Learning Full Pairwise Affinities for Spectral Segmentation

Tae Hoon Kim, *Member*, *IEEE*, Kyoung Mu Lee, *Member*, *IEEE*, and Sang Uk Lee, *Life Fellow*, *IEEE*

Abstract—Segmenting a single image into multiple coherent groups remains a challenging task in the field of computer vision. Particularly, spectral segmentation which uses the global information embedded in the spectrum of a given image's affinity matrix is a major trend in image segmentation. This paper focuses on the problem of efficiently learning a full range of pairwise affinities gained by integrating local grouping cues for spectral segmentation. We first construct a sparse multilayer graph whose nodes are both the pixels and the oversegmented regions obtained by an unsupervised segmentation algorithm. By applying the semi-supervised learning strategy to this graph, the intra and interlayer affinities between all pairs of nodes can be estimated without iteration. These pairwise affinities are then applied into the spectral segmentation algorithms. In this paper, two types of spectral segmentation algorithms are introduced: K-way segmentation and hierarchical segmentation. Our algorithms provide high-quality segmentations which preserve object details by directly incorporating the full-range connections. Moreover, since our full affinity matrix is defined by the inverse of a sparse matrix, its eigendecomposition can be efficiently computed. The experimental results on the BSDS and MSRC image databases demonstrate the superiority of our segmentation algorithms in terms of relevance and accuracy compared with existing popular methods.

Index Terms—Spectral segmentation, hierarchical segmentation, affinity estimation, semi-supervised learning

1 Introduction

Unsupervised image segmentation, clustering of pixels into meaningful image regions without any prior knowledge, is a fundamental but challenging problem in computer vision. The main research directions for this include mode-seeking [1], [2], deterministic annealing [3], stochastic clustering [4], [5], mixture model [6] [7], rate distortion [8], graph-based model [9], [10], [11], contour-based model [12], [13], and other variational methods [14], [15]. In most researches, the image segmentation problem is described as assigning a label to every pixel in a specific globalization framework. However, the appearance-based local methods such as mean shift (MShift) [1] are still popular for obtaining oversegmented regions with detailed boundaries. Thus, combining the advantages of local and global approaches is beneficial in image segmentation [16], [17].

In recent years, spectral segmentation has become a major trend in image segmentation. It typically starts from local information encoded in a graph-based representation of a given image, and partitions that image according to the global information embedded in the spectrum of the graph affinity matrix. Most methods [9], [18] take K eigenvectors of the affinity matrix, and then invoke another technique, such as K-means, to cluster pixels by their respective K components in the eigenvectors. However, these methods often break uniform regions where the eigenvectors have smooth gradients. To circumvent this difficulty, other methods [13] construct a hierarchy of regions that is consistent with the contour signals in the eigenvectors themselves.

1.1 Motivation

In natural images, many problems make segmentation difficult, such as determining faint object boundaries and separating highly cluttered backgrounds. As a popular way to solve these problems, several recent approaches [19], [18] have attempted to integrate local grouping cues across long-range connections in image space. Although their segmentation results are usually impressive, the direct use of long-range connections entails great computational costs. Therefore, the use of approximate techniques to propagate local grouping cues into larger image areas has been proposed.

Sampling-based model: In [20], the Nyström method to find numerical approximations to large-scale problems allows one to extrapolate the complete solution using a small random subset of pixels. Although this sampling-based model substantially reduces the computational costs, its accuracy is affected by the random sample selection.

Region-based model: In [21], [22], [8], the segmentation results are obtained by grouping nonoverlapping regions, instead of pixels. This region-based model is inspired by the hard constraint that the pixels in a particular region should have the same label. It has the advantage of using more informative features extracted from the inner pixels of a region, as well as of transferring local grouping cues to a larger image area with links across regions. However, since it enforces region constraints strongly, the segmentation results highly depend on the initial region boundaries.

• *The authors are with the Department of Electrical and Computer Engineering, Automation and Systems Research Institute (ASRI), Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 151-744, Korea. E-mail: {th33, kyoungmu}@snu.ac.kr, sanguk@ipl.snu.ac.kr.*

Manuscript received 29 Mar. 2012; revised 20 Sept. 2012; accepted 16 Oct. 2012; published online 26 Oct. 2012.
Recommended for acceptance by D. Forsyth.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2012-03-0227.
Digital Object Identifier no. 10.1109/TPAMI.2012.237.

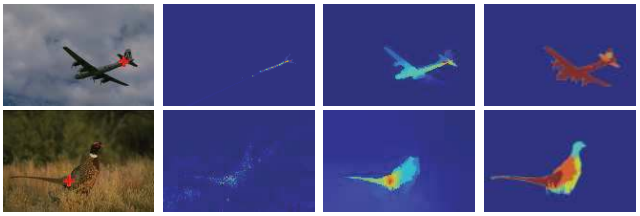

Fig. 1. Learning pairwise affinities. (a) Image with a user-given pixel point +. (b)-(d) Affinity images between + and other pixels in the image, where red represents a greater affinity. These affinities are obtained by the general color+boundary affinity model used in MNCut [18], our proposed model, and the ground-truth affinity function from human annotations, respectively.

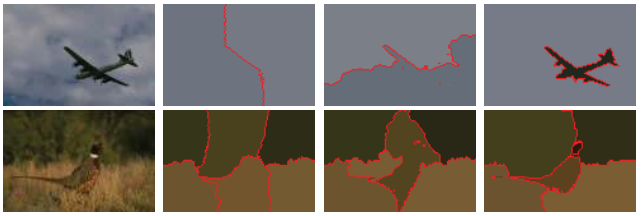

Fig. 2. Introducing K-way segmentation. (a) Original image. (b)-(d) K-way partitioning by NCut [9], MNCut [18], and our algorithm FNCut with the boundaries drawn in red, respectively.

Multiscale model: In [23], [24], [18], the segmentation methods are based on multiscale model to combine both coarse- and fine-level image details. Although this model gives efficient approximation for the incorporation of long-range connections with low complexity, due to inherent coarsening error fine-level details along object boundaries are not well recovered.

Multilayer model: In [16], [17], the multilayer graph was designed. In each layer, the nodes consist of pixels or over-segmented regions by MShift with different parameters. By defining the intra and interlayer affinities between graph nodes, local grouping cues can be propagated over a large area. However, since using long-range affinities leads to excessive complexity, the range of affinities is generally limited.

Unlike the above-mentioned approximation models, we try to design and use a full range affinity model in the spectral segmentation framework so that we can obtain high-quality segmentation results efficiently by using the proposed affinity model.

1.2 Contributions

The key contributions of our algorithm are as follows:

1. A new affinity model for spectral segmentation is introduced. Unlike previous models learned from a large labeled dataset [19], or from only local properties of adjacent pixels [26], [18], using the relevance scores between all pairs of pixels, estimated by semi-supervised learning (SSL) [27], [28], [29], as affinities is proposed. A multilayer graph whose nodes consist of the oversegmented regions by MShift as well as the pixels is first constructed. Then, the affinities between each node and other nodes are estimated by applying the SSL strategy to this graph through assuming the current node as labeled data and the others as unlabeled data. Fig. 1 clearly shows that our model intuitively provides much better affinities in highly textured images compared with the general affinity model [18] that heuristically unifies the color and boundary cues.
2. Spectral analysis of our full affinity matrix is done efficiently. In general, performing spectral analysis of a full large matrix requires a prohibitively expensive computation. However, since our full affinity matrix is expressed as the inverse of a sparse matrix, its eigendecomposition is very efficient using the basics of matrix computation [30].
3. We introduce two types of spectral segmentation algorithms based on our affinity matrix: K-way segmentation [31], [9], [32] and hierarchical segmentation [33]. For K-way segmentation, our algorithm FNCut clusters all pixels and regions simultaneously into the K visually coherent groups in a single multilayer framework of Normalized Cuts. For hierarchical segmentation, our algorithm fPb-OWT-UCM produces a hierarchy of regions from the contour signals in the eigenvectors using a sequence of two transformations: Oriented Watershed Transform (OWT) [34] and Ultrametric Contour Map (UCM) [35].
4. Our spectral segmentation algorithms, FNCut and fPb-OWT-UCM, produce high-quality segmentation results by considering all intra and interlayer affinities. Fig. 2 shows that FNCut produces much better K-way segmentations with object details than other spectral segmentation methods such as Normalized Cuts (NCut) [9] and Multiscale NCut (MNCut) [18]. Fig. 3 shows a comparison of fPb-OWT-UCM with other existing hierarchical segmentation methods based on the sequence of operations OWT and UCM: Canny-OWT-UCM and gPb-OWT-UCM [33] that use the Canny edge detector (Canny) [36] and the global boundary detector (gPb) [25] for contour detection, respectively. The results by fPb-OWT-UCM contain fewer superfluous edges in highly textured regions than those by Canny-OWT-UCM and more detailed object boundaries than those by gPb-OWT-UCM.

1.3 Overview of This Paper

The rest of the paper is organized as follows: We first review the previous affinity models in Section 2. Our affinity model is then discussed in Section 3. The spectral

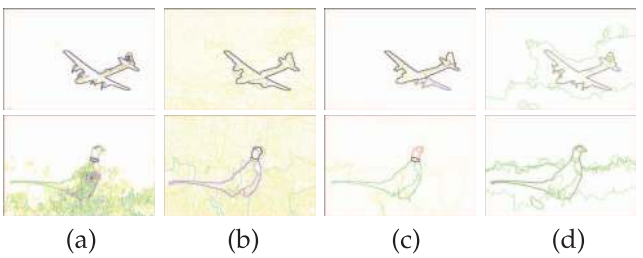

Fig. 3. Introducing hierarchical segmentation. (a)-(c) Hierarchical segmentations of an input image in Fig. 2a by Canny-OWT-UCM, gPb-OWT-UCM [25], and our algorithm fPb-OWT-UCM, respectively. (d) Human boundaries. Darker color means greater edgeness.

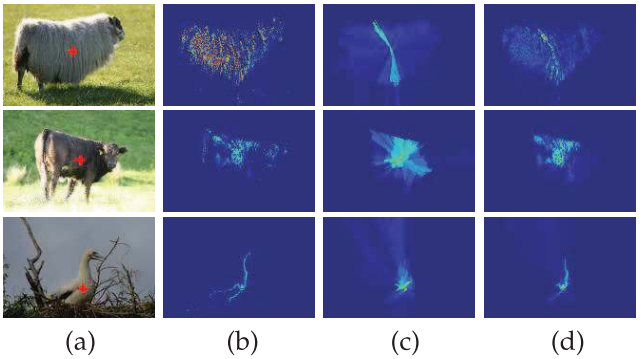

Fig. 4. Examples of affinity images by the previous affinity models. (a) Image with a user-given pixel (+). (b)-(d) Affinities between + and all other pixels by (1)-(3), respectively.

segmentation algorithms based on our full affinities are presented in Section 4. The experiments are given in Section 5, and finally, discussions are made in Section 6.

2 Previous Affinity Models

In the spectral segmentation algorithms, defining the affinity model gained by integrating local grouping cues such as color and boundary is important. Now, we review several previous affinity models, except the learned ones [19] from a large training dataset with the manually segmented images. The affinity w_{ij} between two pixels i and j is modeled according to the grouping cues used as follows:

1. **Color-based model:** Close-by pixels with similar colors likely belong to the same segment. The color-based affinity model φ_{c} is usually formulated by [9] [37]

$$w_{ij} = \varphi_{\mathrm{c}}(i,j) = exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\rho_x} - \frac{\|\mathbf{g}_i - \mathbf{g}_j\|^2}{\rho_g}\right), \tag{1}$$

where $\mathbf{x}_i$ and $\mathbf{g}_i$ denote the position and color values of pixel i, respectively. Fig. 4b shows that connecting pixels by color is useful when linking disjointed object parts. However, this process results in errors if the background has a similar color distribution as the object parts.

2. **Boundary-based model:** Edgeness is an important cue to detect a potential object boundary. The boundary-based affinity model φ_{b} is commonly formulated by measuring the edge magnitude between two pixels [26]:

$$w_{ij} = \varphi_{\mathrm{b}}(i,j) = exp\left(-\frac{\max_{i' \in \overline{ij}} \|\kappa_{i'}\|^2}{\rho_\kappa}\right), \tag{2}$$

where $\overline{ij}$ is a straight line joining two pixels i and j, and κ_i is the edge strength of pixel i. This boundary-based model is particularly useful when the background clutter has a similar color as the object body, such as in Fig. 4c. However, since it is based on the edgeness along the straight line between two pixels without considering all possible paths in image space, texture edges often disturb the affinity estimation.

3. **Combined model:** To design a better affinity model for all natural images, the combination of the color and boundary cues is helpful. They can be simply combined with a parameter α for the combined affinity model φ_{m} as follows [18]:

$$w_{ij} = \varphi_{\mathrm{m}}(i,j) = \sqrt{\varphi_{\mathrm{c}}(i,j) \times \varphi_{\mathrm{b}}(i,j)} + \alpha \cdot \varphi_{\mathrm{b}}(i,j), \tag{3}$$

where there are two models φ_{c} and φ_{b} in (1) and (2), respectively. Fig. 4d shows some affinity images gained through this combined model. However, the model φ_{m} still has some weaknesses in long-range affinity estimation since it is formulated by naively mixing two color and boundary affinity models.

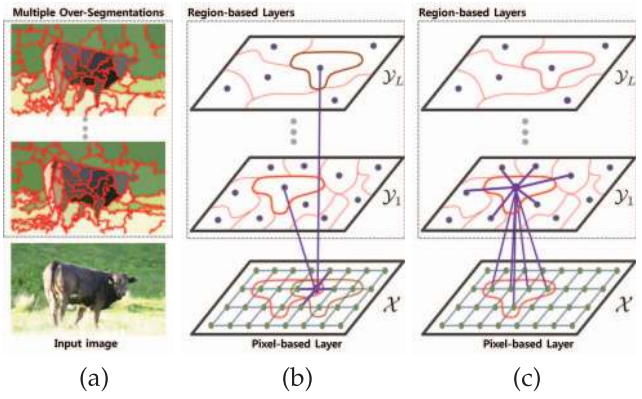

Fig. 5. Our multilayer graph $G = (\mathcal{V}, \mathcal{E})$. In (a), the graph nodes $\mathcal{V}$ consist of pixels $\mathcal{X}$ and regions $\mathcal{Y} = \{\mathcal{Y}_1 \cup \cdots \cup \mathcal{Y}_L\}$. The undirected edges $\mathcal{E}$ represent the relationship between pairs of nodes. (b) and (c) show the examples of edges (violet lines) connected to one pixel and to one region, respectively.

3 Proposed Full Affinity Model

To estimate the full pairwise affinities by integrating local grouping cues extracted from the entire image, we propose to use the SSL method [27], [28], [29], which is a very successful technique to learn the global relevance scores between labeled and unlabeled data in a sparse graph. We first design a sparsely connected graph with multilayers for efficiently combining local grouping cues. The relevance scores defined by Zhou et al. [28] between all pairs of the graph nodes are then used as the full affinities.

3.1 Graph Design

Let us construct a multilayer graph $G = (\mathcal{V}, \mathcal{E})$, where the nodes $\mathcal{V} = \{\mathcal{X} \cup \mathcal{Y}\}$ consist of a set of pixels $\mathcal{X}$ and the sets of regions $\mathcal{Y} = \{\mathcal{Y}_1 \cup \cdots \cup \mathcal{Y}_L\}$, and the edges $\mathcal{E}$ are the undirected links, as shown in Fig. 5. The node subset $\mathcal{Y}_l$ contains the N_l nonoverlapping regions, generated by an unsupervised segmentation algorithm such as MShift [1], at the lth region-based layer. In this work, the L different oversegmentations are used since it is known that the use of multiple overlapped regions can reduce the errors of those regions that may contain many objects [16]. The edges $\mathcal{E}$ are linked by different criteria according to the node types. An

undirected edge $e_{ij} \in \mathcal{E}$ exists if one of the following conditions is satisfied:

1. If two pixels $i, j \in \mathcal{X}$ are adjacent, usually in the 4 or 8 neighborhood system, an intralayer edge e_{ij} between i and j exists with the following weight ω_{ij}:

$$\omega_{ij} = exp\left(-\frac{\|\mathbf{g}_i - \mathbf{g}_j\|}{\rho_g}\right), \tag{4}$$

where the constant ρ_g that controls the strength of the weight is defined as the variance of the total color values.

2. If a region i and its adjacent region j share a common boundary at the same region-based layer ($i, j \in \mathcal{Y}_l$), another intralayer edge e_{ij} is linked with the following weight ω_{ij}:

$$\omega_{ij} = exp\left(-\frac{\|\bar{\mathbf{g}}_i - \bar{\mathbf{g}}_j\|}{\rho_g}\right), \tag{5}$$

where $\bar{\mathbf{g}}_i$ denotes the mean color values of the inner pixels of region i.

3. If a pixel $i \in \mathcal{X}$ is included in its corresponding region $j \in \mathcal{Y}_l$, an interlayer edge e_{ij} is added with the following weight ω_{ij}:

$$\omega_{ij} = \tau, \tag{6}$$

where τ is a parameter that controls a positive correlation between pixel- and region-based layers.

The intralayer weights in (4) and (5) encode the color cues in *Lab* color space, similarly as in (1). Since the correlation between the pixels in the same region is more emphasized through the interlayer connections between the region and its inner pixels in (6), the region itself provides implicit boundary information as the boundary cue instead of image edges such as in (2). Therefore, the tradeoff between our color and boundary cues is captured by a parameter τ.

3.2 Learning Full Affinities

Let $\mathbf{\Pi} = [\pi_{ij}]_{N\times N}$, where $N = |\mathcal{V}|$ is the number of nodes, denote the affinity matrix. To learn the full affinities $\mathbf{\Pi}$, we borrow ideas from SSL. Each node m and all other nodes $\mathcal{V} - \{m\}$ are first assigned as labeled and unlabeled nodes, respectively, and then the relevance scores between labeled and unlabeled nodes are determined by applying the SSL strategy to our graph G. We propose to use these scores as the full affinities $\vec{\pi}_m = [\pi_{im}]_{N\times 1}$ between that node m and all nodes $\mathcal{V}$. In this paper, two popular SSL functions ψ_{no} and ψ_{un} are discussed as found in the work of Zhou et al. [28]. They can make the symmetric affinity matrix $\mathbf{\Pi}$ whose elements are sufficiently smooth with respect to the intrinsic structure of the graph.

1. ψ_{no}: First, the affinity vector $\vec{\pi}_m$ of all nodes $\mathcal{V}$ from a node m can be formulated:

$$\vec{\pi}_m = [\psi_{\text{no}}(\cdot, m)]_{N\times 1} = c(\mathbf{I} - (1-c)\mathbf{P})^{-1}\vec{b}_m, \tag{7}$$

where $\mathbf{I}$ is the identity matrix of size N, and $\mathbf{P}$ is the bi-normalized adjacency matrix $\mathbf{\Omega} = [\omega_{ij}]_{N\times N}$ in (4)-(6): $\mathbf{P} = \mathbf{D}^{-\frac{1}{2}}\mathbf{\Omega}\mathbf{D}^{-\frac{1}{2}}$, where the degree matrix $\mathbf{D} = \text{diag}([d_1, \ldots, d_N])$ is diagonal with $d_i = \sum_{j=1}^{N}\omega_{ij}$. The vector $\vec{b}_m = [b_{im}]_{N\times 1}$ indicating labeled node m is N-dimensional with $b_{im} = 1$ if labeled node $i = m$ and 0 otherwise. The vector $\vec{\pi}_m$ in (7) can be interpreted as the solution to minimize the following cost function $E_{\text{no}}(m)$:

$$E_{\text{no}}(m) = \sum_{i,j=1}^{N}\omega_{ij}\left|\frac{\pi_{im}}{\sqrt{d_i}} - \frac{\pi_{jm}}{\sqrt{d_j}}\right|^2 + \mu\sum_{i=1}^{N}|\pi_{im} - b_{im}|^2, \tag{8}$$

where $\mu = \frac{c}{1-c}$ $(0 < c < 1)$. The first term in $E_{\text{no}}(m)$ is the smoothness constraint in which good affinities should not change too much between neighboring nodes. However, this term is defined as local variation by the difference in normalized affinities between two end nodes of each edge. Thus, the smooth affinity variation between close-by nodes cannot be guaranteed, as shown in Figs. 6e, 6f, 6g, and 6h. The second term in $E_{\text{no}}(m)$ is the fitting constraint in which good affinities should not change too much from the initial label assignment. A positive parameter μ (or c) specifies the relative amount of smoothness and fitting constraints.

2. ψ_{un}: Second, the affinity vector $\vec{\pi}_m$ of all nodes $\mathcal{V}$ from a node m is defined as

$$\vec{\pi}_m = [\psi_{\text{un}}(\cdot, m)]_{N\times 1} = c(\mathbf{D} - (1-c)\mathbf{\Omega})^{-1}\vec{b}_m. \tag{9}$$

The vector $\vec{\pi}_m$ in (9) is the same as the solution to minimize the following cost function $E_{\text{un}}(m)$, similar to $E_{\text{no}}(m)$ in (8):

$$E_{\text{un}}(m) = \sum_{i,j=1}^{N}\omega_{ij}|\pi_{im} - \pi_{jm}|^2 + \mu\sum_{i=1}^{N}d_i\left|\pi_{im} - \frac{b_{im}}{d_i}\right|^2. \tag{10}$$

The function $E_{\text{un}}(m)$ also consists of smoothness and fitting constraints. However, compared with $E_{\text{no}}(m)$ in (8), the smoothness constraint in $E_{\text{un}}(m)$ is more helpful in estimating the pairwise affinities with local smooth variation, as shown in Figs. 6i, 6j, 6k, and 6l. Moreover, according to the increase in the number of over-segmentations, the intrinsic image structure with object details is well represented in the affinity image by solving the affinity function in (9).

Although the function ψ_{no} in (7) is known to be better than ψ_{un} in (9) for classification problems [28], ψ_{un} is perceptually more suited for the affinity estimation. The quantitative results of these two functions will be compared in the experiments. Therefore, in this work the affinity matrix $\mathbf{\Pi} = [\pi_{ij}]_{N\times N}$ is defined by using $\vec{\pi}_m$ in (9) as follows:

$$\begin{aligned}\mathbf{\Pi} &= [\vec{\pi}_1, \ldots, \vec{\pi}_N] = [\psi_{\text{un}}(i,j)]_{N\times N} \\ &= c(\mathbf{D} - (1-c)\mathbf{\Omega})^{-1} = c\mathbf{Z}^{-1},\end{aligned} \tag{11}$$

where $\mathbf{Z} = \mathbf{D} - (1-c)\mathbf{\Omega}$ is positive definite. Since the matrix $\mathbf{Z}$ is sparse but very large, directly inversing it without any approximation technique is generally difficult. Fortunately, however, our full affinity matrix $\mathbf{\Pi}$ in (11) can be efficiently used in a spectral framework.

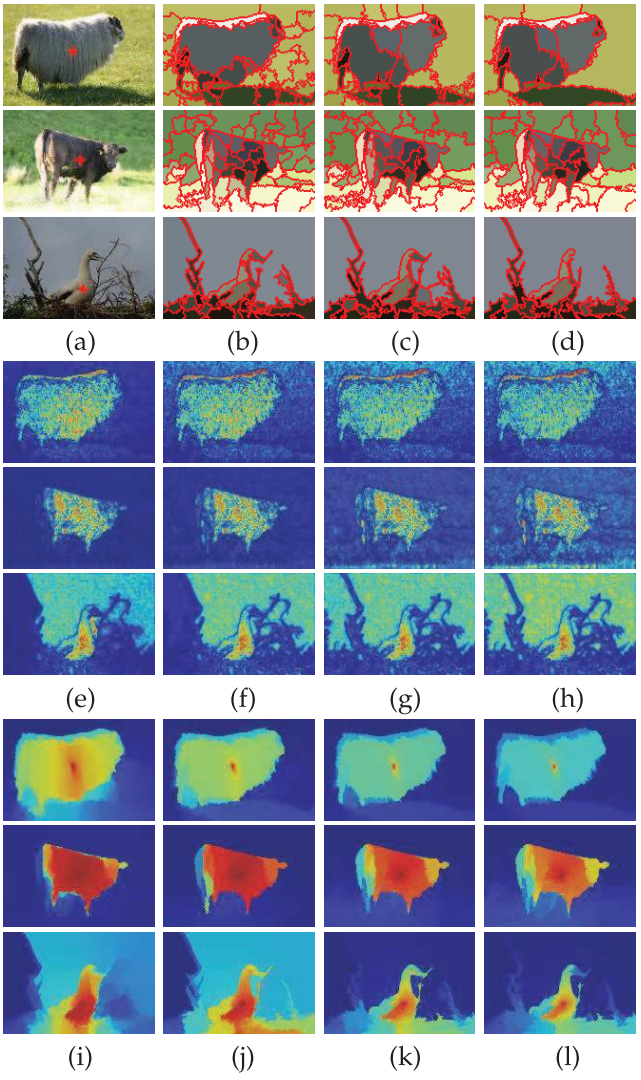

Fig. 6. Examples of our affinity images using SSL [28]. (a) Image with one pixel + selected. (b)-(d) Three oversegmentations, obtained by varying the parameters of the MShift algorithm [1]. (e)-(h) and (i)-(l) Affinity images $\vec{\pi}_+$ by ψ_{no} in (7) and by ψ_{un} in (9) with none, one (b), two (b)-(c), and three (b)-(d) oversegmentations, respectively.

3.3 Spectral Analysis

Let $\mathbf{D}_{\mathbf{\Pi}} = \text{diag}([d_1^{\Pi}, \ldots, d_N^{\Pi}])$, whose diagonal element is $d_i^{\Pi} = \sum_{j=1}^{N} \pi_{ij}$, be the degree matrix of our affinity matrix $\mathbf{\Pi}$. The following generalized eigenvalue problem based on the affinity matrix $\mathbf{\Pi}$ is generally solved:

$$(\mathbf{D}_{\mathbf{\Pi}} - \mathbf{\Pi})\vec{v}_k = \lambda_k \mathbf{D}_{\mathbf{\Pi}} \vec{v}_k. \tag{12}$$

Here, the generalized eigenvectors $\{\vec{v}_k\}_{k=1,\ldots,K}$, corresponding to the K smallest eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_K$, can be utilized for K-way segmentation [9] and hierarchical segmentation [33]. In general, the eigendecomposition of the large-size full matrix requires a great computation, which is undesirable. However, since the proposed full affinity matrix $\mathbf{\Pi}$ is expressed as the inverse of a sparse matrix $\mathbf{Z}$ in (11), efficiently solving the system in (12) is possible. We first transform this system into a standard eigensystem as follows:

$$\mathbf{D}_{\mathbf{\Pi}}^{-\frac{1}{2}} \mathbf{\Pi} \mathbf{D}_{\mathbf{\Pi}}^{-\frac{1}{2}} \vec{q}_k = (1 - \lambda_k)\vec{q}_k, \tag{13}$$

where $\vec{q}_k = \mathbf{D}_{\mathbf{\Pi}}^{\frac{1}{2}} \vec{v}_k$ is an eigenvector of $\mathbf{D}_{\mathbf{\Pi}}^{-\frac{1}{2}} \mathbf{\Pi} \mathbf{D}_{\mathbf{\Pi}}^{-\frac{1}{2}}$. By substituting $\mathbf{\Pi}$ in (11) and using a simple spectral theory [30], we then rewrite (13) as the following system based on the sparse matrix $\mathbf{Z}$:

$$\mathbf{D}_{\mathbf{\Pi}}^{\frac{1}{2}} \mathbf{Z} \mathbf{D}_{\mathbf{\Pi}}^{\frac{1}{2}} \vec{q}_k = \varepsilon_k \vec{q}_k, \tag{14}$$

where $\varepsilon_k = \frac{c}{1-\lambda_k}$ is an eigenvalue of $\mathbf{D}_{\mathbf{\Pi}}^{\frac{1}{2}} \mathbf{Z} \mathbf{D}_{\mathbf{\Pi}}^{\frac{1}{2}}$, and the diagonal elements of $\mathbf{D}_{\mathbf{\Pi}}$ are computed by

$$\left[d_1^{\Pi}, \ldots, d_N^{\Pi}\right]^T = \mathbf{\Pi}\, \vec{1}_{N\times 1} = c\mathbf{Z}^{-1}\, \vec{1}_{N\times 1}, \tag{15}$$

where the multiplication of $\mathbf{Z}$'s inversion by the all-ones vector $\vec{1}_{N\times 1}$ can be efficiently done using the linear system solver implemented by the MATLAB division operator "\". Thus, the K smallest eigenvalues $\{\varepsilon_k\}_{k=1,\ldots,K}$ and their corresponding eigenvectors $\{\vec{q}_k\}_{k=1,\ldots,K}$ of the very sparse matrix $\mathbf{D}_{\mathbf{\Pi}}^{\frac{1}{2}} \mathbf{Z} \mathbf{D}_{\mathbf{\Pi}}^{\frac{1}{2}}$ in (14) are determined instead of the eigendecomposition of the very dense and large matrix $\mathbf{D}_{\mathbf{\Pi}}^{-\frac{1}{2}} \mathbf{\Pi} \mathbf{D}_{\mathbf{\Pi}}^{-\frac{1}{2}}$ in (13). Finally, we find the generalized eigenvalue $\{\lambda_k\}_{k=1,\ldots,K}$ and their corresponding eigenvectors $\{\vec{v}_k\}_{k=1,\ldots,K}$ in (12) using $\lambda_k = 1 - \frac{c}{\varepsilon_k}$ and $\vec{v}_k = \mathbf{D}_{\mathbf{\Pi}}^{-\frac{1}{2}} \vec{q}_k$, respectively. The process of spectral analysis on our full affinity matrix $\mathbf{\Pi}$ is briefly summarized in Algorithm 1.

Algorithm 1. Proposed Spectral Analysis

1: Given an image, construct a multilayer graph G, and calculate its weight matrix $\mathbf{\Omega}$ using (4)-(6).
2: Find the K smallest eigenvalues $\{\varepsilon_k\}_{k=1,\ldots,K}$ and their corresponding eigenvectors $\{\vec{q}_k\}_{k=1,\ldots,K}$ in (14) using our affinity matrix $\mathbf{\Pi}$ in (11) and its degree matrix $\mathbf{D}_{\mathbf{\Pi}}$ in (15).
3: Estimate the generalized eigenvalues $\{\lambda_k\}_{k=1,\ldots,K}$ and eigenvectors $\{\vec{v}_k\}_{k=1,\ldots,K}$ in (12) using $\lambda_k = 1 - \frac{c}{\varepsilon_k}$ and $\vec{v}_k = \mathbf{D}_{\mathbf{\Pi}}^{-\frac{1}{2}} \vec{q}_k$, respectively.

Fig. 7 presents the average running time comparison with respect to the number of pixels and some examples of the generalized eigenvectors, produced by solving (12). Spectral analysis of our full matrix $\mathbf{\Pi}$, which additionally computes the degree matrix $\mathbf{D}_{\mathbf{\Pi}}$ in (15), takes a little more time than spectral analysis obtained by defining the sparse weight matrix $\mathbf{\Omega}$ as affinity matrix, as shown in Fig. 7b. However, the eigenvectors based on our full affinity matrix $\mathbf{\Pi}$ in Figs. 7e and 7f vary in large uniform areas less than those based on the very sparse matrix $\mathbf{\Omega}$ in Figs. 7c and 7d. They also contain more detailed object boundaries such as the tree's branches in the bottom images of Figs. 7e and 7f.

4 Spectral Segmentation

This paper presents two types of spectral segmentation algorithms based on our full affinity matrix:

1. **K-way segmentation**: We borrow the basic idea of Normalized Cuts [9], which associates with each pixel a length K descriptor formed from entries of the K eigenvectors and uses a clustering algorithm such as K-means to create a hard partition of the image.
2. **Hierarchical segmentation**: We transform the contour signals, produced by combining contour information in different eigenvectors, into a hierarchy regions using OWT and UCM, similarly as in [33].

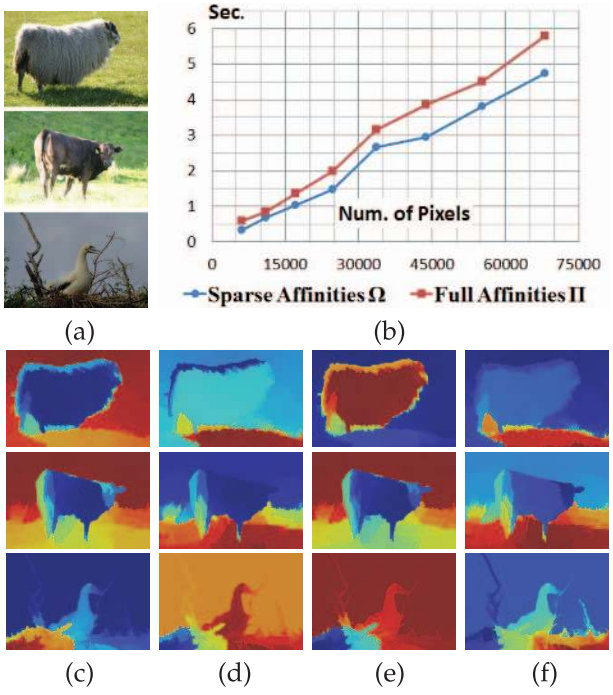

Fig. 7. Comparison of spectral analysis with respect to the sparse matrix $\mathbf{\Omega}$ and our full affinity matrix $\mathbf{\Pi}$. (a) Test images. (b) The average running times to solve the generalized eigenvalue problem in (12) by MATLAB 7.13 on a quad-core 3.3 GHz desktop as a function of the number of pixels ($K = 5$). (c)-(d) and (e)-(f) The second and third smallest eigenvectors by solving the standard generalized eigenvalue problem $(\mathbf{D} - \mathbf{\Omega})\vec{v}_k = \lambda_k \mathbf{D}\vec{v}_k$ and the proposed problem in (12) based on the affinity matrices $\mathbf{\Omega}$ and $\mathbf{\Pi}$, respectively.

4.1 K-Way Segmentation

Let us consider the image segmentation as a labeling problem in which one label $k \in \{1, \ldots, K\}$ is assigned to each pixel i. Let $\vec{z}_k = [z_{ik}]_{N\times 1}$ denote a partitioning vector with $z_{ik} = 1$ if i belongs to the kth segment and 0 otherwise. Here, the segmentation criterion of our algorithm FNCut is same to that of NCut [9]:

$$\text{minimize } \mathcal{C}(\mathbf{Z}) = \frac{1}{K}\sum_{k=1}^{K} \frac{\vec{z}_k^T \mathbf{\Pi} \vec{z}_k}{\vec{z}_k^T \mathbf{D}_{\mathbf{\Pi}} \vec{z}_k}, \tag{16}$$

subject to $\mathbf{\Pi}$ in (11) and $\mathbf{Z}\mathbf{Z}^T = \mathbf{I}$, where the partitioning matrix $\mathbf{Z} = [\vec{z}_1, \ldots, \vec{z}_K]$. If $\vec{z}_k$ is relaxed to take on real values, we can minimize (16) by solving the generalized eignevalue system in (12). Namely, the optimal solution of $\mathcal{C}(\mathbf{Z})$ in (16) is the subspace spanned by the K smallest generalized eigenvectors $\mathbf{V}_k = [\vec{v}_1, \ldots, \vec{v}_K]$ in (12). As this subspace is discretized by K-means clustering, our K-way segmentation is obtained.

Fig. 8 shows an example which illustrates our K-way segmentation. The four smallest eigenvectors in (12) are shown in Figs. 8b, 8c, 8d, and 8e. The 2-4 way partitions in Figs. 8f, 8g, and 8h are obtained using the 2-4 different eigenvectors, respectively. Fig. 9 presents one example of our multilayer segmentation. The three oversegmentations in Figs. 9b, 9c, and 9d contain some incorrect regions since some parts of an airplane have a similar color distribution as the sky in Fig. 9a. Nevertheless, our multilayer approach produces an impressive final segmentation in Fig. 9e. Further, as depicted in Figs. 9e, 9f, 9g, and 9h the segmentation results at different layers are very similar to one another. This finding is attributed to the label continuity between two nodes at different layers, which is enforced by our segmentation criterion in (16) based on the full affinity matrix between all pixels and all regions.

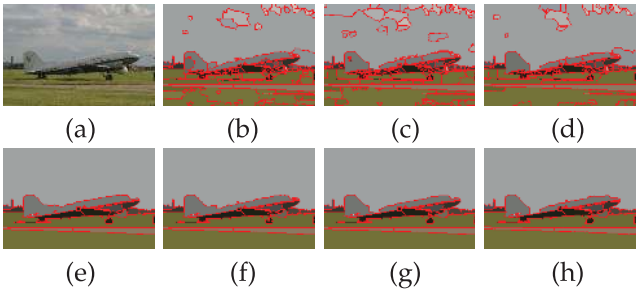

Fig. 9. Example of multilayer segmentation by FNCut. (a) Test image. (b)-(d) The three oversegmentations by varying the parameters of the MShift algorithm. (e)-(h) Our K-way segmentations at each layer (a)-(d), respectively ($K = 20$).

4.2 Hierarchical Segmentation

The generalized eigenvectors $\{\vec{v}_1, \ldots, \vec{v}_K\}$ in (12) corresponding to the K smallest eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_K$ can be used for contour detection since the eigenvectors themselves carry contour information [25]. Treating each eigenvector $\vec{v}_k$ as an image, $\vec{v}_k$ is convolved with Gaussian directional derivative filters at multiple orientations $\{\theta\}$ for obtaining the oriented signals $\{\nabla_\theta \vec{v}_k\}$. By combining the contour signals in the different eigenvectors, the boundary detector fPb based on spectral analysis of our full affinity matrix $\mathbf{\Pi}$ is defined as

$$\text{fPb}(\theta) = \sum_{k=2}^{K} \frac{1}{\sqrt{\lambda_k}} \cdot \nabla_\theta \vec{v}_k, \tag{17}$$

where the weighting by the eigenvalue λ_k is motivated by the physical interpretation of the generalized eigenvalue problem as a mass-spring system [38].

From the contour signals obtained by fPb in (17), our algorithm fPb-OWT-UCM can produce a hierarchical segmentation result as follows: Similarly as in [33], we first generate the nonoverlapped regions based on these contour signals by using the OWT [34], and then construct a hierarchy of regions in the UCM [35]. This procedure can be seen as generic machinery for going from the contours to a

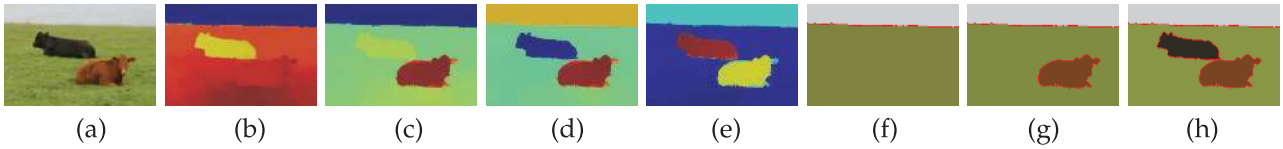

Fig. 8. Overview of our K-way segmentation algorithm FNCut. (a) Test image. (b)-(e) The four smallest eigenvectors $\{\vec{v}_1, \ldots, \vec{v}_4\}$ in (12). The first eigenvector in (b) is very close to a constant vector. (f)-(h) The K-way segmentation results ($K = 2$, 3, and 4).

Fig. 10. Overview of our hierarchical segmentation algorithm fPb-OWT-UCM. (a) Test image. (b) Maximal response of contour signals by fPb in (17). (c) Region tree in the UCM. (d)-(h) Segmentations produced by thresholding the UCM in (c) at level 0.01, 0.05, 0.1, 0.3, and 0.5, respectively.

hierarchical region tree. Fig. 10 shows an example which illustrates our hierarchical segmentation. The UCM result in Fig. 10c is expressed as a region tree, which preserves the quality of contours, extracted by fPb in Fig. 10b. Therefore, the segmentation result at any scale can be easily retrieved by thresholding the UCM at that scale, such as in Figs. 10d, 10e, 10f, 10g, and 10h.

5 Experiments

5.1 Parameter Setting

To construct our multilayer graph, we initially generate the regions by the unsupervised image segmentation methods such as graph-based image segmentation (GBIS) [10] and MShift [1], which attempt to partition image pixels into components such that the resulting segmentation is neither too coarse nor too fine. Fig. 11 shows our spectral segmentation results using two different oversegmentation methods: GBIS and MShift. The oversegmentations by MShift in Figs. 11g, 11h, and 11i better detect thin elongated parts such as the ski and ski pole, and contain fewer superfluous regions around the object boundaries than those by GBIS in Figs. 11b, 11c, and 11d. Therefore, we used MShift as the oversegmentation method in our algorithms.

The MShift method needs two bandwidth parameters (h_s, h_r) for the spatial and range domains, respectively. In this paper, three region-based layers with the different sets of two MShift parameters were obtained. Now, we consider the selection of parameters (h_s, h_r). Fig. 12 shows our segmentation results by changing this selection. If multiple region-based layers of quite different scales in Figs. 12c, 12d, and 12e are used, our segmentation results in Figs. 12i, 12j, 12k, and 12l mainly depend on the quality of the largest-scale region-based layer in Fig. 12e, namely, some objects whose boundaries exist in Fig. 12e can be well detected such as "dock" (top) and "car" (bottom) in Figs. 12i, 12j, 12k, and 12l. By contrast, if the boundaries of the object parts are missing in Fig. 12e, finding their exact boundaries is very difficult such as "boat" (top) and "tree" (bottom) in Figs. 12i, 12j, 12k, and 12l. To make up for this large-scale limitation, similar small-scale region-based layers are used, such as in Figs. 12f, 12g, and 12h. Since they give multiple object boundary candidates, the object details appear on the results in Figs. 12m, 12n, 12o, and 12p by increasing the number of segments K for K-way segmentation or by decreasing a threshold for hierarchical segmentation. In our experiments, we set $(h_s, h_r) = \{(5,7),(7,5),(7,7)\}$ empirically.

Note that our affinity model has two parameters: the interlayer edge weight τ in (6) and the balanced weight c in (11). Fig. 13 shows the variation in affinity images with respect to these two parameters. With a larger τ, the region consistency is more emphasized and the affinities are more discretized on the region boundaries, as shown in the top row of Fig. 13. With a smaller τ, the affinities are more oversmoothed around the user-given point. With a larger c, the fitting constraint is more emphasized and the only affinities within a smaller distance are presented as shown in the bottom row of Fig. 13. With a smaller c, long-range connections around the user-given point are more emphasized. The parameters $\tau, c \in \{10^{-1}, 10^{-3}, 10^{-5}\}$ were experimentally chosen and set $\tau = 10^{-3}$ and $c = 10^{-5}$ for all test images.

5.2 Measurements

For quantitative comparison, the following four measures are used:

1. **F-measure**: The traditional F-measure (F) is the harmonic mean of precision and recall as follows:

$$\mathrm{F} = 2 \cdot \frac{\mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}, \tag{18}$$

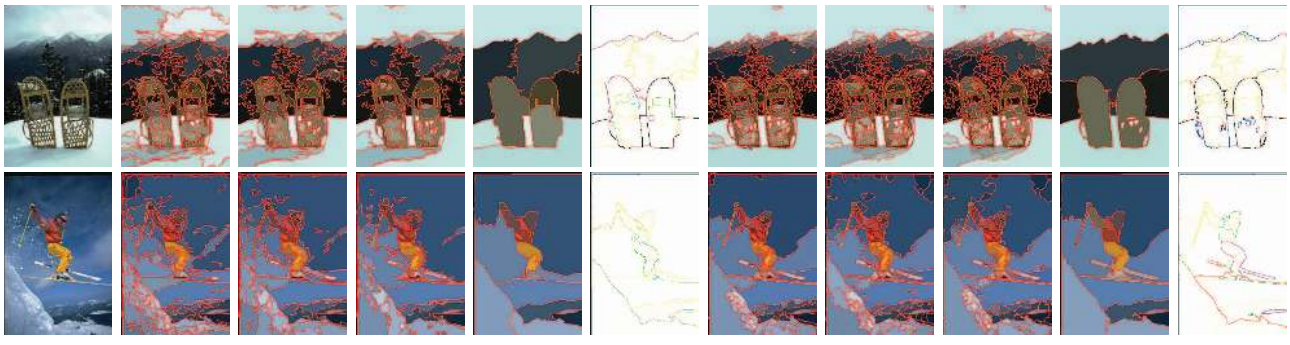

Fig. 11. Comparison of oversegmentations by GBIS [10] and MShift [1] for spectral segmentation. (a) Test images. (b)-(d) and (g)-(i) The three different oversegmentations by GBIS and MShift, respectively. (e) and (j) K-way segmentations using the regions in (b)-(d) and (g)-(i), respectively (top: $K = 10$, bottom: $K = 20$). (f) and (k) Hierarchical segmentations using the regions in (b)-(d) and (g)-(i), respectively.

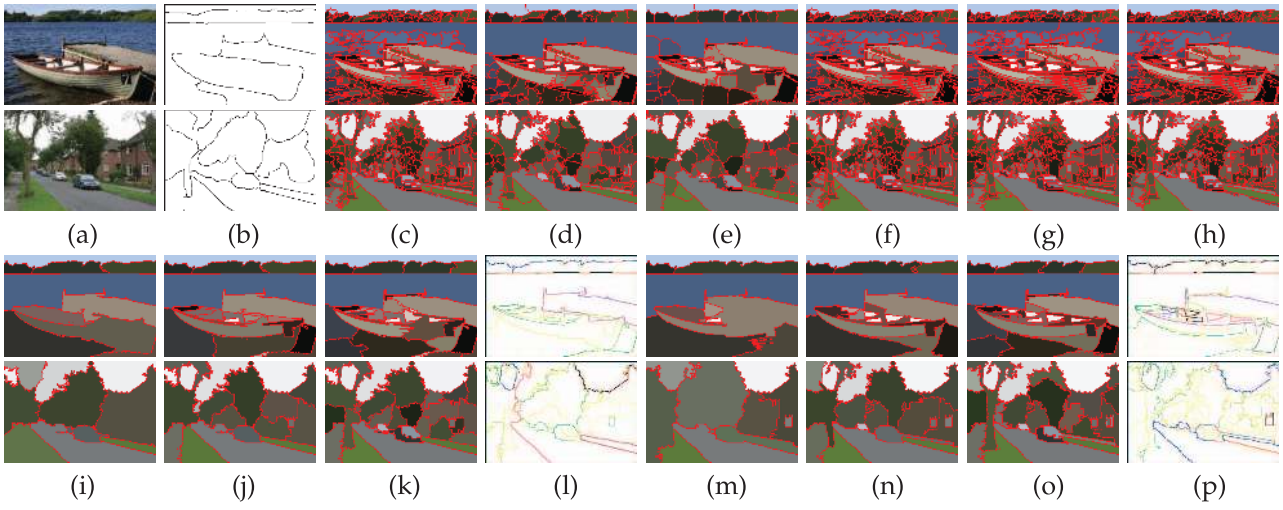

Fig. 12. Examples of our segmentation results using different region-based layers $\mathcal{Y}$ produced by MShift in the multilayer graph G. (a) Test images. (b) Ground-truth boundaries. (c)-(e) and (f)-(h) The three oversegmentations in accordance with the different sets of the MShift parameters $(h_s, h_r) = \{(5,7),(10,15),(15,25)\}$ and $(h_s, h_r) = \{(5,7),(7,5),(7,7)\}$, respectively. (i)-(k) and (m)-(o) 20/30/40-way segmentations using the region-based layers (c)-(e) and (f)-(h), respectively. (l) and (p) Hierarchical segmentations using the region-based layers (c)-(e) and (f)-(h), respectively.

where Precision is the number of correct results divided by the number of all returned results and Recall is the number of correct results divided by the number of results that should have been returned. The F score reaches its best value at 1 and worst score at 0.

2. **Segmentation covering**: The overlap between two regions R and R' is defined as

$$\mathcal{O}(R, R') = \frac{|R \cap R'|}{|R \cup R'|}. \tag{19}$$

This measure has been used for the evaluation of the pixel-wise classification task in recognition [39]. We define the covering (Covering) of a segmentation S by a ground-truth segmentation T as

$$\text{Covering}(T \rightarrow S) = \frac{1}{|\mathcal{X}|} \sum_{R \in S} |R| \cdot \max_{R' \in T} \mathcal{O}(R, R'), \tag{20}$$

where $\mathcal{X}$ is a set of pixels in the image. Similarly, the covering of a machine segmentation S by a family of ground-truth segmentations $\{T_i\}$ is defined by first covering S separately with each human segmentation T_i and then averaging over the different humans.

3. **Probabilistic Rand Index (PRI)**: The Rand Index (RI) [40] between test and ground-truth segmentation is calculated by the sum of the number of pairs of pixels that have the same label and those that have different labels in both segmentations, divided by the total number of pairs of pixels. The probabilistic rand index, a variant of RI, has been proposed for dealing with the case of multiple ground-truth segmentations [41]. The PRI amounts to averaging the RI among different ground-truth segmentations.

4. **Variation of Information (VI)**: The Variation of Information score was introduced for the purpose of clustering comparison [42]. The VI measures the distance between a segmentation S and a ground-truth segmentation T in terms of their average conditional entropy given by

$$\text{VI}(S;T) = H(S) + H(T) - 2I(S;T), \tag{21}$$

where $H(\cdot)$ and $I(\cdot)$ represent, respectively, the entropies and mutual information between two segmentations S and T. In the presence of several ground-truth segmentations $\{T_i\}$, we average the VI among different ground-truth segmentations.

The segmentation is viewed better if VI is smaller or the other three are larger.

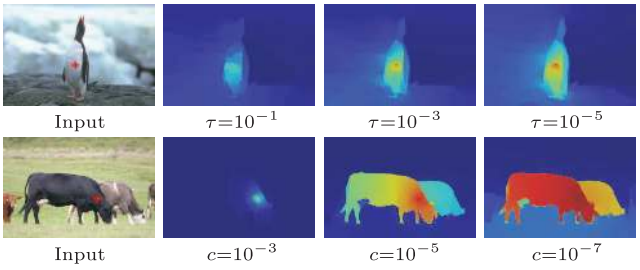

Fig. 13. Examples of our affinity images with respect to the variation of parameters τ (top row, $c = 10^{-5}$) and c (bottom row, $\tau = 10^{-3}$) in (6) and (11), respectively.

5.3 Results

Our spectral segmentation algorithms: FNCut for K-way segmentation and fPb-OWT-UCM for hierarchical segmentation, are quantitatively evaluated on two Berkeley image datasets: BSDS300[1] and BSDS500.[2] The BSDS300 (or BSDS500) dataset consists of 200 (or 300) training images and 100 (or 200) test images. The above-mentioned measures are performed on the test images. More visual segmentation results are then given on the MSRC object recognition database.[3]

In the experiments, a single segmentation that involves a choice of scale as output is produced. The optimal scale is chosen according to the following two conditions: optimal

1. http://www.cs.berkeley.edu/projects/vision/bsds.
2. http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/resources.html.
3. http://research.microsoft.com/en-us/projects/objectclassrecognition.

TABLE 1
Boundary Benchmarks on the BSDS300 and BSDS500 Datasets

	BSDS300			BSDS500		
	ODS	OIS	AP	ODS	OIS	AP
Human	0.79	0.79	-	0.80	0.80	-
[1] MShift	0.63	0.66	0.54	0.64	0.68	0.56
[18] MNCut	0.62	0.66	0.43	0.64	0.68	0.45
Canny-OWT-UCM	0.58	0.63	0.58	0.60	0.64	0.58
[10] GBIS	0.58	0.62	0.53	0.61	0.64	0.56
[43] SWA	0.56	0.59	0.54	-	-	-
[35] mPb-OWT-UCM	**0.68**	**0.72**	**0.70**	**0.70**	**0.74**	**0.71**
[33] gPb-OWT-UCM	**0.71**	**0.74**	**0.73**	**0.73**	**0.76**	**0.73**
FNCut	0.64	0.68	0.41	0.67	0.71	0.44
fPb-OWT-UCM	0.67	0.70	0.61	0.69	0.71	0.62
cPb-OWT-UCM	**0.70**	**0.72**	**0.72**	**0.72**	**0.75**	**0.73**

Results for 10 different segmentation algorithms are given. The F *scores when choosing an optimal scale for the entire dataset (ODS) or per image (OIS), as well as the average precision (AP). The best three results are highlighted in colors: red, green, and blue in descending order.*

dataset scale (ODS) and optimal image scale (OIS). The former is to use a fixed parameter for all images in the dataset. The latter is to select the optimal parameter by an oracle on a per-image basis. The scale parameter is the number of segments $K \in \{2, 3, \ldots, 40\}$ for K-way segmentation or a threshold in range of $[0, 1]$ for hierarchical segmentation.

5.3.1 Boundary Quality

In our experiments, we report three different boundary quantities for each algorithm: the best F on the dataset for a fixed scale (ODS), the aggregate F on the dataset for the best scale in each image (OIS), and the average precision (AP) on the full recall range. Table 1 shows these quantities for the BSDS300 and BSDS500 datasets. To provide a basis of comparison, the seven existing image segmentation algorithms are reported: one K-way spectral method (MNCut [18]), three region merging methods (MShift [1], GBIS [10], SWA [43]), and three contour-based methods (Canny-OWT-UCM, mPb-OWT-UCM [35], gPb-OWT-UCM [33]) which make the UCM from the contour signals produced by Canny, multiscale boundary detector (mPb) [44], and gPb [25], respectively. Fig. 14 displays the full precision-recall curves on two BSDS datasets.

In Table 1, our algorithms FNCut and fPb-OWT-UCM better detect the object boundaries than other nonlearning methods: MNCut, MShift, GBIS, SWA, and Canny-OWT-UCM. Particularly, fPb-OWT-UCM has better performance than FNCut since the K-way segmentation algorithms such as FNCut often make large uniform regions in which the eigenvectors vary smoothly broken up. However, the segmentation results by fPb-OWT-UCM have poorer boundary quality than those by mPb-OWT-UCM and gPb-OWT-UCM, which are learned using the training images and their corresponding ground-truth segmentations. To improve our performance, we additionally propose a learning-based method cPb-OWT-UCM whose boundary detector cPb is designed as the combination of two detectors mPb and fPb using the balancing weights derived from the training set, similarly to gPb, which is written as a weighted sum of mPb and its spectral component. Although cPb-OWT-UCM has a little bit poorer performance than gPb-OWT-UCM, it has better performance than mPb-OWT-UCM by adding our globalization machinery. Fig. 15 gives a visual comparison of our algorithms: FNCut, fPb-OWT-UCM, and cPb-OWT-UCM.

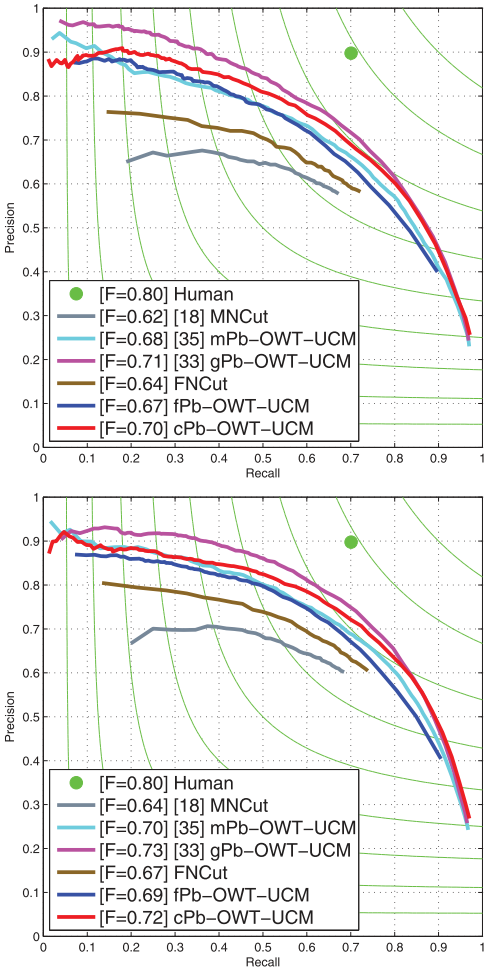

Fig. 14. Boundary benchmarks on the BSDS300 (top) and BSDS500 (bottom) datasets. In the box, several segmentation approaches are ranked according to their maximum F score in (18) with respect to human ground-truth boundaries. Iso-F curves are shown in green. Average agreement between human subjects is indicated by the green dot.

5.3.2 Region Quality

Table 2 presents region benchmarks on the BSDS300 and BSDS500 datasets using three measures: Covering, PRI, and VI. For each measure, two scores, which correspond to selecting regions from the segmentation result at a universal fixed scale (ODS) or a fixed scale per image (OIS), are reported.

K-way segmentation: In general, the spectral methods such as NCut [9] and MNCut [18] follow a K-way segmentation scheme. In Table 2, the proposed K-way segmentation algorithm FNCut has better region quality than MNCut. In detail, Fig. 16 shows a comparison of FNCut with NCut and MNCut on the BSDS datasets by varying a fixed segment number K from 5 to 40 (in steps of five). These experiments prove that FNCut outperforms MNCut, which indirectly considers long-range connections in a multiscale framework, as well as NCut in all cases. Fig. 17 shows a more visual comparison of segmentation results on the MSRC database. Compared with NCut and MNCut, FNCut produces perceptually high-quality segmentations which

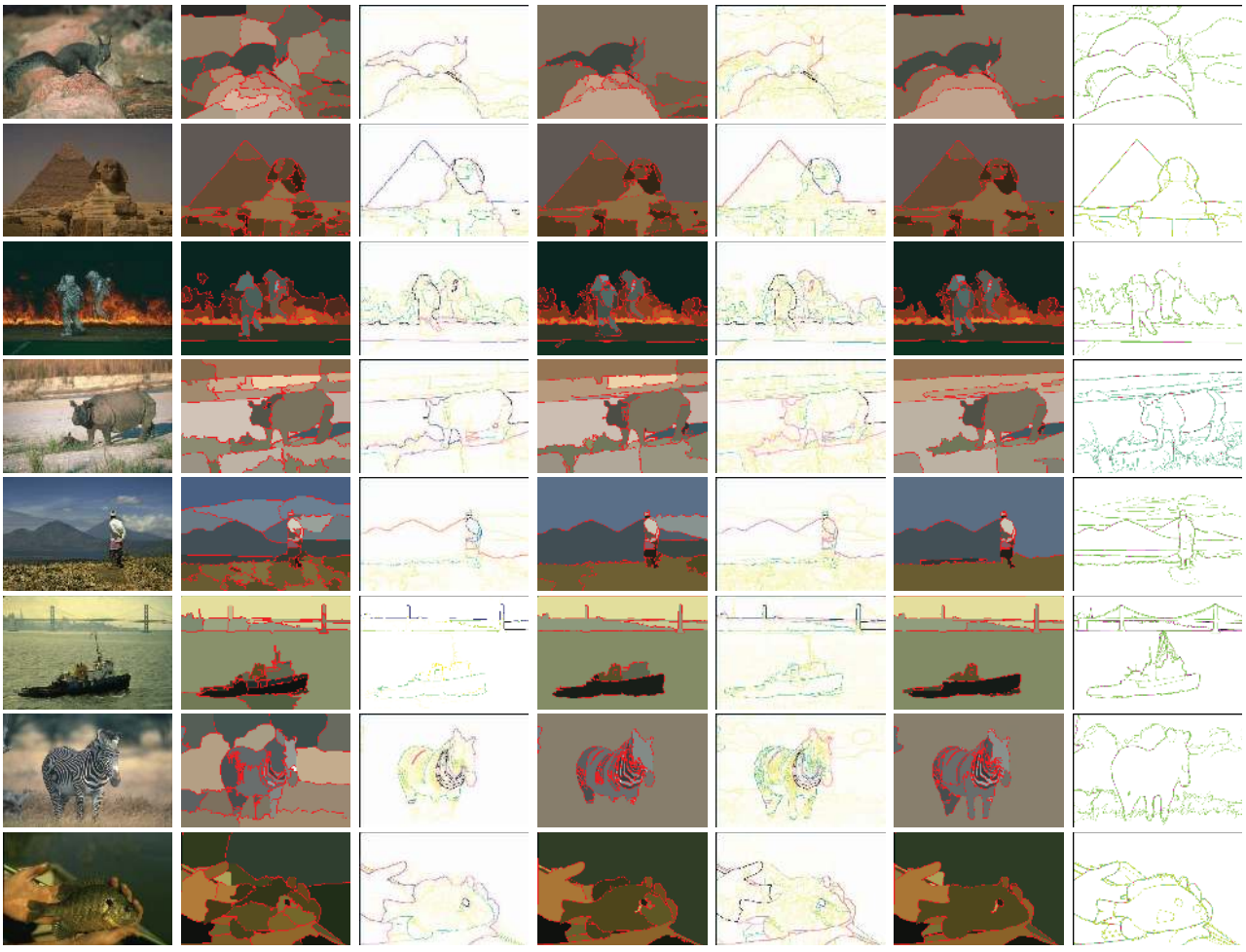

Fig. 15. Example segmentations by our algorithms: FNCut, fPb-OWT-UCM, and cPb-OWT-UCM. (a) Test images. (b), (d) and (f) Segmentation results by FNCut, fPb-OWT-UCM, and cPb-OWT-UCM at the optimal dataset scale in terms of F in (18), respectively. (c) and (e) UCMs produced by fPb-OWT-UCM and cPb-OWT-UCM, respectively. (g) Human boundaries in multiple ground-truth segmentations.

TABLE 2
Region Benchmarks on the BSDS300 and BSDS500 Datasets

	BSDS300						BSDS500					
	Covering		PRI		VI		Covering		PRI		VI	
	ODS	OIS	ODS	OIS	ODS	OIS	ODS	OIS	ODS	OIS	ODS	OIS
Human	0.73	0.73	0.87	0.87	1.16	1.16	0.72	0.72	0.88	0.88	1.17	1.17
[1] MShift	0.54	0.58	0.78	0.80	1.83	1.63	0.54	0.58	0.79	0.81	1.85	1.64
[10] GBIS	0.51	0.58	0.77	0.82	2.15	1.79	0.52	0.57	0.80	0.82	2.21	1.87
Canny-OWT-UCM	0.48	0.56	0.77	0.82	2.11	1.81	0.49	0.55	0.79	0.83	2.19	1.89
[18] MNCut	0.44	0.53	0.75	0.79	2.18	1.84	0.45	0.53	0.78	0.80	2.23	1.89
[43] SWA	0.47	0.55	0.75	0.80	2.06	1.75	-	-	-	-	-	-
[15] TVS	0.57	-	0.78	-	1.81	-	-	-	-	-	-	-
[8] TBES	-	-	0.80	0.81	1.76	1.71	-	-	-	-	-	-
[35] mPb-OWT-UCM	0.560	0.621	0.790	**0.844**	1.783	1.557	0.566	0.629	0.821	**0.852**	1.829	1.578
[33] gPb-OWT-UCM	**0.588**	**0.646**	**0.808**	**0.852**	**1.653**	**1.466**	**0.588**	**0.647**	**0.827**	**0.856**	**1.690**	**1.475**
FNCut	0.530	0.602	0.788	0.824	1.869	1.585	0.529	0.602	0.805	0.835	1.859	1.586
fPb-OWT-UCM	**0.573**	**0.633**	**0.796**	0.842	**1.693**	**1.489**	**0.582**	**0.633**	**0.819**	0.851	**1.698**	**1.500**
cPb-OWT-UCM	**0.591**	**0.645**	**0.805**	**0.851**	**1.657**	**1.460**	**0.594**	**0.653**	**0.829**	**0.861**	**1.654**	**1.449**
NCut with Ω	0.513	0.603	0.775	0.823	1.907	1.598	0.518	0.598	0.798	0.832	1.901	1.594
FNCut by ψ_{no}	0.519	0.601	0.780	0.824	1.889	1.591	0.523	0.605	0.801	0.834	1.884	1.580
sPb-OWT-UCM with Ω	0.568	0.624	0.792	0.840	1.735	1.533	0.576	0.627	0.811	0.849	1.723	1.537
fPb-OWT-UCM by ψ_{no}	0.567	0.629	0.793	0.839	1.694	1.508	0.577	0.629	0.815	0.848	1.710	1.512

sPb is the boundary detector based on spectral analysis of a sparse affinity matrix such as the weight matrix Ω.

detect large textured regions as well as elongated object parts by considering the well-defined full pairwise affinities.

Hierarchical segmentation: Unlike the state-of-the-art learning-based methods mPb-OWT-UCM [35] and gPb-OWT-UCM [33], our algorithm fPb-OWT-UCM works without training. In spite of that, the regions produced by fPb-OWT-UCM have better quality than those by mPb-OWT-UCM and a little poorer quality than those by gPb-OWT-UCM in Table 2. By additionally using local edge information in mPb, another proposed hierarchical

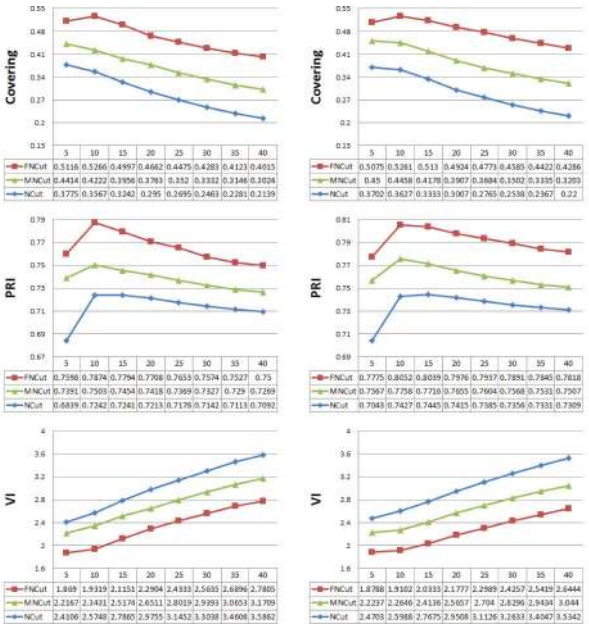

Fig. 16. Statistics of our K-way segmentation method FNCut on Covering, PRI, and VI over the BSDS300 and BSDS500 datasets with $K = \{5, 10, \ldots, 40\}$, compared with the general spectral methods NCut and MNCut.

segmentation algorithm cPb-OWT-UCM achieves the best performance in region benchmarks. Fig. 18 compares our algorithms: fPb-OWT-UCM and cPb-OWT-UCM, with gPb-OWT-UCM on the BSDS500 dataset. The region hierarchy produced by gPb-OWT-UCM conceptually well reflects the likelihood that each contour is a true boundary when the contents of a test image contains in training images, but the regions usually have oversmoothed boundaries. On the other hand, fPb-OWT-UCM provides the regions which preserve object details by directly incorporating the full-range connections, but it may fail to separate the object parts with similar color distributions. By combining advantages of these two algorithms, cPb-OWT-UCM constructs a hierarchy of object parts with detailed boundaries well.

5.3.3 Affinity Quality

We first consider that the sparse weight matrix $\mathbf{\Omega}$ in our multilayer graph, instead of the full matrix $\mathbf{\Pi}$, is directly used as the affinity matrix in the spectral segmentation algorithms. The performance of NCut and sPb-OWT-UCM under this condition is less optimal than that of our algorithms FNCut and fPb-OWT-UCM in Table 2. This comparison demonstrates the advantage of using a full affinity alone. We also evaluate the performance of the SSL function ψ_{no} in (7), instead of ψ_{un} in (9), in our algorithms FNCut and fPb-OWT-UCM. In Table 2, we know that the function ψ_{un} is quantitatively more suited than ψ_{no} for the affinity estimation in spectral segmentation.

6 Conclusion

This paper deals with two types of spectral segmentation algorithms: K-way segmentation and hierarchical segmentation. Our work is novel in that it sheds understanding on the full pairwise affinities gained in integrating local grouping cues by taking the SSL strategy. Since these

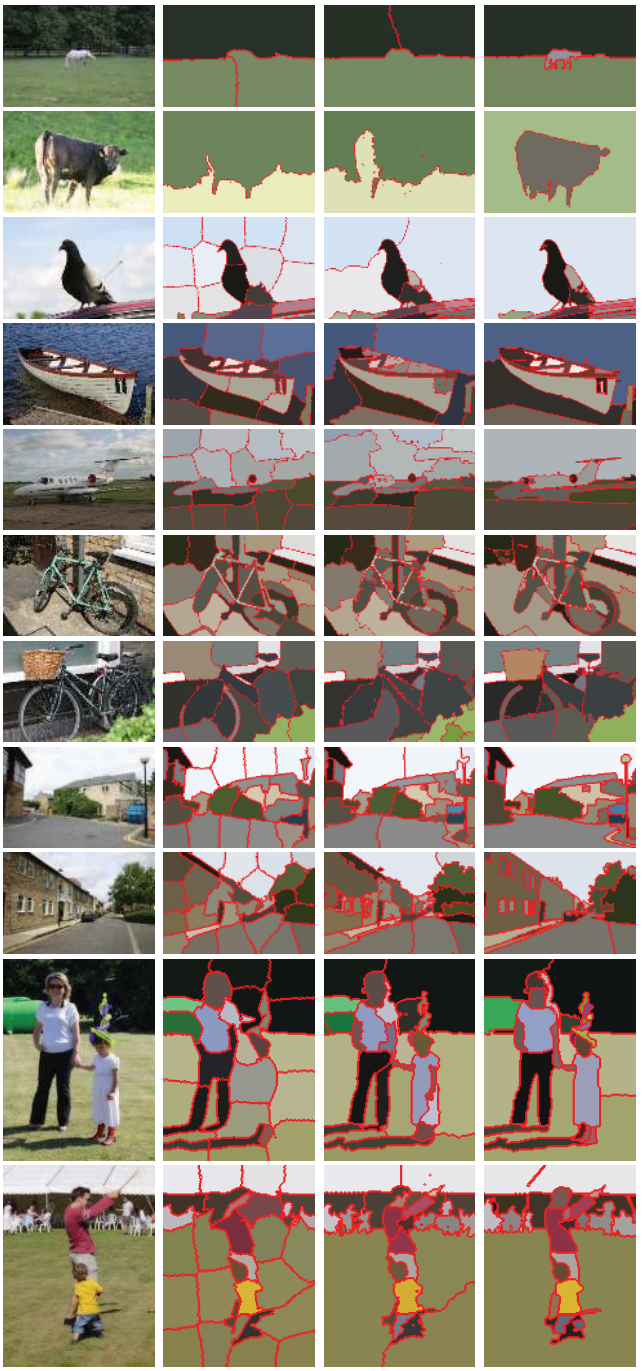

Fig. 17. Visual comparison of our algorithm FNCut with the general K-way spectral methods NCut [9] and MNCut [18] on the MSRC database. (a) Test images. (b)-(d) Segmentation results by NCut, MNCut, and FNCut, respectively.

well-defined affinities are directly embedded in our spectral segmentation algorithms FNCut and fPb-OWT-UCM without any approximation, our algorithms produce high-quality segmentation results with object details in natural images. Further, despite the use of full affinities, their spectral analysis is computationally very efficient. Finally, in case of hierarchical segmentation, by simply combining local and spectral contour signals, the advanced algorithm cPb-OWT-UCM can achieve the best performance in region benchmarks.

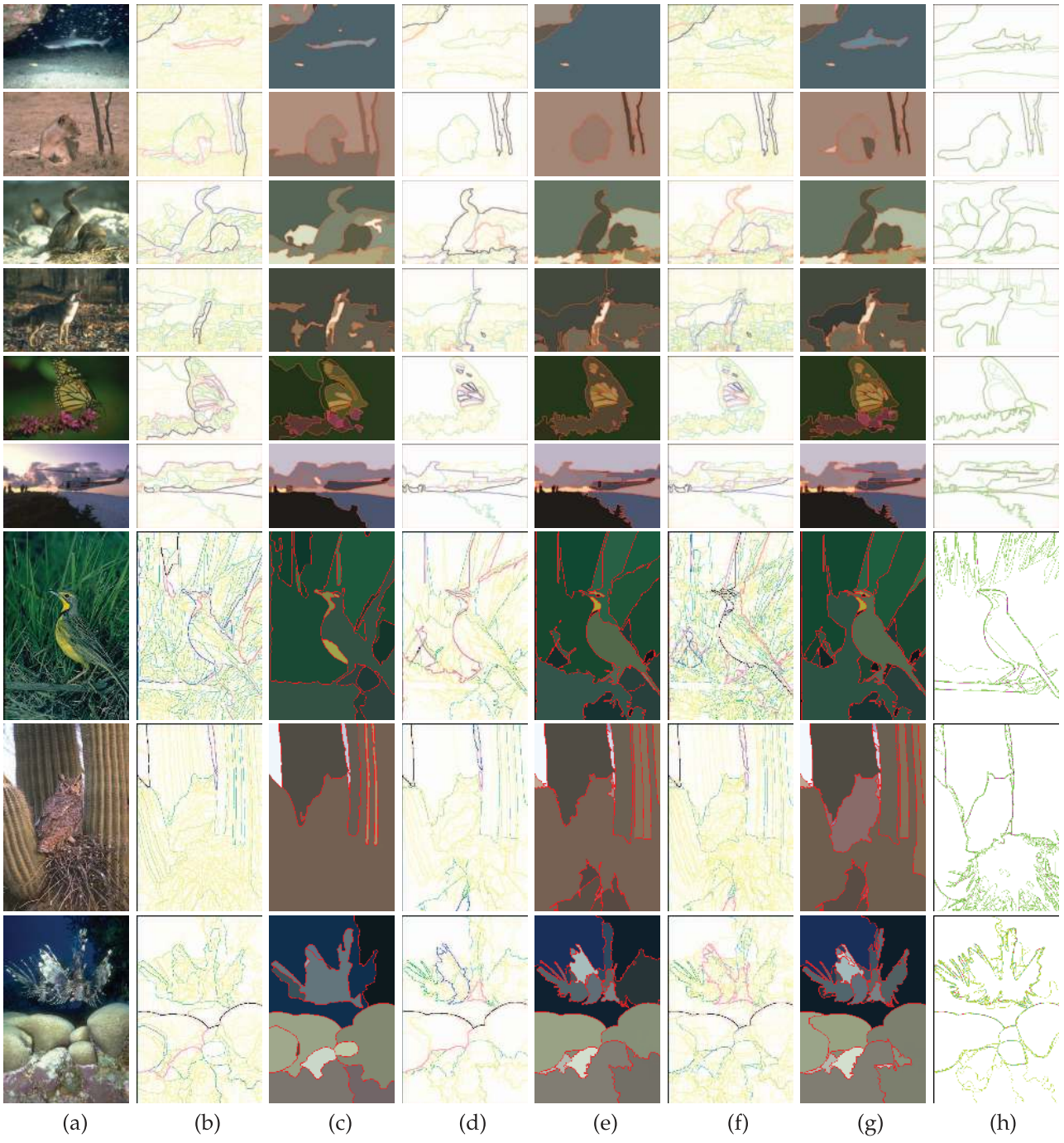

Fig. 18. Visual comparison of our algorithms fPb-OWT-UCM and cPb-OWT-UCM with the state-of-the-art hierarchical segmentation method gPb-OWT-UCM [33] on the BSDS500 dataset. (a) Test images. (b), (d), and (f) UCMs produced by gPb-OWT-UCM, fPb-OWT-UCM, and cPb-OWT-UCM, respectively. (c), (e), and (g) Segmentation results by gPb-OWT-UCM, fPb-OWT-UCM, and cPb-OWT-UCM at the optimal dataset scale in terms of the Covering measure in (20), respectively. (h) Human boundaries.

To produce an ideal segmentation result, an optimal scale should be chosen. Therefore, our future work will deal with automatic estimation of the optimal scale. To improve the performance of our algorithms, a more effective graph or SSL function also needs to be designed for affinity estimation.

REFERENCES

[1] D. Comaniciu and P. Meer, "Mean Shift: A Robust Approach toward Feature Space Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603-619, May 2002.

[2] M.Á. Carreira-Perpiñán, "Fast Nonparametric Clustering with Gaussian Blurring Mean-Shift," *Proc. Int'l Conf. Machine Learning*, 2006.

[3] J. Puzicha, T. Hofmann, and J.M. Buhmann, "Non-Parametric Similarity Measures for Unsupervised Texture Segmentation and Image Retrieval," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1997.

[4] Y. Gdalyahu, D. Weinshall, and M. Werman, "Stochastic Image Segmentation by Typical Cuts," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1999.

[5] N. Shental, A. Zomet, T. Hertz, and Y. Weiss, "Learning and Inferring Image Segmentations with the GBP Typical Cut Algorithm," *Proc. IEEE Int'l Conf. Computer Vision*, 2003.

[6] S. Sanjay-Gopal and T.J. Hebert, "Bayesian Pixel Classification Using Spatially Variant Finite Mixtures and the Generalized EM Algorithm," *IEEE Trans. Image Processing,* vol. 7, no. 7, pp. 1014-1028, July 1998.

[7] G. Sfikas, C. Nikou, and N. Galatsanos, "Edge Preserving Spatially Varying Mixtures for Image Segmentation," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2008.

[8] H. Mobahi, S. Rao, A. Yang, S. Sastry, and Y. Ma, "Segmentation of Natural Images by Texture and Boundary Compression," *Int'l J. Computer Vision,* vol. 95, no. 1, pp. 86-98, 2011.

[9] J. Shi and J. Malik, "Normalized Cuts and Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 8, pp. 888-905, Aug. 2000.

[10] P.F. Felzenszwalb and D.P. Huttenlocher, "Efficient Graph-Based Image Segmentation," *Int'l J. Computer Vision,* vol. 59, no. 2, pp. 167-181, 2004.

[11] R. Zabih and V. Kolmogorov, "Spatially Coherent Clustering Using Graph Cuts," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2004.

[12] S. Wang, T. Kubota, J.M. Siskind, and J. Wang, "Salient Closed Boundary Extraction with Ratio Contour," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 27, no. 4, pp. 546-561, Apr. 2005.

[13] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "From Contours to Regions: An Empirical Evaluation," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2009.

[14] T. Pock, A. Chambolle, D. Cremers, and H. Bischof, "A Convex Relaxation Approach for Computing Minimal Partitions," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2009.

[15] M. Donoser, M. Urschler, M. Hirzer, and H. Bischof, "Saliency Driven Total Variation Segmentation," *Proc. IEEE Int'l Conf. Computer Vision,* 2009.

[16] P. Kohli, L. Ladický, and P. Torr, "Robust Higher Order Potentials for Enforcing Label Consistency," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2008.

[17] T.H. Kim, K.M. Lee, and S.U. Lee, "Nonparametric Higher-Order Learning for Interactive Segmentation," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2010.

[18] T. Cour, F. Bénézit, and J. Shi, "Spectral Segmentation with Multiscale Graph Decomposition," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2005.

[19] C. Fowlkes, D. Martin, and J. Malik, "Learning Affinity Functions for Image Segmentation: Combining Patch-Based and Gradient-Based Approaches," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2003.

[20] C. Fowlkes, S. Belongie, F. Chung, and J. Malik, "Spectral Grouping Using the Nyström Method," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, no. 2, pp. 214-225, Feb. 2004.

[21] J. Wang, Y. Jia, X.-S. Hua, C. Zhang, and L. Quan, "Normalized Tree Partitioning for Image Segmentation," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2008.

[22] P. Arbeláez and L. Cohen, "Constrained Image Segmentation from Hierarchical Boundaries," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2008.

[23] E. Sharon, A. Brandt, and R. Basri, "Fast Multiscale Image Segmentation," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2000.

[24] S.X. Yu, "Segmenation Using Multiscale Cues," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2004.

[25] M. Maire, P. Arbelaez, C. Fowlkes, and J. Malik, "Using Contours to Detect and Localize Junctions in Natural Images," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* 2008.

[26] T. Leung and J. Malik, "Contour Continuity in Region-Based Image Segmentation," *Proc. European Conf. Computer Vision,* 1998.

[27] S. Brin and L. Page, "The Anatomy of a Large-Scale Hypertextual Web Search Engine," *Proc. Int'l Conf. World Wide Web,* 1998.

[28] D. Zhou, O. Bousquet, T.N. Lal, J. Weston, and B. Schölkopf, "Learning with Local and Global Consistency," *Proc. Neural Information Processing Systems,* 2003.

[29] J.-Y. Pan, H.-J. Yang, C. Faloutsos, and P. Duygulu, "Automatic Multimedia Cross-Modal Correlation Discovery," *Proc. ACM SIGKDD 10th Int'l Conf. Knowledge Discovery and Data Mining,* 2004.

[30] G.H. Golub and C.F.V. Loan, *Matrix Computations.* Johns Hopkins Univ. Press, 1996.

[31] Y. Weiss, "Segmentation Using Eigenvectors: A Unifying View," *Proc. IEEE Int'l Conf. Computer Vision,* 1999.

[32] S.X. Yu and J. Shi, "Multiclass Spectral Clustering," *Proc. IEEE Int'l Conf. Computer Vision,* 2003.

[33] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour Detection and Hierarchical Image Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 33, no. 5, pp. 898-916, May 2011.

[34] L. Najman and M. Schmitt, "Geodesic Saliency of Watershed Contours and Hierarchical Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 18, no. 12, pp. 1163-1173, Dec. 1996.

[35] P. Arbelaez, "Boundary Extraction in Natural Images Using Ultrametric Contour Maps," *Proc. IEEE Workshop Perceptual Organization in Computer Vision,* 2006.

[36] J. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 8, no. 6, pp. 679-698, Nov. 1986.

[37] Y. Boykov and G. Funka-Lea, "Graph Cuts and Efficient N-D Image Segmentation," *Int'l J. Computer Vision* , vol. 70, no. 2, pp. 109-131, 2006.

[38] S. Belongie and J. Malik, "Finding Boundaries in Natural Images: A New Method Using Point Descriptors and Area Completion," *Proc. European Conf. Computer Vision,* 1998.

[39] T. Malisiewicz and A.A. Efros, "Improving Spatial Support for Objects via Multiple Segmentations," *Proc. British Machine Vision Conf.,* 2007.

[40] W.M. Rand, "Objective Criteria for the Evaluation of Clustering Methods," *J. Am. Statistical Assoc.,* vol. 66, no. 336, pp. 846-850, 1971.

[41] R. Unnikrishnan, C. Pantofaru, and M. Hebert, "Toward Objective Evaluation of Image Segmentation Algorithms," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 6, pp. 929-944, June 2007.

[42] M. Meilă, "Comparing Clusterings: An Axiomatic View," *Proc. Int'l Conf. Machine Learning,* 2005.

[43] E. Sharon, M. Galun, D. Sharon, A. Brandt, and R. Basri, "Hierarchy and Adaptivity in Segmenting Visual Scenes," *Nature,* vol. 442, no. 7104, pp. 810-813, 2006.

[44] D. Martin, C. Fowlkes, and J. Malik, "Learning to Detect Natural Image Boundaries Using Local Brightness, Color and Texture Cues," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, no. 5, pp. 530-549, May 2004.

Tae Hoon Kim received the BS degree in electrical engineering from Seoul National University (SNU), Seoul, Korea, in 2006. He is currently working toward the unified MS and PhD degrees in electrical engineering and computer science at Seoul National University. His research interests include image segmentation, feature matching, and object recognition. He is a member of the IEEE and of the IEEE Computer Society.

Kyoung Mu Lee received the BS and MS degrees in control and instrumentation engineering from Seoul National University (SNU), Seoul, Korea, in 1984 and 1986, respectively, and the PhD degree in electrical engineering from the University of Southern California (USC) in 1993. He was awarded the Korean Government Overseas Scholarship during his PhD studies. He is currently with the Department of Electrical and Computer Engineering at Seoul National University as a professor, where he leads the Computer Vision Laboratory. His primary research is focused on statistical methods in computer vision that can be applied to various applications, including object recognition, segmentation, tracking, and 3D reconstruction. He has received several awards, in particular, the Most Influential Paper over the Decade Award by the IAPR Machine Vision Application in 2009, the ACCV Honorable Mention Award in 2007, the Okawa Foundation Research Grant Award in 2006, and the Outstanding Research Award by the College of Engineering of SNU in 2010. He served as an editorial board member of the *EURASIP Journal of Applied Signal Processing*, and is an associate editor of the *Machine Vision Application Journal*, the *IPSJ Transactions on Computer Vision and Applications*, and the *Journal of Information Hiding and Multimedia Signal Processing*, and *IEEE Signal Processing Letters*. He is a distinguished lecturer of the Asia-Pacific Signal and Information Processing Association (APSIPA) for 2012-2013. He has (co)authored more than 120 publications in refereed journals and conferences including the *IEEE Transactions on Pattern Analysis and Machine Intelligence*, the *International Journal of Ccomputer Vision*, CVPR, ICCV, and ECCV. He is a member of the IEEE and of the IEEE Computer Society.

Sang Uk Lee received the BS degree from Seoul National University, Korea, in 1973, the MS degree from Iowa State University, Ames, in 1976, and the PhD degree from the University of Southern California, Los Angeles, in 1980, all in electrical engineering. From 1980 to 1981, he was with General Electric Company, Lynchburg, Virginia, working on the development of digital mobile radio. From 1981 to 1983, he was a member of the technical staff at M/A-COM Research Center, Rockville, Maryland. In 1983, he joined the Department of Control and Instrumentation Engineering, Seoul National University, as an assistant professor, where he is now a professor in the Department of Electrical Engineering and Computer Science. He is also affiliated with the Automation and Systems Research Institute and the Institute of NewMedia and Communications, both at Seoul National University. He was the president of the Korean Institute of Communication Science in 2005. His current research interests include the areas of image and video signal processing, digital communication, and computer vision. He served as editor-in-chief for the *Transactions of the Korean Institute of Communication Science* from 1994 to 1996. He was an associate editor for the *IEEE Transactions on Circuits and Systems for Video Technology* from 2002 to 2005, and was on the editorial board of the *EURASIP Journal of Applied Signal Processing* from 2003 to 2004. He is currently on the editorial board of the *Journal of Visual Communication and Image Representation*. He is a member of Phi Kappa Phi. He is a life fellow of the IEEE and a member of the IEEE Computer Society.