Image Segmentation Using Higher-Order Correlation Clustering

Sungwoong Kim, *Member, IEEE,* Chang D. Yoo, *Senior Member, IEEE,* Sebastian Nowozin, and Pushmeet Kohli

Abstract—In this paper, a hypergraph-based image segmentation framework is formulated in a supervised manner for many high-level computer vision tasks. To consider short- and long-range dependency among various regions of an image and also to incorporate wider selection of features, a higher-order correlation clustering (HO-CC) is incorporated in the framework. Correlation clustering (CC), which is a graph-partitioning algorithm, was recently shown to be effective in a number of applications such as natural language processing, document clustering, and image segmentation. It derives its partitioning result from a pairwise graph by optimizing a global objective function such that it simultaneously maximizes both intra-cluster similarity and inter-cluster dissimilarity. In the HO-CC, the pairwise graph which is used in the CC is generalized to a hypergraph which can alleviate local boundary ambiguities that can occur in the CC. Fast inference is possible by linear programming relaxation, and effective parameter learning by structured support vector machine is also possible by incorporating a decomposable structured loss function. Experimental results on various datasets show that the proposed HO-CC outperforms other state-of-the-art image segmentation algorithms. The HO-CC framework is therefore an efficient and flexible image segmentation framework.

Index Terms—Image segmentation, correlation clustering, structural learning.

1 Introduction

Image segmentation which can defined as a clustering of image pixels into disjoint coherent regions is currently being used in many of the state-of-the-art high-level image/scene understanding tasks such as object class segmentation, scene segmentation, surface layout labeling, and single view 3D reconstruction [1]–[5]. Its use provides the following three benefits: (1) coherent support regions, commonly assumed to be of a single label, serve as a good prior for many labeling tasks; (2) these coherent regions allow extraction of a more consistent feature that provides surrounding contextual information through pooling many feature responses over the region; and (3) a small number of larger coherent regions, compared to large number of pixels, significantly reduces the computational cost for a labeling task.

Many segmentation algorithms have been proposed in the literature that can be broadly classified into two groups – graph based (examples include min-cuts [6], normalized cuts [7] and Felzenszwalb-Huttenlocher (FH) segmentation algorithm [8]) and non graph based (examples include K-means [9], mean-shift [10], and EM [11]). Compared to non-graph-based segmentations, *graph-based* segmentations have been shown to produce more consistent segmentations by adaptively balancing local judgements of similarity [12]. Graph-based image segmentation algorithms can be further categorized into either node-labeling or edge-labeling algorithms. In contrast to the node-labeling framework of the min-cuts and normalized cuts, the *edge-labeling* framework of the FH algorithm does not require a pre-specified number of segmentations in an image.

Correlation clustering (CC) is a graph-partitioning algorithm [13] that infers the edge labels of the graph by simultaneously maximizing intra-cluster similarity and inter-cluster dissimilarity by optimization of a global objective (discriminant) function. Furthermore, the CC can be formulated as a linear discriminant function which allows for approximate polynomial-time inference by linear programming (LP) and also allows large margin training based on structured support vector machine (S-SVM) [14]. Finley *et al.* [15] consider a framework that uses the S-SVM for training the parameters in the CC for noun-phrase clustering and news article clustering. Taskar derived a max-margin formulation, different from the S-SVM, for learning the edge scores in the CC [16] for applications involving two different segmentations of a single image. No experimental comparisons or quantitative results are provided in [16].

We have recently explored a supervised CC over a pairwise superpixel graph for task-specific image segmentation [17], and it has been shown to perform

- *S. Kim is with Qualcomm Research Korea, 119 Nonhyeon Dong, Gangnam Gu, Seoul 135-820, South Korea.*
  E-mail: sungwoong.kim01@gmail.com
- *C. Yoo is with the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, 373-1 Guseong Dong, Yuseong Gu, Daejeon 305-701, South Korea.*
  E-mail: cdyoo@ee.kaist.ac.kr
- *S. Nowozin and P. Kohli are with Microsoft Research Cambridge.*
  E-mail: Sebastian.Nowozin@microsoft.com, pkohli@microsoft.com

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. NRF-2011-0017202 and No. NRF-2010-0028680).

better than other state-of-the-art image segmentation algorithms.

Although it derives its segmentation result by optimizing a global objective function, which leads to a discriminatively-trained discriminant function, the pairwise CC (PW-CC) is restricted to resolving segment boundary ambiguities corresponding to only local pairwise edge labels of a graph. Therefore, to capture long-range dependencies of distant nodes in a global context, this paper proposes *higher-order correlation clustering* (HO-CC) to incorporate higher-order relations. Generalizing the PW-CC over a pairwise superpixel graph, we develop a HO-CC over a *hypergraph* that considers higher-order relations among superpixels. An edge in the hypergraph of the proposed HO-CC can connect to two or more nodes representing the superpixels as in [18].

Hypergraphs have been previously used to lift certain limitations of conventional pairwise graphs [19]–[21]. However, previously proposed hypergraphs for image segmentation are restricted to partitioning based on the generalization of a normalized-cut framework, which suffer from the following three difficulties. *First*, inference is slow and difficult especially with increasing graph size. To approximate the inference process, a number of algorithms have been introduced based on the coarsening algorithm [20] and the hypergraph Laplacian matrices [19]. These are heuristic approaches and therefore sub-optimal. *Second*, incorporating a supervised learning algorithm for parameter estimation under the spectral hypergraph partitioning framework is difficult. This is in line with the difficulties in learning spectral graph partitioning. This requires a complex and unstable eigenvector approximation which must be differentiable [22], [23]. *Third*, region-based features are utilized in a restricted manner. Almost all previous hypergraph-based image segmentation algorithms have been restricted to color variances as region features.

The proposed HO-CC framework alleviates all of the above difficulties by generalizing the PW-CC and making use of the hypergraph. The hypergraph which is constructed based on the correlation information of the superpixels can be equivalently formulated as a linear discriminant function. A richer feature vector involving higher-order relations among visual cues of the superpixels can be utilized. For fast inference, a LP relaxation is used, and for tractable S-SVM training of the parameters with unbalance class labeled data, a decomposable structured-loss function is defined, which allows the efficient use of the cutting-plane algorithm to approximately solve the constrained optimization. Experimental results on various datasets show that the proposed HO-CC outperforms other state-of-the-art image segmentation algorithms.

An earlier version of this paper appeared as Kim *et al.* [24]. This paper provides a more detailed description of the proposed HO-CC, additional empirical results, and in-depth analysis of the performances on image segmentation tasks.

Our main contributions can be summarized as follows: (1) the hypergraph-based HO-CC approach that takes into account higher-order relationships between super-pixels; (2) inference using a LP relaxation of the problem; (3) using supervised learning for discriminative clustering via a cutting plane algorithm that can handle a decomposable loss function; and (4) the demonstration of segmentation results that improve on those obtained by state-of-the-art segmentations methods.

The rest of the paper is organized as follows. Section 2 describes the PW-CC in [17], and Section 3 presents the proposed HO-CC. Section 4 describes structural learning for supervised image segmentation based on the S-SVM and cutting plane algorithm. A number of experimental and comparative results are presented and discussed in Section 5, followed by a conclusion in Section 6.

2 Pairwise Correlation Clustering

As alluded earlier, the CC is basically an algorithm to partition a *pairwise* graph into disjoint groups of coherent nodes [13], and it has been used in natural language processing and document clustering [15], [25], [26]. This section presents the *PW-CC* that has been developed to solve an image segmentation task by partitioning a pairwise superpixel graph [17].

2.1 Superpixels

The proposed image segmentation is based on superpixels which are small coherent regions preserving almost all boundaries between different regions. This is an advantage since superpixels significantly reduce computational cost and allow feature extraction to be conducted from a larger coherent region. Both the pairwise and higher-order CC merges superpixels into disjoint coherent regions over a superpixel graph. Therefore, the proposed CC is not a replacement to existing superpixel algorithms, and performances might be influenced by baseline superpixels.

2.2 Pairwise Correlation Clustering over a Pairwise Superpixel Graph

Define a pairwise undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where a node corresponds to a superpixel and a link between adjacent superpixels corresponds to an edge (see Figure 1.(a)). A binary label y_{jk} for an edge $(j,k) \in \mathcal{E}$ between nodes j and k is defined such that

$$y_{jk} = \begin{cases} 1, & \text{if } j \text{ and } k \text{ belong to the same region,} \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

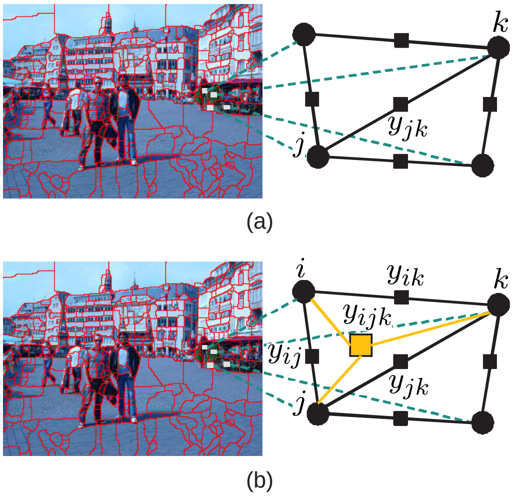

Fig. 1. Illustrations of a part of (a) the pairwise graph (b) and the triplet graph built on superpixels.

A discriminant function is defined over image $\mathbf{x}$ and label $\mathbf{y}$ of all edges as

$$\begin{aligned} F(\mathbf{x},\mathbf{y};\mathbf{w}) &= \sum_{(j,k)\in\mathcal{E}} \mathrm{Sim}_{\mathbf{w}}(\mathbf{x},j,k) y_{jk} && (2)\\ &= \sum_{(j,k)\in\mathcal{E}} \langle \mathbf{w}, \phi_{jk}(\mathbf{x})\rangle y_{jk} && (3)\\ &= \langle \mathbf{w}, \sum_{(j,k)\in\mathcal{E}} \phi_{jk}(\mathbf{x}) y_{jk}\rangle && (4)\\ &= \langle \mathbf{w}, \Phi(\mathbf{x},\mathbf{y})\rangle, && (5) \end{aligned}$$

where the similarity measure between nodes j and k, $\mathrm{Sim}_{\mathbf{w}}(\mathbf{x},j,k)$, is parameterized by $\mathbf{w}$ and takes values of both signs such that a large positive value indicates strong similarity while a large negative value indicates strong dissimilarity. Note that the discriminant function $F(\mathbf{x},\mathbf{y};\mathbf{w})$ is assumed to be linear in both the parameter vector $\mathbf{w}$ and the joint feature map $\Phi(\mathbf{x},\mathbf{y})$, and $\phi_{jk}(\mathbf{x})$ is a pairwise feature vector which reflects the correspondence between the jth and the kth superpixels. An image segmentation is to infer the edge label $\mathbf{y}$ over the pairwise superpixel graph $\mathcal{G}$ by maximizing F such that

$$\hat{\mathbf{y}} = \operatorname*{argmax}_{\mathbf{y}\in\mathcal{Y}(\mathcal{G})} F(\mathbf{x},\mathbf{y};\mathbf{w}), \tag{6}$$

where $\mathcal{Y}(\mathcal{G})$ is a subset of $\{0,1\}^{\mathcal{E}}$ that corresponds to a *valid segmentation* and is the set of multicuts [27] of the graph $\mathcal{G}$. However, solving (6) over $\mathcal{Y}(\mathcal{G})$ is generally NP-hard.

2.3 LP Relaxation for Pairwise Correlation Clustering

We approximate $\mathcal{Y}(\mathcal{G})$ by means of a common multicut LP relaxation [27], [28] with the following two constraints: (1) cycle inequality and (2) odd-wheel inequality. The LP relaxation to approximately solve (6) can be formulated as

$$\begin{aligned} \operatorname*{argmax}_{\mathbf{y}} \quad & \sum_{(j,k)\in\mathcal{E}} \langle \mathbf{w}, \phi_{jk}(\mathbf{x})\rangle y_{jk} \\ \text{s.t.} \quad & \mathbf{y}\in\mathcal{Z}(\mathcal{G}), \end{aligned} \tag{7}$$

where $\mathcal{Z}(\mathcal{G}) \supset \mathcal{Y}(\mathcal{G})$ is a relaxed polytope defined by the following two linear inequalities.

1) Cycle inequality: Let $\mathrm{Path}(j,k)$ be the set of paths between nodes j and k. The cycle inequality is a generalization of the triangle inequality [27] and is defined as

$$(1-y_{jk}) \le \sum_{(s,t)\in p} (1-y_{st}), \quad p\in\mathrm{Path}(j,k). \tag{8}$$

2) Odd-wheel inequality: Let a q-wheel be a connected subgraph $\mathcal{S} = (\mathcal{V}_s,\mathcal{E}_s)$ with a central vertex $j\in\mathcal{V}_s$ and a cycle of q vertices in $\mathcal{C} = \mathcal{V}_s\setminus\{j\}$. For every odd $q(\ge 3)$-wheel, a valid segmentation $\mathbf{y}$ satisfies

$$\sum_{(s,t)\in\mathcal{E}(\mathcal{C})} (1-y_{st}) - \sum_{k\in\mathcal{C}} (1-y_{jk}) \le \lfloor \frac{1}{2} q \rfloor, \tag{9}$$

where $\mathcal{E}(\mathcal{C})$ denotes the set of all edges in the outer cycle $\mathcal{C}$.

Although the number of inequalities (8) and (9) is exponentially large in the size of the graph, it is nevertheless possible to optimize (7) in polynomial time. The identification of a violated inequality –the so called *separation problem*– from both sets (8) and (9) is possible in polynomial time [29], [30]. A famous result in combinatorial optimization states the equivalence between optimization and separation [31]. Thus, the polynomial time solvability of (7) is guaranteed.

The relation between the solutions of (6) and (7) is as follows: if the LP solution to (7) is integral, that is for all $(j,k)\in\mathcal{E}$ we have $y_{jk}\in\{0,1\}$, then the solution $\mathbf{y}$ is the exact solution to (6). If instead, it is fractional, then we take the floor of a fractionally-predicted label of each edge independently for simply obtaining a feasible but potentially sub-optimal solution to (6).

2.4 The Need for Higher-Order Models

Even though the PW-CC described above can use a rich pairwise feature vector with an optimized parameter vector (which will be presented later), it often produces incorrectly predicted segments due to segment boundary ambiguities caused by limited pairwise relations of neighboring superpixels (see Figure 2). Therefore, to incorporate higher-order relations of distant superpixels, we develop a HO-CC by generalizing the CC over a hypergraph.

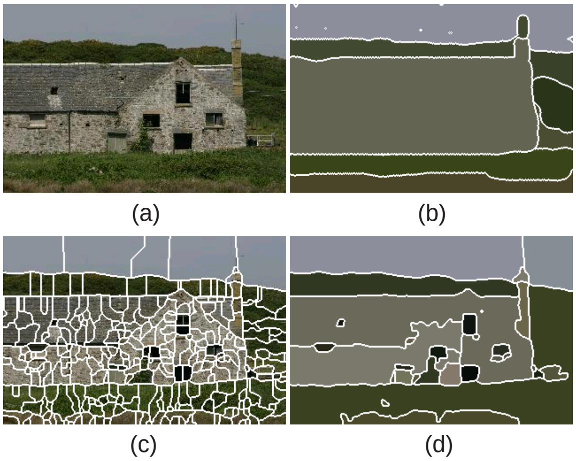

Fig. 2. Example of segmentation result by PW-CC. (a) Original image. (b) Ground-truth. (c) Superpixels. (d) Segments obtained by PW-CC.

3 Higher-Order Correlation Clustering

This section describes the proposed HO-CC for image segmentation in three steps. In the first step, we define the hypergraph representation. Second, we generalize the LP relaxation (7) for hypergraphs. Finally, a feature vector consisting of pairwise and higher-order feature vectors to characterize relationship among superpixels over a hypergraph is presented.

3.1 Hypergraph

The proposed HO-CC is defined over a hypergraph in which an edge referred to as *hyperedge* can connect to two or more nodes. For example, as shown in Figure 1.(b), one can introduce binary labels for each adjacent vertices forming a triplet such that $y_{ijk} = 1$ if all vertices in $\{i, j, k\}$ are in the same cluster; otherwise, $y_{ijk} = 0$. Define a hypergraph $\mathcal{HG} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ is the set of all nodes (superpixels) and $\mathcal{E}$ is the set of all hyperedges (subsets of $\mathcal{V}$) such that $\bigcup_{e \in \mathcal{E}} = \mathcal{V}$. Here, a hyperedge e has at least two nodes, i.e. $|e| \geq 2$. Therefore, the hyperedge set $\mathcal{E}$ can be divided into two disjoint subsets: pairwise edge set $\mathcal{E}_p = \{e \in \mathcal{E} \mid |e| = 2\}$ and higher-order edge set $\mathcal{E}_h = \{e \in \mathcal{E} \mid |e| > 2\}$ such that $\mathcal{E}_p \bigcup \mathcal{E}_h = \mathcal{E}$. Note that in the proposed hypergraph for HO-CC all hyperedges containing just two nodes ($\forall e_p \in \mathcal{E}_p$) are linked between adjacent superpixels. The pairwise superpixel graph is a special hypergraph where all hyperedges contain just two (neighboring) superpixels: $\mathcal{E}_p = \mathcal{E}$. A binary label y_e for a hyperedge $e \in \mathcal{E}$ is defined such that

$$y_e = \begin{cases} 1, & \text{if all nodes in } e \text{ belong to the same region,} \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

3.2 Higher-Order Correlation Clustering over a Hypergraph

Similar to the PW-CC, a linear discriminant function is defined over image $\mathbf{x}$ and label $\mathbf{y}$ of all hyperedges as

$$\begin{aligned} F(\mathbf{x}, \mathbf{y}; \mathbf{w}) & \\ &= \sum_{e \in \mathcal{E}} \mathrm{Hom}_{\mathbf{w}}(\mathbf{x}, e) y_e & (11) \\ &= \sum_{e \in \mathcal{E}} \langle \mathbf{w}, \phi_e(\mathbf{x}) \rangle y_e & (12) \\ &= \sum_{e_p \in \mathcal{E}_p} \langle \mathbf{w}_p, \phi_{e_p}(\mathbf{x}) \rangle y_{e_p} + \sum_{e_h \in \mathcal{E}_h} \langle \mathbf{w}_h, \phi_{e_h}(\mathbf{x}) \rangle y_{e_h} & (13) \\ &= \langle \mathbf{w}, \Phi(\mathbf{x}, \mathbf{y}) \rangle, & (14) \end{aligned}$$

where the homogeneity measure among nodes in e, $\mathrm{Hom}_{\mathbf{w}}(\mathbf{x}, e)$, is also the inner product of the parameter vector $\mathbf{w}$ and the feature vector $\phi_e(\mathbf{x})$ and takes values of both signs such that a large positive value indicates strong homogeneity while a large negative value indicates high degree of non-homogeneity. Note that the proposed discriminant function for the HO-CC is decomposed into two terms by assigning different parameter vectors to the pairwise edge set $\mathcal{E}_p$ and the higher-order edge set $\mathcal{E}_h$ such that $\mathbf{w} = [\mathbf{w}_p; \mathbf{w}_h]$. Thus, in addition to the pairwise similarity between neighboring superpixels, the proposed HO-CC considers a broad homogeneous region reflecting higher-order relations among superpixels.

From a given image, a hypergraph is constructed as follows. First, unsupervised multiple partitionings are obtained by merging not pixels but superpixels with different image quantizations using the ultrametric contour maps [32]. Then, the obtained regions are used to define hyperedges of the hypergraph. For example, in Figure 3, there are three region layers, one superpixel (pairwise) layer and two higher-order layers. All edges (black line) in the pairwise superpixel graph from the first layer are incorporated into the pairwise edge set $\mathcal{E}_p$. Hyperedges (yellow line) corresponding to regions (groups of superpixels) in the second and third layers are included in the higher-order edge set $\mathcal{E}_h$. Note that we can further decompose the higher-order term in (13) into two terms associated with the second and third layers, respectively, by assigning different parameter vectors; however for simplicity, this paper aggregates all higher-order edges from all higher-order layers into a single higher-order edge set assigning the same parameter vector.

The use of unsupervised multiple partitionings enables to obtain reasonable candidate regions for defining higher-order edges. Other methods to define higher-order edges are also possible. For instance, from the baseline pairwise superpixel graph, the fully connected subgraphs referred to as cliques which have more than two nodes can be obtained, and these cliques can be associated to the higher-order

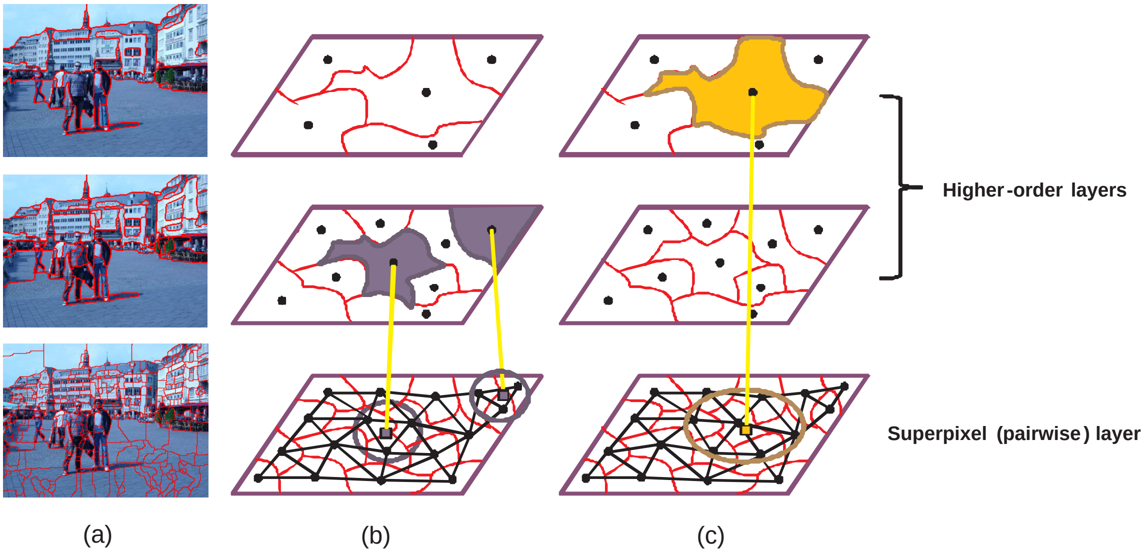

Fig. 3. Hypergraph construction from multiple partitionings. (a) Multiple partitionings from baseline superpixels. (b) Hyperedge (yellow line) corresponding to a region in the second layer. (c) Hyperedge (yellow line) corresponding to a region in the third layer.

edges. However, the use of the cliques in the proposed framework is empirically hard to produce broad regions which consist of more than four fully connected superpixels.

3.3 LP Relaxation for Higher-Order Correlation Clustering

An image segmentation is to infer the hyperedge label, $\hat{\mathbf{y}}$, over the hypergraph $\mathcal{HG}$ by maximizing the discriminant function F such that

$$\hat{\mathbf{y}} = \operatorname*{argmax}_{\mathbf{y} \in \mathcal{Y}(\mathcal{HG})} F(\mathbf{x}, \mathbf{y}; \mathbf{w}), \tag{15}$$

where $\mathcal{Y}(\mathcal{HG})$ is also the subset of $\{0,1\}^{\mathcal{E}}$ that corresponds to a *valid segmentation.*

In order to define the state of the higher-order edge variables in relations to the pairwise edge variables, we introduce two types of inequalities: the *first* enforces that when a pairwise edge places (or labels) adjacent superpixels belonging to a certain higher-order edge as being in different clusters, the higher-order edge cannot place the two in the same cluster; and the *second* enforces that when all pairwise edges of a set of superpixels agree that all superpixels in the set are in the same cluster, then the higher-order edge of the set must place all the superpixels as belonging to one cluster (see Table 1). We define novel constraints for labels on pairwise and higher-order edges, referred to as *higher-order inequalities*, to formalize this intuition as follows:

$$\begin{aligned} y_{e_h} &\le y_{e_p}, \quad \forall e_p \in \mathcal{E}_p | e_p \subset e_h, \\ (1 - y_{e_h}) &\le \sum_{e_p \in \mathcal{E}_p | e_p \subset e_h} (1 - y_{e_p}). \end{aligned} \tag{16}$$

Proposition The set of all binary solutions satisfying the inequalities (8), (9), and (16), which forms the HO-CC problem, represents exactly the set of consistent cluster assignments.

Proof. All solutions satisfying the pairwise inequalities (8) and (9) lead to consistent pairwise edge label assignments. Inclusion of (16) does not make any pairwise solution inconsistent. Also, by formalizing the above intuitive reasoning as (16), for binary variables, all higher-order edge assignments are consistent with all pairwise edge assignments.

The LP relaxation to approximately solve (15) is formulated as

$$\begin{aligned} &\operatorname*{argmax}_{\mathbf{y}} \sum_{e_p \in \mathcal{E}_p} \langle \mathbf{w}_p, \phi_{e_p}(\mathbf{x}) \rangle y_{e_p} + \sum_{e_h \in \mathcal{E}_h} \langle \mathbf{w}_h, \phi_{e_h}(\mathbf{x}) \rangle y_{e_h} \quad (17) \\ &\text{s.t.} \qquad \mathbf{y} \in \mathcal{Z}(\mathcal{HG}), \end{aligned}$$

where $\mathcal{Z}(\mathcal{HG}) \supset \mathcal{Y}(\mathcal{HG})$ is the relaxed polytope defined by the cycle inequality of (8), odd-wheel inequality of (9), and higher-order inequality of (16).

Due to the exponentially large number of constraints, we use the cutting plane algorithm [33], which is summarized in Algorithm 1, to solve (17) efficiently. The algorithm works with a small set of constraints that defines a loose relaxation S to the feasible set. It iteratively tightens S by means of violated inequalities. In each iteration, the optimal $\mathbf{y}$ on the current set of constraints is found, then violated inequalities are searched. When a violated inequality is found, it is added to the current constraint set to reduce S, and (17) is re-solved with the tightened relaxation (reduced S). Here, the search for a violated inequality runs in polynomial time.

Note that the proposed HO-CC follows the concept

TABLE 1
Label validity for segmentation from the hypergraph (triplet graph) in Figure 1.(b).

y_{ijk}	0	0	0	0	0	0	0	0
y_{ij}	0	0	0	0	1	1	1	1
y_{jk}	0	0	1	1	0	0	1	1
y_{ik}	0	1	0	1	0	1	0	1
Validity	valid	valid	valid	invalid	valid	invalid	invalid	invalid
y_{ijk}	1	1	1	1	1	1	1	1
y_{ij}	0	0	0	0	1	1	1	1
y_{jk}	0	0	1	1	0	0	1	1
y_{ik}	0	1	0	1	0	1	0	1
Validity	invalid	invalid	invalid	invalid	invalid	invalid	invalid	valid

Algorithm 1 Cutting Plane Algorithm for Inference

```
Input: $\mathbf{w}$, $S \leftarrow [0,1]^{\mathcal{E}}$
repeat
  Solve LP relaxation on the current constraint set:
  $\mathbf{y} \leftarrow \operatorname{argmax}_{\mathbf{y}\in S} \sum_{e_p\in\mathcal{E}_p} \langle \mathbf{w}_p, \phi_{e_p}(\mathbf{x})\rangle y_{e_p} + \sum_{e_h\in\mathcal{E}_h} \langle \mathbf{w}_h, \phi_{e_h}(\mathbf{x})\rangle y_{e_h}$
  $S_{\text{violated}} \leftarrow$ VIOLATE CYCLE INEQUALITIES $(\mathbf{y})$: check (8)
  if no violated inequality found then
    $S_{\text{violated}} \leftarrow$ VIOLATE HIGHER-ORDER INEQUALITIES $(\mathbf{y})$: check (16)
    if no violated inequality found then
      Integrality check
      if no fractional-predicted label then
        break
      else
        $S_{\text{violated}} \leftarrow$ VIOLATE ODD-WHEEL INEQUALITIES $(\mathbf{y})$: check (9)
      end if
    end if
  end if
  $S \leftarrow S \cap S_{\text{violated}}$
until no $S$ has changed
```

of *soft constraints*: superpixels belonging to a hyperedge are not forced but encouraged to merge if a hyperedge is highly homogeneous. This is in line with recent higher-order models for high-level image understanding [1], [34], [35].

3.4 Feature Vector

We construct a 481-dimensional feature vector $\phi_e(\mathbf{x}) = [\phi_{e_p}(\mathbf{x}); \phi_{e_h}(\mathbf{x})]$ by concatenating several visual cues with different quantization levels and thresholds. The pairwise feature vector $\phi_{e_p}(\mathbf{x})$ reflects the correspondence between neighboring superpixels, and the higher-order feature vector $\phi_{e_h}(\mathbf{x})$ characterizes a more complex relations among superpixels in a broader region to measure homogeneity. The magnitude of $\mathbf{w}$ determines the importance of each feature, and this importance is task-dependent. Thus, $\mathbf{w}$ is estimated by supervised training described in Section 4.

3.4.1 Pairwise feature vector

We extract several visual cues from a superpixel, including brightness (intensity), color, texture, and shape. Based on these visual cues, we construct a 321-dimensional pairwise feature vector ϕ_{e_p} by concatenating a color difference feature ϕ^c, texture difference feature ϕ^t, shape/location difference feature ϕ^s, edge strength feature ϕ^e, joint visual word posterior feature ϕ^v, and bias as follows:

$$\phi_{e_p} = [\phi^c_{e_p}; \phi^t_{e_p}; \phi^s_{e_p}; \phi^e_{e_p}; \phi^v_{e_p}; 1]. \tag{18}$$

- Color difference feature $\phi^c_{e_p}$: The color difference feature $\phi^c_{e_p}$ is composed of 26 color distances between two adjacent superpixels based on RGB and HSV channels. Specifically, we calculate 18 earth mover's distances (EMDs) [36] between two color histograms extracted from each superpixel with various numbers of bins and thresholds for ground distance. In addition, six absolute differences (one for each color channel) between the means of the two superpixels and two χ^2-distances between hue/saturation histograms of the two superpixels are concatenated in $\phi^c_{e_p}$.
- Texture difference feature $\phi^t_{e_p}$: The 64-dimensional texture difference feature $\phi^t_{e_p}$ is composed of 15 absolute differences (one for each texture-response) between the means of two superpixels using 15 Leung-Malik (LM) filter banks [37] and one χ^2-distance and 48 EMDs (from various numbers of bins and thresholds for ground distance) between texture histograms of the two superpixels.
- Shape/location difference feature $\phi^s_{e_p}$: The 5-dimensional shape/location difference feature $\phi^s_{e_p}$ is composed of two absolute differences between the normalized (x/y) center positions of the two superpixels, the ratio of the size of the smaller superpixel to that of the larger superpixel, the percentage of boundary with respect to the smaller superpixel, and the straightness of boundary [4].

- Edge strength feature $\phi^e_{e_p}$: The 15-dimensional edge strength feature $\phi^e_{e_p}$ is a 1-of-15 coding of the quantized edge strength proposed by Arbelaez *et al.* [32].
- Joint visual word posterior feature $\phi^v_{e_p}$: The 210-dimensional joint visual word posterior feature $\phi^v_{e_p}$ is defined as the vector holding the joint visual word posteriors for a pair of neighboring superpixels using 20 visual words [38] as follows. First, a 52-dimensional raw feature vector x_j based on color, texture, location, and shape features described in [4] is extracted from the jth superpixel. Then, the visual word posterior distribution $P(v_i|x_j)$ is computed using the Gaussian RBF kernel where v_i denotes the ith visual word. Let $V_{jk}(\mathbf{x})$ be a 20-by-20 matrix whose elements are the joint visual word posteriors between nodes j and k defined such that

$$V_{jk}(\mathbf{x}) = \begin{bmatrix} P(v_1|x_j)P(v_1|x_k) \cdots P(v_1|x_j)P(v_{20}|x_k) \\ P(v_2|x_j)P(v_1|x_k) \cdots P(v_2|x_j)P(v_{20}|x_k) \\ \vdots \qquad \ddots \qquad \vdots \\ P(v_{20}|x_j)P(v_1|x_k) \cdots P(v_{20}|x_j)P(v_{20}|x_k) \end{bmatrix}. \quad (19)$$

The joint visual word posterior feature between nodes j and k, $\phi^v_{jk}(\mathbf{x})$, is defined as

$$\phi^v_{jk}(\mathbf{x}) = \mathrm{vec}\Big(V_{jk}(\mathbf{x})\Big) + \mathrm{vec}\Big(V^T_{jk}(\mathbf{x})\Big), \quad (20)$$

where vec(V) be the $210 (= 20 \times 21/2)$-dimensional vector whose elements are from the upper triangular part of V.
This joint visual word posterior feature could overcome the weakness of class-agnostic features and incorporate the contextual information.

3.4.2 Higher-order feature vector

We construct a 160-dimensional higher-order feature vector ϕ_{e_h} by concatenating the variance feature $\phi^{va}_{e_h}$, edge strength feature $\phi^e_{e_h}$, template matching feature $\phi^{tm}_{e_h}$ and bias as follows:

$$\phi_{e_h} = [\phi^{va}_{e_h}; \phi^e_{e_h}; \phi^{tm}_{e_h}; 1]. \quad (21)$$

- Variance feature $\phi^{va}_{e_h}$: The 44-dimensional variance feature is a generalized version of the color/texture difference feature used in the pairwise graph. We calculate 14 color variances among superpixels in a hyperedge based on the average RGB and HSV values and the hue/saturation histograms with 8 bins. In addition, 30 texture variances from 15 mean texture responses and texture response histogram with 15 bins are incorporated into the variance feature vector.
- Edge strength feature $\phi^e_{e_h}$: The 15-dimensional edge strength feature $\phi^e_{e_h}$ is a ℓ_1-normalized histogram of the quantized edge strengths of neighboring superpixels in e_h.
- Template matching feature $\phi^{tm}_{e_h}$: The 44-dimensional color/texture features and 5-dimensional shape/location features of all (task-specific ground truth) regions in the training images are clustered using k-means with $k = 100$ to obtain 100 representative templates of distinct regions. The 100-dimensional template matching feature vector is composed of the matching scores between a region defined by hyperedge and these templates using the Gaussian RBF kernel.

Note that in each feature vector, the bias (=1) is augmented in order to obtain a proper similarity/homogeneity measure which can either be positive or negative.

4 Structural learning

The proposed discriminant function is defined over the superpixel graph, and therefore, the ground-truth segmentation needs to be transformed to the ground-truth edge labels in the superpixel graph. For this, we first assign a single dominant segment label to each superpixel by majority voting over the superpixel's constituent pixels and then obtain the ground-truth edge labels according to whether dominant labels of superpixels in a hyperedge are equal or not.

Using this ground-truth edge labels of the training data, we use the S-SVM to estimate the parameter vector for task-specific CC. We use the cutting plane algorithm with LP relaxation (17) for loss-augmented inference to solve the optimization problem of the S-SVM, since fast convergence and high robustness of the cutting plane algorithm in handling a large number of margin constraints are well-known [14].

4.1 Structured Support Vector Machine

Given N training samples $\{(\mathbf{x}^n, \mathbf{y}^n)\}_{n=1}^N$ where $\mathbf{y}^n$ is the ground-truth edge labels for the nth training image $\mathbf{x}^n$, the S-SVM [14] optimizes $\mathbf{w}$ by minimizing a quadratic objective function subject to a set of linear margin constraints:

$$\begin{aligned} \min_{\mathbf{w},\xi} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^N \xi_n \qquad (22)\\ \text{s.t.} \quad & \forall n, \mathbf{y} \in \mathcal{Z}(\mathcal{HG}), \\ & \langle \mathbf{w}, \Delta\Phi(\mathbf{x}^n, \mathbf{y})\rangle \geq \Delta(\mathbf{y}^n, \mathbf{y}) - \xi_n, \\ & \forall n, \quad \xi_n \geq 0, \end{aligned}$$

where $\Delta\Phi(\mathbf{x}^n, \mathbf{y}) = \Phi(\mathbf{x}^n, \mathbf{y}^n) - \Phi(\mathbf{x}^n, \mathbf{y})$, and $C > 0$ is a constant that controls the trade-off between margin maximization and training error minimization. In the S-SVM, the margin is scaled with a loss $\Delta(\mathbf{y}^n, \mathbf{y})$, which is the difference measure between prediction $\mathbf{y}$ and ground-truth label $\mathbf{y}^n$ of the nth image. The S-SVM offers good generalization ability as well as the flexibility to choose any loss function [14].

Algorithm 2 Cutting Plane Algorithm for S-SVM

```
**Choose:** $\mathbf{w}_0$, $C$, $R$, $\epsilon$
$S_n \leftarrow \emptyset, \ \forall n, \quad \mathbf{w} \leftarrow \mathbf{w}_0, \quad \xi \leftarrow \mathbf{0}$
**repeat**
  **for** $n = 1, ..., N$ **do**
    Perform the loss-augmented inference by LP relaxation:
      $\hat{\mathbf{y}}^n = \operatorname*{argmax}_{\mathbf{y} \in \mathcal{Z}(\mathcal{HG})} \Big( \langle \mathbf{w}, \Phi(\mathbf{x}^n, \mathbf{y}) \rangle + \Delta(\mathbf{y}^n, \mathbf{y}) \Big)$
    **if** $-\langle \mathbf{w}, \delta\Phi(\mathbf{x}^n, \hat{\mathbf{y}}^n) \rangle + \Delta(\mathbf{y}^n, \hat{\mathbf{y}}^n) > \xi_n + \epsilon$ **then**
      $S_n \leftarrow S_n \cup \{\hat{\mathbf{y}}^n\}$
    **end if**
  **end for**
  Solve the restricted problem of (22) on the current set of constraints:
    $(\mathbf{w}^*, \xi^*) = \operatorname*{argmin}_{\mathbf{w}', \xi'} \ \frac{1}{2}\|\mathbf{w}'\|^2 + C \sum_{n=1}^{N} \xi'_n$
  s.t. $\langle \mathbf{w}', \delta\Phi(\mathbf{x}^n, \mathbf{y}) \rangle \geq \Delta(\mathbf{y}^n, \mathbf{y}) - \xi'_n, \ \forall n, \mathbf{y} \in S_n,$
       $\xi'_n \geq 0, \quad \forall n$
  Update: $\mathbf{w} \leftarrow \mathbf{w}^*, \quad \xi \leftarrow \xi^*$
**until** no $S_n$ has changed
```

4.2 Cutting Plane Algorithm

The exponentially large number of margin constraints and the intractability of the loss-augmented inference problem make it difficult to solve the constrained optimization problem of (22). Therefore, we apply the cutting plane algorithm [14] to approximately solve the constrained optimization problem. The cutting plane algorithm is summarized in Algorithm 2. In each iteration, the most violated constraint for each training sample is approximately found by performing the loss-augmented inference using the LP relaxation. The computational cost for inference can be greatly reduced when a decomposable loss such as the Hamming loss is used. When a loss function can be decomposed in the same manner as the joint feature map, it can be added to each edge score in the inference. It can then be checked whether the constraint found tightens the feasible set of (22) or not, and when it does, then the parameter vector $\mathbf{w}$ and ξ are updated by solving the restricted problem of (22) on the current set of active constraints that includes it. The theoretical convergence and robustness of the cutting plane algorithm was studied by Tsochantaridis *et al.* [14]. The LP relaxations for loss-augmented inferences are considered to be well suited to structured learning [39]–[41].

4.3 Label Loss

A non-negative and decomposable loss function $\Delta : \mathcal{Z} \times \mathcal{Z} \rightarrow \mathbb{R}_+$ enables efficient loss-augmented inference in the cutting plane algorithm. The loss can be absorbed into the edge homogeneity, and the loss-augmented inferencing can be performed by the LP relaxation which is used in the original inference.

TABLE 2
Label loss at the edge level.

y_e^n	0	1	0	1
y_e	0	1	1	0
Δ_e	0	0	1	R

The most popular loss function that is non-negative and decomposable is the Hamming loss which is equivalent to the number of mismatches between $\mathbf{y}^n$ and $\mathbf{y}$ at the edge level in this CC. In the proposed CC for image segmentation, however, the number of edges which are labeled as 1 is considerably higher than that of edges which are labeled as 0. This imbalance leads to the clustering of the whole image as one segment when we use the Hamming loss in the S-SVM. Therefore, we use the following modified Hamming loss function:

$$\Delta(\mathbf{y}^n, \mathbf{y}) = \sum_{e \in \mathcal{E}} \Delta_e(y_e^n, y_e) \tag{23}$$

$$= \sum_{e_p \in \mathcal{E}_p} \Big(R_p y_{e_p}^n + y_{e_p} - (R_p + 1) y_{e_p}^n y_{e_p} \Big)$$

$$+ D_n \sum_{e_h \in \mathcal{E}_h} \Big(R_h y_{e_h}^n + y_{e_h} - (R_h + 1) y_{e_h}^n y_{e_h} \Big), \tag{24}$$

where D_n is the relative weight of the loss at higher-order edge level to that of the loss at pairwise edge level. In addition, R_p and R_h control the relative importance between the incorrect merging of the superpixels and the incorrect separation of the superpixels by imposing different weights to the false negative and the false positive, as shown in Table 2. Here, we set $D_n = \frac{|\mathcal{E}_p|}{|\mathcal{E}_h|}$, and both R_p and R_h are set to be less than 1 to overcome the unbalanced problem mentioned above.

5 Experiments

To evaluate segmentations obtained by various algorithms against the ground-truth segmentation, we conducted image segmentations on three benchmark datasets: Stanford background dataset [2] (SBD), Berkeley segmentation dataset (BSDS) [42], and MSRC dataset [43]. For image segmentation based on CC, we initially obtain baseline superpixels (438 superpixels per image on average) by the gPb contour detector and the oriented watershed transform [32] and then construct a hypergraph. The function parameters are initially set to zero, and then based on the S-SVM, the structured output learning is used to estimate the parameter vectors. Note that the relaxed solutions in loss-augmented inference are used during training, while in testing, our simple rounding method is used to produce valid segmentation results. Rounding is only necessary when the LP relaxation fails to be exact, that is, when fractional solutions from LP-relaxed CC are obtained.

We compared the proposed HO-CC to the following three unsupervised and three supervised image segmentation algorithms:

- Mean-shift: Comaniciu and Meer [10] devised a mode-seeking algorithm to locate points of locally-maximal density in a feature space.
- Multiscale NCut: Cour *et al.* [44] devised a multiscale spectral image segmentation algorithm by decomposing an image partitioning graph into different scales in the normalized cut framework.
- gPb-owt-ucm: The oriented watershed transform - ultrametric contour map algorithm [32] produces hierarchical regions of superpixels obtained by using the gPb contour detector.
- gPb-Hoiem: Hoiem *et al.* [4] grouped superpixels based on pairwise same-label likelihoods. The superpixels were obtained by the gPb contour detector, and the pairwise same-label likelihoods estimated by a boosted decision tree were independently learnt from the training data where the same 321-dimensional pairwise feature vector was used as an input to the boosted decision tree.
- Supervised NCut: A supervised learning algorithm for parameter estimation under the normalized cut framework is applied. For this, the affinity matrix on the same pairwise superpixel graph is defined as

  $$A_{jk} = \begin{cases} \min(1, \exp\{-\langle \mathbf{w}, \phi_{jk} \rangle\}), & \text{if } (j,k) \in \mathcal{E}, \\ 0, & \text{otherwise,} \end{cases}$$

  where the same 321-dimensional pairwise feature vector ϕ_{jk} was used. Afterwards, the standard pairwise affinity learning with the square-square loss function and the gradient descent algorithm [45] is used for supervised training.
- PW-CC: The PW-CC is described in Section 2. A pairwise superpixel graph is obtained with the same 321-dimensional pairwise feature vector.

Note that we used the codes publicly released by the authors for Mean-shift, (multiscale) NCut, gPb-owt-ucm, and gPb-hoiem. Specifically, when we performed the supervised image segmentation algorithms such as the gPb-hoiem and supervised NCut, we modified each code to use the same pairwise feature vector as for our method.

We consider four performance measures: *probabilistic Rand index* (PRI) [46], *segmentation covering* (SCO) [32], *variation of information* (VOI) [47], and *boundary displacement error* (BDE) [48]. When the predicted segmentation is close to the ground-truth segmentation, the PRI and SCO increases while the VOI and BDE decreases.

An implementation of the HO-CC is available at http://slsp.kaist.ac.kr/xe/?mid=software.

5.1 Stanford Background Dataset

The SBD consists of 715 outdoor images with corresponding pixel-wise annotations such that each pixel is labeled with either one of 7 background classes or a generic foreground class. From the given pixel-wise ground-truth annotations, we obtain a ground-truth segmentation for each image. We employed 5-fold cross-validation with the dataset randomly split into 572 training images and 143 test images for each fold.

Figure 4 shows the four measures obtained from segmentation results according to the average number of regions. Note that the performance varies with different numbers of regions, and for this reason, we designed each algorithm to produce multiple segmentations (20 to 40 regions). Specifically, multiple segmentations in the proposed algorithm were obtained by varying R_p (0.01∼0.15) and R_h (0.4∼0.6) in the loss function during training. When R_h is fixed, as R_p increases, the number of segmented regions of a test image tends to decrease, since the false negative error is penalized more compared to the false positive error. The same observation is also verified when R_p is fixed and R_h increases. Irrespective of the measure, the proposed HO-CC performed better than other algorithms including the PW-CC.

Figure 5 shows some examples of segmentations. The proposed HO-CC yielded the best segmentation results. Incorrectly predicted segments by the PW-CC were reduced in the segmentation results obtained by the HO-CC owing to the higher-order relations in broad regions. The gPb-Hoiem and the supervised NCut treat each edge as an independent pairwise instance during training, therefore, the segmentation results are not stable (producing inconsistant local regions) even though it uses the same pairwise features.

Regarding the runtime of our algorithm, we observed that for test-time inference it took on average around 15 seconds (graph construction and feature extraction: 14s, LP: 1s) per image on a 2.67GHz processor, whereas the overall training took 20 hours on the training set. In terms of the LP runtime, HO-CC took about four times more time than PW-CC on average.

The performance improvement is obtained from both higher-order features and higher-order constraints. Segmentation results obtained by HO-CC without higher-order features were observed to be very similar to those obtained by PW-CC: without higher-order features, higher-order constraints did not tighten the relaxation for PW-CC. However, as shown in Figure 6, we observed that the performance gap between the HO-CC with the full higher-order feature vector (160-dim) and the HO-CC with the simple higher-order feature vector (45-dim, variances only) was smaller than that between the HO-CC with the simple higher-order feature vector and the PW-CC.

In order to confirm improvements obtained by HO-

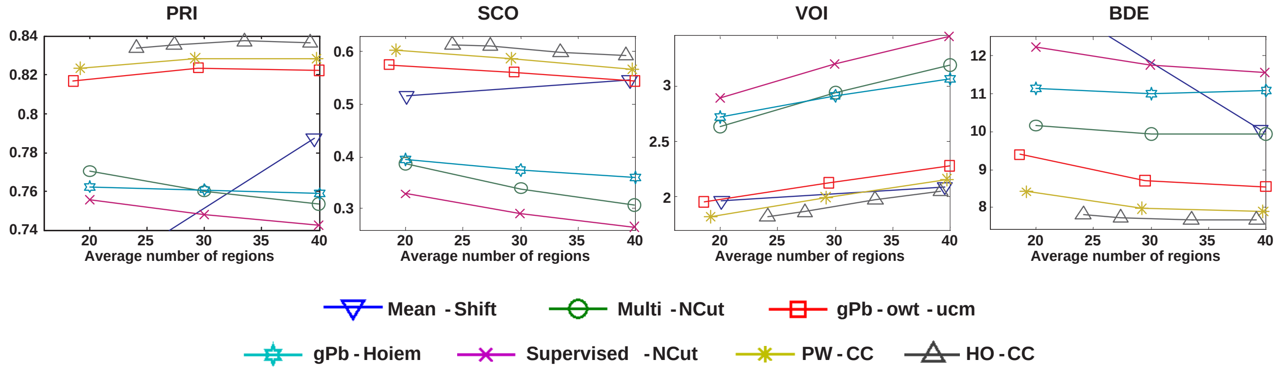

Fig. 4. Obtained evaluation measures from segmentation results on the SBD.

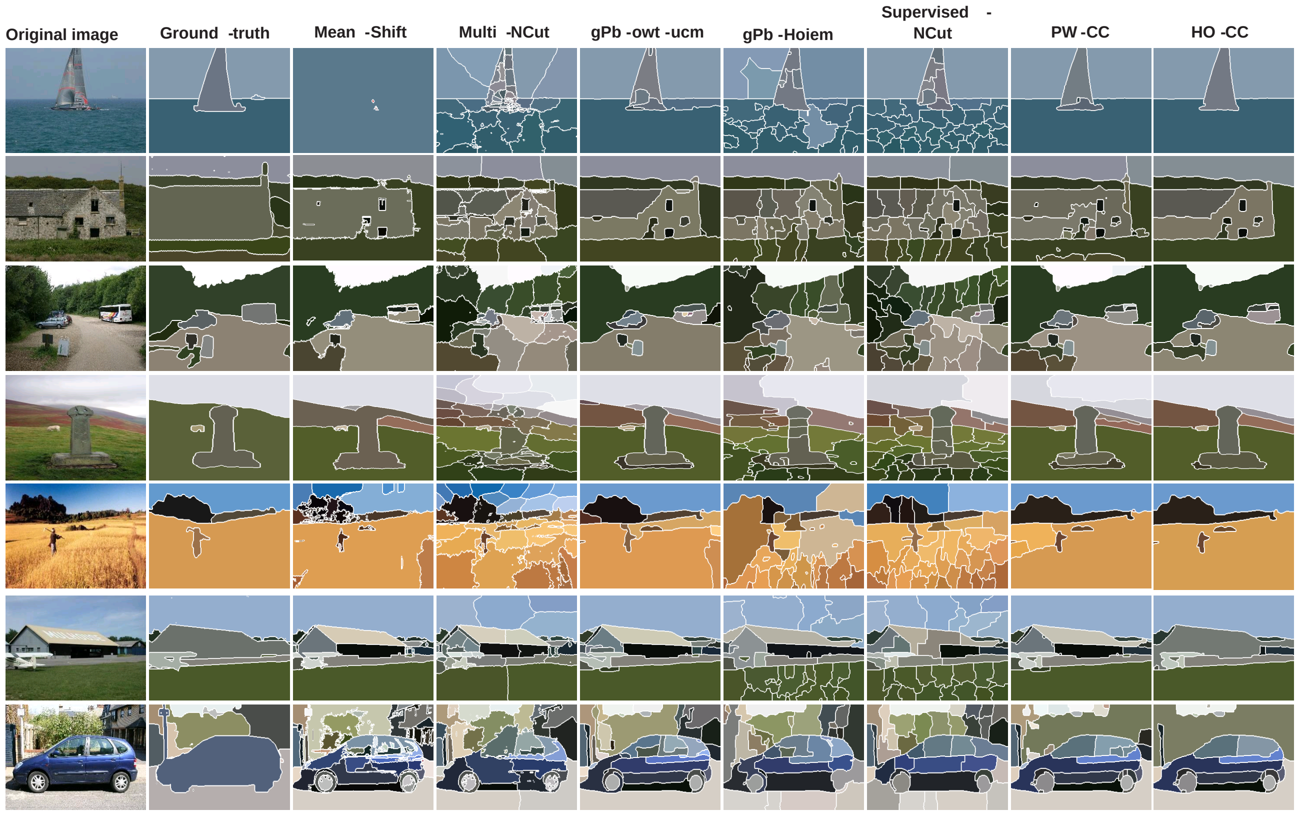

Fig. 5. Results of image segmentation on the SBD.

CC are statistically significant, we performed statistical hypothesis tests for each performance measure. The Friedman test [49], [50] was used to evaluate the null-hypothesis that all the algorithms perform equally well. Table 3 shows the obtained average ranks. Under the null-hypothesis, all average ranks should be equal. However, as shown in Table 3, the ranks are different, and the null-hypothesis is rejected for all the four measures. This is also verified by the obtained p-values which are numerically equal to zeros for all the four measures. Furthermore, we performed a post-hoc test, called Nemenyi test [50], [51] for pairwise comparison of algorithms, testing for the null-hypothesis of pairwise equal performance. The Nemenyi test is based on the difference of the average performance ranks achieved by the algorithms; if the difference between two ranks exceeds a critical value, the null-hypothesis is refuted. As a result, at the level $\alpha = 0.05$, with the PRI and BDE measures, the HO-CC is statistically significantly superior to all other algorithms except PW-CC, with the VOI measures, the HO-CC is statistically significantly superior to all other algorithms except Mean-shift, and with the SCO measures, the HO-CC is statistically significantly superior to all other algorithms.

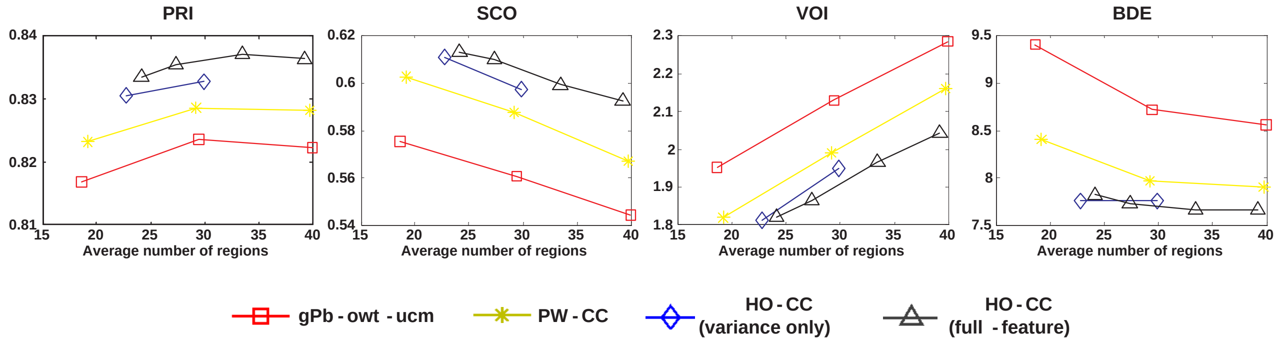

Fig. 6. Obtained evaluation measures from segmentation results according to the different set of features on the SBD.

TABLE 3
Average ranks by Friedman test on the SBD.

Average ranks	Mean-shift	Multi-NCut	gPb-owt-ucm	gPb-Hoiem	Supervised-NCut	PW-CC	HO-CC
PRI	4.0168	5.0951	3.1524	4.6923	5.5692	2.8797	2.5944
SCO	2.8559	5.8350	2.8587	4.9860	6.5497	2.6462	2.2685
VOI	2.5287	5.6587	3.1203	5.0895	6.4671	2.8028	2.3329
BDE	4.2783	4.4042	3.3203	4.9133	5.4350	2.9399	2.7091

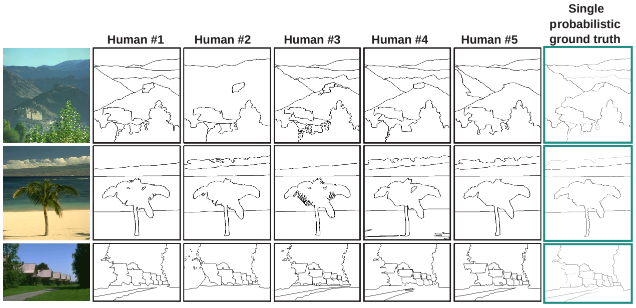

Fig. 7. Examples of partitionings by multiple human subjects and single probabilistic (real-valued) ground-truth partitioning.

5.2 Berkeley Segmentation Dataset

The BSDS300 contains 300 natural images split into the 200 training images and 100 test images. Since each image is segmented by multiple human subjects, we defined a single probabilistic (real-valued) ground-truth segmentation of each image for training in the proposed HO-CC (see Figure 7). The gPb-Hoiem and the supervised NCut used a different ground-truth for training on the BSDS: declare two superpixels to lie in the same segment only if all human subjects declare them to lie in the same segment.

Table 4 and Figure 8 shows the obtained results at a universal fixed scale (ODS) in terms of various performance measures including the boundary F-measure and the boundary precision-recall curve. Note that for each algorithm, the same parameters which produce the best F-measure were used for all other performance measures in evaluating algorithms.

TABLE 4
Quantitative results on the BSDS300 test set.

BSDS300 test set	PRI	SCO	VOI	BDE	F
Mean-shift	0.668	0.501	1.962	25.945	0.512
Multi-NCut	0.718	0.263	3.458	14.383	0.595
gPb-owt-ucm	0.807	0.571	2.039	11.001	0.710
gPb-Hoiem	0.724	0.334	3.014	14.651	0.621
Supervised-NCut	0.713	0.235	3.632	16.443	0.545
PW-CC	0.806	0.585	1.829	11.194	0.715
HO-CC	**0.814**	**0.599**	**1.743**	**10.377**	**0.722**

For example, the level-threshold of 0.12 for gPb-owt-ucm, R_p of 0.15 for the PW-CC, and (R_p, R_h) of (0.01,0.1) for the HO-CC were used for producing segmentation results at ODS listed in Table 4, since these values gave the best results with regards to the F-measure. Irrespective of the measure, the proposed HO-CC gave the best results, which are similar or even better than the best results ever reported on the BSDS300 [32], [52], [53].

We changed the level-threshold for the gPb-owt-ucm and R_p and R_h for the PW-CC and HO-CC to produce different numbers of regions per image, on average, and observed that the HO-CC always performed better than the PW-CC and the gPb-owt-ucm (see Figure 9), as on the SBD. Improvement of 1% in PRI, 1.5% in SCO, 0.1 in VOI, and 1 pixel in BDE on the BSDS test set is comparable to the improvements reported in [32], [52] (1% in PRI, 2% in SCO, 0.05 in VOI, and 1 pixel in BDE). We observed that in comparison to the PW-CC, by the proposed HO-CC, 78 segmentation results were improved, 9 results did not change, and the rest 13 results got worse on the

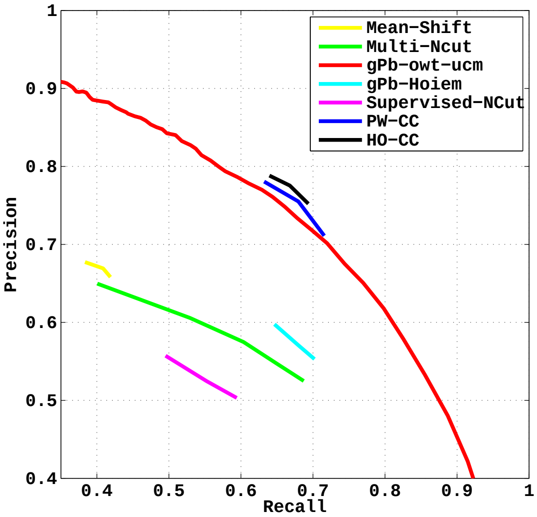

Fig. 8. Boundary precision-recall curve on the BSDS300 test set.

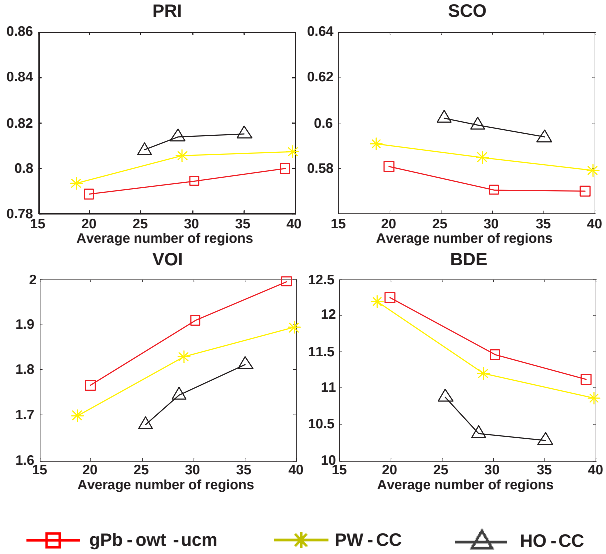

Fig. 9. Obtained evaluation measures from segmentation results of gPb-owt-ucm, PW-CC, and HO-CC on the BSDS300 test set according to the average number of regions.

TABLE 5
Quantitative results on the BSDS500 test set.

BSDS500 test set	PRI	SCO	VOI	BDE	F
gPb-owt-ucm	0.825	0.579	1.971	9.995	0.726
PW-CC	0.826	0.589	1.859	9.812	0.728
HO-CC	**0.828**	**0.595**	**1.791**	**9.770**	**0.730**

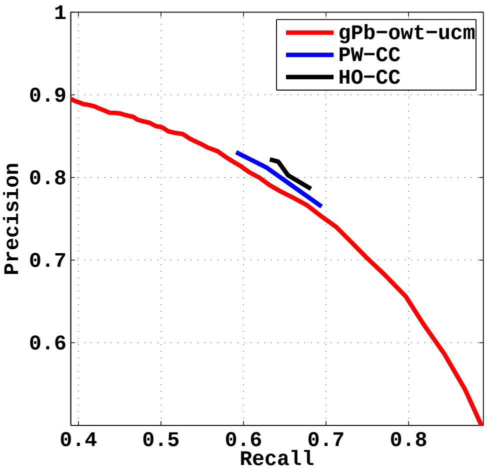

Fig. 10. Boundary precision-recall curve on the BSDS500 test set.

BSDS test set.

We also performed experiments on the BSDS500 dataset and obtained the results at ODS. As shown in Table 5 and Figure 10, the HO-CC performed the best on the BSDS500.

We increased the number of layers from two to three by splitting the original higher-order layer into two layers according to the edge-strengths obtained from the gPb-owt, then assigned different parameter vectors to each layer. The obtained performance is shown in Table 6. The performance of the hypergraph which has the three layers (HO-CC-Layer3) was a little improved in comparison to that of the hypergraph which has the two layers (HO-CC-Layer2). This small improvement is due to a small number of hyperedges associated with the third layer.

The performances obtained by HO-CC might be influenced by candidate regions for defining higher-order edges. Therefore, we used a different superpixel-grouping method – category independent object proposals (CIOP) [54]. As shown in the Table 7, the hypergraphs based on the gPb-owt performed a little better than that based on the gPb-CIOP, but the gap is not critical.

Figure 11 shows some example segmentations on BSDS test images obtained by various segmentation algorithms. The proposed HO-CC yielded the best segmentation results.

5.3 MSRC Dataset

The MSRC dataset is composed of 591 natural images. We split the data into 45% training, 10% validation, and 45% test sets, following [43]. We used the ground-truth object instance labeling of [55], which does not contain void regions and is more precise than the original ground-truth, for both training and testing (including the performance evaluation) on the MSRC. On average, all partitioning algorithms were set to produce approximately 15 disjoint regions per image on the MSRC dataset. Regarding the performances according to the number of regions, we observed the same tendency on the MSRC dataset as on the

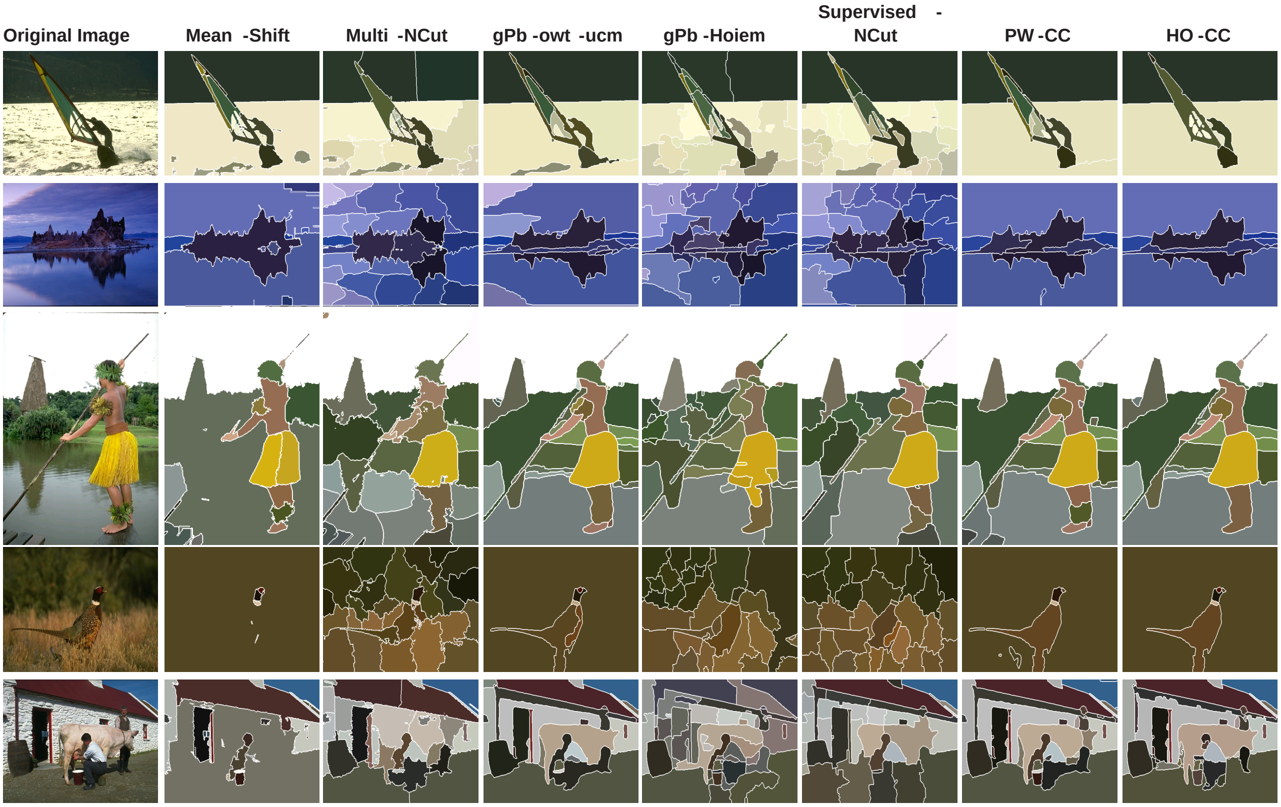

Fig. 11. Results of image segmentation on the BSDS test set.

TABLE 6
Quantitative results on the BSDS500 test set according to the number of layers.

BSDS500 test set	PRI	SCO	VOI	BDE	F
HO-CC-Layer2	0.828	0.595	1.791	9.770	0.730
HO-CC-Layer3	0.829	0.599	1.786	9.764	0.730

TABLE 7
Quantitative results on the BSDS500 test set according to different superpixel-groupings for hypergraph construction.

BSDS500 test set	PRI	SCO	VOI	BDE	F
HO-CC-gPb-owt	0.828	0.595	1.791	9.770	0.730
HO-CC-gPb-CIOP	0.826	0.592	1.801	9.797	0.728

BSDS dataset. As shown in Table 8 and Figure 12, the proposed HO-CC gave the best results on the test set.

We also trained on the MSRC dataset and tested on the BSDS dataset. This decreases the performance over training and testing on the BSDS dataset. This observation is also true in the reverse direction, i.e. when training on the BSDS dataset and testing on the MSRC dataset. Overall, this suggests that the two datasets have different statistics, and the proposed framework allows the segmentation to be tuned to the particular dataset at hand.

TABLE 8
Quantitative results on the MSRC test set.

MSRC test set	PRI	SCO	VOI	BDE
Mean-shift	0.734	0.606	1.649	13.944
Multi-NCut	0.628	0.341	2.765	11.941
gPb-owt-ucm	0.779	0.628	1.675	9.800
gPb-Hoiem	0.614	0.353	2.847	13.533
Supervised-NCut	0.601	0.287	3.101	13.498
PW-CC	0.773	0.632	1.648	9.194
HO-CC	**0.784**	**0.648**	**1.594**	**9.040**

6 Conclusion

This paper proposed the HO-CC over a hypergraph to merge superpixels into homogeneous regions. The LP relaxation was used to approximately solve the inference problem over a hypergraph where a rich feature vector was defined based on several visual cues involving higher-order relations among superpixels. The S-SVM was used for supervised training of parameters in CC, and the cutting plane algorithm with LP-relaxed inference was applied to solve the optimization problem of S-SVM. Experimental results showed that the proposed HO-CC outperformed other image segmentation algorithms on various datasets. The proposed framework is applicable to a variety of other tasks.

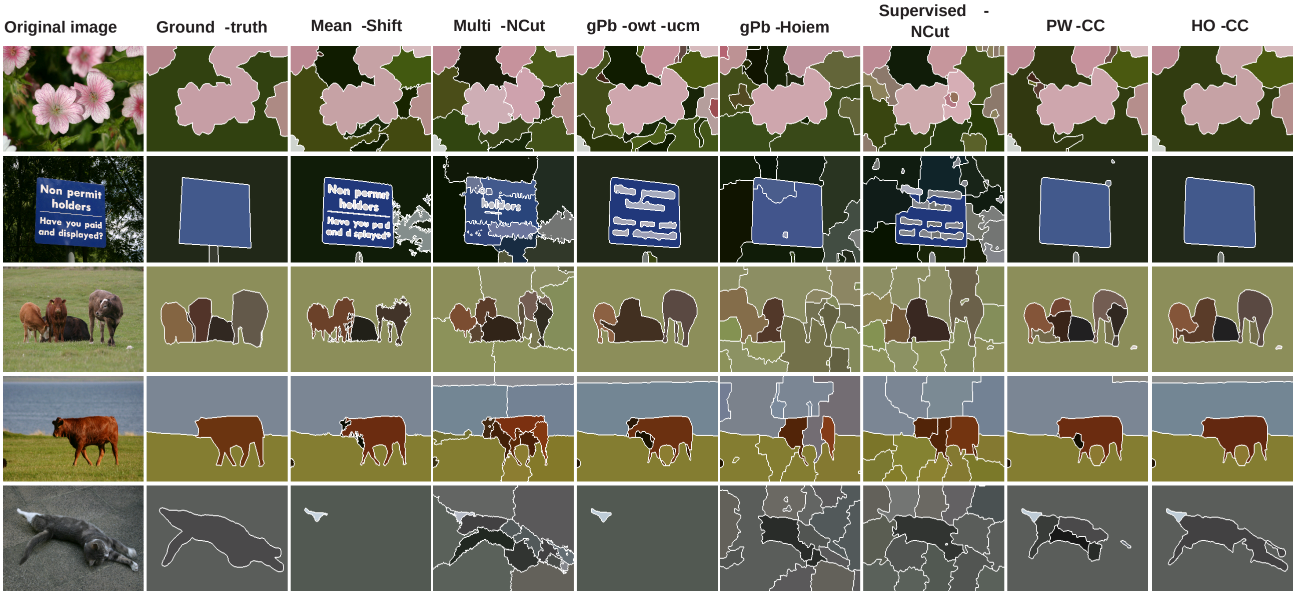

Fig. 12. Results of image segmentation on the MSRC test set.

References

[1] L. Ladický, C. Russell, P. Kohli, and P. H. S. Torr, "Associative hierarchical CRFs for object class image segmentation," in *Proc. IEEE International Conference on Computer Vision*, 2009.

[2] S. Gould, R. Fulton, and D. Koller, "Decomposing a scene into geometric and semantically consistent regions," in *Proc. IEEE International Conference on Computer Vision*, 2009.

[3] M. P. Kumar and D. Koller, "Efficiently selecting regions for scene understanding," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[4] D. Hoiem, A. A. Efros, and M. Hebert, "Recovering surface layout from an image," *International Journal of Computer Vision*, vol. 75, pp. 151–172, 2007.

[5] B. Liu, S. Gould, and D. Koller, "Single image depth estimation from predicted semantic labels," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[6] F. Estrada and A. Jepson, "Spectral embedding and mincut for image segmentation," in *Proc. British Machine Vision Conference (BMVC)*, 2004.

[7] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 888–905, 2000.

[8] P. Felzenszwalb and D. Huttenlocher, "Efficient graph-based image segmentation," *International Journal of Computer Vision*, vol. 59, pp. 167–181, 2004.

[9] T. Kanungo, D. Mount, N. Netanyahu, C. Piatko, R. Silverman, and A. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 881–892, 2002.

[10] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 603–619, 2002.

[11] C. Carson, S. Belongie, H. Greenspan, and J. Malik, "Blobworld: image segmentation using expectation-maximization and its application to image querying," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 1026–1038, 2002.

[12] F. Estrada and A. Jepson, "Benchmarking image segmentation algorithms," *International Journal of Computer Vision*, vol. 85, pp. 167–181, 2009.

[13] N. Bansal, A. Blum, and S. Chawla, "Correlation clustering," *Machine Learning*, vol. 56, pp. 89–113, 2004.

[14] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and independent output variables," *Journal of Machine Learning Research*, vol. 6, pp. 1453–1484, 2005.

[15] T. Finley and T. Joachims, "Supervised clustering with support vector machines," in *Proc. International Conference on Machine Learning*, 2005.

[16] B. Taskar, "Learning structured prediction models: a large margin approach," *Ph.D. thesis, Stanford University*, 2004.

[17] S. Kim, S. Nowozin, P. Kohli, and C. D. Yoo, "Task-specific image partitioning," *IEEE Transactions on Image Processing*, vol. 22, pp. 488–500, 2013.

[18] C. Berge, *Hypergraphs*. North-Holland, Amsterdam, 1989.

[19] L. Ding and A. Yilmaz, "Image segmentation as learning on hypergraphs," in *Proc. International Conference on Machine Learning and Applications*, 2008.

[20] S. Rital, "Hypergraph cuts and unsupervised representation for image segmentation," *Fundamenta Informaticae*, vol. 96, pp. 153–179, 2009.

[21] A. Ducournau, S. Rital, A. Bretto, and B. Laget, "A multilevel spectral hypergraph partitioning approach for color image segmentation," in *Proc. IEEE International Conference on Signal and Image Processing Applications*, 2009.

[22] F. Bach and M. I. Jordan, "Learning spectral clustering," in *Proc. Neural Information Processing Systems*, 2003.

[23] T. Cour, N. Gogin, and J. Shi, "Learning spectral graph segmentation," in *Proc. International Conference on Artificial Intelligence and Statistics*, 2005.

[24] S. Kim, S. Nowozin, P. Kohli, and C. D. Yoo, "Higher-order correlation clustering for image segmentation," in *Proc. Neural Information Processing Systems*, 2011.

[25] T. Joachims and J. E. Hopcroft, "Error bounds for correlation clustering," in *Proc. International Conference on Machine Learning*, 2005.

[26] A. McCallum and B. Wellner, "Toward conditional models of identity uncertainty with application to proper noun coreference," in *Proc. IJCAI Workshop on Information Integration on the Web*, 2003.

[27] S. Chopra and M. R. Rao, "The partition problem," *Math. Program*, vol. 59, pp. 87–115, 1993.

[28] S. Nowozin and S. Jegelka, "Solution stability in linear programming relaxations: Graph partitioning and unsupervised learning," in *Proc. International Conference on Machine Learning*, 2009.

[29] M. M. Deza, M. Grötschel, and M. Laurent, "Clique-web facets for multicut polytopes," *Mathematics of Operations Research*, vol. 17, no. 4, pp. 981–1000, 1992.

[30] M. M. Deza and M. Laurent, *Geometry of cuts and metrics*, ser. Algorithms and Combinatorics, 1997, vol. 15.

[31] M. Grötschel, L. Lovász, and A. Schrijver, "The ellipsoid

method and its consequences in combinatorial optimization," *Combinatorica*, vol. 1, pp. 169–197, 1981.

[32] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, pp. 898–916, 2011.

[33] L. Wolsey, *Integer programming*. John Wiley, 1998.

[34] P. Kohli, L. Ladickỳ, and P. H. S. Torr, "Robust higher order potentials for enforcing label consistency," *International Journal of Computer Vision*, vol. 82, pp. 302–324, 2009.

[35] L. Ding and A. Yilmaz, "Interactive image segmentation using probabilistic hypergraphs," *Pattern Recognition*, vol. 43, pp. 1863–1873, 2010.

[36] O. Pele and M. Werman, "Fast and robust earth mover's distances," in *Proc. IEEE International Conference on Computer Vision*, 2009.

[37] T. Leung and J. Malik, "Representing and recognizing the visual appearance of materials using three-dimensional textons," *International Journal of Computer Vision*, vol. 43, pp. 29–44, 2001.

[38] D. Batra, R. Sukthankar, and T. Chen, "Learning class-specific affinities for image labelling," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[39] T. Finley and T. Joachims, "Training structural SVMs when exact inference is intractable," in *Proc. International Conference on Machine Learning*, 2008.

[40] A. Kulesza and F. Pereira, "Structured learning with approximate inference," in *Proc. Neural Information Processing Systems*, 2007.

[41] A. F. T. Martins, N. A. Smith, and E. P. Xing, "Polyhedral outer approximations with application to natural language parsing," in *Proc. International Conference on Machine Learning*, 2009.

[42] C. Fowlkes, D. Martin, and J. Malik, *The Berkeley Segmentation Dataset and Benchmark (BSDB)*, http://www.cs.berkeley.edu/projects/vision/grouping/.

[43] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: joint appearence, shape and context modeling for multi-class object recognition and segmentation," in *Proc. European Conference on Computer Vision (ECCV)*, 2006.

[44] T. Cour, F. Benezit, and J. Shi, "Spectral segmentation with multiscale graph decomposition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[45] S. Turaga, K. Briggman, M. Helmstaedter, W. Denk, and H. Seung, "Maximin affinity learning of image segmentation," in *Proc. Neural Information Processing Systems*, 2009.

[46] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical Association*, vol. 66, pp. 846–850, 1971.

[47] M. Meila, "Computing clusterings: An axiomatic view," in *Proc. International Conference on Machine Learning*, 2005.

[48] J. Freixenet, X. Munoz, D. Raba, J. Marti, and X. Cufi, "Yet another survey on image segmentation: Region and boundary information integration," in *Proc. European Conference on Computer Vision (ECCV)*, 2002.

[49] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," *Annals of Mathematical Statistics*, vol. 11, pp. 86–92, 1940.

[50] J. Demsar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

[51] P. B. Nemenyi, "Distribution-free multiple comparisons," *Ph.D. thesis, Princeton University*, 1963.

[52] T. Kim, K. Lee, and S. Lee, "Learning full pairwise affinities for spectral segmentation," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[53] S. R. Rao, H. Mobahi, A. Y. Yang, S. S. Sastry, and Y. Ma, "Natural image segmentation with adaptive texture and boundary encoding," in *Proc. Asian Conference on Computer Vision (ACCV)*, 2009.

[54] I. Endres and D. Hoiem, "Category independent object proposals," in *Proc. European Conference on Computer Vision (ECCV)*, 2010.

[55] T. Malisiewicz and A. A. Efros, "Improving spatial support for objects via multiple segmentations," in *Proc. British Machine Vision Conference (BMVC)*, 2007.

Sungwoong Kim (S'07-M'12) received the B.S. and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2004 and 2011, respectively. Since 2012 he is with Qualcomm Research Korea where he is a senior engineer. His research interests include machine learning for multimedia signal processing, discriminative training, and graphical modeling.

Chang D. Yoo (S'92-M'96-SM'11) received the B.S. degree in Engineering and Applied Science from California Institute of Technology in 1986, the M.S. degree in Electrical Engineering from Cornell University in 1988 and the Ph.D. degree in Electrical Engineering from Massachusetts Institute of Technology in 1996. From January 1997 to March 1999 he worked at Korea Telecom as a Senior Researcher. He joined the Department of Electrical Engineering at Korea Advanced Institute of Science and Technology in April 1999. From March 2005 to March 2006, he was with Research Laboratory of Electronics at MIT. His current research interests are in the application of machine learning and digital signal processing in multimedia.

Sebastian Nowozin is a researcher in the Machine Learning and Perception group at Microsoft Research Cambridge. He received his Master of Engineering degree from the Shanghai Jiaotong University (SJTU) and his diploma degree in computer science with distinction from the Technical University of Berlin in 2006. He received his PhD degree summa cum laude in 2009 for his thesis on learning with structured data in computer vision, completed at the Max Planck Institute for Biological Cybernetics, Tübingen and the Technical University of Berlin. His research interest is at the intersection of computer vision and machine learning. He regularly serves as PC-member and reviewer for machine learning (NIPS, ICML, AISTATS, UAI, ECML, JMLR) and computer vision (CVPR, ICCV, ECCV, PAMI, IJCV) conferences and journals.

Pushmeet Kohli is a senior research scientist in the Machine Learning and Perception group at Microsoft Research Cambridge, and is a part of the Association for Computing Machinery's (ACM) Distinguished Speaker Program. His research has appeared in conferences and journals in Computer Vision, Machine Learning, Robotics, AI, Computer Graphics, and HCI conferences. He has won best paper awards in ICVGIP 2006, 2010, ECCV 2010 and ISMAR 2011. His PhD thesis, titled "Minimizing Dynamic and Higher Order Energy Functions using Graph Cuts", was the winner of the British Machine Vision Association's "Sullivan Doctoral Thesis Award", and was a runner-up for the British Computer Society's "Distinguished Dissertation Award". Dr. Kohli's research has also been featured in popular media outlets such as Forbes, The Economic Times, New Scientist and MIT Technology Review.