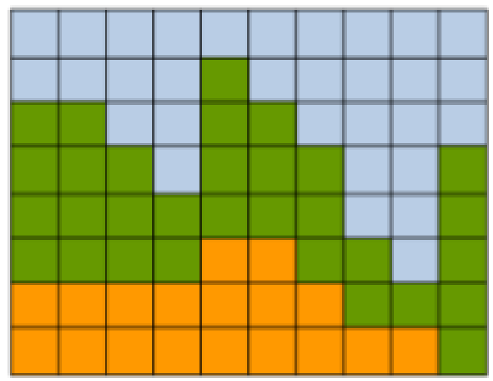
Dynamic Programming for General Pairwise MRFs (4 papers from 2010 – 2012)



Petar Ivanov

Graphics and Media Lab

Moscow State University

Feb 2013

Problem statement

Given a pairwise Markov Random Field with energy **E**:

$$E(f) = \sum_{k=1}^{n} U_k(s_k) + \sum_{k=1}^{n-1} H_k(s_k, s_{k+1})$$

$$\begin{aligned} U_k(s_k) &= \sum_{p \in P_k} D_p(f_p(s_k)) + \\ & \sum_{pq \in N_k} V_{pq}(f_p(s_k), f_q(s_k)) \\ H_k(s_k, s_{k+1}) &= \sum_{pq \in N_{k,k+1}} V_{pq}(f_p(s_k), f_q(s_{k+1})) \end{aligned}$$

To optimize **E** by **f(P)**

Felzenszwalb, Veksler – "*Tiered Scene Labeling with Dynamic Programming*", CVPR 2010

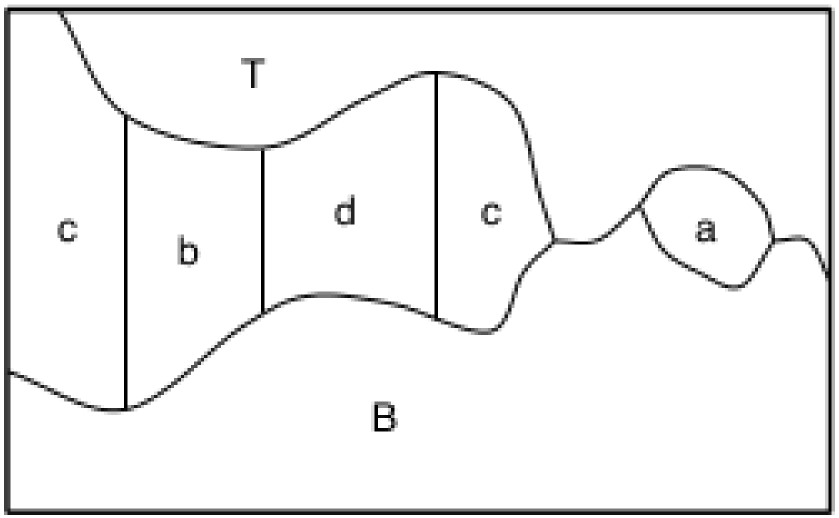


Figure 1. A *tiered* labeling. The labeling is defined by two horizontal curves α and β that partition the image into a top, bottom and middle region. The middle region is subpartitioned by vertical boundaries. The top region is labeled T, the bottom region is labeled B and the middle regions take labels from M.

Each column has its own label (+regions T and B)

Used for:
- Foreground/background
- Sky/ground/tree

$O(mn^2K^2)$ after a nice $O(n^2)$ optimization

where we have **m** columns, **n** rows, **K** labels

...results

Note: there may be at most
1 different label vertically (+Top & Bottom)

Figure 6. Some results on the dataset from [9]. Top row: original images, second row: confidence only results [9], last row: our results.

9.4sec on 300x250, K = 5

Vineet, Warrell, Torr – "*A Tiered Move-making Algorithm for General Pairwise MRFs*", CVPR 2012

iterative algorithm (slow but of high quality):

current labeling

→ DP: K-label optimal tiered move (region choice)

→ next labeling

$O(mn^2K^2)$ on iteration

...results

Note: there may be many
different labels vertically

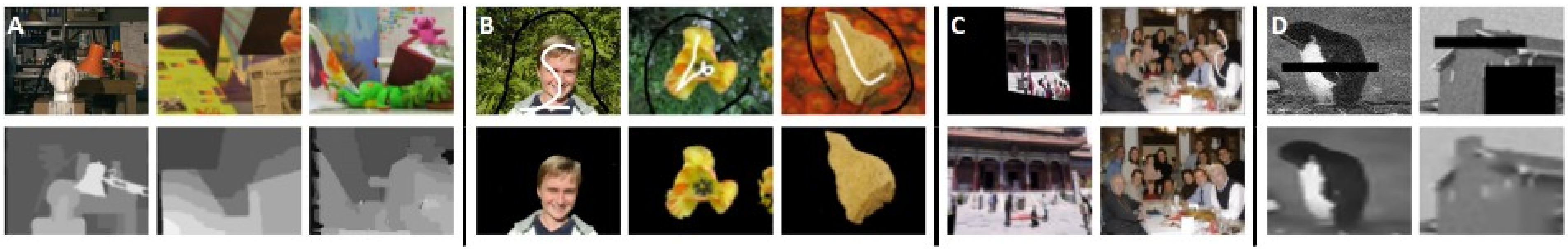
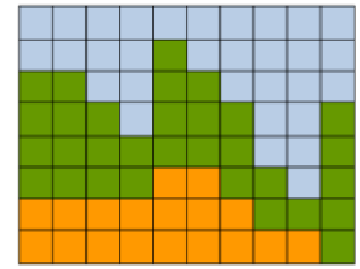


Figure 4. Shown are the input (first row) and output (second row) images of our tiered move making algorithm corresponding to different benchmark problems. (From left to right) **Stereo (A)**: tsukuba (192x144, 8 disparities), venus (217x192, 10 disparities) and teddy (112x93, 15 disparities) images; **Segmentation (B)**: person (600x450), flower (600x450) and sponge (640x480) images; **Image stitching (C)**: pano (178x120, 7 labels), and family (376x283, 5 labels); **Image denoising (D)**: penguin (30x44, 256 labels), and family (32x32, 256 labels).

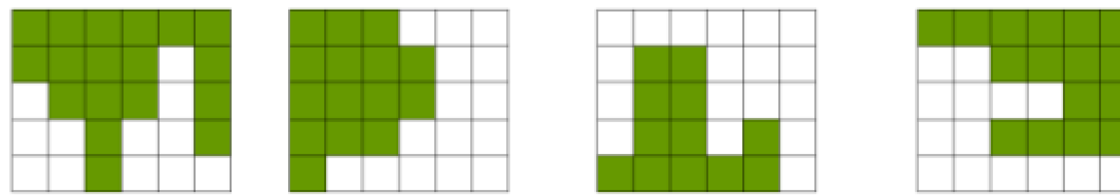
10-15 times slower than α-expansion.

Olga Veksler – "*Dynamic Programming for Approximate Expansion Algorithm*", CVPR 2012

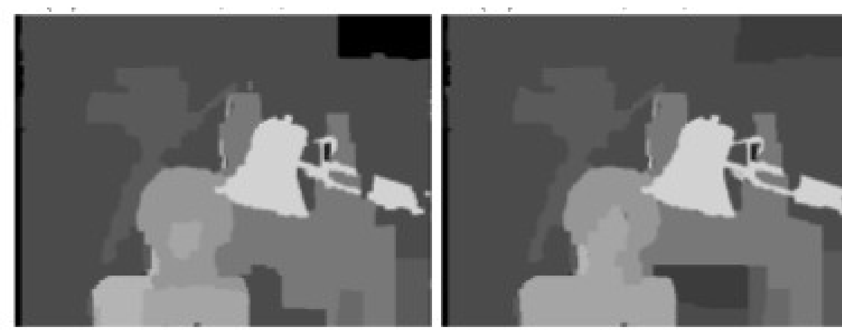
- Single green label (+Top & Bottom):



- Simple regions:



- Some depth map results:

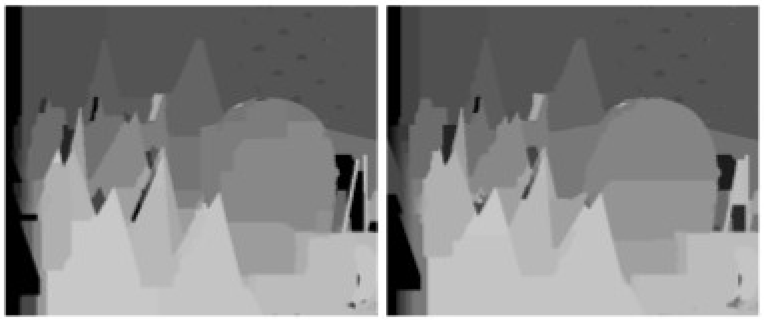


(e) E=350,958, 1.1 sec. (f) E = 346,085, 2.2 sec.

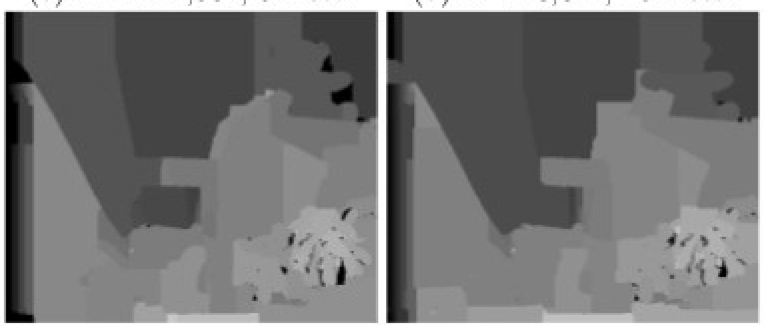
- Great complexity: O(mn) on iteration

...but the problem of iterating remains

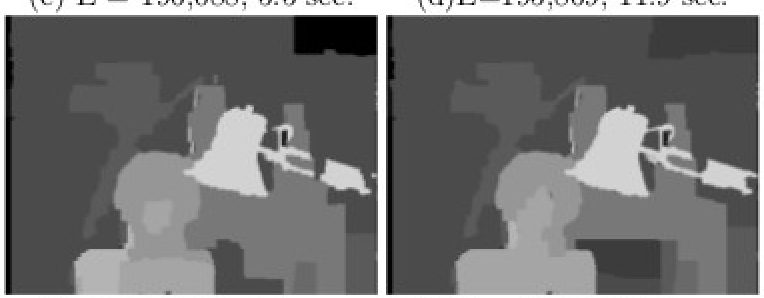
...results



(a) E = 221,884, 5.2 sec. (b) E=223,314, 10.2 sec.



(c) E = 196,088, 5.5 sec. (d)E=195,869, 11.9 sec.

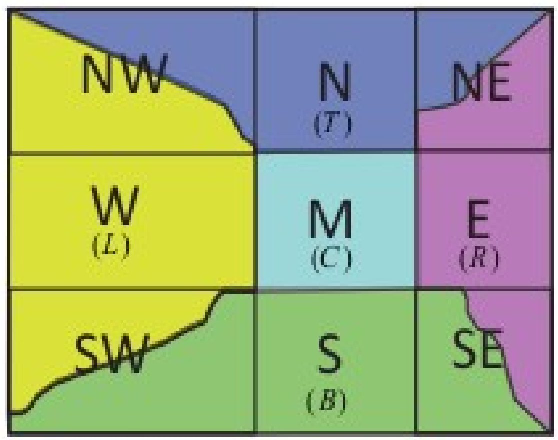


(e) E=350,958, 1.1 sec. (f) E = 346,085, 2.2 sec.

Left: the new algorithm (Veksler, 2012)
Right: Some max-flow algo

Note: The new algorithm is fast but not very accurate so the author doesn't compare it with others

Bai, Song, Veksler, Wu – "*Fast Dynamic Programming for Labeling Problems with Ordering Constraints*", CVPR 2012



N, W, E, S – integral images
NW, NE, SW, SE – DP
M – the best of O(n^4) rectangles is chosen
by O(n^3) queries for O(1)
Overall complexity O(n^3)

...results

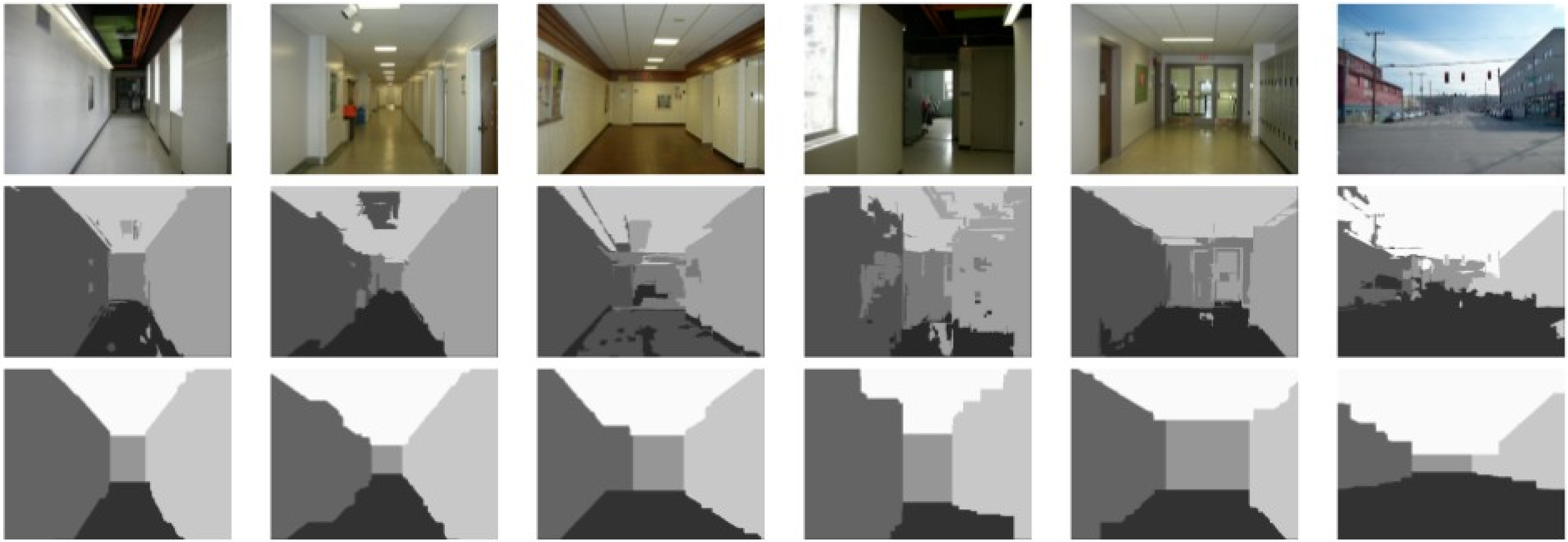


Figure 3. Some labeling results. Top row: original images; second row: SVM classifier results using data term only; last row: our results

2.2 seconds on 320 × 240 images