Combinatorial Optimization in Computer Vision
Lecture: F. R. Schmidt and C. Domokos
Exercises: T. Möllenhoff and T. Windheuser
Winter Semester 2015/2016

Computer Vision Group
Institut für Informatik
Technische Universität München

Weekly Exercises 5

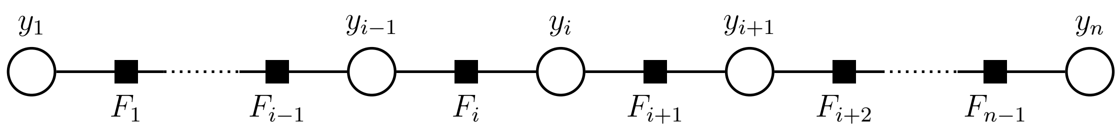
Room: 02.09.023

Tuesday, 24.11.2015, 14:15-15:45

Submission deadline: Tuesday, 24.11.2015, 11:15 , Room 02.09.023

Probabilistic Inference (5 Points)

Exercise 1 (Inference on chains, 2 Points)**.** Consider the following factor graph, which is a chain:



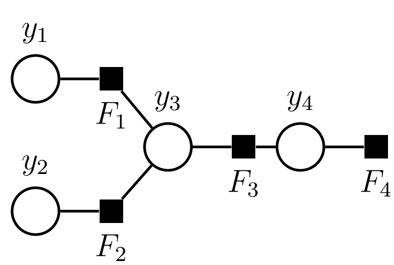
The joint distribution can be written in the form

$$p(\mathbf{y}) = \frac{1}{Z} F_1(y_1, y_2) F_2(y_2, y_3) \cdot \ldots \cdot F_{n-1}(y_{n-1}, y_n),$$

where $Z = \sum_{\mathbf{y}} \prod_{i=1}^{n-1} F_i(y_i, y_{i+1})$ denotes the partition function. Show that the marginal distribution $p(y_i)$ decomposes into the product of two factors:

$$p(y_i) = \frac{1}{Z} r_{F_i \to y_i}(y_i) r_{F_{i+1} \to y_i}(y_i).$$

Exercise 2 (Max-sum algorithm, 3 Points)**.** Execute the max-sum algorithm to find a maximizing configuration $\mathbf{y} \in \{0, 1, 2\}^4$ of the following factor graph:



The factors are defined through the following:

$$\begin{aligned} &F_x(y_i, y_j) = \exp\left(-(|y_i - y_j| + (c_x - y_i)^2)\right), x \in \{1, 2, 3\}, \\ &c_1 = 0, c_2 = 1, c_3 = 1, c_4 = 2, \\ &F_4(y_4) = \exp(-(c_4 - y_4)^2). \end{aligned} \tag{1}$$

Show the intermediate steps in detail. Pick y_4 as the root node.

Roof Duality (5 Points)

Exercise 3 (Roof duality, 5 Points)**.** Consider the following pseudo-Boolean energy function $f : \mathbb{B}^5 \to \mathbb{R}$:

$$f(x_1, \ldots, x_5) = 10 - 4x_1 - 4x_3 - 2x_4 + 4x_1x_2 - 2x_2x_3 + 4x_3x_4 - 2x_4x_5.$$

a) Show that f is not submodular.

b) Find the global minimizer $\widehat{x} \in \mathbb{B}^5$ of f using roof duality.

c) Show that f is submodular with respect to x_1, $\bar{x}_2$, $\bar{x}_3$, x_4, x_5.

Programming (2 weeks time, 10 Points)

Presentation of the programming exercise will be on Tuesday, December 1st.

Exercise 4 (Texture Denoising, 10 Points)**.**
Let $\mathcal{I} : \Omega \to [0, 1]$, where $\Omega = \{0, \ldots, W-1\} \times \{0, \ldots, H-1\}, W, H \in \mathbb{N}$, be a binary image with additive Gaussian noise. We would like to recover the original binary image $\mathcal{I}_0$ by removing the noise.

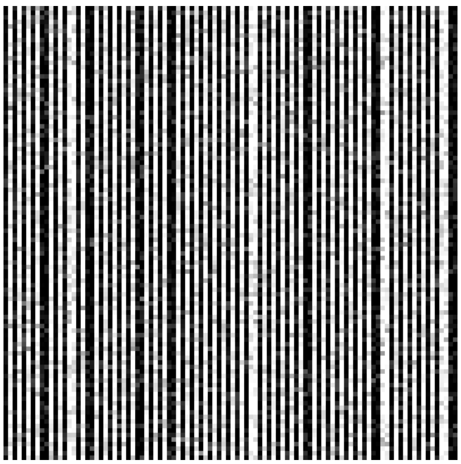
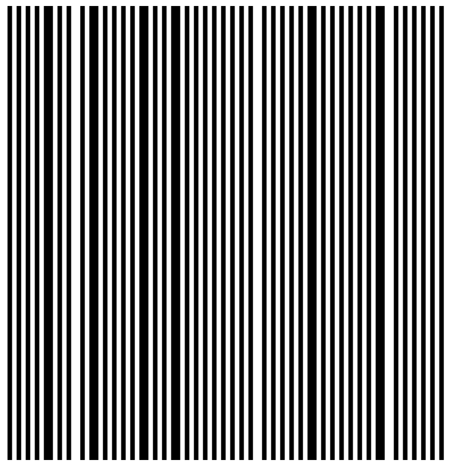


image $\mathcal{I}$ with noise


original image $\mathcal{I}_0$

Fortunately, we know that the original image contains mostly vertical stripes and only black or white pixels, i.e. $\mathcal{I}_0 : \Omega \to \mathbb{B}$. Therefore we can describe the denoised image as the optimizer of the following energy function:

$$E(X) = \lambda \sum_{i,j} f_{i,j}(X(i,j)) + \sum_{i,j} c_{\mathrm{h}}(X(i,j), X(i+1,j)) + c_{\mathrm{v}}(X(i,j), X(i,j+1)), \tag{2}$$

where $X : \Omega \to \mathbb{B}$ is a binary image and $\lambda > 0$ is a positive scalar parameter. The functions $f_{i,j} : \mathbb{B} \to \mathbb{R}$ encode the data given by image $\mathcal{I}$ and are defined by

$$f_{i,j}(x) = (\mathcal{I}(i,j) - x)^2. \tag{3}$$

The pairwise regularizers $c_{\mathrm{h}}, c_{\mathrm{v}} : \mathbb{B} \times \mathbb{B} \to \mathbb{R}$ encode the knowledge about the stripe pattern of the original image. The horizontal term c_{h} prefers neighboring pixels that have different intensities:

$$c_{\mathrm{h}}(x_1, x_2) = 1 - |x_1 - x_2|.$$

The vertical term c_{v} prefers neighboring pixels that have the same intensities:

$$c_{\mathrm{v}}(x_1, x_2) = |x_1 - x_2|.$$

Which energy is submodular? c_{h} or c_{v}?

Write a program that tries to find the optimizer of energy function (2). Since half of the pairwise terms are non-submodular try the following strategies:

a) Remove the non-submodular terms from energy function (2) and use the basic graph-cut algorithm.

b) Use QPBO to improve the result of a) a partial labeling.

You can use the code and the images inside `05_supp.zip` from the lecture website as a starting point. The file `texturedenoising.cpp` already includes the code to load the images and textures and to compute the pairwise energy functions.

The exercise uses a simplified version of the model in Cremers and Grady [1] `https://vision.in.tum.de/_media/spezial/bib/cremers_grady_eccv06.pdf`.

References

[1] Cremers, D., Grady, L.: Statistical priors for combinatorial optimization: efficient solutions via Graph Cuts. In Leonardis, A., Bischof, H., Pinz, A., eds.: European Conference on Computer Vision (ECCV). Volume 3953 of LNCS., Graz, Austria, Springer (2006) 263–274