Journal of
Applied Remote Sensing

RemoteSensing.SPIEDigitalLibrary.org

Semantic segmentation of multisensor remote sensing imagery with deep ConvNets and higher-order conditional random fields

Yansong Liu
Sankaranarayanan Piramanayagam
Sildomar T. Monteiro
Eli Saber

SPIE.

Yansong Liu, Sankaranarayanan Piramanayagam, Sildomar T. Monteiro, Eli Saber, "Semantic segmentation of multisensor remote sensing imagery with deep ConvNets and higher-order conditional random fields," *J. Appl. Remote Sens.* **13**(1), 016501 (2019), doi: 10.1117/1.JRS.13.016501.

Semantic segmentation of multisensor remote sensing imagery with deep ConvNets and higher-order conditional random fields

Yansong Liu,[a,*] Sankaranarayanan Piramanayagam,[a] Sildomar T. Monteiro,[a,b] and Eli Saber[a,b]

[a]Rochester Institute of Technology, Chester F. Carlson Center for Imaging Science, Rochester, New York, United States

[b]Rochester Institute of Technology, Department of Electrical and Microelectronic Engineering, Rochester, New York, United States

Abstract. Aerial images acquired by multiple sensors provide comprehensive and diverse information of materials and objects within a surveyed area. The current use of pretrained deep convolutional neural networks (DCNNs) is usually constrained to three-band images (i.e., RGB) obtained from a single optical sensor. Additional spectral bands from a multiple sensor setup introduce challenges for the use of DCNN. We fuse the RGB feature information obtained from a deep learning framework with light detection and ranging (LiDAR) features to obtain semantic labeling. Specifically, we propose a decision-level multisensor fusion technique for semantic labeling of the very-high-resolution optical imagery and LiDAR data. Our approach first obtains initial probabilistic predictions from two different sources: one from a pretrained neural network fine-tuned on a three-band optical image, and another from a probabilistic classifier trained on LiDAR data. These two predictions are then combined as the unary potential using a higher-order conditional random field (CRF) framework, which resolves fusion ambiguities by exploiting the spatial–contextual information. We utilize graph cut to efficiently infer the final semantic labeling for our proposed higher-order CRF framework. Experiments performed on three benchmarking multisensor datasets demonstrate the performance advantages of our proposed method. [DOI: 10.1117/1.JRS.13.016501]

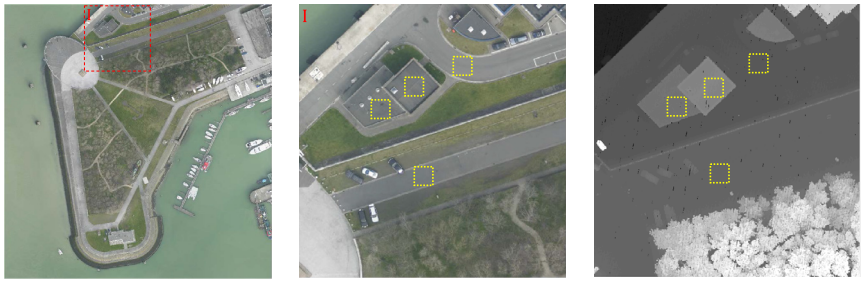
Keywords: semantic segmentation; multisensor remote sensing; light detection and ranging; deep convolutional neural networks; conditional random fields.

Paper 180777 received Sep. 24, 2018; accepted for publication Dec. 10, 2018; published online Jan. 11, 2019.

1 Introduction

Classification of aerial imagery has been one of the central tasks in many remote sensing applications, e.g., environmental assessment and monitoring, city planning, and land cover change detection. Typical methods for classifying these types of images follow a bottom-up approach in which information is processed sequentially from pixel-to-object level. Pixel-wise image processing techniques are suitable for aerial images with a lower spatial resolution, where the region of interest includes natural terrains of a large area of, for example, forests, water bodies, grass, and tree canopies. With the advance in sensor technology, very-high-resolution (VHR) aerial images are now readily available and enable the extraction of fine details with a ground spatial resolution of about 10 cm. Using VHR aerial imagery, state-of-the-art deep learning and structured prediction methods can be utilized to generate dense semantic segmentation. For instance, methods based on Markov random fields (MRFs) for classification[1–4] have been proposed to perform land cover mapping utilizing the enhanced spatial contextual information in VHR images.

*Address all correspondence to Yansong Liu, E-mail: yxl3624@rit.edu



(a) (b) (c)

Fig. 1 (a) RGB orthophoto from the Zeebruges dataset; (b) Zoom-in of an area with rooftops and impervious surface, image patches appear very similar in both color and texture; (c) LiDAR nDSM map (derived from ENVI using point cloud) of the same area. Intensity values indicate the relative elevation of each pixel. The yellow patches are indistinguishable in the color image but have distinct heights.

The remarkable success of deep learning and, in particular, convolutional neural networks (CNNs) on achieving state-of-the-art results in computer vision tasks has motivated researchers to apply these methods to other fields, such as remote sensing. However, training deep neural networks typically require extensive labeled datasets. Unfortunately, obtaining ground truth for aerial images that span large areas can be expensive and unfeasible. Although there has been an increase in the number of publicly available labeled datasets in remote sensing, researchers have overcome the issue of limited ground truth using a large labeled dataset that has similar characteristics to pretrain a neural network, most commonly object detection trained on the ImageNet dataset. After that, one or more convolutional layers of the pretrained network are fine-tuned on aerial data.[5–9] Despite the difference in viewing perspective and object scale between ImageNet and aerial images, this approach called transfer learning has shown to achieve higher classification performance[10] compared with competing methods, such as support vector machines and random forests. Transfer learning has also shown excellent results in other applications such as vehicle detection[11] and scene classification[12] from remote sensing data.

In aerial remote sensing data, the scene image is captured from above, at either an oblique angle or directly from nadir. Because of this limited field of view, it can be challenging to discriminate objects and materials at the ground level just by looking at their appearance. For example, some flat rooftops can have similar color and spatial shape as those of impervious surface (shown in Fig. 1). To address this challenge, we make use of a different sensing modality that can measure complementary information that can be used to discriminate ground objects with similar color characteristics. Light detection and ranging (LiDAR) systems can provide relevant height information. Combining aerial optical images and georeferenced LiDAR data can provide a better representation of a given scene. However, this multisensor imagery data pose a challenge to the use of pretrained deep networks for image classification and object detection, which are typically trained using only RGB bands. Developing deep networks for combining optical images (e.g., RGB and IR) and LiDAR data has become a hot research topic in remote sensing. An approach to address this challenge is to train two separate neural networks, one to process the optical images and the other to process the corresponding LiDAR data. The raw three-dimensional (3-D) point cloud from LiDAR data needs to be preprocessed to be represented as a three-band image, typically using a digital surface model, height variation, and surface norm. The learned features from the two neural networks can then be concatenated, e.g., using a convolutional layer. This approach performs feature-level fusion.[5,7] However, training two separate neural networks can be computationally expensive and may present poor performance in scenarios where either color or geometry has low local variability. Also, the robustness of training the false three-band LiDAR images is still under investigation.

We propose a fusion approach for combining and exploiting high-resolution optical imagery and corresponding LiDAR data. Our proposed method requires less training time and resources (see Sec. 4.2 for details) to achieve competitive classification results in comparison with state-of-the-art neural networks. The overall flowchart of our proposed method is shown in Fig. 2.

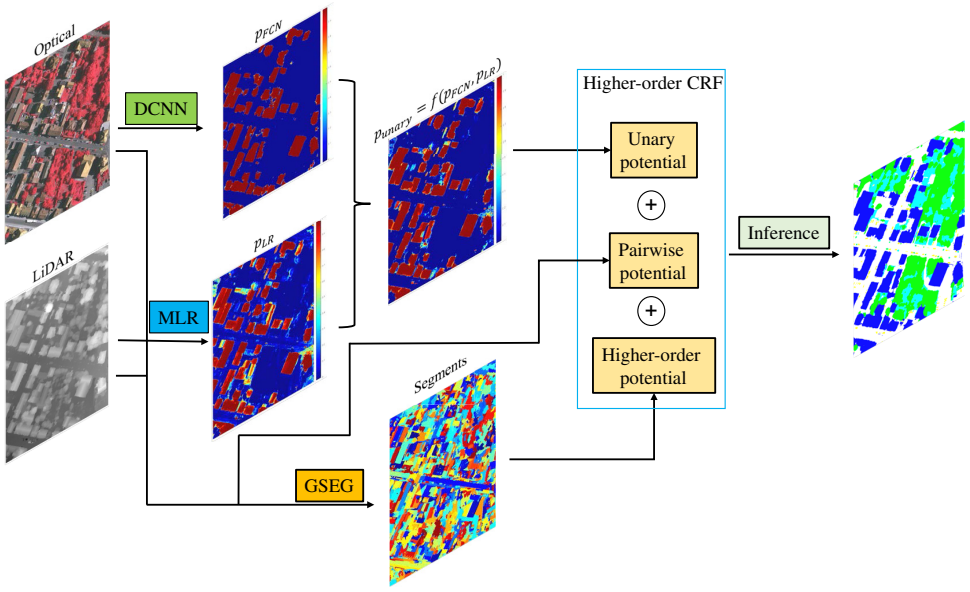


Fig. 2 Proposed decision-level multisensor semantic segmentation method.

We generate two separate initial probabilistic predictions from the optical imagery, using fully convolutional networks (FCN), and the LiDAR data, using multinomial logistic regression (MLR). We then combine the initial predictions as unary potential functions modeled as a conditional random field (CRF). Instead of using only pixel-level estimates, we propose to use segments obtained using a gradient-based segmentation algorithm (GSEG)[13] in a higher-order CRF model. We apply the robust P^N Potts' model as higher-order potential functions, which can enforce label consistency within segments. Inference in higher-order CRFs to predict final labels can be done efficiently using graph cuts. The main advantages of higher-order CRFs in our proposed method are (1) to resolve decision ambiguity between the two initial estimates by enforcing label consistency at the local and global levels simultaneously and (2) to preserve local object boundaries based on a robust segmentation algorithm.

This paper is an extended version of our recently published conference paper,[14] the original contributions of this paper are: (1) a higher-order CRF model for efficient decision-level multisensor semantic segmentation of aerial images and (2) a thorough study of advantages and limitations of the robust higher-order Potts' model for remote sensing image classification. In addition, in this paper, we tested our method on two more datasets and did a more detailed analysis based on the experimental results of these two datasets. We also added experimental results of applying different segmentations and classifiers, which shows our proposed framework can be easily integrated with various current state-of-the-art classifiers and segmentation methods. The results are promising compared with other state-of-the-art methods.

The rest of this paper is organized as follows: in Sec. 2, a brief review of recent developments of DCNNs and CRFs on remote sensing images is presented. The proposed method is described in Sec. 3. Benchmarking results on three publicly available multisensor datasets are provided in Sec. 4. In Sec. 5, we thoroughly discuss the empirical performance and practical issues of our proposed CRF model. Finally, conclusions are presented in Sec. 6.

2 Related Works

Over the last decade, significant research has been done on semantic segmentation of aerial imagery. Some of the important methods are described in the review papers by Gómez-Chova et al.[15] and Debes et al.[16] In the following section, we will discuss some recent developments on the multisensor classification of optical imagery and LiDAR data.

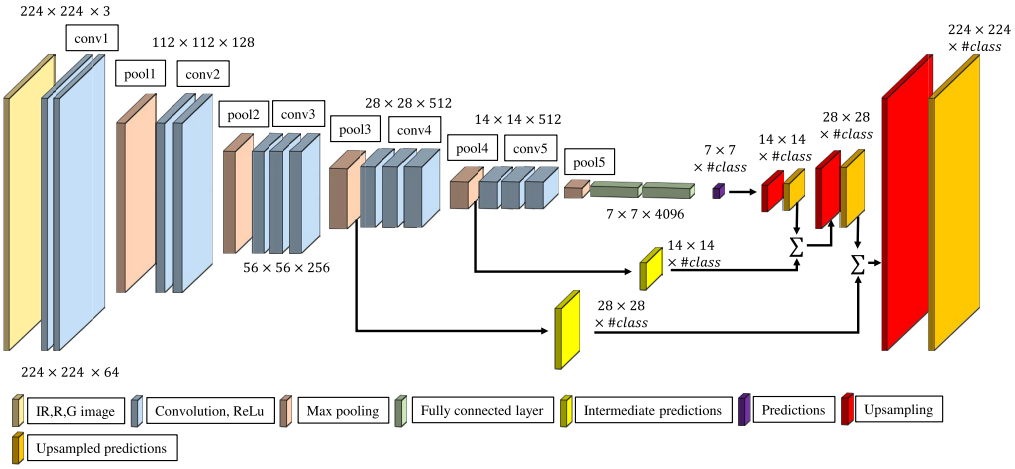


Fig. 3 Proposed FCN-8s architecture with two skip layers. This network architecture learns to combine coarse, deeper layer information with fine, shallow layer information. It combines predictions from the final layer and the pool4 layer to generate a finer prediction. Including the output from pool3 can provide an even more detailed prediction.

2.1 *Deep Convolutional Neural Networks*

Recent research has shown that neural networks trained on general image categorization tasks can be used as feature extractor for aerial imagery,[8,17] even though the viewing perspectives of overhead imagery and general images are quite different. This has allowed scene classification,[12] vehicle detection,[11] tree species mapping,[18] and road detection[19] in aerial images. The algorithms are applied on patches or tiles of the large aerial images as opposed to the entire image.

In deep convolutional neural networks (DCNNs), downsampling operation, e.g., max pooling, is performed after convolutional layers to capture the longer range contextual information and extract more abstract features. The downsampling process usually leads to coarse labeling or prediction at a lower spatial resolution. Fully convolutional neural networks (FCNNs) have been proposed to improve the coarse classification results and achieve dense end-to-end prediction. In Ref. 20, the authors proposed a skip architecture that combines semantic information from a deep, coarse layer with appearance information from a shallow, fine layer to produce accurate and dense semantic segmentation. Figure 3 shows the detailed architecture of FCN-8s skip network. In Ref. 21, authors proposed a semantic pixel-wise segmentation method called SegNet that utilizes deconvolutional decoder layers to map low-resolution encoder features to full input resolution features. The authors in Ref. 22 utilized an atrous convolution method to expand the support of the filter and removed most of the down-sampling operations to obtain dense labeling. These approaches have been successfully adapted to the dense semantic labeling of remote sensing images[5,6,23] and outperform the traditional pixel-level classification methods (such as SVMs[24]) that employ hand-crafted feature descriptors. In our paper, we adopt the FCN in Long et al.[20] paper as our pretrained network and fine-tuned it using the VHR optical imagery with a two-stage training scheme. We utilized the per pixel probability output from the network as one of the local potential functions in our higher-order CRF framework.

2.2 *Conditional Random Fields*

In CRF models, each pixel is assigned a label by examining its color or spectral characteristics and also spatial relations. Pixels of an object can have complex local or global dependencies. For image classification, in general, two important assumptions are made with regard to spatial smoothness and contextual coherence: (1) neighboring pixels tend to belong to the same class except on the object boundaries. (2) adjacent pixels/objects from natural images must follow a certain practical meaning, e.g., the car is more likely found on the ground than on a tree. CRFs can be used to model these dependencies at a local and global scale. These energy-based random field methods have been widely used for exploiting contextual information

for both computer vision and remote sensing images.[1–3,25–28] The full potential of the random field methods has not been realized due to the limitation of inference methods for higher-order node connections, particularly for the large-scale data. Recently, research has been done to employ higher-order random fields with efficient inference methods.[29–33] The robust higher-order potential and graph cut inference method[29,30] has stood out due to its efficiency on the relatively large-scale dataset and the state-of-the-art semantic segmentation performance. It has been applied to the remote sensing applications such as road and rooftop extraction.[25,26]

2.3 *Multisensor Classification*

There are two commonly used fusion techniques for the classification of multisensor images. In one methodology, features are initially extracted from all imaging modalities and then fused together by concatenation or selection. The combined features are then utilized in a supervised training scheme to obtain a label map.[34] In the second fusion method, classification results are initially obtained for each imaging modality, and then these predictions are merged to achieve an optimal output (referred to as decision-level fusion[35–37]). Sherrah[5] trained two separate neural networks for each modality and concatenated the learned features at the last convolutional layer. In Ref. 7, multiple predictions from individual modalities are averaged. They also introduce a network that combines features at the final layers by a residual correction framework. The later network achieved slightly better quantitative results in comparison with the averaging of outcomes. Recent research in Audebert et al.[38] provides a solid investigation of applying early and late fusion in deep neural networks for LiDAR and multisepctural data.

Authors in Ref. 39 represent all the bands of multisensor data as a single image and utilize it to train a multiresolution CNN. They also extract hand-crafted features from the image to train a second classifier. They combine the individual results at a decision-level. Our proposed method differs from the previous algorithm in three different aspects. First, we apply fully convolutional neural networks only on the three bands (optical imagery) of multisensor data, whereas in their work, the input is a combination of all bands (requires training of neural network from random initialization). As our network uses only three bands, pretrained network weights could be fine-tuned with existing limited ground truth. Next, we learn the weight parameters to combine individual probabilistic outputs in a CRF framework using training data. Their approach combines results from two classifiers directly in a rigid way. Lastly, our method uses a higher-order CRF model, whereas their model uses only standard unary and pairwise potentials.

3 Methodology

Our proposed method consists of four major steps: (1) obtain two probabilistic predictions for optical images and LiDAR data, respectively, using DCNNs and MLR, (2) formulate the energy function of the higher-order CRF with the previous two predictions as unary potential, and robust P^N Potts model as the higher-order potential, (3) generate segments for the given imagery, and (4) train all the parameters in CRF-based algorithm and then obtain the image classification results using graph cut inference method. The details of each step are discussed below.

3.1 *Deep Fully-Convolutional Neural Network*

VHR optical imagery provides rich low-level and high-level features. To fully take advantage of such information, we used features from neural networks to generate high-resolution per class probability prediction. Pixel-wise semantic segmentation of an image through neural networks was initially proposed by Long et al.[20] by extending a popular image classification neural network named VGG-16. The VGG-16 network consists of multiple convolutional layers followed by two fully connected and a final scoring layer. The last layer acts as a classifier to give scores to different predefined objects. The authors in Ref. 20, in their work, modified the fully connected layers to add spatial support and thereby generate the labels for each pixel. Specifically, the output of the fully connected layer has dimensions of $n \times n \times 4096$ instead of 1×4096, where n indicates the size of the down sampled spatial support of the fully connected layer.

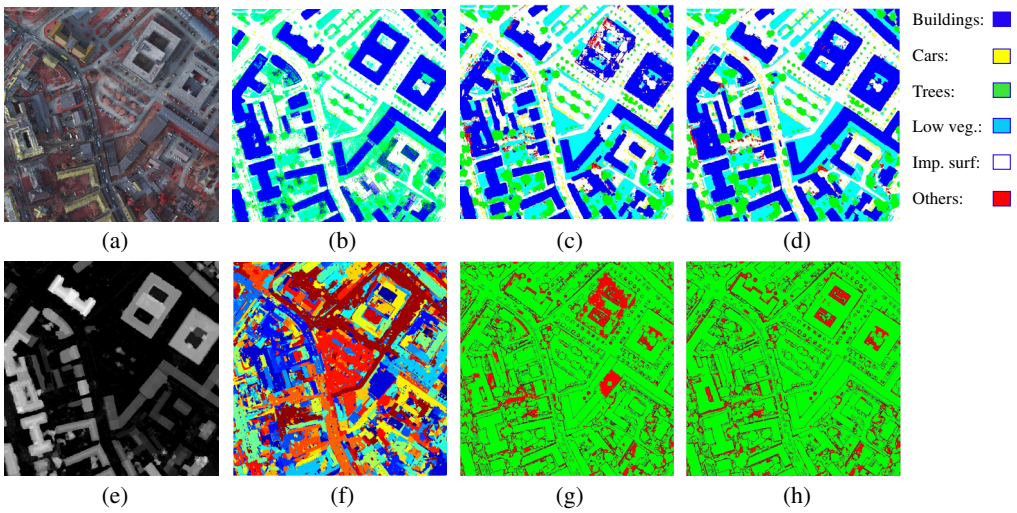


Fig. 4 Classification results of ISPRS Potsdam dataset 4_15 and their corresponding error maps. (a) Color-infrared images, (b) classification results of MLR using LiDAR features, (c) classification results of FCN-8s using IR, R, and G channels, (d) classification results of our proposed method, (e) normalized DSM map, (f) segmentation map obtained from the segmentation algorithm. (g) and (h) Corresponding error maps for (c) and (d), respectively, where red pixels indicate misclassifications.

They called the resultant network the fully convolutional neural network (FCN). FCN-8s contains five layers with multiple convolutions and rectified linear activation functions. Each of these layers is succeeded by a max pooling (downsampling) operation. The architecture is shown in Fig. 3. The sixth and seventh layers are two fully connected convolutional layers. The eighth convolution (score) layer generates outputs corresponding to the number of classes in the ground truth. Upsampled outputs from the eighth layer are combined with the outputs from pool 4 layer. The result is again upsampled and merged with pool 3 layer outputs. Fusing the output of pool 3 and pool 4 layers (skip connection) assists in obtaining finer semantic labels. The fine-tuned network was then used to generate per pixel probability maps, which are denoted as P_{FCN}.

One of the FCN-8s classification results and its corresponding error map are shown in Figs. 4(c) and 4(g). We notice that part of the rooftops is misclassified as impervious surface. This demonstrates that objects and land covers sometimes can have very similar spectral and spatial characterizations in aerial imagery and thus FCN-8s is not able to correctly distinguish them. Fortunately, LiDAR data provide us the complementary information, i.e., elevation, to improve the performance of classification.

3.2 *Multinomial Logistic Regression*

The LiDAR data are provided as normalized digital surface maps (nDSMs), which contain limited contextual information in comparison with the VHR optical imagery. Previous works demonstrated that training an additional neural network on the LiDAR data can increase the overall performance of the classification task. We, however, hypothesize that utilizing an efficient linear classifier shall be sufficient to take advantage of the LiDAR information. In our work, we employ an MLR on hand-crafted features derived from both LiDAR data and optical imagery. MLR has been used in numerous remote sensing classification tasks,[40] and it can produce multiclass probabilistic estimates. More importantly, MLR is fully probabilistic, therefore it provides calibrated probabilities off-the-shelf, whereas SVM and random-forest require postprocessing to compute multiclass (e.g., one versus all) and probabilities (Platt method[41] using cross validation). The probability that a pixel x_i belongs to a particular category c can be calculated as follows:

$$P(x_i = m|Z_i) = \frac{e^{\left(a_m + \sum_{k=1}^{K} \beta_{mk} Z_{i,k}\right)}}{1 + \sum_{m=1}^{M-1} e^{\left(a_m + \sum_{k=1}^{K} \beta_{mk} Z_{i,k}\right)}}, \tag{1}$$

where M is the number of categories, K is the number of features, β_{ck} is a vector of weights, and $Z_{i,k}$ are the features extracted from the LiDAR data for MLR. The hand-crafted features include height, height variations, surface norm, and the normalized difference vegetation index (NDVI). NDVI shows higher values in the presence of vegetation, and it has been largely utilized for vegetation detection in several remote sensing applications.[42–44] It is defined as follows:

$$\text{NDVI} = \frac{\text{NIR} - \text{VISR}}{\text{NIR} + \text{VISR}}, \tag{2}$$

where NIR and VISR stand for spectral reflectance measurements in the near-infrared and visible red regions, respectively. For training MLR parameters a_c and β_{ck}, we randomly chose 10,000 points per class, and the trained MLR model is later used to predict pixel-wise probability map for the test images. The probabilistic output is denoted as P_{MLR}. One of the MLR classification results is shown in Fig. 4(b). The output is fairly noisy, but it recognizes the building structure accurately due to the additional height information. However, we can also notice that MLR does not perform well in detecting car pixels. One of the reasons for poor performance is the vertical resolution of nDSM that is not fine enough to capture the shape of a car.

3.3 *Higher-Order Conditional Random Field Formulation*

The two probabilistic predictions generated using the optical and LiDAR data, respectively, provide complementary information to the optimal class labeling. FCN-8s trained on the visible bands exploits more spectral and spatial information and, therefore, it learns object structure much better than MLR does. For land covers/objects that lack textures or have indistinguishable spectral characterization, MLR takes advantage of height information and generates very reliable and efficient predictions. We combine these two outputs in the higher-order CRF framework such that the final classification results can have spatial/contextual coherence and in addition preserve object boundaries. The CRF is an undirected graphical model defined as follows:

$$P(X = x|O) = \frac{1}{Z} \prod_{c \in C} \Psi_c(x_c), \tag{3}$$

where $P(X = x|O)$ is the conditional probability distribution of the label $X = x$ given observations O, Ψ_c can be modeled as a Gibbs distribution:

$$\Psi_c(x_c) = e^{-\psi_c(x_c)}, \tag{4}$$

and $\psi_c(x_c)$ is the potential function. C is a set of cliques, over which the potential functions are used to encode the conditional dependence assumptions. $Z = \sum_X \prod_{c \in C} e^{-\psi_c(x_c)}$ is the partition function that normalizes the probabilistic distribution. Therefore, the final labeling process in a CRF framework becomes a maximum *a posteriori* (MAP) estimation, where we find the set of label x that maximizes the conditional probability distribution: $\arg\max_x P(X = x|O)$. As the partition function Z is usually intractable, we define the negative log of the conditional distribution as the energy:

$$E(x) = -\log P(X = x|O) - \log Z = \sum_{c \in C} \psi_c(x_c). \tag{5}$$

Here, $\psi_c(x_c)$ is a set of potential functions that can encode a priori knowledge about the interdependence of the random variables within different cliques. The MAP estimation thereby is converted into an energy minimization problem: $\arg\min_x E(x)$. We form our proposed higher-order CRF framework with three different scales.

3.3.1 *Unary potential*

The very first scale would be the prior belief of the individual random variable itself:

$$\psi_u(x) = \sum_{i \in v} \psi_i(x_i), \tag{6}$$

where $x = [x_1, x_2, x_3, \ldots, x_n]$ represents one realization of the label assignments for n pixels: $v = \{1,2,\ldots,n\}$. x_i takes the label from M object classes: $x_i \in L^M$. This is also known as the unary potential, which is usually defined as the negative log-likelihood of the per class probability as shown in the following equation:

$$\psi_i(x_i) = -\log\, P_u(x_i). \tag{7}$$

We have obtained two pixel-wise class probability predictions from the FCN-8s and MLR, i.e., P_{FCN} and P_{MLR}. We here define $P_u(x_i)$ as

$$P_u(x_i = m) = \frac{e^{-f_m(P_{\mathrm{FCN}}, P_{\mathrm{MLR}}; \sigma_m)}}{\sum_{m=1}^{M} e^{-f_m(P_{\mathrm{FCN}}, P_{\mathrm{MLR}}; \sigma_m)}}, \tag{8}$$

where $f_m(P_{\mathrm{FCN}}, P_{\mathrm{MLR}}; \sigma_m)$ is a linear combination of P_{FCN} and P_{MLR}, and it takes the form

$$f_m(P_{\mathrm{FCN}}, P_{\mathrm{MLR}}; \sigma_m) = \sigma_{m0} + \sigma_{m1} P_{\mathrm{FCN}} + \sigma_{m2} P_{\mathrm{MLR}}. \tag{9}$$

The parameters σ_m are trained separately on a hold-out training set.

3.3.2 *Pairwise potential*

The second scale of the CRF encodes the local smoothness assumptions by introducing a penalty for neighboring pixels that take heterogeneous labels. It is also known as the pairwise potential:

$$\psi_p(x) = \sum_{i \in v, j \in N(i)} \psi_{ij}(x_i, x_j), \tag{10}$$

where $N(i)$ is the four-way connected neighborhood of pixel i. This connection encodes the shortest range of local context. The most common form for the pairwise potential is the Potts model, which takes the form:

$$\psi_{ij}(x_i, x_j) = \mu \cdot \Delta(x_i \neq x_j), \tag{11}$$

where $\Delta(\cdot)$ is an indicator function, and μ is the cost that penalizes the heterogeneous labels. However, using Potts model has two issues: (1) it tends to over-smooth the labeling results; (2) it applies the same amount of penalties for different combinations of labels (but, in practice, it should impose less penalty for the very likely combinations of labels and a large penalty for the most unlikely). To overcome these two issues, we utilize the color contrast sensitivity cost to take into account the gradient of images so that we can penalize less on the potential object boundaries. The reformed pairwise potential is expressed as

$$\psi_{ij}(x_i, x_j) = [\theta_\alpha + \theta_\beta \exp(-\theta_\gamma \|I_i - I_j\|^2)] \cdot T(x_i \neq x_j), \tag{12}$$

where I_i and I_j are the features or observations of pixel x_i and x_j. $\|I_i - I_j\|$ is the Euclidean distance in the feature space for the pair of pixels. For multisensor imagery, I_i can include all physical measurements, e.g., color and height. $T(x_i \neq x_j)$ is a $M \times M$ symmetric matrix that has the diagonal values of zeros and others are the costs for different combinations of class labels. This model is intended to enforce the label consistency without overly smoothing the object boundaries. Also, it takes into account the local contextual information to avoid unlikely combinations of labels. However, it is still incapable of extracting the fine-contours of certain objects.

3.3.3 *Higher-order potential*

The higher-order potential extends the smoothness assumption from the neighboring pixels to the local regions. The idea is to first group pixels in the images into coherent regions so that each region ideally belongs to one object. Using an edge awareness segmentation algorithm, this higher-order potential is particularly useful in preserving the object boundaries. Given a set of segments denoted as C, we can define our higher-order potential as

$$\psi_h(x) = \sum_{c \in C} \psi_c(x_c). \tag{13}$$

Segments C are usually generated by an unsupervised segmentation algorithm.[13,45] We choose to take the form of the robust P^N Potts' potential proposed in Ref. 29 for $\psi_c(x_c)$, which has been proved to be particularly useful. It takes the form of

$$\psi_c(x_c) = \begin{cases} N_i(x_c)\frac{1}{Q}\gamma_c^{\max}, & \text{if } N_i(x_c) < Q \\ \gamma_c^{\max}, & \text{otherwise} \end{cases}, \tag{14}$$

where $N_i(x_c)$ denotes the number of pixels taking different labels from the dominant label of the segment. $\gamma_c^{\max}$ is the maximum cost that will be added. $\gamma_c^{\max}$ can be defined in a way that takes into account the quality of the segmentation and the contextual information. We define $\gamma_c^{\max}$ in our case as the following:

$$\gamma_c^{\max} = \theta_c |c| \cdot e^{\left[-\theta_h \frac{\sum_{i \in c}(I_i - \mu_i)^2}{|c|}\right]}, \tag{15}$$

where $|c|$ counts the number of pixels in the segment c. I_i is the features of each pixel in the segment. μ is the mean observations over the current segment. The exponential term in Eq. (15) calculates the variance of observations within one segment, which indicates the quality of the segment. Therefore, the higher-order potential will impose the cost accordingly. For instance, if the variance is small, the segment very likely captures a coherent region. Therefore, taking different labels in those coherent regions can cost more penalties. On the contrary, when the variance is large, the cost of taking different labels in one segment is small. That is because large variance usually means that the region might contain multiple objects so that forcing all the pixels in this segment to take the same label can cause misclassification. Unlike the P^N Potts model, which imposes a strict cost to any heterogeneous labeling within one segment, the robust P^N Potts model proposes a linear truncated cost that is dependent on the number of pixels taking different labels from the dominant label of the segment. The heterogeneity of the labeling is controlled by the parameters θ_c and Q, respectively.

Now let us combine the Eqs. (5), (6), (10), and (13) to formulate our higher-order CRF energy function as

$$E(x) = \sum_{i \in v} \psi_i(x_i) + \sum_{i \in v, j \in N(i)} \psi_{ij}(x_i, x_j) + \sum_{c \in C} \psi_c(x_c), \tag{16}$$

where $\psi_i(x_i)$, $\psi_{ij}(x_i, x_j)$, and $\psi_c(x_c)$ take the form in Eqs. (7), (12), and (14), respectively. Figure 5 shows the configuration of our proposed higher-order CRF. To be noticed, for the global node of a segment, it can be one of the predefined labels in L^M, or if there is no dominant label for the segment, the segment can take a free label L^F, which means that CRF imposes no cost on heterogeneous labels in this segment. In such a case, it is equivalent to only applying the pairwise CRF.

3.4 *Learning Conditional Random Field Parameters*

The parameters of the proposed higher-order CRF are learned in a stepwise training procedure. We first learn to combine the two probabilities obtained from the FCN-8s: P_{FCN} and MLR: P_{MLR} to form the unary potential $\psi_i(x_i)$. We construct the unary potential using a softmax classifier to recast the two probabilities into a single probability. The parameters σ_m can be learned using a

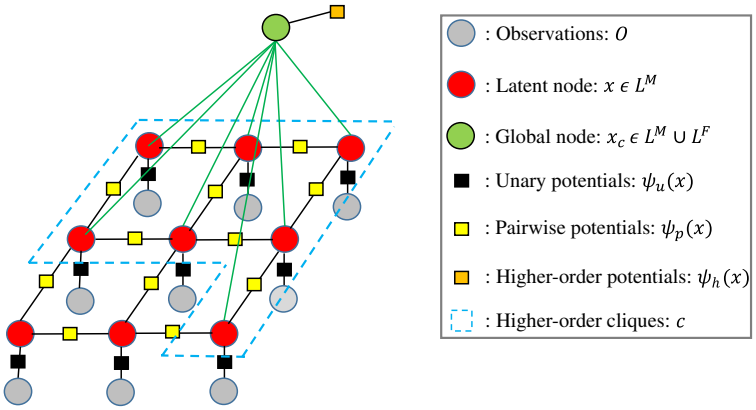


Fig. 5 Probabilistic graphical model for the proposed higher-order CRF.

cross-validation set with a standard maximum likelihood estimation (MLE) procedure. We then keep σ_m constant and proceed to train the pairwise CRF parameters: θ_α, θ_β, θ_γ, and the matrix $T(x_i, x_j)$ without the higher-order term using an approximate marginal inference method proposed in Ref. 46. The marginal inference procedure takes into account model mis-specification and inference approximation. Finally, the higher-order parameters such as θ_c and Q are learned by performing cross validation within an empirical range. Moreover, the value of θ_h is picked empirically, and we use $\theta_h = 2$ for all of our experiments. The tuning of parameters θ_c and Q will be discussed in Sec. 4.

3.5 *Inference Using Graph Cuts*

The inference problem in a higher-order CRF is to find the set of class labels that minimizes the proposed energy function in Eq. (16), i.e., arg $\min_x E(x)$. In general, this energy optimization problem is an NP-hard problem. However, there are certain classes of tractable functions that can be solved in polynomial time, e.g., submodular functions. The energy that we propose happens to belong to those functions. There are two main methods to approximate such an energy minimization problem: message passing algorithms such as belief propagation, and "move making" algorithms, such as graph cut. In this paper, we choose to use the "move making" graph cut algorithm due to its computational efficiency.

The "move making" graph cuts inference algorithm (e.g., α-expansion and $\alpha\beta$-swap) has been successfully used to infer the higher-order CRFs.[29,30] We will review the α-expansion graph cut and then show how to deal with the higher-order CRFs with α-expansion graph cut by adding auxiliary nodes. We refer readers to papers[29–31] for a more detailed description.

The "move making" graph cut algorithm was proposed to efficiently solve the multiclass classification using s–t min-cut-based graph cut algorithm, which was first introduced for binary classification problems.[47] As shown in Fig. 6(a), the "move making" algorithm usually starts from an initial set of labels and then iteratively updates the labels to find the solution that has the lowest energy. To utilize s–t min-cut algorithm, each update in the "move making" algorithm has to be a binary decision. For instance, α-expansion allows any random variable to either maintain its current label or take the proposed label α. We form a transformation function that converts the energy from the label space into the "move" space. We then deduce its corresponding move energy. The transformation function $T_\alpha(\cdot)$ for the α-expansion as

$$T_\alpha(x_i, t_{x_i}) = \begin{cases} \alpha, & t_{x_i} = 0 \\ x_i, & t_{x_i} = 1 \end{cases}. \tag{17}$$

The energy of a move t is the amount of energy induced by the labeling change during the move, i.e., $E(t) = E[T_\alpha(x, t)]$. Therefore, the task of optimizing the CRF energy $E(x)$ is transformed into the problem of optimizing the move energy, i.e., arg $\min_t E[T_\alpha(x, t)]$, where $E(t)$ is a pseudoboolean function. This optimization can be achieved in polynomial time by solving

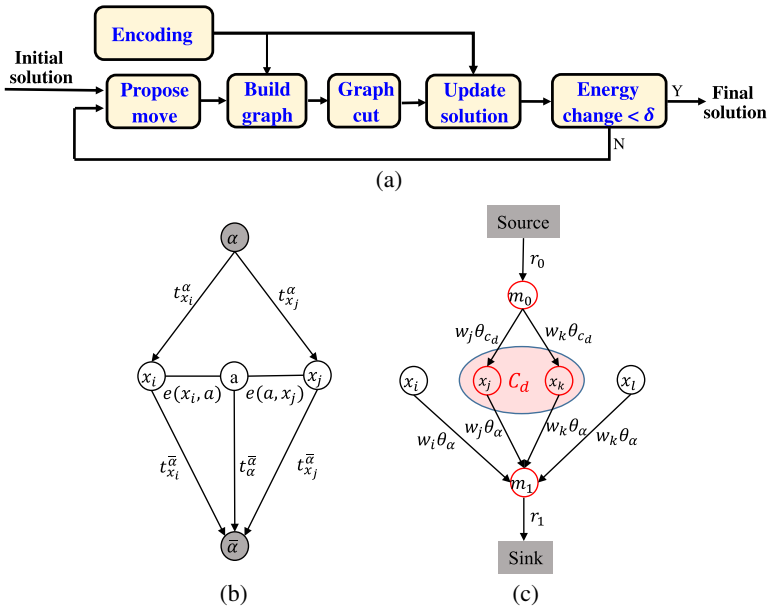


Fig. 6 (a) Flowchart of the move-making graph cut algorithm, (b) S–t min-cut for α-expansion, and (c) α-expansion for higher-order CRF by adding auxiliary variables.

an s–t min-cut as long as $E(t)$ is submodular.[47] See Fig. 6(b) for the illustration of s–t min-cut for α-expansion. For the higher-order CRF, we can first rewrite Eq. (14) into the form of

$$\psi_c(x_c) = \min\left(\min_{k\in L^M}\left\{[|c| - n_k(x_c)]\frac{r_c^{\max}}{Q}, r_c^{\max}\right\}\right), \tag{18}$$

where $|c|$ is the number of pixels in the clique c, k is the potential dominant label for the clique c, and $n_k(x_c)$ is the number of pixels that take the dominant label k in the clique. In other words, $[|c| - n_k(x_c)]$ is equivalent to $N_i(x_c)$ in Eq. (14), which denotes the number of pixels that do not take the dominant label in the clique c. We can further generalize Eq. (18) into the following form:

$$\psi_c(x_c) = \min\left\{\min_{k\in L^M}[(P - f_k(x_c)]\frac{r_c^{\max}}{Q}, \gamma_c^{\max}\right\}, \tag{19}$$

where P and $f_k(x_c)$ are defined as

$$P = \sum_{i\in c} w_i^k, \quad \forall\, k \in L^M, \tag{20}$$

$$f_k(x_c) = \sum_{i\in c} w_i^k \Delta(x_i = k). \tag{21}$$

There is an additional label, which is introduced to form an extended label set: $L^E = L^M \cup \{L^F\}$, where L_F represents a free label. A segment takes the free label when there is no dominant variable found in it. Its corresponding move energy can be written as

$$\psi_c(t_c) = \min\left\{\theta_\alpha \sum_{i\in c} w_i t_i, \gamma_c^{\max}\right\}. \tag{22}$$

The move energy functions above can be minimized using the s–t min-cut algorithm shown in Fig. 6(c).

(a) (b) (c) (d)

Fig. 7 (a) RGB image patch from Potsdam dataset. (b), (c), and (d) Segmentation results of GSEG algorithm with initial seed size of 15, 100, and 150, respectively.

3.6 *Segmentation for Higher-Order Potential*

In the higher-order CRF framework, the quality of the segments impacts the success of the final labeling. It is critical to choose a robust segmentation algorithm that can yield a dense semantic labeling with fine boundaries. GSEG algorithm is one such unsupervised multichannel image segmentation algorithm that utilizes the gradient histogram acquired from the color images to iteratively cluster pixels from lower gradient to higher gradient. The GSEG algorithm is primarily based on color-edge detection, dynamic region growth, and a unique multiresolution region merging procedure. The detailed description of the algorithm can be found in Ref. 13.

As GSEG uses the gradient histogram, which helps to preserve the object boundaries. See one of the illustrations of GSEG segmentation results with different initial seed sizes in Fig. 7. GSEG is particularly suited for aerial image segmentation because the size of the objects in the VHR aerial images can vary from a several hundred pixels to tens of thousands. GSEG does not pose strict constraints of the segment size. Instead, GSEG keeps growing coherent regions until no local low-gradient pixels can be found. It is, therefore, able to generate segments for objects different scales. The performance of GSEG is mainly impacted by two parameters, one is the initial seed size τ, and the other is similarity ratio γ. The former is to determine the initial size of the low-gradient region, and the latter controls the sensitivity of local feature change.

4 Experiments

In this section, we tested our proposed method on three publicly released multisensor remote sensing datasets, namely Postdam dataset,[48] Vaihingen dataset[49] (both of these are from the ISPRS 2-D semantic labeling contest), and Zeebruges dataset[50] from the IGARSS 2015 data fusion contest. We also conducted several experiments to validate the robustness of our fusion scheme by integrating our higher-order CRF framework with two different segmentation algorithms (GSEG and SLIC) and classifiers (MLR and SVM), which are used to generate the complimentary prediction with LiDAR features.

4.1 *Remote Sensing Datasets*

Potsdam has a ground sampling distance of 5 cm and includes optical orthophotos with four spectral channels, IR, R, G, B and the corresponding coregistered normalized DSMs. The entire collection is divided into 38 image patches, from which 24 images with ground truth labels are used for training, and the remaining 14 images are employed for testing. The Vaihingen dataset has a ground sampling distance of 9 cm and consists of 33 images, with 17 images for training and 16 for testing. The optical images have three spectral channels only: IR, R, and G are accompanied by the coregistered DSMs. The imaging data of Zeebruges were acquired using an airborne platform flying at the altitude of 300 m over the urban and harbor areas of Zeebruges, Belgium (51.33 N, 3.20 E). The data were simultaneously collected and georeferenced to WGS-84. The point density for the LiDAR sensor was $\sim$65 points/m^2, which translates to a point spacing of $\sim$10 cm. The color orthophotos were taken at nadir and had a spatial resolution of about 5 cm. The dataset is organized into seven separate tiles. Each tile includes a $10{,}000 \times 10{,}000$ pixel-sized portion of the color orthophoto (GeoTIFF, RGB), and

(a) (b) (c)

Fig. 8 Example of training image from the Zeebruges dataset. (a) RGB orthophoto, (b) digital surface model, and (c) ground-truth labels.

Color legend	Class	Potsdam dataset	Vaihingen dataset	Zeebruges dataset
	Building	✓	✓	✓
	Car	✓	✓	✓
	Tree	✓	✓	✓
	Low vegetation	✓	✓	✓
	Impervious surface	✓	✓	✓
	Clutter	✓	×	✓
	Boat	×	×	✓
	Water	×	×	✓

Fig. 9 Semantic labeling color legend and the corresponding categories that are classified in each dataset.

a corresponding 5000×5000 DSM. Five of the tiles are used for training, and the remaining two for testing.

In our work, we selected the following five classes: impervious surface, buildings, low vegetation, trees, and cars from the Potsdam and Vaihingin given that their ground truth is readily available. For Zeebruges, we selected two additional categories of interest: boats and water for the same reasoning. Figure 8 shows one of the training images in Zeebruges along with its DSM and ground truth. Figure 9 shows the color legend for each class and the corresponding categories that were identified in each dataset.

4.2 *Training the Fully-Convolutional Neural-8s Network*

To effectively train and test our proposed algorithm, we selected an image resolution of 224×224 for both ground truth and test images, respectively. The image patches could be cropped from the dataset in numerous ways. In our experiments, we generated the training set using the following guidelines: (a) for each class, randomly select nonoverlapping image patches of size 224×224 from each image, (b) for each class, randomly select 1000 pixels in each training image and obtain a 224×224 patch with selected pixel as starting point, and (c) randomly choose 50 cars from the training images to ensure adequate car samples (and also select additional boats in Zeebruges dataset). The process outlined above yielded a training image set of 43,516, 18,780, and 36,000, using the Potsdam dataset, Vaihingen dataset, and Zeebruges dataset, respectively.

We initialized the parameters of the FCN-8s network with pretrained weights, which were obtained using a large dataset of color images and corresponding labels.[20] The weights were then fine-tuned using the corresponding training dataset.

The training process includes two stages. In stage 1, we initialized the weights for convolutional layers 1 through 5, and fully connected layers 6 and 7 with pretrained weights that were kept constant during the first learning phase. We then train the remaining layers—with the exception of the connection and score layers—using randomly initialized weights. This training was

done with a learning rate of 1×10^{-3} for 35 epochs. The learning rate was decreased by 0.1 after 15 and 30 epochs. In stage 2, we set the parameters using the learned weights from the previous stage and then fine-tuned all of the layers with a reduced learning rate of 1×10^{-5} for 35 epochs. Again, the learning rate was decreased by 0.1 after 15 and 30 epochs. A stochastic gradient descent algorithm was utilized for the fine-tuning using the Caffe [Caffe: Convolutional Architecture for Fast Feature Embedding] toolbox.[51] For the test images, and to avoid blocky artifacts, we chose tiles with a stride of 112, i.e., with an overlap rate of 50%. The tiles were then processed by the network for classification.

Our method only requires training one network compared with other competitive deep learning-based fusion frameworks, such as Refs. 5 and 52. The number of parameters for training one FCN-8s is 134,279,076 and for two FCN-8s is 268,558,152. The training time for one FCN-8s network takes 8 h and 15 min for the first 35 epochs and 20 h and 30 min for the second 35 epochs. While for training two FCN-8s, it takes 10 h 49 min for the first 35 epochs and 20 h and 45 min for the second 35 epochs. Training one neural network consumes much less memory as well as takes less training time.

4.3 *Learning the Higher-Order Conditional Random Field Model*

The classification results of the higher-order CRF are, in general, affected by the quality of the segmentation map,[5,29] which is employed by the algorithm. To minimize misclassification, we utilized the GSEG algorithm[13] to provide all underlying segmentations due to its optimal performance against the state of the art.[53] To this effect, we have generated multiple segmentations using various parameters to test the robustness of our proposed higher-order CRF algorithm. The results were compared against the available ground truth that was generated from the training dataset, where each segment articulates the proper object boundaries.

We found that different segmentations do affect the labeling performance of our higher-order CRF fusion method. At the initial seed size of 15 pixels, the vehicle F_1-score is even lower than the one without using CRF and fusion of LiDAR (see the comparisons in Table 1, where HCRF_# means that it uses our proposed higher-order CRF framework, and the segments are obtained by using GSEG algorithm with initial seed size of #). With an increase in the initial seed size, the vehicle F_1-score improves a noticeable amount, and every other category's F_1-score keeps increasing and peaks at the initial seed size of 100. Comparing with using the ground truth segmentation, our proposed method achieves an overall accuracy of 96.05%. This demonstrates that our proposed higher-order CRF can leverage the global knowledge if given

Table 1 Quantitative results of three validation images with different segments. FCN-8s_CIR: fully-convolutional neural network trained on only IR, R, and G channels; nDSM: multisensor fusion with normalized DSM; PCRF: pairwise CRF; HCRF_#: higher-order CRF using segments with initial seed size of #. HCRF_GT: ground truth segmentation, used as our upper-bound performance.

Method	Average F_1-score per class on three validation images					Avg. F_1-score (SD)	Overall Acc. (%)
	Imp. surf.	Building	Low veg.	Tree	Car		
FCN-8s_CIR	0.8844	0.9479	0.8650	0.8280	0.9388	0.8928 (0.05)	88.32
nDSM + PCRF	0.8914	0.9530	0.8658	0.8290	0.9354	0.8949 (0.05)	88.61
nDSM + HCRF_15	0.8985	0.9604	0.8712	0.8311	0.9010	0.8924 (0.047)	89.15
nDSM + HCRF_50	0.9012	0.9612	0.8712	0.8315	0.9394	0.9009 (0.046)	89.37
nDSM + HCRF_100	0.9036	0.9634	0.8720	0.8317	0.9424	0.9026 (0.053)	**89.43**
nDSM + HCRF_150	0.9031	0.9632	0.8719	0.8314	0.9424	0.9023 (0.053)	89.41
nDSM + HCRF_GT	0.9503	0.9822	0.9432	0.9897	0.9683	0.9607 (0.02)	96.05

Note: Bold value indicates the highest overall accuracy achieved using the seed size of 100.

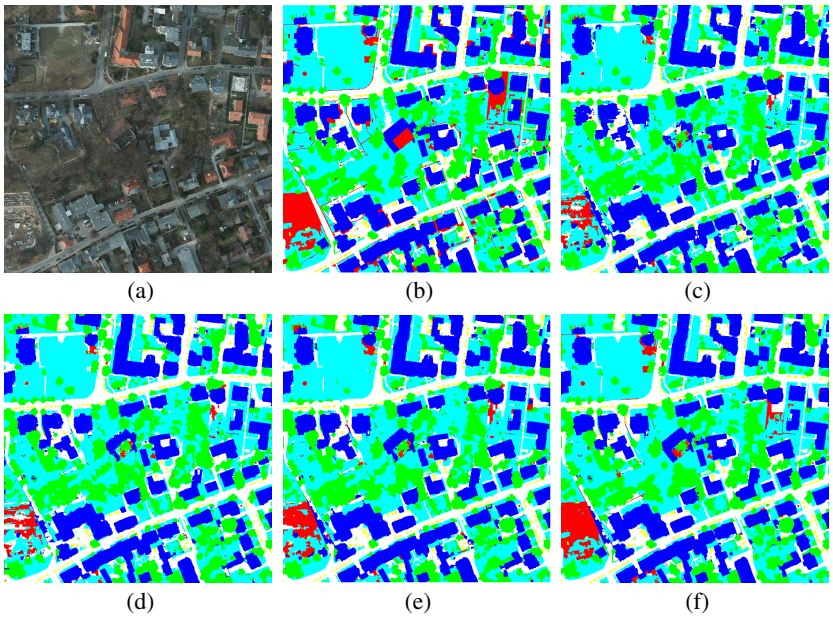


Fig. 10 Semantic labeling results of our proposed higher-order CRF method with different segments. (a) RGB image patch from Potsdam training set, area 4_10, (b) ground-truth labeling, (c) results of FCN-8s trained on IR, R, and G channels only. (d) and (e) Results of our proposed higher-order CRF with GSEG algorithm using initial seed size of 15 and 100, respectively. (f) Classification result of our proposed higher-order CRF with ground-truth segments (upper bound).

correctly. Although we acknowledge that ground-truth segmentation map is rarely accessible by an unsupervised segmentation technique, we would argue that by choosing a suitable segmentation algorithm with its appropriate parameters, applying higher-order CRF tends to improve the final dense semantic labeling (Fig. 10).

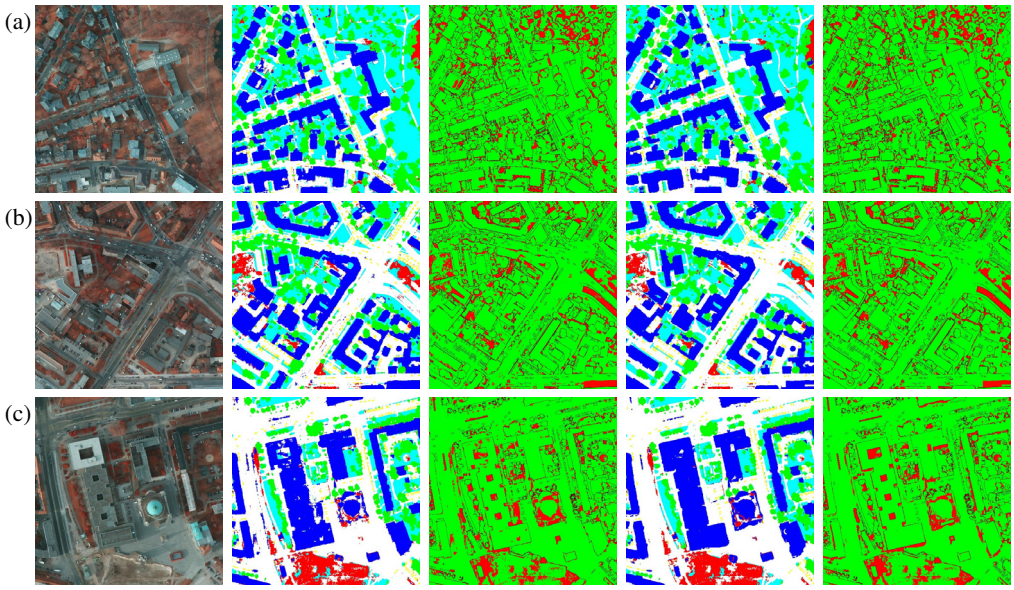


Fig. 11 Results on the Potsdam dataset. Rows I, II, and III illustrate testing patches 3_14, 5_15, and 7_13, respectively. Column #2 shows the results from FCN8s trained only on IR, R, and G channels and column #4 provides the results of our proposed method. The corresponding classification error maps are shown in column #3 and #5, where the red regions indicate the misclassified pixels.

4.4 *Results of the Potsdam Dataset*

We tested our proposed method on the Potsdam dataset and compared its performance qualitatively and quantitatively against the most competitive methods that are published on the ISPRS benchmarking website. The results are shown in Fig. 11, and tabulated in Tables 1 and 3, respectively. SVL_1 serves as the baseline method and does not employ a neural network-based design.[6] UZ_1 utilizes a CNN-based system relying on a downsample-then-upsample architecture to generate dense semantic labeling without any postprocessing. KLab_3 adopts a DCNN for classifying multisensor remote sensed images. DST_6 employs two separate FCNs for optical and LiDAR data, respectively, incorporates a pairwise CRF at the end of their FCN framework to gain a 0.6% boost in overall accuracy and omits the downsampling operation that has been used in most FCNs.[5] The qualitative results of our proposed method can be found in Fig. 11 (denoted as DNN_HCRF in Table 3 and RIT_L7 on the benchmarking website). The results tend to preserve more object boundaries when in comparison with the FCN-8s-based technique (see Fig. 11 column #2). This fact is also shown in Table 3 across the various classes. Compared with Ref. 5, our method eliminates the expense of training the second neural network and utilizes the higher-order CRF framework to improve the overall accuracy by taking advantage of the context information found in most images. This yields more refined object boundaries (especially for buildings) as shown in Fig. 11 and confirmed in Tables 1–3 by the quantitative accuracy. Most of the errors observed in our experiments are the result of inaccuracies in the underlying segmentation maps.

Table 2 Experimental comparison of using two different segmentation algorithms: SLIC[45] and GSEG. In addition, we also compared the fusion of FCN-8s and the prediction with LiDAR features using SVM classifier and MLR.

Method	Average F_1-score per class on three validation images					Avg. F_1-score (SD)	Overall Acc. (%)
	Imp. surf.	Building	Low veg.	Tree	Car		
FCN-8s_CIR	0.8844	0.9479	0.8650	0.8280	0.9388	0.8928 (0.05)	88.32
nDSM + HCRF_100	0.9036	0.9634	0.8720	0.8317	0.9424	0.9026 (0.053)	**89.43**
+ nDSM(SVM) + HCRF_100	0.8945	0.9582	0.8765	0.8320	0.9414	0.9005 (0.051)	89.31
+ nDSM(MLR) + HCRF_SLIC	0.8885	0.9512	0.8653	0.8291	0.9350	0.8938 (0.05)	88.95
+ nDSM(SVM) + HCRF_SLIC	0.8922	0.9532	0.8656	0.8303	0.9364	0.8955 (0.05)	89.01

Note: Bold value indicates the highest overall accuracy achieved using the seed size of 100.

Table 3 Results on 14 test images of the Potsdam dataset. DNN_HCRF: results of using proposed higher-order CRF fusion method. SVL_1, UZ_1, KLab_3, and DST_6 are the published benchmarking methods on ISPRS Potsdam 2-D labeling contest website.

Method	Average F_1-score per class on 14 test images					Avg. F_1-score (SD)	Overall Acc. (%)
	Imp. surf.	Building	Low veg.	Tree	Car		
SVL_1	0.835	0.917	0.722	0.632	0.622	0.7456 (0.129)	77.8
UZ_1[6]	0.893	0.954	0.818	0.805	0.865	0.8670 (0.06)	85.8
KLab_3[54]	0.893	0.92	0.835	0.838	0.92	0.8812 (0.042)	86.4
DST_6[5]	0.924	0.964	0.868	0.877	0.934	0.9134 (0.04)	90.2
DNN_HCRF	0.912	0.946	0.851	0.851	0.928	0.8976 (0.044)	**88.4**

Note: Bold value indicates the highest overall accuracy achieved using the seed size of 100.

4.5 *Results of the Vaihingen Dataset*

The Vaihingen dataset has a slightly lower ground spatial resolution (9 cm) compared with the Potsdam dataset (5 cm). Furthermore, since Vaihingen dataset does not provide the corresponding normalized DSMs, these were generated by manually filtering the ground points. The qualitative and quantitative results—along with the most up to date benchmark comparisons from the ISPRS website—are shown in Fig. 12 and Table 4, respectively. It should be noted that our proposed technique performed adequately well as compared with state of the art. Errors were primarily due to the underlying segmentations and manually generated DSMs that contain large flat regions with gradually changing elevation. Those regions can be misclassified as buildings (e.g., see Fig. 12, row #2, column #5). Note that the Vaihingen dataset contains more flat areas with gradually changing elevation than in Potsdam dataset. Both the FCN-8s and MLR are unable, in general, to correctly recognize the gradually elevated regions without an accurate nDSM.

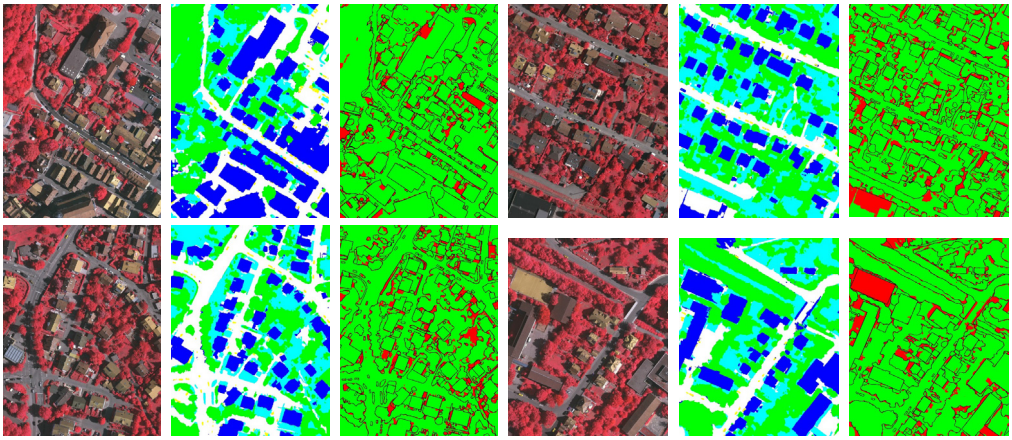


Fig. 12 Results on the Vaihingen dataset. Column #1 and #4 include four Vaihingen dataset test patches, namely area_6 (column#1 and row#1), area_12 (column#4 and row#1), area_22 (column#1 and row#2), and area_20 (column#4 and row#2). Column#2 and column#5 are the labeling results of our proposed method. Column#3 and column#6 are the error maps where the red regions indicate the misclassified pixels.

Table 4 Results on 17 test images of the Vaihingen dataset. SVL_3, ADL_3, ONE_7, UZ_1, DLR_10, and UOA are the published benchmarking results on the ISPRS Vaihingen 2-D labeling contest website. Our proposed method is denoted as DNN_HCRF in the paper or RIT_L7 on the benchmarking website.

	Average F_1-score per class on 17 test images						
Method	Imp. surf.	Building	Low veg.	Tree	Car	Avg. F_1-score (SD)	Overall Acc. (%)
SVL_3	0.866	0.91	0.77	0.85	0.556	0.7904 (0.141)	84.8
ADL_3[39]	0.895	0.932	0.823	0.882	0.633	0.8387 (0.119)	88.0
ONE_7[7]	0.91	0.945	0.844	0.899	0.778	0.8752 (0.065)	89.8
UZ_1[6]	0.892	0.925	0.816	0.869	0.573	0.8150 (0.141)	87.3
DLR_10[52]	0.923	0.941	0.841	0.90	0.793	0.8796 (0.061)	90.3
UOA[55]	0.898	0.921	0.804	0.882	0.82	0.8650 (0.051)	87.6
DNN_HCRF	0.901	0.932	0.814	0.872	0.72	0.8478 (0.084)	87.8

Note: Bold value indicates the highest overall accuracy achieved using the seed size of 100.

Table 5 Results on the Zeebruges dataset. HSVDGr/SVM, blesaux, and RGBd^+ trained AlexNet are the most recent benchmarking results presented in the review paper.[10] FCN-8s only uses FCN-8s without CRFs. DNN_HCRF is the proposed method.

Method	Overall accuracy (%)	Cohen Kappa
HSVDGr/SVM[10]	73.60	0.65
blesaux	76.56	—
RGBd^+ trained AlexNet[10]	83.32	0.78
FCN-8s only	85.50	0.81
DNN_HCRF	**87.85**	**0.84**

Note: Bold value indicates the highest overall accuracy achieved using the seed size of 100.

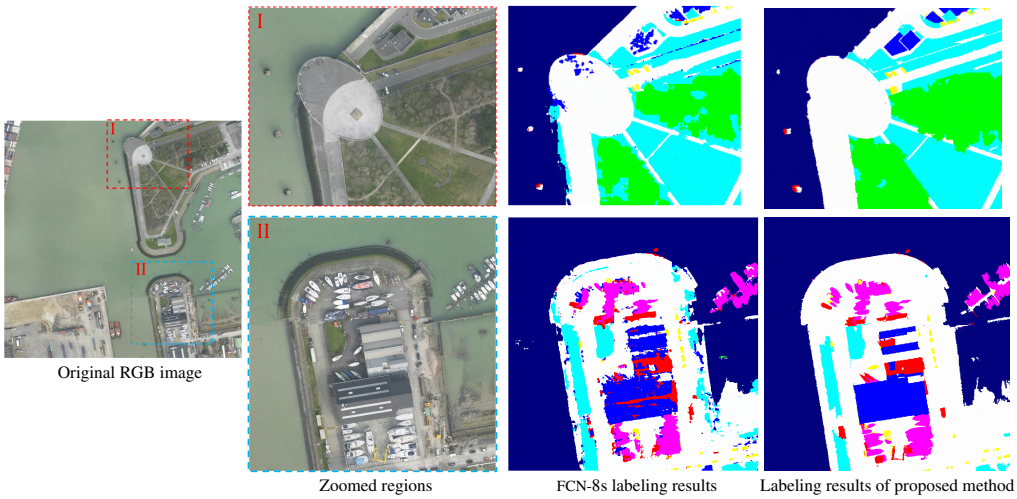


Fig. 13 Qualitative results on the Zeebruges dataset. The illustration demonstrates that our proposed higher-order CRF can produce more accurate object boundaries compared with FCN-8s as seen from the zoomed images.

4.6 *Results of the Zeebruges Dataset*

The Zeebruges dataset was first introduced in the 2015 IEEE GRSS data fusion contest. The most competitive experimental results of this contest have been published in Ref. 10. The best numerical result reported in Ref. 10 is achieved by completely retraining a CNN end-to-end using R, G, B, and height data. Several non-CNN-based techniques are listed as the baseline methods. As shown in Table 5, the results using a pretrained FCN-8s outperform the best results reported in the 2015 contest. Our proposed method further improves the overall accuracy by a large margin (2.35%) compared with only applying FCN-8s. Figure 13 shows the visual improvements in object boundaries by using our proposed higher-order CRF. This is attributed to the high-resolution (ground resolution of 5 cm) RGB images and the extremely high density of LiDAR data. VHR aerial imagery in the Zeebruges results in a high quality of segmentation, which subsequently boosts the performance of our proposed higher-order CRF framework.

5 Discussion

In this section, we will discuss how we fine-tune the pretrained neural networks, the learning of hyper parameters of the higher-order CRF, and the advantages of the higher-order CRF framework.

Table 6 Classification results using only FCN8s with two different training strategies. The one-stage training method is denoted as DNN_T1, and the two-stage training method is denoted as DNN_T2.

	Average F_1-score per class on 14 test images						
Method	Imp. surf.	Building	Low veg.	Tree	Car	Avg. F_1-score (SD)	Overall Acc. (%)
DNN_T1	0.887	0.915	0.822	0.822	0.908	0.8708 (0.044)	85.5
DNN_T2	0.907	0.939	0.848	0.851	0.924	0.8938 (0.042)	**87.8**

Note: The two-stage training method achieves better overall accuracy and is labeled as bold.

5.1 *Fine-Tuning of the Pretrained Neural Network*

In addition to the training procedure introduced in Sec. 4.2, we also trained the FCN-8s neural network using a one-stage (namely DNN_T1) instead of a two-stage training strategy (namely DNN_T2). To this effect, we initialized the parameters of convolutional layers 1 to 5 and fully connected layer 6 and 7 with pretrained weights in Ref. 20 and assigned random weights for the remaining layers. The parameters of convolutional layers 1 to 5 were held constant during the training, and the parameters of the other layers were fine-tuned with a learning rate at 1×10^{-3}, 35 epochs, multiplying learning rate at 15 and 30, up by 0.1. We evaluated these two training strategies on the Potsdam testing dataset and found that the two-stage training strategy outperformed the one-stage training strategy regarding overall classification accuracy by 2.3% as shown in Table 6.

As our experiments showed, different training strategies for learning fully convolutional neural networks parameters have a significant impact on the overall accuracy of semantic labeling. We think that the second step of fine-tuning all the layers in the two-stage training strategy is attributed to the performance improvement. Because the pretrained neural networks are usually trained based on general viewing perspective of objects, the shallow convolutional layers also need to be fine-tuned to adapt to the different appearance of aerial images. We found that changing the fine-tuning strategy significantly impacts the classification results.

5.2 *Parameter Learning of the Higher-Order Term*

Based on Eq. (18), to achieve the optimal performance of the higher-order CRF, we need to train the hyper-parameter $\frac{1}{Q}$. $\frac{1}{Q}$ can be interpreted as a truncation term that determines the percentages of the number of pixels in one segment are allowed to take a different label from the dominant label for the segment. We refer the readers to this paper[14] for more details on the hyperparameters training process.

5.3 *Higher-Order CRF Modeling*

We have discussed the quantitative improvements of using the higher-order CRFs compared to only using the pairwise CRFs in the previously published paper.[14] We have seen the same quantitative and qualitative improvements for the Vaihingen and Zeebruges dataset. Especially for the Zeebruges dataset, as it has the even higher spatial resolution RGB images, the higher-order CRF gains more advantages in terms of resolving the object boundaries (see Fig. 13).

The need for a higher-order CRF is discussed in Ref. 39, in which the authors argued that a higher-order CRF sometimes had an adverse impact on classification accuracy. We agree on the point that the performance of higher-order CRFs is sensitive to the quality of segments, which is scene dependent. As shown in Table 1, the vehicle F_1-score drops when higher order CRF takes a small initial seed size, which resulted in over-segmented objects. Based on our experiments, as long as we choose a proper segmentation algorithm and find its appropriate parameters, using higher-order CRF gains an overall quantitative and qualitative improvement for dense semantic labeling compared with only using pairwise CRF. Furthermore, incorporating a higher-order

CRF provides potential opportunities for further improvement by utilizing object-level contextual information in a hierarchical random field, as proposed in Refs. 30–32.

The main contribution of our paper is mainly focusing on the qualitative and quantitative improvements by introducing the higher-order CRFs after we obtained the probabilistic results from the state-of-the-art CNNs. The individual results from each CNN architecture may differ. As we treated these two parts (CNNs and CRFs) separately, we expect improvements even if we use other CNN architectures. Higher-order CRFs leverage the boundary and context information that is not necessarily learned by CNNs. The performance of our approach combining with other CNN architectures needs more investigation, but one would expect incremental improvement in accuracy by carefully comparing and selecting other deep learning architectures.

6 Conclusion

We proposed a decision-level multimodal classification method for dense semantic labeling of VHR aerial optical imagery and its coregistered LiDAR data. An FCN network and MLR were utilized for generating the initial predictions for the optical imagery and LiDAR data, respectively. We proposed a higher-order CRF fusion method to combine the predictions from each sensor and to simultaneously reason about long-term relations between objects in the scene. We demonstrated and analyzed the performance of the proposed method with experiments using remote sensing benchmark data, the Potsdam, and Vaihingen datasets. The performance of our proposed higher-order CRF model is affected by three main factors: (1) the accuracy of the initial probabilistic predictions, (2) the quality of the low-level segmentation, and (3) the hyperparameters of the training algorithm.

Acknowledgments

The authors would like to acknowledge the Department of Defense for its support of this research as well as the usage of the dataset provided by ISPRS and BSF Swissphoto, released in conjunction with the ISPRS, led by ISPRS WG II/4. The authors also would like to thank the Belgian Royal Military Academy for acquiring and providing the data used in this study, and the IEEE GRSS Image Analysis and Data Fusion Technical Committee.

References

1. M. Xu, H. Chen, and P. K. Varshney, "An image fusion approach based on Markov random fields," *IEEE Trans. Geosci. Remote Sens.* **49**(12), 5116–5127 (2011).
2. A. H. S. Solberg, T. Taxt, and A. K. Jain, "A Markov random field model for classification of multisource satellite imagery," *IEEE Trans. Geosci. Remote Sens.* **34**(1), 100–113 (1996).
3. G. Moser, S. B. Serpico, and J. A. Benediktsson, "Land-cover mapping by Markov modeling of spatial–contextual information in very-high-resolution remote sensing images," *Proc. IEEE* **101**(3), 631–651 (2013).
4. L. Cianci, G. Moser, and S. Serpico, "Change detection from very highresolution multisensor remote-sensing images by a Markovian approach," in *Proc. IEEE-GOLD* (2012).
5. J. Sherrah, "Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery," arXiv:1606.02585 (2016).
6. M. Volpi and D. Tuia, "Dense semantic labeling of subdecimeter resolution images with convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.* **55**(2), 881–893 (2017).
7. N. Audebert, B. L. Saux, and S. Lefèvre, "Semantic segmentation of earth observation data using multimodal and multi-scale deep networks," in *Asia Conf. on Computer Vision*, pp. 180–196, Springer, Cham (2016).
8. D. Marmanis et al., "Deep learning earth observation classification using imagenet pretrained networks," *IEEE Geosci. Remote Sens. Lett.* **13**(1), 105–109 (2016).
9. E. Maggiori et al., "High-resolution semantic labeling with convolutional neural networks," arXiv:1611.01962 (2016).

10. M. Campos-Taberner et al., "Processing of extremely high-resolution Lidar and RGB data: outcome of the 2015 IEEE GRSS data fusion contest–part a: 2-D contest," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.* **9**(12), 5547–5559 (2016).
11. X. Chen et al., "Vehicle detection in satellite images by hybrid deep convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.* **11**(10), 1797–1801 (2014).
12. M. Castelluccio et al., "Land use classification in remote sensing images by convolutional neural networks," arXiv:1508.00092 (2015).
13. L. Ugarriza et al., "Automatic image segmentation by dynamic region growth and multiresolution merging," *IEEE Trans. Image Process.* **18**(10), 2275–2288 (2009).
14. Y. Liu et al., "Dense semantic labeling of very-high-resolution aerial imagery and lidar with fully-convolutional neural networks and higher-order CRFs," in *IEEE Conf. Computer Vision and Pattern Recognition Workshops (CVPRW)* (2017).
15. L. Gómez-Chova et al., "Multimodal classification of remote sensing images: a review and future directions," *Proc. IEEE* **103**(9), 1560–1584 (2015).
16. C. Debes et al., "Hyperspectral and LiDAR data fusion: outcome of the 2013 GRSS data fusion contest," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.* **7**(6), 2405–2418 (2014).
17. O. Penatti, K. Nogueira, and J. A. dos Santos, "Do deep features generalize from everyday objects to remote sensing and aerial scenes domains?" in *IEEE Conf. Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 44–51 (2015).
18. M. Alonzo, B. Bookhagen, and D. A. Roberts, "Urban tree species mapping using hyperspectral and LiDAR data fusion," *Remote Sens. Environ.* **148**, 70–83 (2014).
19. V. Mnih and G. Hinton, "Learning to detect roads in high-resolution aerial images," *Lect. Notes Comput. Sci.* **6316**, 210–223 (2010).
20. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440 (2015).
21. V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: a deep convolutional encoder-decoder architecture for image segmentation," arXiv:1511.00561 (2015).
22. L.-C. Chen et al., "Semantic image segmentation with deep convolutional nets and fully connected CRFs," arXiv:1412.7062 (2014).
23. A. Romero, C. Gatta, and G. Camps-Valls, "Unsupervised deep feature extraction for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.* **54**(3), 1349–1362 (2016).
24. B. Waske and J. A. Benediktsson, "Fusion of support vector machines for classification of multisensor data," *IEEE Trans. Geosci. Remote Sens.* **45**(12), 3858–3866 (2007).
25. J. D. Wegner, J. A. Montoya-Zegarra, and K. Schindler, "A higher-order CRF model for road network extraction," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pp. 1698–1705 (2013).
26. E. Li et al., "Robust rooftop extraction from visible band images using higher order CRF," *IEEE Trans. Geosci. Remote Sens.* **53**(8), 4483–4495 (2015).
27. S. Kluckner et al., "Semantic classification in aerial imagery by integrating appearance and height information," *Lect. Notes Comput. Sci.* **5995**, 477–488 (2010).
28. D. Marmanis et al., "Semantic segmentation of aerial images with an ensemble of CNSS," *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **III-3**, 473–480 (2016).
29. P. Kohli and P. H. Torr, "Robust higher order potentials for enforcing label consistency," *Int. J. Comput. Vision* **82**(3), 302–324 (2009).
30. L. Ladick? et al., "Associative hierarchical random fields," *IEEE Trans. Pattern Anal. Mach. Intell.* **36**(6), 1056–1077 (2014).
31. X. Boix et al., "Harmony potentials," *Int. J. Comput. Vision* **96**(1), 83–102 (2012).
32. L. Ladicky et al., "Graph cut based inference with co-occurrence statistics," *Lect. Notes Comput. Sci.* **6315**, 239–253 (2010).
33. K. Philipp and V. Koltun, "Efficient inference in fully connected CRFs with Gaussian edge potentials," in *Adv. Neural Inf. Process. Syst.* **2**(3), 109–117 (2011).
34. J. Marcello, A. Medina, and F. Eugenio, "Evaluation of spatial and spectral effectiveness of pixel-level fusion techniques," *IEEE Geosci. Remote Sens. Lett.* **10**(3), 432–436 (2013).
35. W. Li, S. Prasad, and J. Fowler, "Decision fusion in kernel-induced spaces for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.* **52**(6), 3399–3411 (2014).

36. C. Pohl and J. L. Van Genderen, "Review article multisensor image fusion in remote sensing: concepts, methods and applications," *Int. J. Remote Sens.* **19**(5), 823–854 (1998).
37. J. A. Benediktsson, P. H. Swain, and O. K. Ersoy, "Neural network approaches versus statistical methods in classification of multisource remote sensing data," *IEEE Trans. Geosci. Remote Sens.* **28**(4), 540–552 (1990).
38. N. Audebert, B. Le Saux, and S. Lefèvre, "Beyond RGB: very high resolution urban remote sensing with multimodal deep networks," *ISPRS J. Photogramm. Remote Sens.* **140**, 20–32 (2017).
39. S. Paisitkriangkrai et al., "Effective semantic pixel labelling with convolutional networks and conditional random fields," in *IEEE Conf. Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 36–43 (2015).
40. H. Khurshid and M. F. Khan, "Segmentation and classification using logistic regression in remote sensing imagery," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **8**(1), 224–232 (2015).
41. J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Adv. Large Margin Classifiers* **10**(3), 61–74 (1999).
42. S. S. Panda, D. P. Ames, and S. Panigrahi, "Application of vegetation indices for agricultural crop yield prediction using neural network techniques," *Remote Sens.* **2**(3), 673–696 (2010).
43. N. Pettorelli et al., "Using the satellite-derived NDVI to assess ecological responses to environmental change," *Trends Ecol. Evol.* **20**(9), 503–510 (2005).
44. H. Nouri et al., "High spatial resolution worldview-2 imagery for mapping NDVI and its relationship to temporal urban landscape evapotranspiration factors," *Remote Sens.* **6**(1), 580–602 (2014).
45. R. Achanta et al., "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(11), 2274–2282 (2012).
46. J. Domke, "Learning graphical model parameters with approximate marginal inference," *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(10), 2454–2467 (2013).
47. V. Kolmogorov and R. Zabin, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(2), 147–159 (2004).
48. "ISPRS 2D semantic labeling contest—Potsdam," http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-potsdam.html (12 May 2017).
49. "ISPRS 2D semantic labeling contest—Vaihingen," http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-vaihingen.html (12 May 2017).
50. "IEEE GRSS data fusion contest," http://www.grss-ieee.org/community/technical-committees/data-fusion (18 September 2017).
51. Y. Jia et al., "Caffe: convolutional architecture for fast feature embedding," in *Proc. of the 22nd ACM Int. Conf. on Multimedia*, pp. 675–678, ACM (2014).
52. D. Marmanis et al., "Classification with an edge: improving semantic image segmentation with boundary detection," *ISPRS J. Photogramm. Remote Sens.* **135**, 158–172 (2018).
53. S. R. Vantaram and E. Saber, "Survey of contemporary trends in color image segmentation," *J. Electron. Imaging* **21**(4), 040901 (2012).
54. R. Kemker and C. Kanan, "Deep neural networks for semantic segmentation of multispectral remote sensing imagery," *CoRR* Vol. abs/1703.06452 (2017).
55. G. Lin et al., "Efficient piecewise training of deep structured models for semantic segmentation," in *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pp. 3194–3203 (2016).

Yansong Liu received his BS degree in electrical and communication department from Shanghai Jiao Tong University, Shanghai, China, in 2008 and his MS degree in electrical and electronics department from Rochester Institute of Technology in 2013. Currently, he is a PhD candidate in the Chester F. Carlson center of imaging science department at Rochester Institute of Technology. His research interests are in the area of digital image/video processing and machine learning, including image segmentation and classification, object recognition, and probabilistic graphic models.

Sankaranarayanan Piramanayagam is currently pursuing his PhD in imaging science at Rochester Institute of Technology (RIT). His thesis is focused on image and video segmentation,

and aerial image classification and his research interests include computer vision, HDR, and machine learning. He received his Bachelor of Engineering in electronics and instrumentation at Easwari Engineering College from the Anna University, India, and his MS degree in electrical and microelectronic engineering from RIT.

Sildomar T. Monteiroa served as a guest editor of the *Journal of Field Robotics* special issue on alternative sensing techniques for robot perception in 2015 and organized a workshop on the same topic at the Robotics Science and Systems Conference in 2012. He served as a program cochair of the Australasian Conference on Robotics and Automation in 2009. He was a recipient of the prestigious Japan Society for the Promotion of Science Postdoctoral Fellowship in 2007. He serves as vice-chair of the IEEE Geoscience and Remote Sensing Society's Western New York Chapter since 2016. He is a member of the IEEE and ACM.

Eli Saber is a professor in the electrical and microelectronic engineering department and the Chester F. Carlson Center for Imaging Science at the Rochester Institute of Technology. He received his BS degree in electrical and computer engineering from the University of Buffalo in 1988 and his MS and PhD degrees in the same discipline from the University of Rochester in 1992 and 1996, respectively. From 1997 until 2004, he was an adjunct faculty member at the electrical engineering department of the Rochester Institute of Technology and at the Electrical & Computer Engineering Department of the University of Rochester, responsible for teaching undergraduate and graduate coursework in signal, image, and video processing, pattern recognition and communications, and performing research in multimedia applications, pattern recognition, image understanding, and color engineering.