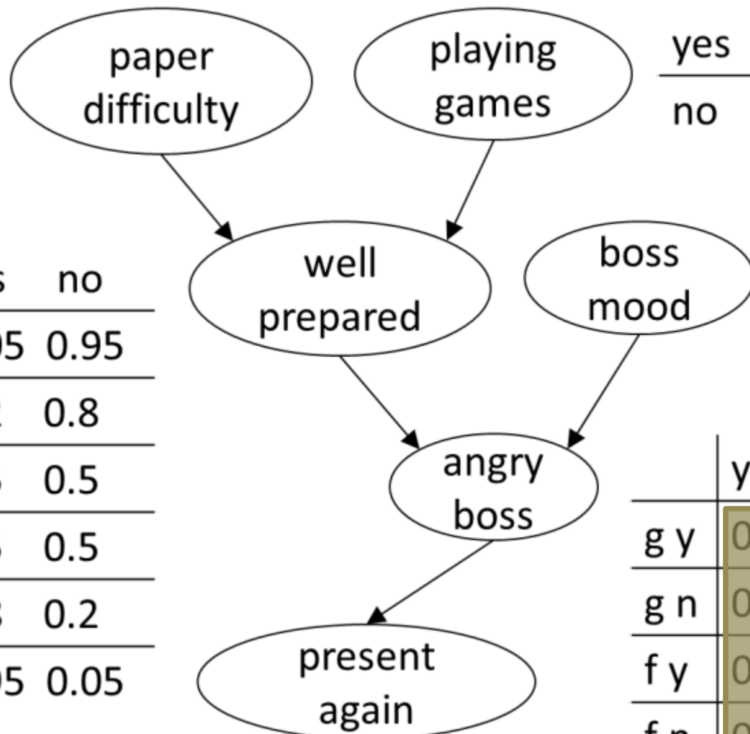
	yes	no
g y	0.01	0.99
g n	0.4	0.6
f y	0.1	0.9
f n	0.7	0.3
b y	0.2	0.8
b n	0.95	0.05

	yes	no
ab y	0.95	0.05
ab n	0.05	0.95

蒐集資料 估計參數

hard	0.25
fair	0.5
easy	0.25

yes	0.7
no	0.3



	yes	no
y h	0.05	0.95
y f	0.2	0.8
y e	0.5	0.5
n h	0.5	0.5
n f	0.8	0.2
n e	0.95	0.05

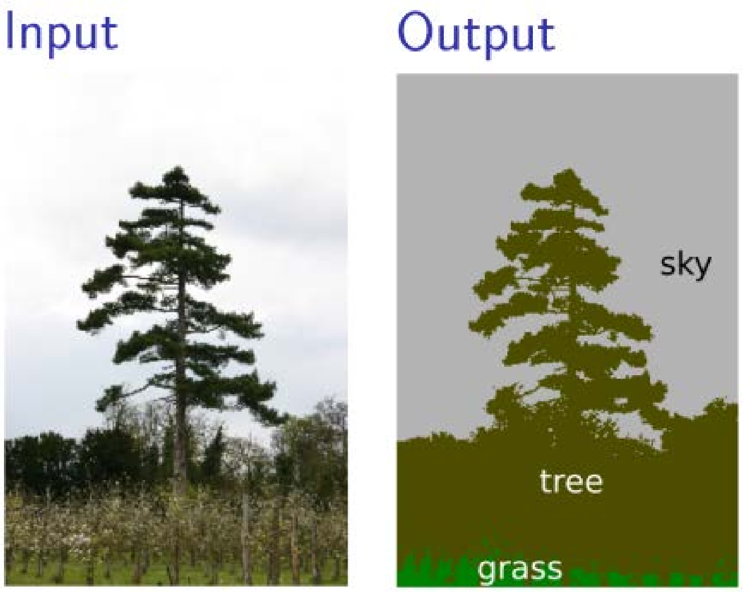
good	0.2
fair	0.6
bad	0.2

	yes	no
g y	0.01	0.99
g n	0.4	0.6
f y	0.1	0.9
f n	0.7	0.3
b y	0.2	0.8
b n	0.95	0.05

	yes	no
ab y	0.95	0.05
ab n	0.05	0.95

實例五

- *Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials*
  - P. Krahenbuhl and V. Koltun, NIPS 2011.
  - http://graphics.stanford.edu/projects/densecrf/
  - representation
  - inference
  - learning



Main idea: representation

conditional random field (CRF) model:

Gibbs distribution

$$P(\mathbf{X}|\mathbf{I}) = \frac{1}{Z(\mathbf{I})} \exp\left(-\sum_{c \in C_g} \phi_c(\mathbf{X}_c|\mathbf{I})\right)$$

- given a labeling $\mathbf{x} \in \mathcal{L}^N$

$$E(\mathbf{x}|\mathbf{I}) = \sum_{c \in C_g} \phi_c(\mathbf{x}_c|\mathbf{I})$$

- maximum a posteriori (MAP)

$$\mathbf{x}^* = \arg\max_{\mathbf{x} \in \mathcal{L}^N} P(\mathbf{x}|\mathbf{I})$$

Fully connected CRF model

$$E(\mathbf{x}) = \sum_i \psi_u(x_i) + \sum_{i<j} \psi_p(x_i, x_j), \quad i, j \in \{1, \dots, N\}.$$

- ψ_u: unary potential
  - from classifiers
  - features: shape, texture, color, location
  - noisy, inconsistent
- ψ_p: pairwise potential

$$\psi_p(x_i, x_j) = \mu(x_i, x_j) \sum_{m=1}^{K} w^{(m)} k^{(m)}(f_i, f_j)$$

$k^{(m)}$: Gaussian kernel $k^{(m)}(f_i, f_j) = \exp(-\frac{1}{2}(f_i - f_j)^T \Lambda^{(m)}(f_i - f_j))$

$K = 2$: $k(f_i, f_j) = w^{(1)} \exp\left(-\frac{|p_i - p_j|^2}{2\theta_\alpha^2} - \frac{|I_i - I_j|^2}{2\theta_\beta^2}\right) + w^{(2)} \exp\left(\frac{-|p_i - p_j|^2}{2\theta_\gamma^2}\right)$

$\mu(x_i, x_j)$: local compatibility function

Main idea: inference

Mean field approximation

- maximizing $P(\mathbf{X})$ is hard: $Q(\mathbf{X})$ product of independent marginals $Q(\mathbf{X}) = \prod_i Q_i(X_i)$

Algorithm

1. maximizing $Q(\mathbf{X})$
2. minimizing KL-divergence $KL(Q \| P)$, constraining $Q(\mathbf{X})$, $Q_i(X_i)$ to be valid distributions

Main idea: learning

Unary classifiers

- JointBoost algorithm

Parameters $w^{(1)}$, θ_α, θ_β

- grid search

$\mu(a, b)$

- L-BFGS maximizing log-likelihood $L(\mu : \mathcal{I}, \mathcal{T})$, with ground-truth labeling $\mathcal{T}$
  - estimate gradient of Z

$$
\begin{aligned}
\frac{\partial}{\partial \mu(a,b)} L(\mu : \mathcal{I}^{(n)}, \mathcal{T}^{(n)}) &\approx -\sum_i \mathcal{T}_i^{(n)}(a) \sum_{j \neq i} k(f_i, f_j) \mathcal{T}_j^{(n)}(b) \\
&+ \sum_i Q_i(a) \sum_{j \neq i} k(f_i, f_j) Q_j(b)
\end{aligned}
$$

目標

- 甚麼是 probabilistic graphical models?
- 可以用來解決甚麼問題? 怎麼用?

Summary

- representation
- inference
- learning

- 電腦視覺、影像處理

- 參考 “Probabilistic Graphical Models: Principles and Techniques”
  - Daphne Koller and Nir Friedman
  - http://pgm.stanford.edu/
  - MOOC course on *Coursera*
  - “Graphical Models in a Nutshell” http://ai.stanford.edu/~koller/Papers/Koller+al:SRL07.pdf

Approximation on $\psi_F(\cdot)$

$$\psi_F(x_1,\ldots,x_N|x_i) \approx \prod_{t\in S\backslash i} \psi_{F_t}(x_t|x_i)$$

$$\psi_{Ft}(x_t|x_i) = 1 - \delta(x_t - x_i)$$

$$m_{fi}(x_i = l) \approx \prod_{t\in S\backslash i} \sum_{x_t} \psi_{F_t}(x_t|x_i = l) m_{tf}(x_t)$$

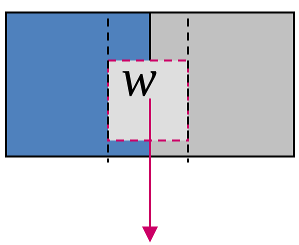
$$= \prod_{t\in S\backslash i} (1 - m_{tf}(x_t = l))$$

Compatibility among patches

$$\psi_{i,j}(k,l) = \psi^A_{i,j}(k,l)\psi^B_{i,j}(k,l)$$

GSM FOE
natural image statistics

patch i patch j

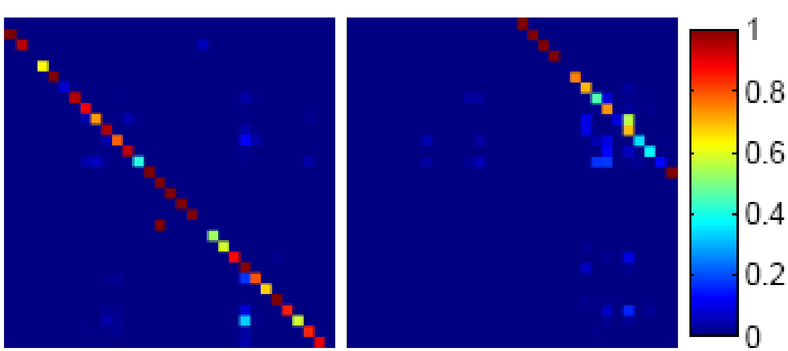


$$\psi^A_{i,j}(k,l) = \frac{1}{Z}\prod_{n,m}\sum_{q=1}^{J}\left\{\frac{\pi_q}{\sigma_q}\exp(-w_n^T I_m(k,l))\right\}$$

$$\psi^B_{i,j}(k,l) \propto \exp\left(-\frac{(r(k)-r(l))^2}{\sigma^2_{clr}})\right)$$

color compatibility along the boundary

$p_{LR}(i,j)$ $p_{DU}(i,j)$



Mean field iteration

$$Q_i(x_i=\ell) = \frac{1}{Z}\exp\left\{-\psi_u(x_i) - \sum_{\ell'\in\mathcal{L}}\mu(\ell,\ell')\sum_{m=1}^{K} w^{(m)}\sum_{j\neq i} k^{(m)}(f_i,f_j)Q_j(\ell')\right\}$$

1. initializing $Q_i(x_i) \longleftarrow \frac{1}{Z_i}\exp\{-\psi_u(x_i)\}$
2. while not converged, do

$$\begin{aligned}
\widetilde{Q}_i^{(m)}(\ell) &\longleftarrow \sum_{j\neq i} k^{(m)}(f_i,f_j)Q_j(\ell), \quad m=1,2 \\
\widehat{Q}_i(x_i)) &\longleftarrow \sum_{\ell\in\mathcal{L}}\mu(x_i,\ell)\sum_m w^{(m)}\widetilde{Q}_i^{(m)}(\ell) \\
Q_i(x_i) &\longleftarrow \exp\{-\psi_u(x_i) - \widehat{Q}_i(x_i)\} \\
\text{normalize} \quad & Q_i(x_i)
\end{aligned}$$

Complexity

- message passing: all $x_j \to x_i$, $O(N^2)$
- compatibility: linear time
- local update: linear time