Superpixel Coherency and Uncertainty Models for Semantic Segmentation

SeungRyul Baek, Taegyu Lim, Yong Seok Heo, Sungbum Park, Hantak Kwak and Woosung Shim
Multimedia R&D Team, DMC R&D Center, Samsung Electronics
129, Maetan-dong, Yeongtong-gu, Suwon-si, Gyeonggi-do, South Korea
{sryul.baek,tgx.lim,yo.heo,sb0916.park,hantak.kwak,sim1}@samsung.com

Abstract

We present an efficient semantic segmentation algorithm based on contextual information which is constructed using superpixel-level cues. Although several semantic segmentation algorithms employing superpixel-level cues have been proposed and significant technical advances have been achieved recently, these algorithms still suffer from inaccurate superpixel estimation, recognition failure, time complexity and so on. To address problems, we propose novel superpixel coherency and uncertainty models which measure coherency of superpixel regions and uncertainty of the superpixel-wise preference, respectively. Also, we incorporate two superpixel models in an efficient inference method for the conditional random field (CRF) model. We evaluate the proposed algorithm based on MSRC and PASCAL datasets, and compare it with state-of-the-art algorithms quantitatively and qualitatively. We conclude that the proposed algorithm outperforms previous algorithms in terms of accuracy with reasonable time complexity.

1. Introduction

Semantic segmentation is one of the most challenging research topics in computer vision since various high-level applications such as visual surveillance, human motion analysis, video content understanding and object-based image enhancement often depend on its performance. The main task of semantic segmentation is to automatically recognize predefined labels (*i.e.* human, vehicle, grass, sky and etc.) pixel-wisely and segment regions in an observed image as shown in Fig. 1.

The semantic segmentation has been investigated intensively in recent years and significant performance improvement has been achieved so far [6, 7, 10, 11, 12, 14]. One of the most successful approaches in this field is the energy minimization technique based on conditional random field (CRF) model. At first, Shotton *et al.* [12] proposed *TextonBoost* algorithm which combines object recognition and image segmentation in a single framework. In the method, the pixel-wise potential for object classes is computed based on weak classifiers which are trained with texton features, and object class label for each pixel is determined by optimizing the CRF model considering pixel-wise and pair-wise relations. Recently, CRF-based semantic segmentation algorithms attempt to utilize contextual information by generating coherent image segments (called *superpixel*) to obtain more accurate segmentation results. These approaches assume that pixels constituting a particular superpixel belong to the same object class. Kohli *et al.* [7] employed superpixel-level cues in the higher-order potential by penalizing differently labeled pixels in a superpixel. Ladicky *et al.* [10, 11] proposed hierarchy of superpixels as well as image-level label co-occurrences, and their experiments yielded the state-of-the-art result. Gonfaus *et al.* [6] introduced more expressive constraints called harmony potentials, which model multiple superpixel-wise label preferences and achieved higher accuracy.

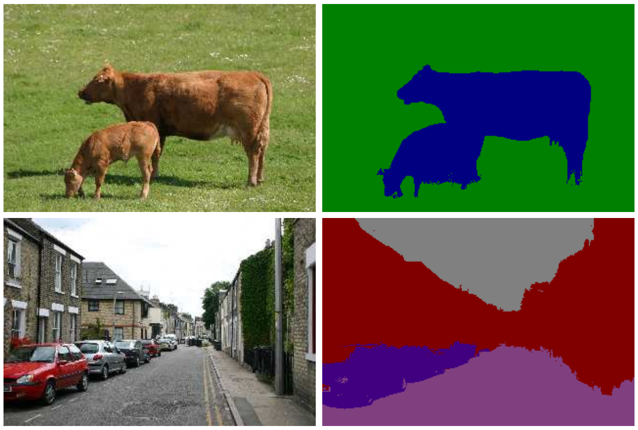


Figure 1. Examples of semantic segmentation. Predefined labels are assigned to each pixel, then images are segmented automatically. Note that these results are obtained from our proposed algorithm. (a) Observed images. (b) Semantic segmetnation results.

As mentioned above, researches on higher-order potentials using superpixel-level cues have been pursued for

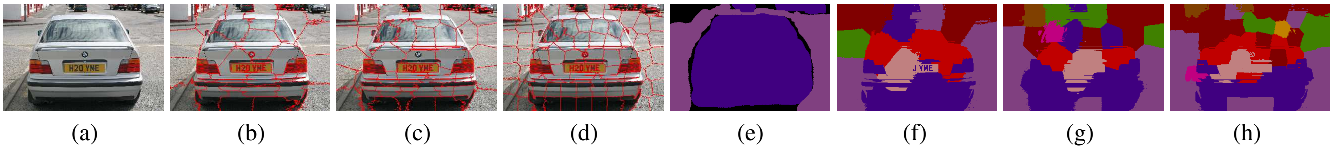


Figure 2. Difficulties in employing superpixel cues. (a) Observed image, (b)-(d) Superpixel sets in coarse-to-fine level, (e) Ground-truth, (f)-(h) Superpixel-wise recognition in each level. These are obtained as in [11]. We can see that portions of superpixel sets fail to follow the accurate boundary of the *car* object and superpixel-wise recognitions are inaccurate for representing the label preference of superpixels.

many years, and have produced many practical algorithms. However, these algorithms have three main limitations. First, as shown in Figs. 2b, 2c and 2d, accurate superpixel generation is very difficult since it frequently faces various challenges such as lighting conditions, occlusion, clutter scene and image noise. Also, typical superpixel generation algorithms [2, 3] employ heuristic parameters (*e.g.* total number, size, threshold and so on.) that make it hard to generate optimal superpixel sets. Second, as shown in Figs. 2f, 2g and 2h, superpixel-wise probabilities for object classes are not sufficient to represent overall label preference of pixels in the superpixel. Lastly, the inference for the CRF model with the higher-order potential suffers from high computational cost.

To overcome such limitations, we focus on improving the higher-order potential of the CRF model and its inference method. Hence, we propose superpixel coherency and uncertainty models. The coherency model alleviates the inaccurate region problem by measuring the coherency of superpixel regions and controlling the effect of superpixel regions based on it. The uncertainty model deals with the ambiguity problem in the superpixel-wise recognition by measuring the certainty of the superpixel-wise recognition and controlling the effect of it. To reduce the computational complexity, we also propose a inference rule based on mean field approximation algorithm [14] which is known to be highly efficient and utilized to obtain the most probable segments based on pixel-wise, pair-wise and higher-order potentials. Penalties calculated from coherency and uncertainty models are integrated in the inference method for the higher-order potential of the CRF model.

Our semantic segmentation algorithm has the following contributions and characteristics, which are efficient in overcoming limitations observed in other methods:

- We propose superpixel coherency and uncertainty models which are constructed based on dissimilarities of codewords and superpixel-wise probability distributions with respect to object classes, respectively.

- We offer an effective way to integrate our superpixel models in the inference method of the CRF-based semantic segmentation algorithm.

- We obtain an efficient semantic segmentation algorithm which performs well in terms of segmentation accuracy with reasonable computational cost.

The paper is organized as follows. Section 2 provides an overview of technical terms and overall algorithms. Section 3 discusses the proposed higher-order potential with superpixel models in detail. Inference for proposed algorithm is presented in Section 4. Section 5 evaluates the performance of the proposed algorithm and illustrates experimental results.

2. Overview of Algorithm

Our goal is to perform both recognition and segmentation in an observed image. For this purpose, we estimate the MAP (*Maximum a Posteriori*) solution over object class labels of each pixel given the image, which is formulated as

$$\mathbf{x}^* = \arg\max_{\mathbf{x}\in\mathcal{L}^N} \mathcal{P}(\mathbf{x}|\mathbf{I}), \tag{1}$$

where $\mathbf{x}$ denotes a configuration of object class labels and $\mathbf{I}$ represents an observed image whose N pixels are mapped to a random field $\mathbf{X} \triangleq \{X_1, ..., X_N\}$. The configuration denoted by $\mathbf{x} \triangleq \{x_1, \ldots, x_N\} \in \mathcal{L}^N$ is composed of a set of object class labels, where x_i denotes a possible assignment of label to random variable X_i from the label set $\mathcal{L} \triangleq \{\ell_1, ...\ell_L\}$ with L predefined object classes. To obtain $\mathcal{P}(\mathbf{x}|\mathbf{I})$, off-line learning and processing phases are excuted as in Fig. 3. In the off-line learning phase, a codebook and a codeword dissimilarity matrix are constructed. In the processing phase, the first step is to generate codewords and superpixels. The second step is to recognize object classes pixel-wisely and superpixel-wisely. In the third step, responses for superpixel coherency and uncertainty models are calculated and various cues are combined in the CRF model, which are followed by inference step to obtain the most probable semantic segmentation.

Codebook learning and codeword generation To capture a significant proportion of the complex real image, we represent features as codewords based on the Bag-of-Words (BoW) technique [13]. We extract multiple features such as SIFT, color SIFT, local binary pattern (LBP), and Texton [12] for each pixel. They are resistant to occlusions, geometric deformations and illumination changes. In the off-line learning phase, feature vectors are extracted from

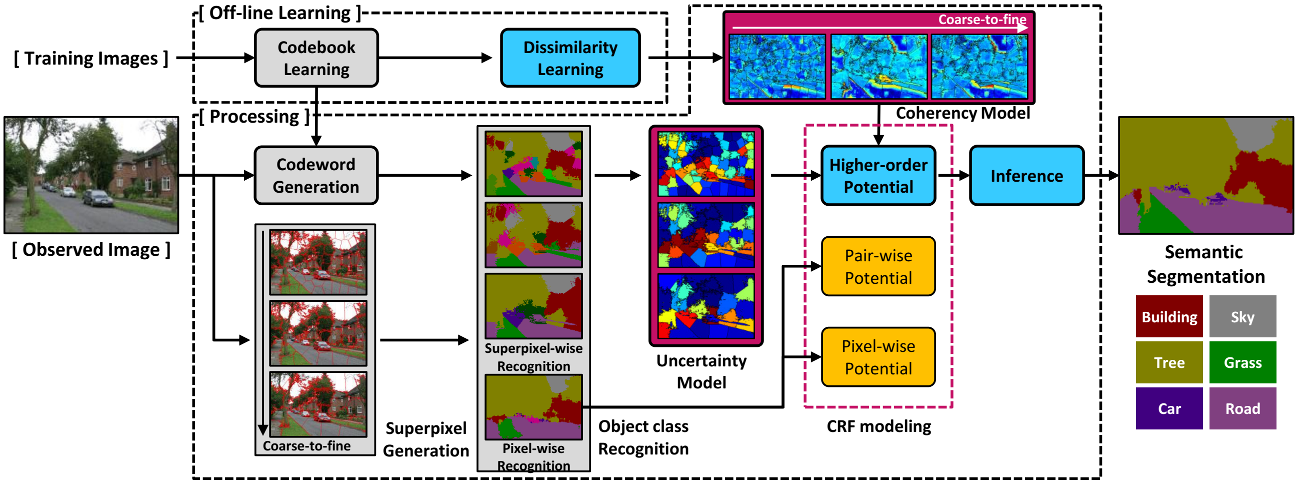


Figure 3. Overview of our algorithm. Codebook and codeword dissimilarity matrix are obtained in the off-line learning phase. In the processing phase, the responses of two superpixel models are calculated and the most probable labels for semantic segments are obtained by the inference algorithm.

whole training images and are clustered by well-defined clustering algorithms in each feature space. Each cluster is mapped to codewords and collections of codewords are grouped into a codebook. Note that, in the processing phase, each feature vector is mapped to the nearest codeword index based on the Euclidean distance. Then a set of codeword for the i-th pixel $\mathbf{v}_i \triangleq \{v_i^k | k \in [1,n], v_i^k \in [1, V_k]\}$ is generated to represent the appearance, where n and V_k denote number of features and number of codewords in the k-th codebook respectively.

Superpixel generation Superpixels are obtained by unsupervised clustering algorithms to generate coherent regions by merging pixels based on the similarity of low-level features such as color and location. Similarly to [11], we used the K-means and mean-shift clustering algorithms by varying the parameter K and the kernel size, in the coarse-to-fine manner. In our experiments, 6 and 10 levels of coarse-to-fine superpixels are generated for MSRC-21 and PASCAL VOC-2010 datasets, respectively.

Object class recognition To obtain a probability distribution for object classes pixel-wisely and superpixel-wisely, we employ *TextonBoost* algorithm [12]. Weak classifiers for pixel-wise recognition are trained on the appearance of each pixel v_i^k in the k-th feature space. For superpixel-wise recognition, feature vector is defined as a normalized histogram of v_i^k for $i \in c$ and weak classifiers are trained based on it. In both recognition tasks, responses of weak classifiers for object classes are summed and normalized into a probability distribution. They are used to model potentials of the CRF model.

Conditional random field model The posterior probability over labels given an image $\mathbf{I}$ is defined as $\mathcal{P}(\mathbf{x}|\mathbf{I}) = \frac{1}{Z}\exp(-E(\mathbf{x}|\mathbf{I}))$, where Z is the normalizing constant and $E(\mathbf{x}|\mathbf{I})$ is the Gibbs energy, which is defined on the set of cliques C and expressed as $E(\mathbf{x}|\mathbf{I}) = \sum_{c\in C}\psi_c(\mathbf{x}_c)$. A clique $c \in C$ is defined as a set of random variables which are conditionally dependent on each other. The set of cliques C is composed of pixel-wise, pair-wise and higher-order cliques, which are defined as cliques of size one, two and three or beyond, respectively. Then, the Gibbs energy for the CRF model can be re-expressed as follows:

$$E(\mathbf{x}|\mathbf{I}) = \underbrace{\sum_{i\in\mathcal{V}}\psi_i(x_i)}_{\mathrm{Pixel-wise}} + \underbrace{\sum_{(i,j)\in\mathcal{E}}\psi_{ij}(x_i,x_j)}_{\mathrm{Pair-wise}} + \underbrace{\sum_{c\in\mathcal{S}}\psi_c^h(\mathbf{x}_c)}_{\mathrm{Higher-order}}, \quad (2)$$

where $\mathcal{V}$, $\mathcal{E}$, and $\mathcal{S}$ refer to pixel-wise, pair-wise, and higher-order clique sets[1], and ψ_i, ψ_{ij}, and ψ_c^h refer to potentials defined on each clique, respectively.

3. Higher-order term with Superpixel Models

In conventional CRF-based semantic segmentation algorithms [6, 7, 10, 11], higher-order potential is modeled to encourage all pixels in a superpixel to be assigned with the same label by taking the following form of the Potts model [1]:

$$\psi_c^h(\mathbf{x}_c) = \begin{cases} \gamma_\ell & \text{if}, \forall i \in c, x_i = \ell, \ell \in \mathcal{L} \\ \gamma_{\max} & \text{otherwise} \end{cases}, \quad (3)$$

where $\gamma_\ell \leq \gamma_{\max}$ is associated with superpixel-wise probability for object classes obtained from *TextonBoost* and

[1] Higher-order clique sets $\mathcal{S}$ are defined as a set of superpixels in the image domain

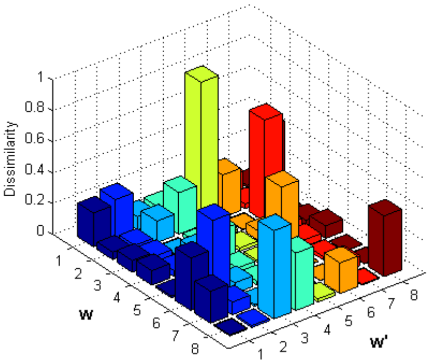


Figure 4. Sample result for the Texton codeword in MSRC-21 dataset. The size of matrix $\mathbf{M}^{Texton}_{w|w'}$ is 8×8, since codeword size V_k is set to 8 for visualization. In our experiments, V_k is set to 150 for each feature space as in [11]. Note that the matrix $\mathbf{M}^k_{w|w'}$ is learned for each feature space (*i.e.* SIFT, Texton, LBP, Color SIFT) in the off-line learning phase.

$\gamma_{\max}$ is an available maximum potential for $\psi^h_c(\mathbf{x}_c)$. In Eq. (3), higher-order potentials are generated based on superpixel-level cues: 1) the region of each superpixel, which is defined as pixel sets inside and 2) its preference for object classes. Both are important since if either one is incorrect, we cannot achieve the optimal semantic segmentation by solving Eq. (1). To make it robust, we propose two superpixel models: superpixel coherency model $\mathcal{M}_{coh}$ and superpixel uncertainty model $\mathcal{M}_{unc}$, which correspond to two aspects.

3.1. Superpixel Coherency Model

To robustly utilize the superpixel's region information, we define a superpixel coherency model $\mathcal{M}_{coh}$ whose response is a function of a superpixel c and a pixel $i \in c$. The importance of this model is that it enables to estimate the effect of superpixel c on the pixel i according to the dissimilarity between pixel i and superpixel c. For a superpixel c, if the region of a superpixel is incorrect, pixels from different object classes can be included in the same superpixel. It violates the assumption that pixels constituting a particular superpixel belong to the same object class. Since the Gibbs energy in Eq. (2) is inversely proportional to the probability in Eq. (1), we decrease the response of the model $\mathcal{M}_{coh}$ when the similarity is high, and increase it otherwise.

Dissimilarity learning To model the superpixel coherency, the codeword dissimilarity matrix for the k-th codebook, $\mathbf{M}^k_{w|w'}$ is trained in the off-line learning phase as in Fig. 4. It stores dissimilarities between two codewords $1 \leq w, w' \leq V_k$ by representing the likelihood of false co-occurrence for codeword index w given that codeword index w' occurs in the same superpixel. In codebooks, there are codeword pairs which co-occur incorrectly in the same superpixel, even though they are from different object classes. Using a set of training images, we record the statistic in the matrix $\mathbf{M}^k_{w|w'}$, including their pixel-wise ground-truth, superpixel channels and corresponding codeword indices. First, we determine the ground-truth label $\ell^* \in \mathcal{L}^N$ for a superpixel c by counting the number of pixel's ground-truth labels inside and selecting the dominant one. Second, we divide regions of superpixel c into two sets, $c = c_a \cup c_b$ where c_a includes pixels that have label ℓ^* as their ground-truth, while c_b includes pixels that have labels other than ℓ^* as their ground-truth. Third, the matrix is obtained based on the pixels appearance vector $\mathbf{v}_i = \{v^1_i, \ldots, v^n_i\}$ as follows:

$$\mathbf{M}^k_{w|w'} = \sum_{c\in\mathcal{S}}\left(\sum_{i\in c_a}\sum_{j\in c_b}\frac{\delta(w=v^k_j)}{N_c}\cdot\frac{\delta(w'=v^k_i)}{N_c}\right), \qquad (4)$$

where $1 \leq k \leq n$ denotes the feature space, $\delta(\cdot)$ is an indicator function, which is 1 if the statement is true and 0 otherwise. N_c is the number of pixels in the superpixel c. Lastly, we normalize the model to be $\sum_{w=1}^{V_k}\mathbf{M}^k_{w|w'} = 1$.

Response of the coherency model We calculate the response of the superpixel coherency model for pixel $i \in c$, $\mathcal{M}_{coh}(c, i)$ based on dissimilarities between two pixels $i, j \in c$, $j \neq i$. Using the dissimilarity matrix $\mathbf{M}^k_{w|w'}$ in Eq. (4), obtained from the off-line learning phase, the response value is calculated as follows:

$$\mathcal{M}_{coh}(c,i) = \prod_{j\in c, j\neq i}\left(\frac{1}{n}\sum_{k=1}^{n}\mathbf{M}^k_{v^k_j|v^k_i}\right). \qquad (5)$$

Note that dissimilarities between two codewords for a pixel i and j are averaged in multiple feature spaces and multiplied for all pixels $j \in c$, $j \neq i$. Then, we normalize the value by $\mathcal{M}_{coh}(c, i) \cdot \dfrac{1}{\sum_{k\in c}\mathcal{M}_{coh}(c,k)}$

3.2. Superpixel Uncertainty Model

To robustly utilize superpixel-wisely recognized probabilities for object classes, we define a superpixel uncertainty model $\mathcal{M}_{unc}$ whose response is a function of a superpixel c. Its goal is to represent the amount of certainty we can have in a probability distribution for object classes, and control the effect of a superpixel c on pixel i based on it. For a superpixel c, its preference for object classes is obtained from *TextonBoost*, and if it is incorrect, pixels in the superpixel region have incorrect preference information and it degrades the accuracy of semantic segmentation.

Response of the uncertainty model Superpixel-wise preference for object classes, obtained from *TextonBoost* is expressed as a probability distribution on $\mathcal{P}(x_c = \ell)$, where

$\ell \in \mathcal{L}$ and x_c denotes a possible assignment of label to superpixel c from the label set $\mathcal{L}$. The probability distribution tends to be uniform when superpixel-wise recognition is ambiguous, while there appears a dominant label when it is given with the certainty. Motivated by the observation, we propose a method for calculating the response of the uncertainty model, which uses the entropy measure on the probability distribution as follows:

$$\mathcal{M}_{unc}(c) = \sum_{\ell \in \mathcal{L}} \Big(- \mathcal{P}(x_c = \ell) \cdot \log\left(\mathcal{P}(x_c = \ell)\right) \Big). \tag{6}$$

For, $0 \leq \mathcal{P}(x_c = \ell) \leq 1$, the entropy measure decreases as a dominant label appears in the probability distribution, while it increases as probabilities become uniform. Therefore, based on the response, the uncertainty model captures the uncertainty in the probability distribution and enables to control the effect of superpixels on the pixels inside.

4. Inference by Mean-field Approximation

We now integrate the proposed superpixel models in the efficient inference method for the CRF model. Before describing our inference method, we briefly review the mean-field approximation method. Also, sub-optimal parameter learning is introduced for the optimal inference. The pseudo code of overall algorithm is described in Algorithm 1.

4.1. Mean-field approximation

In [9, 14], the efficient optimization method for the CRF model is derived based on mean-field inference method. In the mean-field inference, joint distribution $\mathcal{Q}(\mathbf{x})$ is defined as an approximation for the exact distribution $\mathcal{P}(\mathbf{x})$, which is assumed to be a simple multinomial distribution $\mathcal{P}(\mathbf{x}) \approx \mathcal{Q}(\mathbf{x}) = \prod_{i=1}^{N} \mathcal{Q}_i(x_i)$, where we denote $\mathcal{Q}(x_i = \ell)$ as $\mathcal{Q}_i(x_i)$ for notational clarity. The mean field inference method optimizes the CRF model by minimizing KL divergence between distributions $\mathcal{P}$ and $\mathcal{Q}$. Then, the general form of marginal distribution $\mathcal{Q}_i(x_i)$ is derived in [8] as follows:

$$\mathcal{Q}_i(x_i) = \frac{1}{Z_i} \exp\left(- \sum_{c \in C} \sum_{\{\mathbf{x}_c | x_i = \ell\}} \mathcal{Q}_{c-i}(\mathbf{x}_{c-i}) \psi_c(\mathbf{x}_c) \right) \tag{7}$$

where $\mathbf{x}_{c-i}$ denotes an assignment of variables apart from X_i, $\mathcal{Q}_{c-i}$ denotes the marginal distribution of all variables in c apart from X_i derived from the joint distribution $\mathcal{Q}$.

4.2. Inference for our method

Now, we derive the inference method for our algorithm. The marginal distribution for a pixel i can be derived based on Eq. (7) as follows:

$$\mathcal{Q}_i(x_i) = \frac{1}{Z_i} \exp(-\phi^{Pixel} - \phi^{Pair} - \phi^{Higher}), \tag{8}$$

Algorithm 1: Proposed semantic segmentation

```
——— Off-line Learning Phase ———
Input: Training images, Ground-truth
Output: Codebooks, $\mathbf{M}^k_{w|w'}$ $(1 \leq k \leq n)$
iterate
    Codebook learning for the $k$-th feature
    Feature extraction from whole train images
    Build a codebook with size $V_k$ by clustering each feature space
    iterate
        Codeword dissimilarity learning
            Find a ground-truth label for superpixel $c$
            Bipartite region of $c$ into two $c = c_a \cup c_b$
    until number of superpixels;
    Calculate dissimilarity matrix $\mathbf{M}^k_{w|w'}$ (Eq. (4))
    Normalize the matrix to achieve $\sum_{w=1}^{V_k} \mathbf{M}^k_{w|w'} = 1$
until $n$;
——— Processing Phase ———
Input: Test image $\mathbf{I}$
Output: Pixel-wise label set $\mathcal{L}^N$
Feature extraction and representation
Superpixel sets generation
Pixel- and Superpixel-wise recognition
iterate
    Mean-field inference
        Pixelwise potential (Eq. (9))
        Pairwise potential (Eq. (10))
        Higher-order potential from (Eq. (14))
            - Superpixel coherency model (Eq. (5))
                . Use off-line trained $\mathbf{M}^k_{w|w'}$ (Eq. (4))
            - Superpixel uncertainty model (Eq. (6))
until max iteration;
```

where

$$\phi^{Pixel} = \psi_i(x_i), \tag{9}$$

$$\phi^{Pair} = \sum_{\ell' \in \mathcal{L}} \sum_{(i,j) \in \mathcal{E}} Q_j(x_j = \ell') \cdot \psi_{ij}(x_i, x_j), \tag{10}$$

and

$$\phi^{Higher} = \sum_{c \in C} (\mathcal{Q}^h_{c-i} \cdot \gamma_\ell + (1 - \mathcal{Q}^h_{c-i}) \cdot \gamma_{\max}), \tag{11}$$

where $\mathcal{Q}^h_{c-i} = \sum_{j \in c, j \neq i} Q_j(x_j)$. This rule is iterated for each pixel i to obtain more accurate results. Four terms, shown in Eq. (11) are related to superpixel-level cues, where $\mathcal{Q}^h_{c-i}$ and $(1 - \mathcal{Q}^h_{c-i})$ correspond to the joint probability for all pixels other than i inside a superpixel, while γ_ℓ and $\gamma_{\max}$ correspond to superpixel-wise probabilities for object classes. To model the higher-order term robustly, in our algorithm, factors in Eq. (11) are transformed as follows:

$$\hat{\gamma}_\ell = \gamma_\ell \cdot \mathcal{M}_{unc}(c), \tag{12}$$

and

$$\hat{\mathcal{Q}}^h_{c-i} = \mathcal{Q}^h_{c-i} \cdot \mathcal{M}_{coh}(c, i). \tag{13}$$

Note that, in Eq. (12) and Eq. (13), superpixel-wise probabilities for object classes and the joint probability in superpixel regions are penalized based on responses of the uncertainty and coherency models, respectively. In Eq. (13), since $\mathcal{Q}^h_{c-i} = \sum_{j\in c, j\neq i} \mathcal{Q}_j(x_j)$ and $\mathcal{M}_{coh}(c,i) = \prod_{j\in c,j\neq i}\left(\frac{1}{n}\sum_{k=1}^{n} \mathbf{M}^k_{v_j^k|v_i^k}\right)$. , it can be interpreted that each $Q_j(x_j)$'s influence on $\mathcal{Q}^h_{c-i}$ is penalized based on the codeword dissimilarity $\mathbf{M}^k_{v_j^k|v_i^k}$. In our inference algorithm, we substitute Eq. (12) and Eq. (13) into Eq. (11) as follows:

$$\phi^{Higher} = \sum_{c\in\mathcal{C}}\left(\theta_1 \frac{\gamma_{\max}}{\gamma_{\max}-\gamma_\ell \mathcal{M}_{unc}(c)} - \theta_2 \mathcal{M}_{coh}(c,i)\mathcal{Q}^h_{c-i}\right), \tag{14}$$

where θ_1 and θ_2 are regarded as weight parameters for responses of two superpixel models.

4.3. Balancing two superpixel models

It is non-trivial to determine proper weights between two superpixel models in the CRF model. We find sub-optimal weights both for between-potentials and between two models via logistic regression [5]. We first split the Eq. (14) into two parts as follows:

$$f^1 = \sum_{c\in C}\left(\frac{\gamma_{\max}}{\gamma_{\max}-\gamma_\ell \mathcal{M}_{unc}(c)}\right), \tag{15}$$

$$f^2 = \sum_{c\in C}(-\mathcal{M}_{coh}(c,i)\cdot \mathcal{Q}^h_{c-i}), \tag{16}$$

Here, the model defined by Eq. (8) is simplified by removing the pairwise terms. Then, we define the probability for a pixel as follows:

$$P(x_i|\mathbf{I};\mathbf{w}) \propto \exp(-f^0 - f^1 - f^2), \tag{17}$$

where $f^0 = \psi_i(x_i)$, obtained from pixel-wise recognition step. Weights for logistic model $\mathbf{w} = \{w_0, w_1, w_2\}$ are learned to maximize the likelihood score in Eq. (17) over labeled training data via logistic regression. Then, we assign $\theta_1 = w_1/w_0$ and $\theta_2 = w_2/w_0$ in Eq. (14) to obtain sub-optimal parameters balancing two superpixel models.

5. Experiments

We evaluated the performance of our algorithm qualitatively and quantitatively on two well-known datasets: MSRC-21 [12] and PASCAL VOC-2010 [4]. In this section, we describe two datasets and present our results compared to the *state-of-the-art* algorithms [6, 7, 10, 11, 12].

5.1. Datasets

MSRC-21 dataset contains 591 images of resolution 213x320 with 21 object classes. PASCAL VOC-2010 dataset contains 1928 images with 20 foreground objects and 1 background class. We split two datasets into standard train, test and validation sets (45%, 45%, 10%, respectively) as in [12] and [9] respectively. For MSRC-21 dataset, algorithms are typically compared based on the overall precision of pixels correctly labeled (*Global*) and the average of intersection/union (IU) score per class (*Avg. IU*). The $IU = TP/(TP+FP+FN)$ score is defined in terms of the true positives (TP), false positives (FP) and false negatives (FN). Also, though standard methods for PASCAL VOC-2010 dataset are evaluated based on the *Avg. IU*, we report both *Global* and *Avg. IU* for more accurate comparison. For the time complexity, we measure the mean execution time for overall images in the inference step using Intel(R) Xeon(R) 2.8GHz processor.

5.2. Results and discussions

We present quantitative comparisons for both time complexity and classification accuracy of several state-of-the-art algorithms [6, 7, 10, 11] in Table 1a and 1b for MSRC-21 and PASCAL VOC-2010 datasets, respectively. Also, we present IU scores for each class in Table 1. Qualitative comparisons are presented in Figs. 5a and 5b for MSRC-21 dataset and PASCAL VOC 2010 dataset, respectively.

Our experiments demonstrate that two superpixel models result in more accurate semantic segmentation with reasonable time complexity. The proposed algorithm achieves the state-of-the-art performance in terms of segmentation accuracy on MSRC-21 dataset. Our algorithm performs about 8 times faster than the previous state-of-the-art algorithm [11] on MSRC dataset while achieving 1% gain in the Global and 3% gain in the Avg. IU. Though it performs slower than two algorithms [9, 14], it achieves higher accuracy: 2% gain in the Global and $2 \sim 6$% gain in the Avg. IU than those approaches. Also, for PASCAL VOC-2010 dataset, our algorithm outperforms previous approaches in terms of segmentation accuracy with reasonable speed. Though [11] performs slightly better in the Global , we achieved 5% gain in the Avg. IU with much faster speed. Also, though the time complexity for our algorithm is higher than two algorithms [9, 14], ours achieves $1 \sim 4$% gain in the Global and $1 \sim 3$% gain in the Avg. IU, respectively. Note that all methods [9, 14, 11] including ours are based on the same unary and pairwise potentials. Hence, the performance gain of ours in Table 1 comes from higher-order potentials with proposed superpixel models and inference method.

From results, we can see that our algorithm is much faster than previous approach[11] which use graph-cut based higher-order inference and yield more accurate results. Also, it is robust to rigid objects such as aeroplane, bird etc, rather than amorphous backgrounds such as road, sky etc. It reflects the fact that accurate object boundaries of superpixels are more important in rigid objects than backgrounds. Also, most of the time gap between ours and [9, 14] is due to calculating responses of superpixel models, which can be reduced via parallelizing method.

	Shotton *et al*. [12]	Ladicky *et al*. [11]	Krahen-buhl *et al*. [9]	Vineet *et al*. [14]	Ours
Building	62	**82**	67	72	76
Grass	95	95	97	**98**	97
Tree	85	88	90	89	**92**
Cow	65	73	80	71	**81**
Sheep	74	88	**91**	84	87
Sky	89	**100**	96	98	95
Airplane	76	83	**90**	75	87
Water	61	**92**	82	91	88
Face	84	88	**91**	85	90
Car	67	**87**	82	74	84
Bicycle	87	88	**94**	78	90
Flower	85	**96**	92	91	93
Sign	48	**96**	53	68	62
Bird	33	27	38	44	**49**
Book	94	85	**95**	94	**95**
Chair	52	37	61	51	**63**
Road	82	93	88	**94**	90
Cat	71	49	86	62	**87**
Dog	46	**80**	62	44	63
Body	70	65	82	81	**84**
Boat	**28**	20	20	14	19
Global	72	86	85	85	**87**
Avg. IU	69	77	78	74	**80**
Time	0.5s	15s	0.3s	1s	2s

(a)

	Shotton *et al*. [12]	Ladicky *et al*. [11]	Krahen-buhl *et al*. [9]	Vineet *et al*. [14]	Ours
Background	69	**90**	84	87	86
Aeroplane	47	40	53	50	**59**
Bicycle	11	5	**23**	11	11
Bird	12	16	15	24	**25**
Boat	**21**	12	19	20	**21**
Bottle	4	5	9	12	**13**
Bus	26	36	33	**37**	**37**
Car	30	53	50	54	**56**
Cat	33	**53**	**53**	50	**53**
Chair	13	13	9	16	**18**
Cow	**10**	4	7	5	6
Table	40	**43**	39	**43**	42
Dog	10	10	**15**	12	13
Horse	12	20	18	25	**26**
Motorbike	46	42	40	50	**52**
Person	54	63	60	64	**66**
Potted plant	**23**	10	16	14	13
Sheep	30	30	34	34	**35**
Sofa	14	14	**22**	13	13
Train	27	35	40	41	**42**
TV/Monitor	41	37	40	47	**47**
Global	59	**76**	71	74	75
Avg. IU	27	30	32	34	**35**
Time	3s	60s	2s	5s	6s

(b)

Table 1. Quantitative results for (a) MSRC-21 and (b) PASCAL VOC-2010 dataset

6. Conclusion

We presented an efficient semantic segmentation algorithm with superpixel coherency and uncertainty models. Two superpixel models are proposed to control effects of superpixel cues and integrated in the inference method for the CRF model. In the experiment, we gained about $3 \sim 5\%$ accuracy gain with moderate time complexity.

References

[1] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(11):1, 2001. 3

[2] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002. 2

[3] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000. 2

[4] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 6

[5] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008. 6

[6] J. M. Gonfaus, X. Boix, J. Van De Weijer, A. D. Bagdanov, J. Serrat, and J. Gonzalez. Harmony potentials for joint classification and segmentation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition,* San Francisco, CA, pages 3280–3287, June 2010. 1, 3, 6

[7] P. Kohli, P. H. Torr, et al. Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision*, 82(3):302–324, 2009. 1, 3, 6

[8] D. Kollar and N. Friedman. *Probabilistic graphical models: principles and techniques*. The MIT Press, 2009. 5

[9] P. Krähenbühl and V. Koltun. Efficient inference in fully connected CRFs with gaussian edge potentials. In *Proc. Advances in Neural Information Processing Systems 24*, pages 109–117, 2011. 5, 6, 7, 8

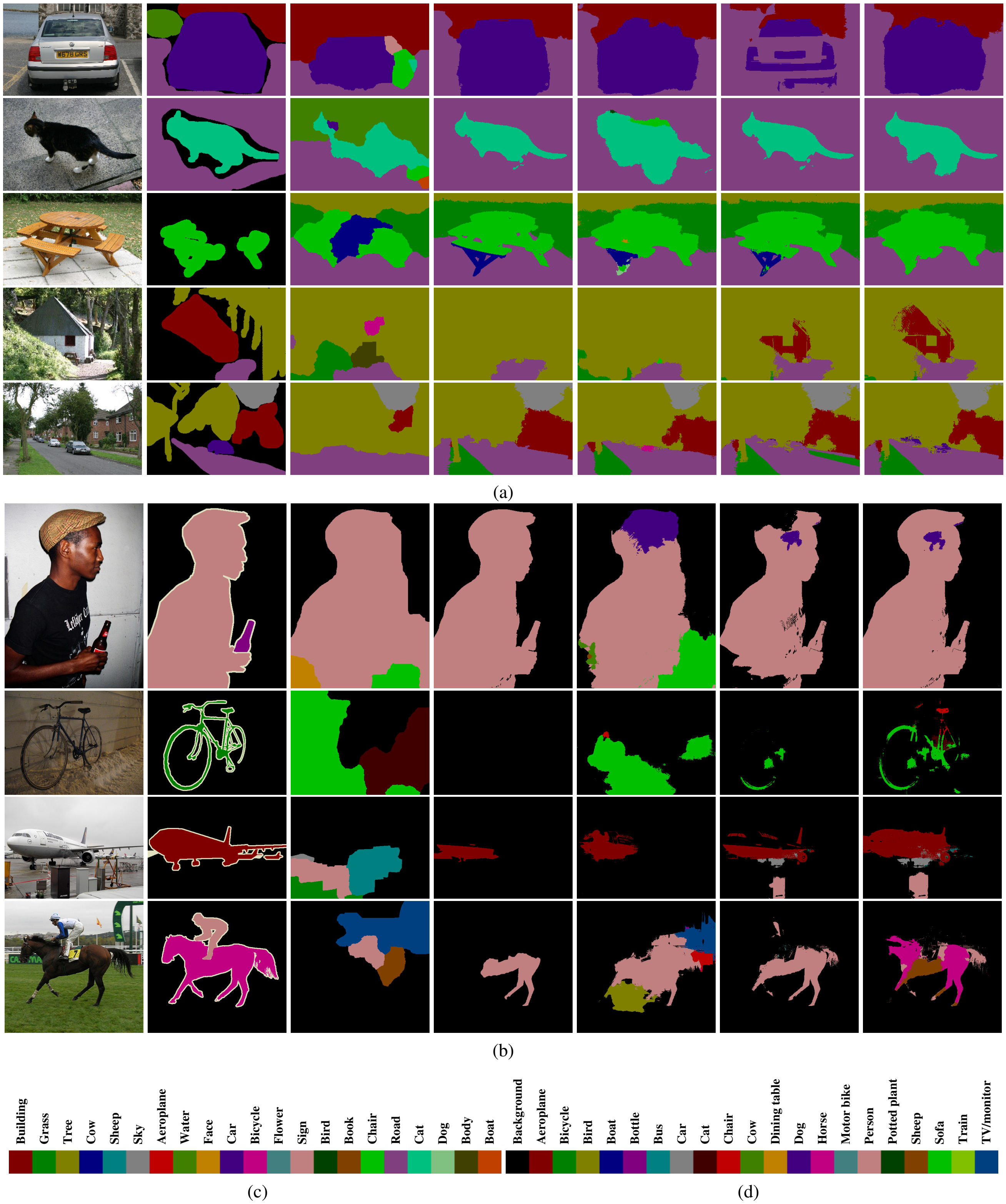
[10] L. Ladicky, C. Russell, P. Kohli, and P. H. Torr. Associative hierarchical crfs for object class image segmentation. In *Proc. 12th Intl. Conf. on Computer Vision,* Kyoto, Japan, pages 739–746, Oct. 2009. 1, 3, 6

[11] L. Ladicky, C. Russell, P. Kohli, and P. H. Torr. Graph cut based inference with co-occurrence statistics. In *Proc. European Conf. on Computer Vision,* Crete, Greece, pages 239–253, Sept. 2010. 1, 2, 3, 4, 6, 7, 8

[12] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *International Journal of Computer Vision*, 81(1):2–23, 2009. 1, 2, 3, 6, 7, 8

[13] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering objects and their location in images. In *Proc. 10th Intl. Conf. on Computer Vision,*Beijing, China. 2

[14] V. Vineet, J. Warrell, and P. H. Torr. Filter-based mean-field inference for random fields with higher-order terms and product label-spaces. In *Proc. European Conf. on Computer Vision,* Florence, Italy, pages 31–44, Nov. 2012. 1, 2, 5, 6, 7, 8


(a)

(b)

(c)

(d)

Figure 5. Some qualitative results for (a) MSRC dataset: (Col 1) input images, (Col 2) ground-truth, (Col 3) results by Shotton *et al.* [12]], (Col 4) results by Ladicky *et al.* [11], (Col 5) results by Krahenbuhl *et al.* [9], (Col 6) results by Vineet *et al.* [14], (Col 7) results by our algorithm. (b) PASCAL VOC 2010 dataset: (Col 1) input images, (Col 2) ground-truth, (Col 3) results by Shotton *et al.* [12]], (Col 4) results by Ladicky *et al.* [11], (Col 5) results by Krahenbuhl *et al.* [9], (Col 6) results by Vineet *et al.* [14], (Col 7) results by our algorithm. (c) Color map for MSRC-21 dataset. (d) Color map for PASCAL VOC-2010 dataset.