Expert Systems With Applications 63 (2016) 231–240

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Modeling spatial layout for scene image understanding via a novel multiscale sum-product network

Zehuan Yuan[a], Hao Wang[a], Limin Wang[b], Tong Lu[a,*], Shivakumara Palaiahnakote[c], Chew Lim Tan[d]

[a] National Key Lab for Novel Software Technology, Nanjing University, China
[b] Computer Vision Laboratory, ETH Zurich, Switzerland
[c] Faculty of Computer Science and Information Technology, University of Malaya, Malaysia
[d] School of Computing, National University of Singapore, Singapore

ARTICLE INFO

Article history:
Received 19 January 2016
Revised 8 July 2016
Accepted 9 July 2016

Keywords:
Scene image understanding
Spatial layout
Multiscale unary potentials
Multiscale sum-product network
MSPN

ABSTRACT

Semantic image segmentation is challenging due to the large intra-class variations and the complex spatial layouts inside natural scenes. This paper investigates this problem by designing a new deep architecture, called *multiscale sum-product network* (MSPN), which utilizes *multiscale unary potentials* as the inputs and models the *spatial layouts* of image content in a hierarchical manner. That is, the proposed MSPN models the joint distribution of multiscale unary potentials and object classes instead of single unary potentials in popular settings. Besides, MSPN characterizes scene spatial layouts in a fine-to-coarse manner to enforce the consistency in labeling. Multiscale unary potentials at different scales can thus help overcome semantic ambiguities caused by only evaluating single local regions, while long-range spatial correlations can further refine image labeling. In addition, higher orders are able to pose the constraints among labels. By this way, multi-scale unary potentials, long-range spatial correlations, higher-order priors are well modeled under the uniform framework in MSPN. We conduct experiments on two challenging benchmarks consisting of the MSRC-21 dataset and the SIFT FLOW dataset. The results demonstrate the superior performance of our method comparing with the previous graphical models for understanding scene images.

1. Introduction

Natural scene image understanding, which aims at labeling each pixel of a scene image to a predefined object class and simultaneously performing segmentation or recognition of multiple objects that occur in the scene, has been extensively studied in the past years (Farabet, Couprie, Najman, & LeCun, 2013; Gritti, Damkat, & Monaci, 2013; Liu, Xu, & Feng, 2011; Rincón, Bachiller, & Mira, 2005; Shotton, Winn, Rother, & Criminisi, 2009; Tighe & Lazebnik, 2013; Tu, Chen, Yuille, & Zhu, 2005; Yin, Jiao, Chai, & Fang, 2015). However, since even the objects of the same class tend to exhibit large intra-class variations in natural scenes, automatically providing satisfying high-level semantics from complex images is still a very challenging task. With the recent development of vision-based hardware and network techniques, it becomes an active research topic and has attracted more and more researchers in computer vision and pattern recognition community.

Fortunately, in addition to low-level cues, most natural scene images contain intrinsic spatial structures, namely, scene contextual information. Thus the state-of-the-art approaches consider the spatial layout of each scene image as a kind of prior information to improve image understanding results. These methods jointly model the low-level appearances of every single patch and the spatial structures between adjacent patch pairs through a unified framework. A common choice for spatial layout modeling is to resort to graphical models typically like Conditional Random Field (CRF) (Krähenbühl & Koltun, 2011; Ladicky, Russell, Kohli, & Torr, 2009), where nodes are built on image pixels or superpixels, while edges incorporate the second order priors like the smoothness between adjacent nodes. Then the problem of image understanding is treated as Maximum-a-Posterior (MAP) inference in the graphical model. These methods are particularly effective for modeling the relationship between adjacent objects. However, they may not perform well for complex scene images due to

[*] Corresponding author.

E-mail addresses: zhyuan001@gmail.com (Z. Yuan), wanghao0711@gmail.com (H. Wang), lmwang.nju@gmail.com (L. Wang), lutong@nju.edu.cn (T. Lu), hudempsk@yahoo.com (S. Palaiahnakote), tancl@comp.nus.edu.sg (C. Lim Tan).

URL: http://cs.nju.edu.cn/lutong/ (T. Lu)

http://dx.doi.org/10.1016/j.eswa.2016.07.015
0957-4174/© 2016 Elsevier Ltd. All rights reserved.

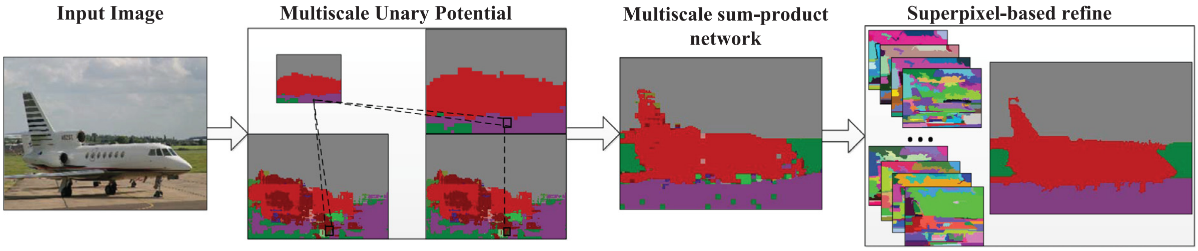


Fig. 1. The pipeline of scene image understanding using the learned MSPN. Left: the original scene image. Middle-left: computing the multiscale unary potentials from the input image. Middle-right: inferring the label for each pixel inside the original scene image by maximizing the posterior with the learned MSPN, which can be conducted efficiently by a two-pass algorithm. Right: refining the results using over-segmented regions.

the following limitations. First, CRF with a grid structure has the ability to utilize adjacent spatial relations, but cannot characterize a wider range of spatial context constraints among non-adjacent scene objects, which sometimes play a more important role than adjacent relations in automatically understanding scene image contents. For example, it is always hard to decide scene content directly from local visual features due to their instabilities brought by intensity, color, texture, illumination, occlusion and viewpoint variations. Instead, by combining a wider range of spatial relations on different scales, scene content understanding results can be greatly improved. Unfortunately, how to combine both the adjacent (short-range) and the nonadjacent (long-range) spatial layouts of image content to enforce the consistency of scene image parsing with such graphical models is still admittedly a hard problem. Second, the inference and training of some complex graphical models may be inefficient, which make the use of graphical models in real-life computer vision related applications inflexible and difficult. For example, CRF with a full connected structure is often hard to infer and train (Farabet et al., 2013). Finally, the state-of-the-art graphical models often face the difficulty on how to integrate high-order object shape priors, which are essential in parsing or understanding image semantics accurately. However, object shape priors sometimes are not well integrated into such models.

In our previous research, we explored scene image co-segmentation by a topic-level random walk framework (Yuan, Lu, & Shivakumara, 2014) and object category discovery in natural scenes using a context-aware graphical model (Yuan & Lu, 2014). To further address the discussed issues in visual scene image understanding, this paper proposes a novel deep architecture named *Multiscale Sum-Product Network* (MSPN), which can be viewed as a stacked sum-product network (SPN) (Poon & Domingos, 2011) to jointly model the distribution over image-level labels and unary potentials from different scene scales. Due to the deep structure of MSPN, the proposed model is able to characterize both the local (short-range) and the global (long-range) spatial relations on different scales from pixel-level, patch-level to image level through a hierarchical manner for better parsing the semantics from complex scene images. Ideally, the combination of both the two types of spatial relations can help understand scene images more accurately since long-range interactions among image patches can be well characterized by MSPN. Additionally, by stacked MSPN, we have the ability to characterize high-order relations among pixels in a flexible and implicit way. To the best of our knowledge, this is the first work by introducing the conceptual deep sum-product network into scene image understanding or content parsing research to reduce the instabilities brought by the unpredictable variations of low-level local visual appearances.

In our architecture, the product operation models various correlations between every two adjacent patches, on which the sum operation further integrates these correlations into the “feature” of a larger patch. On the bottom layer of MSPN, an SPN is designed for each patch to model the joint distribution over the unary potentials and image labels from the previous scale, aiming at modeling the local context information within the image patch. On the up layer of MSPN, a global SPN is proposed for the whole image, aggregating the information from all the SPNs of the patches on the bottom layer, the unary potentials and image labels. Thus, the up layer of MSPN is able to capture long-range interactions among image patches and thereby successfully models the global context information of image content for parsing complex scene semantics more accurately.

In addition to the deep modeling of spatial layouts in every scene, MSPN also allows for efficient inference during the testing phase, which benefits from the fact that the proposed MSPN is a deep tractable model and only contains relatively simple *product* and *max* operations. This is particularly useful for designing real-life systems with a much lower computational load. We show the overall pipeline for understanding the semantics of an unknown scene image in Fig. 1, where we first compute the multiscale unary potentials (middle-left in Fig. 1), and then infer the label for each pixel inside it by maximizing the posterior with the learned MSPN (middle-right), which can be conducted efficiently by a two-pass algorithm. Finally, scene image understanding results will be further refined by using the over-segmented region information from the original scene image (right).

Our main contributions are two-folds: (1) a novel deep network framework named MSPN is proposed to perform semantic image segmentation by a more effective and efficient way comparing with the popular graphic models; and (2) the architecture of MSPN is elaborately designed to model multi-scale features, either local or global spatial layout of any scene image, and higher-order priors under an unified framework. The results on two popular benchmarks show that the proposed MSPN addresses semantic image segmentation effectively.

The rest of the paper is organized as follows. Section 2 discusses the related work. In Section 3, we introduce the structure of MSPN. Section 4 gives the training and inference methods for scene image understanding. Experimental results and discussions are given in Section 5, and finally Section 6 concludes the proposed model.

2. Related work

The problem of image understanding or parsing has been extensively studied in the previous research, and the existing approaches can be roughly classified into three categories, namely, bottom-up scoring, top-down refinement, and region label reasoning.

In *bottom-up scoring methods*, a fairly large number of object hypotheses is first generated and then low-level color, texture and shape features are used on these segments for classifying object regions. For example, Gu, Lim, Arbelaez, and Malik (2009) present a unified max-margin framework for object detection, segmentation, and classification using region-based features, from which the shape and scale information of objects are naturally encoded. In their recent work (Arbelaez et al., 2012), the problem of segmenting and recognizing scene objects is performed by producing class-specific scores for bottom-up regions and aggregating the votes of multiple overlapping candidates through pixel classification. However, bottom-up segmentation at object level is still an ill-defined problem since shape and texture in regions tend to exhibit large intra-class variations in their appearances. Carreira, Li, and Sminchisescu (2012) present a pipeline that combines multiple figure-ground hypotheses, which are generated by bottom-up computational processes without exploiting the knowledge of specific categories. Differing from other methods, image parsing is formulated as a regression problem, producing a globally consistent ranking with close ties to segment quality. Similarly, (Vijayanarasimhan & Grauman, 2011) propose a branch-and-cut strategy by determining the subset of spatially contiguous regions whose collective features will maximize a classifier's score in over-segmented images (Farabet et al., 2013; Shotton et al., 2009; Tighe & Lazebnik, 2013; Tu et al., 2005).

As for humans object recognition and analysis in image understanding are heavily interwined, thus *top-down refinement methods* segment an image using shape priors predicted by statistical shape models. Leibe and Schiele (2003) use a probabilistic formulation to incorporate knowledge about the recognised category as well as the supporting information in the image to segment objects from the background. Malisiewicz, Gupta, and Efros (2011) learn a separate classifier for each exemplar that is represented using a grid HOG template, aiming at combining the effectiveness of a discriminative object detector with the explicit correspondence offered by a nearest-neighbor scheme. However, the class and even the coarse pose of an object as well as its existence are actually strong assumptions and thus can be hardly satisfied in practice. Sometimes top-down information is also obtained from contemporary object detectors. For example, contemporary object detectors, each of which has a rich part structure to provide an excellent basis for top-down image parsing, are run on a subsampled grid for object segmentation in (Brox, Bourdev, Maji, & Malik, 2011). Note that in top-down methods, bottom-up cues are often accompanied towards higher precision. Han and Zhu (2009) present a generative representation for man-made scene objects such as buildings, hallways, kitchens and living rooms using attribute graph grammar, in which the bottom-up step detects an excessive number of rectangles as weighted components and top-down predications of occluded or missing components are activated through a group of grammar rules.

Region label reasoning methods use different kinds of models to ensure the consistency of labels during parsing (Brox et al., 2011; He, Zemel, & Carreira-Perpiñán, 2004; Kohli, Ladicky, & Torr, 2008; Kohli, Osokin, & Jegelka, 2013; Ladicky et al., 2009). Shotton et al. (2009) propose to use CRF model to jointly model patch texture, layout and context for image labeling. Hierarchical connection (He et al., 2004) or higher-order potentials (Kohli et al., 2008) to enforce label consistency have also been incorporated to improve the accuracy. Recently, Krähenbühl and Koltun (2011) propose a fully connected CRF with Gaussian edge potentials together with a highly efficient approximate inference algorithm. In addition to CRF, generative models such as MRF have also been explored (Kohli & Kumar, 2010). Additionally, Todorovic and Nechyba (2007) address the problem of object detection and recognition in complex scenes by generative dynamic tree-structure belief networks. In the recent work (Steinberg, Pizarro, & Williams, 2015), hierarchical Bayesian models for unsupervised scene understanding are also proposed. Recently, deep models have also been deployed in exploring the labeling problem by modeling image priors. In (Eslami, Heess, & Winn, 2012), Shape Boltzmann Machine (SBM) is presented for modeling binary object masks. For example, Chen, Yu, Hu, and Zeng (2013) use the deep Boltzmann machine to learn the hierarchical architecture of shape priors. Sum product network is a new deep architecture proposed for modeling the probability distribution with variables as leaves, while sum and product operations are described as internal nodes and weighted edges (Poon & Domingos, 2011). The learning and inference of such a network is much faster and more accurate than other deep models. Sum product networks have been adopted in particular vision computation tasks like (Amer & Todorovic, 2012; Gens & Domingos, 2012; Luo, Wang, & Tang, 2013; Poon & Domingos, 2011). For example, Poon and Domingos (2011) propose the SPN and verified its superior performance to other deep models in image completion. Gens and Domingos (2012) present a discriminative training algorithm for the generative SPN. They demonstrate the advantages of discriminative learning of SPN in the image classification task. Amer and Todorovic (2012) use SPN to model the stochastic structure in videos and adopt this method to action recognition. Luo et al. (2013) propose a deep sum-product architecture to model the correlations among facial attribute and proved its effectiveness in robust attribute estimation. However, none of these research focuses on exploring image semantics for scene content understanding.

3. Multiscale sum-product network for scene understanding

To model both the short-range scene context for every image patch and the long-range scene context among different patches, we put forward MSPN to characterize these correlations. Essentially, our MSPN can be considered as a stacked SPN as introduced. The SPN on the bottom layer aims at modeling local context for each scene image patch, while the SPN on the upper layer makes use of the output of the bottom layer, namely, local correlations inside patches, and the unary potentials on this scale to characterize long-range correlations among patches. The process can be iteratively repeated until each patch is corresponded to the overall image, and the SPN on the lower layer can be regarded as the input of the upper SPN in the proposed network. The iterative process here indicates that the patches on the lower layer will be composed into larger ones on the higher layer, and the interaction relationships will be modeled accordingly. This process is iteratively repeated until finally there is only one patch, namely, the whole image, on the top layer.

Note that both the short-range relations inside every scene image patch and the long-range relations among patches are modeled by different SPNs, respectively. The long-range interaction is modeled in the middle layers of the upper-level SPN. That is, the interaction of two patches is modeled by the connection of their ancestral product nodes to a common sum node in a hierarchical way. In another word, the joint probability over $\{\mathbf{Y}, \mathbf{\Phi}\}$ is

$$P(\mathbf{Y}, \mathbf{\Phi}) = \sum_{\mathbf{i_1} \in \{0,1\}^N, \mathbf{i_2} \in (0,1)^N} \big(P(\mathbf{i_1}, \mathbf{i_2}) \cdot \mathbf{1}(\mathbf{Y} = \mathbf{i_1}) \cdot \mathbf{1}(\mathbf{\Phi} = \mathbf{i_2})\big) \qquad (1)$$

where $\mathbf{Y}$ and $\mathbf{\Phi}$ denote the labels and the unary potential variables inside a patch, respectively, and P is represented by the *network polynomial* (Darwiche, 2003). Here $\mathbf{i_1}$ and $\mathbf{i_2}$ denote all possible values of $\mathbf{Y}$ and $\mathbf{\Phi}$, respectively. Thus as in (Poon & Domingos, 2011), the distribution P can be represented by an SPN, namely, a rooted directed acyclic graph, in which the leaves are indicators $\mathbf{1}(x_i = 1)$ and $\mathbf{1}(x_i = 0)$, where x_i is any variable in $\mathbf{Y}$ or $\mathbf{\Phi}$, the internal nodes of which are sums and products. Each edge (i, j) emanating from a sum node i has a non-negative weight ω_{ij}. The value of a product node is the product of the values of its children. Then the value of a sum node is $\Sigma_{j \in Ch(i)} \omega_{ij} v_j$, where $Ch(i)$ is the children of i and v_j is the value of node j. The value of an SPN is thus the value of its root. Thereby, the key to model the short-range or long-range correlations inside each complex scene is converted to

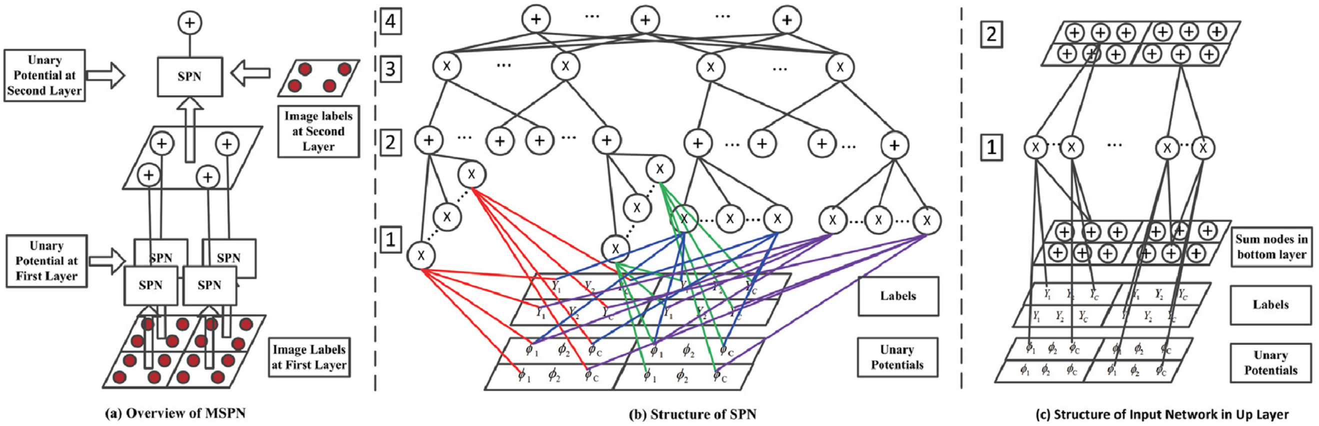

Unary Potential at Second Layer
SPN
Image labels at Second Layer
Unary Potential at First Layer
SPN
SPN
SPN
SPN
Image Labels at First Layer
(a) Overview of MSPN
4
3
2
1
Y_1
Y_2
Y_C
ϕ_1
ϕ_2
ϕ_C
Labels
Unary Potentials
(b) Structure of SPN
2
1
Sum nodes in bottom layer
Labels
Unary Potentials
(c) Structure of Input Network in Up Layer

Fig. 2. The illustration of MSPN. Left: we give an overview of MSPN for modeling the joint distribution over image labels and unary potentials from multiscales. MSPN can be divided into two layers, where the bottom layer mainly models the correlations within each local patch, and the up layer captures the interactions among patches. Middle: we provide a detailed description of the SPN structure on the bottom layer. It is organized in a hierarchical manner, modeling the correlations from adjacent pixels to the whole patch. Right: the difference between the SPNs on the up layer and the bottom layer lies on the network inputs and the number of root nodes. We give the network structure of the up layer to handle three inputs: image labels, unary potentials, and the sum nodes from bottom layer.

the problem of how to construct the SPN for each layer. On the other side, if we have joint probability $P(\mathbf{Y}, \mathbf{\Phi})$ over $\{\mathbf{Y}, \mathbf{\Phi}\}$ for any entire image, the scene understanding task will thereby be turned out to be the inference of the posterior distribution $P(\mathbf{Y}|\mathbf{\Phi})$ since Φ of each scale denotes observable variables. We will give the details on multiscale unary potentials, the proposed multiscale sum-product network, and the pipeline for scene image understanding in the following sub-sections.

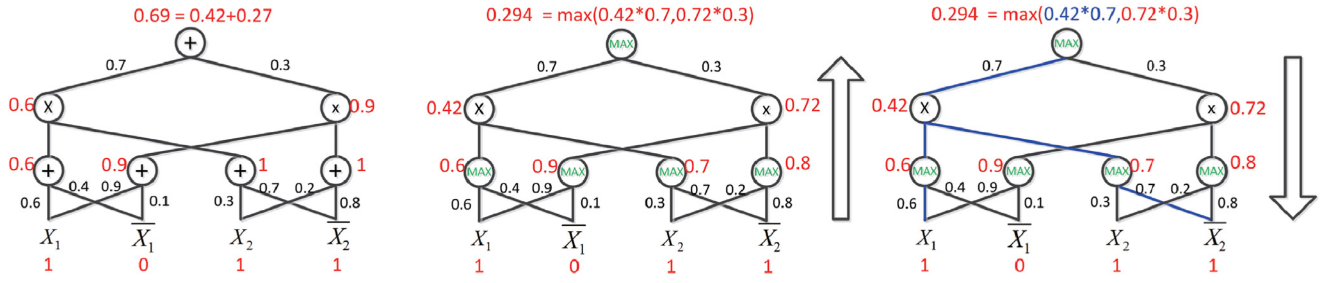
3.1. Obtaining multiscale unary potentials

Unary potentials in MSPN evaluate the compatibility of a label y_c assigned to a local scene structure (superpixel, pixel or patch), which is usually modeled by classifier responses. Traditional unary potentials are designed only on the raw scale, which is sensitive to the variations of visual appearances and may be greatly influenced by semantic ambiguity. Intuitively, unary potentials incorporating multiple scales can leverage the different context around each local image structure, and hence assist making more precise predictions on their high-level semantics. In general, any pixel-level unary potential can be adopted into the proposed MSPN framework. We choose the Textonboost (Shotton et al., 2009) technique to build our multiscale unary potentials. Specifically, we first reshape all the training images into pyramids with different scales of $s \in \{1, 2, \ldots S\}$, where S is the number of scales. Then we train a number of pixel-level Textonboost classifiers on each of these scales. For every input image, we collect the response maps from different $\{\phi^s\}_{s=1}^S$ as our multiscale unary potentials. The response maps from the raw-scale produce the pixel-level unary potentials and those from coarser-scales are used to generate patch-level unary potentials.

3.2. Multiscale sum-product network structure

In our framework, we use a two-layer MSPN as shown in Fig. 2 for modeling the joint distribution over image labels and unary potentials from multiscales. That is, the proposed MSPN can be divided into two layers, where the bottom layer mainly models the correlations for each local patch, and the up layer captures the interactions among patches. The overview of the proposed MSPN is shown by the left of Fig. 2. We also provide a detailed description of the SPN structure on the bottom layer. It is organized in a hierarchical manner, modeling the correlations from adjacent pixels to the whole patch (middle of Fig. 2). The difference between the SPNs on the up layer and the bottom layer lies on the network inputs and the number of root nodes. That is, on the bottom layer of MSPN, SPNs are used to model the distributions of labels and potentials in different disjoint $N \times N$ patches. In each SPN, the bottom layer models the correlation of two adjacent pixels (corresponding to a 1×2 patch) by different product nodes. On the second layer, these product nodes in the 1×2 patch connect to some sum nodes representing different mixtures of the joint distribution in the 1×2 patch (like Gaussian mixture). The two-stage construction is iterated to combine any smaller adjacent patches until it reaches the root $N \times N$ patch. We also show the graphical network structure of the up layer to handle three inputs: image labels, unary potentials, and the sum nodes from the bottom layer (right).

Bottom layer representation. The middle of Fig. 2 gives a detailed description about the structure of the SPN used by the bottom layer of MSPN. It models the joint distribution over pixel labels and unary potentials within each local patch. We choose the patch size as 4 and suppose there are altogether C object classes. The inputs are the label indicator function $\{Y_1, Y_2, \ldots, Y_C\}$ and the normalized unary potentials $\{\phi_1, \phi_2, \ldots, \phi_C\}$ of each pixel. Regarding the structure of the network, it is organized in a hierarchical manner, modeling the correlations from adjacent pixels to the whole patch. *On layer 1*, we use the product node to model the correlations of labels and potentials from adjacent pixels. Here the label indicator function is coupled with its corresponding unary potentials, i.e., we only consider the correlation between the label indicator function and its corresponding unary potential. Thus, for a pair of adjacent pixels, there are C^2 product nodes to model the correlations. For a patch of size 4, there are totally 4 pairs of adjacent pixels. Thus, the number of the product nodes on *layer 1* is $4 \times C^2$ and each product node falls into a 1×2 patch. *On layer 2*, we use sum nodes (in 1×2 patches) to model the mixture of different kinds of correlations on the same pixel pair by connecting all the product nodes from the same patch. The number of the sum nodes on this layer can be set manually. *On the above layer*, similarly a product node connects the sum nodes from the adjacent patches on the previous layer, while a sum node connects the product node belonging to the same patch. This process can be repeated until the root node corresponding to the whole patch are reached. Intuitively, each sum node belongs to an image patch and can be viewed as the “feature” of this patch, while each product node connects two adjacent pixels (or patches) and models the correlations between them.



(a) calculation of marginal probability

(b) bottom-up of MPE

(c) top-down of MPE

Fig. 3. An example of the deep model over two random variables $\{X_1, X_2\}$. The left figure (Poon & Domingos, 2011) shows the calculation of marginal probability $P(X_1 = 1)$. The other two figures show the two passes for inferring the unobserved variable X_2 using most probable explanation (MPE) when X_1 is observed as 1.

Up layer representation. The upper layer of MSPN models the correlations among patches. It takes the unary potentials and the image labels on the second scale, and the sum nodes from the bottom layer as the inputs. In essence, the structure of the network on this layer is quite similar to that of the bottom layer except that there is only one root node, which models the interactions between patches in a hierarchical way from the adjacent patches until to the whole scene. Considering the sum nodes from the bottom layer, we handle this input separately for the up layer. The right of Fig. 2 gives the detailed illustration of the network structure for handling the inputs. That is, on *layer 1*, we use product nodes to model the correlations between the coupled image labels, the unary potentials, and one of its corresponding sum nodes from the bottom layer. Then on *layer 2*, we use sum nodes to connect the product nodes from the same patch. In addition to the difference of handling the inputs and the number of root nodes, the structure of the SPN on the up layer is the same with that of the bottom layer.

Note that there should not be over-counting in the proposed MSPN model. First, the proposed MSPN is more like a tree structure built on the bottom layer, which is used to predict pixel labels. Since MSPN is purely a semantic model, it can be combined with segmentation-based refinement to supplement precise contour information, which is not used to merge predictions. Second, we count multiscale labels combined with corresponding unaries to train the proposed MSPN model. For each input layer, the labels represent the evidence from a specific scale. For other layers, all the combinations of the labels from lower layers are modeled.

3.3. Scene image understanding via MSPN

For scene image understanding, we first generate the multiscale unary potentials for any input unknown scene image using the method described in Section 3.1. Note that during the training phase, we jointly model the distribution $P(\mathbf{Y}, \Phi)$ over multiscale image labels $\mathbf{Y}$ and multiscale unary potentials Φ using MSPN. Then during testing phase, we infer image labels given the multiscale unary potentials Φ:

$$\mathbf{Y}^* = \arg\max_{\mathbf{Y}} = P(\mathbf{Y}|\Phi) = \arg\max_{\mathbf{Y}} \frac{S(Y, \Phi)}{S(\Phi)_{\mathbf{Y}=1}} \qquad (2)$$

where $S(Y, \Phi)$ is MSPN value, and $S(\Phi)_{\mathbf{Y}=1}$ is a constant when given a specific Φ by marginalizing over $\mathbf{Y}$. Since the structure of our MSPN is also composed of interleaved *sum* and *max* nodes, the Most Probable Explanation (MPE) can be efficiently conducted by a two-pass algorithm. Finally, we refine the inference results using the superpixel cues as shown in Fig. 1.

The details are as follows. We use an oracle example to show how the two-pass algorithm works. Fig. 3 shows an example of the deep network for two variables $\mathbf{X} = (X_1, X_2)$, where we denote the indicator function $\mathbf{1}(x_i = 1)$ as X_i and $\mathbf{1}(x_i = 0)$ as $\overline{X_i}$. To start, we show how the network is convenient to compute the marginal probability. For instance, when X_1 is observed as 1 and X_2 is unobserved, the marginal probability $P(X_1 = 1) = S(1, 0, 1, 1) = 0.7 * (1 * 0.6 + 0 * 0.4) * (1 * 0.3 + 1 * 0.7) + 0.3 * (1 * 0.9 + 0 * 0.1) * (1 * 0.2 + 1 * 0.8) = 0.69$. In additional to marginal probability, we can also efficiently infer the unobserved variable using MPE in the network. We first replace the SUM node with the MAX node in the network and then conduct a two-pass inference procedure. In the first pass, we compute the value of each node in a bottom-up manner and in the second pass, we backtrack along the path that the MAX operation selects until the leaf node is reached. For the example, in Fig. 3, the MPE of X_2 is 0 when $X_1 = 1$.

Finally, the image understanding results can also be refined by oversegmentations on the original scene image. This is because both the multiscale unary potentials and MSPN focus on making pixel-level or patch-level predictions, which usually contain noises and lack precise delineation of objects. Superpixels generated from bottom-up segmentation can help capture object boundaries and maintain local consistency, which are visually perceptive and complementary for our MSPN results. Since the bottom-up segmentation suffers the lack of semantics, some of its superpixels may contain several object parts. To obtain proper superpixels, we vary the parameters of the segmentation algorithm to generate multiple overlapping superpixels. Then each pixel is contained by several superpixels, which forms multiple local context to analyze our MSPN results. Thus the pixel label prediction can be refined as:

$$\mathbf{y}'(j) = D(Q^*), \quad Q^* = \arg\min_{Q, j\in Q} E(Q)/\log(A(Q)) \qquad (3)$$

where j is the pixel for predicting contained by superpixel Q, $E(Q)$ is the entropy of the distribution of the MPSN predicting labels within Q, $A(Q)$ is the area of Q, and $D(Q)$ is the dominant label within Q. Note that for each superpixel containing the target pixel, we compute the score and select the minimum directly. Then the predicted label is refined by the dominant one within the selected superpixel. Additionally, the SPNs of each layer in the proposed MSPN actually model the distribution of patches with a specific scale. Therefore, SPNs of smaller patches are input to those of the larger patches that cover the smaller ones. This is our intention to design the proposed MSPN since SPNs of different levels can model detailed and specific characteristics of corresponding granularity.

4. Learning multiscale sum-product network

In the proposed MSPN, generally there are a large number of nodes and densely connected edges. For a certain dataset with images $I = \{I_i\}_{i=1}^N$, corresponding groundtruth $\{\mathbf{Y}_i\}_{i=1}^N$ and multiscale unary potentials $\{\Phi_i\}_{i=1}^N$, our goal is to obtain the learned MSPN that is parameterized by proper weights and structures to best explain the training dataset. For this purpose, we learn the MSPN in a generative fashion, aiming at maximizing the following log likelihood:

$$\omega = \arg\max_{\omega} \sum_{i=1}^{N} \log P(\mathbf{Y}_i, \Phi_i; \omega) \tag{4}$$

where the joint probability $P(\mathbf{Y}_i, \Phi_i; \boldsymbol{\omega})$ is modeled by MSPN. The learning algorithm is summarized in Algorithm 1, in which the details are shown. Note that "renormlize" in step 9 is defined as normalizing the sum of edge weights connected to the same sum node to 1, and "convergence" in step 11 has the following two criterion. The first is that the difference of current joint probability and previous one is below a pre-defined threshold. The second is that the number of iterations is beyond a pre-defined max one.

Algorithm 1. Learning MSPN.

```
Input: Training images $I = \{I_i\}$, and corresponding groundtruth $\{\mathbf{y}_i\}$
Output: MSPN with learned structure and parameters $\omega$
1: for all $I_i \in I$ do
2:     Compute multiscale unary potential $\Phi_i = \{\phi_i^s\}_{s=1}^S$
3:     Generate multiscale image labels $\mathbf{Y}_i = \{\mathbf{y}_i^s\}_{s=1}^S$
4: end for
5: Initialize $\omega$ and network architecture
6: repeat
7:     Activate MSPN with $\Phi_i$ and $\mathbf{Y}_i$ by a upward pass
8:     E-step: Infer the MPE state of $\mathbf{Y}_i$ by a downward pass
9:     M-step: Renormlize $\omega$
10: until convergence
11: Prune edges with zero weights
```

Given a specific image goundtruth pair $(I_i, \mathbf{y}_i)$, we compute the multiscale unary potential $\Phi_i = \{\phi_i^s\}_{s=1}^M$ and multiscale groundtruth $\mathbf{Y}_i = \{\mathbf{y}_i^s\}_{s=1}^S$, where $\mathbf{y}_i^s (s \geq 2)$ on the coarser-scale are considered as the patch-level groundtruth on the raw scale $s = 1$, which is generated by the distribution of $\mathbf{y}_i^1$ within the patch. MSPN can then be learned with a hard EM algorithm following (Poon & Domingos, 2011). Initially, we maintain a count for each edge protruding from sum nodes and each sum node maintains the total count of its edge counts. Next, the proposed MSPN is activated with $\{\Phi_i, \mathbf{Y}_i\}$ by an upward pass, where the root value is $P(\mathbf{Y}_i, \Phi_i; \boldsymbol{\omega})$. Then the E-step actually executes an MPE inference as in Section 3.3 by a downward pass. In the M-step, the counts of the edges in the inference patches are increased by 1 and the weights are re-normalized as the ratio of the counts to maintain the probabilistic explanation. Note that since the MSPN interleaves sum nodes and products, the inference can be actually achieved by the two-pass algorithm the same as the original SPN.

The edge prune process in the last step of the loop is required. The arbitrary combination of adjacent pixels or patches will bring in a densely connected structure. For a dataset with the images of size $N \times N$ and C class labels, there will be roughly N^2C^2 product nodes on the bottom layer of MSPN and VN^2C^2 (V is the number of the sum nodes in one patch) edges connected to their parent sum nodes. Generally, since the total number of the edges is far beyond that of the patches in each image, the edge prune process will assist us removing zero weight edges and non-parent nodes, which saves the memory space and keeps the learning algorithm efficient. Specifically, we prune the edges after the full EM iteration by eliminating both the edges that connect sum nodes and product nodes with zero weights, and the edges without connecting to any parent node.

Note that in the proposed model, we do not distinguish the two kinds of variables, namely, $\mathbf{Y}$ and $\boldsymbol{\Phi}$, and treat both of them as continuous ones. The weights of edges are also continuous ones, which seem more like posterior probabilities of mixture components. The sum and production operation do not change and thus the inference and learning are the same as SPN.

5. Experiments

We evaluate our method on two benchmarks consisting of the MSRC-21 dataset (Shotton et al., 2009) and the SIFT FLOW dataset (Liu, Yuen, & Torralba, 2009). As both these two datasets contain a number of object classes organized with particular spatial layouts, they are very suitable for evaluating the proposed scene understanding framework. The MSRC-21 dataset consists of 591 color images with the size of 213 × 320 pixels and the corresponding groundtruth labels of altogether 21 classes. The SIFT FLOW dataset is composed of 2688 color images with the size of 256 × 256 pixels and 33 class labels roughly labeled by LabelMe users.

Implementation details. We use the Textonboost (Shotton et al., 2009) method to generate our multiscale unary potentials. In our experiments, we choose the scale as 2 since it is compatible with our two-layer stacked MSPN. The first scale is for the raw data, while the second scale is set as the quarter of the first one, which means 60 × 80 pixels for each of the images in the MSRC-21 dataset and 64 × 64 pixels for each of the images in the SIFT FLOW dataset on the second scale. We use four kinds of features, namely, color (3 channel features on the Lab space), hog, location (by relative coordinates), and texture (the responses from Gaussian filters) to train two texton dictionaries with respectively 128, 150, 144 and 400 entries. The dictionaries are trained separately on two different scales. Then the images are quantized into textons to train the Textonboost classifier. The resulting scores are finally transformed into probabilities to feed into the proposed MSPN model.

Due to the dense structure of MSPN, we first conduct down-sampling for the multiscale unary potentials to reduce the possible memory cost in computations. Specifically, the sizes of the first scale unary potential are down-sampled as 120 × 160 pixels for the images in the MSRC-21 dataset and 64 × 64 pixels for the images in the SIFT FLOW dataset. The sizes of the unary potentials on the second scale are 30 × 40 and 16 × 16 for these two datasets, respectively. Note that we set the patch size as 4 × 4 for both the two datasets. During testing, the results of MSPN are up-sampled into the original scale for evaluation and comparison.

The number of the sum nodes in MSPN for input layers (pixel or patch) is fixed to the number of classes, and the root layer (image) has only one sum node. For remaining layers, we set the number of the sum nodes as 25 to increase the descriptive power of MSPN and preserve the high efficiency as well. In the superpixel refinement stage, we use (Felzenszwalb & Huttenlocher, 2004) to generate superpixels due to its efficiency and shape preserving property. Totally, we use 27 groups of fixed parameters to generate superpixel maps. To study the effectiveness of MSPN, we compare it with other two typical graphical model baselines, namely, pairwise CRF and robust P^N CRF (Kohli et al., 2008). To verify the role of the stages in our scene image understanding pipeline, we also compare the proposed MSPN with only one single scale (S-MSPN) and the MSPN without superpixel refinement (N-R-MSPN, namely, Non-Refinement-MSPN). Note that the maximal number of SPN layers is related to image resolution. Other hypeparameters are adjusted on a held-out dataset sampled from training images. Note that we do not use the held-out dataset in training stage.

Results on the MSRC-21 dataset. We follow the same experiment evaluation scheme in (Shotton et al., 2009) for evaluations. The multiscale unary potentials are learned on the training set. We use two metrics to evaluate the parsing results, namely, the global pixel-wise classification accuracy and the per-category classification accuracy. We also test our method by gradually adding each step in the pipeline and comparing with the two pairwise CRF

Table 1
Scene image understanding results on the MSRC-21 dataset. For each category, the pixel-wise classification accuracy is provided as well as its average. The last column provides the global pixel-wise classification accuracy. Bold entries indicate the best performances. We evaluate our method by adding each element in the pipeline and comparing with two other baselines and the other three state-of-the-art graphical model based methods.

	Building	Grass	Tree	Cow	Sheep	Sky	Airplane	Water	Face	Car	Bicycle	Flower	Sign	Bird	Book	Chair	Road	Cat	Dog	Body	Boat	Average	Global
Unary^{coarse}	63	86	79	61	62	84	69	67	72	67	73	89	78	25	93	38	74	81	56	69	21	67.0	74.6
Unary^{fine}	72	98	**90**	84	80	93	82	68	88	84	91	91	70	47	94	59	89	75	46	**81**	25	76.7	84.2
S-MSPN	74	96	87	82	82	92	84	68	90	**88**	90	96	77	41	**97**	78	87	**83**	56	67	30	78.4	84.5
MSPN	72	96	89	86	83	93	83	68	**92**	86	90	96	78	51	**97**	72	88	82	56	74	29	79.2	85.0
N-R-MSPN	74	98	**90**	84	82	96	80	70	91	86	91	**98**	81	49	**97**	74	89	79	59	74	26	79.4	86.1
Pairwise CRF	73	**99**	88	78	76	95	76	72	87	84	87	93	75	45	95	57	91	74	44	77	20	75.5	84.5
P^N CRF	73	98	**90**	**85**	81	94	82	68	89	85	91	92	71	48	94	60	89	76	47	**81**	24	77.1	84.6
Hierarchical CRF	80	96	86	74	87	99	**94**	87	86	87	82	97	95	30	86	31	**95**	51	59	66	09	75	86
Ladicky et al. (2010)	**82**	95	88	73	**88**	**100**	83	**92**	88	87	88	96	**96**	27	85	37	93	49	**80**	65	20	77.0	**87.0**
Boix et al. (2012)	66	87	84	81	83	93	81	82	78	86	**94**	96	87	48	90	**81**	82	82	75	70	**52**	**80.0**	83.0
Yao et al. (2012)	71	98	**90**	79	86	93	88	86	90	84	**94**	**98**	76	**53**	**97**	71	89	**83**	55	68	17	79.3	86.2
Lin and Xiao (2013)	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	74.0	–
Ravì, Bober, Farinella, Guarnera, and Battiato (2016)	42	92	77	**88**	**92**	92	85	63	91	72	77	73	34	30	92	45	80	78	37	57	24	68	80

and robust P^N CRF baselines. Besides, we compare our results with other three start-of-the-art graphical model based algorithms, that is, Hierarchical CRF with co-occurrence potentials (Ladicky, Russell, Kohli, & Torr, 2010), Harmony potentials (Boix et al., 2012), and the scene analysis method proposed in the recent research (Yao, Fidler, & Urtasun, 2012).

The quantitive results are shown in Table 1. We first explore the effectiveness of the introduction of the deep architecture into scene image understanding. From the comparisons between unary potentials and S-MSPN, we observe that using the deep architecture effectively improves the performances of label prediction both on the global pixel-wise accuracy and the average per-category accuracy. We also study the complementary property of unary potentials from different scales. We find that although the coarse-scale (patch-level) unary potential is not precise, it can provide extra hints for some categories (e.g., cat, dog, and sign). We then compare the performances between MSPN and S-MSPN, and can conclude that MSPN outperforms the latter on both the global pixel-wise accuracy and the average per-category accuracy. It indicates the multiscale representation, including multiscale unary potentials and multiscale modeling spatial layout, is an important and essential cue for improving the accuracy of natural scene image understanding. Finally, we demonstrate the usefulness of the proposed superpixel based refinement step, which can obtain 1.1% improvement for the global accuracy averagely. This is because the parsing results of N-R-MSPN is not so smooth and sometimes probably contains noise predictions inside objects or around boundaries, which may be refined by the homogeneous superpixels. Although the influence of superpixel based refinement varies across different object classes, it is able to increase the average per-category accuracy according to our experiments on the MSRC-21 dataset. Some example results are given in Fig. 4. Note that the refined MSPN has better pixel accuracy along boundaries, but it also has side effects that the small occupied objects may be fused with into larger objects and thus lose true labels. As a whole, the refinement will improve the MSPN performance.

In the second group of experiments, we compare our method with two other bottom-up methods, namely, pairwise CRF, robust P^N CRF and hierarchical CRF (Ladicky et al., 2009). The former smooths label assignment within an edge window by evaluating color contrast, and the latter builds high order potentials on pixel cliques indicating by superpixels. Both of them focus on improving the consistency and pursue high-quality segmentation boundaries. In Fig. 4, it can be seen that robust P^N CRF can achieve high consistency segmentations, but can not correct the mis-predictions of unary potentials. Since our MSPN can model object shapes and their spatial layouts, the missing parts (e.g., the cow, the head of a human and the boat in Fig. 4) can be partly recovered. Additionally, our MSPN learns high-order relations automatically in training stage and perform better than those with empirical definition of higher-order priors. As a result, the average per-category accuracy of the proposed MSPN outperforms the other three methods by 2% averagely.

Finally, we compare our method with three other state-of-the-art methods (Ladicky et al., 2010), Boix et al. (2012) and (Boix et al., 2012). These methods also use the CRF framework, but with more elaborately designed unary potentials. For example, Ladicky et al. (2010) model the co-occurrence between object classes, Boix et al. (2012) incorporate class interactions, while (Yao et al., 2012) integrate scene classification and object detection tasks. Comparing with (Ladicky et al., 2010) that obtained the best global accuracy, our method outperforms it on more than a half categories (e.g., cat, chair, and body). Our method also performs better than (Boix et al., 2012), which obtained the best average per-category accuracy. In addition, we compare the proposed method with two recent works on MSRC-21 for semantic image segmentation. Their public results are used for comparison. We can find that the performance of the proposed method is still better. Note that the performance of (Lin & Xiao, 2013) is far lower than those of the others. The main reason is that this method is a generative model without exploiting supervised information. Thus we can say the proposed MSPN outperforms the recent methods for understanding scene images. From the results, we can also foresee that unifying all the cues are better than modeling only a part of them. We also find that handcrafted higher-orders are sometimes inferior to automatically discover knowledge among images. Therefore, modeling all the constraints between labels and appearances together may be a better way to segment images, rather than those efforts only on learning representative features to increase performances.

Results on the SIFT FLOW dataset. We also test our method on the SIFT FLOW dataset following the same evaluation scheme proposed in (Liu et al., 2009). This dataset has 33 object classes in total. However, the class distribution in this dataset is very unbalanced. For example, the background elements (e.g., mountain, sky, tree, grass) occupy a large ratio of the images, while several object classes (e.g., bird and bus) appear much less frequently. The experimental results are shown in Table 2. Due to class unbalance, Textonboost unary potential performs a bit worse on the dataset with the global pixel accuracy of 69.7%. However, with the proposed MSPN, we can improve the global pixel accuracy to 72.9% and precisely predict most large scale scene objects that have regular spatial layouts inside the scene (e.g., building, sea, and mountain). The performance of the proposed MSPN is still superior to

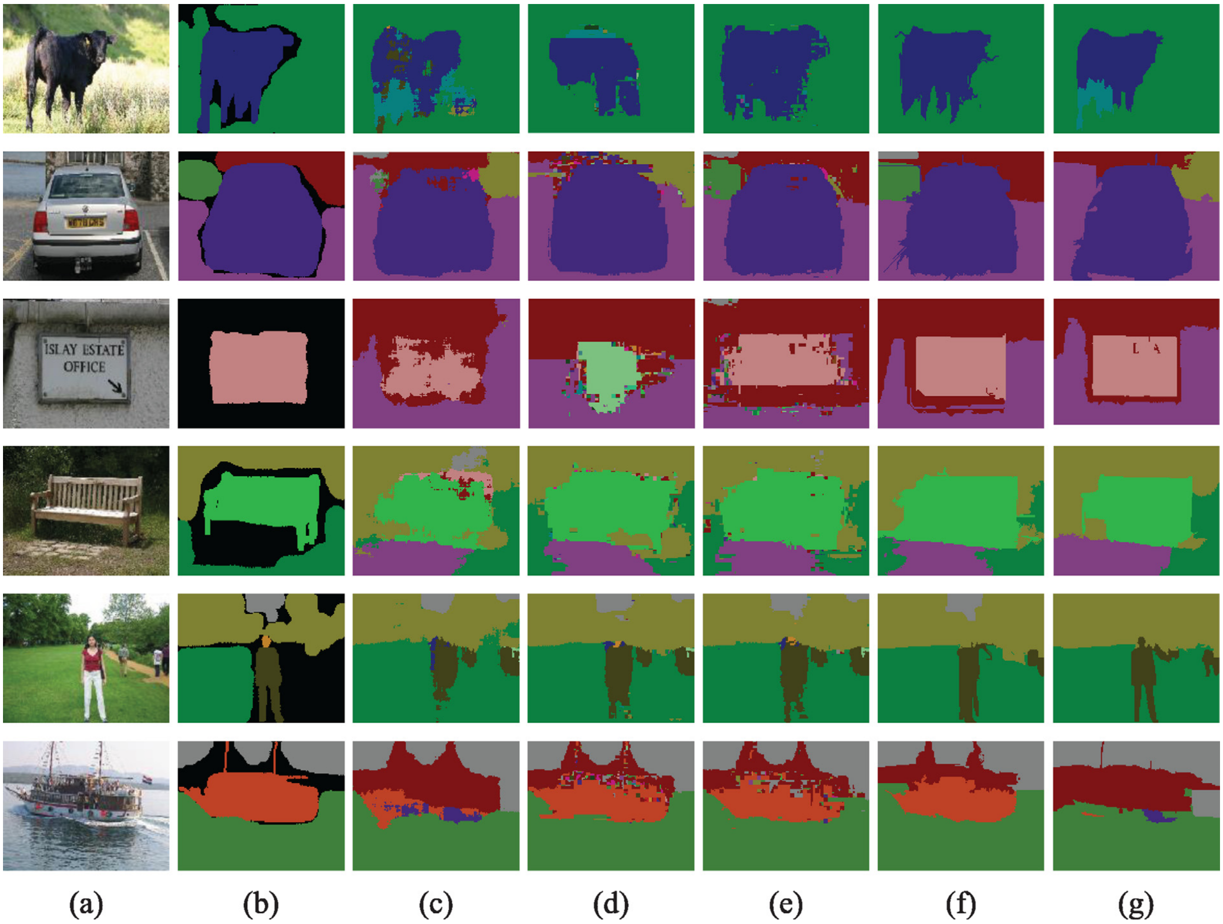


Fig. 4. Scene image understanding result examples from the MSRC-21 dataset. Note we only show the raw-scale unary potential for the limited space. (a) Original images, (b) the groundtruth, (c) unary potential, (d) S-MSPN, (e) the proposed MSPN, (f) N-R-MSPN, and (g) Robust P^N CRF.

Table 2
Scene image understanding results on another benchmark dataset SIFT FLOW. Pixel-level accuracies for scene understanding are provided both for our method and the two baselines, where bold entries indicate the best performance.

	Unary^{coarse}	*Unary*^{fine}	S-MSPN	MSPN	N-R-MSPN	Pairwise	P^N	Hierarchical
Pixel Accu.	69.7	66.4	70.9	71.2	**72.9**	70.8	72.3	72.5

Table 3
The comparison of time complexity per image on SIFT FLOW dataset.

	MSPN	Pairwise CRF	P^N CRF	Hierarchical CRF
Time	0.6s	0.8s	16s	8s

those of Pairwise CRF and P^N CRF according the the experimental results.

Simultaneously, we also conduct the comparison of the average inference time. As we can see in Table 3, our MSPN shows the advantage over other CRF due to its tree structure and simple inference strategy. That is, our method differs with hierarchical CRF models in that we do not use pre-computed unsupervised segments as high-level priors, which often make automatic semantic image segmentation task much more complicated and sometimes difficult to obtain. Additionally, the proposed MSPN can jointly model the interactions in pixel or patch wises on each fine-to-coast level, which will reduce the inclusion of misleading segmentations or inaccuracy of other techniques.

Robustness of MSPN. To further test the robustness of the proposed MSPN for diverse realistic environments, we adopt two extra experiments for testing the images from two benchmarks by adding more noises. That is, for each image from the two benchmarks, we add Gaussian noises and illumination oscillations for the entire image by scaling RGB channels. In our implementation, we adopt MATLAB to preprocess images. For Gaussian noises, the default setting with the mean 0 and variance 0.01, where numerical values correspond to normalized images, is adopted. As for illumination oscillation, given an image, we first sample a scale value from [0.9, 1.1], and then multiply it to all the RGB channels of every pixel. Next, we perform N-R-MSPN on the two benchmarks. Note that we do not process the training images and the validation set. Thus the model is still trained on the original training images. The results are shown in Table 4. According to the results, the proposed MSPN is relatively stable to illumination variations. This is true due to the fact that the proposed MSPN integrates spatial correlations and higher orders to further refine incorrect predictions but not only considering low-level visual features.

Effectiveness of MSPN in handling occlusions in scenes. Our MSPN models the spatial layout of scene images by a hierarchical

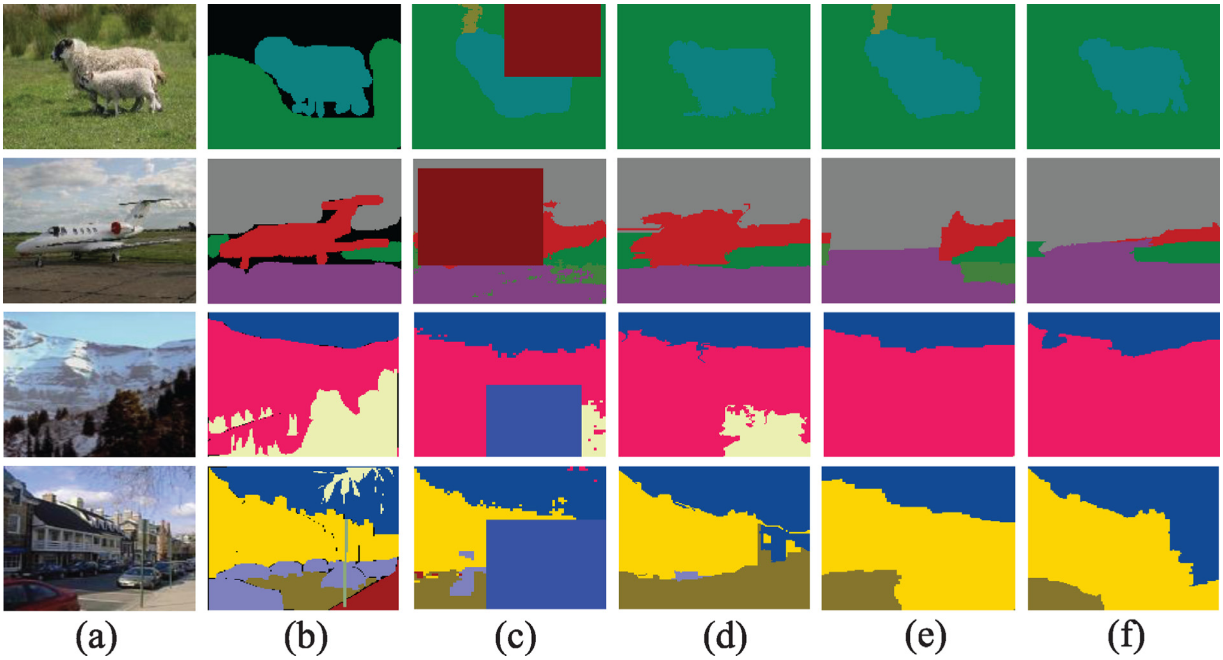


Fig. 5. Scene image understanding result examples from occluded MSRC-21 dataset and SIFT FLOW dataset. Note that MSRC-21 multiscale potentials are occluded by *red* rectangular regions and SIFT FLOW by *blue* rectangular regions, occupying 25% and 45% of the total area. (a) Original images, (b) the groundtruth, (c) occluded unary potentials (only show the raw scale), (d) the results of N-R-MSPN, (e) the results of pairwise CRF, and (f) the results of robust P^N CRF.

Table 4
The average pixel-level accuracy on two benchmarks before and after post-processing.

Original MSRC-21	Noisy version	Original SIFT FLOW	Noisy version
79.4	79.1	72.9	72.2

manner, and is able to capture object shape priors and long-range interactions of multiple objects. We verify the ability of MSPN for modeling such global structure by adding occlusions that often exist in scenes. On the MSRC-21 and SIFT FLOW test sets, we occlude multiscale unary potential by a randomly generated rectangular region on three levels, respectively occupying 5%, 25% and 45% of the total scene area. The occlusion is done by setting the corresponding unary potential to uniform ($\phi_c^s(x_i^{occluded}) = 1/C$). We test our MSPN model to evaluate its power on the unprecise unary potentials. The results and comparisons with two CRF methods are shown in Table 5 and Fig. 5. It can be found that with a small ratio of occlusion (e.g., the sheep in Fig. 5), paiwise CRF and robust P^N CRF can obtain proper label assignments via using the hints from the raw images, and sometimes achieve even a higher global accuracy since some inexact unary potentials can be occluded. However, when the occlusion (or uncertainty of unary potentials) is increased to a relatively large area ratio, both the two CRF-based methods fail in recovering occluded objects. Our MSPN can well handle these cases due to its powerful prior in modeling shape priors and the global spatial layouts of scene objects. For quantitive comparisons, the three methods decrease on both the global accuracy and the average per-category accuracy after increasing the occlusion ratio; however, the proposed MSPN keeps a superior average accuracy, without decreasing to a border line (70% of MSRC-21 and 20% of SIFT FLOW).

Note that there is a large uncertainty of the total memory cost due to the edge prune process. The most memory-consuming stage is the train process, where the space requirement is exponential, and online sequential training MSPN with hundreds of 120 x 160 images needs about 16 GB memories to maintain the complete structure. After edge pruning, the required memory is less than 1 GB since most edges are ineffective. Additionally, according to our experiments, the average computational cost of inference is about 0.6s for each image.

6. Discussion and conclusion

This paper has proposed a multiscale sum-product network (MSPN) to capture the spatial-layout in a coarse-to-fine manner for scene image understanding. We first construct a multiscale representation of unary potentials and object labels. We then use MSPN to model the joint distribution over multiscale unary potentials and object labels. For testing, we infer object labels by using the most probable explanation (MPE) technique. The inference results are then fed into a superpixel based refine method to further improve the scene image understanding results.

Our main contribution is that we extend sum-product network (SPN) to model multi-scale features, long-range spatial correlations, and higher-order priors through a uniform probabilistic graph. We inherit the advantages of SPN, and further propose the improved MSPN for scene parsing scenario. As discussed, SPN can represent most distributions in directed acyclic probabilistic graphical models such as thin junction trees and hierarchical mixtures by a compact way. The compactness means that SPN can model a class of distributions by using exponentially fewer nodes than other networks. For example, SPN can model N-dimensional uniform distribution with N nodes; however, other mixture models have to use 2^N nodes. Our model extends the advantage of SPN by a hierarchical way to model long range spatial correlations, and thus raises its generality to model more distributions than SPN for a lot real situations, for example, the proposed MSPN can handle some occlusion cases. Due to the property of compactness and acyclic graphs, the proposed model can also be trained and inferred efficiently than undirected probabilistic graphs with cycles, such as CRF and its variants which use complex approximations and iterations to achieve convergence. Therefore, the proposed model is more representable and help reduce computational

Table 5
The evaluation of occlusions in understanding scene images on the MSRC-21 dataset and the SIFT FLOW dataset. We occlude the test benchmark scene images with randomly generated rectangular regions respectively occupying 5%, 25% and 45% of the total scene area, and compare the results with pairwise CRF and robust P^N CRF. Both the average per-category accuracy and the global pixel accuracy of the results are provided for illustration.

	MSRC-21			SIFT FLOW		
	N-R-MSPN	Pairwise CRF	P^N CRF	N-R-MSPN	Pairwise CRF	P^N CRF
5%	(77.8, 84.9)	(75.5, 84.5)	(76.9, 84.6)	(20.4, 72.9)	(19.8, 70.8)	(19.9, 72.2)
25%	(75.0, 82.8)	(70.8, 81.2)	(72.6, 82.9)	(20.3, 72.6)	(19.7, 70.3)	(19.9, 71.7)
45%	(72.3, 81.1)	(56.7, 74.2)	(61.2, 77.3)	(20.2, 71.7)	(19.2, 68.8)	(20.0, 71.1)

overloads. This has been validated in our scene parsing explorations in this work. We conduct experiments on two challenging datasets and demonstrate the effectiveness of our method for scene image understanding, in particular for handling occlusions that often occur in natural scenes.

The main weakness of the paper is that the lower representability bounds the further improvement of the performance. We believe the performance can be improved by a large margin through deep features due to the fact that we perform the best in the community that uses shallow features. In our future work, we will explore more representative deep features and unary potentials to further improve the accuracy of scene image understanding. Additionally, exploring scene understanding on image-level by co-understanding large-scale images will be another interesting task in our further research.

Acknowledgment

The work described in this paper was supported by the Natural Science Foundation of China under Grant No. 61272218 and No. 61321491, and the Program for New Century Excellent Talents under NCET-11-0232.

References

Amer, M. R., & Todorovic, S. (2012). Sum-product networks for modeling activities with stochastic structure. In *Cvpr* (pp. 1314–1321).

Arbelaez, P., Hariharan, B., Gu, C., Gupta, S., Bourdev, L. D., & Malik, J. (2012). Semantic segmentation using regions and parts. In *Cvpr* (pp. 3378–3385).

Boix, X., Gonfaus, J. M., van de Weijer, J., Bagdanov, A. D., Gual, J. S., & Gonzàlez, J. (2012). Harmony potentials - fusing global and local scale for semantic image segmentation. *International Journal of Computer Vision, 96*(1), 83–102.

Brox, T., Bourdev, L. D., Maji, S., & Malik, J. (2011). Object segmentation by alignment of poselet activations to image contours. In *Cvpr* (pp. 2225–2232).

Carreira, J., Li, F., & Sminchisescu, C. (2012). Object recognition by sequential figure–ground ranking. *International Journal of Computer Vision, 98*(3), 243–262.

Chen, F., Yu, H., Hu, R., & Zeng, X. (2013). Deep learning shape priors for object segmentation. In *Cvpr* (pp. 1870–1877).

Darwiche, A. (2003). A differential approach to inference in Bayesian networks. *Journal of the ACM, 50*(3), 280–305.

Eslami, S. M. A., Heess, N., & Winn, J. M. (2012). The shape Boltzmann machine: A strong model of object shape. In *Cvpr* (pp. 406–413).

Farabet, C., Couprie, C., Najman, L., & LeCun, Y. (2013). Learning hierarchical features for scene labeling. *IEEE Transaction on Pattern Analysis and Machine Intelligence, 35*(8), 1915–1929.

Felzenszwalb, P. F., & Huttenlocher, D. P. (2004). Efficient graph-based image segmentation. *International Journal of Computer Vision, 59*(2), 167–181.

Gens, R., & Domingos, P. (2012). Discriminative learning of sum-product networks. In *Nips* (pp. 3248–3256).

Gritti, T., Damkat, C., & Monaci, G. (2013). Semantic video scene segmentation and transfer. *Computer Vision and Image Understanding, 122*(1), 172–181.

Gu, C., Lim, J. J., Arbelaez, P., & Malik, J. (2009). Recognition using regions. In *Cvpr* (pp. 1030–1037).

Han, F., & Zhu, S.-C. (2009). Bottom-up/top-down image parsing with attribute grammar. *IEEE Transaction on Pattern Analysis and Machine Intelligence, 31*(1), 59–73.

He, X., Zemel, R. S., & Carreira-Perpiñán, M. Á. (2004). Multiscale conditional random fields for image labeling. In *Cvpr (2)* (pp. 695–702).

Kohli, P., & Kumar, M. P. (2010). Energy minimization for linear envelope MRFS. In *Cvpr* (pp. 1863–1870).

Kohli, P., Ladicky, L., & Torr, P. H. S. (2008). Robust higher order potentials for enforcing label consistency. *Cvpr*.

Kohli, P., Osokin, A., & Jegelka, S. (2013). A principled deep random field model for image segmentation. In *Cvpr* (pp. 1971–1978).

Krähenbühl, P., & Koltun, V. (2011). Efficient inference in fully connected CRFS with gaussian edge potentials. In *Nips* (pp. 109–117).

Ladicky, L., Russell, C., Kohli, P., & Torr, P. H. S. (2009). Associative hierarchical CRFS for object class image segmentation. In *Iccv* (pp. 739–746).

Ladicky, L., Russell, C., Kohli, P., & Torr, P. H. S. (2010). Graph cut based inference with co-occurrence statistics. In *Eccv (5)* (pp. 239–253).

Leibe, B., & Schiele, B. (2003). Interleaved object categorization and segmentation. In *Bmvc* (pp. 759–768).

Lin, D., & Xiao, J. (2013). Characterizing layouts of outdoor scenes using spatial topic processes. In *Iccv* (pp. 841–848).

Liu, C., Yuen, J., & Torralba, A. (2009). Nonparametric scene parsing: Label transfer via dense scene alignment. In *Cvpr* (pp. 1972–1979).

Liu, S., Xu, D., & Feng, S. (2011). Region contextual visual words for scene categorization. *Expert System with Application, 38*(9), 11591–11597.

Luo, P., Wang, X., & Tang, X. (2013). A deep sum-product architecture for robust facial attributes analysis. In *Iccv* (pp. 2864–2871).

Malisiewicz, T., Gupta, A., & Efros, A. A. (2011). Ensemble of exemplar-svms for object detection and beyond. In *Iccv* (pp. 89–96).

Poon, H., & Domingos, P. (2011). Sum-product networks: A new deep architecture. In *Uai* (pp. 337–346).

Ravì, D., Bober, M., Farinella, G. M., Guarnera, M., & Battiato, S. (2016). Semantic segmentation of images exploiting DCT based features and random forest. *Pattern Recognition, 52*, 260–273.

Rincón, M., Bachiller, M., & Mira, J. (2005). Knowledge modeling for the image understanding task as a design task. *Expert System with Application, 29*(1), 207–217.

Shotton, J., Winn, J. M., Rother, C., & Criminisi, A. (2009). 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.

Steinberg, D. M., Pizarro, O., & Williams, S. B. (2015). Hierarchical Bayesian models for unsupervised scene understanding. *Computer Vision and Image Understanding, 131*, 128–144.

Tighe, J., & Lazebnik, S. (2013). Superparsing - scalable nonparametric image parsing with superpixels. *International Journal of Computer Vision, 101*(2), 329–349.

Todorovic, S., & Nechyba, M. C. (2007). Interpretation of complex scenes using dynamic tree-structure Bayesian networks. *Computer Vision and Image Understanding, 106*(1), 71–84.

Tu, Z., Chen, X., Yuille, A. L., & Zhu, S. C. (2005). Image parsing: Unifying segmentation, detection, and recognition. *International Journal of Computer Vision, 63*(2), 113–140.

Vijayanarasimhan, S., & Grauman, K. (2011). Efficient region search for object detection. In *Cvpr* (pp. 1401–1408).

Yao, J., Fidler, S., & Urtasun, R. (2012). Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In *Cvpr* (pp. 702–709).

Yin, H., Jiao, X., Chai, Y., & Fang, B. (2015). Scene classification based on single-layer SAE and SVM. *Expert System with Application, 42*(7), 3368–3380.

Yuan, Z., & Lu, T. (2014). A novel context-aware topic model for category discovery in natural scenes. In *Accv* (pp. 158–171).

Yuan, Z., Lu, T., & Shivakumara, P. (2014). A novel topic-level random walk framework for scene image co-segmentation. In *Eccv* (pp. 695–709).