Journal of information and communication convergence engineering
http://jicce.org

J. Inf. Commun. Converg. Eng. 13(3): 205-214, Sep. 2015 Regular paper

# CRF-Based Figure/Ground Segmentation with Pixel-Level Sparse Coding and Neighborhood Interactions

**Lihe Zhang and Yongri Piao**[*], ***Member, KIICE***

School of Information and Communication Engineering, Dalian University of Technology, Dalian 116-024, China

## Abstract

In this paper, we propose a new approach to learning a discriminative model for figure/ground segmentation by incorporating the bag-of-features and conditional random field (CRF) techniques. We advocate the use of image patches instead of superpixels as the basic processing unit. The latter has a homogeneous appearance and adheres to object boundaries, while an image patch often contains more discriminative information (e.g., local image structure) to distinguish its categories. We use pixel-level sparse coding to represent an image patch. With the proposed feature representation, the unary classifier achieves a considerable binary segmentation performance. Further, we integrate unary and pairwise potentials into the CRF model to refine the segmentation results. The pairwise potentials include color and texture potentials with neighborhood interactions, and an edge potential. High segmentation accuracy is demonstrated on three benchmark datasets: the Weizmann horse dataset, the VOC2006 cow dataset, and the MSRC multiclass dataset. Extensive experiments show that the proposed approach performs favorably against the state-of-the-art approaches.

**Index Terms**: Conditional random field, Figure/ground segmentation, Neighborhood interaction, Sparse coding

## I. INTRODUCTION

Figure/ground segmentation aims at partitioning an image into regions of coherent properties as a means for separating objects from their backgrounds. Considerable effort has been made to develop many advanced techniques in the recent years, where learning segmentation has attracted considerable attention of researchers because of its significant performance in classification applications. Further, probabilistic graphical models have also been remarkably successful in segmentation applications.

In any image system, feature representation is crucial to enhancing the system performance. How to manifest an image and how to capture salient properties of the object regions are still challenging problems. The bag-of-features (BoF) model [1-3] has been widely used in the field of image processing. The model treats an image as a collection of unordered appearance descriptors extracted from local patches, quantizes them into discrete ‘visual words,’ and then computes a compact histogram representation. In this work, we propose a patch-level BoF model to effectively represent an image patch from raw image data. By pixel-level dictionary learning, sparse coding, and spatial pyramid matching, the feature representation can capture the salient properties of the image patch, thus resulting in high patch-wise segmentation accuracy.

Learning segmentation converts the image segmentation problem into a data clustering problem of image elements. One of the core challenges for machine learning is to discover what kind of information can be learned from the

**Received** 14 August 2015, **Revised** 20 August 2015, **Accepted** 04 September 2015
***Corresponding Author** Yongri Piao (E-mail: yrpiao@dlut.edu.cn, Tel: +86-15585556153)
School of Information and Communication Engineering, Dalian University of Technology, Dalian 116-024, China.

Open Access http://dx.doi.org/10.6109/jicce.2015.13.3.205 print ISSN: 2234-8255 online ISSN: 2234-8883

data sources and cluster this data into segments depicting the same object. Ren and Malik [4] proposed a classification model for segmentation, which feeds the Gestalt grouping cues into a linear classifier to discriminate between good and bad segmentation. Wu and Nevatia [5] developed a method to simultaneously detect and segment objects by boosting the edgelet feature classifiers. Duygulu et al. [6] modeled object recognition as a process of annotating image regions with words, and learning a mapping between region types and keywords by using an EM algorithm.

Probabilistic graphical models usually construct a cost function on the basis of some image constraints and formulate the image segmentation problem as a stochastic optimization problem. A condition random field (CRF) provides a principled approach to incorporating data-dependent interactions; the complex joint probability distribution need not be modeled in this case. In this work, we use the CRF model to fuse multiple visual cues. For the CRF model, the definition of unary and pairwise potentials is very important. Previously, the unary potential was directly defined on feature spaces [7]. Lately, researchers have paid more attention on using a classifier to generate a unary potential [8-15], and most of them prefer using a pixel or a superpixel as the basic processing unit. In contrast, we use a regular image patch. Image patches on object boundaries contain rich local structure information of an object (see Fig. 1). While superpixels usually have a homogeneous appearance and an almost uniform size along with edge preservation, particularly for weak boundaries, these properties weaken the discriminative capability of the unary classifier when the superpixel is taken as the sampling unit.

Our main contributions in this paper are twofold. First, we use an image patch as a sample of a unary classifier and propose an upgrade patch feature representation based on pixel-level sparse coding, which can capture more structure information of the local contour of objects. Second, we propose the color and texture pairwise potentials with neighborhood interactions and an edge potential representing edge continuity, which are validated to be very effective in our experiments.

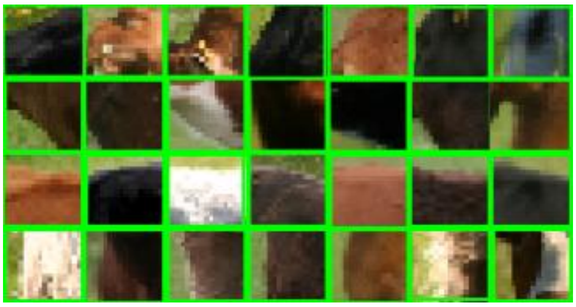

**Fig. 1.** Some image patches on object boundaries. Ignoring their color information, we find that there are many similar spatial structures, which can be considered the common characteristics of the objects.

## II. CONDITIONAL RANDOM FIELDS

CRFs are probabilistic models for segmenting data with structured labels [16], which are defined on a two-dimensional discrete lattice, every site on which corresponds to a graph node. Let $G = (\boldsymbol{V}, \boldsymbol{E})$ be an undirected graph with image patches as nodes $\boldsymbol{V}$ and the links between pairs of nodes as edges $\boldsymbol{E}$. CRFs directly model the distribution $P\ \mathbf{L} \,|\, \mathbf{I}, \mathbf{w}, \mathbf{v}$ of node labels $\mathbf{L}$ conditional on image data $\mathbf{I}$ for node parameters $\mathbf{w}$ and edge parameters $\mathbf{v}$. We are interested in finding the labels $\mathbf{L} = \{l_i\}_{i \in \boldsymbol{V}}$, where $l_i$ denotes the label of the $i^{\text{th}}$ node. In this work, we are concerned with binary segmentation (foreground and background), i.e., $l_i \in \{-1, 1\}$. The joint distribution over the labels $\mathbf{L}$ given the observations $\mathbf{I}$ can be expressed as follows:

$$P\ \mathbf{L} \,|\, \mathbf{I}, \mathbf{w}, \mathbf{v}\ = \frac{1}{z} \exp\left( \sum_{i \in \boldsymbol{V}} A_i\ \ l_i, \mathbf{I}, \mathbf{w}\ + \sum_{i \in \boldsymbol{V}} \sum_{j \in \mathcal{N}_i} I_{ij}\ \ l_i, l_j, \mathbf{I}, \mathbf{v} \right), \quad (1)$$

where $z$ represents a normalized constant known as the partition function, $\mathcal{N}_i$ denotes the set of neighbors of the $i^{\text{th}}$ node in graph $G$, and $A_i$ and $I_{ij}$ indicate the unary and pairwise potentials, respectively. The unary potential is modeled using a local discriminative model that decides the association of a given node to a certain class, ignoring the interaction of its neighbors. In contrast, the pairwise potential is regarded as a data-dependent smoothing function that denotes the interaction between two nodes. Both terms explicitly depend on a predefined set of features from $\mathbf{I}$.

## III. MAIN WORK

We use a CRF model to learn the conditional distribution over figure/ground labeling given an image, which allows us to incorporate different levels and different types of features in a single unified model.

### A. Unary Potentials

In this work, the unary potentials are defined by the prediction probability obtained from a linear support vector machine (SVM) classifier. Different from the existing feature descriptions, we train a pixel-level over-complete dictionary to sparsely represent image patches in a high-dimensional space.

#### 1) Pixel-Level Texture Descriptor

Gabor wavelets have received considerable attention because of biological reasons and their optimal resolution in both frequency and spatial domains. The Gabor wavelet representation can capture the local structure corresponding to the spatial scale, spatial localization, and orientation selectivity. It can characterize the spatial frequency structure in the image, while preserving the information of spatial relations. However, many existing image representation approaches in the Gabor domain merely consider the magnitude information. In this work, we proposed a new pixel-level feature descriptor, which fuses the Gabor magnitude and the Gabor phase.

To eliminate local noise interference, a simple smoothing filter is used for removing image noise in advance. Then, we perform the Gabor transform in $D$ directions and $S$ scales on a given gray image, and respectively, denote the magnitude response and the phase response in direction $\theta$ and scale $\sigma$ as $\rho_{\theta,\sigma}$ and $\alpha_{\theta,\sigma}$. Further, the $2\pi$ phase space is uniformly quantized into $J$ intervals as $\mathbf{\Phi}_j = [\phi_{j,\min}, \phi_{j,\min} + \varsigma]$, $j = 1,\dots,J$. $\varsigma = 2\pi / J$ represents the quantization step, and $\phi_{j,\min}$ denotes the margin value between two phase intervals $\mathbf{\Phi}_{j-1}$ and $\mathbf{\Phi}_j$. Suppose that phase response $\alpha_{\theta,\sigma}$ belongs to the $j^{\text{th}}$ interval $\mathbf{\Phi}_j$. Then, we compute Eq. (2) to get a $J$-dimensional vector $\boldsymbol{y}_{\theta,\sigma}$ in direction $\theta$ and scale $\sigma$ as follows:

$$\boldsymbol{y}_{\theta,\sigma} = [0,\dots,0,y^j_{\theta,\sigma},y^{j+1}_{\theta,\sigma},0,\dots,0], \tag{2}$$

where

$$\begin{cases} y^j_{\theta,\sigma} = \cos(\alpha_{\theta,\sigma} - \phi_{j,\min}) \times \rho_{\theta,\sigma} \\ y^{j+1}_{\theta,\sigma} = \cos(\alpha_{\theta,\sigma} - (\phi_{j,\min} + \varsigma)) \times \rho_{\theta,\sigma} \end{cases}.$$

An example of diagram $J = 8$ is shown in Fig. 2, in which the phase space is quantized into eight intervals and $\varsigma = \pi / 4$. Assuming the phase response $\alpha_{\theta,\sigma} \in \mathbf{\Phi}_2$, we can update the margin value as $\varphi_{1,\min} = 0$, $\varphi_{2,\min} = \pi/4,\dots,$ $\varphi_{8,\min} = 7\pi/4$; the $J$-dimensional vector $\boldsymbol{y}_{\theta,\sigma} = [0,0,y^2_{\theta,\sigma},y^3_{\theta,\sigma},0,0,0,0]$.

We concatenate all vectors $\boldsymbol{y}_{\theta,\sigma}$ of the given $D$ directions and $S$ scales as the pixel-level descriptor $\boldsymbol{y} = [\boldsymbol{y}_{1,1},\boldsymbol{y}_{1,2},\dots,\boldsymbol{y}_{D,S}]$. The $\ell = D \times S \times J$ -dimensional feature vector not only describes the distribution of the phase response in each scale and direction but also reflects the magnitude response.

### 2) Pixel-Level Dictionary Learning and Coding

Sparse representations have demonstrated considerable success in numerous applications, and the sparse modeling of signals has been proven to be very effective in signal reconstruction and classification. We randomly sample some pixels from the training image set to learn an over-complete pixel-level dictionary. Assuming that we collect N training samples, we define a matrix $\mathbf{Y} \in \mathbf{R}^{\ell \times N}$ as the columns of samples:

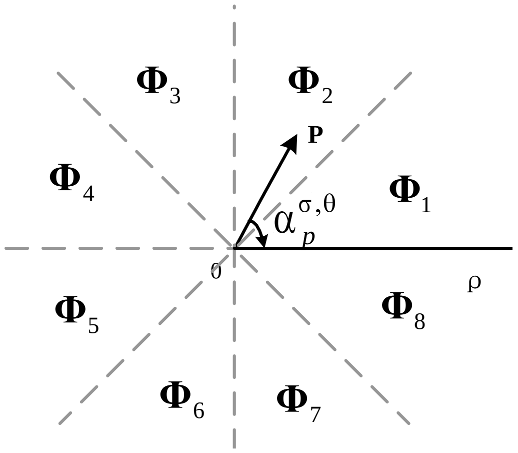

**Fig. 2.** An example diagram of $J = 8$; phase response $\alpha_{\theta,\sigma}$ belongs to the 2nd phase interval $\mathbf{\Phi}_2$.

$$\mathbf{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \dots, \boldsymbol{y}_N], \tag{3}$$

where $\boldsymbol{y}_i$ stands for the $\ell$-dimensional texture feature of the $i^{\text{th}}$ sample.

Using an over-complete dictionary $\mathbf{D} \in \mathbf{R}^{\ell \times L}$, which contains $L$ atoms as column vectors, we can approximate the observed sample $\boldsymbol{y}$ well by using a sparse linear combination of these atoms. In particular, there exists a sparse coefficient vector $\boldsymbol{x}$ such that $\boldsymbol{y}$ can be approximated as $\boldsymbol{y} \approx \mathbf{D}\boldsymbol{x}$, where the vector $\boldsymbol{x}$ represents the weighted contribution of these atoms when reconstructing the observed sample. Given the training samples, we can learn the dictionary $\mathbf{D}$ by solving the following optimization problem:

$$\min_{\mathbf{D},\mathbf{X}} \sum_{i=1}^{N} \| \boldsymbol{y}_i - \mathbf{D}\boldsymbol{x}_i \|_2^2 + \lambda \| \boldsymbol{x}_i \|_1, \tag{4}$$

where $\lambda$ denotes a balance parameter and the second term enforces $\boldsymbol{x}$ to have a small number of nonzero elements.

The optimization problem is convex in $\mathbf{D}$ or $\mathbf{X}$ while fixing the other, but not in both simultaneously [17]. We solve it by alternating the optimization over $\mathbf{D}$ and $\mathbf{X}$; the dictionary $\mathbf{D}$ can be initialized by randomly sampling $L$ columns from $\mathbf{Y}$ or by K-means clustering. When fixing $\mathbf{D}$, the optimization becomes a standard sparse coding problem, which can be solved very efficiently by using the feature-sign search algorithm. When fixing $\mathbf{X}$, the problem reduces to a least squares problem (as shown in Eq. (5)), which can be solved by using the Lagrange dual algorithm.

$$\min_{\mathbf{D}} \sum_{i=1}^{N} \| \boldsymbol{y}_i - \mathbf{D}\boldsymbol{x}_i \|_2^2 . \tag{5}$$

Once the over-complete dictionary $\mathbf{D}$ is given, the texture feature $\boldsymbol{y}$ of each pixel can be coded as L-dimensional sparse vector $\boldsymbol{x}$ by solving the following l1-norm regularization problem:

$$\min_{\boldsymbol{x}} \| \boldsymbol{y} - \mathbf{D}\boldsymbol{x} \|_2^2 + \lambda \| \boldsymbol{x} \|_1 . \tag{6}$$

### 3) Patch Feature Representation

The pivotal role of the unary potential in the CRF-based segmentation model has been demonstrated. It can be taken as a local decision term, which decides the association of a given graph node to a certain class. Usually, the use of the unary classifier alone leads to high accuracy as compared to the full CRF model as it can segment most parts of an object and loses only some details of the object boundaries.

In this work, we integrate the texture feature and the color feature to represent an image patch. We partition a patch into 1×2, 2×2 segments in two different scales, and then, compute the max pooling vector of the sparse codes of pixels within each of the five segments. We finally concatenate all the vectors to form a vector representation of the texture feature. The so-called spatial pyramid matching has had remarkable success in image classification applications. Color information is very useful for identifying the classes of image patches. For example, backgrounds (e.g., sky, water, grass, and tree) are usually distinguishable from objects (e.g., cow, sheep, and bird) in color. For a patch, we compute 64-bin histograms in each CIE Lab color channel as its color feature and then, concatenate the texture vector and the color vector to form the final feature representation. In our experiments, we fixed the size of dictionary $\mathbf{D}$ as 2048; thus, the dimension of the patch feature is $2048 \times 5 + 64 \times 3 = 10432$. We also find that max pooling outperforms the other alternative pooling methods.

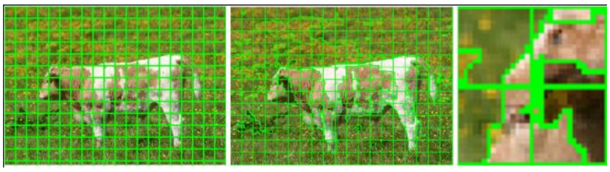

**Fig. 3.** Split patches. (a) Patch partition, where each square stands for an image patch. (b) Overlaying a patch partition on the image segmentation. (c) Local magnifying map, where an irregular region stands for a split patch.

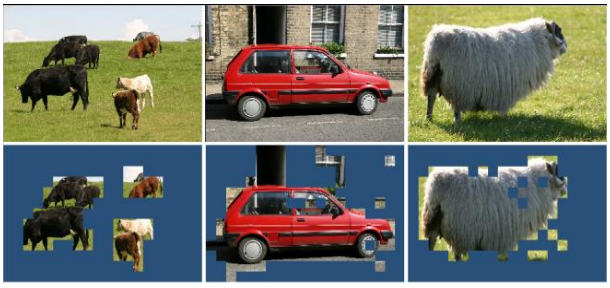

**Fig. 4.** Unary binary segmentation results (above: input images, below: binary classification results).

### 4) Unary Potential Computation

We train a binary linear SVM classifier to predict the figure/ground probabilities of an image patch, which are used for computing the unary potential. However, the boundaries of the foreground segmented by the CRF model using patches as graph nodes have a blocking effect. To generate results close to the ground truth, we split patches into some perceptually meaningful entities by using the over-segmentation boundary map, which is generated by an existing region merging method [18]. As shown in Fig. 3, a patch is split into several regions. These regions are considered the graph nodes, and their unary potential values are defined as the corresponding prediction probability of the host patch. That is, the regions split from a patch are assigned the same probability. In particular, the unary potential in Eq. (1) is defined as follows:

$$A_i(l_i, \mathbf{I}, \mathbf{w}) = \log(P(l_i \mid \mathbf{I})), \tag{7}$$

where $P(l_i \mid \mathbf{I})$ denotes the local class posterior, that is, the prediction probability given by the unary classifier.

## B. Pairwise Potentials

After the unary binary classification, we can already obtain good segmentation results, but the classifier separately processes each image patch. The mutual dependence among neighboring patches is ignored, which results in some neighboring patches with a similar appearance being possibly improperly labeled as opposite classes (see Fig. 4). Therefore, as contextual knowledge is necessary for image segmentation, we define the pairwise potential to address this problem. In this work, the pairwise penalty $I_{ij}$ is defined as the weight $w_{ij}$ of a graph edge, that is,

$$I_{ij}(l_i, l_j, \mathbf{I}, \mathbf{v}) = w_{ij}(\mathbf{I}) l_i l_j . \tag{8}$$

In image segmentation, the weights encode a graph affinity such that a pair of nodes with a high weight edge is considered to be strongly connected and edges with low weights represent the nearly disconnected nodes. We exploit the color, texture, and edge cues to model the connection between nodes and incorporate the three types of potentials in a unified CRF framework using pre-learned parameters. Assuming that the superscripts $c$, $t$, and $e$ denote the color, texture, and edge, respectively, we can rewrite $w_{ij}(\mathbf{I})$ as follows:

$$w_{ij}(\mathbf{I}) = \mathbf{v}g_{ij} = [v^c \;\; v^t \;\; v^e][g^c_{ij}(\mathbf{I}) \;\; g^t_{ij}(\mathbf{I}) \;\; g^e_{ij}(\mathbf{I})]^{\mathrm{T}}, \quad (9)$$

where $g_{ij}(\mathbf{I})$ represents a distance function defined over node pairs $(i,j)$. $w_{ij}(\mathbf{I})$ synthetically reflects the connectivity of nodes $i$ and $j$ on multiple feature spaces.

### 1) Color Potential and Texture Potential

Color information is an essential and typical representation for images and is a key element for distinguishing objects. Mean and histogram are two common color descriptors. Mean only describes the average color component rather than the color distribution in a region. We use the CIE Lab histogram as a color descriptor for computing the color potential. Similarly, each channel is uniformly quantified into 64 levels, and then, three channels are concatenated to form a 192-dimensional color vector. The experimental comparison demonstrates that the histogram descriptor is more effective than the mean descriptor, increasing the overall pixel-wise labeling accuracy by 3.8%.

Every region in natural images is not isolated and is strongly connected with its adjacent regions. When computing $g^c_{ij}(\mathbf{I})$, it is unreasonable to only use the node pair $(i,j)$ and ignore the neighboring nodes. Therefore, we consider the neighborhood interactions in the main steps summarized in Algorithm 1.

**Algorithm 1** Color potential computation

```
input: An image represented as a graph
for $i$ = 1 to $N$ do
        compute color histogram $\boldsymbol{h}_i$ of the $i^{\text{th}}$ node
end for
for $i$ = 1 to $N$ do
        $m_i \leftarrow \arg\min_{k, k \in \mathcal{N}_i} \chi^2(\boldsymbol{h}_i, \boldsymbol{h}_k)$
end for
for $\forall\; i \in \boldsymbol{V}$ do
        $g^c_{ij} \leftarrow D(i,j) \quad j \in \mathcal{N}_i$
end for
output: Color potential $\boldsymbol{g}^c = \{g^c_{ij} \mid (i,j) \in \boldsymbol{E}\}$
```

where $D(i,j) = [\chi^2(\boldsymbol{h}_i + \boldsymbol{h}_{m_i}, \boldsymbol{h}_j) + \chi^2(\boldsymbol{h}_i, \boldsymbol{h}_j + \boldsymbol{h}_{m_j})] / 2$ .

Similarly, we can compute the texture potential $g^t_{ij}(\mathbf{I})$. Further, $\boldsymbol{h}_i$ stands for a texture histogram, which is computed as follows:

$$\boldsymbol{h}_i = \frac{1}{K}\sum\nolimits_{k \in node_i} \boldsymbol{x}_k \,, \quad (10)$$

where $K$ denotes the number of pixels in the region node $i$, and $\boldsymbol{x}$ represents the $L$-dimensional sparse texture vector described in Section III-A. Experiments demonstrate that the step of incorporating neighborhood interactions increases the overall pixel-wise labeling accuracy by 2.3%. We also find that the color potential using the $Lab/\chi^2$ descriptor outperforms the GMM color model as used in [10].

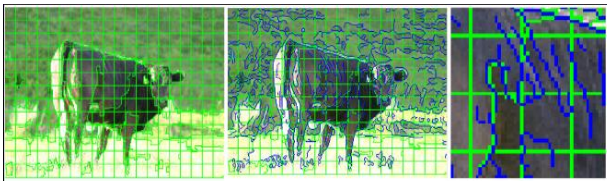
(a) (b) (c)

**Fig. 5.** Edge continuity. (a) Split patches. (b) Overlying image edge map on a split patch partition. (c) Local magnifying map.

### 2) Edge Potential

Inside a very small region, edge information basically indicates local image shape priors; the regions belonging to the same object often have strong edge continuities, which are described as the edge potential in this paper. As shown in Fig. 5, we find that there are many edges (blue lines) going through neighboring nodes. If two neighboring nodes are crossed by an edge, they very possibly belong to a visual unit and have the same figure/ground label. Motivated by this observation, we define the edge potential to capture the cue of edge continuity.

Given an image, we compute its binary gradient magnitude $\mathcal{M}$. Let $\mathcal{S}$ be the node index matrix, which indicates that the graph node that a pixel belongs to. Assuming that $c_n = (x_n, y_n)$ denotes the coordinate of pixel $n$ and $(i,j)$ represents a pair of neighboring nodes, we can denote their common boundaries as a pixel pair set, as follows:

$$\mathcal{E}_{ij} = \{(c_n, c_m) \,\big|\, | x_n - x_m | \le 1, | y_n - y_m | \le 1, \mathcal{S}(c_n) = i, \mathcal{S}(c_m) = j\} . \quad (11)$$

Then, the edge potential is computed as follows:

$$g^e_{ij}(\mathbf{I}) = \| \mathcal{E}^{\mathcal{M}}_{ij} \| \big/ z \,, \quad (12)$$

where $z = \max_{(i,j) \in \mathcal{N}} \| \mathcal{E}^{\mathcal{M}}_{ij} \|$ and

$$\mathcal{E}^{\mathcal{M}}_{ij} = \{(c_n, c_m) \,\big|\, \mathcal{M}(c_n) = 1, \mathcal{M}(c_m) = 1, (c_n, c_m) \in \mathcal{E}_{ij}\} .$$

## C. Parameter Estimation

The parameter vector $\mathbf{v}$ in Eq. (7) is automatically learned from the training data. Given a set of training images $\mathbf{T} = \{(\mathbf{L}^{(n)}, \mathbf{I}^{(n)}), n = 1, \ldots, N\}$, we assume that all the training data are independent and identically distributed. We then use the conditional maximum likelihood (CML) criterion to estimate **v**. Its log likelihood is computed as follows:

$$\begin{aligned} &L(\mathbf{v}) \\ &= \sum_{n=1}^{N} \log P(\mathbf{L}^{(n)} \mid \mathbf{I}^{(n)}, \mathbf{v}) \\ &= \sum_{n=1}^{N} \left[ \left( \sum_{i \in \boldsymbol{V}} A_i(l_i^{(n)}, \mathbf{I}^{(n)}) + \sum_{i \in \boldsymbol{V}} \sum_{j \in \mathcal{N}_i} l_i^{(n)} l_j^{(n)} \mathbf{v} g_{ij}(\mathbf{I}^{(n)}) \right) - \log z^{(n)} \right], \end{aligned} \tag{13}$$

where the last term is the log-partition function. In general, the evaluation of the partition function is a NP-hard problem. We could use either sampling techniques (e.g., the Markov chain Monte Carlo method [19]) or some approximations (e.g., those of the free energy [20], piecewise training [21], pseudo-likelihood [22]) to estimate the parameters.

The optimal parameter $\hat{\mathbf{v}}$ maximizes the log conditional likelihood according to the CML estimation as follows:

$$\hat{\mathbf{v}} = \arg\max_{\mathbf{v}} L(\mathbf{v}). \tag{14}$$

This can be solved by using the gradient descent method. The derivative of the log likelihood $L(\mathbf{v})$ is written as follows:

$$\begin{aligned} &\frac{\partial L(\mathbf{v})}{\partial \mathbf{v}} \\ &= \sum_{n=1}^{N} \left[ \sum_{i} \sum_{j \in \mathcal{N}_i} l_i^{(n)} l_j^{(n)} g_{ij}(\mathbf{I}^{(n)}) - E_P\left( \sum_{i} \sum_{j \in \mathcal{N}_i} l_i l_j g_{ij}(\mathbf{I}^{(n)}) \right) \right], \end{aligned} \tag{15}$$

where the second term $E_P(\cdot)$ denotes the expectation with respect to the distribution $P(l \mid \mathbf{I}^{(n)}, \mathbf{v})$. That is,

$$\begin{aligned} &E_P\left( \sum_{i} \sum_{j \in \mathcal{N}_i} l_i l_j g_{ij}(\mathbf{I}^{(n)}) \right) \\ &= \sum_{l} P(l \mid \mathbf{I}^{(n)}, \mathbf{v}) \sum_{i} \sum_{j \in \mathcal{N}_i} l_i l_j g_{ij}(\mathbf{I}^{(n)}) \cdot \end{aligned} \tag{16}$$

In general, the expectation cannot be computed analytically because of the combinatorial number of elements in the configuration space of labels. In this work, we use belief propagation [23] to approximate it.

## IV. EXPERIMENT AND DISCUSSION

### A. Image Datasets

We evaluate the proposed approach using three datasets. The MSRC dataset [10] contains 591 images with 21 categories. The performance of the unary classifier on this dataset is measured by using the pixel precision. Furthermore, for comparison with a previous work [24], we select the following 13 classes of 231 images with a 7-class foreground (cow, sheep, airplane, car, bird, cat, and dog) and a 6-class background (grass, tree, sky, water, road, and building) as the data subset. The ground truth labeling of the dataset contains pixels labeled as ‘void’ (i.e., color-coded as black), which implies that these pixels do not belong to any of the 21 classes. In our experiments, void pixels are ignored for both the training and the testing of the unary classifier. The dataset is randomly split into roughly 40% training and 60% test sets, while ensuring approximately proportional contributions from each class.

The second dataset is the Weizmann horse dataset [25], which includes the side views of many horses that have different appearances and poses. We have also used the VOC2006 cow database [26] in which ground truth segmentations are manually created. For the two datasets, the numbers of images in the training and test sets are exactly the same as in [27].

### B. Common Parameters

When extracting the pixel-level texture descriptor, we set the parameters of the Gabor filter as scales $\sigma = \{1, 1.2\}$ and directions $\theta = \{0, \pi / 4, \pi / 2, 3\pi / 4\}$. The phase response is uniformly quantified into eight regions. Hence, the size of the pixel-level feature vector is $4 \times 2 \times 8 = 64$. We randomly select roughly 60000 samples from all the training images to learn the dictionary **D** and ensure approximately proportional contributions from each image. We set the dictionary size as 2048. Thus, a 64-dimensional pixel-level vector is sparsely encoded as a 2048-dimensional vector; then, by spatial pyramid matching and max pooling, we extract a patch feature from 16×16 pixel patches, which are densely sampled with a step size of 4 pixels.

During the training of the unary classifier, a patch possibly contains multiple labels; however, we take the label that accounts for more than 75% of all the pixels in this patch as its label. We find that the number of patches of the training images in each class on average is in the order of 10000, and some classes have more than 100000 patches. Considering the memory and computational constraints, we randomly select 8000 patches from each class to construct a patch dataset for the evaluation of the unary classification, and each class sample is randomly split into 25% training and 75% test sets. For efficiency, we reduce the dimension of a patch feature from 10432 to 4000 by using the incremental principal component analysis (PCA) algorithm [28], in which we feed 20% samples to increment PCA in order to approximate the mean vector and the basis vectors.

### C. Unary Accuracy

To evaluate the performance of the proposed patch representation, we use a simple linear SVM classifier to conduct 21-class classification experiments on the MSRC dataset. We select 1200 patches per class as training samples and the rest of the patches as the testing samples. We achieve patch-wise labeling accuracy of 71.0%, while the state-of-the-art approach [10] gives pixel-wise accuracy of 69.6%. For a fair comparison, we further refine the patch precision segmentations to the pixel precision ones by simply post-processing.

In particular, we first get split patches (i.e., graph nodes) by using an existing segmentation method as described in Section III-A. The nodes are not always larger than the patches in size. Then, we take the nodes within the same segment as generated by [18] as the content consistent nodes. Finally, the label of each node is decided by a majority vote of the labels of its neighboring nodes, which must also be its content-consistent nodes. After the above processing, we achieve pixel-wise accuracy of 72.1%.

We also evaluate the binary classification performance on the 13-class dataset. We select 2200 patches per class as the training samples and label the 7-class foreground and the 6-class background as positive samples and negative samples, respectively. Thus, we achieve patch-wise labeling accuracy of 87.5%. After the post-processing, we achieve pixel-wise accuracy of 88.4%. The unary pixel-wise accuracy on the Weizmann dataset and the VOC2006 dataset is 89.9% and 94.5%, respectively.

### D. Potentials Analysis

On the 13-class dataset, we compare the unary potential accuracy with the full model accuracy. The latter is improved by 3.2% on average. This seemingly small numerical improvement corresponds to a large perceptual improvement (see Fig. 6), which shows that our pairwise potentials are effective.

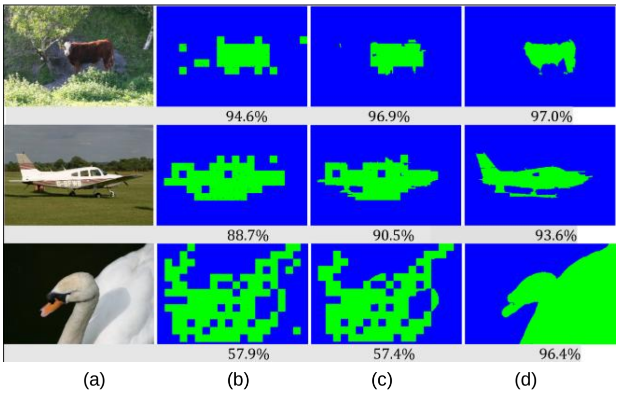

**Fig. 6.** Contribution of pairwise potentials. (a) Input images. (b) Results of the unary classification with patch-wise accuracy. (c) Post-processing results with pixel-wise accuracy. (d) Results of the full CRF model. The first two examples show an increase in accuracy of 0.1% and 3.1%, respectively, while the last example significantly improves the accuracy by 39.0%.

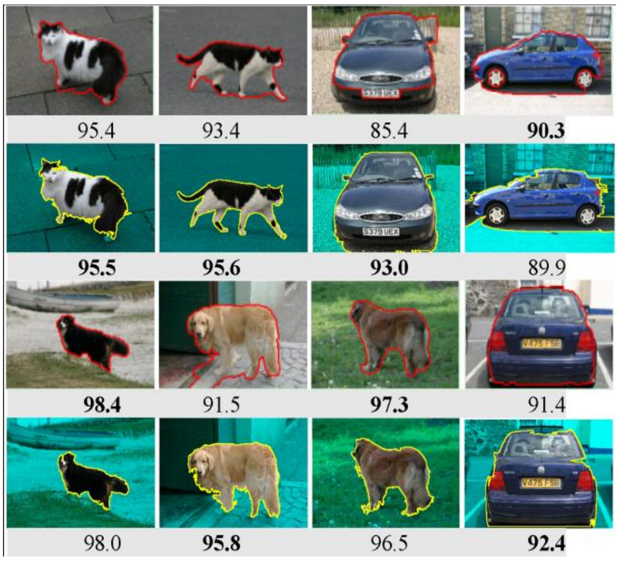

**Fig. 7.** Qualitative comparison with [24] for the MSRC database. The first and the third rows show the results reported in [24]. The second and the fourth rows show our results.

### E. Comparison

We evaluate the performance of the proposed method against that of three state-of-the-art methods [24, 27, 29]. The quantitative measure is the accuracy, namely segmentation cover, which is defined as the percentage of correctly classified pixels in the image (both foreground and background) [24, 29].

On the MSRC dataset, since the performance varies substantially for different classes, we respectively give the accuracy of each class. We list the quantitative comparison of seven classes in Table 1, which shows that our method outperforms the two competitors except for the cat class.

In addition, the method proposed in [24] only selects 10 images for each class such that there is a single object in each image, while we compute the segmentation accuracy on the 13-class sub-dataset of 231 images, and many images contain several object instances. The difference in testing data also indicates that our method is more robust than that proposed in [24]. Fig. 7 shows some visual examples of the same images as reported in [24]. Although the accuracies of some examples are lower than those in the case of the competitor methods, our overall accuracy is higher.

On the Weizmann and VOC2006 datasets, we compute the 2-class confusion matrix, as shown in Table 2, which shows that the proposed method performs favorably against

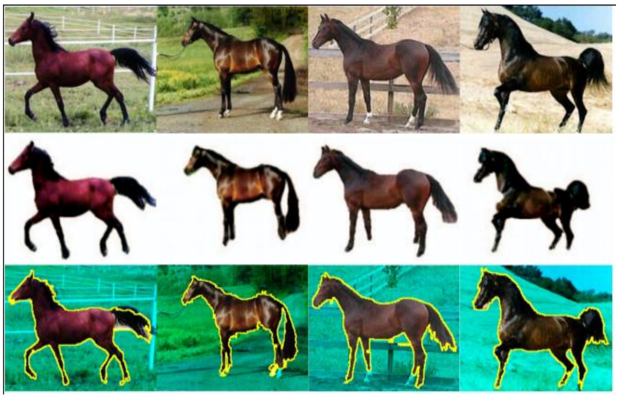

**Fig. 8.** Examples of representative segmentation results on the Weizmann horse dataset. From top to down: input images, results reported in [27], and our results.

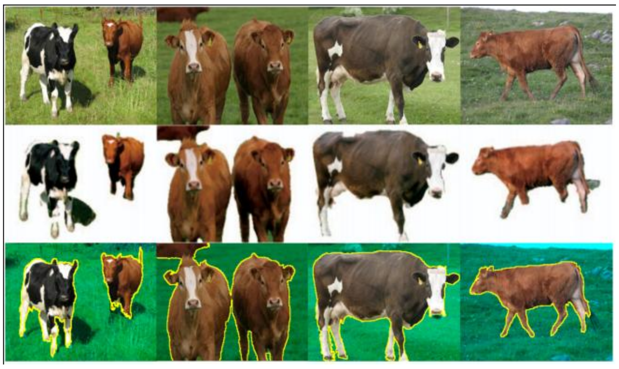

**Fig. 9.** Examples of representative segmentation results on the VOC2006 cow images. From top to down: Input images, results reported in [27], and our results.

the method proposed in [27] on the first dataset and much better on the second one. Figs. 8 and 9 show the same examples as those considered in [27]. The reason that the results on the cow dataset are very goodies that the appearances of the foreground and background are respectively homogeneous and the spatial distribution of the foreground is very compact. Compared to the horse dataset, the foreground of many images are inhomogeneous; in particular, horse shanks are very slim and their colors are different from those of the body, which leads to horse shanks going missing from the final segmentation, as shown in Fig. 8. In addition, the similar appearance of the foreground and the shadow in the background possibly causes some errors.

## V. CONCLUSIONS

In this paper, we propose a new discriminative model for figure/ground segmentation. First, a pixel-level dictionary is learnt from mass pixel-wise Gabor descriptors; second, each pixel is mapped as a high-dimensional sparse vector, and then, all the sparse vectors in a patch are fused to represent the patch by max pooling and spatial matching. The proposed unary features can simultaneously capture the appearance and context information, which significantly enhances the unary classification accuracy. The upgrade color and texture potentials with neighborhood interactions and the proposed edge potential weaken the interference of abnormal nodes during graph affinity computation. The experimental results demonstrate that the proposed approach is powerful with a comparison with three state-of-the-art approaches. In the future, we hope to integrate explicit semantic context and salient information to make the algorithm more intelligent.

**Table 1.** Quantitative comparison on the MSRC dataset

| | Carreira and Sminchisescu [29] | Vicente et al. [24] | Ours |
|---|---|---|---|
| Bird | 90.7 | 95.3 | **97.1** |
| Car | 72.3 | **80.7** | **80.7** |
| Cat | 87.8 | **92.3** | 92.2 |
| Cow | 92.9 | **94.2** | **94.2** |
| Dog | 88.7 | 93.1 | **94.6** |
| Plane | 78.2 | 83.0 | **85.9** |
| Sheep | 94.3 | 94.6 | **96.8** |

**Table 2.** Quantitative comparison on the Weizmann and VOC2006 datasets

| | Zhang and Ji [27] | | Ours | |
|---|---|---|---|---|
| | Fg | Bg | Fg | Bg |
| Weizmann | **91.8** | 96.6 | 90.1 | **98.7** |
| VOC2006 | 93.1 | 97.1 | **95.0** | **99.4** |

## ACKNOWLEDGMENTS

This work was supported by the Natural Science Foundation of China under Grants 61405022 and 61371157.

## REFERENCES

[ 1 ] L. Fei-Fei and P. Perona, “A Bayesian hierarchical model for learning natural scene categoriesm” in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR2005)*, San Diego, CA, pp. 524-531, 2005.

[ 2 ] L. Wu, S. C. Hoi, and N. Yu, “Semantics-preserving bag-of-words models and applications,” *IEEE Transactions on Image Processing*, vol. 19, no. 7, pp. 1908-1920, 2010.

[ 3 ] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: spatial pyramid matching for recognizing natural scene categories," in *Proceedings of 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, New York City, NY, pp. 2169-2178, 2006.

[ 4 ] X. Ren and J. Malik, "Learning a classification model for segmentation," in *Proceedings of 9th IEEE International Conference on Computer Vision*, Nice, France, pp. 10-17, 2003.

[ 5 ] B. Wu and R. Nevatia, "Simultaneous object detection and segmentation by boosting local shape feature based classifier," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR2007),* Minneapolis, MN, pp. 1-8, 2007.

[ 6 ] P. Duygulu, K. Barnard, J. F. de Freitas, and D. A. Forsyth, "Object recognition as machine translation: learning a lexicon for a fixed image vocabulary," in *Proceedings of 7th European Conference on Computer Vision (ECCV2002)*, Copenhagen, Denmark, pp. 97-112, 2002.

[ 7 ] X. He, R. S. Zemel, and M. A. Carreira-Perpiñán, "Multiscale conditional random fields for image labeling," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR2004)*, Washington, DC, pp. 695-702, 2004.

[ 8 ] P. Kohli. L. Ladicky, and P. H. Torr, "Robust higher order potentials for enforcing label consistency," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR2008)*, Anchorage, AK, pp. 1-8, 2008.

[ 9 ] L. Ladicky, C. Russell, P. Kohli, and P. H. Torr, "Associative hierarchical CRFs for object class image segmentation," in *Proceedings of 2009 IEEE 12th International Conference on Computer Vision*, Kyoto, Japan, pp. 739-746, 2009.

[10] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: joint appearance, shape and context modeling for multi-class object recognition and segmentation," in *Proceedings of 9th European Conference on Computer Vision (ECCV2006)*, Graz, Austria, pp. 1-15, 2006.

[11] C. Chen, D. Freedman, and C. H. Lampert, "Enforcing topological constraints in random field image segmentation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR2011)*, Providence, RI, pp. 2089-2096, 2011.

[12] A. Rosenfeld and D. Weinshall, "Extracting foreground masks towards object recognition," in *Proceedings of 2011 IEEE International Conference on Computer Vision (ICCV)*, Barcelona, Spain, pp. 1371-1378, 2011.

[13] D. Singaraju and R. Vidal, "Using global bag of features models in random fields for joint categorization and segmentation of objects," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR2011),* Providence, RI, pp. 2313-2319, 2011.

[14] B. Fulkerson, A. Vedaldi, and S. Soatto, "Class segmentation and object localization with superpixel neighborhoods," in *Proceedings of 2009 IEEE 12th International Conference on Computer Vision,* Kyoto, Japan, pp. 670-677, 2009.

[15] C. Galleguillos, B. McFee, S. Belongie, and G. Lanckriet, "Multi-class object localization by combining local contextual interactions," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR2010)*, San Francisco, CA, pp. 113-120, 2010.

[16] S. Kumar and M. Hebert, "Discriminative random fields: a discriminative framework for contextual interaction in classification," in *Proceedings of 9th IEEE International Conference on Computer Vision*, Nice, France, pp. 1150-1157, 2003.

[17] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Proceedings of Advances in Neural Information Processing Systems (NIPS*2006), Vancouver, Canada, pp. 801-808, 2006.

[18] R. Nock and F. Nielsen, "Statistical region merging," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1452-1458, 2004.

[19] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771-1800, 2002.

[20] S. V. N. Vishwanathan, N. N. Schraudolph, M. W. Schmidt, and K. P. Murphy, "Accelerated training of conditional random fields with stochastic gradient methods," in *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, Pittsburgh, PA, pp. 969-976, 2006.

[21] C. Sutton and A. McCallum, "Piecewise training of undirected models," in *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence (UAI2005)*, Edinburgh, Scotland, pp. 1-8, 2005.

[22] S. Z. Li, *Markov Random Field Modeling in Image Analysis*. Tokyo: Springer, 2001.

[23] B. J. Frey and D. J. MacKay, "A revolution: belief propagation in graphs with cycles," in *Proceedings of Advances in Neural Information Processing Systems (NIPS1998)*, Denver, CO, pp. 479-485, 1998.

[24] S. Vicente, C. Rother, and V. Kolmogorov, "Object cosegmentation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR2011)*, Providence, RI, pp. 2217-2224, 2011.

[25] E. Borenstein and S. Ullman, "Combined top-down and bottom-up segmentation," in *Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04)*, Washington, DC, 2004.

[26] M. Everingham, The VOC 2006 database [Internet]. Available: http://www.pascal-network.org/challenges/VOC/databases.html.

[27] L. Zhang and Q. Ji, "Image segmentation with a unified graphical model," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1406-1425, 2010.

[28] D. A. Ross, J. Lim, R. S. Lin, and M. H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 125-141, 2008.

[29] J. Carreira and C. Sminchisescu, "Constrained parametric min-cuts for automatic object segmentation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2010)*, San Francisco, CA, pp. 3241-3248, 2010.

**Lihe Zhang**
received the Ph.D. degree in signal and information processing from Beijing University of Posts and Telecommunications, Beijing, China, in 2004. He is currently an Associate Professor with the School of Information and Communication Engineering, Dalian University of Technology (DUT). His current research interests include computer vision and pattern recognition.

**Yongri Piao**
received the B.S. degree in automation engineering from Jilin University, China, in 2003, and the M.S. and Ph.D. degrees in information and communication engineering from Pukyong National University, Republic of Korea, in 2005 and 2008, respectively. From September 2008 to December 2011, he was a Research Professor at the 3D display research center of Kwangwoon University. Since March 2012, he has been an Associate Professor at the School of Information and Communication Engineering, Dalian University of Technology, Dalian, China. His research interests include optical imaging and 3D display, optical and digital encryptions, 3D pattern recognition and tracking, 2D/3D image processing.