Multi-Label Moves for Multi-Label Energies

Olga Veksler

University of Western Ontario

some work is joint with Olivier Juan, Xiaoqing Liu, Yu Liu

Western

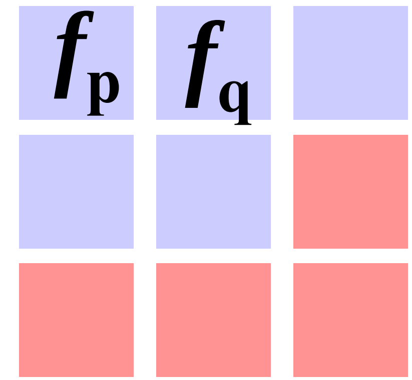
Outline

- Review multi-label optimization with graph cuts
  - Exact methods
  - Approximate methods
- Multi-label moves
  1. For piecewise-smooth smoothness term
     - applications to image restoration, stereo correspondence, simulating classic mosaic
  2. For order-preserving smoothness term
     - Applications to extracting structure from one image and to simple shape priors

Energy Minimization

- Standard energy formulation:

$$E(f) = \sum_{p \in \mathcal{P}} D_p(f_p) + \sum_{(p,q) \in \mathcal{N}} V_{pq}(f_p, f_q)$$

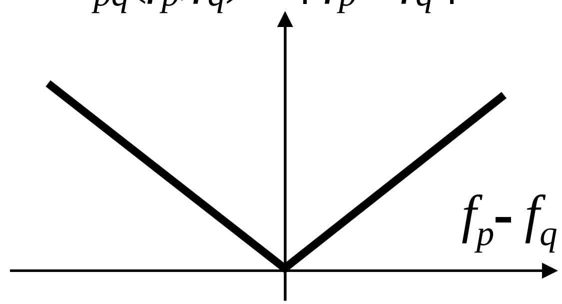


- for binary labels, the energy above can be minimized exactly if $V_{pq}(f_p, f_q)$ are submodular [*Hammer*, *Kolmogorov*]
- We are interested in the multi-label case

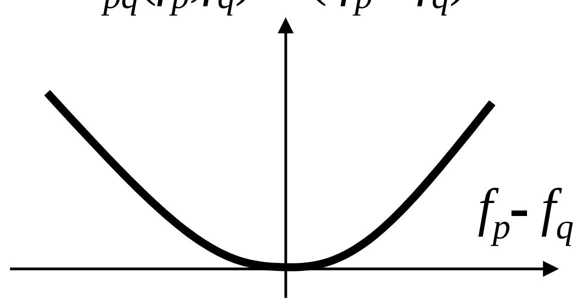
Exact Methods: Ishikawa’2003

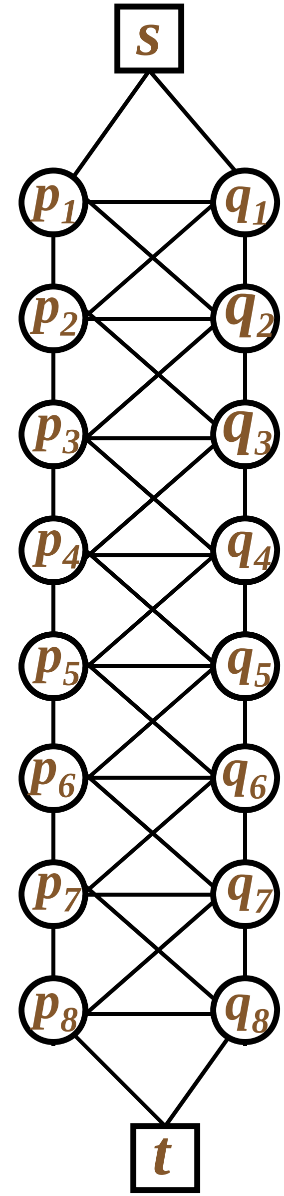
- Labels must be ordered
- If $V_{pq}(f_p,f_q)$ are *convex*, and symmetric in f_p-f_q, then can optimize energy exactly

$V_{pq}(f_p,f_q) = | f_p - f_q |$



$V_{pq}(f_p,f_q) = (f_p - f_q)^2$





Exact Methods: Schlesinger and Flach'2006

- Labels must be ordered
- $V_{pq}(f_p, f_q)$ is submodular if for any

$$\alpha < \beta \quad \text{and} \quad \alpha' < \beta'$$

$$V_{pq}(\alpha, \alpha') + V_{pq}(\beta, \beta') \leq V_{pq}(\alpha, \beta') + V_{pq}(\beta, \alpha')$$

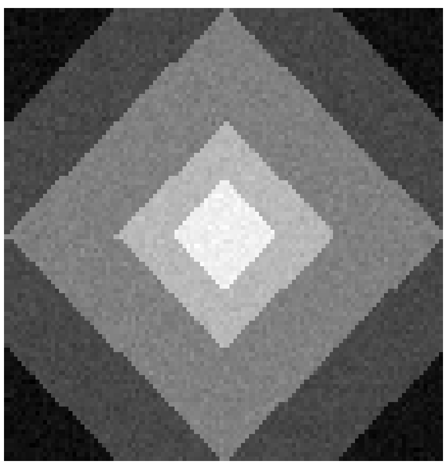
- More general that Ishikawa's construction, for example, does not require

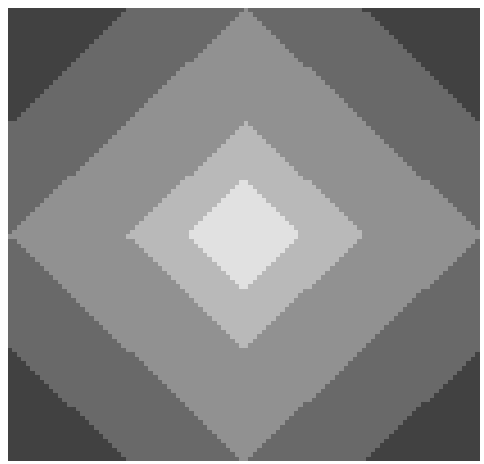
$$V_{pq}(\alpha, \beta) = V_{pq}(\beta, \alpha)$$

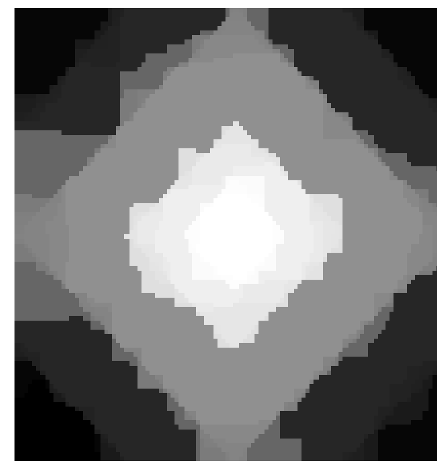
  which is important for some applications
- Submodular multi-label energy is converted to a submodular binary energy
- Graphs are very similar to Ishikawa's construction

Problems with Exact Methods

- As the number of labels increases, submodular $V_{pq}(f_p,f_q)$'s are not bounded from above
- Therefore submodular $V_{pq}(f_p,f_q)$'s are not discontinuity preserving
- Example: Restoration with $V_{pq}(f_p,f_q) = | f_p - f_q |$

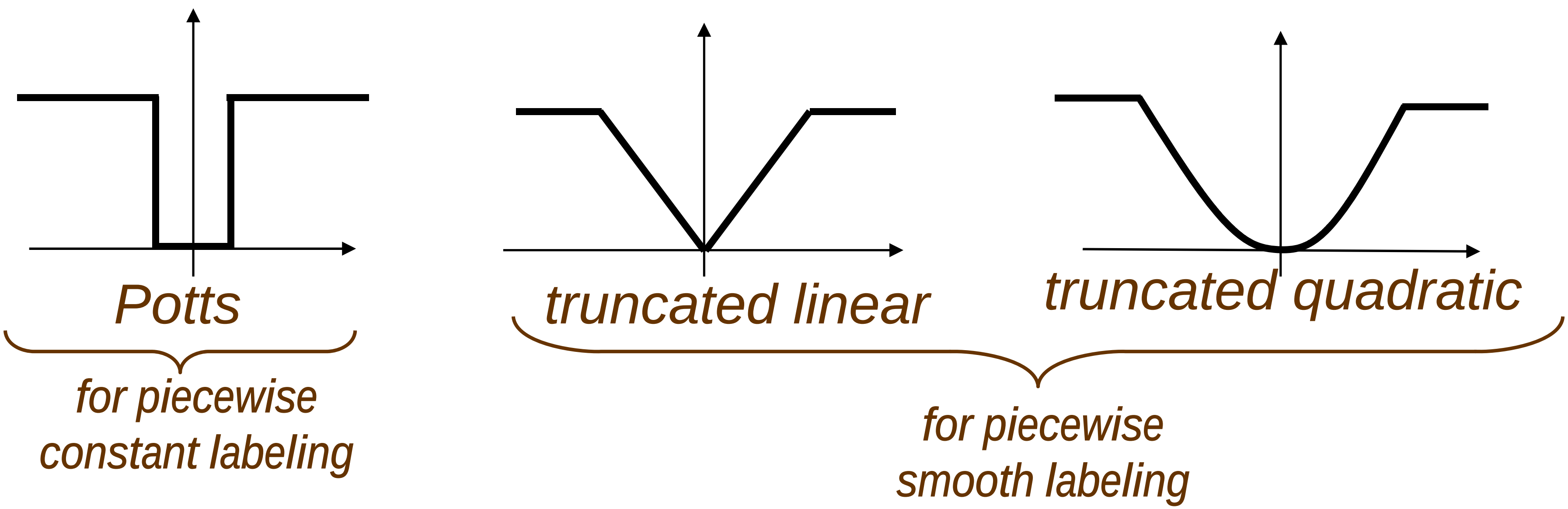

noisy image


want this


get this

Approximate Methods

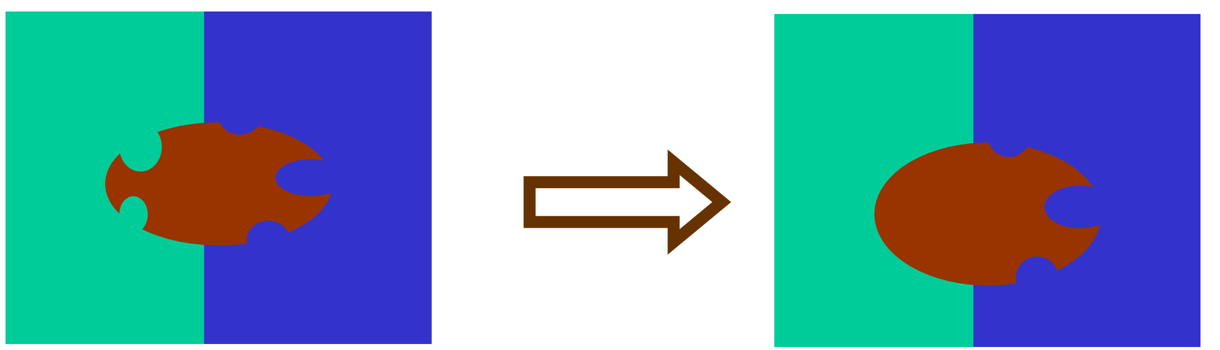
- The following discontinuity preserving $V_{pq}(f_p,f_q)$ are NP-hard to optimize:



Approximate Methods

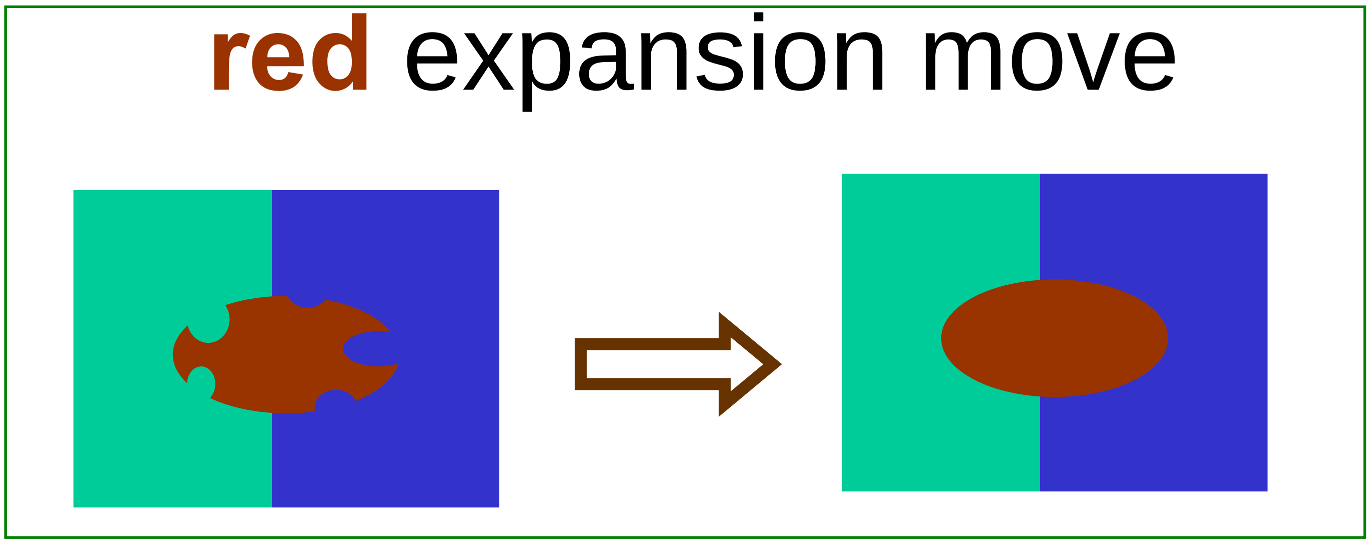
- For the Potts and truncated linear & quadratic models, we can find a local minimum with respect to swap moves [Boykov, Veksler, Zabih'1998]:

green-**red** swap move



Approximate Methods

- For the Potts and truncated linear model, we can find a local minimum with respect to expansion moves [Boykov, Veksler, Zabih'1998]:

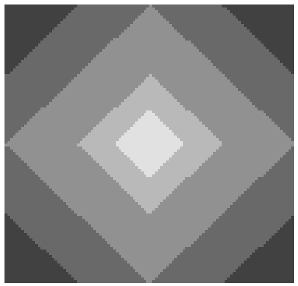


- Local minimum wrt expansion moves is optimal within a factor (2 for the Potts model, worse for truncated linear model)

Limitations of Swap and Expansion Moves

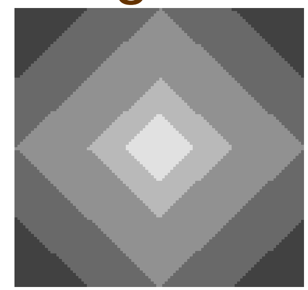
- work well for piecewise-constant labelling:

want:



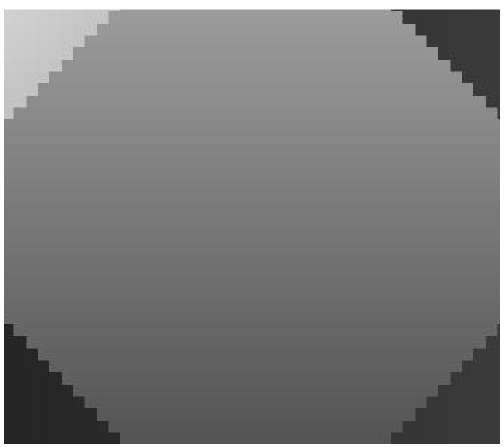
Potts V

get:



- don’t work well for piecewise-smooth labelling:

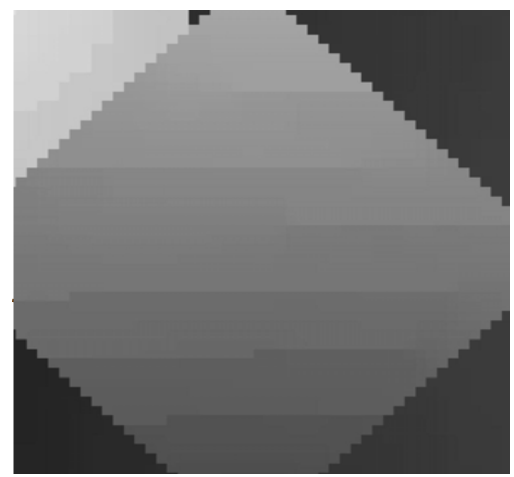
want:



truncated

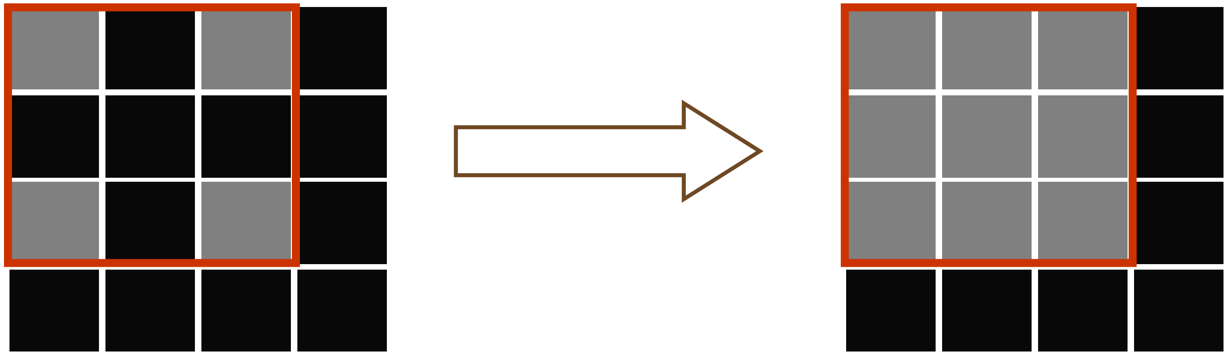
quadratic

get:



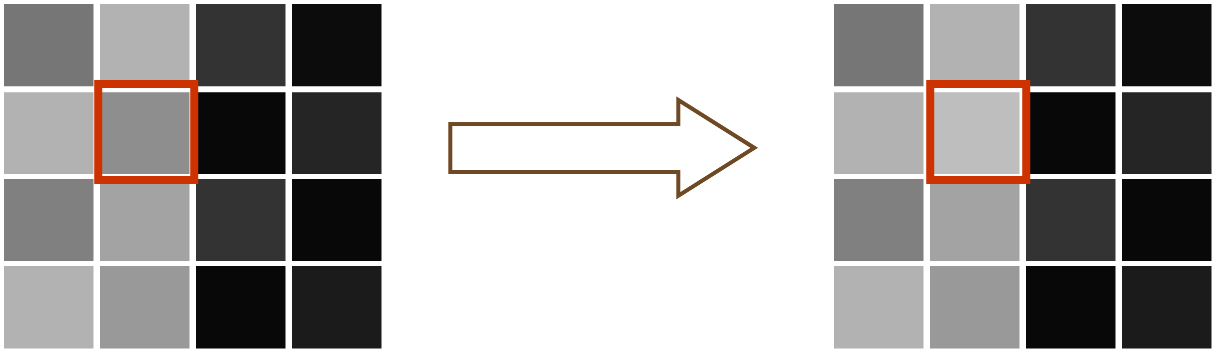
Insight

- Swap and expansion moves are binary in essence
  - each pixel participating in a move can either stay the same or switch to a single "new" label
  - the "new" label is either the same for all pixels (for expansion move) or almost the same (either α or β for the swap move)
- piecewise-constant labelling:
  - many pixels prefer the same label
  - swap & expansion moves are successful because they can operate on large sets of pixels



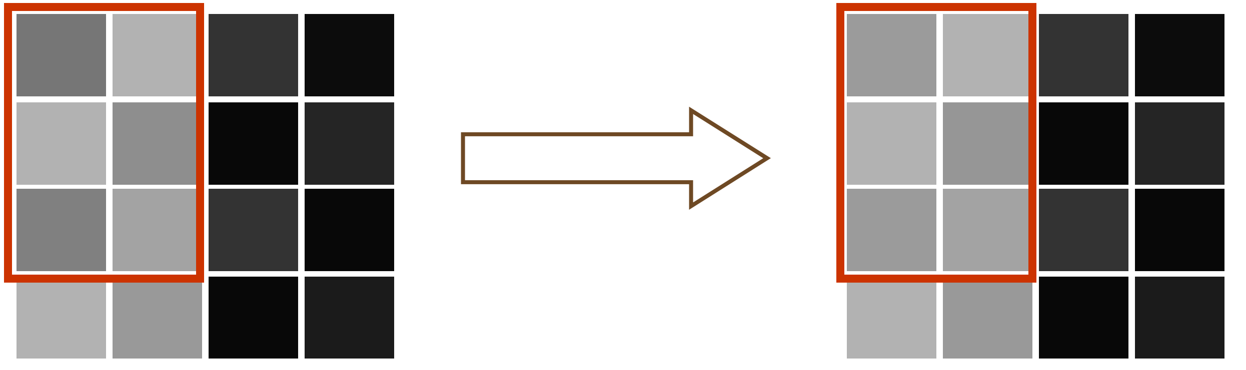
Insight

- piecewise-smooth labeling:
  - Only a small group of pixel wants the same label
  - swap & expansion moves operate on small sets of pixels, and therefore much closer to “standard moves”



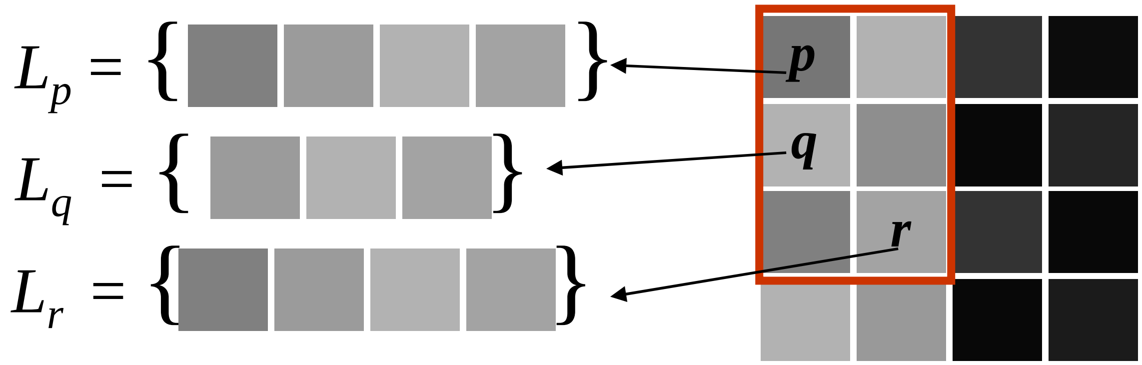
Key Idea

- Need to develop moves that work on larger sets of pixels for piecewise-smooth models
- must give each pixel a larger set of labels to switch to in a single move



Can switch to: 

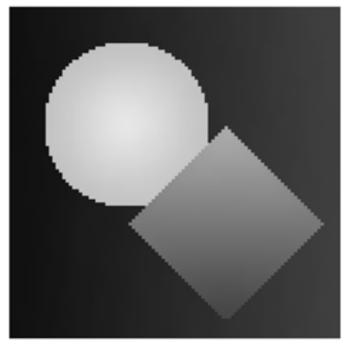
Multi-Label Moves



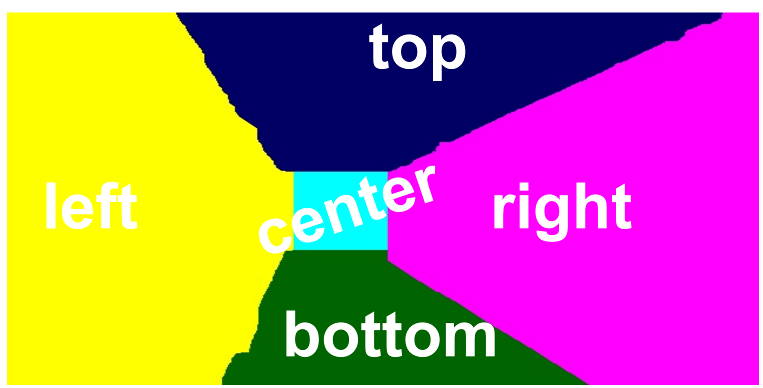
- Given current labelling f find
  - a subset of pixels P and
  - a set of labels L_p for each pixel $p \in P$
  - such that the restriction of the energy to P and L_p's is a submodular energy (multi-label)
    - $V_{pq}(f_p,f_q)$ must be submodular inside
    - $V_{pq}(f_p,f_q)$ can be arbitrary outside
- want $f_p \in L_p$
- want P and L_p's to be as large as possible

Multi Label Moves

- Energy Dependant
- Will explore two major types of moves
  - For energies with piecewise-smooth V_{pq}



  - For energies with order-preserving V_{pq}



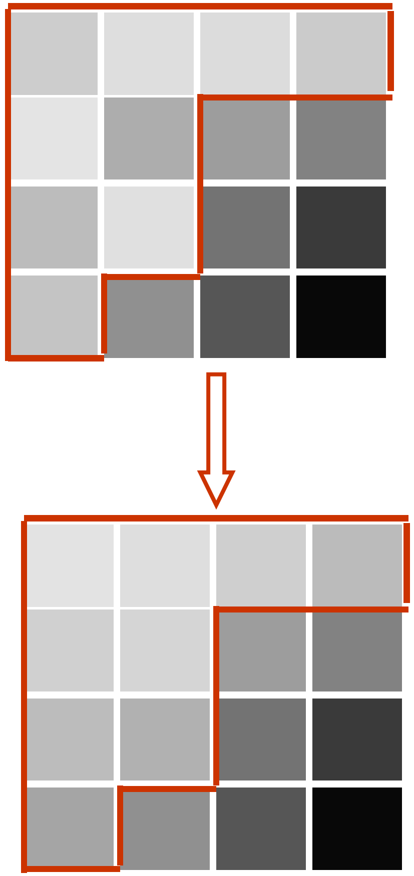
Piecewise-Smooth V_{pq}: Range Moves

- Based on Ishikawa construction [PAMI'2003]
  - have to assume ordered labels
- Applies to truncated convex $V_{pq}(f_p,f_q)$
  - includes truncated linear and quadratic
- Let T be truncation constant, i.e.

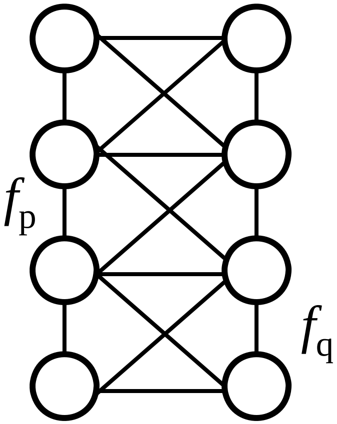
$$V_{pq}(f_p,f_q) \leq T$$

- Reassigns labels of pixels currently labelled $\alpha,\alpha+1,\ldots,\beta$ to labels in $\{\alpha,\alpha+1,\ldots,\beta\}$ in the optimal way
  - Restriction: $|\alpha-\beta| \leq T$
- Generalization (sort of) α - β swap

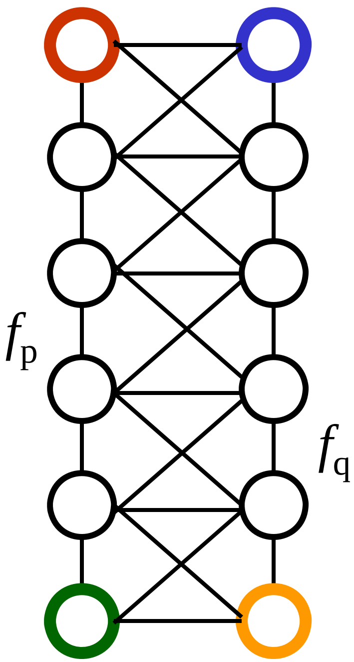
Ishikawa's construction inside, boundary terms dealt with through data terms



P-Smooth V_{pq}: Generalized Range Moves



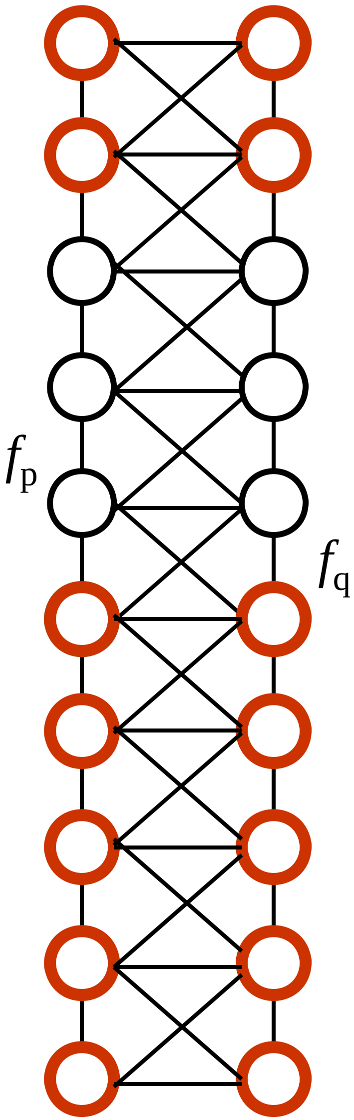
- α-β range moves in case $|\alpha - \beta| < T$
- all $V_{pq}(\mathbf{O},\mathbf{O})$ are correct



- generalized α-β range moves to case $|\alpha - \beta| > T$
- $V_{pq}(\mathbf{O},\mathbf{O})$ and $V_{pq}(\mathbf{O},\mathbf{O})$ are larger than they should be
- all other $V_{pq}(\mathbf{O},\mathbf{O})$ are correct
- in particular, $V_{pq}(f_p, f_q)$ is correct
- old labeling f has correct energy
- all labelings not including have the same (correct) energy as in the construction above
- any labelings involving $\mathbf{O}$,$\mathbf{O}$,$\mathbf{O}$,$\mathbf{O}$ have energy higher than they should

$$E\Big(\underset{|\alpha - \beta| \leq T}{\text{optimal } \alpha\text{-}\beta \text{ r.m.}}\Big) \geq E\Big(\underset{|\alpha - \beta| > T}{\text{generalized } \alpha\text{-}\beta \text{ r.m.}}\Big)$$

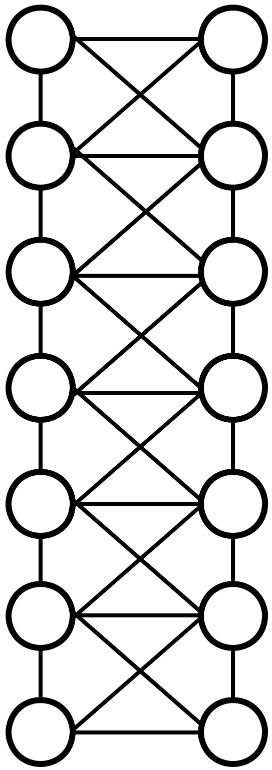
P-Smooth V_{pq}: Generalized Range Moves



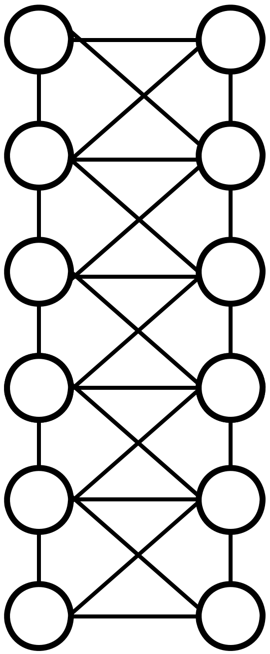
- could “augment” the construction using the whole label range
- however, computing the best move becomes very expensive for little added benefit
- in practice, augment construction by two labels on the top and bottom

P-Smooth V_{pq}: "Graduated non-Submodularity"

- start with un-truncated V_{pq}
- let un-truncated energy be E^{max}
- global minimum f^* of E^{max} is found



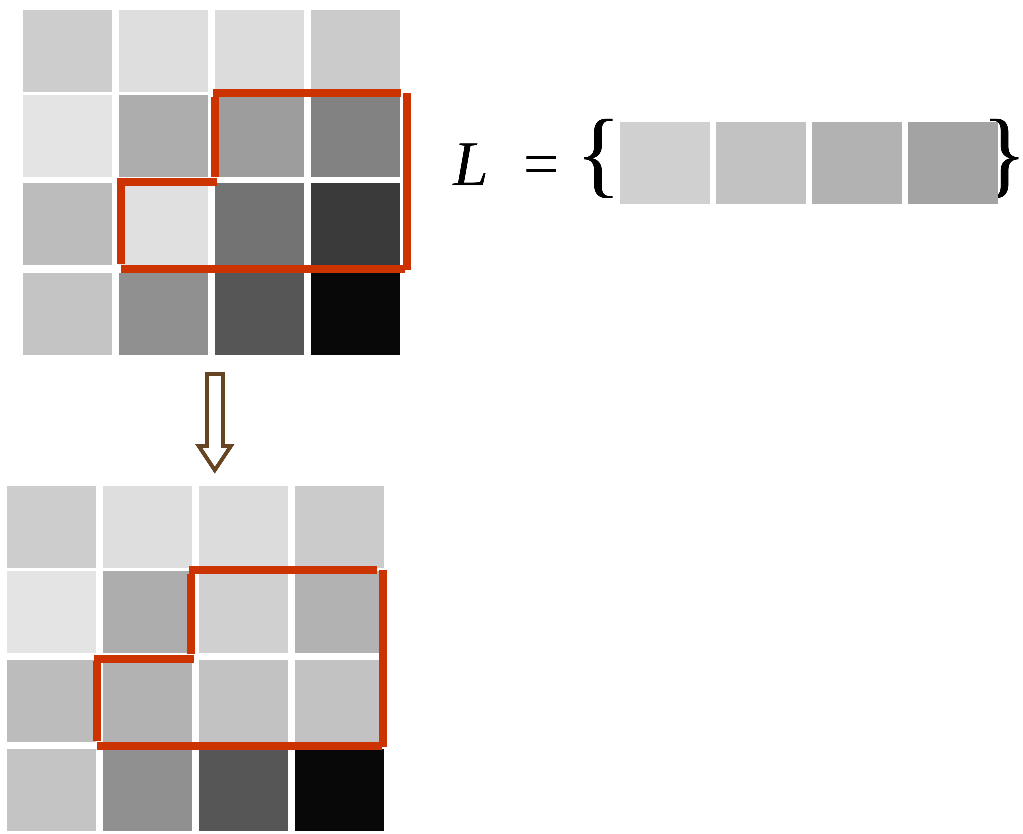
- start with f^* as initial solution
- truncate V_{pq} a little bit
- let this energy be E^{max-1}
- $E^{max-1}(f^*) \leq E^{max}(f^*)$
- apply range moves to optimize E^{max-1}

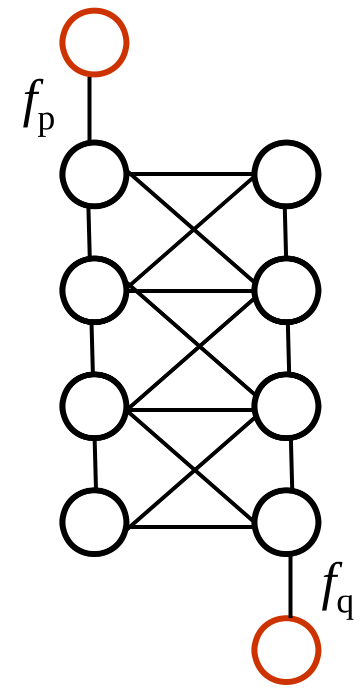


- continue until reach original truncated energy
- works better than range moves, but takes a whiiiiiile….

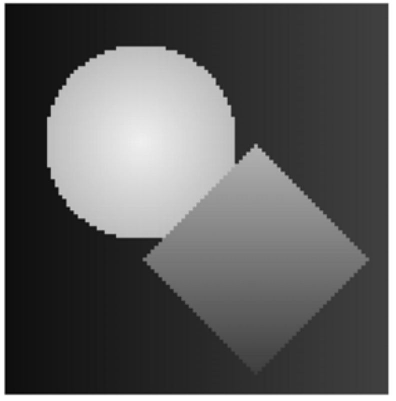
P-Smooth V_{pq}: Generalized Expansion

- can approximately find generalization of α-expansion
- that is find approximately (within a factor of 3) the best subset of pixels to switch to labels in $\{\alpha, \alpha+1, \ldots, \beta\}$

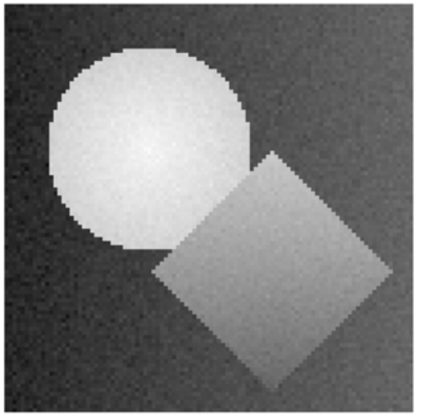




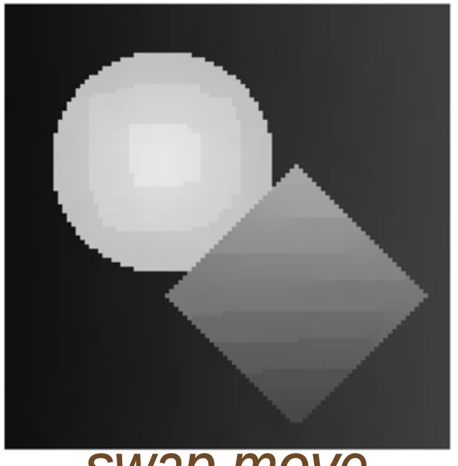
P-Smooth V_{pq}: Restoration with Truncated Quadratic V_{pq}



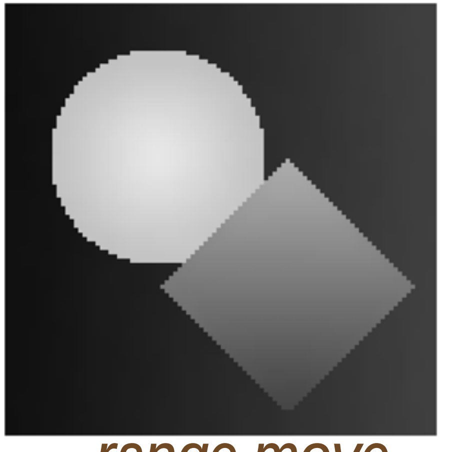
original image
energy: 419,076



original image
with Gaussian Noise



swap move
energy: 453,994
mean error: 1.35



range move
energy: 388,790
mean error: 0.82

P-Smooth V_{pq}: Results on Stereo

- Middlebury Dataset [Scharstein & Szeliski' IJCV2001]
- Truncated Quadratic V_{pq}

	Tsukuba	Venus	Teddy	Cones
Our Algorithm	1,758,136	2,671,875	6,058,678	7,647,529
Swap	1,804,548	2,702,371	6,099,656	7,706,717
Expansion	1,765,386	2,690,970	6,124,697	7,742,709

energies

	Tsukuba	Venus	Teddy	Cones
Our Algorithm	6.7	3.25	15.1	6.79
Swap	7.47	4.04	15.8	7.64
Expansion	7.14	4.19	16.0	7.81

errors

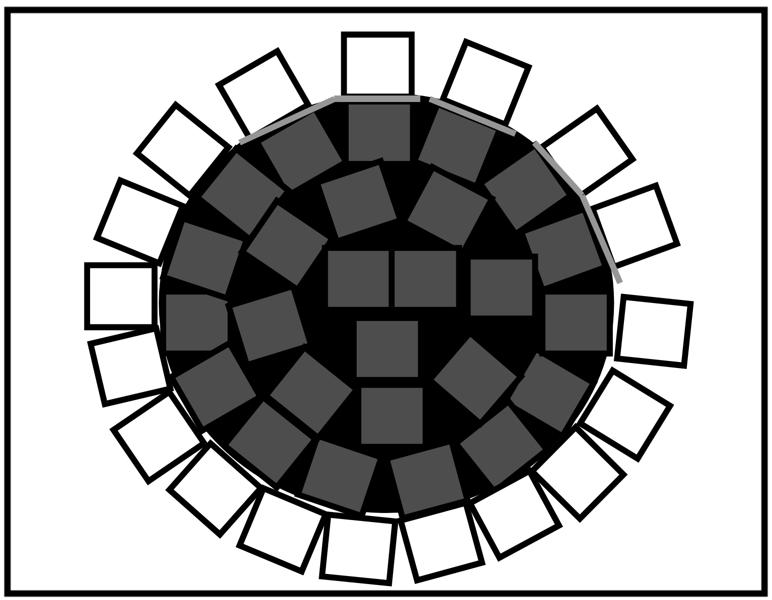
P-Smooth V_{pq}: Results on Stereo

swap

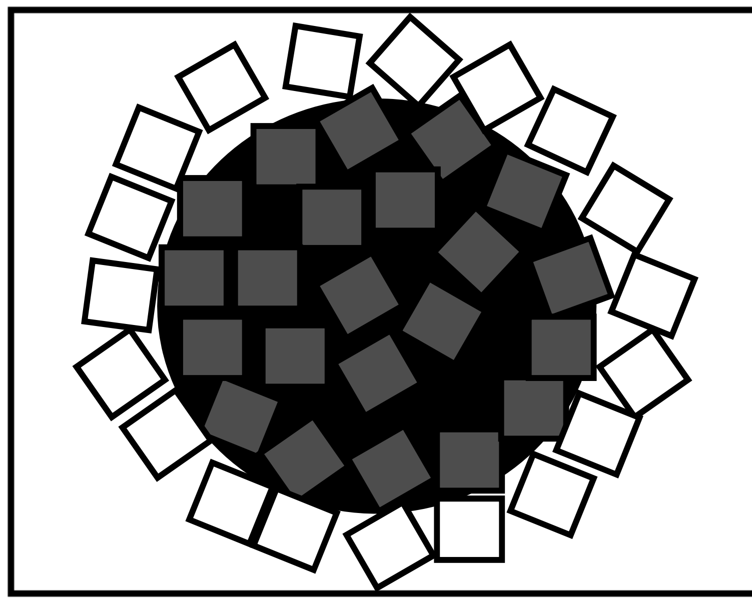
range

P-Smooth V_{pq} *Results*: Simulating Classic Mosaics

- Given a photograph or painting, output a "mosaic" image
  - Non-photorealistic rendering

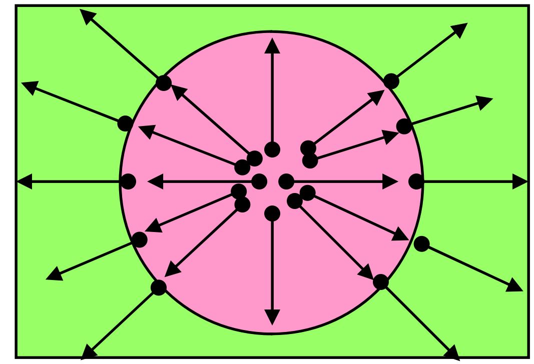


Good Tiling



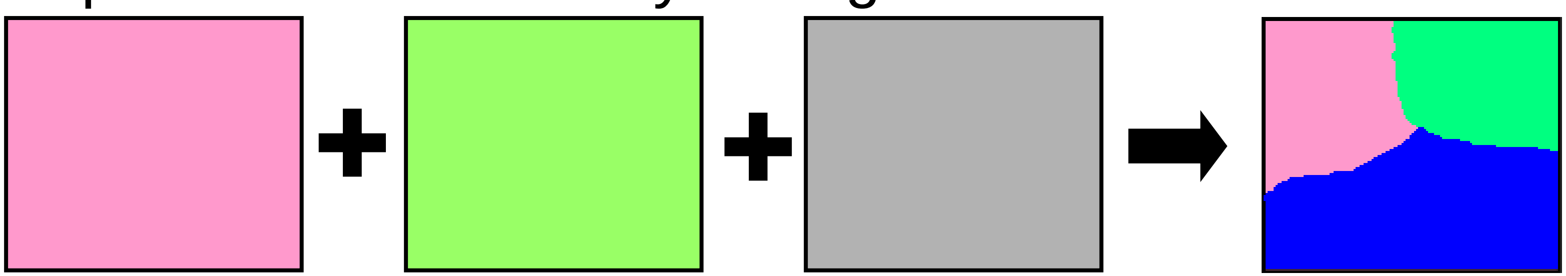
Bad Tiling

P-Smooth V_{pq} *Results*: Simulating Classic Mosaics



- Step 1: Generate tile orientations
  - NP-hard, can be done with range moves
- Step 2: Build candidate mosaic layers

- Step 3: Stitch mosaic layers together

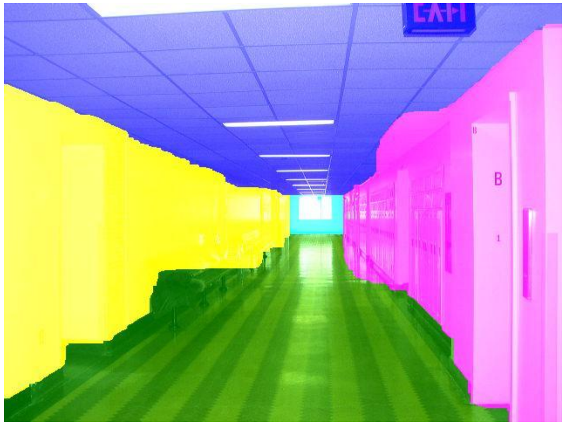


P-Smooth V_{pq} *Results*: Simulating Classic Mosaics

P-Smooth V_{pq} *Results*: Simulating Classic Mosaics

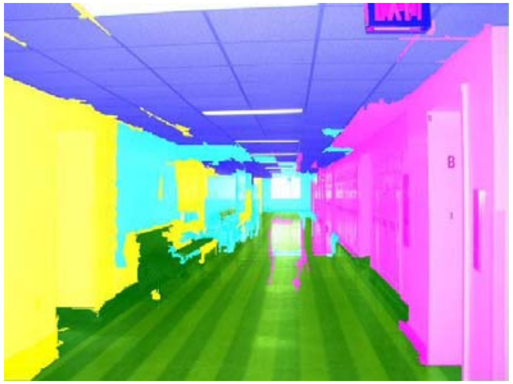
Order-preserving V_{pq}

- In some applications, in addition to smoothness, V_{pq} can be used to express certain ordering constraints

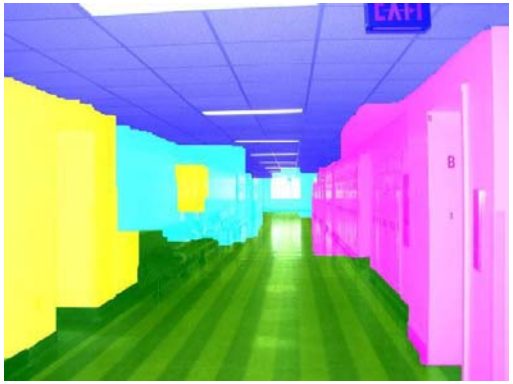


- Geometric class labeling problem [Hoem et.al.]
- Labels are “left”, “right”, “top”, “bottom”, “center”
- For example, V_{pq} (“left”,”right”) = ∞ if pixel p is to the right of pixel q

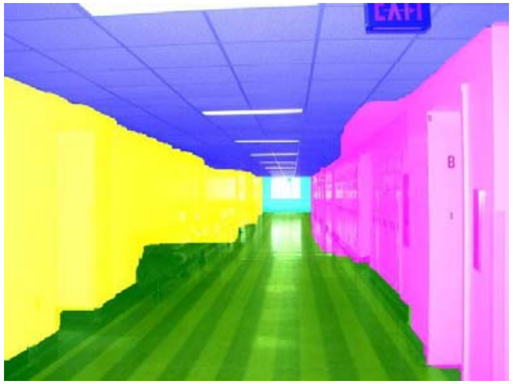
Order-preserving V_{pq}



independent labelling, i.e. only data term is used



V_{pq} for smoothness only

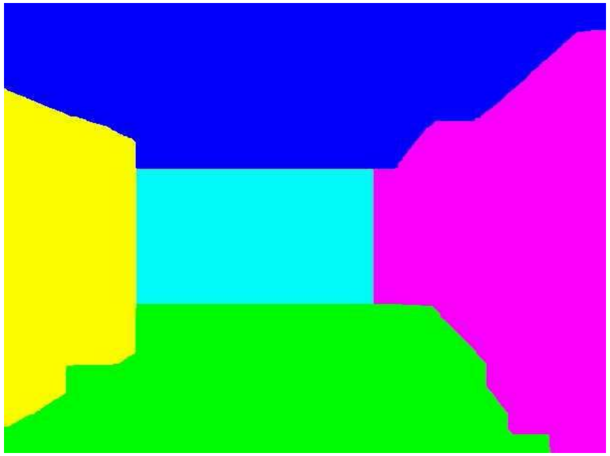


V_{pq} for smoothness and order preservation

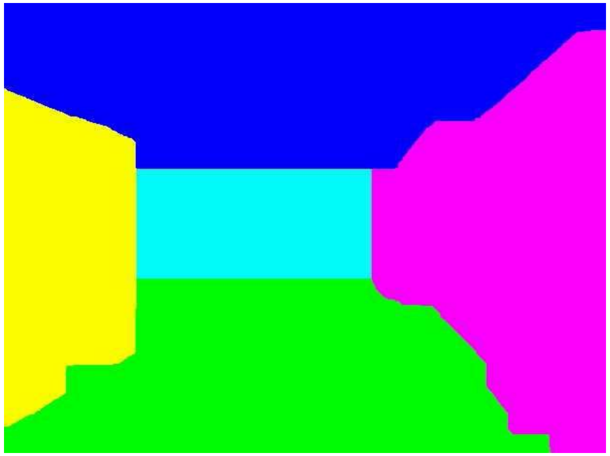
Order-preserving V_{pq}

- We use order-preserving V_{pq} added on top of the Potts model
  - $V_{pq}(\alpha,\alpha) = 0$
  - $V_{pq}(\alpha,\beta) = \infty$ if assigning α to p and β to q violates ordering constraints
  - Otherwise, $V_{pq}(\alpha,\beta) = w_{pq}$
- Expansion algorithm gets stuck in a local minimum easier when order-preserving V_{pq} are present
- In fact, when order-preserving V_{pq} are added to the Potts model, the bound of 2 no longer holds

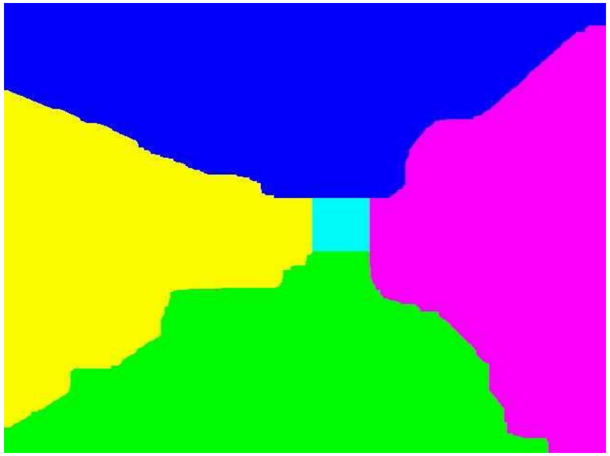
Order-preserving V_{pq}



expansion after one iteration



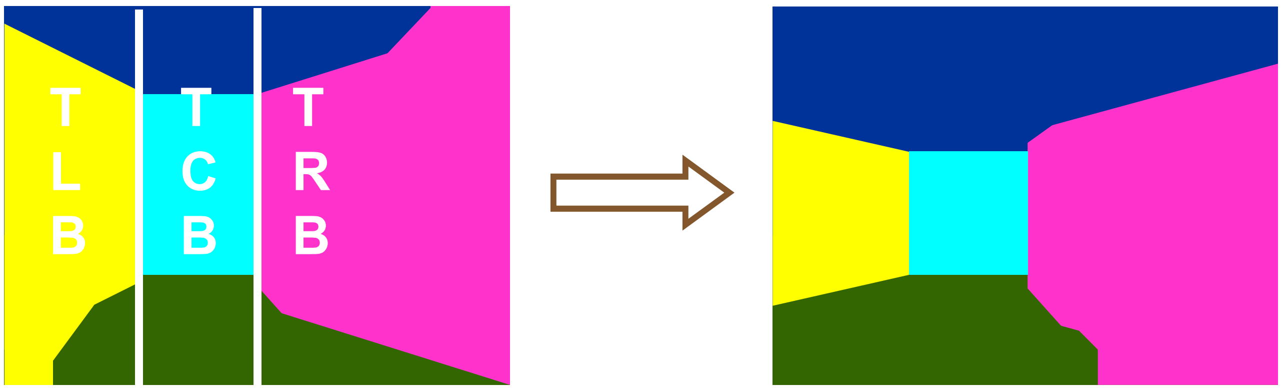
expansion after at convergence



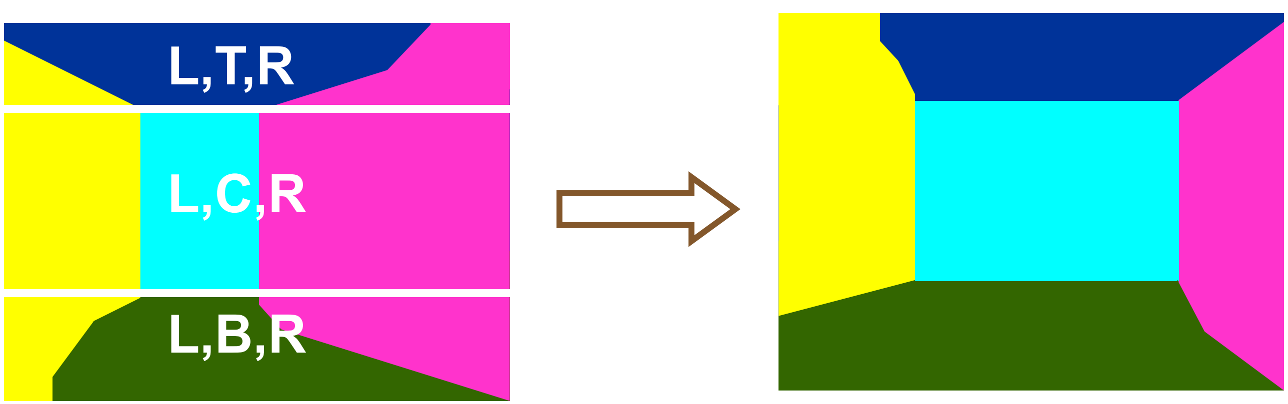
lower energy configuration

Order-preserving V_{pq}

- 2 types of order-preserving moves: "vertical" and "horizontal"
  - horizontal move, width of the "center" is preserved

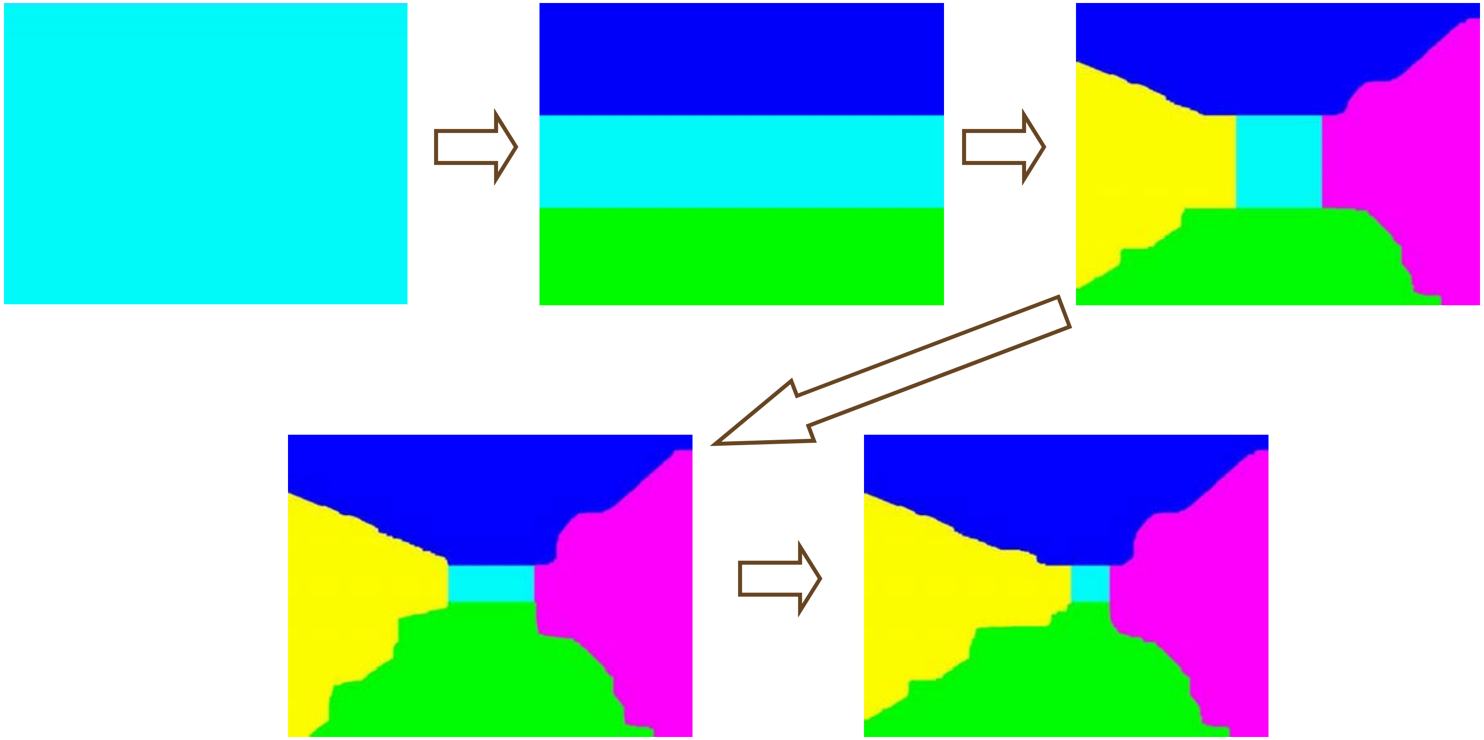


  - vertical move, height of the "center" is preserved

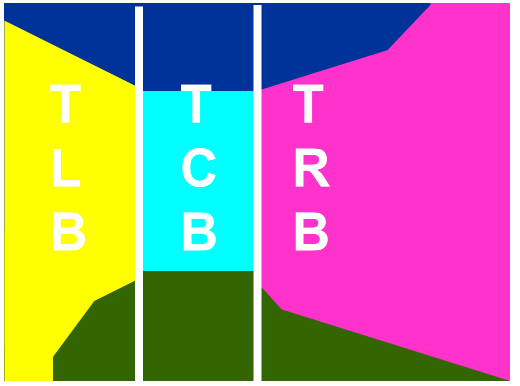


Order-preserving V_{pq}

- Vertical and horizontal moves are alternated until convergence
- Start with labeling “all center”

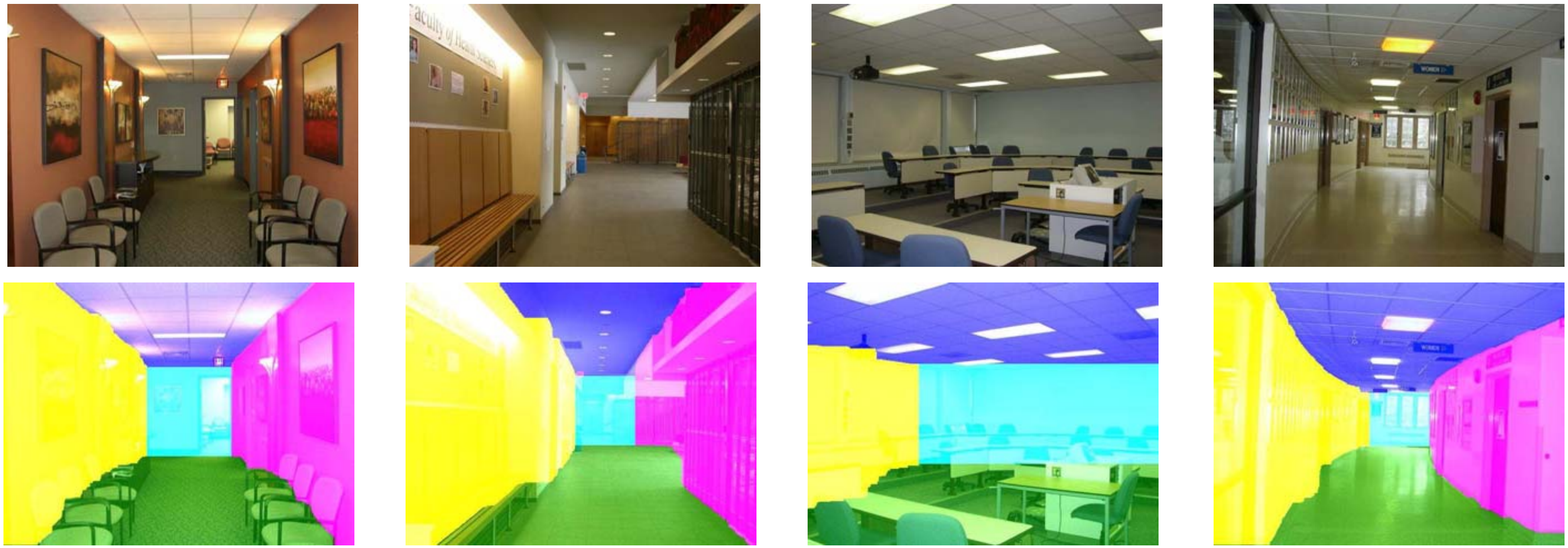


Order-preserving V_{pq}: Horizontal Move



- All pixel participate in a move
- Each pixel has a choice of 3 labels, including its old label
- Construction is based on Schlesinger and Flach'2006
  - sort of tedious to check
- Ishikawa's construction does not work since V_{pq}'s are not symmetric
- Same comments apply to the vertical move

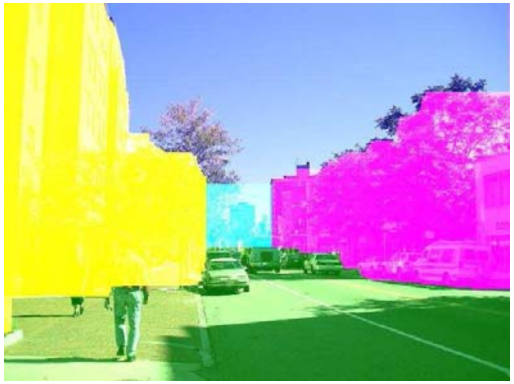
Order-preserving V_{pq}: Results

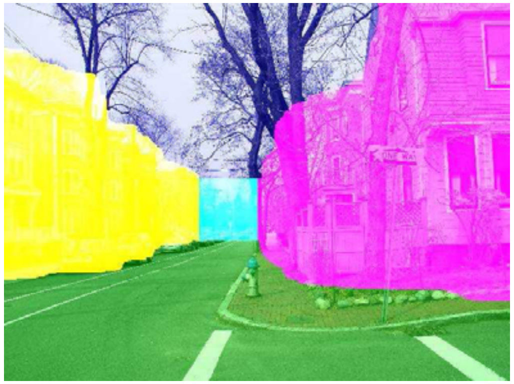


- On 300 images, the energy, on average is about 30% smaller with order-preserving moves, compared to alpha-expansion

Order-preserving V_{pq}: Results





Order-preserving V_{pq}: Results

- You can use order-preserving moves to enforce simple geometric shape priors, such as “rectangle” or a “sort of trapezoid”

Order-preserving V_{pq}: Results

- You can use order-preserving moves to enforce simple geometric shape priors, such as “rectangle” or a “sort of trapezoid”