Image Completion Using Global Optimization

Nikos Komodakis and Georgios Tziritas
University of Crete, Computer Science Department, E-mail: {komod,tziritas}@csd.uoc.gr
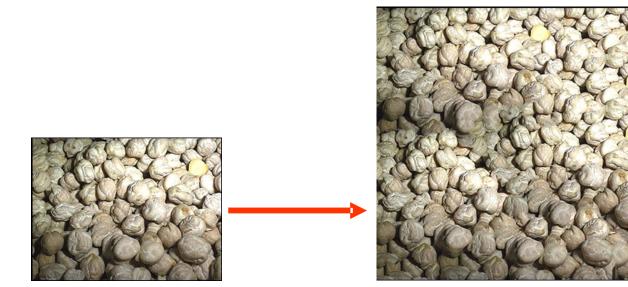
The image completion problem

☐ Fill missing part of an incomplete image in a visually plausible way

☐ We want to be able to handle:
- complex natural images,
- with a (possibly) large missing region
- and all these in an automatic way (i.e. no user intervention)

☐ We also want our method to be able to handle the related problems of **texture synthesis** and image **inpainting**

Related methods

☐ **Statistical-based**: they use parametric statistical models for textures (mostly suitable for synthesis of stochastic textures)

☐ **PDE-based**: they fill region through a diffusion process, simulated by a non-linear PDE (suitable for thin & smooth missing regions)

☐ **Exemplar-based**: they fill by copying patches from observed part (very good at texture synthesis but may lead to bad completions due to greedy decisions)

Contributions to image completion

☐ A new exemplar-based framework is proposed that unifies image completion, texture synthesis and image inpainting

☐ All these tasks are posed as discrete MRF optimization problems with a well-defined energy (but with a huge number of labels)

☐ Completion is solved in a global way (i.e. no greedy decisions)

☐ Our method is fully automatic and does not rely on any image specific prior knowledge (i.e. it can be applied to any image)

Contributions to MRF optimization

☐ A novel MRF optimization scheme, called "**Priority-BP**", is proposed that can handle discrete MRFs with a huge number of labels.

☐ **Priority-BP** carries 2 extensions over standard **Belief Propagation**:
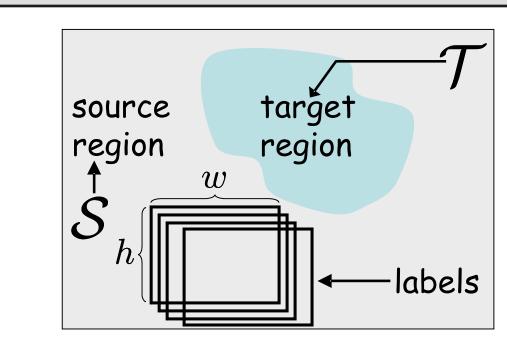- "Dynamic label pruning" and
- "Priority-based message scheduling"

☐ Generic method for efficiently applying Belief Propagation to MRFs with very large discrete state-spaces
(it thus resolves one of the main limitations of Belief Propagation)

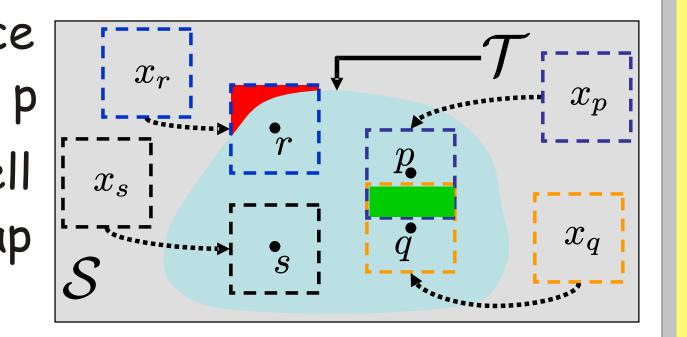☐ Priority-BP is generic (**i.e. it can be applied to any MRF**)

Completion as MRF optimization

Labels L = all w×h patches from S (a huge number!)

MRF nodes = all lattice points whose w×h neighborhood intersects target region T

singleton potential $V_p(x_p)$ = how well source patch x_p agrees with source region around p

pairwise potential $V_{pq}(x_p,x_q)$ = how well source patches x_p,x_q agree on their overlap after placing them above p and q
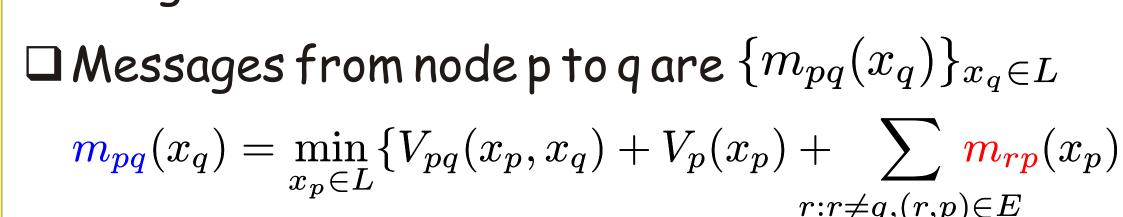

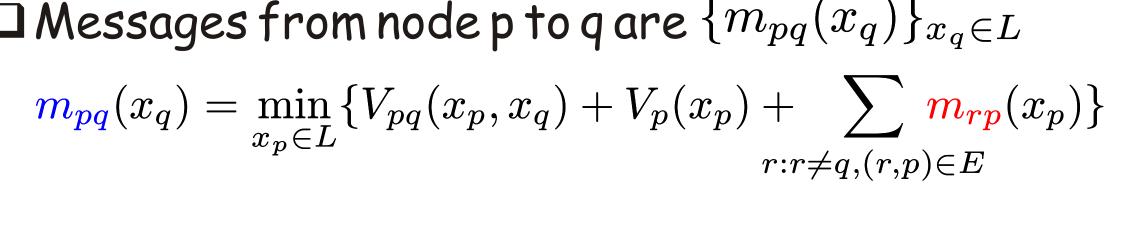E.g. $V_r(x_r)$=match over **red region**, $V_{pq}(x_p,x_q)$=match over **green region**

Completion then reduces to optimizing the following MRF energy:

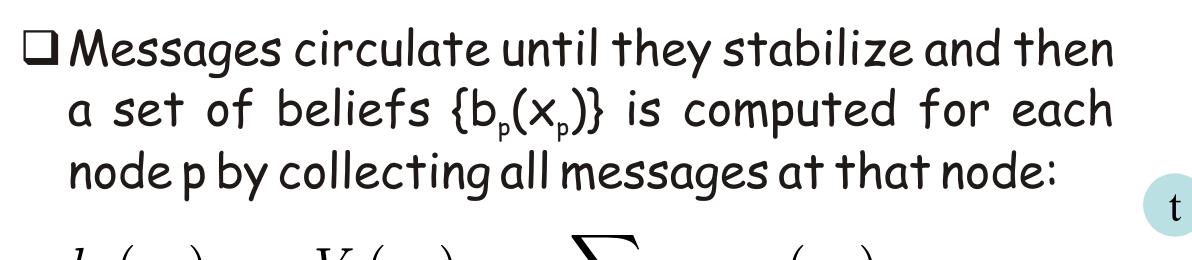$$\sum_p V_p(x_p) + \sum_{(p,q)\in E} V_{pq}(x_p,x_q)$$

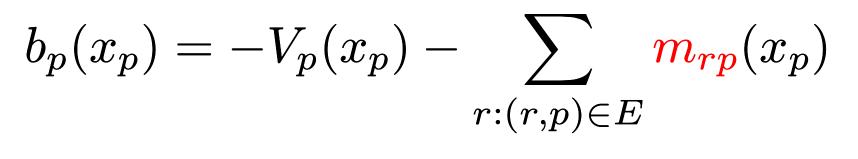Intuitively, one can think of it as trying to solve a huge jigsaw puzzle.

Belief Propagation (BP)

☐ BP is an MRF optimization technique, which is based on the idea of propagating local messages along the nodes of an MRF

☐ Messages from node p to q are $\{m_{pq}(x_q)\}_{x_q \in L}$

$$m_{pq}(x_q) = \min_{x_p \in L}\{V_{pq}(x_p,x_q) + V_p(x_p) + \sum_{r:r\neq q,(r,p)\in E} m_{rp}(x_p)\}$$

☐ Messages circulate until they stabilize and then a set of beliefs $\{b_p(x_p)\}$ is computed for each node p by collecting all messages at that node:

$$b_p(x_p) = -V_p(x_p) - \sum_{r:(r,p)\in E} m_{rp}(x_p)$$

☐ Each node is assigned a label of maximum belief: $\hat{x}_p = \arg\max_{x_p \in L} b_p(x_p)$

Priority-BP

☐ In our case, BP has an intolerable computational cost:
just the basic operation of sending messages from p to q requires $|L|^2$ patch comparisons (recall that $|L|$ is huge!)

☐ To fix this, Priority-BP carries 2 important extensions over BP:
- "**Dynamic label pruning**" and "**Priority-based message scheduling**"

☐ Resolves one of the main limitations of BP:
- Its inefficiency to handle MRFS with a huge number of labels
- Active research topic, but existing methods apply only to restricted class of MRFs, e.g. see [Felzenswalb04]
- On the contrary, our method is generic (i.e. applicable to any MRF)

☐ Priority-BP is based on the following key-idea:
- It tries to drastically reduce the number of labels
- But instead of pruning the labels beforehand, this is done on the fly
- To this end, it makes use of information carried only by BP itself (this ensures that the algorithm will be generic)

Priority-based message scheduling

☐ Not all MRF nodes are equally adequate for pruning their labels
e.g. node **a** is very inappropriate for pruning, whereas node **c** can tolerate a lot of pruning

☐ Therefore, we assign a **measure of "confidence"** to MRF nodes:
- And then apply the following scheduling to messages:

> **Message scheduling principle:** the most "confident" nodes transmit their messages first, i.e. they have higher priority
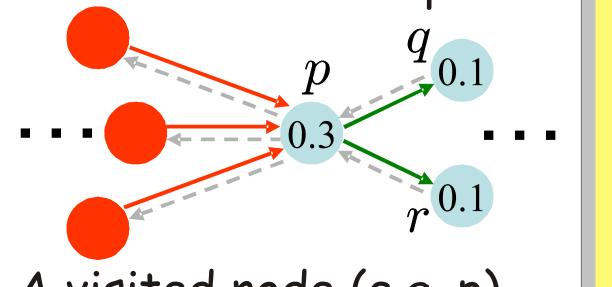
☐ The above scheduling is good for the following reasons:
- The more confident a node, the more pruning it can tolerate. But recall, message from non-pruned node costs $O(|L|)$ time whereas, message from pruned node is cheap: $O(|L_{pruned}|)$ time (**only cheap messages thus circulate in the network**)
- Confident nodes help their neighbors to increase their own confidence (**even more pruning is thus possible**)
- Informative messages are propagated first (**convergence of BP is accelerated**)

Forward/Backward message pass

☐ To ensure that messages across all nodes get transmitted at each iteration, we divide each iteration into a **forward** & **backward** pass:

☐ Message scheduling and pruning takes place only during the forward pass (half of the messages get transmitted here)

☐ At forward pass, we visit nodes by priority. A visited node (e.g. p) sends messages to its unvisited neighbors only (e.g. q and r)

☐ Backward pass just ensures that the other half of the messages gets transmitted as well

Assigning priorities to nodes

☐ Obviously, priority should be proportional to confidence.
But how can we define confidence in a generic way?

☐ Key idea: use set of beliefs $\{b_s(x_s)\}$ calculated by BP for each node p

☐ For each node p, define its confusion set $\mathbb{CS}(p)$ containing the labels with high (relative) belief at that node:

$$\mathbb{CS}(p) = \{x_p \in L : b_p^{rel}(x_p) \geq b_{conf}\}$$

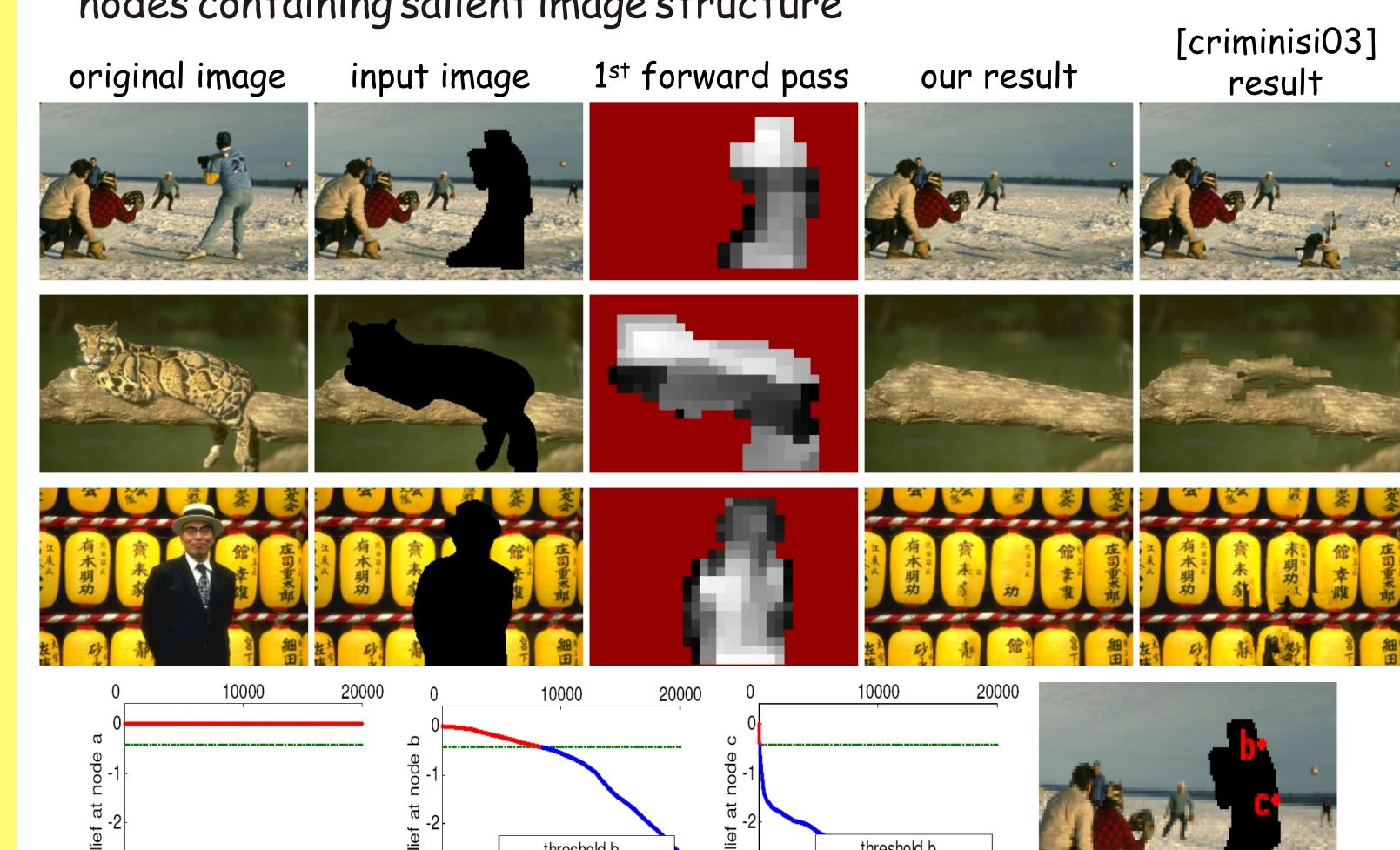☐ Priority of p then defined as: $\mathrm{priority}(p) = 1/|\mathbb{CS}(p)|$
(other entropy-like measures have been tested as well)

Applying Priority-BP to completion

☐ **Novel use of confidence in image completion:** we use it to decide the order of message passing and not for greedily choosing which patch to fill next

☐ Also, **confidence definition is generic** (no image heuristics involved)

☐ Algorithm learns by itself to propagate first the messages of the nodes containing salient image structure
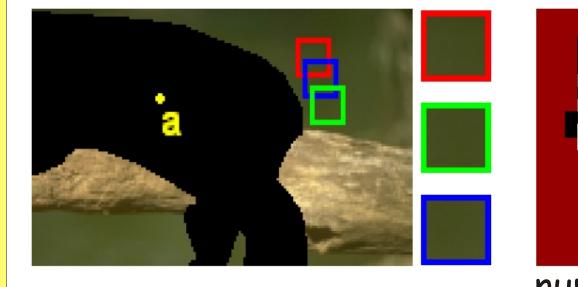

In the above plots, the beliefs for labels in the confusion set of a node are plotted in red. These plots illustrate how priority is defined for nodes **a**, **b** and **c** at the beginning.
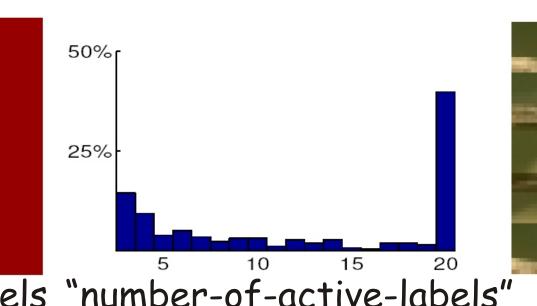
Dynamic label pruning

☐ What criterion should we use for label pruning?
Moreover, how can this criterion be generic as well?

☐ We will again make use of the beliefs calculated by the algorithm:
Active labels will be only those whose belief is high (i.e. belief exceeds a certain threshold, say b_{prune}).

☐ To discard very similar patches, an additional filtering of active labels is also needed (alternatively, texton clustering can be used [malik01])
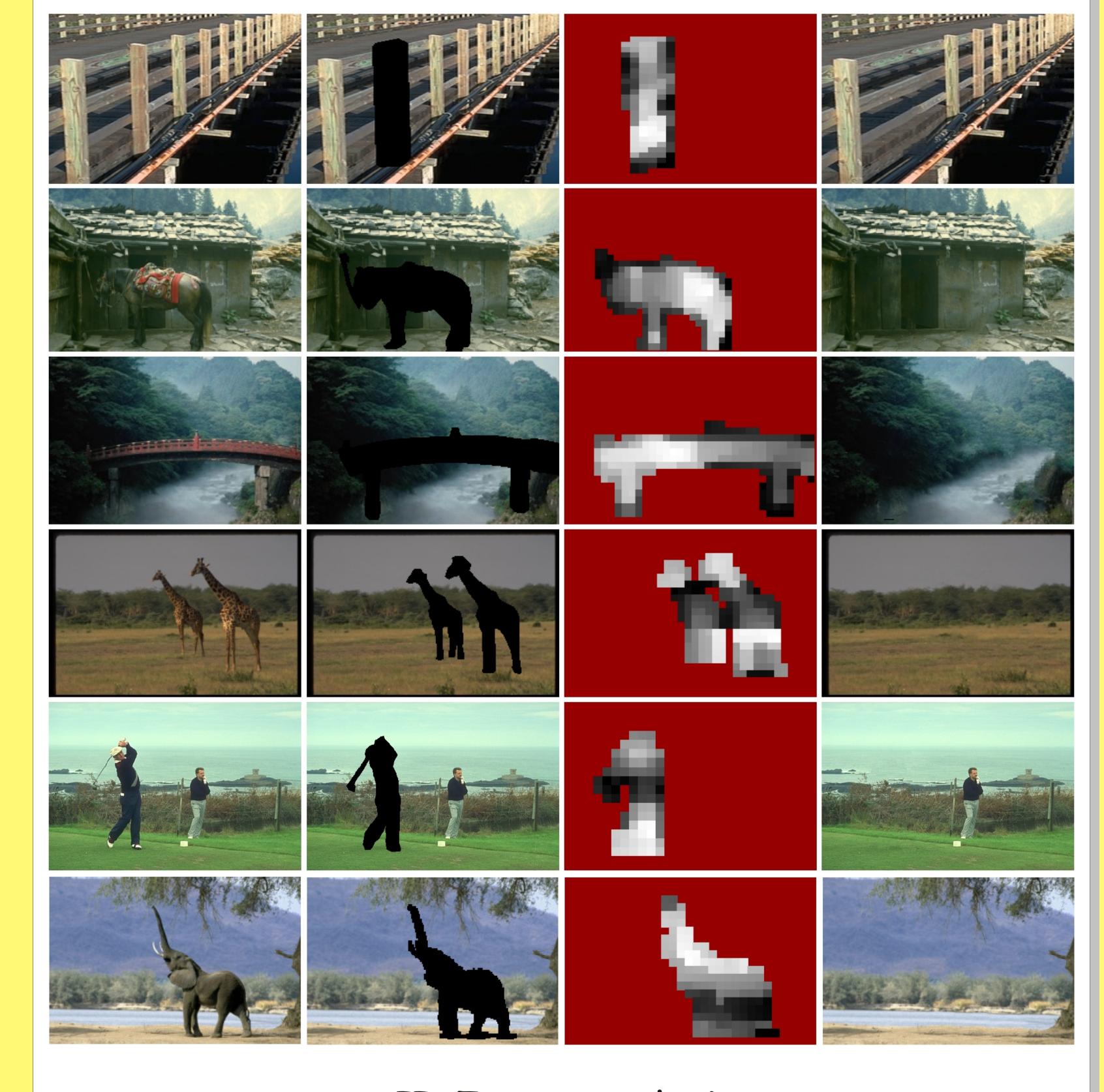

Extensions

☐ **Completion by energy refinement:** one may refine completion simply by refining (i.e. adding terms to) the energy function
E.g., to favor spatial coherence, one may add the following terms:

$$V_{pq}^0(x_p,x_q) = \begin{cases} w_0 & \text{if } x_p - x_q \neq p - q \\ 0 & otherwise \end{cases}$$
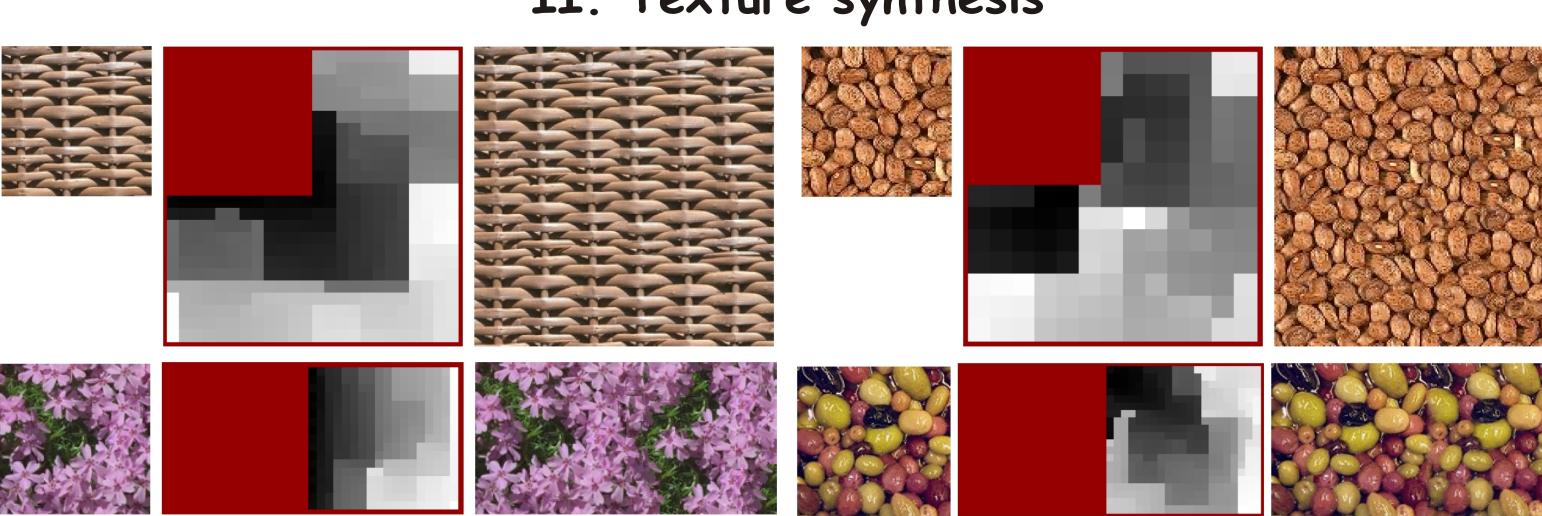
☐ **Multiscale image completion:** use completion at coarse scale to compute singleton-potentials at next finer scale
(Features at multiple scales can thus be captured).

Further results

I. Object Removal

original image   input image   1st forward pass   our result   [criminisi03] result

II. Texture synthesis

III. Image inpainting

IV. Text removal