Pattern Recognition 59 (2016) 312–324

Contents lists available at ScienceDirect

Pattern Recognition

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

Multi-scale context for scene labeling via flexible segmentation graph

Quan Zhou [a,*], Baoyu Zheng [a], Weiping Zhu [b], Longin Jan Latecki [c]

[a] *Key Laboratory of Ministry of Education for Broad Band Communication and Sensor Network Technology, Nanjing University of Posts and Telecommunications, Nanjing 21003, China*
[b] *Department of Electrical and Computer Engineering, Concordia University, Montreal, Canada*
[c] *Department of Computer and Information Science, Temple University, Philadelphia, USA*

ARTICLE INFO

Article history:
Received 31 July 2015
Received in revised form
12 March 2016
Accepted 17 March 2016
Available online 25 March 2016

Keywords:
Scene labeling
Semantic segmentation
Multi-scale context
Flexible segmentation graph
Feature extraction
Classification

ABSTRACT

Using contextual information for scene labeling has gained substantial attention in the fields of image processing and computer vision. In this paper, a fusion model using flexible segmentation graph (FSG) is presented to explore multi-scale context for scene labeling problem. Given a family of segmentations, the representation of FSG is established based on the spatial relationship of these segmentations. In the scenario of FSG, the labeling inference process is formulated as a contextual fusion model, trained from the discriminative classifiers. Compared to previous approaches, which usually employ Conditional Random Fields (CRFs) or hierarchical models to explore contextual information, our FSG representation is flexible and efficient without hierarchical constraint, allowing us to capture a wide variety of visual context for the task of scene labeling. Our approach yields state-of-the-art results on the MSRC dataset (21 classes) and the LHI dataset (15 classes), and near-record results on the SIFT Flow dataset (33 classes) and PASCAL VOC segmentation dataset (20 classes), while producing a 320 × 240 scene labeling in less than a second. A remarkable fact is that our approach also outperforms recent CNN-based methods.

1. Introduction

Multi-class segmentation plays an important role in the field of computer vision, leading to the challenging problem of *scene labeling* that is an important function for image processing and understanding. From the perspective of a human vision system, scene labeling aims to automatically partition an image into semantically meaningful regions. On the other hand, from the perspective of computer vision, the goal of scene labeling is to assign each pixel a semantic label which indicates its potential category, achieving synchronous recognition and delineated segmentation for every object in natural image. Scene labeling is related to many applications, such as object detection [1,2] and recognition [3], visual attention [4], scene understanding [5,6], medical image segmentation [7], image alignment [8] and matching [9–11]. Therefore, the scene labeling problem has been extensively studied in image processing, computer vision and machine learning literature [12–19].

Recently, using contextual information has gained increasing attention for the scene labeling [20–22]. According to the usage of contextual information, most of existing models are mainly divided into two categories: short-range [23–25] and long-range interactions [15,21,26,27]. In spite of achieving promising results, there are still two primarily important issues that need to be considered in the contextual based modeling for scene labeling:

- How to produce a flexible and powerful representation to capture the wide variety visual context in a given scene?
- How to integrate the contextual information into a well-designed model to ensure a globally consistent pixel-level labeling results?

In this paper, we make an effort to address these two questions based on multi-scale contextual formulation. The main idea is based on two common observations: (a) One is that identifying a larger image region provides strong evidence for classifying the contained smaller ones. For example, if a region is recognized as “sky”, it indicates that the covered smaller ones are more likely to be also labeled as “sky”. Thus, the spatial relationships of containing among regions are considered to investigate this kind of context. (b) However, only using one scale context is difficult to assign semantic category for each pixel. The category of a pixel may rely on relatively short-range intersections, but may also depend on long-range information. For instance, recognizing a green pixel belonging to a “grass” or a “tree” requires wide scale contextual clues that show enough of the surroundings to make a discriminative decision. In order to address these two problems, we describe a novel representation, *flexible segmentation graph* (*FSG*), which is able to efficiently investigate covering spatial relationships among the ensemble segmentations. Thereafter, the

[*] Corresponding author.

http://dx.doi.org/10.1016/j.patcog.2016.03.023
0031-3203/© 2016 Elsevier Ltd. All rights reserved.

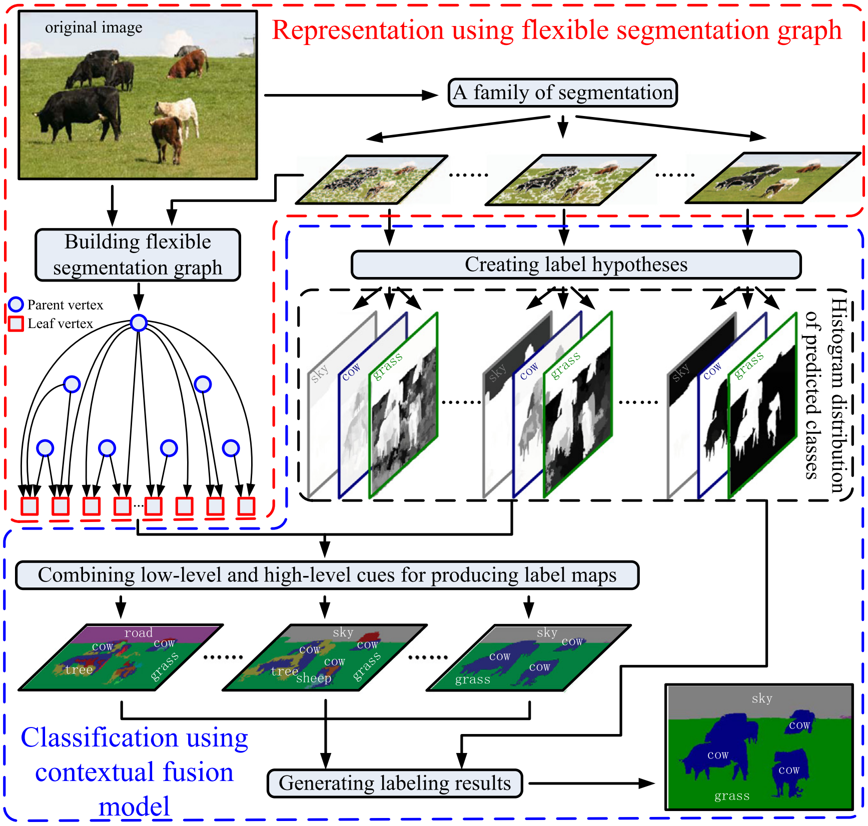

original image
Representation using flexible segmentation graph
A family of segmentation
Building flexible segmentation graph
Parent vertex
Leaf vertex
Creating label hypotheses
sky
cow
grass
Histogram distribution of predicted classes
Combining low-level and high-level cues for producing label maps
road
cow
tree
grass
sky
sheep
Classification using contextual fusion model
Generating labeling results

Fig. 1. Diagram of our scene labeling approach. The raw input image is partitioned into a series of super-pixels to construct a family of segmentations, which is organized in an incrementally coarse to fine manner. Note that each super-pixel is split by white boundary. Our flexible segmentation graph is established based on the spatial interactions between these super-pixels. In parallel, the label hypotheses, as histogram distribution of predicted classes, are created for high level segmentations, where white color represents low confidence, while black color indicates high confidence. Thereafter, a series of label maps are produced based on the low-level features and high-level cues. Two types of maps are integrated to produce the final labeling result (best viewed in color). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

contextual cues are embedded into a contextual fusion model according to the established FSG. More specifically, the proposed scene labeling architecture is depicted in Fig. 1, which relies on the following two components.

1.1. Multi-level, flexible structure graph representation

A family of segmentations in coarse to fine manner is constructed over an input image to investigate contextual cues at multiple levels. In each level, an over-segmentation is performed to partition an input image into disjoint regions. This segmentation family might be any set of segmentations, for instance, a collection of super-pixels either produced via different segmentation algorithms [28,29] or using the same algorithm with different parameter settings [30–32]. Given a family of segmentations, an FSG is established based on the spatial relationship of containing among the segmented super-pixels. The vertex set of FSG consists of two components: leaf vertices and parent vertices. Each leaf vertex is defined in the finest segmentation level, and connects to some parent vertices, where associated regions cover the smaller one of leaf vertex. Meanwhile, one parent vertex may also connect to several leaf vertices in the finest level. Since the parent vertices may be from inconsecutive high level segmentations, our segmentation graph has flexible structure so that we can explore contextual information efficiently and flexibly. To our best knowledge, FSG has been rarely used in the scenario of contextual-based scene labeling, allowing us to capture contextual information within multiple scales to recognize all objects and regions in the given scene.

1.2. Classification using contextual fusion model

In the scenario of FSG, a contextual fusion model is established using Bayesian rule to capture multi-scale contextual information. It treats scene labeling as a hidden variable integration problem, where the hidden variables are the parent vertices and their associated label hypotheses. The corresponding region of each parent vertex is encoded by an aggregated low-level appearance feature vectors (e.g., color, shape and texture), a discriminative classifier is then applied to the aggregated features. As a result, a series of label hypotheses are created as the estimated histogram distribution of all object categories presented in the parent vertices. These high-level label hypotheses predictions, once again aggregated with the appearance cues, are fed into another classifier for identifying the leaf vertices. Finally, the output of two types of classifiers are integrated to assign a final single class per leaf vertex. Our contextual fusion model is very simple and effective. It is able to parse a 320×240 image in less than a second using a conventional personal computer. The computational burden lies in the training procedure of classifiers. Once trained, our contextual fusion model is parameter free, without sophisticated inference process. In summary, the contributions of this paper are mainly summarized as follows:

- We propose a novel FSG representation to capture multi-scale visual context. Compared with existing methods, our FSG representation is flexible and efficient without hierarchical constraint, allowing the contextual cues defined at different scales to contribute to predict region labels.
- Unlike previous methods that often utilize CRF and MRF models to enforce local compatibility and global consistency for labeling

problem, we establish a contextual fusion model to formulate multi-scale context. Our model yields similar to or better labeling result than competing models, and is also computationally efficient.

The remainder of this paper is organized as follows. After a brief discussion of related work in Section 2, we describe the construction of FSG in Section 3. Section 4 elaborates on the details of contextual fusion model based on FSG. Experimental results are given in Section 5. Finally, we give concluding remarks in Section 6.

2. Related work

We review the related work from two aspects of contextual modeling for scene labeling problem: short-range and long-range interactions.

2.1. Short-range intersections

Many of the labeling approaches [23–25,33–35] construct successful systems that capture short-range contextual information based on the image statistics of surrounding patches or regions. As one of the earliest methods, Belonge et al. [24] proposed a "looking around" operation to encode local contextual information. Tu and Bai [33] formulated visual context using surrounding patches within rigid position. Kumar and Hebert introduced a Discriminative Random Field (DRF) [34] which is defined on a graph within a two-dimensional lattice structure. DRF learns pairwise compatibilities including image information between labels of different nodes. The short-range interactions can be also encoded in terms of *generative* models, such as MRFs [23,25]. In MRFs, neighboring label variables are connected to each other so that their values are not independent. By combining local pairwise interactions between variables, MRFs impose a global constraint on the label predictions, leading to more consistent labeling results of an image.

2.2. Long-range intersections

Besides using short-range intersections, vast majority of scene labeling methods attempt to explore global-based context using CRFs. Shotton et al. [35], for example, utilize texton-based integrated images to capture long-range contextual cues; Galleguillos et al. [36] and Gould et al. [37] employe the co-occurrence preference and relative location as contextual features in their probabilistic construction. Yadollahpour et al. [38] proposed a two-stage labeling framework, where in the first stage a set of labeling hypothesis are produced via probabilistic CRF model, and the second stage trained the discriminative re-ranking model to find the best labeling results from this set. An alternative approach of using CRFs to capture long-range context is to integrate object detection into probabilistic graphical models [1,39–41], which combine pixel-based, object-based and scene-based clues for solving the scene labeling problem. Unlike these methods, we employ the covering relationship to capture visual context, and investigate this kind of context in different scales, without requiring any complex inference procedures to yield cleanly delineated predictions, such as sampling technique [23] or graph cut algorithms [42,43].

Another approach for capturing long-range context is to build hierarchical models [14,26,27,42–45]. Some authors employ the families of segmentations to generate the representation of segmentation tree, which is organized within a rigid hierarchical structure via aggregating elementary segments [30,31,45]. The authors of [46] also adopted an iterative grouping technique [3] to form hierarchical segmentation tree, and then trained a joint calibration model to estimate pixel labels. Recently, an alternative strategies to *implicitly* capture contextual clues appeared in [12,14,21] and [47] using deep learning techniques. In their work, a multi-layer convolutional neuronal network (CNN) is adopted to generate hierarchical feature representations, which are fed into a well-trained end-to-end model to predict semantic label for each pixel. Compared with these approaches, we *explicitly* encode context by establishing the representation of FSG, which is more flexible and efficient, allowing a wide variety of long-range contextual cues within different scales to contribute to the confidence in each semantic label. Additionally, the proposed method is fully automatic, without any parameter tuning in postprocessing [12,14,15] or any human interactions as in [27,48].

An early version of this work was first published in [32]. This journal version extends previous one in three aspects: the previous version required a well-designed segmentation hypothesis, still resulting in hierarchical representation, while the proposed FSG is not limited by any hierarchical constraint; besides using texture cues to encode image regions, we also address the cues of color, size, shape and the class distributions to predict region labels; we have implemented more complete experiments, and reported more comparisons and improved results.

3. Image representation using flexible segmentation graph (FSG)

Traditional segmentation approaches to investigate global contextual information consider a segmentation tree [49,50], where the segmented regions are hierarchically organized. An alternative technique is to calculate a set of segmentations using different merging thresholds [31]. In this section, we propose a method to analyze a family of segmentations, which is used to establish the representation of FSG, without any restriction to the hierarchical structure.

As shown in Fig. 2(a), an input image $\mathcal{I}$ is first partitioned into a series of super-pixels using mean shift segmentation technique [28] with different parameter settings. The involved parameters are the spatial resolution parameter h_s, the range resolution parameter h_r and the size of smallest segments *Min*. In practice, we initialize these three parameters as $h_s = 1$, $h_r = 1$, and $Min = 100$. Then a family of segmentations with $\mathcal{L}$ levels is produced in a coarse to fine manner by incremental increase these parameters with updated step as 2, 1, and 100, respectively. Let $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ denote our FSG, in which image $\mathcal{I}$ and containing spatial interactions are encoded based on these super-pixels. As illustrated in Fig. 2(b), the vertex set $\mathcal{V} = \{\mathcal{V}_L, \mathcal{V}_P\}$ is composed of two components: the set of leaf vertices $\mathcal{V}_L$ (denoted as red squares in Fig. 2(b)) defined on finest segmentation level, and the set of parent vertices $\mathcal{V}_P$ defined on other segmentation levels (denoted as blue circles in Fig. 2(b)). Each leaf vertex $v_i \in \mathcal{V}_L$ is associated with a super-pixel r_i and each parent vertex $v_j \in \mathcal{V}_P$ is associated with a super-pixel r_j. Note that in the finest segmentation level, there is no overlap between any two leaf vertices. Since the whole image provides the wide contextual cues to identify each leaf vertex, we directly use it in the coarsest segmentation level of our FSG.

The edge set $\mathcal{E}$ consists of a set of edges $\varepsilon = \{(v_i, v_j) | v_i \in \mathcal{V}_L, v_j \in \mathcal{V}_P\}$, connecting a leaf vertex v_i and a parent vertex v_j, where the associated super-pixel r_i is covered by the larger super-pixel r_j, as shown in Fig. 2(a). For a specific super-pixel r_i, we collect all the larger super-pixels r_j that contain r_i, and denote the associated parent vertices as $\mathcal{S}_i = \{v_j^m, m \in \{1, 2, \ldots, M\}\}$. Note that the parent vertex $v_j^m \in \mathcal{S}_i$ may be from nonconsecutive segmentation levels when super-pixel r_i straddling over multiple super-pixels in higher segmentation levels. As shown in Fig. 2(b), one parent vertex may connect several leaf vertices and vice versa, leading to the flexible structure of our FSG.

Ideally, we would like to consider as many as possible of family segmentations so that we can make full use of multi-scale contextual information. However, this would require a significant computational effort, and thus only small number of levels of

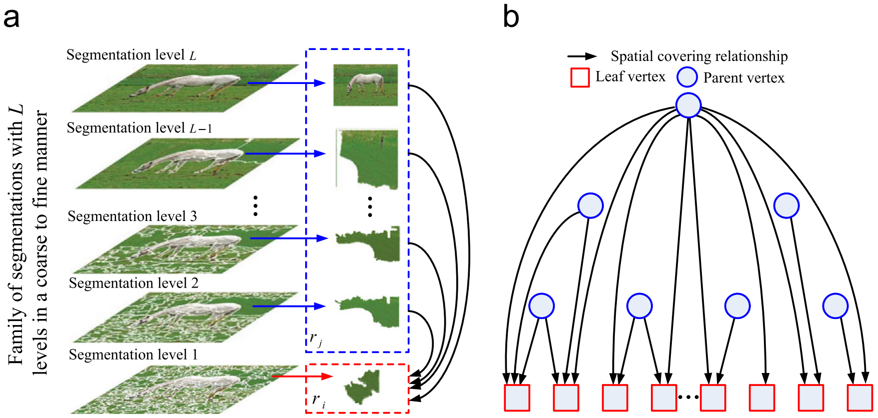


Fig. 2. An illustration of constructing FSG. A family of segmentation and one sample of spatial covering relationship among super-pixels are illustrated in (a), and (b) shows the sketch map of FSG. In (a), different super-pixels are separated by white boundaries (best viewed in color). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

segmentations is considered. The experimental results also demonstrate that several level of segmentations are sufficient to investigate multi-scale context, ensuring the consistency of the labeling output. From Fig. 2(a), although those super-pixels tend to be highly irregular in size and shape, the advantage of using mean shift segmentation [28] lies in the fact that it can often group large homogeneous regions with similar appearance while dividing heterogeneous regions into many smaller ones. This often produces fewer super-pixels in each level of segmentation. Alternative over-segmentation approaches include hierarchical segmentation [49], graph-based segmentation [29], and normalized cuts [51]. These methods require much more processing time to produce super-pixels or the generated super-pixels have imprecise boundaries.

4. Contextual fusion model

In this section, we first elaborate on the details of the contextual fusion model in scenario of FSG, then describe the associated feature and learning algorithm for training this model.

4.1. Problem formulation

Given the observed image $\mathcal{I}$ and associated FSG $\mathcal{G}$ with $|\mathcal{V}_L|$ leaf vertices, our formulation contains a set of discrete random variables $\mathbf{Y}=\{Y_1,Y_2,\ldots,Y_{|\mathcal{V}_L|}\}$. The ith element Y_i corresponds to leaf vertex v_i, and may take a discrete value from the set of semantic labels: $Y_i=c\in\{1,2,\ldots,\mathcal{C}\}$. Any possible assignment of labels to the random variables $\mathbf{Y}$ will be called a labeling problem which takes values from solution space $\Omega=\{1,2,\ldots,\mathcal{C}\}^{\mathcal{V}_L}$. Our objective is to compute $\mathbf{Y}^*$ that maximizes a posteriori probability,

$$\mathbf{Y}^*=\arg\max_{\mathbf{Y}\in\Omega} p(\mathbf{Y}|\mathcal{I},\mathcal{G}) \tag{1}$$

Due to the non-overlapping of r_i in the lattice defined on image $\mathcal{I}$, the image model $p(\mathbf{Y}|\mathcal{I},\mathcal{G})$ is assumed to be conditionally independent on $\mathbf{Y}$, then this posteriori probability is further factorized onto each leaf vertex. We thus have,

$$p(\mathbf{Y}|\mathcal{I},\mathcal{G})=\prod_{i=1}^{|\mathcal{V}_L|} p(Y_i|v_i) \tag{2}$$

In order to investigate contextual information within different scales, we marginalize over all possible parent vertices defined in $\mathcal{S}_i$ that are connected with v_i:

$$p(Y_i|v_i)=\sum_{v_j^m\in\mathcal{S}_i} p(Y_i,v_j^m|v_i)\propto\sum_{v_j^m\in\mathcal{S}_i} p(v_j^m|v_i)p(Y_i|v_i,v_j^m) \tag{3}$$

where $p(Y_i|v_i,v_j^m)$ is the probability to assign a label Y_i to leaf vertex v_i given observed parent vertex $v_j{}^m$. $p(v_j^m|v_i)$ denotes the probability to form the larger super-pixel r_j given the smaller super-pixel r_i. In [31], the authors estimate $p(v_j^m|v_i)$ using region homogeneity to merge super-pixels, assuming super-pixels with similar region homogeneity should also belong to the same category. However, the inverse might not be true since object category might have great visual variance. We assume there is no prior knowledge to guide segmentation, and r_j might be generated by those pixels that are inhomogeneous. As a result, the formation of r_j is independent to the given r_i and the term $p(v_j^m|v_i)$ is considered as a constant in our formulation. Eq. (3) thus can be simplified as:

$$p(Y_i|v_i)\propto\sum_{v_j^m\in\mathcal{S}_i} p(Y_i|v_i,v_j^m) \tag{4}$$

In order to incorporate label hypotheses predictions as high-level semantic contextual clues from parent vertex $v_j^m\in\mathcal{S}_i$, we further marginalize Eq. (4) over all category labels Z_j of $v_j{}^m$

$$p(Y_i|v_i)\propto\sum_{v_j^m\in\mathcal{S}_i}\sum_{Z_j} p(Y_i,Z_j|v_i,v_j^m)\propto\sum_{v_j^m\in\mathcal{S}_i}\sum_{Z_j} p(Z_j|v_i,v_j^m)p(Y_i|Z_j,v_i,v_j^m) \tag{5}$$

Since super-pixel r_j always contains super-pixel r_i, the first term can be treated as independent of v_i. Eq. (5) thus reduces to

$$p(Y_i|v_i)\propto\sum_{v_j^m\in\mathcal{S}_i}\sum_{Z_j} p(Z_j|v_j^m)p(Y_i|Z_j,v_i,v_j^m) \tag{6}$$

As can be seen, by sequentially integrating connected parent vertex $v_j{}^m$ and its associated label hypotheses, the multiple scale contextual information are successively fused into the posterior probability model $p(Y_i|v_i)$ to identify leaf vertex v_i. More preciously, $p(Z_j|v_j^m)$ denotes the normalized probability that assigns semantic label Z_j for parent vertex $v_j{}^m$ based on image features abstracted from corresponding super-pixel r_j. This term indicates that classifying larger regions may be helpful to identify smaller ones. From $p(Y_i|Z_j,v_i,v_j^m)$, it is clear that besides local clues of v_i, identifying leaf vertex v_i requires to consider image statistics and high-level label predictions from parent vertex $v_j{}^m$. Note the sum is over the parent vertices $v_j^m\in\mathcal{S}_i$, rather than all the parent vertices $\mathcal{V}_P$ in FSG, which results in fast computational speed to estimate labeling results of input test image.

The optimal labeling output $\mathbf{Y}^*$ can be achieved by assigning a label with the highest posteriori probability to each leaf vertex. Immediately below, we will entail the associated features and the forms of $p(Z_j|v_j^m)$ and $p(Y_i|Z_j,v_i,v_j^m)$ using simple regression boosted classifiers [52].

4.2. Features

To determine the most probable label for each vertex in FSG, we are required to use all available cues, including low-level image statistics, such as color, size, shape, location, texture, and high-level semantics as the estimated histogram of all object categories. Some of these statistics, however, are only helpful when object category has less visual variance. For example, it is hard to identify "car" that are with different colors. Therefore, our approach computes all cues that might be useful for classification, and allows our classifier (described in Section 4.3) to automatically decide which one should be used and how to use them.

4.2.1. Low-level features

Our low-level statistical features build on those of Barnard et al. [53] and Zhang et al. [54], consisting of mean, standard deviation, skewness, kurtosis, color histograms and bag of features (BoF) over the super-pixel of the following:

- RGB color-space components (3×4) and RGB color histogram distributions (3×10).
- CIELab color-space components (3×4) and CIELab color histogram distributions (10 bins for L-channel).
- HSV color-space components (3×4 with additional 5 bin and 3 bin histograms for hue and saturation, respectively).
- Size cues as the ratio of region area to entire scene (1).
- Location cues as the offsets in x and y direction, and distance from image center (3).
- Shape cues as the ratio of the region area to perimeter squared, the moment of inertia about the center of mass, and the ratio of area to bounding rectangle area (3).
- Texture cues drawn from 17 filter responses, including Gaussian, oriented Gaussian, Laplacian-of-Gaussian, and pattern features such as corners and bars (4×17).
- Texture cues drawn from BoF as histogram distribution of learned visual dictionary words (700).

4.2.2. High-level features

Unlike the low-level clues, high-level features provide semantic information for recognizing objects and image regions [55,56]. According to Eq. (6), identifying the label of leaf vertex v_i requires to consider the label variable Z_j of parent vertex ${v_j}^m$. It is a $\mathcal{C}$-dimensional vector as the distribution of classes present in super-pixel r_j. Given the ground truth segmentation in the training process of $p(Y_i|Z_j, v_i, v_j^m)$, the features can be directly computed. At test stage, however, no ground truth segmentation is available, therefore, we need a function that can predict the cost of class distribution for r_j. In practice, we directly apply the trained classifiers $p(Z_j|v_j^m)$ over super-pixel r_j and collect the outputs of classifiers to form this high-level semantic feature.

Let X_i denote the feature vector to describe the leaf vertex v_i with associated super-pixel r_i. Since we are required to compute low-level features from v_i and ${v_j}^m$, and high-level label hypotheses of ${v_j}^m$ to describe r_i, X_i is a $859+859+\mathcal{C}=1718+\mathcal{C}$-dimensional vector based on Section 4.2.1. As well as [55] does, we append X_i with the additional description vector, considering the weighted average over its neighbors:

$$\frac{\sum_{r_{ik} \in \mathcal{N}(r_i)} |r_{ik}| \cdot X_{ik}}{\sum_{r_{ik} \in \mathcal{N}(r_i)} |r_{ik}|} \tag{7}$$

where $\mathcal{N}(r_i)$ is the set of super-pixels which are adjacent with r_i in the image $\mathcal{I}$, and $|r_{ik}|$ is the number of pixels in super-pixel r_{ik}. Finally, it is $(1718+\mathcal{C}) \times 2$-dimensional for a leaf vertex in FSG. Similarly, the same operation is also applied to X_j associated with super-pixel r_j. The final feature vector is $859 \times 2 = 1718$-dimensional for a parent vertex.

4.3. Classifiers and learning algorithm

In this paper, both $p(Z_j|v_j^m)$ and $p(Y_i|Z_j, v_i, v_j^m)$ can be trained using logistic regression version of Adaboost [52]. For notation simplicity, we use $p(Y|X)$ to represent $p(Z_j|v_j^m)$ and $p(Y_i|Z_j, v_i, v_j^m)$, respectively, then the form of $p(Y|X)$ is

$$p(Y=c|X) \propto \exp\{H^c(X)\} \tag{8}$$

where $H^c(X) = \sum_{k=1}^{K} h_k^c(X)$ is an additive model for the cth category by accumulating the classification confidences of K weak learners $h_k^c(X)$. Each weak learner $h_k^c(X)$ adopts a two-terminal node decision tree ("stump") based on input X, and has the following form:

$$h_k^c(X) = f_{k,left}^c \mathbf{1}_{[x_k^m \le \tau_k]} + f_{k,right}^c \mathbf{1}_{[x_k^m > \tau_k]} \tag{9}$$

where ${x_k}^m$ denotes the mth variable of X selected for the kth weak learner, τ_k is the split-point. $f_{k,left}^c$ and $f_{k,right}^c$ are, respectively, the weighted log-ratio for the left and right terminal nodes, which are learned from the training data.

Decision trees are good weak learners, since they are able to automatically perform feature selection and provide limited modeling of the joint statistics of data. Each week learner partitions the training data by a confidence-weighted decision which is the class-conditional log-likelihood ratio for the current weighted distribution. Note weight update rule of logistic regression version of Adaboost differs from the original ones, but it results in confidence outputs that tend to be well-calibrated probabilities (after applying the simple sigmoid conversion to the log-ratio output).

Algorithm 1. Training boosted decision trees.

```
Input: $X_1, \ldots, X_m$: training data; $\mathbf{w}^1 = \{\omega_1^1, \ldots, \omega_m^1\}$: initial weights, where $\omega_i^1 = \frac{1}{m}$, $i = \{1, \ldots, m\}$; $y_1, \ldots, y_m \in \{1, -1\}$: labels;
$s$=2: number of nodes per decision trees; $K$: number of iterations
Output: $h_1, \ldots, h_K$: decision trees; $f_1^1, \ldots, f_K^s$:
weighted log-ratio for each node of each tree
for $k$= 1 to $K$ do
    Learn $k$−node decision tree $h_k(X)$ based on weighted distribution $\mathbf{w}^k$.
    Assign to each node of $h_k(X): f_{k,s} = \frac{1}{2} \log \frac{\sum_{i:y_i=1, X_i \in f_{k,s}} \omega_i^k}{\sum_{i:y_i=-1, X_i \in f_{k,s}} \omega_i^k}, s = \{left, right\}$.
    Update weights : $\omega_i^{k+1} = \frac{1}{1+\exp(y_i \sum_{k'=1}^{k} f_{k',s_{k'}})}$.
  with $s_{k'} : X_i \in T_{k',s_{k'}}$.Normalize weights so that $\sum_i \omega_i^{k+1} = 1$.
end
```

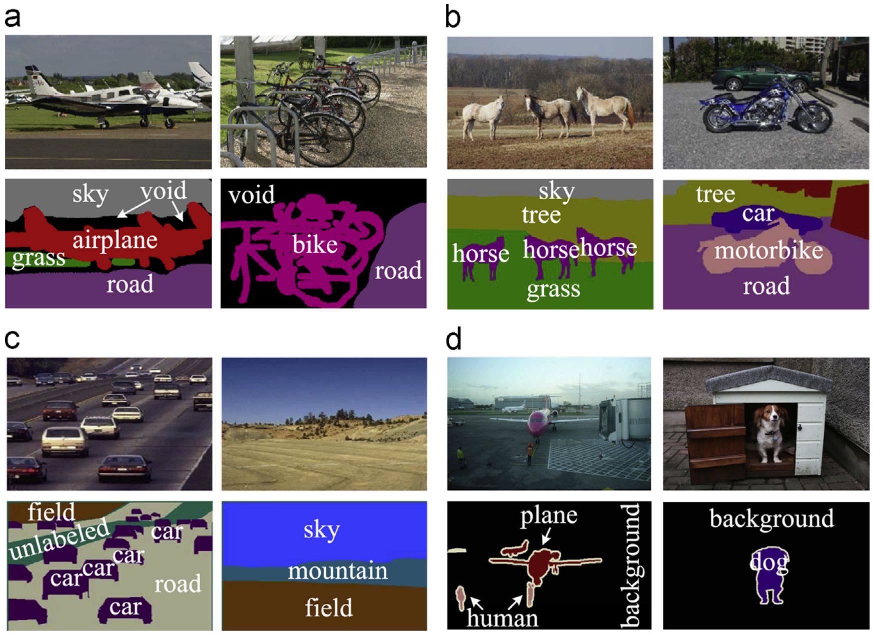


Fig. 3. Example images of (a) MSRC 21-class, (b) LHI 15-class, (c) SIFT flow 33-class and (d) PASCAL VOC segmentation dataset. The first row displays the original images, and the second row is the corresponding ground truth. For clarity, textual labels have also been superimposed on the ground truth (best viewed in color). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

In our implementation, all the classifiers are trained in a one vs. all fashion. Here, we take positive examples as the super-pixels which are assigned to that class in the ground truth labeling, and negative examples as all super-pixels which are assigned to a different class in the ground truth. For instance, to distinguish "tree" class, we train the classifiers that estimate the probability of a super-pixel having the remaining semantic labels. The trained classifiers are applied for each category c to the vector of descriptors X, and normalized over all classes by $\frac{\exp\{H^c(X)\}}{\sum_c \exp\{H^c(X)\}}$ to ensure that the estimated probabilities sum to one. The classifier training algorithm is summarized in Algorithm 1.

5. Experiments

The purpose of our experiments is to evaluate the effectiveness of our method, and better understand the behavior of our labeling system. Our analysis includes comparison with recent state-of-the-art methods in the literature, the impact of different cues on recognition accuracy, and the influence on segmentation level.

5.1. Experimental setting

5.1.1. Dataset

We evaluate our system on four challenging datasets: MSRC 21-class dataset [35], LHI 15-class dataset [57] on which related state-of-the-art methods report labeling accuracy and efficiency, as well as a more challenging dataset with a larger number of images and classes: SIFT flow 33-class dataset [8] and PASCAL VOC segmentation dataset [58]. All images are rescaled to 320 × 210 resolution in four datasets, and some examples and associated ground truth are illustrated in Fig. 3.

The MSRC 21-class dataset [35] is a very popular benchmark for scene labeling, which consists of 591 images including 21 classes: "building", "grass", "tree", "sky", "water", "book", "road", "body", "boat", "flower", "sign", "cow", "sheep", "aeroplane", "face", "car", "bike", "bird", "cat", "dog" and "chair". The pixels labeled as "void" class are not considered during the training and testing for direct comparison.

The LHI 15-class dataset [57] consists of 370 images gathered from Google image search, and includes 15 object categories: "building", "grass", "tree", "sky", "road", "water", "mountain", "airplane", "cow", "horse", "sheep", "car", "elephant", "rhinoceros" and "motorbike". Compared with MSRC 21-class dataset, images in LHI 15-class dataset are well hand-annotated to achieve accurate segmentation.

The SIFT flow 33-class dataset [8] is composed of 2688 images, that have been thoroughly labeled by LabelMe users. The authors used synonym correction to obtain 33 semantic categories, including: "sky", "building", "mountain", "tree", "road", "sea", "field", "grass", "river", "plant", "car", "sand", "rock", "sidewalk", "window", "desert", "door", "bridge", "person", "fence", "balcony", "crosswalk", "staircase", "awning", "sign", "streetlight", "boat", "pole", "sun", "bus", "bird", "moon" and "cow". It is also a fully annotated dataset, most of which are outdoor scenes including street, beach, mountains and fields. Similar as MSRC 21-class dataset, the pixels labeled as "unlabeled" class are not considered during the training and testing for direct comparison.

The PASCAL VOC segmentation dataset [58] is another widely used benchmark for the task of scene labeling. This dataset defines 20 object categories, including: "person", "bird", "cat", "cow", "dog", "horse", "sheep", "aeroplane", "bicycle", "boat", "bus", "car", "rock", "motorbike", "train", "bottle", "chair", "table", "plant", "sofa", and "TV/monitor". This dataset provides around 3000 images, named trainval, with pixel-wise ground truth annotations. It is further divided into half in the training and half in the validation set. Similar as MSRC 21-class dataset, the pixels labeled as "background" class are not considered in our training and testing process.

5.1.2. Baselines

To show the advantages of our approach, we selected 10 state-of-the-art models as baselines. Experimental results of some baseline models are produced using default parameter settings given by the authors, while others are directly borrowed in the literature for comparison. All the baselines are divided into two categories: (1) Modeling scene labeling using CRFs or MRFs, including Texton-Boost (TB, [35]), relative location prior (RL, [37]), hierarchical CRF (HCRF, [42]), dynamic hybrid MRF (DHM, [23]), and full-connected CRF (FCRF, [59]); (2) Modeling scene labeling using hierarchical trees, including region ancestry (RA, [30]), stacked hierarchical labeling (SHL, [26]), and CNN-based approaches, such as hierarchical deep learning (HDL, [21]), fully convolutional networks (FCNN, [15]), deep hierarchical parsing (DHP, [14]), deep convolutional nets and fully connected CRFs (DCNFCC, [60]), CRFs as recurrent neural networks (CRFasRNN, [61]).

5.1.3. Evaluation metrics

We evaluate our labeling models based on the following widely-used criteria, named *global*-based pixel accuracy (*GPA*), *average*-based class accuracy (*ACA*), and mean intersection over Union (*mIoU*) [15,42]. Let N_{mn} be the number of pixels of category m labeled as class n, where there are $\mathcal{C}$ different object classes, then the three evaluation metrics are defined as:

- *GPA* pays the most attention to frequently occurring objects and penalizes infrequent objects. It refers to overall accuracy among all categories:

$$\frac{\sum_m N_{mm}}{\sum_{m,n} N_{mn}} \tag{10}$$

- *ACA* evaluates the recognizable accuracy per category:

$$\frac{1}{\mathcal{C}}\frac{\sum_m N_{mm}}{\sum_n N_{mn}} \tag{11}$$

- *mIoU* is always used to penalizes both over- and under-segmentation for scene labeling, which is defined as the ratio of true positives to the sum of true positive, false positive and false negative, averaged over all object classes:

$$\frac{1}{\mathcal{C}}\frac{\sum_m N_{mm}}{\sum_n N_{mn} + \sum_n N_{nm} - N_{mm}} \tag{12}$$

5.1.4. Implemental details

We use the same split setting [35] for MSRA and LHI datasets that randomly splits all images into three sets: 45% for training, 10% for validation and 45% for testing. While for SIFT flow dataset, we use the evaluation procedure introduced in [21]: 2488 images used for training and 200 images used for testing. Finally, for PASCAL VOC segmentation dataset, we train our contextual fusion model on training images and test on validation images as done in [13,15]. In order to reduce the effect to the performance of randomly selecting training images, we evaluate our system using 10-fold cross validation on MSRA and LHI dataset, where each cross validation has different training and testing images with the same split setting. The training and testing processes are performed on a 4-core i5 personal computer with 2.6 GHz CPU and 16 GB memory.

To construct our FSG, we are required to compute the spatial relationship of super-pixel r_i and r_j. Let $\mathcal{O} = \frac{|r_i \cap r_j|}{|r_i|}$ be the overlap ratio between r_i and r_j. If $\mathcal{O}$ is larger than a predefined threshold η (set as 0.95 in our experience), the associated parent vertex becomes one element of $\mathcal{S}_i$. On the other hand, to train the classifiers $p(Z_j | v_j^m)$, we need to assign ground truth to the automatically created super-pixels. If nearly all (at least 90% by area) of the pixels within a super-pixel have the same ground truth label, the super-pixel is assigned that same label. Otherwise, the super-pixel is labeled as "mixed", which is not used in training process of $p(Z_j | v_j^m)$.

5.2. Results and analysis

5.2.1. Quantitative results

Tables 1 and 2 show our results on MSRA and LHI datasets, and compare them with related works. The results clearly demonstrate that our approach outperforms other state-of-the-art methods, including the approaches based on CRFs and hierarchical models to capture long-range contextual information. Method of [42] also achieves good performance on MSRC dataset, and method of [23] is the fastest in training process. They are, however, at the price of several seconds to label one test image.

Table 1
Performance on MSRC 21-class [35] in terms of recognition accuracy and efficiency. Training times are for the whole training set, test times are per image.

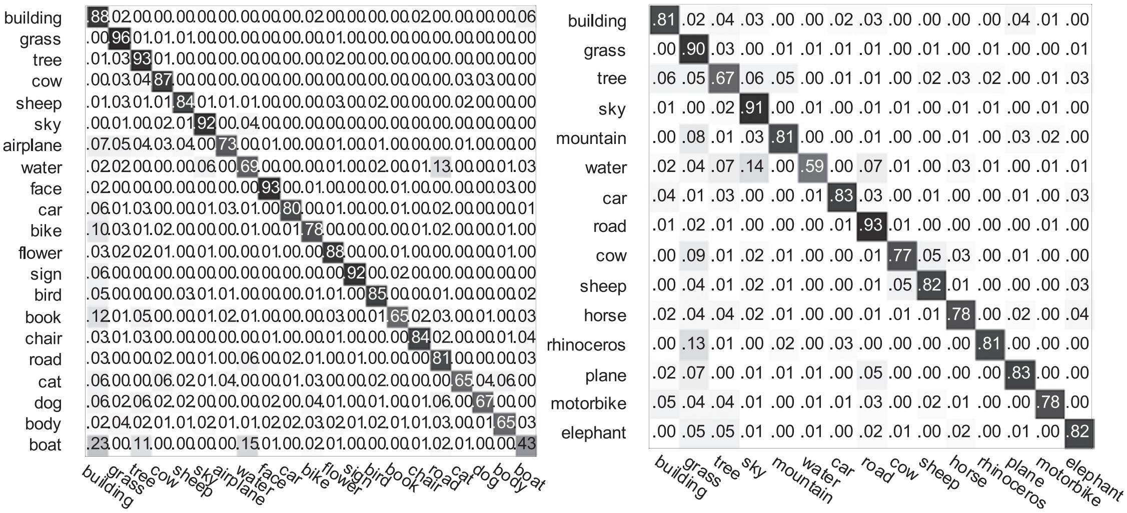
Methods	MSRC 21-class dataset [35]				
	ACA(%)	GPA(%)	mIoU(%)	Training (h)	Testing (s)
Ours	**79.4**	87.3	**47.1**	6.8	0.68
HCRF [42]	78.2	86.8	N/A	5.4	16.8
FCRF [59]	78.3	85.5	N/A	3.4	**0.2**
DHM [23]	76.4	81.7	46.2	**0.6**	8.4
RL [37]	64.3	76.5	N/A	5.5	5.7
TB [35]	57.7	72.2	40.6	6.3	5.4
FCNN [15]	77.9	**91.4**	46.7	N/A	0.2
DHP [14]	77.6	88.6	42.3	N/A	4
HDL [21]	74.6	80.4	45.5	5.8	0.7
SHL [26]	71.2	77.9	N/A	9.1	12
RA [30]	67.3	75.4	N/A	6.8	2.86

Table 2
Performance on LHI 15-class [57] in terms of recognition accuracy and efficiency. Training times are for the whole training set, test times are per image.

Methods	LHI 15-class dataset [57]				
	ACA(%)	GPA(%)	mIoU(%)	Training (h)	Testing (s)
Ours	**80.1**	85.2	**50.9**	2.4	0.63
FCRF [59]	74.6	82.4	N/A	1.9	0.26
DHM [23]	78.2	81.3	47.7	**0.3**	6.8
HCRF [42]	72.5	78.3	N/A	2.7	16.6
RL [37]	64.4	71.6	N/A	2.1	5.1
TB [35]	62.7	69.3	43.2	3.8	5.2
FCNN [15]	79.5	**93.9**	48.8	N/A	**0.24**
DHP [14]	77.1	88.8	41.4	N/A	3.7
HDL [21]	69.3	77.9	46.3	2.5	0.67
RA [30]	63.6	71.8	N/A	3.7	2.88
SHL [26]	61.8	70.7	N/A	4.4	15

Fig. 4 illustrates the confusion matrices obtained by applying our approach on MSRC 21-class and LHI 15-class datasets, in which accuracy values are computed as the image pixels assigned to the correct class labels. The results are about 17 and 12 times better than randomly choosing semantic labels for each pixel on two datasets. We can see that some categories exhibit large errors, e.g., "water" mislabeled as "sky", "book" incorrectly recognized as "building", especially the categories of "boat", "cat", and "dog" on MSRA dataset, which is probably due to their extremely inter-class color/texture similarities, or relative small training samples.

We then demonstrate that our method scales nicely when augmenting the number of images and classes on SIFT flow [8] and PASCAL VOC segmentation datasets [58] in Tables 3 and 4, respectively. Results from Tables 1–4 demonstrate the superior performance of the proposed approach in comparison to the state-of-the-art methods. A remarkable fact is that our approach also outperforms the recent CNN-based methods such as FCNN [15] and DHP [14], and achieves comparable results to those latest CNN-CRF based approaches such as DCNFCC [60] and CRFasRNN [61]. This superior performance can be attributed to our FSG representation, combined with contextual fusion model, which allows us to efficiently capture the contextual relationships within different scales. Moreover, our approach is also computationally efficient. Establishing flexible graph-structure representation by computing the spatial relationship of super-pixels allows us to



	building	grass	tree	cow	sheep	sky	airplane	water	face	car	bike	flower	sign	bird	book	chair	road	cat	dog	body	boat
building	.88	.02	.00	.00	.00	.00	.00	.00	.00	.00	.02	.00	.00	.00	.00	.00	.02	.00	.00	.00	.06
grass	.00	.96	.01	.01	.01	.00	.00	.00	.00	.00	.00	.01	.00	.00	.00	.00	.00	.00	.00	.00	.00
tree	.01	.03	.93	.01	.00	.00	.00	.00	.00	.00	.00	.02	.00	.00	.00	.00	.00	.00	.00	.00	.00
cow	.00	.03	.04	.87	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.03	.03	.00	.00
sheep	.01	.03	.01	.01	.84	.01	.01	.01	.00	.00	.00	.03	.00	.02	.00	.00	.00	.02	.00	.00	.00
sky	.00	.01	.00	.02	.01	.92	.00	.04	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
airplane	.07	.05	.04	.03	.04	.00	.73	.00	.01	.00	.00	.01	.00	.01	.00	.00	.00	.01	.00	.00	.00
water	.02	.02	.00	.00	.00	.06	.00	.69	.00	.00	.00	.01	.00	.02	.00	.01	.13	.00	.00	.01	.03
face	.02	.00	.00	.00	.00	.00	.00	.00	.93	.00	.01	.00	.00	.00	.01	.00	.00	.00	.00	.03	.00
car	.06	.01	.03	.00	.00	.01	.03	.01	.00	.80	.00	.01	.00	.00	.01	.00	.02	.00	.00	.00	.01
bike	.10	.03	.01	.02	.00	.00	.00	.01	.00	.01	.78	.00	.00	.00	.00	.01	.02	.00	.00	.01	.00
flower	.03	.02	.02	.01	.00	.01	.00	.00	.01	.00	.00	.88	.00	.00	.01	.00	.00	.00	.00	.01	.00
sign	.06	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.92	.00	.02	.00	.00	.00	.00	.00	.00
bird	.05	.00	.00	.00	.03	.01	.01	.00	.00	.00	.01	.01	.00	.85	.00	.00	.01	.00	.00	.00	.02
book	.12	.01	.05	.00	.00	.01	.02	.01	.00	.00	.00	.03	.00	.01	.65	.02	.03	.00	.01	.00	.03
chair	.03	.01	.03	.00	.00	.00	.00	.00	.01	.00	.00	.01	.00	.00	.00	.84	.02	.00	.00	.01	.04
road	.03	.00	.00	.02	.00	.01	.00	.06	.00	.02	.01	.00	.01	.00	.00	.00	.81	.00	.00	.00	.03
cat	.06	.00	.00	.06	.02	.01	.04	.00	.00	.01	.03	.00	.00	.02	.00	.00	.00	.65	.04	.06	.00
dog	.06	.02	.06	.02	.02	.00	.00	.00	.02	.00	.04	.01	.00	.01	.00	.01	.06	.00	.67	.00	.00
body	.02	.04	.02	.01	.01	.02	.01	.01	.02	.02	.03	.02	.01	.02	.01	.01	.03	.00	.01	.65	.03
boat	.23	.00	.11	.00	.00	.00	.00	.15	.01	.00	.02	.01	.00	.00	.00	.01	.02	.01	.00	.00	.43

	building	grass	tree	sky	mountain	water	car	road	cow	sheep	horse	rhinoceros	plane	motorbike	elephant
building	.81	.02	.04	.03	.00	.00	.02	.03	.00	.00	.00	.00	.04	.01	.00
grass	.00	.90	.03	.00	.01	.01	.01	.01	.00	.01	.00	.01	.00	.00	.01
tree	.06	.05	.67	.06	.05	.00	.01	.01	.00	.02	.03	.02	.00	.01	.03
sky	.01	.00	.02	.91	.00	.01	.00	.01	.01	.00	.01	.01	.00	.01	.00
mountain	.00	.08	.01	.03	.81	.00	.00	.01	.00	.00	.01	.00	.03	.02	.00
water	.02	.04	.07	.14	.00	.59	.00	.07	.01	.00	.03	.01	.00	.01	.01
car	.04	.01	.03	.00	.00	.01	.83	.03	.00	.01	.00	.00	.01	.00	.03
road	.01	.02	.01	.00	.00	.00	.01	.93	.01	.00	.00	.00	.00	.01	.00
cow	.00	.09	.01	.02	.01	.00	.00	.01	.77	.05	.03	.00	.01	.00	.00
sheep	.00	.04	.01	.02	.01	.00	.00	.01	.05	.82	.01	.00	.00	.00	.03
horse	.02	.04	.04	.02	.01	.00	.00	.01	.01	.01	.78	.00	.02	.00	.04
rhinoceros	.00	.13	.01	.00	.02	.00	.03	.00	.00	.00	.00	.81	.00	.00	.00
plane	.02	.07	.00	.01	.01	.01	.00	.05	.00	.00	.00	.00	.83	.00	.00
motorbike	.05	.04	.04	.01	.00	.01	.01	.03	.00	.02	.01	.00	.00	.78	.00
elephant	.00	.05	.05	.01	.00	.01	.00	.02	.01	.00	.02	.00	.01	.00	.82

Fig. 4. Confusion matrices of our method evaluated on MSRC 21-class dataset (left panel), and LHI 15-class dataset (right panel). The average pixel-wise accuracy is 79.4% and 80.1%, respectively.

Table 3
Performance on SIFT flow [8] in terms of recognition accuracy and efficiency. Training times are for the whole training set, test times are per image.

Methods	SIFT flow 33-class dataset [8]				
	ACA(%)	GPA(%)	mIoU(%)	Training (h)	Testing (s)
Ours	**53.3**	73.9	**40.8**	2.9	0.64
FCRF [59]	51.6	74.4	N/A	1.9	0.26
HCRF [42]	49.1	72.8	N/A	3.2	16.5
DHM [23]	48.4	71.2	37.6	**0.4**	8.9
RL [37]	46.5	70.5	N/A	2.5	5.1
TB [35]	45.7	64.7	31.5	4	5
FCNN [15]	51.7	**85.2**	39.5	N/A	**0.17**
DHP [14]	52.8	75.5	30.2	N/A	1.1
HDL [21]	50.8	72.3	34.7	3.0	0.72
RA [30]	48.6	68.7	N/A	4.4	0.83
SHL [26]	47.8	67.1	N/A	5.2	15

Table 4
Performance on PASCAL VOC segmentation dataset [58] in terms of recognition accuracy and efficiency. Training times are for the whole training set, test times are per image.

Methods	PASCAL VOC segmentation dataset [58]				
	ACA(%)	GPA(%)	mIoU(%)	Training (h)	Testing (s)
Ours	**77.1**	82.4	64.4	2.9	0.61
FCRF [59]	30.2	55.3	53.4	2.5	0.5
TB [35]	34.8	52.1	31.5	2.7	4.7
DCNFCC [60]	–	–	65.21	N/A	N/A
CRFasRNN [61]	–	–	**69.6**	N/A	N/A
FCNN [15]	75.9	**90.3**	62.7	N/A	**0.17**
HDL [21]	72.5	83.3%	60.6	1.7	0.76

label a scene with resolution 320 × 240 in less than 1 s, which is very beneficial for labeling video sequence or massive image datasets.

5.2.2. Qualitative results

Some example results of simultaneous recognition and segmentation on four datasets are shown in Fig. 5. Each example shows both the original image and the color coded output labeling. It is evident that our method can handle large appearance variations of object classes. Except the boundary regions that exhibit relative higher confusion, nearly all super-pixels are correctly classified. In the last column, we show some examples in which the labeling results are not good enough (e.g., "water" and "mountain" incorrectly labeled as "tree", "dog" is confused with "cat", the missed foreground "person" and "train"), however, the foreground objects (e.g., "bird" and "rhino") still achieve good segmentation and recognition on MSRC 21-class and LHI 15-class datasets.

5.3. Other aspects

5.3.1. Analysis of segmentations

In our experiment, two factors directly affecting the performance are the numbers of leaf vertices $|\mathcal{V}_L|$ and of levels of segmentations $\mathcal{L}$. The first one controls the granularity in the finest segmentation of our FSG, and the second one determines how many scales of contextual cues are sufficient for scene labeling. We evaluate the accuracy of our system on MSRC, LHI and PASCAL VOC segmentation datasets by changing the values of these two parameters, using all the available cues (texture, color, shape, size, location and label hypotheses predictions). The selection of these two parameters illustrates the trade-off between computational efficiency and the recognition precision.

We first evaluate the effect of the number of leaf vertex on average accuracy given the FSG. In practice, we repeat our experiments to produce different numbers of leaf vertices, and then produce the FSG based on the establishing criteria described in Section 3. The left panel of Fig. 7 shows the plot of average accuracy for increased number of leaf vertices on the test set. The accuracy of our method peaks at approximately 200 and 250 leaf vertices for MSRC and LHI datasets, respectively, and any refinement to these parameters will result in slightly decrease of performance.

We also measure the effect of changing the number of segmentation levels $\mathcal{L}$. Our implementation uses different levels (from 1 to 15) of segmentations. In the right panel of Fig. 7, we display the global accuracy along with the increasing number of segmentation levels. In these experiments, we use the same classifiers trained based on FSG but generate new sets of segmentations for testing. As can be seen, more levels of segmentations result in higher accuracy, which shows that using multi-scale context plays the most critical role in object classification. It is observed that the highest performance is achieved when $\mathcal{L} = 10$ for MSRC and LHI datasets, and $\mathcal{L} = 13$ for PASCAL VOC segmentation dataset, which indicates PASCAL dataset is more challenging than MSRC and LHI datasets. Furthermore, it demonstrates that a

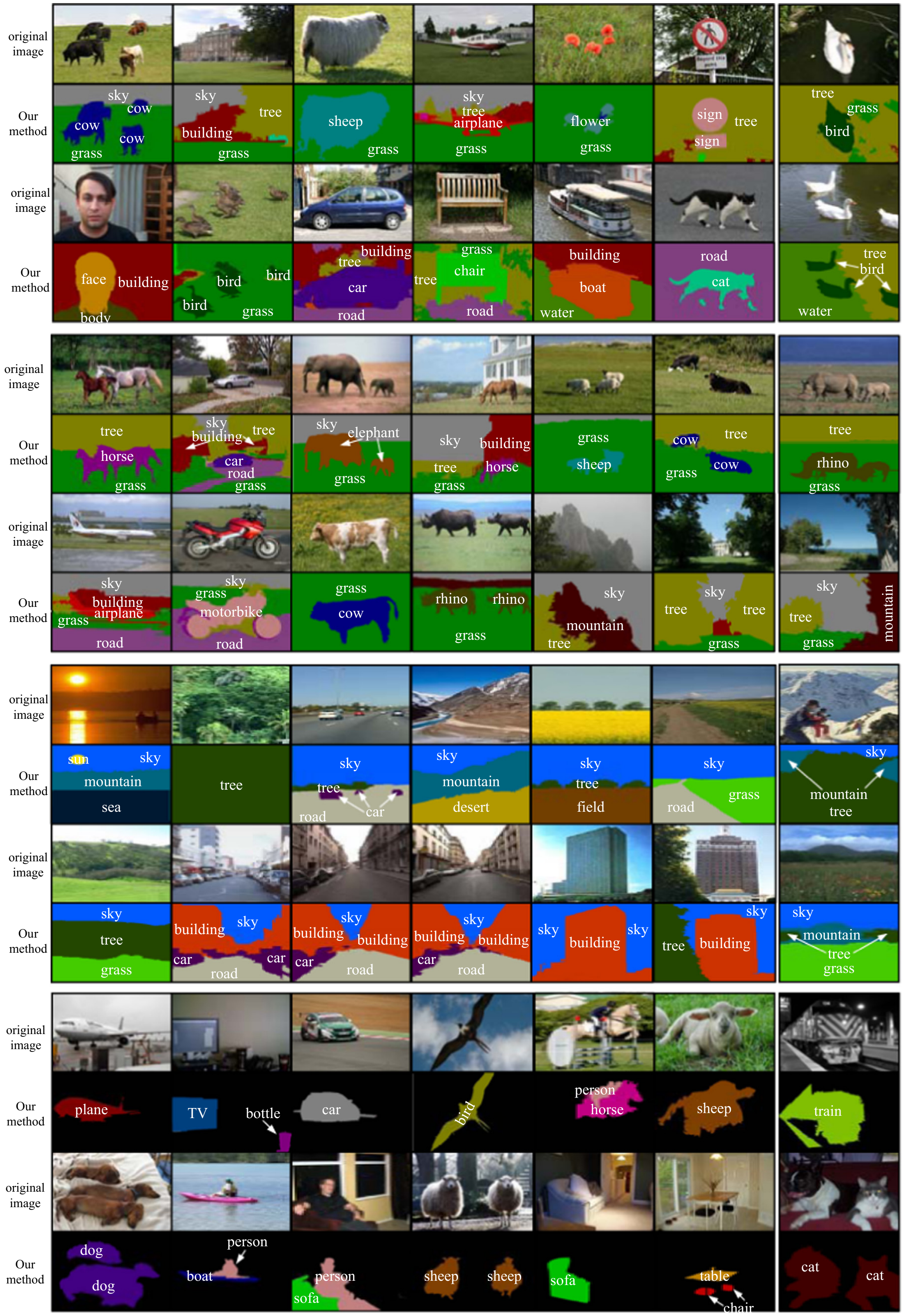


Fig. 5. Some visual examples of our labeling outputs. From top to bottom are results from MSRC 21-class, LHI 15-class, SIFT flow 33-class and PASCAL VOC segmentation datasets, respectively. For clarity, textual labels have also been superimposed on the resulting segmentations, and different color denotes different category (best viewed in color). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

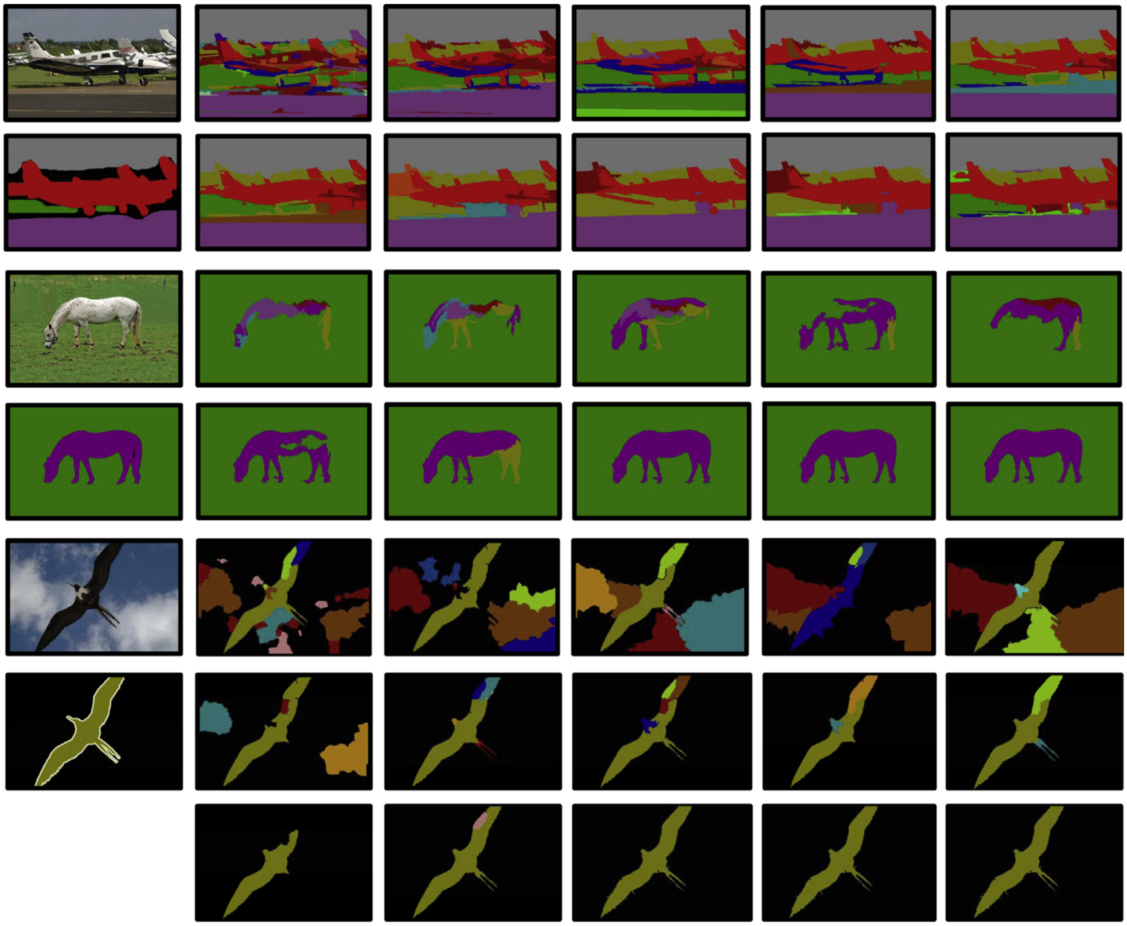


Fig. 6. Examples of object segmentation results for MSRC (upper panel), LHI (middle panel) and PASCAL VOC segmentation (bottom panel) datasets by sequentially inducing multiple scale context cues. The first collum gives the original image and corresponding ground truth. The other images show the labeling results by gradually increasing the level $\mathcal{L}$ of FSG (best viewed in color). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Table 5
Contributions of different cues and their combinations to the performance.

Cues	MSRC [35]			LHI [57]			SIFT flow [8]		
	ACA(%)	GPA(%)	mIoU(%)	ACA(%)	GPA(%)	mIoU(%)	ACA(%)	GPA(%)	mIoU(%)
Txt+Clr+Prd	**78.5**	**85.2**	**46.3**	**79.6**	**84.8**	**49.7**	**51.1**	**73.3**	**40.1**
Txt+Clr	74.7	80.5	45.7	74.9	81.7	48.6	49.3	69.1	39.2
Clr+Prd	74.2	79.2	45.1	74.1	79.4	47.7	48.1	68.2	38.4
Txt+Prd	73.9	76.6	44.9	73.5	78.6	46.5	47.3	67.5	36.8
Txt	70.6	73.4	42.3	72.3	78.2	47.2	48.9	67.8	37.7
Prd	64.4	75.9	35.6	70.0	73.3	43.8	44.7	62.2	37.0
Clr	53.2	65.8	33.2	66.6	69.8	40.3	41.9	56.1	35.1

small number of levels in the segmentation family provides sufficient context for our labeling task. Fig. 6 displays three examples for the three datasets, respectively.

5.3.2. Analysis of cues

We also evaluate the effectiveness of our three main types of cues: color, texture, and label hypotheses predictions as described in Section 4.2. To do this, we train classifiers using different features, and compute the ACA, GCA, and mIoU of all categories over the test images. To be specific, we first only use texture, color and label hypotheses to train our model as baseline. Then different feature combinations are evaluated. In these experiments, we employ FSG representation, using the same segmentation family as were used to report accuracy. The contributions of different cues are listed in Table 5.

Table 5 demonstrates that the combination is superior than individual features. Specifically, using all three types of features gains the best results, i.e., 70%, 81.1% and 45.4% in terms of ACA, GCA and mIoU over three datasets, respectively. On the other hand, combined texture and color, or texture and label hypotheses, or color and label hypotheses cues obtain comparable results. It is also observed that the simple texture feature appears to be surprisingly effective on three datasets, achieving 64.9% ACA, 74.8% GCA, and 42.7% mIoU, respectively. Compared with Tables 1–3, in spite of only using texture cues in our model, it still outperforms some baseline methods.

From Table 5, we can also conclude that although only using individual color cue is ranked at the bottom, it significantly improves the results when combined with other cues. For instance, when combined with texture cues, the performance can be improved by 2.4%, 4.0% and 2.1% in terms of ACA, GPA and mIoU. In the case of combined color and label hypotheses cues, we have

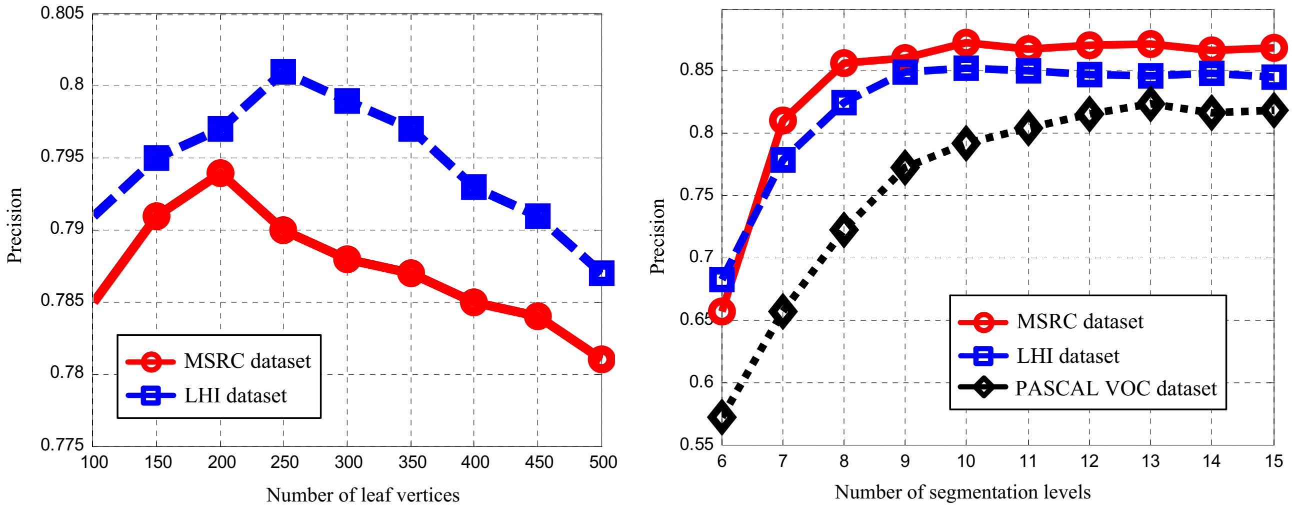


Fig. 7. Effects on performance by number of leaf vertices (left panel), and the levels of segmentation family (right panel).

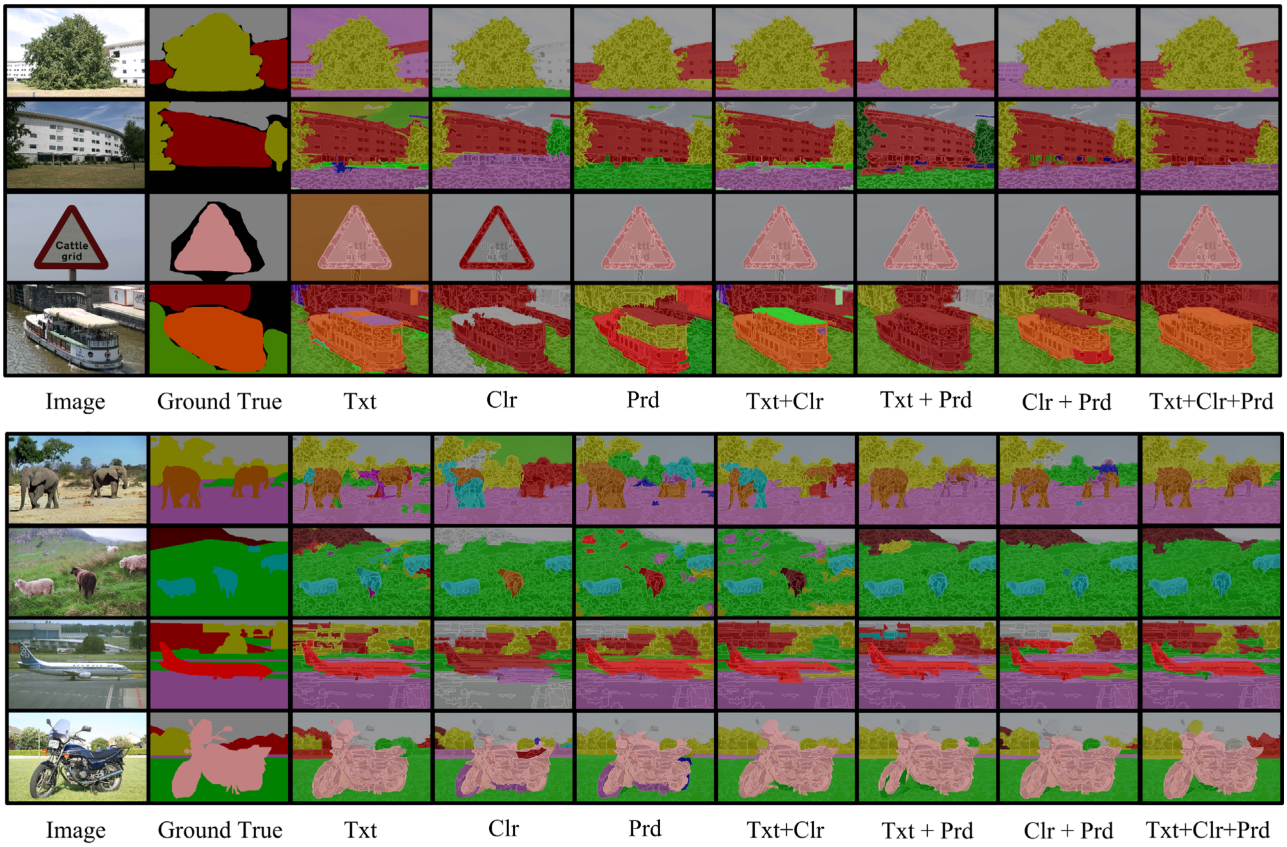


Fig. 8. Labeling results when performing classification based on different cues and their combinations on MSRC (up panel) and LHI (down panel) datasets. In each case, the same segmentation family and FSG are used, and those super-pixels are described with the given type of cues (best viewed in color). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

achieved 65.5% ACA, 75.6% GPA and 43.7% mIoU on three dataset. Similar to color cue, the results reported from Tables 1–5 demonstrate the remaining size, shape and location cues only achieve 1.2%, 1.0%, and 0.9% improvement for ACA, GCA and mIoU, respectively.

Finally, as compared with these low-level cues, the label hypotheses prediction cue by itself seems remarkably effective at discriminating among the classes. When it is induced into other low-level cues, the average performance can be improved by 3.2%, 6.3% and 3.1% in terms of ACA, GPA and mIoU, respectively. Perhaps this high-level feature provides the useful and discriminative information to capture scene-level relationships from different levels of FSG. Fig. 8 shows two examples of visual labeling results on MSRC and LHI datasets, comparing the individual contribution of induced cues and the overall results, respectively. It seems that the color feature is highly effective for some specific classes, such as “sky”, “grass” and “tree”, while not effective for the categories with great color variance, such as “boat”, “elephant” and “airplane”.

6. Conclusion and future work

This paper describes a contextual fusing model using FSG to explore the multi-scale context information for scene labeling problem. The proposed model is trained on fully labeled images in a supervised manner to learn appropriate low-level features and high-level hypotheses to predict pixel labels. Firstly, a FSG is built based on spatial relationships of segmented super-pixels to represent an entire scene. This allows for the integration of cues defined at different scales to contribute to predict region labels. Then, a contextual fusion model, also called integral model, is established to integrate multi-scale visual context for producing consistent recognition and segmentation results. We have evaluated our method on three datasets, and achieved similar or better pixel-wised labeling accuracy with the competing models that employ CRFs or hierarchical models. Additionally, the experimental results also demonstrate that when a wide scale context is considered into our fusion model, the inference process can be greatly reduced.

Despite obtaining impressive results, we believe that even better results can be achieved by automatically learning a flexible structure representation. We are aware of a related work of [21] in this direction. Additionally, we plan to embed deep learning algorithms, such as [14,15], into our multi-scale framework to improve the performance, while retaining high efficiency.

Conflict of interest

None declared.

Acknowledgments

The authors would like to thank the associated editor and all the anonymous reviewers for their valuable comments and insightful suggestions. This work was partly supported by the National Science Foundation (Grant no. IIS-1302164), and the National Natural Science Foundation of China (Grant nos. 61201165, 61271240, 61401228, 61571240, 61502245), and the State Key Development Program of Basic Research of China (Grant no. 2013CB329005), and China Postdoctoral Science Foundation (Grant no. 2015M581841), and Postdoctoral Science Foundation of Jiangsu Province (Grant no. 1501019A).

References

[1] Y. Yang, S. Hallman, D. Ramanan, C. Fowlkes, Layered object detection for multi-class segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 3113–3120.
[2] L. Lin, X. Wang, W. Yang, J. Lai, Discriminatively trained and-or graph models for object shape detection, IEEE Trans. Pattern Anal. Mach. Intell. 37 (5) (2015) 959–972.
[3] J.R. Uijlings, K.E. van de Sande, T. Gevers, A.W. Smeulders, Selective search for object recognition, Int. J. Comput. Vis. 104 (2) (2013) 154–171.
[4] K. Wang, L. Lin, J. Lu, C. Li, K. Shi, Pisa: Pixelwise image saliency by aggregating complementary appearance contrast measures with edge-preserving coherence, IEEE Trans. Image Process. 24 (10) (2015) 3019–3033.
[5] D. Hoiem, A. Efros, M. Hebert, Closing the loop in scene interpretation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
[6] L. Lin, R. Zhang, X. Duan, Adaptive scene category discovery with generative learning and compositional sampling, IEEE Trans. Circuits Syst. Video Technol. 25 (2) (2015) 251–260.
[7] J. Lerouge, R. Herault, C. Chatelain, F. Jardin, R. Modzelewski, Ioda: an input/output deep architecture for image labeling, Pattern Recognit. 48 (9) (2015) 2847–2858.
[8] C. Liu, J. Yuen, A. Torralba, Nonparametric scene parsing: label transfer via dense scene alignment, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1972–1979.
[9] C. Liu, J. Yuen, A. Torralba, Nonparametric scene parsing via label transfer, IEEE Trans. Pattern Anal. Mach. Intell. 33 (12) (2011) 2368–2382.
[10] J. Ma, H. Zhou, J. Zhao, J. Tian, Robust feature matching for remote sensing image registration via locally linear transforming, IEEE Trans. Geosci. Remote Sens. 53 (12) (2015) 6469–6481.
[11] J. Ma, J. Zhao, A.L. Yuille, Non-rigid point set registration by preserving global and local structures, IEEE Trans. Image Process. 25 (1) (2016) 53–64.
[12] B. Shuai, G. Wang, Z. Zuo, B. Wang, L. Zhao, Integrating parametric and non-parametric models for scene labeling, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 4249–4258.
[13] A. Roy, S. Todorovic, Scene labeling using beam search under mutex constraints, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1178–1185.
[14] A. Sharma, O. Tuzel, D.W. Jacobs, Deep hierarchical parsing for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 530–538.
[15] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–8.
[16] F. Li, J. Carreira, G. Lebanon, C. Sminchisescu, Composite statistical inference for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 4249–4258.
[17] Y. Liu, J. Liu, Z. Li, J. Tang, H. Lu, Weakly-supervised dual clustering for image semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2075–2082.
[18] E. Borenstein, S. Ullman, Combined top-down/bottom-up segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 30 (12) (2008) 2109–2125.
[19] J. Xu, A.G. Schwing, R. Urtasun, Learning to segment under various forms of weak supervision, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3781–3790.
[20] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, A. Yuille, The role of context for object detection and semantic segmentation in the wild, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1–8.
[21] C. Farabet, C. Couprie, L. Najman, Y. LeCun, Learning hierarchical features for scene labeling, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1915–1929.
[22] J. Yang, B. Price, S. Cohen, M.-H. Yang, Context driven scene parsing with attention to rare classes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 3294–3301.
[23] Q. Zhou, J. Zhu, W. Liu, Learning dynamic hybrid Markov random field for image labeling, IEEE Trans. Image Process. 22 (6) (2013) 2219–2232.
[24] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Trans. Pattern Anal. Mach. Intell. 24 (24) (2002) 509–522.
[25] S. Geman, D. Geman, Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images, IEEE Trans. Pattern Anal. Mach. Intell. 6 (6) (1984) 721–741.
[26] D. Munoz, J.A. Bagnell, M. Hebert, Stacked hierarchical labeling, in: Proceedings of the Europe Conference on Computer Vision, 2010, pp. 57–70.
[27] T. Mensink, J. Verbeek, G. Csurka, Tree-structured CRF models for interactive image labeling, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2) (2013) 476–489.
[28] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE Trans. Pattern Anal. Mach. Intell. 24 (5) (2002) 603–619.
[29] P.F. Felzenszwalb, D.P. Huttenlocher, Efficient graph-based image segmentation, Int. J. Comput. Vis. 59 (2) (2004) 167–181.
[30] J.J. Lim, P. Arbeláez, C. Gu, J. Malik, Context by region ancestry, in: Proceedings of the IEEE Conference on Computer Vision, 2009, pp. 1978–1985.
[31] C. Pantofaru, C. Schmid, M. Hebert, Object recognition by integrating multiple image segmentations, in: Proceedings of the Europe Conference on Computer Vision, 2008, pp. 481–494.
[32] Q. Zhou, C. Yan, Y. Zhu, X. Bai, W. Liu, Image labeling by multiple segmentation, in: Proceedings of the IEEE Conference on Image Processing, 2011, pp. 3129–3132.
[33] Z.W. Tu, X. Bai, Auto-context and its application to high-level vision tasks and 3d brain image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 32 (10) (2010) 1744–1757.
[34] S. Kumar, M. Hebert, Discriminative random fields, Int. J. Comput. Vis. 68 (2) (2006) 179–201.
[35] J. Shotton, J.M. Winn, C. Rother, A. Criminisi, TextonBoost for image understanding: multi-class object recognition and segmentation by jointly modeling texture, layout, and context, Int. J. Comput. Vis. 81 (1) (2009) 2–23.
[36] C. Galleguillos, A. Rabinovich, S. Belongie, Object categorization using co-occurrence, location and appearance, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
[37] S. Gould, J. Rodgers, D. Cohen, G. Elidan, D. Koller, Multi-class segmentation with relative location prior, Int. J. Comput. Vis. 80 (3) (2008) 1239–1253.
[38] P. Yadollahpour, D. Batra, G. Shakhnarovich, Discriminative re-ranking of diverse segmentations, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 1923–1930.
[39] L. Ladicky, P. Sturgess, K. Alahari, C. Russell, P. Torr, What, where and how many? Combining object detectors and CRFs, in: Proceedings of the Europe Conference on Computer Vision, 2010, pp. 424–437.
[40] S. Gould, T. Gao, D. Koller, Region-based segmentation and object detection, in: Proceedings of the Conference on Neural Information Processing Systems, 2009, pp. 655–663.
[41] J. Yao, S. Fidler, R. Urtasun, Describing the scene as a whole: joint object detection, scene classification and semantic segmentation, in: Proceedings of

the IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 702–709.

[42] L. Ladicky, C. Russell, P. Kohli, P.H.S. Torr, Associative hierarchical random fields, IEEE Trans. Pattern Anal. Mach. Intell. 36 (6) (2014) 1056–1077.

[43] V. Lempitsky, A. Vedaldi, A. Zisserman, Pylon model for semantic segmentation, in: Proceedings of the Conference on Neural Information Processing Systems, 2011.

[44] J. Zhu, T. Wu, J. Zhu, X. Yang, W. Zhang, Learning reconfigurable scene representation by tangram model, in: Proceedings of the IEEE Workshop on the Applications of Computer Vision, 2012, pp. 449–456.

[45] L. Zhu, Y. Chen, Y. Lin, C. Lin, A. Yuille, Recursive segmentation and recognition templates for image parsing, IEEE Trans. Pattern Anal. Mach. Intell. 34 (2) (2012) 359–371.

[46] H. Caesar, J. Uijlings, V. Ferrari, Joint calibration for semantic segmentation, arXiv preprint arXiv:1507.01581.

[47] F. Liu, G. Lin, C. Shen, CRF learning with CNN features for image segmentation, Pattern Recognit. 48 (10) (2015) 2983–2992.

[48] R. Mottaghi, S. Fidler, A. Yuille, R. Urtasun, D. Parikh, Human–machine CRFs for identifying bottlenecks in scene understanding, Trans. Pattern Anal. Mach. Intell. IEEE Trans. Pattern Anal. Mach. Intell. 38 (1) (2016) 74–87.

[49] P. Arbelaez, M. Maire, C. Fowlkes, J. Malik, Contour detection and hierarchical image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (5) (2011) 898–916.

[50] L. Najman, M. Schmitt, Geodesic saliency of watershed contours and hierarchical segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 18 (12) (1996) 1163–1173.

[51] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.

[52] J. Friedman, T. Hastie, R. Tibshirani, Additive logistic regression: a statistical view of boosting, Ann. Stat. 28 (2) (2000) 337–407.

[53] K. Barnard, P. Duygulu, D. Forsyth, N. De Freitas, D.M. Blei, M.I. Jordan, Matching words and pictures, J. Mach. Learn. Res. 3 (2003) 1107–1135.

[54] J. Zhang, M. Marszałek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: a comprehensive study, Int. J. Comput. Vis. 73 (2) (2007) 213–238.

[55] M.J. Choi, J.J. Lim, A. Torralba, A.S. Willsky, Exploiting hierarchical context on a large database of object categories, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 129–136.

[56] A. Oliva, A. Torralba, The role of context in object recognition, Trends Cogn. Sci. 11 (12) (2007) 520–527.

[57] B. Yao, X. Yang, S.C. Zhu, Introduction to a large scale general purpose groundtruth dataset: methodology, annotation tool, and benchmarks, in: Proceedings of the Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition, 2007.

[58] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, A. Zisserman, The pascal visual object classes (VOC) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.

[59] P. Krahenbuhl, V. Koltun, Efficient inference in fully connected CRFs with gaussian edge potentials, in: Proceedings of the Conference on Neural Information Processing Systems, 2011, pp. 109–117.

[60] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Semantic image segmentation with deep convolutional nets and fully connected CRFs, arXiv preprint arXiv:1412.7062.

[61] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, P. H. S. Torr, Conditional random fields as recurrent neural networks, in: Proceedings of the IEEE Conference on Computer Vision, 2015, pp. 1529–1537.

Quan Zhou received Ph.D. degree in electronics and information engineering from Huazhong University of Science and Technology (HUST), Wuhan, China in 2013. Now he is an assistant professor in the college of Telecommunications and Information engineering at Nanjing University of Posts and Telecommunications. His research interests include computer vision and pattern recognition.

Baoyu Zheng received M.S. degree in College of telecommunications and information engineering from Nanjing University of Posts and Telecommunications (NJUPT), Nanjing, China. He is now a full professor in the college of Telecommunications and Information engineering at NJUPT. His research interests include multiple media signal processing. He is a senior number of IEEE.

Weiping Zhu received Ph.D. degree from Southeast University, Nanjing, China in 1991. He is a professor in Department of Electrical and Computer Engineering, Concordia University, Montreal, Canada. His research interests include multiple media signal processing. He was served as an Associate Editor of the IEEE Transactions on Circuits and Systems Part I and Part II, and a Technical Program Committee member for a large number of IEEE sponsored conferences.

Longin Jan Latecki received the Ph.D. degree in computer science from Hamburg University, Germany, in 1992. He is a professor of computer science at Temple University, Philadelphia. His main research interests include shape representation and similarity, object detection and recognition in images, robot perception, data mining, and digital geometry. He is an editorial board member of Pattern Recognition and the International Journal of Mathematical Imaging.