Optimizing Binary MRFs with Higher Order Cliques

Asem M. Ali[1], Aly A. Farag[1], and Georgy L. Gimel'farb[2]

[1] Computer Vision and Image Processing Laboratory, University of Louisville, USA
{asem,farag}@cvip.uofl.edu
[2] Department of Computer Science, The University of Auckland, New Zealand
g.gimelfarb@auckland.ac.nz

Abstract. Widespread use of efficient and successful solutions of Computer Vision problems based on pairwise Markov Random Field (MRF) models raises a question: does any link exist between the pairwise and higher order MRFs such that the like solutions can be applied to the latter models? This work explores such a link for binary MRFs that allow us to represent Gibbs energy of signal interaction with a polynomial function. We show how a higher order polynomial can be efficiently transformed into a quadratic function. Then energy minimization tools for the pairwise MRF models can be easily applied to the higher order counterparts. Also, we propose a method to analytically estimate the potential parameter of the asymmetric Potts prior. The proposed framework demonstrates very promising experimental results of image segmentation and can be used to solve other Computer Vision problems.

1 Introduction

Recently, discrete optimizers based on e.g. graph cuts [1,2], loopy belief propagation [3,4], and tree reweighted message passing [5,6] became essential tools in Computer Vision. These tools solve many important Computer Vision problems including image segmentation, restoration, and matching, computational stereo, etc. (see [7] for more detail). Conventional framework for such problems is the search for Maximum A Posteriori (MAP) configurations in a Markov Random Field (MRF) model where the MAP problem is formulated as minimizing an interaction energy for the model. In this work, we focus only on binary MRFs that play an important role in Computer Vision since Boykov et al. [1] proposed an approximate graph-cut algorithm for energy minimization with iterative expansion moves. The algorithm reduces the problem with multivalued variables to a sequence of subproblems with binary variables.

Most of the energy-based Computer Vision frameworks represent the MRF energy on an image lattice in terms of unary and pairwise clique potentials. However, this representation is insufficient for modeling rich statistics of natural scenes [8,9]. The latter require higher order clique potentials being capable to describe complex interactions between variables. Adding potentials for the

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part III, LNCS 5304, pp. 98–111, 2008.

higher order cliques could improve the image model [10,11]. However, optimization algorithms of these models have too high time complexity to be practicable. For example, a conventional approximate energy minimization framework with belief propagation (BP) is too computationally expensive for MRFs with higher order cliques, and Lan et al. [8] proposed approximations to make BP practical in these cases. However, the results are competitive with only simple local optimization based on gradient descent technique. Recently, Kohli et al. [12] proposed a generalized $\mathcal{P}^n$ family of clique potentials for the Potts MRF model and showed that optimal graph-cut moves for the family have polynomial time complexity. However, just as in the standard graph-cut approaches based on the α-expansion or $\alpha\beta$-swap, the energy terms for this family have to be submodular.

Instead of developing efficient energy minimization techniques for higher order MRFs, this paper chooses an alternative strategy of reusing well established approaches that have been successful for the pairwise models and proposes an efficient transformation of an energy function for a higher order MRF into a quadratic function. It is worth to mention that Kolmogorov and Rother [13] referred to a similar transformation [14] in their future work. In order to reduce an energy function for a higher order MRF into a quadratic function, first we convert the potential energy for higher order cliques into a polynomial form and show explicitly when this form can be graph representable and how this graph can be constructed for such an energy. Then we reduce the higher-order polynomial to a specific quadratic one. The later may have submodular and/or nonsubmodular terms, and few approaches have been proposed to minimize such functions. For instance, Rother et al. [15] truncate nonsubmodular terms in order to obtain an approximate submodular function to be minimized. This truncation leads to a reasonable solution when the number of the nonsubmodular terms is small. Recently, Rother et al. [16] proposed an efficient optimization algorithm for nonsubmodular binary MRFs, called the extended roof duality. However, it is limited to only quadratic energy functions. Our proposal expands notably the class of the nonsubmodular MRFs that can be minimized using this algorithm. Below, we use it to minimize the proposed quadratic version of the higher order energy. To illustrate potentialities of the higher order MRFs in modeling complex scenes, the performance of our approach has been assessed experimentally in application to image segmentation. The potential parameter of the asymmetric Potts prior that has been used in the experiments is analytically estimated. The obtained results confirm that the proposed optimized MRF framework can be efficiently used in practice.

2 Preliminaries

The goal image labeling $\mathbf{x}$ in the MAP approach is a realization of a Markov-Gibbs random field (MGRF) $\mathbf{X}$ defined over an arithmetic 2D lattice $\mathcal{V} = \{1, 2, \cdots, n\}$ with a neighborhood system $\mathcal{N}$. The system explicitly specifies neighboring random variables that have spatial interaction. Let $\mathcal{X}$ denote a set of all possible configurations of an MGRF. Then the probability of a particular

configuration $\mathbf{x} \in \mathcal{X}$ is given by a Gibbs probability distribution: $P(\mathbf{x}) = \frac{1}{Z}e^{-\mathcal{E}(\mathbf{x})}$, where Z denotes a normalizing constant (the partition function) and $\mathcal{E}(\mathbf{x})$ is the Gibbs energy function. The latter sums Gibbs potentials supported by cliques of the interaction graph. As defined in [17], a clique is a set of sites $i \in \mathcal{V}$ (e.g. pixels in an image) such that all pairs of sites are mutual neighbors in accord with $\mathcal{N}$. The maximal clique size determines the Gibbs energy order.

Energy functions for an MGRF with only unary and pairwise cliques can be written in the following form:

$$\mathcal{E}(\mathbf{x}) = \sum_{i \in \mathcal{V}} \varphi(x_i) + \sum_{\{i,j\} \in \mathcal{N}} \varphi(x_i, x_j) \ , \tag{1}$$

where $\varphi(.)$ denotes the clique potential. The energy minimum $\mathcal{E}(\mathbf{x}^*) = \min_{\mathbf{x}} \mathcal{E}(\mathbf{x})$ corresponds to the MAP labeling $\mathbf{x}^*$. For a binary MGRF, the set of labels consists of two values, $\mathcal{B} = \{0, 1\}$, each variable x_i is a binary variable, and the energy function in (1) can be written in a quadratic form:

$$\mathcal{E}(\mathbf{x}) = a_{\{\varnothing\}} + \sum_{i \in \mathcal{V}} a_{\{i\}} x_i + \sum_{\{i,j\} \in \mathcal{N}} a_{\{i,j\}} x_i x_j \ , \tag{2}$$

where the coefficients $a_{\{\varnothing\}}$, $a_{\{i\}}$ and $a_{\{i,j\}}$ are real numbers depending on $\varphi(0)$, $\varphi(1)$, $\ldots$, $\varphi(1,1)$ in a straightforward way.

Generally, let $\mathcal{B}^n = \{(x_1, x_2, \cdots, x_n) |\ \ x_i \in \mathcal{B}; \forall\, i = 1, \cdots, n\}$, and let $E(\mathbf{x}) = E(x_1, x_2, \cdots, x_n)$ be a real valued polynomial function of n bivalent variables and real coefficients defining a Gibbs energy with higher order potentials (in contrast to the above quadratic function $\mathcal{E}$). Such function $E(\mathbf{x})$ is called a pseudo-Boolean function [18] and can be uniquely represented as a multi-linear polynomial [14] as follows:

$$E(\mathbf{x}) = \sum_{S \subseteq \mathcal{V}} a_S \prod_{i \in S} x_i \ , \tag{3}$$

where a_S are non-zero real numbers, and the product over the empty set is 1 by definition.

3 Polynomial Forms of Clique Potentials

To be transformed into a quadratic energy, the higher order energy function should be represented in a multi-linear polynomial form (3). Hereafter, we will consider how the clique potentials can be represented in a polynomial form. An unary term has an obvious polynomial form $\varphi_{x_i} = \varphi(x_i) = (\varphi_1 - \varphi_0)x_i + \varphi_0$ where φ_1 and φ_0 are the potential values for the labels 1 and 0 for the variable $x_i \in \mathcal{B}$, respectively. A clique of size $\mathtt{k}$ has a potential energy $\varphi(x_i, x_j, \cdots, x_{\mathtt{k}})$, where $\mathtt{k} \leq n$. The coefficients of the polynomial that represents the energy of a clique of size $\mathtt{k}$ can be estimated using Algorithm 1 below. It is worth mentioning that many works tried to compute these coefficients, such as the formula proposed by Freedman and Drineas [19], an observation in [20], and Proposition 2 in [14]. However, these works are somewhat complicated and are not explicit, and we believe Algorithm 1 is much easier for implementation.

Algorithm 1. Computing coefficients $a_{\mathcal{S}}$ of energy (3). $\mathcal{H}$ denotes the set of pixels in a clique which its potential represented by (3), $\mathcal{S} \subseteq \mathcal{H}$, $\mathcal{Z} = \mathcal{H} - \mathcal{S}$, and $\mathcal{W}$ is a labeling of the pixels in $\mathcal{Z}$. Initially, $\mathcal{W} = \oslash$.

1. $a_{\mathcal{S}}$
2. **if** $(\mathcal{S} = \oslash)$ Return $\varphi(x_{\mathcal{H}} = 0)$ **end if**
3. $\mathcal{Z} = \mathcal{H} - \mathcal{S}$
4. **if** $(\mathcal{W} = \oslash)$ $\mathcal{W} = \{w_i = 0 \mid i = 1, 2, \ldots, |\mathcal{Z}|\}$ **end if**
5. **if** $(|\mathcal{S}| = 1)$ **then**
6. Return $\varphi(x_{\mathcal{S}} = 1, x_{\mathcal{Z}} = \mathcal{W}) - \varphi(x_{\mathcal{S}} = 0, x_{\mathcal{Z}} = \mathcal{W})$
7. **else**
8. Select first element $i \in \mathcal{S}$. Then: $\mathcal{S} \leftarrow \mathcal{S} - \{i\}$, $\mathcal{W}_1 \leftarrow \{1\} + \mathcal{W}$ and $\mathcal{W}_0 \leftarrow \{0\} + \mathcal{W}$
9. Let $\mathcal{W} = \mathcal{W}_1$ and compute $t_1 = a_{\mathcal{S}}$. Then, let $\mathcal{W} = \mathcal{W}_0$ and compute $t_0 = a_{\mathcal{S}}$
10. Return $t_1 - t_0$
11. **end if**

To verify the correctness of Algorithm 1, let us use it to estimate the polynomial coefficients of the potential $\varphi(x_i, x_j, x_\ell, x_k)$; $x_i, x_j, x_\ell, x_k \in \mathcal{B}$, for a clique of size 4. This polynomial can be represented as follows[1]:

$$\begin{aligned}\varphi_{x_i x_j x_\ell x_k} &= a_{ij\ell k} x_i x_j x_\ell x_k + a_{ij\ell} x_i x_j x_\ell + a_{ijk} x_i x_j x_k + a_{i\ell k} x_i x_\ell x_k + a_{j\ell k} x_j x_\ell x_k \\ &+ a_{ij} x_i x_j + a_{i\ell} x_i x_\ell + a_{ik} x_i x_k + a_{j\ell} x_j x_\ell + a_{jk} x_j x_k + a_{\ell k} x_\ell x_k \\ &+ a_i x_i + a_j x_j + a_\ell x_\ell + a_k x_k + a_\oslash \ , \end{aligned} \tag{4}$$

where the coefficients can be computed using Algorithm 1. Examples of these coefficients' computations are (here $\mathcal{H} = \{i, j, \ell, k\}$):

$$a_\oslash = \varphi_{0000}, \quad a_i = \varphi_{1000} - \varphi_{0000}, \quad a_{ij} \begin{cases} +a_j \ (\text{with } \mathcal{W} = \{100\}) & = \varphi_{1100} - \varphi_{1000} \\ -a_j \ (\text{with } \mathcal{W} = \{000\}) & + \varphi_{0000} - \varphi_{0100} \end{cases},$$

$$a_{ij\ell} \begin{cases} +a_{j\ell} \ (\text{with } \mathcal{W} = \{10\}) \begin{cases} +a_\ell \ (\text{with } \mathcal{W} = \{110\}) & = \varphi_{1110} - \varphi_{1100} \\ -a_\ell \ (\text{with } \mathcal{W} = \{100\}) & + \varphi_{1000} - \varphi_{1010} \end{cases} \\ -a_{j\ell} \ (\text{with } \mathcal{W} = \{00\}) \begin{cases} -a_\ell \ (\text{with } \mathcal{W} = \{010\}) & + \varphi_{0100} - \varphi_{0110} \\ +a_\ell \ (\text{with } \mathcal{W} = \{000\}) & + \varphi_{0010} - \varphi_{0000} \end{cases} \end{cases}, \text{ and}$$

$$a_{ij\ell k} \begin{cases} +a_{j\ell k} \ (\mathcal{W} = \{1\}) \begin{cases} +a_{\ell k} \ (\mathcal{W} = \{11\}) \begin{cases} +a_k \ (\mathcal{W} = \{111\}) & = \varphi_{1111} - \varphi_{1110} \\ -a_k \ (\mathcal{W} = \{110\}) & + \varphi_{1100} - \varphi_{1101} \end{cases} \\ -a_{\ell k} \ (\mathcal{W} = \{10\}) \begin{cases} -a_k \ (\mathcal{W} = \{101\}) & + \varphi_{1010} - \varphi_{1011} \\ +a_k \ (\mathcal{W} = \{100\}) & + \varphi_{1001} - \varphi_{1000} \end{cases} \end{cases} \\ -a_{j\ell k} \ (\mathcal{W} = \{0\}) \begin{cases} -a_{\ell k} \ (\mathcal{W} = \{01\}) \begin{cases} -a_k \ (\mathcal{W} = \{011\}) & + \varphi_{0110} - \varphi_{0111} \\ +a_k \ (\mathcal{W} = \{010\}) & + \varphi_{0101} - \varphi_{0100} \end{cases} \\ +a_{\ell k} \ (\mathcal{W} = \{00\}) \begin{cases} +a_k \ (\mathcal{W} = \{001\}) & + \varphi_{0011} - \varphi_{0010} \\ -a_k \ (\mathcal{W} = \{000\}) & + \varphi_{0000} - \varphi_{0001} \end{cases} \end{cases} \end{cases}. \tag{5}$$

In a similar way, the potential function $\varphi(x_i, x_j, x_\ell)$; $x_i, x_j, x_\ell \in \mathcal{B}$, of a 3^{rd} order clique can be represented as follows:

$$\begin{aligned}\varphi_{x_i x_j x_\ell} &= ((\varphi_{111} + \varphi_{100} - \varphi_{110} - \varphi_{101}) - (\varphi_{011} + \varphi_{000} - \varphi_{001} - \varphi_{010})) x_i x_j x_\ell \\ &+ (\varphi_{011} + \varphi_{000} - \varphi_{001} - \varphi_{010}) x_j x_\ell + (\varphi_{101} + \varphi_{000} - \varphi_{001} - \varphi_{100}) x_i x_\ell \\ &+ (\varphi_{110} + \varphi_{000} - \varphi_{100} - \varphi_{010}) x_i x_j + (\varphi_{010} - \varphi_{000}) x_j + (\varphi_{100} - \varphi_{000}) x_i \\ &+ (\varphi_{001} - \varphi_{000}) x_\ell + \varphi_{000} \ . \end{aligned} \tag{6}$$

[1] For brevity, hereafter, the notation is simplified: e.g. $a_{\{i,j,\ell,k\}}$ becomes $a_{ij\ell k}$ and $\varphi(x_i, x_j, x_\ell, x_k)$ becomes $\varphi_{x_i x_j x_\ell x_k}$.

Indeed, representing potentials in polynomial forms (e.g. (6)) has many advantages. It implies an algebraic proof of the Kolmogorov–Zabih's submodularity condition [2]. As we will explain in Section 5, the first(cubic) term in (6) will be reduced to be a quadratic term with the same coefficient. Thus, and according to a combinatorial optimization theorem [19], such energy can be minimized via graph cuts if and only if

$$\begin{gathered}(\varphi_{111}+\varphi_{100}-\varphi_{110}-\varphi_{101})-(\varphi_{011}+\varphi_{000}-\varphi_{001}-\varphi_{010})\leq 0,\\ \varphi_{011}+\varphi_{000}-\varphi_{001}-\varphi_{010}\leq 0,\quad \varphi_{101}+\varphi_{000}-\varphi_{001}-\varphi_{100}\leq 0,\\ \text{and}\qquad \varphi_{110}+\varphi_{000}-\varphi_{100}-\varphi_{010}\leq 0\quad .\end{gathered}\tag{7}$$

These inequalities represent all the projections of the $\varphi_{x_i x_j x_\ell}$ on 2 variables and follow Definition 1.

Definition 1. [Kolmogorov–Zabih; 2004] *A function of one binary variable is always submodular. A function $\varphi(x_i, x_j)$ from the family $\mathcal{F}^2$ is submodular if and only if $\varphi_{11}+\varphi_{00}\leq\varphi_{01}+\varphi_{10}$. A function from the family $\mathcal{F}^k$ is submodular if and only if all its projections on 2 variables are submodular.*

Moreover, the polynomial representation can be explicitly related to the corresponding graph. For example potential $\varphi(x_i, x_j)$; $x_i, x_j \in \mathcal{B}$, for a clique of size two can be generally represented as follows: $\varphi(x_i, x_j) = (\varphi_{11}+\varphi_{00}-\varphi_{01}-\varphi_{10})x_i x_j + (\varphi_{01}-\varphi_{00})x_j + (\varphi_{10}-\varphi_{00})x_i + \varphi_{00}$. This expression explicitly shows edges in the graph related to the polynomial coefficients. This construction is similar to what has been introduced in [2] but here each term in the previous expression excluding the constant, directly represents a part of the graph.

4 MGRF Parameter Estimation

The scaling parameter of a pairwise homogenous isotropic MGRF specifying the symmetric Potts prior can be analytically estimated similarly to [21]. However, we focus on asymmetric pairwise co-occurrences of the region labels. The asymmetric Potts model provides more chances to guarantee that the energy function is submodular. The Gibbs potential governing the asymmetric pairwise co-occurrences of the region labels is as follows:

$$\varphi(x_i, x_j) = \gamma\delta(x_i \neq x_j)\ ,\tag{8}$$

where γ is the model's parameter and the indicator function $\delta(\boldsymbol{C})$ equals 1 when the condition $\boldsymbol{C}$ is true and zero otherwise. So that the MGRF model of region maps is specified by the following Gibbs distribution:

$$P(\mathbf{x}) = \frac{1}{Z}\exp\Big(-\sum_{\{i,j\}\in\mathcal{N}}\varphi(x_i, x_j)\Big) = \frac{1}{Z}\exp\big(-\gamma|\mathcal{T}|f_{\text{neq}}(\mathbf{x})\big)\ .\tag{9}$$

Here, $\mathcal{T} = \{\{i,j\} : i,j \in \mathcal{V}; \{i,j\} \in \mathcal{N}\}$ is a family of the neighboring pixel pairs (second-order cliques) supporting the pairwise Gibbs potentials, $|\mathcal{T}|$ is the cardinality of the family, the partition function $Z = \sum_{\hat{\mathbf{x}}\in\mathcal{X}}\exp\big(-\gamma|\mathcal{T}|f_{\text{neq}}(\hat{\mathbf{x}})\big)$, and $f_{\text{neq}}(\mathbf{x})$ denotes the relative frequency of the non-equal label pairs over $\mathcal{T}$:

$$f_{\text{neq}}(\mathbf{x}) = \frac{1}{|\mathcal{T}|} \sum_{\{i,j\}\in\mathcal{T}} \delta(x_i \neq x_j) \ . \tag{10}$$

To completely identify the Potts model, its potential can be estimated for a given training label images $\mathbf{x}$ using a reasonably close first approximation of the maximum likelihood estimate (MLE) of γ. It is derived in accord with [21] from the specific log-likelihood $L(\mathbf{x}|\gamma) = \frac{1}{|\mathcal{V}|}\log P(\mathbf{x})$ that can be rewritten as:

$$L(\mathbf{x}|\gamma) = -\gamma\rho f_{\text{neq}}(\mathbf{x}) - \frac{1}{|\mathcal{V}|}\log \sum_{\hat{\mathbf{x}}\in\mathcal{X}} \exp\big(-\gamma|\mathcal{T}|f_{\text{neq}}(\hat{\mathbf{x}})\big) \ , \tag{11}$$

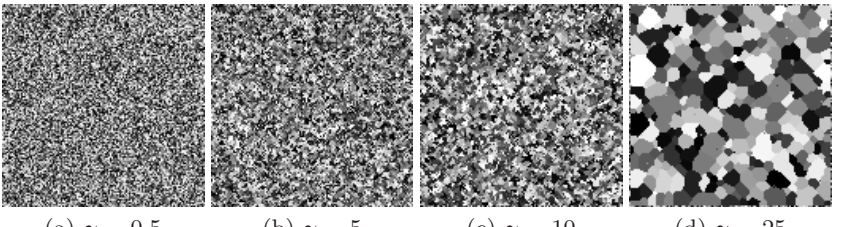
where $\rho = \frac{|\mathcal{T}|}{|\mathcal{V}|}$. The approximation is obtained by truncating the Taylor's series expansion of $L(\mathbf{x}|\gamma)$ to the first three terms in the close vicinity of the zero potential, $\gamma = 0$. Then, it is easily to show that the resulting approximate log likelihood is:

$$L(\mathbf{x}|\gamma) \approx -|\mathcal{V}|\log K + \rho\gamma\big(\frac{K-1}{K} - f_{\text{neq}}(\mathbf{x})\big) - \frac{1}{2}\gamma^2\rho\frac{K-1}{K^2} \ , \tag{12}$$

where, K is generally the number of labels (classes) in multi-label MRFs (in binary MRFs $K = 2$). For the approximate likelihood (12), let $\frac{dL(\mathbf{x}|\gamma)}{d\gamma} = 0$. The resulting approximate MLE of γ is:

$$\gamma^* = K\left(1 - \frac{K}{K-1}f_{\text{neq}}(\mathbf{x})\right) \ . \tag{13}$$

We tested the robustness of the obtained MLE using label images having been simulated with the known potential values. The simulated images were generated using the Gibbs sampler [22] and four variants of the asymmetric Potts model with 32 color labels (i.e. $K = 32$). Examples of the generated images of the size 128×128 are shown in Fig. 1. To get accurate statistics, 100 realizations are generated from each variant, and the proposed MLE (13) for the model parameter γ was computed for these data sets. The mean values and the variances of γ^* for the 100 realizations for each type are shown in Table 1.



(a) $\gamma = 0.5$ (b) $\gamma = 5$ (c) $\gamma = 10$ (d) $\gamma = 25$

Fig. 1. Samples of synthetic images of the size 128×128

Table 1. Accuracy of the proposed MLE in (13): its mean (variance) – for the 100 synthetic images of the size 128×128

Actual parameter γ	0.5	5	10	25
Our MLE γ^*	0.51 (0.03)	5.4 (0.05)	10.1 (0.06)	25.7 (0.11)

5 Energy Reduction – The Proposed Approach

We showed so far how MGRF parameters can be estimated and how the Gibbs energy can be represented in the polynomial form (3) for any clique size. Now let us discuss the minimization of such energies. Quite successful optimization techniques for graphical models have been proposed to minimize quadratic energies with submodular terms (e.g. [1]) and nonsubmodular terms (e.g. [16,13]). We intend to apply the same techniques to minimize higher order Gibbs energies by transforming the latter to the quadratic ones. The transformation is based on adding dummy variables, each one substituting the product of two initial variables. Some known theoretical works [18,14] consider the reduction of optimization of a general pseudo-Boolean function $E(\mathbf{x})$ in polynomial time to optimization of a quadratic pseudo-Boolean function. In contrast to [18], Algorithm 2 guarantees that the goal quadratic function has the same minimum and at the same variables as the initial general pseudo-Boolean function. Also, as distinct from what has been proposed in [14], we present both a detailed proof to verify the introduced algorithm and an efficient implementation of the latter on a relevant graph. To obtain the quadratic pseudo-Boolean function from the pseudo-Boolean function $E(\mathbf{x})$ (3), we replace the occurrence of $x_i x_j$ by the dummy variable x_{n+1} and adding the term $N \cdot (x_i x_j + 3x_{n+1} - 2x_i x_{n+1} - 2x_j x_{n+1})$ to $E(\mathbf{x})$: This gives the following function $\mathcal{E}(x_1, x_2, \cdots, x_{n+1})$:

$$\mathcal{E}(\mathbf{x}) = N \cdot (x_i x_j + 3x_{n+1} - 2x_i x_{n+1} - 2x_j x_{n+1}) + \sum_{S^* \subseteq \mathcal{V}^*} a_{S^*} \prod_{i \in S^*} x_i \ , \quad (14)$$

where

$$S^* = \begin{cases} (S - \{i,j\}) \cup \{n+1\} & \text{if} \{i,j\} \subseteq S \\ S & \text{if} \{i,j\} \nsubseteq S \end{cases} ,$$

and $\mathcal{V}^* = \{S^* | S \in \mathcal{V}\}$. To compute the constant N, (3) is rewritten first as follows:

$$E(\mathbf{x}) = a_{\{\varnothing\}} + \sum_{S_1 \subseteq \mathcal{V}} a^-_{S_1} \prod_{i \in S_1} x_i + \sum_{S_2 \subseteq \mathcal{V}} a^+_{S_2} \prod_{i \in S_2} x_i \ , \quad (15)$$

where $a_{\{\varnothing\}}$ is the absolute term, $a^-_{S_1}$'s are the negative coefficients, and $a^+_{S_2}$'s are the positive coefficients. Then let $A = a_{\{\varnothing\}} + \sum_{S_1 \subseteq \mathcal{V}} a^-_{S_1}$ be the sum of all the negative coefficients in (15) plus the absolute term. Note that $A \leq \min_{\mathbf{x} \in \mathcal{B}^n} E(\mathbf{x})$. Also, denote r a real number being greater than the minimal value of $E(\mathbf{x})$ on $\mathcal{B}^n$ (i.e. $r > \min_{\mathbf{x} \in \mathcal{B}^n} E(\mathbf{x})$). Practically, r can be any number being greater than a particular value of $E(\mathbf{x})$ on $\mathcal{B}^n$. Finally, the chosen value N has to satisfy the relationship $N \geq r - A$.

This replacement is repeated until we get a quadratic pseudo-Boolean function. Algorithm 2 shows these steps in detail. At each step, $\mathcal{E}(x_1, x_2, \cdots, x_{n+1})$ must satisfy the following.

Algorithm 2. Transforming to Quadratic

Input: general pseudo-Boolean function $E(\mathbf{x})$ (3).
set $A = a_{\{\varnothing\}} + \sum_{S_1 \subseteq \mathcal{V}} a^-_{S_1}$, set $r > \min_{\mathbf{x}\in\mathcal{B}^n} E(\mathbf{x})$ (e.g., $r = E(\mathbf{0}) + 1$), and set $N \geq r - A$
2. **while** ($\exists S \subseteq \mathcal{V}$ and $|S| > 2$) **do**
Select a pair $\{i, j\} \subseteq S$ and update the coefficients
$a_{\{i,j\}} = a_{\{i,j\}} + N$, $a_{\{n+1\}} = 3N$, $a_{\{i,n+1\}} = a_{\{j,n+1\}} = -2N$,
$a_{(S-\{i,j\})\cup\{n+1\}} = a_S$, set $a_S = 0 \quad \forall S \supseteq \{i, j\}$
4. $n = n + 1$, update the function as shown in (14)
end while
Output: The quadratic pseudo-Boolean function $\mathcal{E}(\mathbf{x})$

Lemma 1. *Let $\mathcal{M}_E = \{\mathbf{y} \in \mathcal{B}^n | E(\mathbf{y}) = \min_{\mathbf{x}\in\mathcal{B}^n} E(\mathbf{x})\}$ be a set of all $\mathbf{y} \in \mathcal{B}^n$ such that $E(\mathbf{y})$ is the global minimum of the function E on $\mathcal{B}^n$. Then*

1. *$\mathcal{E}(x_1, x_2, \cdots, x_{n+1}) = E(x_1, x_2, \cdots, x_n)$,*
2. *$(y_1, y_2, \cdots, y_{n+1}) \in \mathcal{M}_\mathcal{E}$ iff $(y_1, y_2, \cdots, y_n) \in \mathcal{M}_E$.*

Proof. For $x, y, z \in \mathcal{B}$, the function $g(x, y, z) = xy + 3z - 2xz - 2yz$ equals 0 if $xy = z$, and $g(.) \geq 1$ otherwise. If $x_i x_j = x_{n+1}$ then $\mathcal{E}(\mathbf{x}) = N \cdot g(x_i, x_j, x_{n+1}) + \sum_{S^*\subseteq\mathcal{V}^*} a_{S^*} \prod_{i\in S^*} x_i = 0 + \sum_{S^*\subseteq\mathcal{V}^*} a_{S^*} \prod_{i\in S^*} x_i$, i.e. $\mathcal{E}(\mathbf{x}) = \sum_{S\subseteq\mathcal{V}} a_S \prod_{i\in S} x_i = E(\mathbf{x})$. More specifically, $\mathcal{E}(\mathbf{x})$ has the same minimum value as $E(\mathbf{x})$ on $\mathcal{B}^n$. On the other hand, let $y_i y_j \neq y_{n+1}$ which implies $g(y_i, y_j, y_{n+1}) \geqslant 1$. Assuming that $(y_1, y_2, \cdots, y_{n+1}) \in \mathcal{M}_\mathcal{E}$ we have $\mathcal{E}(\mathbf{y}) = N \cdot g(y_i, y_j, y_{n+1}) + \sum_{S^*\subseteq\mathcal{V}^*} a_{S^*} \prod_{i\in S^*} y_i$. As follows from choices of A and r, $N > 0$. So that $\mathcal{E}(\mathbf{y}) \geqslant N + \sum_{S^*\subseteq\mathcal{V}^*} a_{S^*} \prod_{i\in S^*} y_i \geqslant N + A \geqslant r$ due to our choice of $N \geqslant r - A$. This contradicts the assumption $(y_1, y_2, \cdots, y_{n+1}) \in \mathcal{M}_\mathcal{E}$. Thus, $(y_1, y_2, \cdots, y_{n+1}) \notin \mathcal{M}_\mathcal{E}$ if $y_i y_j \neq y_{n+1}$, and the lemma follows. □

By repeatedly applying the construction in Lemma 1, we get the following theorem (different versions of this theorem can be found in [14,18]):

Theorem 1. *Given a general pseudo-Boolean function $E(x_1, x_2, \cdots, x_n)$, there exists at most a quadratic pseudo-Boolean function $\mathcal{E}(x_1, x_2, \cdots, x_{n+m})$ where $m \geq 0$ such that*

1. *$(y_1, y_2, \cdots, y_{n+m}) \in \mathcal{M}_\mathcal{E} \Longleftrightarrow (y_1, y_2, \cdots, y_n) \in \mathcal{M}_E$*
2. *The size of the quadratic pseudo-Boolean function is bounded polynomially in the size of E, so the reduction algorithm terminates at polynomial time.*

Proof. Repeated application of the construction in the proof of Lemma 1 yields Point 1 of the theorem.

To prove Point 2, let us define M_3 the number of terms with $|S| > 2$ (i.e. of higher order terms containing more than two variables) in the function $E(x_1,$

$x_2, \cdots, x_n$).[2] In the loop of Algorithm 2, the term of size n (i.e. $|S| = n$) needs at most $n-2$ iterations. Also, at each iteration in this loop, at least one of the terms with $|S| > 2$ will decrease in size. Hence, the algorithm must terminate in at most $T \ll M_3(n-2)$ iterations because the average number of iterations for each term is less than $n-2$. Indeed, the larger number of variables in each energy term indicates that these terms share several common variables, so that they will be reduced concurrently. For example, a function with ten variables contains at most 968 terms with $|S| > 2$. Using Algorithm 2, it is reduced with $T = 68 \ll 968 \times 8$ iterations. This proves the claim about complexity. □

Although our work in this section is similar to the work in [14], it has to be mentioned that our paper gives a formal derivation for selection of the value N such that the resulting quadratic energy has the same minimum and at the same values of variables as the initial general pseudo-Boolean function. A different formula has been only stated, but not derived, in [14] before the minimization issues were proved.

5.1 Efficient Implementation

The number of dummy variables in the generated quadratic pseudo-Boolean function depends on the selection of the pairs $\{i, j\}$ in the loop of Algorithm 2. Finding the optimal selection to minimize this number is an NP-hard problem [14]. Also, searching for this pair in other terms will be exhaustive. However, in most Computer Vision problems, we deal with images on an arithmetic 2D lattice $\mathcal{V}$ with n pixels. The order of the Gibbs energy function to be minimized depends on the particular neighborhood system and the maximal clique size.

A natural 2D image structure helps us to define a general neighborhood system $\mathcal{N}$, e.g. the system of neighbors within a square of a fixed size centered on each pixel [17]. The size of this square determines the order of the system. The neighborhood system specifies the available types of cliques. The prior knowledge about the neighborhood system and the clique size can be used to minimize the number of dummy variables and to eliminate the search for the repeated pair in other terms. We will demonstrate this process on the second order neighborhood system and the cliques of the size 3 and 4, but it can be generalized for the higher orders. Figure 2(a) suggests that the second order neighborhood system contains four different cliques of the size 3 (i.e. $\mathcal{C}_{31}, \mathcal{C}_{32}, \mathcal{C}_{33}$, and $\mathcal{C}_{34}$). Thus, we can convert the cubic terms that correspond to the cliques of the size 3, to quadratic terms as follows:

- At each pixel (e.g. m) select the cubic term that corresponds to clique type $\mathcal{C}_{32}$.
- Reduce this term and the cubic term of the clique type $\mathcal{C}_{31}$ at the diagonal pixel (e.g. i), if possible, by eliminating common variables (e.g. j and ℓ).

[2] Obviously, a function E of n binary variables contains at most 2^n terms and at most $2^n - \frac{n^2+n+2}{2}$ terms with more than two variables ($|S| > 2$).

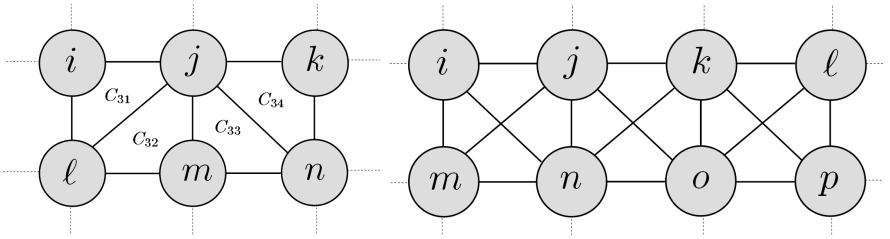


Fig. 2. The 2nd-order neighborhood on a 2D lattice: cliques of size 3 (a) and 4 (b)

– At each pixel (e.g. m) select the cubic term that corresponds to the clique type $\mathcal{C}_{33}$
– Reduce this term and the cubic term of the clique type $\mathcal{C}_{34}$ at the diagonal pixel (e.g. k), if possible, by eliminating common variables (e.g. j and n).

After a single scanning of the image, all the cubic terms will be converted to the quadratic terms, and every term will be visited only once.

As shown in Fig. 2(b), for a second order neighborhood system and clique of size four, the three neighbor cliques have the following potentials functions[3]:

$$\begin{aligned}
\varphi_{x_i x_m x_j x_n} &= x_i x_m x_j x_n + x_i x_m x_j + x_i x_j x_n + x_i x_m x_n + x_m x_j x_n \dots \\
\varphi_{x_j x_n x_k x_o} &= x_j x_n x_k x_o + x_j x_n x_k + x_j x_n x_o + x_j x_k x_o + x_k x_o x_n \dots \quad . \\
\varphi_{x_k x_o x_\ell x_p} &= x_k x_o x_\ell x_p + x_k x_o x_\ell + x_k x_o x_p + x_k x_\ell x_p + x_\ell x_p x_o \dots
\end{aligned}$$

For this configuration, one can notice that $\varphi_{x_i x_m x_j x_n}$ and $\varphi_{x_j x_n x_k x_o}$ cliques' potential functions share the elements x_j and x_n. Also $\varphi_{x_j x_n x_k x_o}$ and $\varphi_{x_k x_o x_\ell x_p}$ share the elements x_k and x_o. Therefore, by replacing $x_j x_n$ and $x_k x_o$ with two new elements using Algorithm 2, the clique's potential function $\varphi_{x_j x_n x_k x_o}$ will be quadratic. Repeating this for every three neighbor cliques through the whole grid, and assuming a circular grid, i.e. the first and the last column are neighbors, all the cliques' potential functions will be converted to quadratic. Notice that using this technique in the reduction provides the minimum number of dummy variables that equals the number of cliques of size four in the grid.

Notice that these scenarios are not unique. Many other scenarios can be chosen for image scanning and selection of the higher order cliques to be reduced. However, in the efficient scenario every higher order term must be converted to a quadratic term after being visited only once. To illustrate the enhancement introduced by the proposed implementation, we give the following example. The linear search in a list runs in $O(n)$ where n is the number of elements. An image of size $R \times C$ has $4(R-1)(C-1)$ triple cliques in the 2nd-order neighborhood system. Each triple clique has 4 terms with $|\mathcal{S}| > 1$ with total 9 elements as shown in (6). So applying Algorithm 2 directly without the proposed implementation has an overhead proportional to $36(R-1)(C-1)$.

[3] For brevity, only the higher-order terms that appear in the discussion are typed, assuming that all coefficients are 1.

6 Experimental Results

To illustrate the potential of the higher order cliques in modelling complex objects and assess the performance of the proposed algorithm, let us consider image segmentation into two classes: object and background. Following a popular conventional approach, an input image and the desired region map (the labeled image) are described by a joint MGRF model of independent image signals and interdependent region labels. The desired map is the mapping $\mathbf{x} : \mathcal{V} \longrightarrow \mathcal{B}$, where $\mathcal{B}$ is the set of two labels $\{0 \equiv \text{"background"}, 1 \equiv \text{"object"}\}$. The MAP estimate of $\mathbf{x}$, given an input image, is obtained by minimizing an energy function (3) where each label x_i is a binary variable in the energy function. The unary term $\varphi(x_i)$ in this function specifies the data penalty. This term is chosen to be $\varphi(x_i) = ||I_i - \widehat{I}_{x_i}||^2$ where I_i is the input feature vector for the pixel i, e.g. a 4D vector $I_i = (I_{\mathrm{L}i}, I_{\mathrm{a}i}, I_{\mathrm{b}i}, I_{\mathrm{t}i})$ [23] where the first three components are the pixel-wise color L*a*b* components and $I_{\mathrm{t}i}$ is a local texture descriptor [24]. Seeds selected from the input image can be used to estimate feature vectors for the object, $\widehat{I}_1$, and background, $\widehat{I}_0$.

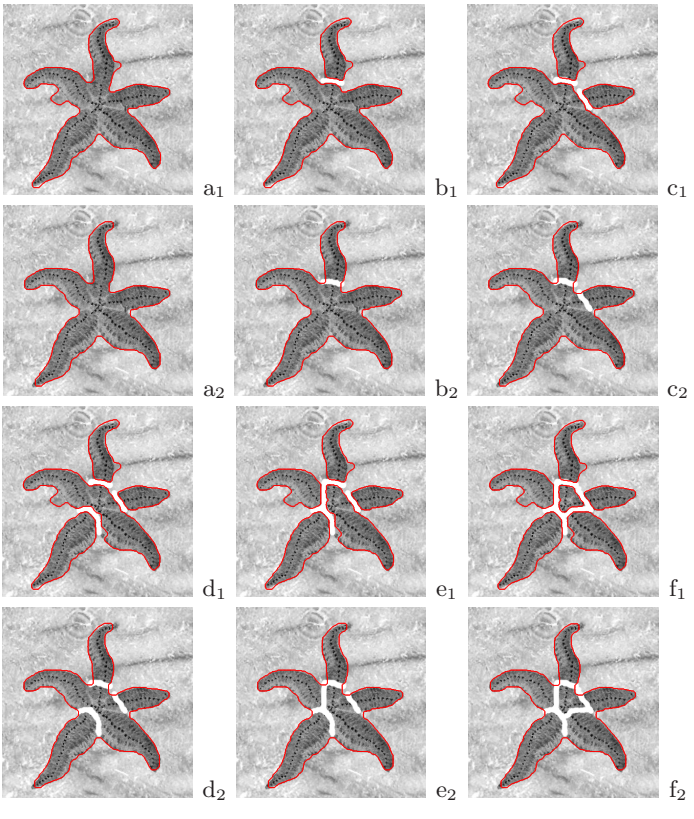


Fig. 3. Starfish segmentation – the pairwise (a_1–f_1) vs. third-order (a_2–f_2) cliques

Using feature vectors $\widehat{I}_1$ and $\widehat{I}_0$, an initial binary map can be estimated. For the cliques of size 2, the pairwise potentials were analytically estimated from the initial map using the proposed method described in Section 4. The potential for the third order cliques have the same analytical form (13) but with the frequency $f_{\text{neq}}(\mathbf{x}) = \frac{1}{|\mathcal{T}|}\sum_{\{i,j,\ell\}\in\mathcal{T}}(1-\delta(x_i = x_j = x_\ell))$.

In all our experiments, we selected the second order neighborhood system with the clique sizes from 1 to 3. By defining the clique potentials (unary, pairwise, and third-order), we identify the target segmentation energy to be minimized. After that, Algorithm 1 is used to compute the coefficients of the polynomial that represents the segmentation energy and Algorithm 2 generates a quadratic version of this polynomial. Finally, we use the extended roof duality algorithm (QPBOP) [16] to minimize the quadratic pseudo-Boolean function. For all the presented examples, QPBOP technique was able to label all pixels. In the experiments below, images are segmented with unary and pairwise cliques and with unary and third order cliques in the MGRF model. Of course, cliques of greater sizes can be more efficient for describing complex regions, but we used the third order for illustration purposes only.

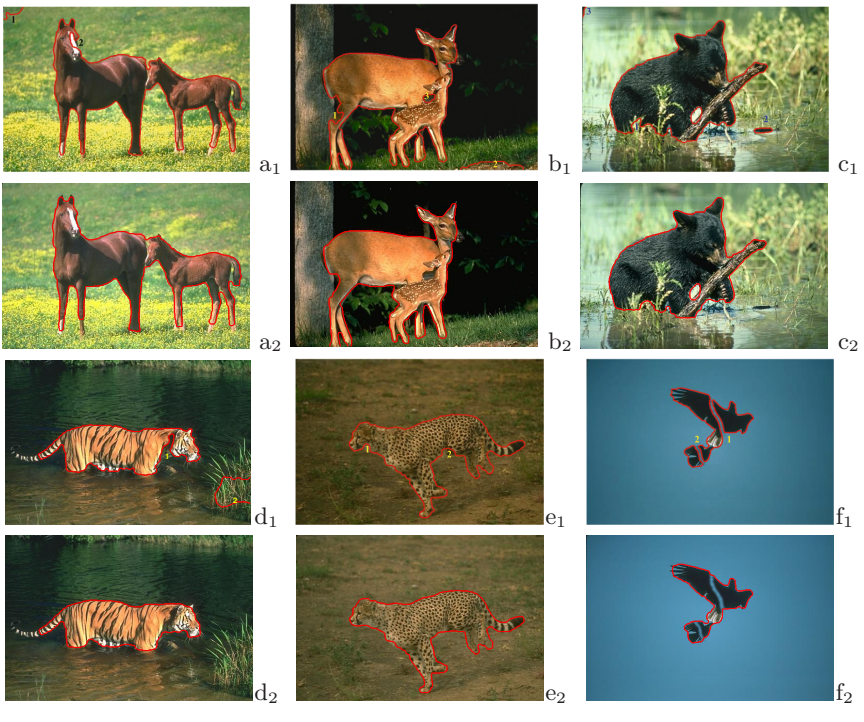


Fig. 4. More segmentation results: the pairwise (a_1–f_1) and third-order (a_2–f_2) cliques; numbers in images refer to regions with inhomogeneities (a–e) and partial artificial occlusions (f)

Figure 3 shows the starfish segmentation. Unlike the pairwise interaction in Fig. 3,a_1, the higher order interaction in Fig. 3,a_2 overcomes the intensity inhomogeneities of the starfish and its background. For more challenging situations, we occluded some parts from the starfish in Figs. 3,b–f. The higher order interaction successes to get the correct boundary of the starfish while only the pairwise interaction could not. The average processing time for this experiment: in the third-order case 6 sec, comparing to 2 sec in the pairwise case.

More segmentation results are shown in Fig. 4 for different color objects from the Berkeley Segmentation Dataset [25]. Numbers 1 and 2 in Fig. 4,a_1, indicate regions with inhomogeneities where the pairwise interaction fails. As expected, the higher order interaction overcomes the problem (Fig. 4,a_2). Similar regions exist in Figs. 4,b_1 (♯1, 2, and 3); c_1 (♯1, 2 and 3); d_1 (♯1 and 2), and e_1 (♯1 and 2). In Fig. 4,f artificial occlusions were inserted by letting some object regions take the background color (regions 1 and 2 in f_1). These results show the higher order interactions are instrumental for more correct segmentation.

Finally, we can notice the segmentation results' improvements. Of course, if we use cliques of sizes greater than three, we can model more complex interactions that lay outside the domain of uniform spatial interaction assumed in the 3rd-order MGRF model. However, we used third order MGRF for illustration purposes only. Recall that, our goal is to introduce a link between the higher order energies and the quadratic to Computer Vision community. This helps to use the well established tools of a quadratic energy's optimization to optimize a higher order one.

7 Conclusions

This paper has introduced an efficient link between the MGRF models with higher order and pairwise cliques. We have proposed an algorithm that transforms a general pseudo-Boolean function into a quadratic pseudo-Boolean function and provably guarantees the obtained quadratic function has the same minimum and at the same variables as the initial higher order one. The algorithm is efficiently implemented for image-related graphical models. Thus, we can apply the well known pairwise MGRFs solvers to the higher order MGRFs. The MGRF parameters are analytically estimated. Experimental results show the proposed framework notably improves image segmentation and therefore may be useful for solving many other Computer Vision problems.

References

1. Boykov, Y., Veksler, O., Zabih, R.: Fast Approximation Energy Minimization via Graph Cuts. IEEE Trans. PAMI 23(11), 1222–1239 (2001)
2. Kolmogorov, V., Zabih, R.: What Energy Functions Can be Minimized via Graph Cuts? IEEE Trans. PAMI 26(2), 147–159 (2004)
3. Yedidia, J.S., Freeman, W.T., Weiss, Y.: Generalized Belief Propagation. In: NIPS, pp. 689–695 (2000)

4. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient Belief Propagation for Early Vision. Int. J. Computer Vision 70(1), 41–54 (2006)
5. Kolmogorov, V.: Convergent Tree-Reweighted Message Passing for Energy Minimization. IEEE Trans. PAMI 28(10), 1568–1583 (2006)
6. Wainwright, M.J., Jaakkola, T., Willsky, A.S.: Tree-Based Reparameterization for Approximate Inference on Loopy Graphs. In: NIPS, pp. 1001–1008 (2001)
7. Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M.F., Rother, C.: A Comparative Study of Energy Minimization Methods for Markov Random Fields. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 16–29. Springer, Heidelberg (2006)
8. Lan, X., Roth, S., Huttenlocher, D.P., Black, M.J.: Efficient Belief Propagation with Learned Higher-Order Markov Random Fields. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 269–282. Springer, Heidelberg (2006)
9. Potetz, B.: Efficient Belief Propagation for Vision Using Linear Constraint Nodes. In: CVPR (2007)
10. Paget, R., Longstaff, I.D.: Texture Synthesis via a Noncausal Nonparametric Multiscale Markov Random Field. IEEE Trans. Image Processing 7(6), 925–931 (1998)
11. Roth, S., Black, M.J.: Fields of Experts: A Framework for Learning Image Priors. In: CVPR, pp. 860–867 (2005)
12. Kohli, P., Kumar, M., Torr, P.: $\mathcal{P}^3$ & beyond: Solving Energies with Higher Order Cliques. In: CVPR (2007)
13. Kolmogorov, V., Rother, C.: Minimizing Nonsubmodular Functions with Graph Cuts-A Review. IEEE Trans. PAMI 29(7), 1274–1279 (2007)
14. Boros, E., Hammer, P.L.: Pseudo-Boolean Optimization. Discrete Appl. Math. 123(1-3), 155–225 (2002)
15. Rother, C., Kumar, S., Kolmogorov, V., Blake, A.: Digital Tapestry. In: CVPR, pp. 589–596 (2005)
16. Rother, C., Kolmogorov, V., Lempitsky, V.S., Szummer, M.: Optimizing Binary MRFs via Extended Roof Duality. In: CVPR (2007)
17. Geman, S., Geman, D.: Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. IEEE Trans. PAMI 6, 721–741 (1984)
18. Rosenberg, I.G.: Reduction of Bivalent Maximization to The Quadratic Case. Cahiers du Centre d'Etudes de Recherche Operationnelle 17, 71–74 (1975)
19. Freedman, D., Drineas, P.: Energy Minimization via Graph Cuts: Settling What is Possible. In: CVPR, pp. 939–946 (2005)
20. Cunningham, W.: Minimum Cuts, Modular Functions, and Matroid Polyhedra. Networks 15, 205–215 (1985)
21. Gimel'farb, G.L.: Image Textures and Gibbs Random Fields. Kluwer Academic Publishers, Dordrecht (1999)
22. Chen, C.C.: Markov Random Field Models in Image Analysis. PhD thesis, Michigan State University, East Lansing (1988)
23. Chen, S., Cao, L., J.L., Tang, X.: Iterative MAP and ML Estimations for Image Segmentation. In: CVPR (2007)
24. Carson, C., Belongie, S., Greenspan, H., Malik, J.: Blobworld: Image Segmentation Using Expectation-Maximization and Its Application to Image Querying. IEEE Trans. PAMI 24(8), 1026–1038 (2002)
25. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A Database of Human Segmented Natural Images and Its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics. In: ICCV, pp. 416–423 (2001)