Image and Vision Computing 28 (2010) 467–473

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Graph cuts with many-pixel interactions: Theory and applications to shape modelling

Daniel Freedman [a,*], Matthew W. Turek [b]

[a] Computer Science Department, RPI, United States and Hewlett-Packard Laboratories, Israel
[b] Kitware, Inc., United States

ARTICLE INFO

Article history:
Received 7 November 2007
Received in revised form 8 July 2009
Accepted 18 July 2009

Keywords:
Graph cuts
Shape modelling

ABSTRACT

Many problems in computer vision can be posed in terms of energy minimization, where the relevant energy function models the interactions of many pixels. Finding the global or near-global minimum of such functions tends to be difficult, precisely due to these interactions of large (> 3) numbers of pixels. In this paper, we derive a set of sufficient conditions under which energies which are functions of discrete binary variables may be minimized using graph cut techniques. We apply these conditions to the problem of incorporating shape priors in segmentation. Experimental results demonstrate the validity of this approach.

1. Introduction

Many problems in computer vision are posed within a framework of energy minimization. Further, it is often the case that the energy is a function of discrete variables. This latter situation may arise naturally, as in the case of a foreground–background segmentation problem where the variables are binary (0 for background, 1 for foreground); it may also arise by design, as in the case of optical flow in which the flow vectors are taken to belong to a discrete set. In either case, there is a need for algorithms which compute global or near-global minima for such combinatorial problems. The graph cuts methodology [4,15,16] has proven highly effective at tackling these types of problems; examples include stereo [14], motion [5], and segmentation [3].

A natural and important question arises: what functions of discrete binary variables can be minimized via graph cuts? The papers by Kolomogorov and Zabih [15,16] made good headway in attacking this problem. These papers showed necessary and sufficient conditions for the exact minimization of energy functions where there are terms depending on pairs of pixels (the so-called $\mathscr{F}^2$ class) and where there are terms depending on triples of pixels (the so-called $\mathscr{F}^3$ class). In practice, most research that has applied the graph cuts methodology has relied on energy functions with pairwise pixel interactions: the pairwise energies usually capture a robust smoothness term.

In this paper, we focus on the problem of minimizing discrete binary energy functions with many pixel interactions. This is an important problem, as there is a large class of computer vision problems which feature the interactions of more than three pixels. For example, many vision problems can be posed in terms of a Markov Random Field, which is a handy way of capturing the interaction of a pixel with its neighbours. Finding a MAP estimate of the underlying variables is the same as minimizing an energy function which is minus the log of the MRF; this energy function naturally has many pixel interactions, as a pixel interacts with all of its neighbours. A second example pertains to shape modelling; more on this shortly.

There are two main contributions of this paper. The first is a set of sufficient conditions for the minimization of discrete binary energy functions with k-wise pixel interactions. These conditions are useful, as they allow one to determine whether a given function may be minimized using graph cut techniques. The second contribution is an application of these techniques to the problem of shape modelling. In particular, we show how to introduce a type of shape information into a graph cut style segmentation. This has long been known to be a difficult, yet important problem. Our solution is not the final word on this subject, but it does represent a useful point of departure. Note that while the conditions for energy minimization were first introduced by the authors in [10], the shape modelling technique and accompanying results are entirely new.

The remainder of the paper is organized as follows. In Section 2, we derive the set of sufficient conditions for the minimization of energy functions with k-wise pixel interactions. In Section 3, we introduce a technique for incorporating shape information into segmentation, and show the results of experiments. Section 4 concludes.

* Corresponding author. Address: Hameyasdim 68/B, 30900 Zikhron Ya'aqov, Israel. Tel.: +972 52 446 7138.
E-mail address: d.e.freedman@gmail.com (D. Freedman).

0262-8856/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2009.07.006

2. Theory of *k*-pixel interactions

In this section, we extend the results of Kolmogorov and Zabih [15,16] on exact minimization of energy functions via graph cuts. We begin, in Section 2.1, by laying out a useful theorem on the types of pairwise functions that can be minimized via graph cut constructions. Using this simple result, the rest of the section is posed in an entirely algebraic manner, without explicit reference to graphs. In Section 2.2, we show how the regularity conditions can be derived very easily in the case of pairwise functions; in Section 2.3, we use a somewhat longer proof for the triplewise case. Although both of these results have already been proven in [15,16], our proofs serve two purposes: they are simplified, purely algebraic proofs, making them easier to parse; and they show how these ideas can be generalized to the *k*-wise case, which we do in Section 2.4. In Section 2.5, we sketch the graph construction from which min-cut may be applied to minimize the energy. In Section 2.6, we discuss the relationship of our conditions to the submodularity conditions.

2.1. A useful theorem

We begin by stating a theorem which is very important in the subsequent work. We note that this theorem is generally considered to be part of combinatorial optimization folklore, and a version of it may be found in [19].

Theorem 1. *Let* $x_i \in \{0,1\}$ *and let* $E(x_1,\ldots,x_n) = \sum_{i,j} a_{ij} x_i x_j + L$, *where L represents terms that are linear in the* x_i *plus any constants (i.e.* $L = \sum_i a_i x_i + c$*). Then E can be minimized via graph cut techniques if and only if* $a_{ij} \leqslant 0$ for all i,j.

Proof. We only prove the "if" direction; a proof of the "only if" direction may be found in [19]. With a little manipulation, such an E can be rewritten as

$$E = \sum_{i,j} a'_{ij} x_i (1 - x_j) + L'$$

where $a'_{ij} = -a_{ij}$ and the linear term L' is altered. Ignoring the linear term L' for the moment, it is easy to see that minimizing E over the binary variables x_i is the same as finding a minimum cut in a complete graph with n vertices, one vertex corresponding to each x_i, and edge weights given by $w_{ij} = a'_{ij}$. The cut itself splits those vertices with $x_i = 0$ from those with $x_i = 1$; this is because choosing $x_i = 1$ and $x_j = 0$ adds a'_{ij} to the energy, whereas any other setting of x_i and x_j does not add a'_{ij} to the energy.[1] It is well known, from the theory of combinatorial optimization [19], that solving min-cut in polynomial time is possible if and only if the edge weights are non-negative. Thus, we must have that $a'_{ij} \geqslant 0$, so that $a_{ij} \leqslant 0$.

We may now turn to the issue of the linear terms L'. Note that

$$L' = \sum_i a'_i x_i + c' = \sum_{i: a'_i \geqslant 0} a'_i x_i + \sum_{i: a'_i < 0} |a'_i|(1 - x_i) + c''$$

Thus, we can add such terms into the graph formulation by simply adding in source (S) and sink (T) nodes, where S corresponds to 0 and T corresponds to 1. In this case, for each i for which $a'_i \geqslant 0$, we add in an edge from the node i to S with weight a'_i; and for each i for which $a'_i < 0$, we add in an edge from the node i to T with weight $|a'_i|$. All of these weights are non-negative, and thus we can apply graph cut techniques to optimize in polynomial time. □

[1] Note that we would usually set $a'_{ji} = a'_{ij}$, so that $x_i = 0$ and $x_j = 1$ also yields the same result; this is the distinction between cuts across directed and undirected graphs.

2.2. Recasting the $\mathscr{F}^2$ case

Before going on to discuss the *k*-wise case, we will discuss the simpler pairwise and triplewise cases. Of course, the results for these cases have already been demonstrated in [15,16]; however, we use the same approach here as we do for the *k*-wise case, so it is worth reviewing these cases. (We also believe that the proofs presented here, which are purely algebraic, are simpler than those in [15,16].)

The class of energy functions belonging to $\mathscr{F}^2$ includes all those with pairwise pixel interactions, i.e.

$$E(x_1,\ldots,x_n) = \sum_i E_i(x_i) + \sum_{i,j} E_{ij}(x_i,x_j) \tag{1}$$

We may now reprove the regularity results of [15,16] very simply using Theorem 1. Note that we may write

$$E_{ij}(x_i,x_j) = E_{ij}^{00}(1-x_i)(1-x_j) + E_{ij}^{01}(1-x_i)x_j + E_{ij}^{10}x_i(1-x_j) + E_{ij}^{11}x_i x_j$$

where $E_{ij}^{qr} = E_{ij}(x_i = q, x_j = r)$. Similarly, we may write

$$E_i(x_i) = E_i^0(1-x_i) + E_i^1 x_i$$

Putting these terms back into Eq. (1) gives

$$E(x_1,\ldots,x_n) = \sum_{i,j} \left(E_{ij}^{00} + E_{ij}^{11} - E_{ij}^{01} - E_{ij}^{10}\right) x_i x_j + L$$

where again L includes terms that are linear in the x_i, as well as any constants. Applying Theorem 1 says that such an energy can be minimized via graph cuts if and only if

$$E_{ij}^{00} + E_{ij}^{11} - E_{ij}^{01} - E_{ij}^{10} \leqslant 0 \quad \forall i,j$$

which is precisely the regularity condition of [15,16].

2.3. Recasting the $\mathscr{F}^3$ case

The class of energy functions belonging to $\mathscr{F}^3$ includes all those with triplewise pixel interactions, i.e.

$$E(x_1,\ldots,x_n) = \sum_i E_i(x_i) + \sum_{i,j} E_{ij}(x_i,x_j) + \sum_{i,j,k} E_{ijk}(x_i,x_j,x_k)$$

Before proving any results, let us introduce some notation. Greek letters, such as α and β, will typically refer to subsets of $\{1,\ldots,n\}$. We define $\mathbf{x}_\alpha$ to be $\prod_{\ell\in\alpha} x_\ell$. Also, we let $E_\beta^{ijk} = E_{ijk}(x_i = 1, i \in \beta)$.

Let us begin by expanding the function $E_{ijk}(x_i,x_j,x_k)$ in a polynomial series:

$$E_{ijk}(x_i,x_j,x_k) = \sum_{\alpha\subset\{i,j,k\}} a_\alpha \mathbf{x}_\alpha$$

To solve for the coefficients of the expansion, a_α, we can plug in all values of the binary variables, leading to eight equations in eight unknowns. After some algebra, these equations can be solved to yield

$$a_\alpha = \sum_{\beta\subset\alpha} (-1)^{|\alpha|-|\beta|} E_\beta^{ijk}$$

The function E_{ijk} may therefore be written

$$E_{ijk}(x_i,x_j,x_k) = a_{ij}x_i x_j + a_{ik}x_i x_k + a_{jk}x_j x_k + a_{ijk}x_i x_j x_k + L$$

where L is a subquadratic term.

The key step is to convert E_{ijk}, which is an $\mathscr{F}^3$ function, into an $\mathscr{F}^2$ function via the introduction of an extra binary variable y_{ijk}. In particular, note that

$$x_i x_j x_k = \max_{y_{ijk}\in\{0,1\}} [(x_i + x_j + x_k - 2)y_{ijk}]$$

If $a_{ijk} \leqslant 0$, we may write

$$a_{ijk}x_ix_jx_k = \min_{y_{ijk}\in\{0,1\}}[a_{ijk}(x_i + x_j + x_k - 2)y_{ijk}]$$

which therefore gives

$$E_{ijk}(x_i,x_j,x_k) = \min_{y_{ijk}}[a_{ij}x_ix_j + a_{ik}x_ix_k + a_{jk}x_jx_k + a_{ijk}x_iy_{ijk} + a_{ijk}x_jy_{ijk} + a_{ijk}x_ky_{ijk} + L]$$

(L is a modified subquadratic term from the L introduced above.) Thus, E_{ijk} is written as a pairwise ($\mathscr{F}^2$) function, where we have introduced the extra variable y_{ijk}. In fact, we must take the minimum over y_{ijk}; however, since the entire function E will ultimately be minimized, this step simply introduces some extra variables to minimize over.

Now, what if $a_{ijk} > 0$? In a similar manner to the above, we can introduce an expansion

$$E_{ijk} = \sum_{\alpha\subset\{i,j,k\}} \bar{a}_\alpha \bar{\mathbf{x}}_\alpha$$

where $\bar{x}_i = 1 - x_i$ (and following the previous convention, $\bar{\mathbf{x}}_\alpha = \prod_{\ell\in\alpha}\bar{x}_\ell$). It can be shown that

$$\bar{a}_\alpha = \sum_{\beta\subset\alpha}(-1)^{|\alpha|-|\beta|}\bar{E}^{ijk}_\beta$$

where $\bar{E}^{ijk}_\beta = E_{ijk}(x_i = 0, i\in\beta)$. In this case, some inspection shows that $\bar{a}_{ijk} = -a_{ijk}$. Therefore, if $a_{ijk} > 0$, then $\bar{a}_{ijk} < 0$, and (after some manipulation) we can write

$$E_{ijk}(x_i,x_j,x_k) = \min_{y_{ijk}}[\bar{a}_{ij}x_ix_j + \bar{a}_{ik}x_ix_k + \bar{a}_{jk}x_jx_k + \bar{a}_{ijk}x_iy_{ijk} + \bar{a}_{ijk}x_jy_{ijk} + \bar{a}_{ijk}x_ky_{ijk} + \bar{L}]$$

Note that the variables above are x_i and not $\bar{x}_i$. This is due to the fact that

$$\bar{x}_i\bar{x}_j = (1 - x_i)(1 - x_j) = x_ix_j + \text{linear term} + \text{constant}$$

so that any terms of the form $\bar{x}_i\bar{x}_j$ can be effectively replaced by x_ix_j without affecting the expression (except through the precise forms of the subquadratic terms, which we do not care about).

Finally, let

$$b_\alpha = \begin{cases} a_\alpha & \text{if } a_\alpha \leqslant 0, \\ \bar{a}_\alpha & \text{otherwise.} \end{cases}$$

Then we have that

$$E = \min_{\text{all}\, y_{ijk}}\left\{\sum_{i,j,k}[b_{ij}x_ix_j + b_{ik}x_ix_k + b_{jk}x_jx_k + b_{ijk}x_iy_{ijk} + b_{ijk}x_jy_{ijk} + b_{ijk}x_ky_{ijk} + L_{ijk}]\right\}$$

Due to Theorem 1, we can ignore the linear terms L_{ijk}. We also note that since $\bar{a}_{ijk} = -a_{ijk}$, we must have that $b_{ijk} \leqslant 0$. Thus, we know that the terms involving the y_{ijk} variables satisfy the conditions of Theorem 1 (namely, that their coefficients be non-positive). Thus, we can look at the remainder of the function, i.e.

$$E' = \sum_{i,j,k} b_{ij}x_ix_j + b_{ik}x_ix_k + b_{jk}x_jx_k = \sum_{ij} q_{ij}x_ix_j$$

where $q_{ij} = \sum_k b_{ij}$. In this case, according to Theorem 1, the conditions under which $q_{ij} \leqslant 0$ are identical to the conditions under which the energy can be minimized via graph cut methods.

Using the expressions for a_α and $\bar{a}_\alpha$, a little algebra shows that

$$b_{ij} = E_{ijk}(0,0,x_k) + E_{ijk}(1,1,x_k) - E_{ijk}(0,1,x_k) - E_{ijk}(1,0,x_k)$$

where $x_k = 0$ if $b_{ij} = a_{ij}$ and $x_k = 1$ otherwise. Thus,

$$q_{ij} = \sum_k[E_{ijk}(0,0,x_k) + E_{ijk}(1,1,x_k) - E_{ijk}(0,1,x_k) - E_{ijk}(1,0,x_k)] \quad (2)$$

It turns out that the condition that $q_{ij} \leqslant 0$ is precisely the regularity condition of [15,16]. To see this, let us introduce the notation $x_{-ij} = \{x_\ell\}_{\ell\neq i,j}$, and

$$E^{proj}_{ij}(x_i,x_j) = E(x_i,x_j,x_{-ij})$$

where we have assumed the x_{-ij} are fixed, and therefore have suppressed them on the left-hand side. Then

$$E^{proj}_{ij}(0,0) = \sum_k E_{ijk}(0,0,x_k) + \sum_{i'\neq i,j'\neq j,k'} E_{i'j'k'}(x_{i'},x_{j'},x_{k'})$$

The second term does not depend on x_i or x_j. Thus using Eq. (2), $q_{ij} \leqslant 0$ becomes

$$E^{proj}_{ij}(0,0) + E^{proj}_{ij}(1,1) - E^{proj}_{ij}(0,1) - E^{proj}_{ij}(1,0) \leqslant 0$$

which is exactly the regularity condition of [15,16].

2.4. The generic $\mathscr{F}^k$ case

We now come to the most generic case of energy functions with k-wise pixel interactions, labelled $\mathscr{F}^k$. We may use similar, though perhaps simpler, arguments as in the case of $\mathscr{F}^3$ to establish sufficient conditions for a function in $\mathscr{F}^k$ to be minimized via graph cut methods.

The first step is to realize that any function in $\mathscr{F}^k$ can be written as

$$E(x_1,\ldots,x_n) = \sum_{\alpha\subset\{1,\ldots,n\},|\alpha|\leqslant k} a_\alpha\mathbf{x}_\alpha \quad (3)$$

where again $\mathbf{x}_\alpha = \prod_{\ell\in\alpha}x_\ell$. This fact can easily be proven, though we do not do so here. As described in Section 2.3, we can solve for the coefficients a_α by means of a linear system of 2^k equations in 2^k unknowns; the result (whose precise derivation is omitted here) is the same as in the case of $\mathscr{F}^3$ functions, i.e.

$$a_\alpha = \sum_{\beta\subset\alpha}(-1)^{|\alpha|-|\beta|}E_\beta \quad (4)$$

The second step is to convert an $\mathscr{F}^k$ function to an $\mathscr{F}^2$ function through the introduction of extra variables; this is precisely analogous to what was done in Section 2.3. Note that if $|\alpha| > 2$

$$\mathbf{x}_\alpha = \max_{y_\alpha\in\{0,1\}}\left[\left(\sum_{\ell\in\alpha}x_\ell - (|\alpha| - 1)\right)y_\alpha\right]$$

where y_α is the extra binary variable. If $a_\alpha \leqslant 0$, we may write

$$a_\alpha\mathbf{x}_\alpha = \min_{y_\alpha\in\{0,1\}}\left[a_\alpha\left(\sum_{\ell\in\alpha}x_\ell - (|\alpha| - 1)\right)y_\alpha\right] \quad (5)$$

The final step is to use the above fact to note that if $a_\alpha \leqslant 0$ for all α, we can combine Eqs. (3) and (5) to yield

$$E(x_1,\ldots,x_n) = L + \sum_{i,j} a_{ij}x_ix_j + \sum_{\alpha:|\alpha|>2}\min_{y_\alpha\in\{0,1\}}\left[a_\alpha\left(\sum_{\ell\in\alpha}x_\ell - (|\alpha| - 1)\right)y_\alpha\right]$$

where as usual, L represents linear terms and the constant. Thus, the minimization of E can be rewritten as follows:

$$\min_{x_i} E(x_i) = \min_{x_i,y_\alpha}\widetilde{E}(x_i,y_\alpha) \quad (6)$$

where

$$\widetilde{E}(x_i,y_\alpha) = \sum_{\alpha:|\alpha|>2}\sum_{\ell\in\alpha}a_\alpha x_\ell y_\alpha + \sum_{i,j}a_{ij}x_ix_j + L \quad (7)$$

We can apply Theorem 1 to this function to discover that E can be minimized by graph cut techniques if

$a_\alpha \leqslant 0 \quad \forall\alpha : 2 \leqslant |\alpha| \leqslant k$

Plugging in the expression for a_α from Eq. (4) leads to the following sufficient conditions for minimization of E via graph cut methods:

$$\sum_{\beta\subset\alpha}(-1)^{|\alpha|-|\beta|}E_\beta \leqslant 0 \quad \forall\alpha : 2 \leqslant |\alpha| \leqslant k \tag{8}$$

where as before $E_\beta = E(x_i = 1, i \in \beta)$. The inequalities of (8) represent the main result of this section of the paper.

2.4.1. Relationship of condition (8) to the $k = 2$ and $k = 3$ cases

We have used similar arguments, in particular auxiliary variables, to generate the conditions for the general k case as we did for the $k = 2$ and $k = 3$ cases. In the case of $k = 2$, the conditions in (8) reduce precisely the regularity conditions of [15,16]. To see this, expand (8) for $k = 2$; note that $|\alpha| = 2$, so that we have all α's are sets of the form $\{i, j\}$. For a specific $\alpha = \{i, j\}$, we have that the lefthand side of (8) becomes

$$\begin{aligned}\sum_{\beta\subset\{i,j\}}(-1)^{2-|\beta|}E_\beta &= E_\emptyset - E_{\{i\}} - E_{\{j\}} + E_{\{i,j\}}\\ &= E_{ij}(0,0) - E_{ij}(1,0) - E_{ij}(0,1) + E_{ij}(1,1)\end{aligned}$$

thus yielding the regularity conditions of [15,16].

The case $k = 3$ is somewhat different: the condition (8) will *not* reduce to the $k = 3$ version of the regularity conditions. The reason is that we invoke an extra argument in Section 2.3 for the case of $\mathscr{F}^3$, to eliminate the condition that $a_\alpha \leqslant 0$ for $|\alpha| = 3$. Why not use this extra argument in the general case? Such an argument relied on the fact that an expansion could also be performed on the $\bar{x}_i$ variables, where $\bar{x}_i = 1 - x_i$; it was then shown that $\bar{a}_\alpha = -a_\alpha$ for $|\alpha| = 3$, so that in this case either $a_\alpha \leqslant 0$ or $\bar{a}_\alpha \leqslant 0$. Unfortunately, this is not true for $k > 3$; indeed, for $k = 4$ we have that $\bar{a}_\alpha = a_\alpha$.

2.5. Sketch of graph construction

Thus far, we have derived sufficient conditions to minimize energies in the general $\mathscr{F}^k$ class. However, sufficient conditions do not constitute an algorithm. Fortunately, implicit in the derivations is the information required to construct the graph whose minimum cut yields the minimum of the desired energy function. In this section, we will sketch the means by which the relevant graph may be constructed.

First, let us return to the proof of Theorem 1. In this theorem, we treated the problem of optimizing an energy function of binary variables, where the function contains pairwise and singleton interactions. We showed, in proving the theorem, how to construct a graph whose minimum cut yields the minimum of the relevant energy. To wit, suppose that the energy is $E(x_1, \ldots, x_n) = \sum_{i,j}a_{ij}x_ix_j + \sum_i a_ix_i + C$. Then as we showed in the proof of the theorem, we can rewrite

$$\begin{aligned}E &= \sum_{i,j}a_{ij}x_ix_j + \sum_i a_ix_i + C\\ &= \sum_{i,j}-a_{ij}x_i(1-x_j) + \sum_i\left(\sum_j a_{ij}\right)x_i + \sum_i a_ix_i + C\\ &= \sum_{i,j}-a_{ij}x_i(1-x_j) + \sum_i b_ix_i + C\\ &= \sum_{i,j}-a_{ij}x_i(1-x_j) + \sum_{i:b_i\geqslant 0}b_ix_i + \sum_{i:b_i<0}|b_i|(1-x_i) + C'\end{aligned}$$

where $b_i = a_i + \sum_j a_{ij}$ and $C' = C - \sum_{i:b_i<0}|b_i|$. In order to minimize this latter expression, one builds a graph with $n + 2$ nodes: a node corresponding to each variable, plus source S and sink T nodes corresponding to 0 and 1 respectively. The weights are then as follows (as explained in the proof of the theorem): a weight of $-a_{ij}$ is placed on the edge running from node i to node j; a weight of b_i is placed on the edge between i and S, if $b_i > 0$; and a weight of $|b_i|$ is placed on the edge between i and T, if $b_i < 0$. If $a_{ij} \leqslant 0$, as is the case in all of our constructions, then the graph has entirely positive weights, and the energy may be minimized by finding the minimum cut of the graph so constructed.

Now, let us return to the case of $\mathscr{F}^k$ energies. In Eqs. (6) and (7), we show that the minimization of E may be written as the minimization of an energy with only pairwise and singleton terms, through the use of auxiliary variables; that is

$$\min_{x_i} E(x_i) = \min_{x_i,y_\alpha}\widetilde{E}(x_i, y_\alpha)$$

where

$$\widetilde{E}(x_i, y_\alpha) = \sum_{\alpha:|\alpha|>2}\sum_{\ell\in\alpha}a_\alpha x_\ell y_\alpha + \sum_{i,j}a_{ij}x_ix_j + L$$

(For the precise meaning of notation, the reader is urged to return to the discussion in Section 2.4.) But the foregoing discussion tells us precisely how to construct a graph whose minimum cut yields the minimum of such an energy (i.e., one with only pairwise and singleton terms). By applying this logic to the energy in question, it is straightforward to construct the relevant graph.

2.6. Submodularity

A well known fact from the theory of combinatorial optimization is that the class of submodular functions can be optimized in polynomial time [18]. This fact was noted in [16] (though not [15]), but we wish to add some further discussion of these functions here.

Suppose S is a set with n elements. If we denote by 2^S the power set of S, then a set-valued function $f : 2^S \to \mathbb{R}$ is said to be submodular if

$$f(X\cap Y) + f(X\cup Y) \leqslant f(X) + f(Y)\forall \quad X, Y \subset S$$

One can, of course, easily move from set-valued functions to binary-valued functions, by letting inclusion of element i in a set correspond to $x_i = 1$, and exclusion to $x_i = 0$. We wish to make two comments regarding the relationship between submodularity and the conditions described here:

1. The relationship between the conditions for $\mathscr{F}^k$ derived in (8) and submodularity is unknown, but the conditions are not the same. This can be clearly seen from the fact that the submodularity conditions always involve exactly four terms, whereas the inequalities in (8) can involve more.
2. It is not obvious from inspection as to how to specialize the submodularity conditions to classes of functions like $\mathscr{F}^k$; these conditions will look the same, no matter how many pixel are allowed to interact. (Of course, the number of such conditions applying may decrease, but the way in which this takes place is also not obvious from inspection.) The new conditions, by contrast, relate to precisely the function classes $\mathscr{F}^k$ which are relevant for computer vision; in many vision applications, the number of interacting pixels k is fixed. Thus, from a computer vision point-of-view, these conditions are important. For example, it is clear from the inequalities of (8) precisely which new inequalities get added as k increases.

Note that the results derived here bear a relationship to those from the pseudo-boolean optimization literature, see for example [1].

3. Introducing shape modelling into graph cuts

In this section, we will see how to apply the formalism from Section 2 to the modelling of shape. In particular, we are interested in the problem of introducing shape priors into segmentation algorithms. Segmentation has a rich history of being treated as combinatorial optimization problem, beginning with the work of Greig et al. [13]. We will begin our investigation in Section 3.1 by reviewing the segmentation algorithm of Boykov and Jolly [2]. In Section 3.2, we turn to the problem of formulating a shape prior term using the theory we have developed in Section 2. This term is then added to the Boykov–Jolly energy function, and experimental results of using the shape prior are shown in Section 3.3.

Note that the problem of introducing shape modelling and priors into segmentation has been well researched, and many practical and useful techniques already exist. Examples include [6,8,9,7,12,20]. Many of these methods are based on continuous optimization techniques, such as gradient descent, which commonly translate into geometric partial differential equations (i.e., curve evolutions). Due to the fact that the optimum sought is only local, there are generally little or no restrictions on the form of the objective function. As a result, the objective functions tend to more accurately capture the problem of shape modelling in segmentation than the objective function we introduce below. This, then, is the main advantage of most existing methods compared to the proposed method: more accurate objective functions.

Our contribution, then, should be seen for what it is: a way of framing the shape modelling in segmentation problem *which admits a global optimization technique*. As we have noted, other methods often work quite well in practice, despite the fact that they rely on local optimization. However, we believe that global optimization – when possible – is always preferable to local optimization, and that the proposed method should be seen as a first step in that direction.

3.1. A graph cut approach to segmentation

Boykov and Jolly [2] introduced a novel interactive method for segmentation. The idea is as follows: the user marks some pixels as being part of the object of interest, and some as lying outside the object i.e. within the background. Given these constraints, the algorithm tries to find the optimal segmentation such that these hard constraints are satisfied. In particular, a segmentation is scored according to the following criteria:

1. Each pixel inside the object is given a value according to whether its intensity matches the object's appearance model; low values represent better matches.
2. Each pixel in the background is given a value according to whether its intensity matches the appearance model of the background; low values represent better matches.
3. A pair of adjacent pixels, where one is inside the object and the other is outside, is given a value according to whether the two pixels have similar intensities; low values correspond to contrasting intensities (i.e. to an edge).

Note that the appearance models can be learned *a priori*, or they can be learned by examining the points selected by the user as hard constraints.

These considerations may be formalized into the following energy function:

$$E = \sum_{p\in\mathscr{P}} R_p(x_p) + \sum_{(p,q)\in\mathscr{N}} B_{pq}(x_p,x_q) = \sum_{p\in\mathscr{P}}[-\log P_b(I_p)(1-x_p) - \log P_o(I_p)x_p] + \lambda \sum_{(p,q)\in\mathscr{N}} \frac{e^{-(I_p-I_q)^2/2\sigma^2}}{\|p-q\|}[x_p(1-x_q)+(1-x_p)x_q]$$

where I_p is the intensity at pixel $p; P_b$ is the background probability model; P_o is the object (foreground) probability model; and $\mathscr{N}$ is the set of neighbouring pixels. The form of this energy function is easily seen to satisfy the conditions of Theorem 1, so a straightforward application of minimum cut may be applied to globally minimize E.

3.2. The shape prior

The Boykov–Jolly energy is most effective when the user provides good foreground and background seeds; these hard constraints then guide the segmentation. We now introduce a modification to this energy, in which no user intervention is required. Instead, the segmentation is guided by a shape model. Note that despite the existence of prior work which attempts to reconcile shape priors with a graph cut formalism, such as [11,17], this reconciliation is still considered an open problem. This is due to the fact that these prior approaches generally have a kind of "hybrid" structure: one part graph cuts, one part something else (usually some other kind of optimization). In what follows, we try to formulate the problem entirely within a graph cuts framework.

The new energy function may be written

$$E = \sum_{(p,q)\in\mathscr{N}} B_{pq}(x_p,x_q) + KE_{shape}$$

i.e., the boundary term from the Boykov–Jolly energy, along with a new shape term. Let us fix some notation. Let $\omega \subset \mathscr{P}$ be a region of the image plane; a shape is specified by such an ω, i.e., we take a shape to be not just the boundary of an object, but the entire object including its interior. A shape library is then a collection $\Omega = \{\omega\}$ of such shapes. In this case, we may write the shape energy as

$$E_{shape} = -\sum_{\omega\in\Omega} C_\omega \prod_{p\in\omega} x_p$$

where C_ω is a constant for each shape, whose value we discuss shortly. The shape energy rewards the selection of all pixels in a particular shape ω with a reduction in the energy equal to $-KC_\omega$.

A natural question arises. Suppose that ω_1 and ω_2 are both shapes in the shape library Ω. What is to prevent us from selecting the foreground to consist of $\omega_1 \cup \omega_2$? Such a choice would yield a reduction in energy equal to $-K(C_{\omega_1} + C_{\omega_2})$, so it may be energetically advantageous to make this choice. However, we would very

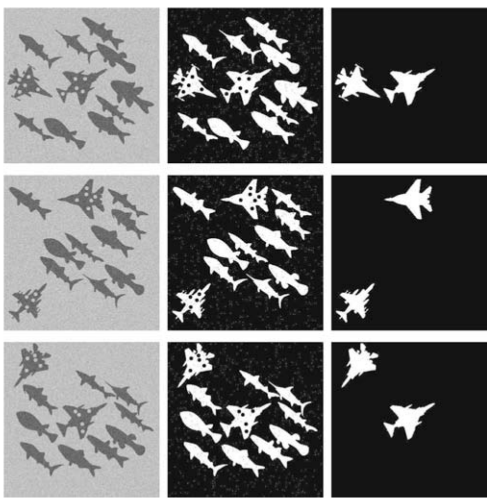

Fig. 1. Segmenting airplanes in clutter. Left column: original images. Middle column: segmentation without shape information. Right column: segmentation with shape information.

Fig. 2. Segmenting a variety of objects using shape information.

much like to avoid this choice, as the union of the two shapes is not a shape in the library – and in most cases looks nothing like the shape prior at all. To avoid this choice, we write C_ω as

$$C_\omega = \begin{cases} 1 & |\{p \in \omega : P_o(I_p) \geqslant P_b(I_p)\}| \geqslant \phi|\omega|, \\ 0 & \text{otherwise}. \end{cases}$$

where $0 \leqslant \phi \leqslant 1$ is a parameter to be chosen. The concept here is the following: if a region is such that the pixels in its interior satisfy the object appearance model, then the set $\{p \in \omega : P_o(I_p) \geqslant P_b(I_p)\}$ should form a large fraction of the entire region ω. (For concreteness, one may imagine $\phi \approx 0.6$, i.e. 60% of the pixels must match the object appearance model.) In this case, we set $C_\omega = 1$, and choosing ω leads to a reduction in energy of $-K$. If this is not the case – i.e., if most of the pixels within ω do not really match the object appearance model well – then $C_\omega = 0$, so that choosing ω does not lead to any decrease in the energy. Therefore, it is only energetically favourable to choose regions whose interiors match the object appearance model, which means that the above scenario, in which two overlapping regions are chosen, is unlikely to happen.

There is another natural question: why is the region term from the Boykov–Jolly energy left out in our new formulation? That is, our proposed energy function contains only the boundary term from the Boykov–Jolly energy, as well as the new shape prior term, but the region term from Boykov–Jolly is missing. The reason for this is straightforward: in formulating the shape weights C_ω, we have effectively included the region term of Boykov–Jolly *within* the shape prior. That is, unless the shape ω matches the object model, given by the distribution P_o, to a strong enough extent – i.e., at least a fraction ϕ of the pixels within ω must match the object model better than they match the background model – then adding the shape ω into the segmentation does not lower the energy function at all. Thus, the region term is effectively captured by the shape priors.

Finally, one may wonder what is to prevent the segmentation from choosing the union of all of the shapes ω with positive C_ω in the shape library? The answer, of course, is the boundary term; adding all of these shapes would likely lead to a high boundary cost, as the boundary would not correspond to a high contrast contour.[2] The boundary term plays a similar role in the Boykov–Jolly energy, particularly when no good background model is available.

Clearly, the new energy function has multi-pixel interactions, where the number of pixels in the relevant terms may easily be on the order of hundreds or thousands. However, it is not hard to verify that the new energy function satisfies the conditions of Eq. (8). Indeed, each region requires the introduction of exactly one extra variable, as in Eq. (5).

3.3. Experimental results

To demonstrate the validity of the approach described in Section 3.2, we have tested the algorithm on two sets of data. The first set consists of synthetic binary images which have been corrupted by Gaussian noise. The images contain both airplanes and fish; our goal is to segment the airplanes. Note that in addition to the Gaussian noise, the images contain an extra source of noise: the airplanes have fairly large holes. The set Ω consists of airplanes in various positions and orientations scattered throughout the image; we take $\phi = 0.55$. The left column of Fig. 1 shows the original images; the middle column the results of segmentation using Boykov–Jolly; and the right column the result of using the new energy with shape priors. As one would expect, the Boykov–Jolly segmentation, which does not use shape information, does not distinguish between planes and fish; by contrast, the segmentation using shape information successfully finds only the planes. Furthermore, by using the shape information, the segmentation is able to fill in the holes in the planes, as well as deal more successfully with the Gaussian noise.

Our second set of data consists of standard grayscale images. In each case, the image has an "obvious" foreground object superimposed on the background. Once again, the set Ω consists of the relevant object in various positions and orientations scattered throughout the image. Results are shown in Fig. 2. Although the segmentations are not perfect, they are generally quite good.

4. Conclusions

In this paper, we have contributed a set of sufficient conditions for the application of graph cut techniques to the minimization of discrete binary energy functions with k-wise pixel interactions. We have used these sufficient conditions to design an energy function which incorporates shape information into segmentation. Experimental results have shown the promise of this approach. A key direction for future research involves determining the applicability of the sufficient conditions to other energy functions of interest, such as those related to MRFs.

Acknowledgement

This work was supported in part by the US National Science Foundation, under Award IIS-0133144.

References

[1] E. Boros, P.L. Hammer, Pseudo-Boolean optimization, Discrete Applied Mathematics 123 (1–3) (2002) 155–225.
[2] Y. Boykov, M.P. Jolly, Interactive graph cuts for optimal boundary and region segmentation of objects in NDimages, in: International Conference on Computer Vision, vol. 1, 2001, pp. 105–112 .
[3] Y. Boykov, V. Kolmogorov,Computing geodesics and minimal surfaces via graph cuts, in: Proc. ICCV, 2003.
[4] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, in: Proc. ICCV, vol. 1, 1999, pp. 377–384.
[5] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (11) (2001) 1222–1239.
[6] T. Chan, W. Zhu, Level set based shape prior segmentation, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 2, 2005.
[7] D. Cremers, S.J. Osher, S. Soatto, Kernel density estimation and intrinsic alignment for shape priors in level set segmentation, International Journal of Computer Vision 69 (3) (2006) 335–351.
[8] D. Cremers, S. Soatto, A pseudo-distance for shape priors in level set segmentation, in: IEEE Workshop on Variational, Geometric and Level Set Methods in Computer Vision, 2003, pp. 169–176.

[2] In order to compute the boundary term at the border of the image, we zero-pad the image. This results in non-zero, and generally large, boundary terms at the border of the image. Thus, the trivial solution of choosing the entire image as the segmentation is ruled out.

[9] D. Cremers, N. Sochen, C. Schnorr, Towards recognition-based variational segmentation using shape priors and dynamic labeling, Lecture Notes in Computer Science (2003) 388–400.
[10] D. Freedman, P. Drineas, Energy minimization via graph cuts: settling what is possible, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVRP 2005), vol. 2, 2005.
[11] D. Freedman, T. Zhang, Interactive graph cut based segmentation with shape priors, in: Proc. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1–01, 2005, pp. 755–762.
[12] M. Gastaud, M. Barlaud, G. Aubert, Combining shape prior and statistical features for active contour segmentation, IEEE Transactions on Circuits and Systems for Video Technology 14 (5) (2004) 726–734.
[13] D.M. Greig, B.T. Porteous, A.H. Seheult, Exact maximum a posteriori estimation for binary images, Journal of the Royal Statistical Society Series B (Methodological) 51 (2) (1989) 271–279.
[14] V. Kolmogorov, R. Zabih, Multi-camera scene reconstruction via graph cuts, in: Proc. ECCV, 2002.
[15] V. Kolmogorov, R. Zabih, What energy functions can be minimized via graph cuts, in: Proc. ECCV, 2002.
[16] V. Kolmogorov, R. Zabih, What energy functions can be minimized via graph cuts?, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (2) (2004) 147–159
[17] MP Kumar, PHS Torr, A. Zisserman, OBJ CUT, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVRP 2005), vol. 1, 2005.
[18] L. Lovasz, Combinatorial Optimization: Algorithms and Complexity, Dover Publications, 1998.
[19] C.H. Papadimitriou, K. Steiglitz, An Algorithmic Theory of Numbers Graphs and Convexity, Soc for Industrial and Applied Math, 1986.
[20] M. Rousson, N. Paragios, Shape priors for level set representations, Lecture Notes in Computer Science (2002) 78–92.