Supplementary Material for Graph Cut based Inference with Co-occurrence Statistics

Ľubor Ladický[1,3], Chris Russell[1,3], Pushmeet Kohli[2], and Philip H.S. Torr[1]

[1] Oxford Brookes
[2] Microsoft Research Cambridge

1 The Integer Programming formulation, and its Linear Relaxation

In this section we show, that the minimisation of the extended energy function

$$E(\mathbf{x}) = \sum_{c \in \mathcal{C}} \psi_c(\mathbf{x}_c) + C(L(\mathbf{x})) \tag{1}$$

can be formulated as an Integer Program (IP). First the objective function is linearised using a vector $\mathbf{z}$ of binary indicator variables to represent the assignment of labels. $\mathbf{z}$ is composed of $z_{i;a} \forall a \in \mathcal{L}, \forall i \in \mathcal{V}$, and, $z_{ij;ab} \forall a, b \in \mathcal{L}, (i,j) \in \mathcal{E}$ where $\mathcal{E}$ is the set of edges, to represent the state of variables x_i, x_j such that,

$$z_{i;a} = \begin{cases} 1 & \text{if } x_i = a \\ 0 & \text{otherwise} \end{cases}, \; z_{ij;ab} = \begin{cases} 1 & \text{if } x_i = a \text{ and } x_j = b \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

In addition $\mathbf{z}$ is composed of z_L, there are indicator variables that show which subset of labels $L(\mathbf{x})$ is used for the assignment. There are $2^{|\mathcal{L}|}$ such variables in total, one variable z_L for every $L \subseteq \mathcal{L}$. We write

$$z_L = \begin{cases} 1 & \text{if } L = L(\mathbf{x}) \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

This $\mathbf{z}$ is a binary vector of length $|\mathcal{V}| \cdot |\mathcal{L}| + |\mathcal{E}| \cdot |\mathcal{L}|^2 + 2^{|\mathcal{L}|}$.

The resulting IP can be written as

$$\min_{\mathbf{z}} \sum_{i \in \mathcal{V}, a \in \mathcal{L}} \psi_i(a) z_{i;a} + \sum_{\substack{(i,j) \in \mathcal{E}, \\ a,b \in \mathcal{L}}} \psi_{i,j}(a,b) z_{ij;ab} + \sum_{L \subseteq \mathcal{L}} C(L) z_L \tag{4}$$

[3] The authors assert equal contribution and joint first authorship

This work was supported by EPSRC, HMGCC and the PASCAL2 Network of Excellence. Professor Torr is in receipt of a Royal Society Wolfson Research Merit Award.

such that

$$\begin{aligned}
&\textstyle\sum_a z_{ij;ab} = z_{j;b}, && \forall (i,j) \in \mathcal{E}, b \in \mathcal{L}, && (5)\\
&\textstyle\sum_b z_{ij;ab} = z_{i;a}, && \forall (i,j) \in \mathcal{E}, a \in \mathcal{L}, && (6)\\
&\textstyle\sum_a z_{i;a} = 1, && \forall i \in \mathcal{V}, && (7)\\
&\textstyle\sum_{i \in \mathcal{V}} z_{i;a} \geq z_L, && \forall a \in L \subseteq \mathcal{L}, && (8)\\
&\textstyle\sum_{L \subseteq \mathcal{L}} z_L = 1 && && (9)\\
&\textstyle\sum_{L \ni a} z_L \geq z_{i;a}, && \forall i \in \mathcal{V}, a \in \mathcal{L} && (10)\\
&z_{i;a}, z_{ij;ab}, z_L \in \{0,1\} && \forall i \in \mathcal{V}, \forall (i,j) \in \mathcal{E}, &&\\
&\forall a, b \in \mathcal{L}, && \forall L \subseteq \mathcal{L}. && (11)
\end{aligned}$$

The marginal consistency and uniqueness constraints (5 - 7) are well-known and used in the standard IP formulation of the labelling problem [3,4,7,8]. However, the constraints (8,9, 10) are specific to our co-occurrence formulation, and enforce that $z_L = 1$, if and only if, $L(\mathbf{x}) = L$.

Constraint (9) ensures that only one of the indicator variables corresponding to a subset of the label set $\mathcal{L}$ is 1. Constraint (8) ensures that if $z_L = 1$, then each label $a \in L$ should be taken by at least one variable $i \in \mathcal{V}$, *i.e.* $L \subseteq L(\mathbf{x})$. The constraint (10) ensures that if $z_L = 1$, then no variable is assigned a label not present in $L(\mathbf{x})$ ie $L(\mathbf{x}) \subseteq L$. The last constraint (11) ensures that all indicator variables are binary.

The IP can be converted to a linear program (LP) by relaxing the integral constraints (11) to

$$\begin{aligned}
&z_{i;a}, z_{ij;ab}, z_L \in [0,1] && \forall i \in \mathcal{V}, \forall (i,j) \in \mathcal{E}, &&\\
&\forall a, b \in \mathcal{L}, && \forall L \subseteq \mathcal{L}. && (12)
\end{aligned}$$

The resulting linear program can be solved using any general purpose LP solver. That said, the size of the LP is a concern as a typical computer vision problem contains a vast number of implicit variables and constraints, in the supplementary materials we describe a reduction of the model complexity which allows these potentials to be effectively solved by LP solvers, and standard energy minimisation algorithms such as TRW-S.

While this approach allows co-occurrence to be computed effectively for small images, over large images the memory and time requirements of standard LP solvers make this approach infeasible. We next show that, under a particular choice of relaxation, the higher order energy $C(L(\mathbf{x}))$ can be transformed into a pairwise energy function with the addition of a single auxiliary variable L that takes $2^{|\mathcal{L}|}$ states. This approach allows us to use the wide body of standard inference techniques [1,2,5] to minimise this function.

2 Pairwise Representation of Co-occurrence Potentials

To encode the above IP as a pairwise model, we write the sub-cost $\sum_{c \in \mathcal{C}} \psi_c(\mathbf{x}_c)$ represented by equations (5-7), as a standard MRF, with unary and pairwise potentials defined

over a graph of size $|\mathcal{V}|$. We represent the state of all z_L, as one multi-state random variable Z taking $2^{|\mathcal{L}|}$ states. Such that each state of Z, which is an element of the power set of $\mathcal{L}$, thus any state of L defines a corresponding label set L. We form the Lagrangian constraint that $L = L(\mathbf{x})$ (which follows from a combination of (3), and constraints (8,10)). In place of (1) we write

$$E(\mathbf{x}, L) = \sum_{c \in \mathcal{C}} \psi_c(\mathbf{x}_c) + C(L) + \lambda T_1(\mathbf{x}, Z) + \lambda T_2(\mathbf{x}, Z), \tag{13}$$

where, as $\lambda \to \infty$, the function $T_1(\cdot)$ enforces that $L(\mathbf{x}) \subseteq L$, while $T_2(\cdot)$ enforces that $L \subseteq L(\mathbf{x})$ providing the following inequalities hold

$$T_1(\mathbf{x}, Z) = \begin{cases} 0 & \text{if } L(\mathbf{x}) \subseteq L, \\ k_{\mathbf{x},Z} > 0 & \text{otherwise} \end{cases}$$

$$T_2(\mathbf{x}, Z) = \begin{cases} 0 & \text{if } l \subseteq L(\mathbf{x}) \\ k'_{\mathbf{x},Z} > 0 & \text{otherwise.} \end{cases} \tag{14}$$

$T_1(\cdot)$ can be embedded as a sum of pairwise expressions of the form

$$\psi_{Z,i}(l, x_i) = \begin{cases} 0 & x_i \in L \\ 1 & x_i \notin L \end{cases} \quad \forall i \in \mathcal{V} \tag{15}$$

by choosing $T_1(\cdot)$ as

$$T_1(Z, \mathbf{x}) = \sum_{i \in \mathcal{V}} \psi_{Z,i}(L, x_i). \tag{16}$$

Similarly, we can encode the cost $T_2(\cdot)$ by the addition of $|\mathcal{L}|$ random variables, each taking $|\mathcal{V}| + 1$ states. Denoting each new variable as $\{Y_l : \forall l \in \mathcal{L}\}$ and the set of states taken by each of these variables as $\mathcal{V} \cup \{\bar{\mathcal{V}}\}$ we associate the set of pairwise costs

$$\psi_{Y_l,i}(y_l, x_i) = \begin{cases} 1 & \text{if } y_l = i \text{ and } x_i \neq l \\ 0 & \text{otherwise,} \end{cases} \tag{17}$$

and

$$\psi_{Y_l,H}(y_l, h) = \begin{cases} 1 & \text{if } y_l = \bar{\mathcal{V}} \text{ and } l \in h \\ 0 & \text{otherwise.} \end{cases} \tag{18}$$

It can be readily be seen that

$$\min_{\mathbf{y}} \left(\sum_{l \in L} \psi_{Y_l,H}(y_l, h) + \psi_{y_l,i}(y_l, x_i) \right) \tag{19}$$

satisfies the second set of constraints 14, and that $T_2(\cdot)$ is expressible as the minimum of a pairwise energy.

Owing to the large number of states required, inference over the construction of $T_2(\cdot)$ carries too high a computational cost (violating the efficiency requirement *(iii)*),

and we instead choose to remove it. This is equivalent to a relaxation which removes constraint (8), that $L \subseteq L(\mathbf{x})$.

However, a minimal cost labelling of the relaxed energy

$$E^r(\mathbf{x}, L) = \sum_{c \in \mathcal{C}} \psi_c(\mathbf{x}_c) + C(L) + \lambda T_1(\mathbf{x}, Z), \tag{20}$$

will satisfy constraint (10), that $L(\mathbf{x}) \subseteq L$. This relaxation allows labels in L not to occur in the labelling $\mathbf{x}$.

Theorem 1. *Minimisation of $E^r(\mathbf{x}, Z)$ is equivalent to the minimisation of the original energy $E(\mathbf{x})$ if and only if the cost function $C(L)$ is* monotone increasing *with respect to L.*

A relaxation, or removal of a constraint, from an integer or linear program is said to be *tight*, if the cost of the minimal labelling does not change with the removal of the constraint.

Assume $C(L(\mathbf{x}))$ is monotone increasing, then given any fixed $\mathbf{x}$,

$$\lim_{\lambda \to \infty} \min_{L \subseteq \mathcal{L}} (C(L) + \lambda T_1(\mathbf{x}, Z)) = \min_{L(x) \subseteq L \subseteq \mathcal{L}} C(L) = C(L(\mathbf{x})). \tag{21}$$

For the only if case, we assume that $C(\cdot)$ is not monotone increasing, *i.e.* there exists some $L, L' \in \mathcal{L}$ such that

$$C(L) < C(L \cup L'). \tag{22}$$

Then, picking any $\mathbf{x}$ such that $L(\mathbf{x}) = L$ we find

$$\lim_{\lambda \to \infty} \min_{S \subseteq \mathcal{L}} (C(L) + \lambda T_1(\mathbf{x}, Z)) = \min_{L(\mathbf{x})) \subseteq L \subseteq \mathcal{L}} C(L) \neq C(L(\mathbf{x})) \square \tag{23}$$

Note that by (23) if $C(\cdot)$ is not monotone increasing, the solution to the new relaxation is equivalent to the solving the original IP formulation where the cost $C(\cdot)$ is replaced with a new expression

$$C'(L(\mathbf{x})) = \min_{L(x) \subseteq L \subseteq \mathcal{L}} C(L(\mathbf{x})), \tag{24}$$

which is monotone increasing.

3 Efficient Representation of Cost Functions

In this section, we describe an LP formulation which requires relatively very few auxiliary variables making it feasible for some small graphs to be solved exactly. This approach leads to a more compact representation of the costs $C(L(\mathbf{x}))$, necessary for the efficient application of LP solvers, and was used in the experimental comparison with α expansion. It can also be applied to the to the pairwise formulation of section 4, enabling a more efficient use of reparameterisation based methods such as TRW-S or BP.

Explicitly assigning a cost to each and every possible set of labels is both computationally difficult and prone to over-fitting. This is particularly important when we are dealing with a large label space. To overcome these difficulties, we can restrict the form of the cost function $C(L(\mathbf{x}))$. One simple choice is to represent $C(L(\mathbf{x}))$ as the sum of costs based on the occurrence of some smaller set of labels. For instance, consider the representation

$$C(L(\mathbf{x})) = \sum_{l \in \mathcal{L}} w_l \Delta(l \in L(\mathbf{x})) \tag{25}$$

where $\Delta(l \in L(\mathbf{x}))$ takes the value 1 if label l is present in the set $L(\mathbf{x})$, and 0 otherwise. This formulation assigns a cost w_l if the label l is present in the image and corresponds to a scene based label cost, similar to the work of Torralba[6] discussed in section 2.2. Such a representation can be formulate as an IP as we show below.

To formulate the problem efficiently, we consider a family of functions $\mathcal{F}$ such that each $f_i \in \mathcal{F}$ maps to and from the power-set of $\mathcal{L}$. More formally

$$\mathcal{F} = \{f : \mathcal{P}(\mathcal{L}) \rightarrow \mathcal{P}(\mathcal{L})\} \tag{26}$$

We associate each $f \in \mathcal{F}$ with a unique cost function $C_f : \mathcal{F} \rightarrow \Re_0^+$, and subject all f to the additional requirement that if $L \subseteq L' \subseteq \mathcal{L}$ then

$$f(L) \subseteq f(L') \tag{27}$$

These functions allow cost functions to be described compactly, for example: to assign the same cost k to $L(\mathbf{x})$ for all $L(\mathbf{x}) \subseteq S \subseteq \mathcal{L}$ choose some f such that

$$f(s') = f(S) \, \forall s' \subseteq S. \tag{28}$$

and assign a sparse cost

$$C_f(L) = \begin{cases} k & \text{if } L = S \\ 0 & \text{otherwise.} \end{cases} \tag{29}$$

We can use this to represent C as an approximate sum of these new functions C_f

$$C(L(\mathbf{x})) \approx \sum_{f \in \mathcal{F}} C_f(f(L(\mathbf{x}))) \tag{30}$$

Having decomposed the cost function into a set of symmetric distributions, we alter the integer program of equations refeg:ip as follows: Letting $S_f = \{f(L) : L \subseteq \mathcal{L}\}$ for all f, we write

$$\begin{aligned} \min_{\mathbf{z}} \sum_{i \in \mathcal{V}, a \in \mathcal{L}} \psi_i(a) z_{i;a} + \sum_{\substack{(i,j) \in \mathcal{E}, \\ a,b \in \mathcal{L}}} \psi_{i,j}(a,b) z_{ij;ab} \\ + \sum_{\substack{L_f \in S_f, \\ f \in \mathcal{F}}} C_f(L_f) z_{f;L} \end{aligned} \tag{31}$$

Such that

$$\begin{aligned}
&\textstyle\sum_a z_{ab;ij} = z_{b;j}, && \forall (i,j) \in \mathcal{E}, b \in \mathcal{L}, \\
&\textstyle\sum_b z_{ab;ij} = z_{a;i}, && \forall (i,j) \in \mathcal{E}, a \in \mathcal{L}, \\
&\textstyle\sum_a z_{a;i} = 1, && \forall i \in \mathcal{V}, \\
&\textstyle\sum_{f(a) \in L_f \in S_f} z_{f;L_f} \leq \sum_{i \in \mathcal{V}} z_{i;a}, && \forall a \in \mathcal{L}, \qquad (32) \\
&z_{f;L} \geq z_{a;i}, && \forall i \in \mathcal{V}, a \in \mathcal{L} : a \in f(L) \qquad (33) \\
&z_{a;i} \in \{0,1\}, z_{ab;ij} \in \{0,1\} && \forall i \in \mathcal{V} \qquad (34) \\
&\forall (i,j) \in \mathcal{E}, && \forall a, b \in \mathcal{L}, \\
&z_{f;L_f} \in \{0,1\}, && \forall f \in \mathcal{F}, L_f \subseteq f(\mathcal{L}). \qquad (35)
\end{aligned}$$

Where the new term within the cost function (31), $C_f(L_f)$, represents the cost associated with $f(L(\mathbf{x})) = L_f$. The terms (32, 33, 35) are an application of the constraints (13 - 15 *main paper*) to each cost function $\{C_f : \forall f \in \mathcal{F}\}$, enforcing the constraint that $f(L(\mathbf{x})) = L_f$.

This approach can be used to build a first order approximation of a cost over a label space. We take the sum of cost functions C_β defined over β and $\mathcal{L} \backslash \{\beta\}$ $\forall \beta \in \mathcal{L}$ and approximate $C(L(\mathbf{x}))$ as follows

$$C(L(\mathbf{x})) \approx \sum_{\beta \in \mathcal{L}} C_\beta(f_\beta(L)) \qquad (36)$$

where

$$f_\beta(L) = \begin{cases} \mathcal{L} \backslash \{\beta\} & \text{if } \beta \notin L \\ \mathcal{L} & \text{otherwise,} \end{cases} \qquad (37)$$

and

$$C_\beta(f_\beta(L)) = \begin{cases} k_\beta & \text{if } f_\beta(L) = \beta \\ 0 & \text{otherwise} \end{cases} \qquad (38)$$

Such a first order approximation is an *occurrence* potential, which captures the frequency with which each label is likely to occur.

Similarly, a second order approximation can be formed as follows

$$C(L(\mathbf{x})) \approx \sum_{\beta \in \mathcal{L}} C_\beta(f_\beta(L)) + \sum_{\beta_1, \beta_2 \in \mathcal{L}} C_{\beta_1,\beta_2}(f_{\beta_1,\beta_2}(L)) \qquad (39)$$

where C_β, f_β is defined as above,

$$f_{\beta_1,\beta_2}(L) = \begin{cases} \mathcal{L} \backslash \{\beta_1, \beta_2\} & \text{if } \beta_1 \notin L \\ \mathcal{L} \backslash \{\beta_2\} \text{ if} & \beta_1 \in L, \beta_2 \notin L \\ \mathcal{L} & \text{otherwise} \end{cases} \qquad (40)$$

and

$$C_{\beta_1,\beta_2}(f_{\beta_1,\beta_2}(L)) = \begin{cases} k_{\beta_1,\beta_2} - k_{\beta_1} - k_{\beta_2} & \text{if } \beta_1, \beta_2 \in L \\ 0 & \text{otherwise,} \end{cases} \qquad (41)$$

These second order approximations are *co-occurrence* potentials that as well as capturing the relative occurrence of each label, also express the likelihood of pairs of labels occurring together.

This combination of *first order* occurrence potentials and *second order* co-occurrence potentials is equivalent to a Taylor series based approximation of $C(\cdot)$ truncated beyond the second degree.

While the above simplification of the initial formulation allow for more efficient inference, it still requires a large linear program to be solved, and does not scale efficiently. In some sense we are hampered by the fact that standard LP solvers are ill-suited for inference over many large problems encountered in computer vision, necessitating the use of the alternate methods of inference.

References

1. Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23:2001, 2001. 2
2. V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(10):1568–1583, 2006. 2
3. N. Komodakis, G. Tziritas, and N. Paragios. Fast, approximately optimal solutions for single and dynamic mrfs. In *CVPR*, 2007. 2
4. M. Kumar and P. Torr. Efficiently solving convex relaxations for map estimation. In *ICML*, 2008. 2
5. R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields. In *ECCV*, 2006. 2
6. A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Context-based vision system for place and object recognition. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 273–280 vol.1, 2003. 5
7. M. Wainwright, T. Jaakkola, and A. Willsky. Map estimation via agreement on trees: message-passing and linear programming. *IEEE Transactions on Information Theory*, 51(11):3697–3717, 2005. 2
8. T. Werner. A linear programming approach to max-sum problem: A review. Research Report CTU–CMP–2005–25, Center for Machine Perception, Czech Technical University, December 2005. 2