Dense Semantic Labeling of Very-High-Resolution Aerial Imagery and LiDAR with Fully-Convolutional Neural Networks and Higher-Order CRFs

Yansong Liu Sankaranarayanan Piramanayagam Sildomar T. Monteiro Eli Saber
Rochester Institute of Technology
1 Lomb Memorial Dr, Rochester, NY 14623, USA
{yxl3624,sxp1899,sildomar.monteiro,esseee}@rit.edu

Abstract

The increasing availability of very-high-resolution (VHR) aerial optical images as well as coregistered LiDAR data opens great opportunities for improving object-level dense semantic labeling of airborne remote sensing imagery. As a result, efficient and effective multisensor fusion techniques are needed to fully exploit these complementary data modalities. Recent researches demonstrated how to process remote sensing images using pre-trained deep convolutional neural networks (DCNNs) at the feature level. In this paper, we propose a decision-level fusion approach using a probabilistic graphical model for the task of dense semantic labeling. Our proposed method first obtains two initial probabilistic labeling predictions from a fully-convolutional neural network and a linear classifier, e.g. *logistic regression, respectively. These two predictions are then combined within a higher-order conditional random field (CRF). We utilize graph cut inference to estimate the final dense semantic labeling results. Higher-order CRF modeling helps to resolve fusion ambiguities by explicitly using the spatial contextual information, which can be learned from the training data. Experiments on the ISPRS 2D semantic labeling Potsdam dataset show that our proposed approach compares favorably to the state-of-the-art baseline methods.*

1. Introduction

Dense semantic labeling for the aerial images of urban regions with complex configurations has been a challenging task for remote sensing applications. Two significant challenges prevented researchers from obtaining the accurate and detailed semantic labeling results: 1) the spatial resolution of the aerial imaging systems were not fine enough to capture relatively small individual objects (*e.g.* vehicles) in the urban environment. Besides coarser resolutions restrict the use of more expressive image features and more sophisticated classification methods. 2) Aerial images typically have a top side view of the scene, whose viewing perspectives differ from the ones in other general computer vision tasks. Without contextual information, objects that are easy to differentiate on the ground can be much less distinguishable from the top side view as aerial imagery presents.

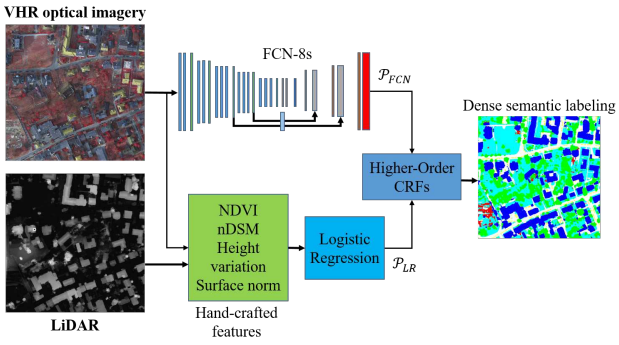

Figure 1. Our proposed decision-level fusion scheme: training one fully-convolutional neural network on the color-infrared image (CIR) and one logistic regression using hand-crafted features. Two probabilistic results: P_{FCN} and P_{LR} are then combined in a higher-order CRF framework.

The first challenge has been well addressed recently, thanks to the advances of the very-high-resolution (VHR) aerial imagery, which now has the ground spatial resolution of around 10 cm. Under this spatial resolution, more powerful spatial features and sophisticated structured prediction methods can be utilized to generate a more accurate semantic labeling. For instance, several Markov random fields (MRFs) approaches were proposed to take advantage of the increased spatial contextual information of VHR images to improve the performance of land cover mapping and classification [39, 34, 28, 11]. Moreover, the remarkable success of deep convolutional neural networks (DCNNs) on general image classification tasks intrigues more and more remote sensing researchers to explore the use of

DCNNs on the VHR aerial images. However, training DCNNs requires large enough labeled datasets to avoid overfitting. Unfortunately, labeling aerial images that cover large ground regions can be a tedious and extensively labor cost task. The available training data is thereby limited, although the amount of labeled data is increasing dramatically recently to meet such demand.

Recent works on applying DCNNs to remote sensing imagery tackles the training problem by adopting a pre-trained neural network designed for other general image classification tasks (*e.g*. ImageNet) and then fine-tuned one or several convolutional layers on the limited remote sensing data [33, 36, 4, 25]. Despite the differences of viewing perspectives and object scales between general RGB images and aerial imagery, this approach delivers the best classification/semantic labeling results so far. Other remote sensing applications such as vehicle detection [10], scene classification [8] also benefit from the similar scheme.

The second challenge that objects have similar spectral appearances cannot be addressed merely by using DCNNs. There needs another imaging modality that can capture the complementary information about the same observed region. Light detection and ranging (LiDAR) system is one of such sensing technologies that can provide relevant height information that can be used to discriminate ground objects with similar spectral characteristics. On the one hand, the joint use of aerial optical images and its coregistered LiDAR data provide a complete representation of the given scene. On the other hand, the multisensor data poses new challenges for the use of pre-trained DCNNs approaches, since most of the pre-trained DCNNs are specifically designed for RGB three-band images. However, how to use the pre-trained networks for both spectral channels (*e.g*. R, G, B and IR) and the LiDAR data remains an active research topic. One intuitive way to address this issue is to train two separate neural networks: one for optical imagery and another one with artificially created three-band images by using the LiDAR data, *e.g*. DSM, height variation, surface norm, etc. The learned features from two neural networks are then concatenated after certain convolutional layer. This feature-level fusion method has been proved to be relatively successful [33, 4]. But training two neural networks can be computationally expensive. Also, the robustness of training the artificial three-band LiDAR images remains unanswered.

We propose an alternative approach for the joint use of the optical imagery and its corresponding LiDAR data. We first generate the probabilistic outputs for two modalities separately: training a fully-convolutional network [23] on optical imagery and a multinomial logistic regression using hand-crafted LiDAR features. We then feed the weighted outputs of these two classifiers as the unary potential in a higher-order CRF, and we obtained the weights and CRF parameters through the maximum likelihood training.

Main original contributions of our work are: 1) the use of energy based CRFs for efficient decision-level multisensor data fusion for the task of dense semantic labeling. 2) the use of higher-order CRFs for generating labeling outputs with accurate object boundaries. 3) the proposed fusion scheme has a simpler architecture than training two separate neural networks, yet it still yields the state-of-the-art dense semantic labeling results.

2. Related Work

There is a significant amount of works that have been done on classification/semantic labeling of multimodal remote sensing data. We refer readers to the detailed review papers on this topic [14, 12]. Here, we are going to review some of the previous works that are related to dense semantic labeling of VHR aerial imagery and LiDAR data.

2.1. Deep convolutional neural networks (DCNNs)

Due to the increasing availability of VHR aerial imagery, the use of DCNNs for classification/semantic labeling of remote sensing images has become more and more popular. Recent researches have shown that the neural networks that are trained on general image classification tasks can be used as a universal feature extractor for aerial imagery as well [25, 30], despite that the viewing perspectives of the same class category in general images can be quite different in the overhead images. Some works have been done on patch/tile based classification tasks such as scene classification of aerial images [8], vehicle detection [10], tree species mapping [3] and road detection [27]. A deep convolutional neural network requires a down-sampling operation, (*e.g*. max pooling) after convolutional layers to further capture longer range contextual information and extract more abstract features. However, this usually results in a lower spatial resolution of output label map.

Numerous fully-convolutional neural networks (FCNNs) have been proposed to overcome this downsampling effect so that the end-to-end dense prediction can be achieved. Jonathan *et al*. [23] 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. Badrinarayanan *et al*. [5] proposed a semantic pixel-wise segmentation method using a fully-convolutional neural network (SegNet), which uses decoder/deconvolutional layers to map the low-resolution encoder feature maps to the full input resolution feature maps. Chen *et al*. [9] utilized an atrous method to expand the support of the filter and reduce the down-sampling for input feature map to achieve dense labeling. These approaches have been successfully adopted for dense semantic labeling of remote sensing images [33, 36, 32] and outperformed the traditional pixel-

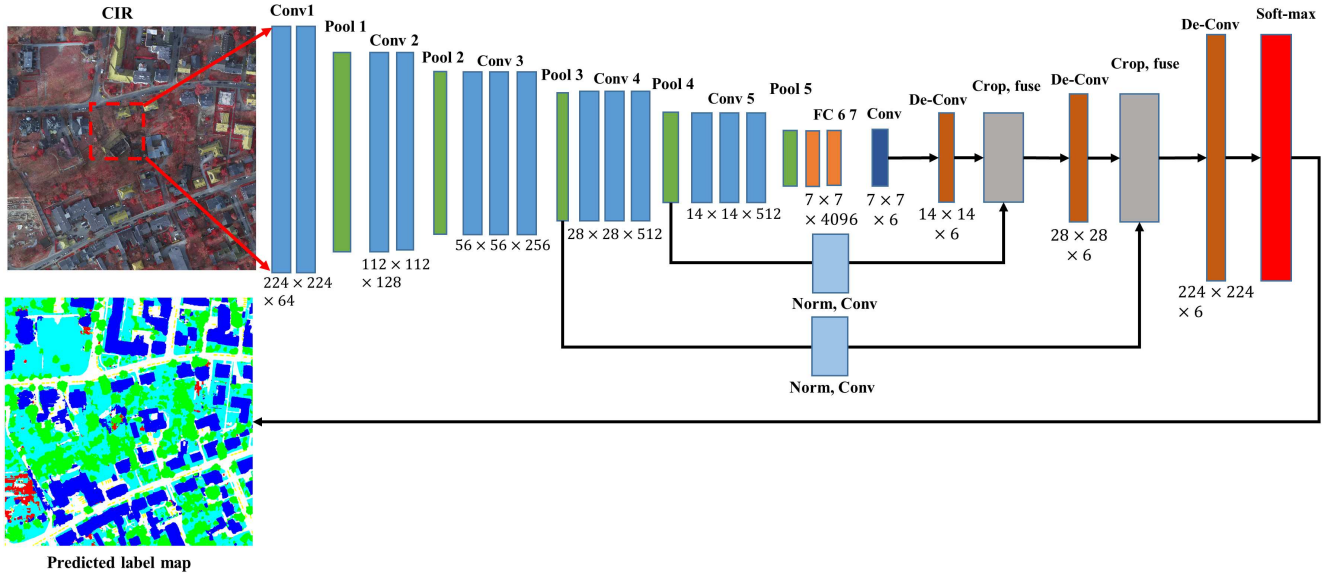

Figure 2. An illustration of FCN-8s architecture. The training process includes two stages. Stage 1: we initialized the weights for convolutional layer 1 to 5, and fully-connected layer 6 and 7 with pre-trained weights and keep them constant during the first learning phase. We then train the rest layers with randomly initialized weights. Stage 2: We set the parameters using the learned weights from the previous stage. We then fine-tune all layers with a slower learning rate.

level classification methods using hand-crafted feature descriptors, such as SVMs [37]. In our paper, we applied the FCN-8s network in [23] to train the VHR optical imagery with a two-stage training scheme and used its per pixel probability output as one of the probabilistic decisions in our fusion framework.

2.2. Random field method

Energy based random field methods (*e.g.* MRFs, CRFs) have been widely used for exploiting contextual information for both general and remote sensing images [39, 34, 28, 38, 21, 15, 26]. The full potential of the random field methods has not been reached due to the limitation of inference methods for higher-order node connections, particularly on the large scale data. Until recently, a lot of excellent works have been proposed to explore the use of higher-order random fields with efficient inference methods [16, 20, 7, 19, 18]. Among them, the one that used the robust higher-order potential and graph cut inference method [16, 20] 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 [38, 21]. The need of higher-order random fields in our work is mainly for 1) resolving decision ambiguity between two probabilistic outputs by enforcing label consistency within one segment. 2) preserving realistic object boundaries with the help of gradient based segmentation (GSEG) algorithm [35].

2.3. Multisensor fusion

In the context of semantic labeling, multisensor fusion typically has two distinct procedures: a) Feature extraction for all data modality followed by fusion at the feature level, this process can include feature concatenation and selection. The fused features later will be fed into a supervised training scheme for classification purpose [24]. b) Another fusion procedure is to use different processing paths for each modality and combine the individual decisions of the set of trained classifiers to obtain the optimal output, which is referred as the decision-level fusion [22, 31, 6]. Both methods have been explored in recent multisensor fusion works. J. Sherrah [33] proposed to train two separate neural networks for each modality and concatenate the learned features at the last convolutional layer. N. Audebert *et al*. [4] compared both fusion procedures. They proposed an averaging strategy for the decision-level fusion and feature correction for feature level fusion. Their results showed that the feature correction performed slightly better. P. Krahenbuhl and V. Koltun [29] proposed a decision-level fusion that has a similar structure as ours. But ours differs from their method in three ways. First, they combined the probability outputs from two classifiers directly in a fixed way, while we learned fusion weights through CRFs on the training data. Second, our method incorporates the higher-order CRFs, and they did not. Last but not the least, we applied fully-convolutional neural networks that learn to combine coarse layer information with fine layer information while they used a multi-resolution CNN as the feature extractor.

3. Model learning

Before we start formulating our higher-order CRFs fusion framework, we need to first obtain two initial probabilistic per-class labeling predictions for each modality, *i.e.* VHR optical imagery and LiDAR data in our case, separately.

3.1. Leaning FCN for VHR optical imagery

VHR optical imagery contains rich low-level and high-level features. To fully take advantage of such information, we applied the fully-convolutional neural network that uses a skip architecture to combine the coarse layer information with finer layer information to yield high-resolution per class probability prediction; this network is referred as FCN-8s [23], whose architecture is shown in Figure 2. 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 sixth and seventh layers are two fully-connected convolutional layers. 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.

We initialized the parameters of the FCN-8s neural network with pre-trained weights[23], which were obtained by training the network on the large dataset of color images and corresponding labels. We then fine-tuned the network on the aerial image training set with IR, R, and G three bands and corresponding ground truth. Specifically, we generated 36,000 images and corresponding ground truth of size 224×224 from the 21 training images by image tiling randomly cropping, and choosing extra data for car category for data balancing.

The training process includes two stages. Stage 1: we initialized the weights for convolutional layer 1 to 5, and fully-connected layer 6 and 7 with pre-trained weights and keep them constant during the first learning phase. We then train the rest layers (skip connection and score layers) with randomly initialized weights (train from scratch). This training was done with a learning rate of 1e-3 for 35 epochs. The learning rate was decreased by 0.1 after 15 and 30 epochs. Stage 2: We set the parameters using the learned weights from the previous stage. We then fine-tune all layers with a reduced learning rate of 1e-5 for 35 epochs. Again, the learning rate was decreased by 0.1 after 15 and 30 epochs. Stochastic gradient descent algorithm was utilized for the fine-tuning and was done using Caffe [Caffe: Convolutional Architecture for Fast Feature Embedding] toolbox. The fine-tuned network was then used to generate per pixel probability maps, which are denoted as P_{FCN}. For the test images, to avoid blocky artifacts, we chose tiles with a stride of 112, *i.e.* with an overlap rate of 50%. Thus, excluding the pixels at the borders, rest of the image has two predictions. The predictions were averaged to obtain the final probability map.

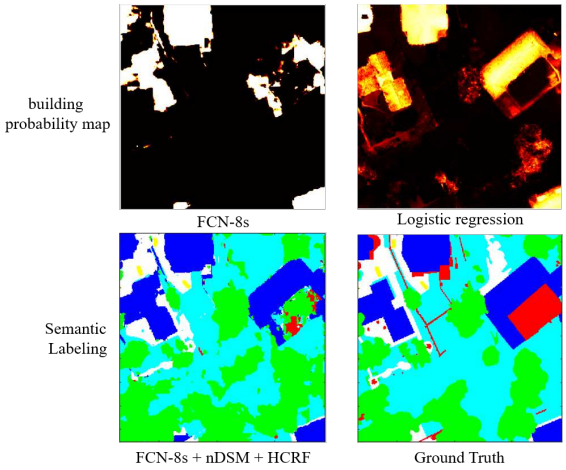

Figure 3. An illustration of the result of using the output of logistic regression to compensate for the miss classification of FCN-8s due to its lack of knowledge of object height. (In the two label maps at the bottom, blue color indicates building class).

3.2. Logistic regression with hand-crafted features

LiDAR data is given as the normalized digital surface maps (nDSMs), which do not have as much contextual information as the VHR optical imagery contains. We, therefore, assumed that a baseline classifier should be enough to take advantage of the LiDAR data. In this paper, we simply chose the multinomial logistic regression with hand-crafted features derived from LiDAR data and optical imagery. The hand-crafted features include height, height variations, surface norm, and the normalized difference vegetation index (NDVI). Regarding the use of NDVI, it is typically used for assessing whether the target being observed contains live green vegetation or not. NDVI can be estimated by:

$$NDVI = \frac{NIR - VIS}{NIR + VIS} \tag{1}$$

where NIR and VIS stand for spectral reflectance measurements in the near-infrared and visible red regions, respectively. For the training process, we randomly chose 10,000 points per class, with a total of 60,000 points for training. The trained multinomial logistic regression model is later used to predict pixel-wise probability map for the test images. The probabilistic output is denoted as P_{LR}.

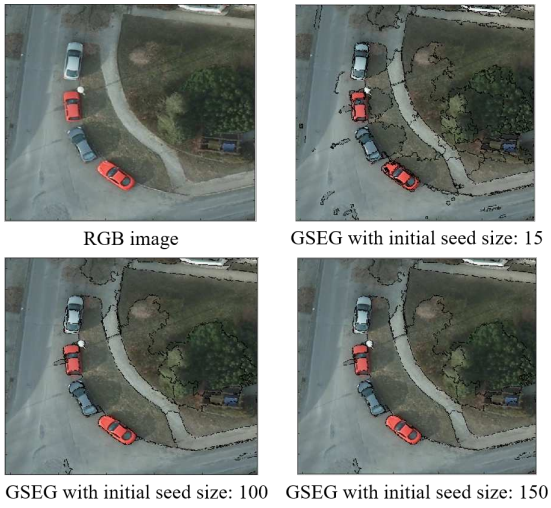

Figure 4. Illustrations of GSEG segmentations with different initial seed sizes: $\tau = 15, 100, 150$.

4. Higher-order CRFs

After we obtained two individual probabilistic outputs: P_{FCN} and P_{LR}, we believe that two such different classifiers must contribute differently to the optimal labeling of a certain class. Therefore we introduce the higher-order CRFs to learn and predict the optimal labels.

4.1. Formulating higher-order CRFs

The pairwise CRFs consist of the unary and pairwise potential, which has the form as:

$$E_p(x) = \sum_{i \in v} \psi_i(x_i) + \sum_{i \in v, j \in N_i} \psi_{ij}(x_i, x_j) \qquad (2)$$

where $x = [x_1, x_2, x_3, \ldots, x_n]$ represents one realization of the label assignment for n pixels: $v = \{1, 2, , n\}$. N_i is the 4-way connected first order neighborhood of pixel i. This connection encodes the shortest range of local context. x_i takes the label from M object classes: $x_i \in L^M$. The unary potential $\psi_i(x_i)$ here takes the form of:

$$\psi(x_i) = -\theta_u ln(\sigma_f P_{FCN}(x_i) + \sigma_l P_{LR}(x_i)) \qquad (3)$$

Where σ_f and σ_l are the fusion weights for estimating how much does fully-convolutional neural network and logistic regression contribute to the unary potential, respectively.

The pairwise potential $\psi_{ij}(x_i, x_j)$ takes the form of color contrast sensitivity Potts model, which is expressed as:

$$\psi_{ij}(x_i, x_j) = (\theta_\alpha + \theta_\beta exp(-\theta_\gamma \|I_i - I_j\|^2)) \cdot \Delta(x_i \neq x_j) \qquad (4)$$

where $\Delta(\cdot)$ is an indicator function and $\|I_i - I_j\|$ is the Euclidean distance between the spectral bands of the pair of pixels (*i.e.* IR, R, G in this paper). In our case, I_i will also include the height component obtained from the normalized digital surface maps (nDSMs). This model makes the assumption that neighboring pixels with similar color appearance and height shall share the same class label. However, the pairwise CRF framework can hardly capture the meaningful spatial context due to its short range connections. Also, it is known that the pairwise potential tends to over-smooth the image boundaries, which is undesired for our dense semantic labeling task.

Higher-order CRF, therefore, is needed to exploit longer range context. Higher-order potentials were added to further enforce the label consistency for the pixels in the same segment. A common higher-order CRF is formed as:

$$E_c(x) = E_p(x) + \sum_{c \in S} \psi_c(x_c) \qquad (5)$$

where S is the set of cliques/segments that are usually generated by an unsupervised segmentation algorithm [2, 35]. $\psi_c(x_c)$ is the higher-order potential defined over the segments. The robust P^N Potts potential proposed by [16] has been proved to be particularly useful. It takes the form of:

$$\psi_c(x_c) = \begin{cases} N_i(x_c)\frac{1}{Q}\gamma_c^{max}, & \text{if } N_i(x_c) < Q \\ \gamma_c^{max}, & \text{otherwise} \end{cases} \qquad (6)$$

where $N_i(x_c)$ denotes the number of pixels that take different labels from the dominant label of the segment. $\gamma_c^{max} = \theta_c|c|$, $|c|$ counts the number of pixels in the segment c. Unlike the standard P^N Potts potential, which strictly forces the pixels in the same segment to take the same label, the robust version of it, however, allows $N_i(x_c)$ pixels in the same segment to take different labels from the dominant label. The heterogeneity of the labeling is controlled by the parameters θ_c and Q. More specifically, the larger Q is, the more heterogeneous one segment can be.

4.2. Gradient-based segmentation (GSEG)

It is critical to choose a robust segmentation algorithm for yielding a dense semantic labeling with fine boundaries. Gradient-based segmentation (GSEG) algorithm is such an unsupervised color image segmentation algorithm that utilizes the gradient histogram acquired from the color images to iteratively cluster pixels from lower gradient to higher gradient. It is followed by a region growing and merging process based on the similarity of segments color and texture. The detailed description of the algorithm can be found in [35]. Since 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 Figure 4. GSEG particularly suits 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, which is controlled by the initial seed size τ and similarity ratio γ. It is, therefore, able to generate heterogeneous segments for different scales of objects.

4.3. Learning CRF parameters

The optimal values for all the parameters of the higher-order CRF were obtained in a manner of step by step training procedure. We first learn the unary potential only to find the fusion weights σ_f and σ_l using the maximum likelihood estimation on the validation data sets. We then keep the fusion weights constant and only train the pairwise CRF parameters: θ_u, θ_α, θ_β and θ_γ without the higher-order term using the method proposed in [13], which takes into account model miss-specification and inference approximation. Finally, the higher-order parameters: θ_c and Q will be learned by performing a cross-validation within an empirical range. The learning results of θ_c and Q will be discussed in Section 5.3.

4.4. Inference using move making graph cuts

Minimizing the proposed energy function (5), *i.e.* $argmin_x E_c(x)$ is not trivial. Fortunately, the number of interesting objects in the urban area is limited to six in our case: Impervious surfaces, building, low vegetation, tree, car, and background. It makes possible for us to utilize the move making graph cuts algorithm to infer the solution effectively.

Move making graph cuts inference algorithm (*i.e.*, α-expansion and $\alpha\beta$-swap) has been successfully used to infer the higher-order CRFs as in [16, 20] We will review the α-expansion graph cut and the trick of adding auxiliary nodes to deal with the higher-order CRFs. We refer readers to [16, 20, 7] for a more detailed explanation.

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. Since each move is a binary operation (*i.e.* maintain the current label or not), we will have to first convert the energy function in the label space into the move space and deduce its corresponding move energy. The transformation function $T_\alpha(\cdot)$ for the α-expansion is:

$$T_\alpha(x_i, t_i) = \begin{cases} \alpha, & t_i = 0 \\ x_i, & t_i = 1 \end{cases} \tag{7}$$

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 $E_c(x)$ is transformed into the problem of optimizing the move energy, *i.e.*, $argmin_t E(T_\alpha(x,t))$, where $E(t)$ is a pseudo-boolean function. This optimization can be achieved in polynomial time by solving a st-mincut as long as $E(t)$ is submodular [17]. For the higher-order term: $\psi_c(x_c)$, it is known that auxiliary binary variables are needed for transforming it into a second order clique. For robust P^N Potts potential, two auxiliary variables are added for each higher-order term. Also, there is an additional label 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.

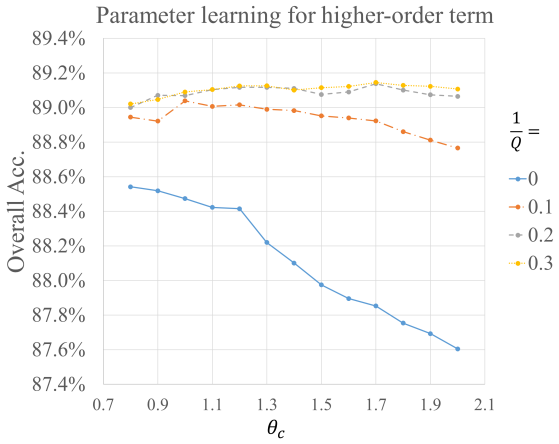

Figure 5. Learning the optimal parameters θ_c and Q for the higher-order term in the three validation images. In practice, instead of tuning Q, we tuned $\frac{1}{Q}$ to represent the percentages of the number of pixels in one segment can take none dominant label.

5. Experiments

5.1. Dataset

The dataset we used in this paper is the 2D semantic labeling contest, Potsdam dataset released by ISPRS Commission II/4 [1]. This dataset includes 38 image patches (each consists of an orthophoto and an nDSM), where 24 images with ground truth labels are used for training, and the rest 14 images are for testing. We then further divided the 24 images with ground truth labels into 21 training images and left out three images for validation, where we trained our CRFs parameters and optimized the GSEG initial seed size.

5.2. One-stage vs. two-stage training of FCN-8s

Except for the training procedure introduced in Section 3.1, we trained the FCN-8s neural network with another one-stage training strategy to test the impact of different training strategies on the labeling performance. For this one-stage training strategy, we initialized the parameters of convolutional layer 1 to 5, and fully-connected layer 6 and 7 with pre-trained weights in [23] and assigned ran-

Table 1. Dense semantic labeling results on three validation images with different training strategies. FCN-8s_CIR: results using fully-convolutional neural network on CIR; TS_2: the two-stage training strategy; nDSM: multisensor fusion with LiDAR data; PCRF: pairwise CRF; HCRF_x: higher-order CRF using segments with initial seed size of x.

Method	Average F_1-score per class on three validation images					Avg. F_1-score	Overall Acc.
	Imp. surf.	Building	Low veg.	Tree	Car		
FCN-8s_CIR + TS_2 (FT_2)	0.8844	0.9479	0.8650	0.8280	0.9388	0.8928	88.32%
FT_2 + nDSM + PCRF	0.8914	0.9530	0.8658	0.8290	0.9354	0.8949	88.61%
FT_2 + nDSM + HCRF_15	0.8985	0.9604	0.8712	0.8311	0.9010	0.8924	89.15%
FT_2 + nDSM + HCRF_50	0.9012	0.9612	0.8712	0.8315	0.9394	0.9009	89.37%
FT_2 + nDSM + HCRF_100	0.9036	0.9634	0.8720	0.8317	0.9424	0.9026	**89.43%**
FT_2 + nDSM + HCRF_150	0.9031	0.9632	0.8719	0.8314	0.9424	0.9023	89.41%

Table 2. Dense semantic labeling results on 14 test images of ISPRS labeling contest Potsdam dataset. FCN-8s_CIR: results using fully-convolutional neural network on CIR; TS_1: one-stage training strategy for FCN-8s; TS_2: the two-stage training strategy for FCN-8s; nDSM: multisensor fusion with LiDAR data; HCRF_x: higher-order CRF using GSEG segments with initial seed size of x.

Method	Average F_1-score per class on 14 test images					Avg. F_1-score	Overall Acc.
	Imp. surf.	Building	Low veg.	Tree	Car		
FCN-8s_CIR+TS_1 (FT_1)	0.887	0.915	0.822	0.822	0.908	0.8708	85.5%
FT_1 + nDSM + PCRF	0.896	0.933	0.830	0.826	0.914	0.8798	86.6%
FT_1 + nDSM + HCRF_15	0.898	0.941	0.830	0.823	0.904	0.8792	86.8%
FT_1 + nDSM + HCRF_100	0.902	0.939	0.830	0.825	0.918	0.8828	86.9%
FT_2	0.907	0.939	0.848	0.851	0.924	0.8938	87.8%
FT_2 + nDSM + HCRF_100	0.912	0.946	0.851	0.851	0.928	0.8976	**88.4%**

dom weights for the rest layers. We kept the parameters of convolutional layers 1 to 5 as constant during the training and fine-tuned other layers weights with a learning rate at 1e-3, 35 epochs, multiplying learning rate at 15 and 30 ep by 0.1. We refer this one-stage training strategy as TS_1 and the two-stage training strategy discussed in Section 3.1 as TS_2. We evaluated these two training strategies on the Potsdam test data set and found that the two-stage training strategy outperformed the one-stage training strategy in terms of overall classification accuracy by 2.3% as shown in Table 2.

5.3. Parameters learning for higher-order term

After we had obtained our pairwise CRF parameters, we learned the optimal parameters of the higher-order term by searching for the values that produce the best overall accuracy of three validation images. To be noticed, we trained $\frac{1}{Q}$ instead of Q for a practical reason. $\frac{1}{Q}$ 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. As Figure 5 shows that when $\frac{1}{Q} > 0.2$ and $\theta_c > 1.5$, the higher-order CRF produces reasonably good results.

5.4. Higher-order CRF with different segments

The labeling results of the higher-order CRF can be affected by the quality of the segments [16, 33]. For this work, we have no means and attempts to test all kinds of different segmentation algorithms in the literature. Instead, we investigated the performances of our proposed higher-order CRF fusion with various choices of initial seed size τ on both leave out validation data sets and test data sets.

We found that initial seed size does 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 on both validation and test data sets. See the comparisons in Table 1. With the increase of the initial seed size, the vehicle F_1-score improves a noticeable amount, and every other categorys F_1-score kept increasing and peaked at the initial seed size of 100. Although the performance of our proposed fusion work fluctuates with the variations of segments, 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.

5.5. Higher-order CRFs vs. pairwise CRFs

As the continuation of Section 5.4, the experiments between using pairwise CRF and higher-order CRF was also performed on both validation and test datasets. Based on the results in Table 1 and 2, with the appropriate segmentation parameters, the higher-order CRF consistently outperforms the pairwise CRF. Quantitatively, the improvements

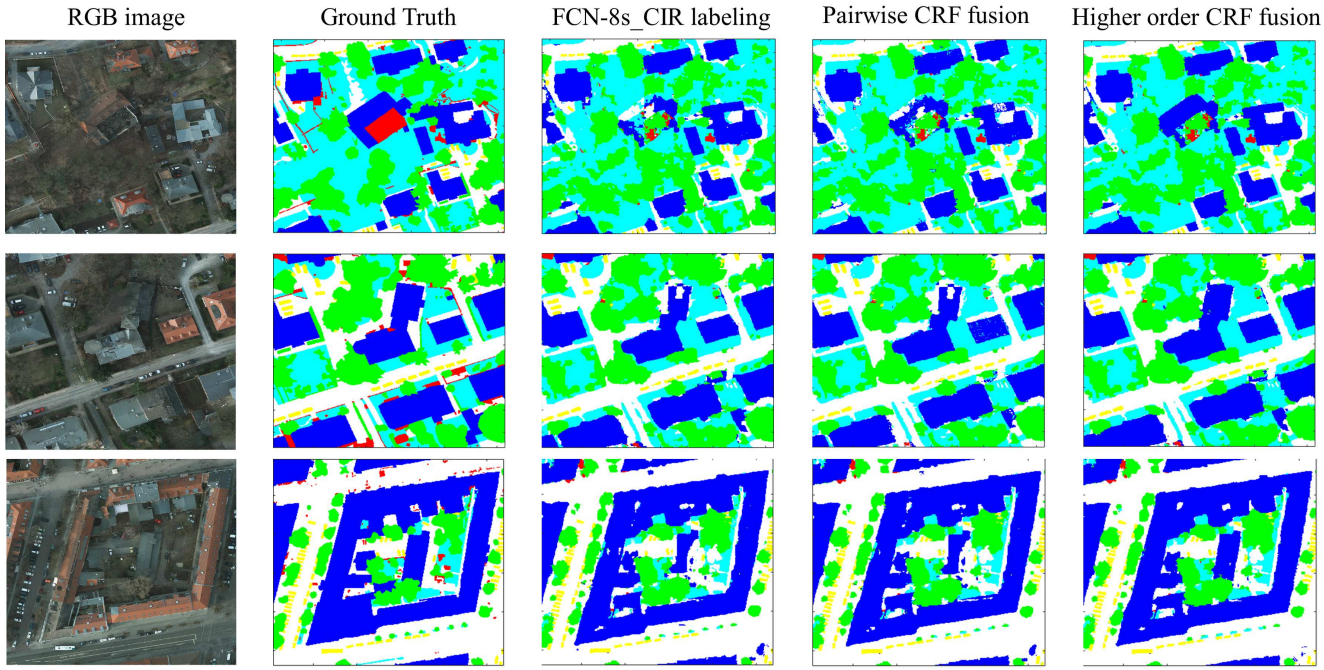

Figure 6. Qualitative results of dense semantic labeling of cropped patches. FCN-8s_CIR labeling is the result of using only the fully-convolutional neural network with CIR. Pairwise CRF fusion shows the result of fusing LiDAR data with FCN-8s_CIR in a pairwise CRF. Higher order CRF fusion is the result of fusing LiDAR data with FCN-8s_CIR in a higher-order CRF, whose segments are generated by GSEG with the initial seed size of 100. The qualitative results show that higher-order CRF tends to preserve better building rooftop.

of using higher-order CRF is not significant enough in some cases. However, the qualitative improvements are quite noticeable especially for building rooftop. See the demonstrations in Figure 6.

6. Discussion

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 pre-trained neural networks are usually trained based on general viewing perspective of objects, therefore the shallow convolutional layers also need to be fine-tuned to compensate for overhead images.

The need of higher-order CRF is discussed in [29], in which the authors argued that higher-order CRF sometimes had an adverse impact on classification accuracy. We agree on the point that the performance of higher-order CRFs is somehow sensitive to the quality of segments, which are scene dependent. As Table 1 shows, the vehicle F_1-score drops when higher order CRF takes a small initial seed size, which can over segment cars. But as we showed in 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 to only using pairwise CRF. Furthermore, incorporating higher-order CRF provides potential opportunities for further improvement by utilizing object-level contextual information in a hierarchical random field as proposed in [20, 7, 19]. We are going to explore this in our future work.

7. Conclusion

In this paper, we proposed a decision-level multisensor fusion method for semantic labeling of VHR aerial imagery and its coregistered LiDAR data. A fully-convolutional neural network and logistic regression classifier are trained for generating individual predictions for the optical imagery and LiDAR data respectively. Two probabilistic classification results are later fused in a higher-order CRF. Based on the experiments on the Potsdam dataset, the proposed higher-order CRF fusion method can yield state-of-the-art semantic labeling results.

8. Acknowledgement

The authors would like to acknowledge the provision of the datasets by ISPRS and BSF Swissphoto, released in conjunction with the ISPRS, led by ISPRS WG II/4.

References

[1] ISPRS 2D semantic labeling contest - Potsdam. http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-potsdam.html. Accessed May 12, 2017.

[2] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(11):2274–2282, 2012.

[3] 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.

[4] N. Audebert, B. L. Saux, and S. Lefèvre. Semantic segmentation of earth observation data using multimodal and multi-scale deep networks. *arXiv preprint arXiv:1609.06846*, 2016.

[5] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv preprint arXiv:1511.00561*, 2015.

[6] 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.

[7] X. Boix, J. M. Gonfaus, J. Van de Weijer, A. D. Bagdanov, J. Serrat, and J. Gonzàlez. Harmony potentials. *Int. J. Comput. Vis.*, 96(1):83–102, 2012.

[8] M. Castelluccio, G. Poggi, C. Sansone, and L. Verdoliva. Land use classification in remote sensing images by convolutional neural networks. *arXiv preprint arXiv:1508.00092*, 2015.

[9] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014.

[10] X. Chen, S. Xiang, C.-L. Liu, and C.-H. Pan. Vehicle detection in satellite images by hybrid deep convolutional neural networks. *IEEE Geosci. Remote Sens. Lett.*, 11(10):1797–1801, 2014.

[11] L. Cianci, G. Moser, and S. Serpico. Change detection from very highresolution multisensor remote-sensing images by a markovian approach. *Proc. of IEEE-GOLD-2012*, 2012.

[12] C. Debes, A. Merentitis, R. Heremans, J. Hahn, N. Frangiadakis, T. van Kasteren, W. Liao, R. Bellens, A. Pižurica, S. Gautama, et al. Hyperspectral and lidar data fusion: Outcome of the 2013 grss data fusion contest. *IEEE J. Sel. Topics Appl. Earth Observ. in Remote Sens.*, 7(6):2405–2418, 2014.

[13] J. Domke. Learning graphical model parameters with approximate marginal inference. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(10):2454–2467, 2013.

[14] L. Gómez-Chova, D. Tuia, G. Moser, and G. Camps-Valls. Multimodal classification of remote sensing images: A review and future directions. *Proc. IEEE*, 103(9):1560–1584, 2015.

[15] S. Kluckner, T. Mauthner, P. Roth, and H. Bischof. Semantic classification in aerial imagery by integrating appearance and height information. In *ACCV*, pages 477–488. Springer, 2010.

[16] P. Kohli and P. H. Torr. Robust higher order potentials for enforcing label consistency. *Int. J. Comput. Vis.*, 82(3):302–324, 2009.

[17] 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.

[18] V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. *Adv. Neural Inf. Process. Syst.*, 2(3):4, 2011.

[19] L. Ladicky, C. Russell, P. Kohli, and P. H. Torr. Graph cut based inference with co-occurrence statistics. In *ECCV*, pages 239–253. Springer, 2010.

[20] L. Ladickỳ, C. Russell, P. Kohli, and P. H. Torr. Associative hierarchical random fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(6):1056–1077, 2014.

[21] E. Li, J. Femiani, S. Xu, X. Zhang, and P. Wonka. Robust rooftop extraction from visible band images using higher order crf. *IEEE Trans. Geosci. Remote Sens.*, 53(8):4483–4495, 2015.

[22] 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.

[23] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, pages 3431–3440, 2015.

[24] 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.

[25] D. Marmanis, M. Datcu, T. Esch, and U. Stilla. Deep learning earth observation classification using imagenet pretrained networks. *IEEE Geosci. Remote Sens. Lett.*, 13(1):105–109, 2016.

[26] D. Marmanis, J. D. Wegner, S. Galliani, K. Schindler, M. Datcu, and U. Stilla. Semantic segmentation of aerial images with an ensemble of cnns. *ISPRS Annals Photogramm. Remote Sens.*, 3:473–480, 2016.

[27] V. Mnih and G. Hinton. Learning to detect roads in high-resolution aerial images. In *ECCV*, pages 210–223. Springer, 2010.

[28] 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.

[29] S. Paisitkriangkrai, J. Sherrah, P. Janney, V.-D. Hengel, et al. Effective semantic pixel labelling with convolutional networks and conditional random fields. In *CVPR Workshops*, pages 36–43, 2015.

[30] O. Penatti, K. Nogueira, and J. A. dos Santos. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In *CVPR Workshops*, pages 44–51, 2015.

[31] 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.

[32] 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.

[33] J. Sherrah. Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery. *arXiv preprint arXiv:1606.02585*, 2016.

[34] 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.

[35] L. Ugarriza, E. Saber, S. R. Vantaram, V. Amuso, M. Shaw, and R. Bhaskar. Automatic image segmentation by dynamic region growth and multiresolution merging. *IEEE Trans. Image Process.*, 18(10):2275–2288, 2009.

[36] 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.

[37] J. A. Waske, B.and Benediktsson. Fusion of support vector machines for classification of multisensor data. *IEEE Trans. Geosci. Remote Sens.*, 45(12):3858–3866, 2007.

[38] J. D. Wegner, J. A. Montoya-Zegarra, and K. Schindler. A higher-order crf model for road network extraction. In *CVPR*, pages 1698–1705, 2013.

[39] 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.