Computer Vision and Image Understanding
journal homepage: www.elsevier.com

Linear Programming-based Submodular Extensions for Marginal Estimation

Pankaj Pansari[a], Chris Russell[b], M. Pawan Kumar[c]

[a] *University of Oxford*
[b] *University of Surrey, The Alan Turing Institute*
[c] *University of Oxford, The Alan Turing Institute*

ABSTRACT

Submodular extensions of an energy function can be used to efficiently compute approximate marginals via variational inference. The accuracy of the marginals depends crucially on the quality of the submodular extension. To identify accurate extensions for different classes of energy functions, we establish a relationship between the submodular extensions of the energy and linear programming (LP) relaxations for the corresponding MAP estimation problem. This allows us to (i) establish the worst-case optimality of the submodular extension for Potts model used in the literature; (ii) identify the worst-case optimal submodular extension for the more general class of metric labeling; (iii) efficiently compute the marginals for the widely used dense CRF model with the help of a recently proposed Gaussian filtering method; and (iv) propose an accurate submodular extension based on an LP relaxation for a higher-order diversity model. Using synthetic and real data, we show that our approach provides comparable upper bounds on the log-partition function to those obtained using tree-reweighted message passing (TRW) in cases where the latter is computationally feasible. Importantly, unlike TRW, our approach provides the first computationally tractable algorithm to compute an upper bound on dense CRF model with higher-order Potts potentials.

© 2019 Elsevier Ltd. All rights reserved.

1. Introduction

The desirable optimization properties of submodular set functions have been widely exploited in the design of approximate MAP estimation algorithms for discrete conditional random fields (CRFs) (Boykov et al., 2001; Kumar et al., 2011). Submodularity has also been recently used to design an elegant variational inference algorithm to compute the marginals of a discrete CRF by minimising an upper-bound on the log-partition function. In the initial work of Djolonga and Krause (2014), the energy of the CRF was restricted to be submodular. In a later work (Zhang et al., 2015), the algorithm was extended to handle more general Potts energy functions. The key idea there was to define a large ground set such that its subsets represent valid labelings, sublabelings or even incorrect labelings (these may assign two separate labels to a random variable and hence be invalid). Given the large ground set, it is possible to define a submodular set function whose value is equal to the energy of the CRF for subsets that specify a valid labeling of the model. We refer to such a set function as a *submodular extension* of the energy.

For a given energy function, there exists a large number of possible submodular extensions. The accuracy of the variational inference algorithm depends crucially on the

e-mail: pankaj@robots.ox.ac.uk (Pankaj Pansari)

choice of the submodular extension. Yet, previous work has largely ignored the question of identifying accurate extensions for different energy classes. Indeed, the difficulty of identifying submodular extensions of general energy functions could be a major reason why the experiments of Zhang et al. (2015) were restricted to the special case of models specified by the Potts energy functions.

In this work, we establish a hitherto unknown connection between the submodular extension of the Potts model proposed by Zhang et al. (2015), and the objective function of an accurate linear programming (LP) relaxation of the corresponding MAP estimation problem (Kleinberg and Tardos, 2002). Specifically, the Lovasz extension of a submodular extension can be shown to be an objective function for the LP relaxation. This connection has four important practical consequences. First, it establishes the optimality of the submodular extension of the Potts model, via the tightest LP relaxation (among first-level relaxations of the Sherali-Adams hierarchy (Sherali and Adams, 1990)) under UGC-hardness assumptions (Manokaran et al., 2008). Second, it provides an accurate submodular extension of the hierarchical Potts model, via the LP relaxation of the corresponding MAP estimation problem proposed by Kleinberg and Tardos (2002). Since any metric can be accurately approximated as a mixture of hierarchical Potts models (Bartal, 1996, 1998), this result also provides a computationally feasible algorithm for estimating the marginals for metric labeling. Third, it establishes the equivalence between the subgradient of the LP relaxation and the conditional gradient of the problem of minimising the upper bound of the log-partition. This allows us to employ the widely used dense CRF, since a subgradient of its LP relaxation can be efficiently computed using a recently proposed modified Gaussian filtering algorithm (Ajanthan et al., 2017). As a consequence, we provide the first efficient algorithm to compute an upper bound of the log-partition function of dense CRFs. This provides complementary information to the popular mean-field inference algorithm for dense CRFs, which computes a lower bound on the log-partition (Koltun and Krahenbuhl, 2011). Fourth, we obtain an accurate submodular extension for a higher-order diversity model based on an LP relaxation. Higher-order models can capture interesting properties of the image that cannot be expressed using pairwise models (Vineet et al., 2014; Kohli et al., 2007). We show that our upper-bounds on synthetic problems are comparable to those from tree reweighted message passing (TRW) (Wainwright et al., 2005) for the case of sparse CRFs. Unlike our approach, TRW is computationally infeasible for dense CRFs, thereby limiting its use in practice. Using pairwise dense CRF models, we perform stereo matching on standard data sets and obtain better results than Koltun and Krahenbuhl (2011). We also perform semantic segmentation on the MSRC-21 dataset using a combination of dense pairwise and higher-order diversity model. The complete code is available at `https://github.com/pankajpansari/denseCRF`.

2. Preliminaries

We now introduce the notation and definitions that we make use of in the remainder of the paper.

Submodular Functions: Given a ground set $U = \{1, \dots, N\}$, let us denote by 2^U its power set. A set function $F : 2^U \to \mathbb{R}$ is *submodular* if, for all subsets $A, B \subseteq U$

$$F(A \cup B) + F(A \cap B) \leq F(A) + F(B). \tag{1}$$

The set function F is *modular* if there exists $\mathbf{s} \in \mathbb{R}^N$ such that $F(A) = \sum_{k \in A} s_k \ \forall \ A \subseteq 2^U$. Henceforth, we will use the shorthand $s(A)$ to denote $\sum_{k \in A} s_k$.

Extended Polymatroid: Associated with any submodular function F is a special polytope known as the *extended polymatroid* defined as

$$EP(F) = \{\mathbf{s} \in \mathbb{R}^N | \ \forall A \subseteq U : s(A) \leq F(A)\}, \tag{2}$$

where $\mathbf{s}$ denotes the modular function $s(.)$ represented as a vector (Bach, 2013).

Lovasz Extension For a given set function F with $F(\emptyset) = 0$, the value of its Lovasz extension $f(\mathbf{w}) : \mathbb{R}^N \to \mathbb{R}$ is defined as follows (Bach, 2013) : order the components of $\mathbf{w}$ in decreasing order such that $w_{j_1} \geq w_{j_2} \geq \cdots \geq w_{j_N}$, where $(j_1, j_2, \ldots, j_N)$ is the corresponding permutation of the indices. Then,

$$f(\mathbf{w}) = \sum_{k=1}^{N} w_{j_k} \left(F(\{j_1, \ldots, j_k\}) - F(\{j_1, \ldots, j_{k-1}\}) \right). \tag{3}$$

The function f is an extension because it equals F on the vertices of the unit cube. That is, for any $A \subseteq V$, $f(\mathbf{1}_A) = F(A)$, where $\mathbf{1}_A$ is the 0-1 indicator vector corresponding to the elements of A.

Property 1. *By Edmond's greedy algorithm (Edmonds, 1970), if* $\mathbf{w} \geq 0$ *(non-negative elements),*

$$f(\mathbf{w}) = \max_{\mathbf{s} \in EP(F)} \langle \mathbf{w}, \mathbf{s} \rangle. \tag{4}$$

Property 1 implies that an LP over $EP(F)$ can be solved by computing the value of the Lovasz extension using equation (3).

Property 2. *The Lovasz extension f of a submodular function F is a convex piecewise linear function.*

Property 2 holds since $f(\mathbf{w})$ is the pointwise maximum of linear functions according to equation (4).

CRF and Energy Functions A CRF is defined as a graph on a set of random variables $\mathcal{X} = \{X_1, \ldots, X_N\}$ related by a set of edges $\mathcal{N}$. We wish to assign every variable X_a one of the labels from the set $\mathcal{L} = \{1, 2, \ldots, L\}$. The quality of a labeling $\mathbf{x}$ is given by an energy function defined as

$$E(\mathbf{x}) = \sum_{a \in \mathcal{X}} \phi_a(x_a) + \sum_{(a,b) \in \mathcal{N}} \phi_{ab}(x_a, x_b), \tag{5}$$

where ϕ_a and ϕ_{ab} are the unary and pairwise potentials respectively. In computer vision, we often think of $\mathcal{X}$ as arranged on a grid. A *sparse CRF* has $\mathcal{N}$ defined by 4-connected or 8-connected neighbourhood relationships. In a *dense CRF*, on the other hand, every variable is connected to every other variable.

We can augment the above pairwise models with higher-order potentials. Let $\mathcal{C}$ be the set of cliques on subgroups of variables. These cliques can, for instance, be the set of superpixels from a clustering method, such as k-means or mean-shift Comaniciu and Meer (2002). Also, let $\mathbf{x}_c$ be the component of $\mathbf{x}$ formed by elements in clique c. The energy function now also contains higher-order potentials

$$E(\mathbf{x}) = \sum_{a \in \mathcal{X}} \phi_a(x_a) + \sum_{(a,b) \in \mathcal{N}} \phi_{ab}(x_a, x_b) + \sum_{c \in \mathcal{C}} \phi_c(\mathbf{x}_c), \tag{6}$$

The energy function can be interpreted as defining a probability distribution $P(\mathbf{x})$ as:

$$P(\mathbf{x}) = \begin{cases} \frac{1}{Z} \exp(-E(\mathbf{x})) & \text{if } \mathbf{x} \in \mathcal{L}^N, \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

The normalization factor $Z = \sum_{\mathbf{x} \in \mathcal{L}^N} \exp(-E(\mathbf{x}))$ is known as the *partition function.*

Inference There are two inference problems in CRFs:

(i) Marginal inference: We want to compute the marginal probabilities $P(X_a = i)$ for every $a = 1, 2, \ldots, N$ and $i = 1, 2, \ldots, L$.

(ii) MAP inference: We want to find a labeling with the minimum energy, that is, $\min_{\mathbf{x} \in \mathcal{L}^N} E(\mathbf{x})$. Equivalently, MAP inference finds the mode of $P(\mathbf{x})$.

3. Review: Variational Inference Using Submodular Extensions

We now summarise the inference method of Zhang et al. (2015) which made use of a submodular extension.

Submodular Extensions A submodular extension is defined using a ground set such that some of its subsets correspond to valid CRF labelings. Note that not every subset needs to represent a valid labeling - some of them could correspond to incomplete or invalid labelings. To obtain such an extension, we need an encoding scheme which gives the sets corresponding to valid CRF labelings.

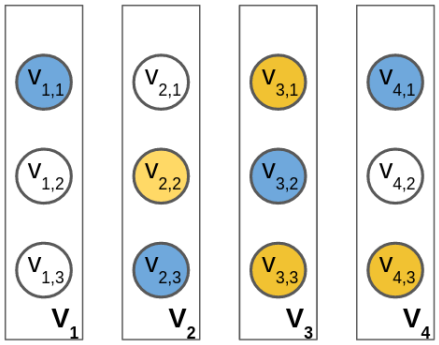


Fig. 1: *Illustration of* 1*-of-L encoding used in Zhang et al. (2015) with 4 variables and 3 labels. The blue labeling, corresponding to* $X_1 = 1, X_2 = 3, X_3 = 2, X_4 = 1$*, is valid. The yellow labeling, corresponding to* $X_2 = 2, X_3 = 1, 3, X_4 = 3$*, is invalid since* X_3 *has been assigned multiple labels and* X_1 *has been assigned none.*

One example of an encoding scheme is the 1-of-L encoding, illustrated in Figure 1. Let each variable X_a take one of L possible labels. In this scheme, we represent the set of possible assignments for X_a by the set $V_a = \{v_{a1}, v_{a2}, \ldots, v_{aL}\}$. If X_a is assigned label i, then we select the element v_{ai}. Extending to all variables, our ground set becomes $V = \cup_{a=1}^{N} V_a$. A valid assignment $A \subseteq V$ assigns each variable exactly one label, that is, $|A \cap V_a| = 1$ for all V_a. We denote the set of valid assignments by $\mathcal{M}$ where $\mathcal{M} = \cap_{a=1}^{N} \mathcal{M}_a$ and $\mathcal{M}_a = \{A : |A \cap V_a| = 1\}$.

Using our ground set V, we can define a submodular function F which equals $E(\mathbf{x})$ for all sets corresponding to valid labelings, that is, $F(A_{\mathbf{x}}) = E(\mathbf{x}),\ \mathbf{x} \in \mathcal{L}^N$ where $A_{\mathbf{x}}$ is the set encoding of $\mathbf{x}$. We call such a function F a *submodular extension* of $E(\mathbf{x})$.

Upper-Bound on Log-Partition Using a submodular extension F and given any $\mathbf{s} \in EP(F)$, we can obtain an upper-bound on the partition function as

$$\mathcal{Z} = \sum_{A \in \mathcal{M}} \exp(-F(A)) \leq \sum_{A \in \mathcal{M}} \exp(-s(A)), \qquad (8)$$

where $\mathcal{M}$ is the set of valid labelings. The upper-bound is the partition function of the distribution $Q(A) \propto \exp(-s(A))$, which factorises fully because $s(.)$ is modular. We can minimise this upper-bound, and the corresponding optimum $\mathbf{s}$ can help us obtain good approximate marginals of the distribution $P(\cdot)$. After taking logs for easy factorisaton, we can equivalently write our optimisation problem as

$$\min_{\mathbf{s} \in EP(F)} g(\mathbf{s}), \text{where } g(\mathbf{s}) = \log \sum_{A \in \mathcal{M}} \exp(-s(A)). \qquad (9)$$

Conditional Gradient Algorithm The conditional gradient algorithm (Algorithm 1) (Frank and Wolfe, 1956) is a good candidate for solving problem (9) due to two reasons. First, problem (9) is convex. Second, as solving an LP over EP(F) is computationally tractable (property 1), the conditional gradient can be found efficiently. The algorithm starts with an initial solution $\mathbf{s_0}$ (line 1). At each iteration, we compute the conditional gradient $\mathbf{s}^*$ (line 3), which minimises the linear approximation $g(\mathbf{s}_k) + \nabla g(\mathbf{s}_k)^T(\mathbf{s} - \mathbf{s}_k)$ of the objective function. Finally, $\mathbf{s}$ is updated by either (i) fixed step size schedule, as in line 7 of algorithm 1, or (ii) by doing a line search and setting $\mathbf{s}_{k+1} = \min_{0 \leq \gamma \leq 1} g(\gamma \mathbf{s}^* + (1-\gamma)\mathbf{s}_k)$.

Algorithm 1 Upper Bound Minimisation using Conditional Gradient Descent

1: Initialize $\mathbf{s} = \mathbf{s}_0 \in EP(F)$
2: **for** k = 1 to `MAX_ITER` **do**
3: $\quad \mathbf{s}^* = \text{argmin}_{\mathbf{s} \in EP(F)} \langle \nabla g(\mathbf{s}_k), \mathbf{s} \rangle$
4: $\quad$ **if** $\langle \mathbf{s}^* - \mathbf{s}_k, \nabla g(\mathbf{s}_k) \rangle \leq \epsilon$ **then**
5: $\quad\quad$ **break**
6: $\quad$ **end if**
7: $\quad \mathbf{s}_{k+1} = \mathbf{s}_k + \gamma(\mathbf{s}^* - \mathbf{s}_k)$ with $\gamma = 2/(k+2)$
8: **end for**
9: **return** $\mathbf{s}$

4. Overview: Accurate Submodular Extensions from LP Relaxations

Different extensions F change the domain in problem (9), leading to different upper bounds on the log-partition function. How does one come up with accurate extensions for different classes of CRF energy? Is it possible to identify optimal extensions which yield the tightest bound?

If we introduce a temperature parameter in $P(\mathbf{x})$ (equation (7)) by using $E(\mathbf{x})/T$ and decrease T, the resulting distribution starts to peak more sharply around its mode. We assume that $P(\mathbf{x})$ is unimodal, and there aren't multiple solutions with the same minima. As $T \to 0$, marginal estimation becomes the same as MAP inference since the resulting distribution $P^0(\mathbf{x})$ has mass 1 at its mode $\mathbf{x}^*$ and is 0 everywhere else. Given the MAP solution $\mathbf{x}^*$, one can compute the marginals as $P(X_i = j) = [x_i^* = j]$, where [.] is the Iverson bracket. We point out that the notion of a temperature parameter T in a probability distribution and indeed variational methods for approximating complicated distributions have their origins in statistical physics literature - the interested reader is referred to MacKay (2003, chapter 33). Motivated by this connection, we ask if one can introduce a temperature parameter to our problem (9) and transform it to an LP relaxation in the limit $T \to 0$? We can then use accurate LP relaxations of MAP problems known in literature to find good submodular extensions for different classes of energy functions.

We answer this question in the affirmative. Specifically, in the following two sections we show how one can select the set encoding and submodular extension to convert problem (9) to accurate LP relaxations for Potts, hierarchical Potts and higher-order diversity models. When the LP relaxation has tightness guarantees, we obtain worst-case optimal submodular extensions - a notion we elucidate shortly.

In this work, we focus on obtaining submodular extensions with closed-form analytical expressions for different classes of energy functions. Formally, for a specific energy class (such as the Potts model) $\mathcal{E}$, we derive a family of submodular functions $\mathcal{F}(.)$. Given an instance of the energy function $E(.)$ from the class $\mathcal{E}$, the corresponding submodular extension is $\mathcal{F}(E)$.

5. Worst-case Optimal Submodular Extensions

Worst-case Optimality

Our extensions are derived from LP relaxations belonging to the first-level of the Sherali-Adams hierarchy. This level yields LP relaxations having $\mathcal{O}(N)$ relaxed variables for N variables in the CRF. In principle, these LP relaxations can be made tighter by introducing more variables and constraints, thereby moving to higher levels in the hierarchy ($\mathcal{O}(N^k)$ relaxed variables for k-th level). However, in practice, working with these higher-order relaxations is computationally infeasible for large-scale vision problems. In the discussion that follows, we assume that the LP relaxations belong to the first-level of Sherali-Adams hierarchy. When the LP relaxation is the tightest possible, the extension family $\mathcal{F}_{opt}$ we obtain is *worst-case optimal*. That is, there does not exist another submodular extension family $\mathcal{F}$ that gives a tighter upper bound for problem (9) than $\mathcal{F}_{opt}$ for all instances of the energy function in $\mathcal{E}$. Alternatively, for any other submodular extension family $\mathcal{F}$, we can find at least one instance of energy function $E(.)$ such that $\mathcal{F}_{opt}$ results in a tighter upper-bound. Formally, $\mathcal{F}_{opt}$ is worst-case optimal if

$$\nexists \mathcal{F} : \min_{\mathbf{s} \in EP(\mathcal{F}(E))} g(\mathbf{s}) \leq \min_{\mathbf{s} \in EP(\mathcal{F}_{opt}(E))} g(\mathbf{s}) \ \ \forall\, E(.) \in \mathcal{E}. \tag{10}$$

Note that the notion of worst-case optimality does not guarantee that the extension is best for every instance of the energy function. It may be possible to solve an optimisation problem to obtain the best extension for every given instance of energy $E(.)$. However, in this paper, we do not take that approach, and instead provide a general solution for a given class of energy functions.

In subsection 5.1, we prove the worst-case optimality of the submodular extension used in literature for the Potts model. In subsection 5.2, we obtain worst-case optimal submodular extension for the more general hierarchical Potts model. Finally, in subsection 5.3, we provide a way to make our inference algorithm efficient for the dense CRF model. Using our approach, the inference algorithm

has linear time-complexity per iteration in the number of variables and labels.

5.1. Pairwise Potts Model

The pairwise Potts model, also known as the uniform metric, specifies the pairwise potentials $\phi_{ab}(x_a, x_b)$ in equation (5) as follows:

$$\phi_{ab}(x_a, x_b) = w_{ab} \cdot [x_a \neq x_b], \tag{11}$$

where w_{ab} is the weight associated with edge (a, b). There are no higher-order potential terms in this model as in equation (5). We note that the Potts model is non-submodular according to the popular encoding used in Ishikawa (2003). However, we use the 1-of-L encoding to construct the submodular extension.

Tightest LP Relaxation Before describing our set encoding and submodular extension, we briefly outline the LP relaxation of the corresponding MAP estimation problem. To this end, we define indicator variables y_{ai} which equal 1 if $X_a = i$, and 0 otherwise. The following is the tightest possible LP relaxation (among first-level LP relaxations) for Potts model in the worst-case, assuming the Unique Games Conjecture is true (see (EM-LP) for metric labeling in Manokaran et al. (2008) and its connection to (P-LP) in Chekuri et al. (2004)):

$$\begin{aligned} \text{(P-LP)} \quad \min_{\mathbf{y}} \; & E(\mathbf{y}) = \sum_a \sum_i \phi_a(i) y_{ai} + \\ & \sum_{(a,b)\in\mathcal{N}} \sum_i \frac{w_{ab}}{2} \cdot |y_{ai} - y_{bi}| \\ \text{s.t} \quad & \mathbf{y} \in \Delta. \end{aligned} \tag{12}$$

The set Δ is the union of N probability simplices:

$$\Delta = \{\mathbf{y}_a \in \mathbb{R}^L | \mathbf{y}_a \geq 0 \text{ and } \langle \mathbf{1}, \mathbf{y}_a \rangle = 1\}, \tag{13}$$

where $\mathbf{y}$ is the vector of all variables and $\mathbf{y}_a$ is the component of $\mathbf{y}$ corresponding to X_a.

Set Encoding We choose to use the 1-of-L encoding for Potts model as described in Section 3. With the encoding

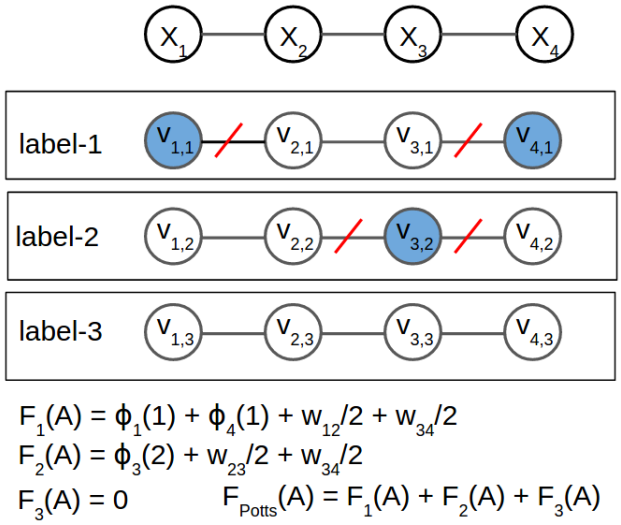


Fig. 2: *An illustration of the worst-case optimal submodular extension for Potts model for a chain graph of 4 variables, each of which can take 3 labels. The figure shows how to compute extension for* $A = \{v_{1,1}, v_{4,1}, v_{3,2}\}$.

scheme for Potts model above, $g(\mathbf{s})$ can be factorised and problem (9) can be rewritten as:

$$\min_{\mathbf{s}\in EP(F)} \quad \sum_{a=1}^{N} \log \sum_{i=1}^{L} \exp(-s_{ai}). \tag{14}$$

(See Remark 1 in appendix A for proof).

We note that the gradient $\nabla g(\mathbf{s})$ can be computed efficiently as $[\nabla g(\mathbf{s})]_{ai} = -\exp^{-s_{ai}} / \sum_{i=1}^{L} \exp^{-s_{ai}}$, these being the marginals of the fully-factorised distribution $Q(A)$.

Marginal Estimation with Temperature We now introduce a temperature parameter $T > 0$ to problem (14) which divides $E(\mathbf{x})$, or equivalently divides $\mathbf{s}$ belonging to $EP(F)$. Also, since $T > 0$, we can multiply the objective by T leaving the problem unchanged. Without changing the solution, we can transform problem (14) as follows

$$\min_{\mathbf{s}\in EP(F)} \quad g_T(\mathbf{s}) = \sum_{a=1}^{N} T \cdot \log \sum_{i=1}^{L} \exp(-\frac{s_{ai}}{T}). \tag{15}$$

Worst-case Optimal Submodular Extension We now connect our marginal estimation problem (9) with LP relaxations using the following proposition.

Proposition 1. *In the limit* $T \to 0$*, problem (15) becomes:*

$$-\min_{\mathbf{y}\in\Delta} f(\mathbf{y}) \tag{16}$$

where $f(.)$ *is the Lovasz extension of* $F(.)$.

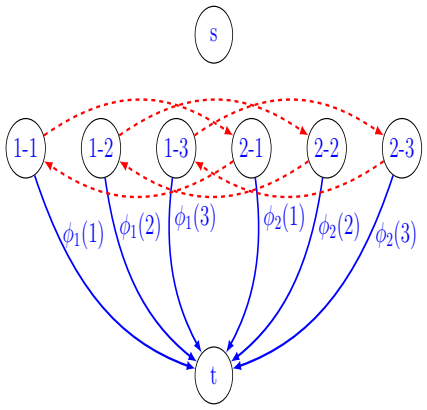


Fig. 3: *An st-graph specifying the worst-case optimal submodular extension for Potts model for 2 variables with 3 labels each and connected to each other. There is a node for each variable and each label, that is, for all elements of the ground set. The nodes have been labeled as 'variable-label', hence node 1-1 represents the element v_{11} and so on. The solid blue arcs model the unary potentials, and the dotted red arcs represent the pairwise potentials. Each dotted red arc has weight $w_{12}/2$.*

Proof. In the limit $T \to 0$, we can rewrite problem (15) as

$$\min_{\mathbf{s}\in EP(F)} \quad \sum_{a=1}^{N} \max_{i}(-s_{ai}). \tag{17}$$

In vector form, the problem becomes

$$\min_{\mathbf{s}\in EP(F)} \max_{\mathbf{y}\in\Delta} -\langle \mathbf{y}, \mathbf{s}\rangle \tag{18}$$

$$= - \max_{\mathbf{s}\in EP(F)} \min_{\mathbf{y}\in\Delta} \langle \mathbf{y}, \mathbf{s}\rangle \tag{19}$$

where Δ is the domain as defined as in equation (13). By the minimax theorem Boyd and Vandenberghe (2004) for LP, we can reorder the terms:

$$- \min_{\mathbf{y}\in\Delta} \max_{\mathbf{s}\in EP(F)} \langle \mathbf{y}, \mathbf{s}\rangle \tag{20}$$

Recall that $\max_{\mathbf{s}\in EP(F)}\langle \mathbf{y}, \mathbf{s}\rangle$ is the value of the Lovasz extension of F at $\mathbf{y}$, that is, $f(\mathbf{y})$. Hence, as $T \to 0$, the marginal inference problem converts to minimising the Lovasz extension under the simplices constraint:

$$- \min_{\mathbf{y}\in\Delta} f(\mathbf{y}) \tag{21}$$

□

The problem (21) is equivalent to an LP relaxation of the corresponding MAP estimation problem (see Lemma 1 in appendix). Indeed, $g_T(\mathbf{s})$ in problem (15) becomes the objective function of an LP relaxation in the limit $T \to 0$. We seek to obtain the worst-case optimal submodular extension by making $g_T(\mathbf{s})$ same as the objective of (P-LP) as $T \to 0$. We note that problems (15) and (14) become equivalent at $T = 1$, and our worst-case optimality guarantee was derived in the limit $T \to 0$. However, it is possible to derive such a guarantee only in the limit case, and we claim the worst-case optimality in this sense.

The question now becomes how to recover the worst-case optimal submodular extension using $E(\mathbf{y})$. The following proposition answers this question.

Proposition 2. *The worst-case optimal submodular extension for Potts model is given by $F_{Potts}(A) = \sum_{i=1}^{L} F_i(A)$, where*

$$F_i(A) = \sum_{a} \phi_a(i)[|A \cap \{v_{ai}\}| = 1] + \sum_{(a,b)\in\mathcal{N}} \frac{w_{ab}}{2} \cdot [|A \cap \{v_{ai}, v_{bi}\}| = 1] \tag{22}$$

Also, $E(\mathbf{y})$ in (P-LP) is the Lovasz extension of F_{Potts}.

Proof. Since F_{Potts} is sum of Ising models F_i, we first focus on a particular label i and then generalize. Consider a graph with only two variables X_a and X_b with an edge between them. The ground set in this case is $\{v_{ai}, v_{bi}\}$. Let the corresponding relaxed indicator variables be $\mathbf{y} = \{y_{aj}, y_{bj}\}$, such that $y_{ai}, y_{bi} \in [0, 1]$ and assume $y_{ai} > y_{bi}$. The Lovasz extension in this case is:

$$\begin{aligned} f(\mathbf{y}) &= y_{ai} \cdot [F_i(\{v_{ai}\}) - F_i(\{\})] \\ &\quad + y_{bi} \cdot [F_i(\{v_{ai}, v_{bi}\}) - F_i(\{v_{ai}\})] \\ &= y_{ai} \cdot [\Big(\phi_a(j) + \frac{w_{ab}}{2}\Big) - 0] + y_{bi} \cdot [(\phi_a(j) + \phi_b(j)) \\ &\quad - \Big(\phi_a(j) + \frac{w_{ab}}{2}\Big)] \\ &= \phi_a(j) \cdot y_{ai} + \phi_b(j) \cdot y_{bi} + \frac{w_{ab}}{2} \cdot (y_{ai} - y_{bi}) \end{aligned} \tag{23}$$

In general for both orderings of y_{ab} and y_{bi}, we can write

$$f(\mathbf{y}) = \phi_a(j) \cdot y_{ai} + \phi_b(j) \cdot y_{bi} + \frac{w_{ab}}{2} \cdot |y_{ai} - y_{bi}| \tag{24}$$

Extending the Lovasz extension of equation (24) to all variables and all labels gives us $E(\mathbf{y})$ in (P-LP).

□

Proposition 2 paves the way for us to identify the worst-case optimal extension for hierarchical Potts model. Figure 2 shows an example where we compute $F_{Potts}(A)$ for a small graph and a given set A representing an invalid labeling. As an alternate representation, Figure 3 shows the *st*-graph corresponding to F_{Potts} for a small instance. Each *st*-cut corresponds to a labeling, and the cost of each cut matches F_{Potts} (subsection 4.1 of Wang et al. (2013)).

5.2. Hierarchical Potts Model

Potts model imposes the same penalty for unequal assignment of labels to neighbouring variables, regardless of the label dissimilarity. In some scenarios, a more natural approach is to vary the penalty based on how different the labels are. A hierarchical Potts model (Kleinberg and Tardos, 2002) permits this by specifying the distance between labels using a tree with the following properties:

1. The vertices are of two types: (i) the leaf nodes representing labels, and (ii) the non-leaf nodes, except the root, representing meta-labels.

2. The lengths of all the edges from a parent to its children are the same.

3. The lengths of the edges along any path from the root to a leaf decreases by a factor of at least $r \geq 2$ at each step.

4. The metric distance between nodes of the tree is the sum of the edge lengths on the unique path between them.

A subtree T of an hierarchical Potts model is a tree comprising all the descendants of some node v (not root). Given a subtree T, l_T denotes the length of the tree-edge leading upward from the root of T and $L(T)$ denotes the set of leaves of T. We call the leaves of the tree as labels and all other nodes of the tree expect the root as *meta-labels*. Figure 4 illustrates the notations in the context of a hierarchical Potts model.

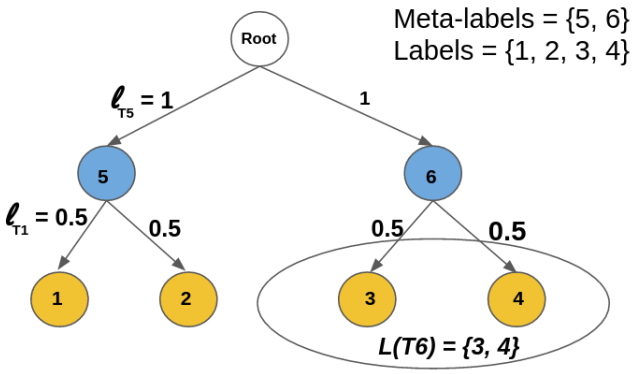


Fig. 4: *A hierarchical Potts model instance illustrating notations with 2 meta-labels (blue) and 4 labels (yellow). All labels are at the same level. $r = 2$, that is, edge-length decreases by 2 at each level. Also, distance between labels 1 and 3 is $d(1, 3) = 0.5 + 1 + 1 + 0.5 = 3$.*

We note that the hierarchical Potts model is non-submodular according to the popular encoding used in Ishikawa (2003). However, we use the 1-of-L encoding to construct its submodular extension.

Tightest LP Relaxation We use the same indicator variables y_{ai} that were employed in the LP relaxation of Potts model. Let $y_a(T) = \sum_{i \in L(T)} y_{ai}$. The following LP relaxation is the tightest known (among the LP relaxations of first-level Sherali-Adams hierarchy) for hierarchical Potts model in the worst-case, assuming the Unique Games Conjecture is true (Manokaran et al., 2008)

$$\text{(T-LP)} \quad \min_{\mathbf{y}} \ \widetilde{E}(\mathbf{y}) = \sum_a \sum_i \phi_a(i) y_{ai} + \sum_{(a,b) \in \mathcal{N}} w_{ab} \sum_T l_T \cdot |y_a(T) - y_b(T)|$$
$$\text{such that} \quad \mathbf{y} \in \Delta. \tag{25}$$

where we sum T over all subtrees. The set Δ is the same domain as defined in equation (13).

Transformed Tightest LP Relaxation We take (T-LP) and rewrite it using indicator variables for all labels and meta-labels. Let $\mathcal{R}$ denote the set of all labels and meta-labels, that is, all nodes in the tree apart from the

root. Also, let $\mathcal{L}$ denote the set of labels, that is, the leaves of the tree. Let T_i denote the subtree which is rooted at the i-th node. We introduce an indicator variable $z_{ai} \in \{0, 1\}$

$$z_{ai} = \begin{cases} y_{ai} & \text{if } i \in \mathcal{L} \\ y_a(T_i) & \text{if } i \in \mathcal{R} - \mathcal{L} \end{cases} \tag{26}$$

We need to extend the definition of unary potentials to the expanded label space as follows:

$$\text{where } \phi'_a(i) = \begin{cases} \phi_a(i) & \text{if } i \in \mathcal{L} \\ 0 & \text{if } i \in \mathcal{R} - L \end{cases} \tag{27}$$

We can now rewrite problem (25) in terms of new indicator variables z_{ai}:

$$\begin{aligned} \text{(T-LP-FULL)} \quad \min \widetilde{E}(\mathbf{z}) = &\sum_{i \in \mathcal{R}} \sum_{a \in \mathcal{X}} \phi'_a(i) \cdot z_{ai} + \\ &\sum_{i \in \mathcal{R}} \sum_{(a,b) \in \mathcal{N}} w_{ab} \cdot l_{T_i} \cdot |z_{ai} - z_{bi}| \\ \text{such that} \quad \mathbf{z} \in \Delta' \end{aligned} \tag{28}$$

where Δ' is the convex hull of the vectors satisfying

$$\sum_{i \in \mathcal{L}} z_{ai} = 1, \;\; z_{ai} \in \{0, 1\} \;\; \forall a \in \mathcal{X}, i \in \mathcal{L} \tag{29}$$

$$\text{and } \; z_{ai} = \sum_{j \in L(T_i)} z_{aj}. \;\; \forall a \in \mathcal{X}, i \in \mathcal{R} - \mathcal{L} \tag{30}$$

Constraint (30) ensures consistency among labels and meta-labels, that is, if a label is assigned then all the meta-labels which lie on the path from the root to the label should be assigned as well.

Set Encoding For any variable X_a, let the set of possible assignment of labels and meta-labels be the set $V_a = \{v_{a1}, \ldots, v_{aM}\}$, where M is the total number of nodes in the tree except the root. Our ground set is $V = \cup_{a=1}^{N} V_a$ of size $N \cdot M$.

A consistent labeling of a variable assigns it one label, and all meta-labels on the path from root to the label. Let us represent the set of consistent assignments for X_a by the set $P_a = \{p_{a1}, \ldots, p_{aL}\}$, where p_{ai} is the collection of elements from V_a for label i and all meta-labels on the path from root to label i. The set of valid labelings $A \subseteq V$ assigns each variable exactly one consistent label. This constraint can be formally written as $\mathcal{M} = \cap_{a=1}^{N} \mathcal{M}_a$ where $\mathcal{M}_a$ has exactly one element from P_a. Let s'_{ai} be the sum of the components of $\mathbf{s}$ corresponding to the elements of p_{ai}, that is,

$$s'_{ai} = \sum_{t \in p_{ai}} s_t. \tag{31}$$

Using our encoding scheme, we rewrite problem (9) as:

$$\min_{\mathbf{s} \in EP(F)} \quad \sum_{a=1}^{N} \log \sum_{i=1}^{L} \exp(-s'_{ai}). \tag{32}$$

Marginal Estimation with Temperature Similar to Potts model, we now introduce a temperature parameter $T > 0$ to problem (32). The transformed problem becomes

$$\min_{\mathbf{s} \in EP(F)} \quad g_T(\mathbf{s}) = \sum_{a=1}^{N} T \cdot \log \sum_{i=1}^{L} \exp(-\frac{s'_{ai}}{T}). \tag{33}$$

Worst-case Optimal Submodular Extension The following proposition connects the marginal estimation problem (9) with LP relaxations:

Proposition 3. *In the limit $T \to 0$, problem (33) becomes:*

$$-\min_{\mathbf{z} \in \Delta'} f(\mathbf{z}) \tag{34}$$

(Proof in appendix).

The above problem is equivalent to an LP relaxation of the corresponding MAP estimation problem (see Lemma 2 in appendix). Hence, $g_T(\mathbf{s})$ becomes the objective function of an LP relaxation in the limit $T \to 0$. We seek to make this objective same as $\widetilde{E}(\mathbf{z})$ of (T-LP-FULL) in the limit $T \to 0$. The question now becomes how to recover the worst-case optimal submodular extension from $\widetilde{E}(\mathbf{z})$.

Proposition 4. *The worst-case optimal submodular extension for hierarchical Potts model is given by $F_{hier}(A) = \sum_{i=1}^{M} F_i(A)$, where*

$$\begin{aligned} F_i(A) = &\sum_{a} \phi'_a(i)[|A \cap \{v_{ai}\}| = 1] + \\ &\sum_{(a,b) \in \mathcal{N}} w_{ab} \cdot l_{T_i} \cdot [|A \cap \{v_{ai}, v_{bi}\}| = 1] \end{aligned} \tag{35}$$

Also, $\widetilde{E}(\mathbf{z})$ in (T-LP-FULL) is the Lovasz extension of F_{hier}. (Proof in appendix)

Since any finite metric space can be probabilistically approximated by mixture of tree metric (Bartal, 1996), the worst-case optimal submodular extension for metric energies can be obtained using F_{hier}. Note that F_{hier} reduces to F_{Potts} for Potts model. One can see this by considering the Potts model as a star-shaped tree with edge weights as 0.5.

5.3. Fast Conditional Gradient Computation for Dense Conditional Random Fields

Dense CRF Energy Function A dense CRF is specified by the following energy function

$$E(\mathbf{x}) = \sum_{a\in\mathcal{X}} \phi_a(x_a) + \sum_{a\in\mathcal{X}} \sum_{b\in\mathcal{X}, b\neq a} \phi_{ab}(x_a, x_b). \tag{36}$$

Note that every random variable is a neighbour of every other random variable in a dense CRF. Similar to previous work (Koltun and Krahenbuhl, 2011), we consider the pairwise potentials to be Gaussian, that is,

$$\phi_{ab}(i,j) = \mu(i,j) \sum_m w^{(m)} k(\mathbf{f}_a^{(m)}, \mathbf{f}_b^{(m)}), \tag{37}$$

$$k(\mathbf{f}_a^{(m)}, \mathbf{f}_b^{(m)}) = \exp\left(\frac{-||\mathbf{f}_a - \mathbf{f}_b||^2}{2}\right). \tag{38}$$

The term $\mu(i,j)$ is known as *label compatibility* function between labels i and j. Potts model and hierarchical Potts models are examples of $\mu(i,j)$. The other term is a mixture of Gaussian kernels $k(.,.)$ and is called the *pixel compatibility* function. The terms $\mathbf{f}_a^{(m)}$ are features that describe the random variable X_a. In practice, similar to Koltun and Krahenbuhl (2011), we use x, y coordinates and RGB values associated to a pixel as its features.

Algorithm 1 assumes that the conditional gradient $\mathbf{s}^*$ in step 3 can be computed efficiently. This is certainly not the case for dense CRFs, since computing s^* involves NL function evaluations of the submodular extension F, where N is the number of variables, and L is the number of labels. Each F evaluation has complexity $\mathcal{O}(N)$ using the efficient Gaussian filtering algorithm of Koltun and Krahenbuhl (2011). However, computation of s^* would still be $\mathcal{O}(N^2)$, which is clearly impractical for computer-vision applications where $N \sim 10^5 - 10^6$.

However, using the equivalence of relaxed LP objectives and the Lovasz extension of submodular extensions in Proposition 1, we are able to compute s^* in $\mathcal{O}(NL)$ time. Specifically, we use the algorithm of Ajanthan et al. (2017), which provides an efficient filtering procedure to compute a subgradient of the LP relaxation objective $E(\mathbf{y})$ of (P-LP).

Proposition 5. *Computing a subgradient of $E(\mathbf{y})$ in (P-LP) is equivalent to computing the conditional gradient for the submodular function F_{Potts}.*

Proof. For the Potts model, we derived the worst-case optimal extension F_{Potts} by making its Lovasz extension $f(\mathbf{y})$ same as the objective function $E(\mathbf{y})$ of the worst-case optimal LP relaxation. Hence, we have

$$\begin{aligned} E(\mathbf{y}) &= f(\mathbf{y}) \\ &= \max_{\mathbf{s}\in EP(F)} \langle \mathbf{y}, \mathbf{s} \rangle. \end{aligned}$$

The subgradient of $E(\mathbf{y})$ at $\mathbf{y_0}$ is an 'active' linear function at $\mathbf{y_0}$. Hence,

$$\partial E(\mathbf{y})|_{\mathbf{y}=\mathbf{y_0}} \in \operatorname*{argmax}_{\mathbf{s}\in EP(F)} \langle \mathbf{y_0}, \mathbf{s} \rangle \tag{39}$$

Equation (39) is equivalent to $\mathbf{s}^*$ computation in line 3 of algorithm 1, with $\mathbf{y_0} = -\nabla g(\mathbf{s}_k)$. □

A similar observation can be made in case of hierarchical Potts model. Hence we have the first practical algorithm to compute upper bound of log-partition function of a dense CRF for Potts and metric energies.

6. Accurate Submodular Extension for Higher-order Diversity Model

The pairwise Potts model often fails to capture useful image statistics because it restricts the order of the potentials to be at most two. Higher order clique potentials can model complex interactions of random variables, and thereby overcome this difficulty.

A higher-order model useful in real-world applications is the *diversity* model, which favours labeling where variables in a clique have fewer number of unique labels. For instance, in semantic segmentation we often first obtain superpixels from a clustering method (Comaniciu and Meer, 2002), which we consider as cliques in our model. We expect the labeling in a superpixel to be homogeneous. As a result, it is preferable to have pixels belonging to a superpixel to be incorrectly labeled with two class labels rather than three or more class labels. Let $\Gamma(\mathbf{x}_c)$ be the set of unique labels assigned to variables in the clique c, and ω_c be the weight associated with the clique. In our case, we take the clique potentials as proportional to the number of unique labels in the clique c:

$$\phi_c(\mathbf{x}_c) = \omega_c |\Gamma(\mathbf{x}_c)| \tag{40}$$

where the notation $|A|$ denotes the cardinality of a set A.

LP Relaxation First, let us consider the IP formulation of the MAP problem for our diversity model. Using the same set of indicator variables $y_{ai} = \{0, 1\}$ as for the Potts model, which are binary now, the clique potential $\phi(\mathbf{x}_c)$ can be represented as

$$\phi_c(\mathbf{x}_c) = \omega_c \cdot \sum_{i=1}^{L} \max_{(a,b)\in c} |y_{ai} - y_{bi}| \tag{41}$$

For example, let the set of unique labels in a clique be $\Gamma = \{l_1, l_2, l_3\}$. This implies that $y_{ai} = 1$ for some, but not all, variables in the clique for $i = \{1, 2, 3\}$. Hence, $|y_{ai} - y_{bi}| = 1$ for some pairs $(a, b) \in c$ for $i = \{1, 2, 3\}$. Also, for any other label i, $|y_{ai} - y_{bi}| = 0$ for all $(a, b) \in c$ since $y_{ai} = 0$ for all variables a for these labels i. For binary variables, the maximum possible value of $|y_{ai} - y_{bi}|$ is 1. Hence, $\phi_c(\mathbf{x}_c) = \omega_c \cdot (1 + 1 + 1) = 3 \cdot \omega_c = \omega_c |\{l_1, l_2, l_3\}|$.

We now relax the indicator variables y_{ai} to lie in $[0, 1]$. Assuming the pairwise potentials in equation (6) to be Potts, an accurate LP relaxation for the higher-order diversity model is given by:

$$\begin{aligned} \text{(HOD-LP)} \quad \min_{\mathbf{y}} \quad & E(\mathbf{y}) = \sum_{a=1}^{N}\sum_{i=1}^{L} \phi_a(i) y_{ai} \\ & + \sum_{(a,b)\in\mathcal{N}} \sum_{i} \frac{w_{ab}}{2} \cdot |y_{ai} - y_{bi}| \\ & + \sum_{c\in\mathcal{C}} w_c \cdot \sum_{i=1}^{L} \max_{(a,b)\in c} |y_{ai} - y_{bi}| \\ \text{s.t} \quad & \mathbf{y} \in \Delta. \end{aligned} \tag{42}$$

where the label (HOD-LP) stands for LP relaxation for higher-order diversity model. The objective function is similar to that for the Potts model along with the additional terms corresponding to higher-order potentials. The constraints for (HOD-LP) are the same as for (P-LP). The above LP has not been formally analysed in literature, and we do not make any optimality claims. However, its similarity in form to (P-LP) is an indication of its accuracy. We establish its accuracy empirically in section 7.

LP-based Submodular Extension We use the same 1-of-L encoding scheme as for Potts model. Problem 9 then factorises as:

$$\min_{\mathbf{s}\in EP(F)} \quad \sum_{a=1}^{N} \log \sum_{i=1}^{L} \exp(-s_{ai}). \tag{43}$$

We introduce a temperature parameter T to problem (43) to obtain the following new problem:

$$\min_{\mathbf{s}\in EP(F)} \quad g_T(\mathbf{s}) = \sum_{a=1}^{N} T \cdot \log \sum_{i=1}^{L} \exp(-\frac{s_{ai}}{T}). \tag{44}$$

In the limit $T \to 0$, problem (44) becomes

$$-\min_{\mathbf{y}\in\Delta} f(\mathbf{y}) \tag{45}$$

where $f(.)$ is the Lovasz extension of $F(.)$. We are interested in making this LP the same as (HOD-LP) of equation (42), thereby enabling us to obtain the submodular extension for the higher-order diversity model. We will make use of the following lemma in our proof.

Lemma 1. *The Lovasz extension of the set function* $F_{HOD}(A) =$

$$F_{Potts}(A) + \sum_{c\in\mathcal{C}} w_c \cdot \sum_{i=1}^{L} \max_{(a,b)\in c} [|A \cap \{v_{ai}, v_{bi}\}| = 1]$$

is the objective function $E(\mathbf{y})$ of the LP relaxation referred to as (HOD-LP).

Proof. Making use of Proposition 2, it suffices for us to show that the Lovasz extension of the i-th Ising model $F_i(A) = \max_{(a,b)\in c}[|A \cap \{v_{ai}, v_{bi}\}| = 1]$ is $\max_{(a,b)\in c} |y_{ai} - y_{bi}|$.

Let the clique c have M variables. Given $\mathbf{y} \in [0,1]^M$, let us order its components in decreasing order $y_{j_1} > \cdots > y_{j_M}$, where $(j_1, \ldots, j_M)$ is a permutation. Then the Lovasz extension of $F_i()$ is:

$$f_i(\mathbf{y}) = \sum_{k=1}^{M} y_{j_k}[F_i(\{j_1, \ldots, j_k\}) - F_i(\{j_1, \ldots, j_{k-1}\})]$$

We note that when $A = \{\}$ or $\{j_1, \ldots, j_M\}$, that is, when A represents homogeneous labelings, $F_i(A) = 0$. For any other $A \subset \{j_1, \ldots, j_M\}$, $F_i(A) = 1$. Hence, the Lovasz extension simplifies to

$$\begin{aligned} f_i(\mathbf{y}) &= y_{j_1} - y_{j_M} \\ &= \max_{(a,b)\in c} |y_{ai} - y_{bi}| \end{aligned}$$

This proves our lemma. □

We are now in a position to state our main result:

Proposition 6. *The set function F_{HOD} of lemma 1 is the submodular extension of our higher-order diversity model whose Lovasz extension is $E(\mathbf{y})$ in problem (HOD-LP).*

Proof. Note that $\forall \mathbf{y} \in \mathcal{L}^N$, that is, for all valid labelings, $F_{HOD}(A_\mathbf{y})$ equals $E(\mathbf{y})$. Hence, F_{HOD} is an extension.

Lemma 1 showed that the Lovasz extension of F_{HOD} is $E(\mathbf{y})$ in problem (HOD-LP). $E(\mathbf{y})$ being a sum of terms corresponding to maximum of linear functions, is convex. The convexity of Lovasz extension implies submodularity of the set function Bach (2013). Hence, F_{HOD} is submodular. □

We now show the effectiveness of the accurate submodular extensions for different classes of models by means of experiments on synthetic and real-world datasets.

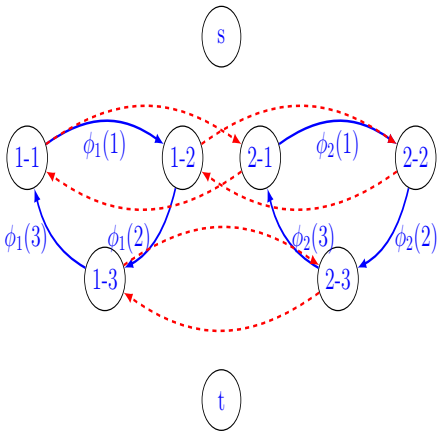


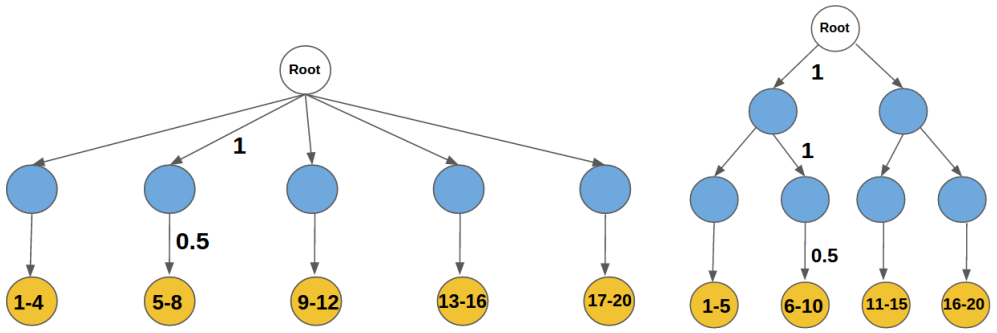
Fig. 5: **Alternate extension for synthetic experiments***: An st-graph specifying the alternate submodular extension for Potts model for 2 variables with 3 labels each and connected to each other. The convention used is same as in figure 3. Each dotted red arc has weight $w_{12}/2$. This alternate extension was also used to derive the extension for hierarchical Potts model.*

7. Experiments

Using synthetic data, we show that our upper-bound compares favorably with TRW for both Potts and hierarchical Potts models. For comparison, we restrict ourselves to sparse CRFs, as the code available for TRW does not scale well to dense CRFs. We also perform stereo matching using dense CRF models and compare our results with the mean-field-based approach of Koltun and Krahenbuhl (2011). All experiments were run on a x86-64, 3.8GHz machine with 16GB RAM. In this section, we refer to our algorithm as *Submod* and mean field as *MF*.

7.1. Upper-bound Comparison using Synthetic Data

Data We generate lattices of size 100 × 100, where each lattice point represents a variable taking one of 20 labels. The pairwise relations of the sparse CRFs are defined by 4-connected neighbourhoods. The unary potentials are uniformly sampled in the range [0, 10]. We consider (a) Potts model and (b) hierarchical Potts models with pairwise distance between labels given by the trees of Figure 6. The pairwise weights are varied in the range $\{1, 2, 5, 10\}$. We compare the results of our worst-case optimal submodular



(a) Tree A (b) Tree B

Fig. 6: **Trees for synthetic experiments:** *The hierarchical Potts models defining pairwise distance among 20 labels used for upper-bound comparison with TRW. Blue nodes are the meta-labels and yellow nodes are labels. All the edges at a particular level have the same edge weights. The sequence of weights from root level to leaf level is 1, 0.5 for tree A and 1, 1, 0.5 for tree B. The yellow node is shown to clump together 4 and 5 leaf nodes for tree A and B respectively.*

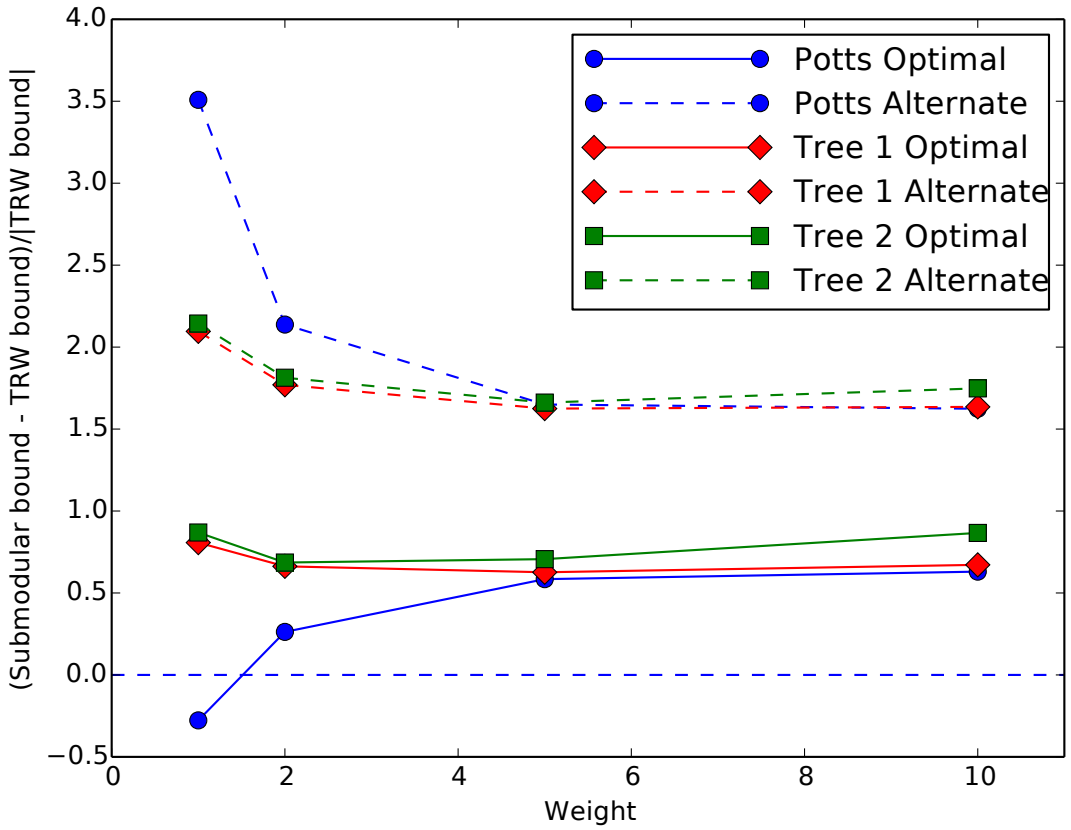


Fig. 7: **Upper-bound comparison using synthetic data:** *The plot shows the ratio (Submod bound - TRW bound)/|TRW bound| averaged over 100 unary instances as a function of pairwise weights using the worst-case optimal and alternate extension for Potts and hierarchical Potts models. We observe that the worst-case optimal extension (solid) results in tighter bounds as compared to the respective alternate extensions (dotted). Also, the worst-case optimal extension bounds are in similar range as the TRW bounds.*

extension with an alternate submodular extension as given in Figure 5.

Method For our algorithm, we use the standard schedule $\gamma = 2/(k+2)$ to obtain step size γ at iteration k. We run our algorithm till convergence - 100 iterations suffices for this. The experiments are repeated for 100 randomly generated unaries for each model and each weight. For TRW, we used the MATLAB toolbox of Domke (2013). The baseline code does not optimise over tree distributions. We varied the edge-appearance probability in trees over the range [0.1 - 0.5] and found 0.5 to give tightest upper bound.

Results We plot the ratio of the normalised difference of the upper bound values of our method with TRW as a function of pairwise weights. The ratios are averaged over 100 instances of unaries. Figure 7 shows the plots for Potts and hierarchical Potts models for the worst-case optimal and alternate extension. We find that the optimal extension (solid) results in tighter upper-bounds than the alternate extension (dotted) for both models. This is because the representation of the submodular function using Figure 5 requires that $\phi_a(i)$ be non-negative. This implies that $F(A)$ values are larger for the worst-case optimal extension of Figure 3 as compared to the alternate extension. Hence the minimisation problem 9 has larger domain $EP(F)$ for the optimal extension, thereby resulting in better minima. Figure 7 indicates that our algorithm does not provide as tight upper-bounds as TRW, however they are of similar magnitude. TRW makes use of the standard LP relaxation (Chekuri et al., 2004), from the second-level of Sherali-Adams hierarchy (having $\mathcal{O}(N^2)$ number of relaxed variables), and is tighter than Kleinberg-Tardos relaxation, resulting in better approximation. However, TRW does not scale well with neighborhood size, thereby prohibiting its use in dense CRFs.

7.2. Stereo Matching using Dense CRFs

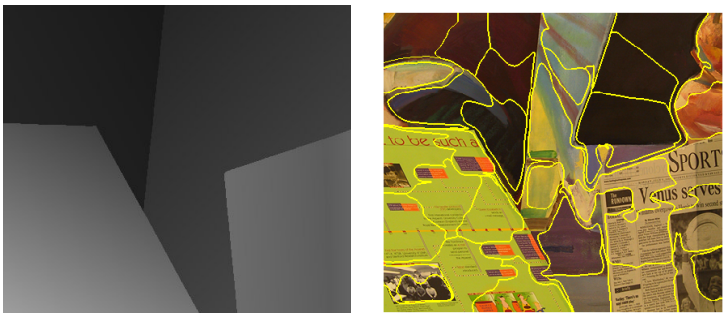
Data We demonstrate the benefit our algorithm for stereo matching on images extracted from the Middlebury stereo matching dataset (Scharstein et al., 2001). We use dense CRF models with Potts compatibility term and Gaussian pairwise potentials. The unary terms are obtained using the absolute difference matching function of Scharstein et al. (2001).

Method We use the implementation of mean-field algorithm for dense CRFs of Koltun and Krahenbuhl (2011) as our baseline. For our algorithm, we make use of the modified Gaussian filtering implementation for dense CRFs by Ajanthan et al. (2017) to compute the conditional gradient at each step. The step size γ at each iteration is selected by doing line search in $[0, 1]$ (we try step sizes at 0.1 interval and pick the one that decreases the objective most). We run our algorithm till 100 iterations, since the visual quality of the solution does not show much improvement beyond this point. We run mean-field up to convergence, with a threshold of 0.001 for change in KL-divergence.

Results Figure 9 shows some example solutions obtained by picking the label with maximum marginal probability for each variable for mean-field and for our algorithm. We also report the time and energy values of the solution for both methods. Though we are not performing MAP estimation, energy values give us a quantitative indication of the quality of solutions. For the full set of 21 image pairs (2006 dataset), the average ratio of the energies of the solutions from our method compared to mean-field is 0.943. The average time ratio is 10.66. We observe that our algorithm results in more natural looking stereo matching results with lower energy values for all images. However, mean-field runs faster than our method for each instance.The set of hyperparameters that we used can be found in the appendix.

7.3. Stereo Matching using Higher-order Diversity Model

Data Next, we use the higher-order diversity model for stereo matching on the Middlebury dataset. A higher-order model is suitable to be used for only some images in the dataset, since others have gradually sloping surfaces.



(a) Ground truth disparity (b) Superpixels

Fig. 8: *An example for stereo matching that is unsuitable to be used with higher-order model. Figure (a) shows the ground truth disparity for this example. The image has gradually sloping surfaces, and hence gradually varying disparity values. Figure (b) shows the superpixels for this example. Using these superpixels as higher-order cliques will force pixels on gradually sloping surfaces to take similar disparity values.*

Higher-order cliques will force all pixels on gradually sloping surfaces to take similar disparity values and we want to avoid this (Figure 8). We take a dense CRF model with Potts compatibility term and Gaussian pairwise potentials, and augment it with higher-order diversity term. We obtain higher-order cliques by using super-pixels obtained from the mean-shift algorithm (Comaniciu and Meer, 2002).

Method We again use the modified Gaussian filtering implementation for dense CRFs by Ajanthan et al. (2017) to compute the conditional gradient at each step. To compute the contribution of the higher-order model, we make use of an efficient strategy to reuse computation. Recall that computing the conditional gradient coordinates $s^*_{\sigma(i)}$ requires us to obtain differences of the nested sets $F(S_i) - F(S_{i-1})$. We store the clique state at each i, and this enables us to compute the difference for each coordinate in a constant amount of time. In this case, computation of conditional gradient takes $\mathcal{O}(cNL)$ where c is the number of higher-order cliques. The step size γ at each iteration is selected by doing line search in $[0, 1]$ (we try step sizes at 0.1 interval and pick the one that decreases the objective most).We ran our algorithm for 200 iterations.

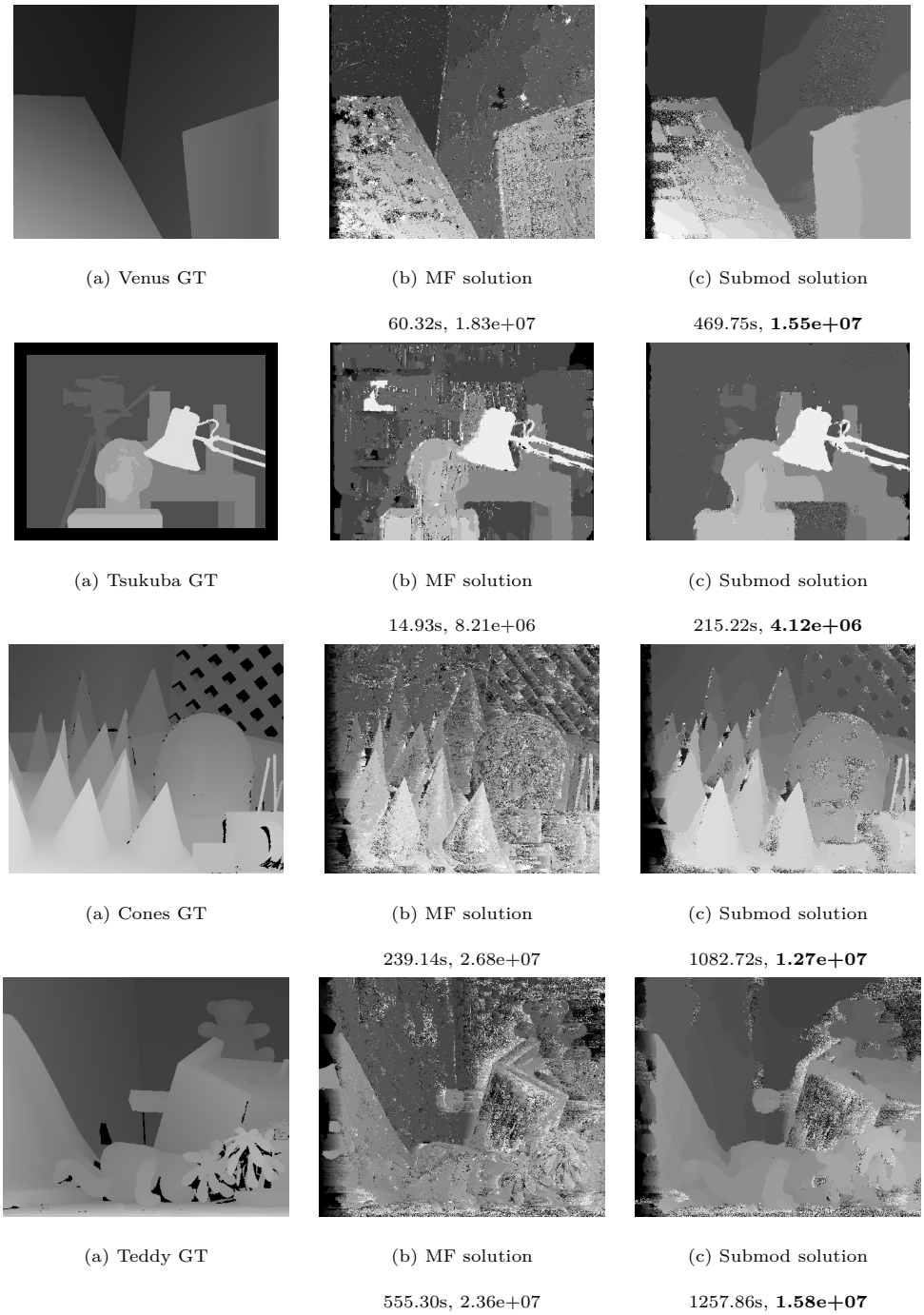
Results Figure 10 shows some example solutions obtained by picking the label with maximum marginal probability for each variable. Our method gives results that are reasonably close to the ground-truth. The set of hyperparameters that we used can be found in the appendix.

7.4. Semantic Segmentation using Higher-order Diversity Model

Data We evaluate our approach on the task of semantic segmentation on the MSRC-21 dataset (Shotton et al., 2009). It consists of 591 color images of size 320×213 with corresponding ground truth labelings of 21 object classes. We made use of the unary features from the *TextonBoost* classifier (Shotton et al., 2009). As for stereo matching (subsection 7.3), we augment the dense CRF with Gaussian pairwise potentials with our higher-order diversity model. Superpixels for higher-order modeling were obtained from the mean-shift algorithm (Comaniciu and Meer, 2002).

Method As for stereo matching, we used the modified Gaussian filtering implementation (Ajanthan et al., 2017) for the contribution of the dense pairwise terms to the conditional gradient. We used the same strategy as in subsection 7.3 to compute the higher-order component of the conditional gradient in $\mathcal{O}(cNL)$ time-complexity. We ran our algorithm for 100 iterations.

Results The ground-truth labelings provided with the MSRC-21 dataset are coarse. We evaluate our algorithm on the set of 94 images for which fine-grain annotations are available (Koltun and Krahenbuhl, 2011). On this subset, our method correctly labeled 81.18% of pixels. It took 275.02s on average per instance to run our method for 100 iterations. Some representative results are shown



(a) Venus GT (b) MF solution 60.32s, 1.83e+07 (c) Submod solution 469.75s, **1.55e+07**

(a) Tsukuba GT (b) MF solution 14.93s, 8.21e+06 (c) Submod solution 215.22s, **4.12e+06**

(a) Cones GT (b) MF solution 239.14s, 2.68e+07 (c) Submod solution 1082.72s, **1.27e+07**

(a) Teddy GT (b) MF solution 555.30s, 2.36e+07 (c) Submod solution 1257.86s, **1.58e+07**

Fig. 9: *Stereo matching using dense CRFs with Potts compatibility and Gaussian pairwise potentials. We compare our solution with the mean-field algorithm of Koltun and Krahenbuhl (2011). We observe that our method gives better-looking solutions with lower energy value at the cost of higher computational time.*

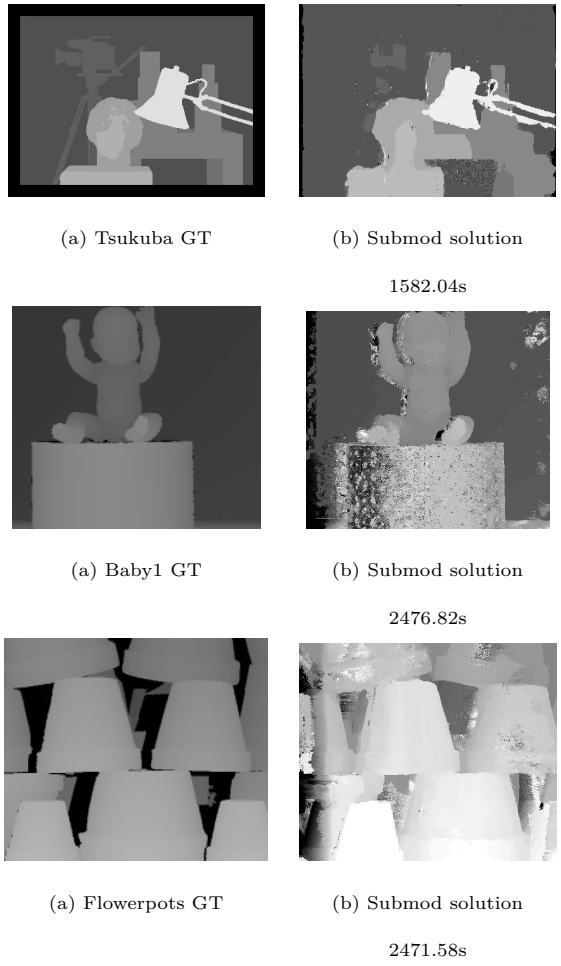
in Figure 11 where we label each pixel with the label having the maximum marginal probability. Our set of hyperparameters can be found in the appendix.

8. Discussion

We have established the relation between submodular extension and the LP relaxation for MAP estimation using Lovasz extension for various CRF models. This allowed us to identify the worst-case optimal submodular extension for Potts as well as the general metric labeling problems. In addition, we obtained an accurate submodular extension for a higher-order model based on label-diversity in cliques. It is worth noting that it might still be possible to obtain an improved submodular extension for a given problem instance. The design of a computationally feasible algorithm for this task is an interesting direction of future research. While our work focused on one class of higher-order model, there is potential for our approach to be used to identify accurate submodular extensions for other energy families, such as truncated max-of-convex models (Pansari and Kumar, 2017).

Acknowledgement

This work was supported by the EPSRC grantsEP/P020658/1, TU/B/000048, and EP/P022529/1 and Google Deepmind PhD studentship.

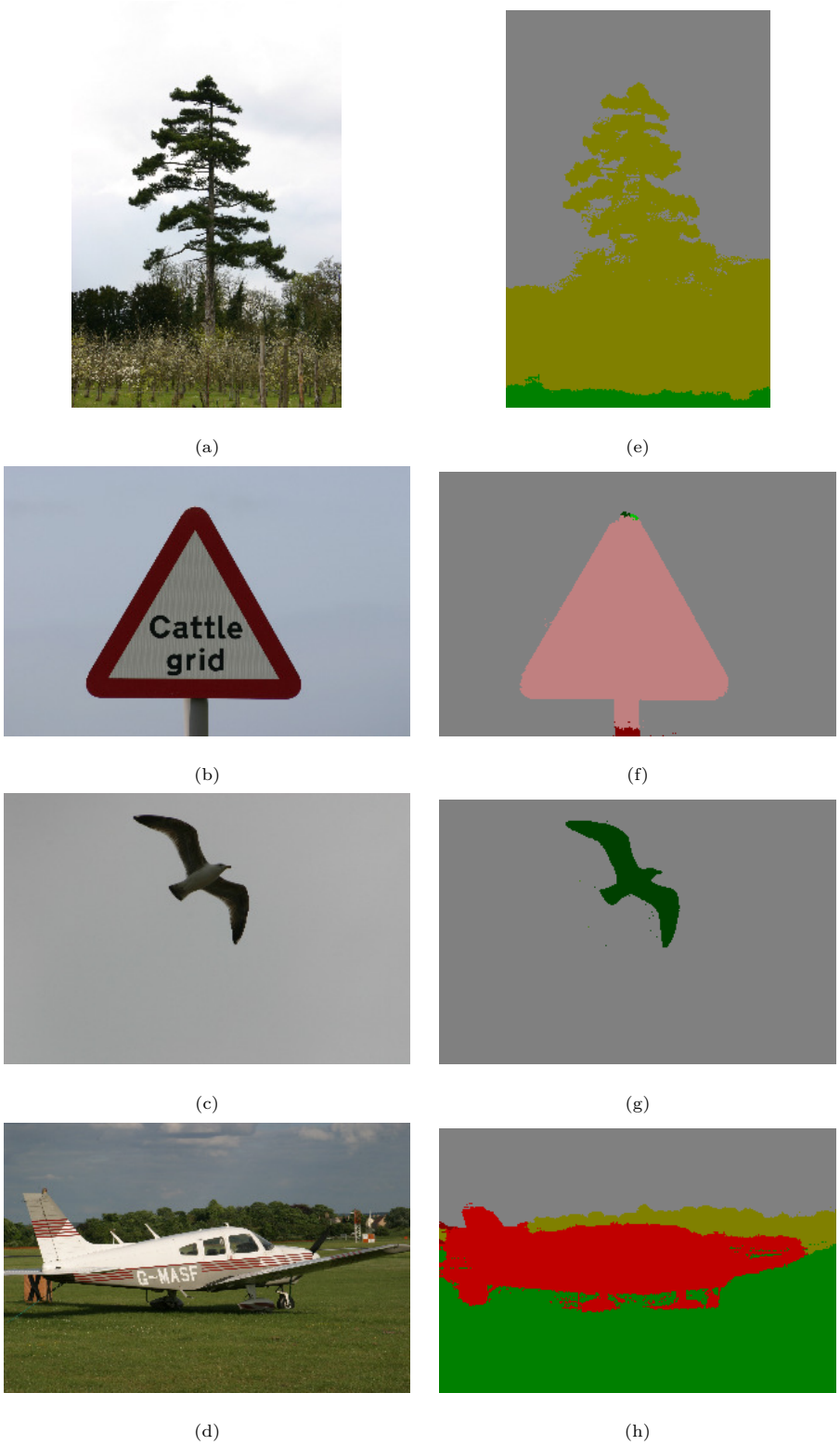


(a) Tsukuba GT (b) Submod solution 1582.04s

(a) Baby1 GT (b) Submod solution 2476.82s

(a) Flowerpots GT (b) Submod solution 2471.58s

Fig. 10: *Stereo matching using dense CRFs with higher-order diversity term in addition to Potts compatibility and Gaussian pairwise potentials.*



(a) (e) (b) (f) (c) (g) (d) (h)

Fig. 11: *Sample results from semantic segmentation on the MSRC-21 dataset using dense CRFs with higher-order diversity term in addition to Potts compatibility and Gaussian pairwise potentials. (a)-(d) show the input image, and (e)-(f) show the corresponding results using our method. We observe that our method is able to segment out objects with complex boundaries fairly accurately.*

References

Ajanthan, T., Desmaison, A., Bunel, R., Salzmann, M., Torr, P., Kumar, M., 2017. Efficient linear programming for dense crfs, in: CVPR.

Bach, F., 2013. Learning with submodular functions: A convex optimization perspective. Foundations and Trends in Machine Learning .

Bartal, Y., 1996. Probabilistic approximation of metric spaces and its algorithmic applications, in: Foundations of Computer Science.

Bartal, Y., 1998. On approximating arbitrary metrices by tree metrics, in: ACM Symposium on Theory of Computing.

Boyd, S., Vandenberghe, L., 2004. Convex optimization. Cambridge university press.

Boykov, Y., Veksler, O., Zabih, R., 2001. Fast approximate energy minimization via graph cuts. PAMI .

Chekuri, C., Khanna, S., Naor, J., Zosin, L., 2004. A linear programming formulation and approximation algorithms for the metric labeling problem. SIAM Journal on Discrete Mathematics .

Comaniciu, D., Meer, P., 2002. Mean shift: A robust approach toward feature space analysis. PAMI .

Djolonga, J., Krause, A., 2014. From map to marginals: Variational inference in bayesian submodular models, in: NIPS.

Domke, J., 2013. Learning graphical model parameters with approximate marginal inference. PAMI .

Edmonds, J., 1970. Submodular functions, matroids, and certain polyhedra. Combinatorial Optimization — Eureka, You Shrink! .

Frank, M., Wolfe, P., 1956. An algorithm for quadratic programming. Naval research logistics quarterly .

Ishikawa, H., 2003. Exact optimization for markov random fields with convex priors. IEEE Trans. Pattern Anal. Mach. Intell. 25, 1333–1336.

Kleinberg, J., Tardos, E., 2002. Approximation algorithms for classification problems with pairwise relationships: Metric labeling and markov random fields. IEEE Symposium on the Foundations of Computer Science .

Kohli, P., Kumar, M.P., Torr, P.H., 2007. P3 & beyond: Solving energies with higher order cliques, in: CVPR.

Koltun, V., Krahenbuhl, P., 2011. Efficient inference in fully connected crfs with gaussian edge potentials. NIPS .

Kumar, M., Veksler, O., Torr, P., 2011. Improved moves for truncated convex models. JMLR .

MacKay, D.J.C., 2003. Information theory, inference, and learning algorithms. Cambridge University Press.

Manokaran, R., Naor, J., Raghavendra, P., Schwartz, R., 2008. Sdp gaps and ugc hardness for multiway cut, 0-extension, and metric labeling, in: ACM Symposium on Theory of Computing.

Pansari, P., Kumar, M.P., 2017. Truncated max-of-convex models., in: CVPR.

Scharstein, D., Szeliski, R., Zabih, R., 2001. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, in: Stereo and Multi-Baseline Vision.

Sherali, H.D., Adams, W.P., 1990. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. SIAM J. Discrete Math. 3, 411–430.

Shotton, J., Winn, J., Rother, C., Criminisi, A., 2009. Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. International Journal of Computer Vision .

Vineet, V., Warrell, J., Torr, P., 2014. Filter-based mean-field inference for random fields with higher-order terms and product label-spaces. IJCV .

Wainwright, M., Jaakkola, T., Willsky, A., 2005. A new class of upper bounds on the log partition function. IEEE Transactions on Information Theory .

Wang, C., Komodakis, N., Paragios, N., 2013. Markov random field modeling, inference & learning in computer vision & image understanding: A survey. Computer Vision and Image Understanding 117, 1610–1627.

Zhang, J., Djolonga, J., Krause, A., 2015. Higher-order inference for multi-class log-supermodular models, in: ICCV.