Building Footprint Generation by Integrating Convolution Neural Network With Feature Pairwise Conditional Random Field (FPCRF)

Qingyu Li, Yilei Shi, *Member, IEEE*, Xin Huang, *Senior Member, IEEE*, and Xiao Xiang Zhu, *Senior Member, IEEE*

***Abstract*— Building footprint maps are vital to many remote sensing (RS) applications, such as 3-D building modeling, urban planning, and disaster management. Due to the complexity of buildings, the accurate and reliable generation of the building footprint from RS imagery is still a challenging task. In this article, an end-to-end building footprint generation approach that integrates convolution neural network (CNN) and graph model is proposed. CNN serves as the feature extractor, while the graph model can take spatial correlation into consideration. Moreover, we propose to implement the feature pairwise conditional random field (FPCRF) as a graph model to preserve sharp boundaries and fine-grained segmentation. Experiments are conducted on four different data sets: 1) Planetscope satellite imagery of the cities of Munich, Paris, Rome, and Zurich; 2) ISPRS Benchmark data from the city of Potsdam; 3) Dstl Kaggle data set; and 4) Inria Aerial Image Labeling data of Austin, Chicago, Kitsap County, Western Tyrol, and Vienna. It is found that the proposed end-to-end building footprint generation framework with the FPCRF as the graph model can further improve the accuracy of building footprint generation by using only CNN, which is the current state of the art.**

***Index Terms*— Building footprint, conditional random field (CRF), convolution neural network (CNN), graph model, semantic segmentation.**

I. Introduction

BUILDING footprint generation is an active field of research with the domain of remote sensing (RS). The established building footprint maps are useful to understand urban dynamics in many important applications, and also facilitate the assessment of the extent of damages after natural disasters such as earthquakes. OpenStreetMap (OSM) can provide manually annotated building footprint information for some urban areas; however, it is not always available in many parts of the world. Therefore, high-resolution RS imagery, which covers global areas and contains huge potential for meaningful ground information extraction, is a reliable source of data for building footprint generation. However, automatic building footprint generation from high-resolution RS imagery is still difficult because of variations in the appearance of buildings, complicated background interference, shooting angle, shadows, and illumination conditions. Moreover, buildings and the other impervious objects in urban areas have similar spectral and spatial characteristics.

Early studies of automatic building footprint generation from high-resolution RS imagery rely on regular shape and line segments of buildings to recognize buildings. Line segments of the building are first detected and extracted by edge drawing lines (EDLines) [1], and then hierarchically grouped into candidate rectangular buildings by a graph search-based perceptual grouping approach in [2]. Some studies also propose some building indices to identify the presence of a building. The morphological building index (MBI) [3], which takes the characteristics of buildings into consideration by integrating multiscale and multidirectional morphological operators, can be implemented to extract buildings automatically. The most widely used approaches are classification-based approaches, which make use of spectral information, structural information, and context information. The pixel shape index (PSI) [4], a shape feature measuring the gray similarity distance in each direction, is integrated with spectral features to extract buildings by using a support vector machine. However, the main problem with these algorithms is that multiple features need to be engineered for the proper classifier, which may consume too much computational resources and thus preclude large scale applications.

Based on learning data representations, deep learning is the state-of-the-art method for many big data analysis applications [5]–[7]. Deep learning architectures such as convolutional neural networks (CNNs), which is an artificial neural network based on multiple processing layers, have been extensively employed in many computer vision tasks. A major advantage of CNN is its independence from prior knowledge and

Manuscript received June 18, 2019; revised September 25, 2019; accepted February 3, 2020. This work was supported in part by the European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation Programme under Grant ERC-2016-StG-714087 (acronym: So2Sat, www.so2sat.eu), in part by the Helmholtz Association under the framework of the Young Investigators Group "SiPEO" under Grant VH-NG-1018 (www.sipeo.bgu.tum.de), and in part by the Helmholtz Excellent Professorship "Data Science in Earth Observation—Big Data Fusion for Urban Research." *(Corresponding author: Xiao Xiang Zhu.)*

Qingyu Li is with Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), 82234 Weßling, Germany, and also with Signal Processing in Earth Observation, Technische Universität München (TUM), 80333 Munich, Germany (e-mail: qingyu.li@dlr.de).

Yilei Shi is with the Chair of Remote Sensing Technology, Technische Universität München (TUM), 80333 Munich, Germany (e-mail: yilei.shi@tum.de).

Xin Huang is with the Department of Remote Sensing, School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China (e-mail: xhuang@whu.edu.cn).

Xiao Xiang Zhu is with Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), 82234 Weßling, Germany, and Signal Processing in Earth Observation, Technische Universität München (TUM), 80333 Munich, Germany (e-mail: xiaoxiang.zhu@dlr.de).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TGRS.2020.2973720

hand-crafted features, which has supported its more powerful generalization capability. CNN is superior to other approaches with respect to accuracy and efficiency. In particular, many CNN models have been proposed and applied in semantic segmentation with quite promising results, such as the fully convolutional network (FCN) [8], U-Net [9] SegNet [10], ResNet [11], ENet [12], DenseNet [13], PSPNet [14], and DeepLabv3+ [15]. Recently, the generative adversarial network (GAN) [16] has shown the potential in solving such problems.

In fact, the task of building footprint generation belongs to the branch of semantic segmentation in computer vision. In the RS community, recent research has also made an effort to improve building footprint generation through the application of the aforementioned CNN models. In order to perform building segmentation, a multiconstraint FCN (MC-FCN) model is proposed in [17], which consists of an FCN architecture and multiconstraints. In [18], a modified and extended architecture of both ResNet and U-Net, named Res-U-Net, is proposed to improve the accuracy of building segmentation results from RS imagery. A comparatively simple and memory-efficient model, SegNet, is used for a multitask (a shared representation for boundary and segmentation prediction) learning for building footprint generation in [19]. A conditional GAN called cwGAN-gp [20], whose loss function is derived from the Wasserstein distance and an added gradient penalty term, is proposed to improve the building footprint generation results.

However, there are usually nonsharp boundaries and visually degraded results in CNN-based semantic segmentation tasks, which results from the inherent invariant to spatial transformations of CNN architectures. In this case, the common approach to improving the accuracy of pixel-level segmentation is to adopt a graph model such as conditional random field (CRF) as a postprocessing step. Fully connected CRF [21] is applied to accurately localize segment boundaries and assign the most probable label to each pixel after the training based on FCN in [22]. In this case, the CRF inference is used as a postprocessing step, which is not integrated with the training of the CNN. In this article, we propose an accurate and reliable building footprint generation framework, which makes three contributions.

1) Since each existing CNN model also has its own limitations, achieving more accurate segmentation results is still critical for automatic building footprint generation. The use of a graph model enables the combination of low-level image information such as the interactions between pixels, which is especially important for capturing fine local details. Therefore, in order to achieve more accurate segmentation results, we propose to combine CNN and a graph model in an end-to-end framework for building footprint generation, which has not been adequately addressed in the current literature.
2) In addition, it should be noted that, in this research, we propose a graph model called feature pairwise CRF (FPCRF) to be exploited in the building footprint generation framework. Specifically, we design a pairwise potential term with localized constraints in CRF. This term combines feature kernels extracted from CNN, which allows more complete feature learning than other traditional graph models. Moreover, the localized processing facilitates the efficient message passing operation.
3) Recently, there has been some development of deep learning methods in the computer vision community that seek to enhance the results of semantic segmentation; this development offers the RS community an opportunity to investigate the application of building footprint generation using deep learning methods. However, there is still a lack of a comprehensive investigation into the state-of-the-art CNN models in the tasks of automated building footprint generation from RS imagery. With the aim of better understanding the usability and generalization ability of the state-of-the-art approaches, we compare and analyze the performances and characteristics of different CNN models for building footprint generation.

This article is organized as follows. In Section II, a brief review of related works is presented. Then, the proposed framework is introduced in Section III, followed by experiments in Section IV and results in Section V. Next, a discussion is provided in Section VI, leading to conclusions in Section VII.

II. Related Work

A. Semantic Segmentation

Deep learning methods have been commonly used in the field of computer vision, from coarse-to-fine inference. Classification is the coarse inference, which makes a prediction for a whole input. Semantic segmentation is the fine-grained inference, which assigns a label to each pixel. CNN can learn an enhanced feature representation end to end for solving the semantic segmentation problems. FCN or encoder–decoder-based architectures have been successfully implemented to produce spatially explicit label maps efficiently.

FCN is a forerunner of semantic segmentation, which transforms popular classification models to fully convolutional ones, and replaces the fully connected layers with transposed convolutions to solve pixel labeling problems. Apart from the FCN architecture, the performance of other variants such as encoder–decoder based architectures is also remarkable. The spatial dimension has been gradually reduced with pooling layers in the encoder, while the local detail and spatial dimension are recovered in the decoder. Moreover, there are skip connections from encoder to decoder in U-Net, which makes the compensation from low-level details to high-level semantic information. In SegNet, the max-pooling indices are reused in the decoding process, which results in a substantial reduction in the number of parameters. ResNet-DUC [23] is similar to U-Net, but uses a ResNet block instead of a normal block. In the ResNet block, the layers are reformulated as learning residual functions of the input layer, which is easier to optimize [11]. ENet consists of a large encoder and a small decoder, where the large encoder can be operated on smaller resolution data and contributes to efficient information processing. The potential of GAN is also investigated in the semantic segmentation domain. GAN comprises two networks: a discriminator and a generator. The discriminator learns the boundary between classes, while the generator learns the distribution of individual classes. The two networks play a two-player min-max game to optimize both of their objective functions. The PSPNet is a typical example of the multiscale processing network, which first generates a feature map from a feature extraction network (ResNet, DenseNet, and so on), and then utilizes a pyramid pooling module to combine multiscale feature maps. DeepLab [24] is a state-of-the-art semantic segmentation model, which now already have four versions with different improvements over time: DeepLab V1, DeepLab V2, DeepLab V3, and DeepLab V3+. Both DeepLab V1 and DeepLab V2 use CRF as a postprocessing step, where the prediction could be refined both qualitatively and quantitatively. DeepLabv3 improves over previous DeepLab versions without CRF postprocessing. This is due to the fact that a better way is designed to encode multiscale context in its network architectures. DeepLabv3 is a network that does multiscale processing, and by using altrous convolution it can achieve satisfactory results without increasing the number of parameters. The DeepLabv3+ model is a quick extension of DeepLabv3 that proposes to add an intermediate decoder module to the DeepLabv3, could recover object boundaries better. Currently, FC-DenseNet has shown superior results on terrestrial scene interpretation tasks. FC-DenseNet extends the DenseNet architecture to FCNs in pixel-level labeling tasks. In the DenseNet block, all preceding features are taken as input, and then its output features are transferred to all subsequent layers [13]. Through this feature reuse, the potential of the network can be utilized to improve the ease of training and parameter efficiency.

The development of CNN has rapidly improved the performance of semantic segmentation algorithms, which has elicited an increasing interest in the RS domain. Many research works have transferred these common CNN models and adapted them for RS imagery, which has already achieved good performance. An efficient multiscale approach is implemented for CNN in [25], leveraging both a large spatial context and high-resolution data to allow better semantic segmentation results. In [26], a multitask learning method for semantic segmentation is proposed that learns the semantic class likelihoods and semantic boundaries across classes by CNN simultaneously. The spatial relation and channel relation modules are combined with CNN in [27], which has achieved competitive semantic segmentation results.

B. CNN for Building Footprint Generation

In RS domain, semantic segmentation is often referred to in numerous applications, such as change detection [28], land-cover classification [29], road extraction [30], and building footprint generation [31], and so on. Since the building is an important object among various terrestrial targets in RS imagery, the task of building footprint generation has been heavily studied in the RS community.

One of the CNN models commonly used for building footprint generation is FCN, which has showed superiority in accuracy as well as computational time. When applied with RS data, FCN is usually adapted. In [32], a multiscale neuron module is designed in FCN, which is able to provide fine-grained building footprint maps. A multilayer perceptron (MLP) network is derived on top of the base FCN in [33], which extracts intermediate features from the base FCN to provide finer results. In [34], three parallel FCNs are first implemented to combine different data sources, and then merged at a late stage to automatically generate a more accurate building footprint map. A variant of FCN, which introduces an additional higher resolution skip connection, is adopted in [21] in order to preserve consistently improved results. The proposed method in [35] employs a similar strategy by adding skip connections, which can minimize information loss from downsampling.

Apart from FCN, other encoder–decoder-based architectures such as SegNet are also preferred in building footprint generation, because its memory requirements are significantly lower than FCNs. In this regard, larger scale problems can be solved in parallel more efficiently at the inference stage. In [36], the building footprints across the entire continental United States are generated by SegNet with better fulfillment of the quality and computational time requirements. However, SegNet has a low edge accuracy, since it only uses a part of the layers to generate predicted output. Another encoder–decoder-based architecture, U-Net, which combines both the low and high layers, is widely exploited to generate building footprint maps with their edges preserved. A Siamese U-Net [37], where original images and their downsampled counterparts are taken into the network separately, is proposed to improve the final results, especially for large buildings. Currently, some newly proposed networks, such as FC-DenseNet and GAN, have also demonstrated promising performances in building footprint generation. In [38], a generator using FC-DenseNet and an adversarial discriminator are jointly trained for the building footprint generation from RS imagery.

C. Graph Model

Exploiting CNN for semantic segmentation tasks is still a significant challenge. The convolutional layer of CNN is a weights sharing architecture. Hence, shift invariant and spatial invariant characteristics limit spatial accuracy for segmentation tasks [39]. The convolution filters with large receptive fields and max-pooling layers in CNN also lead to coarse segmentation output, such as a nonsharp boundary and blob-like shapes [40]. Moreover, CNN fails to refine local details without taking the interactions between pixels into consideration. Graph models enable modeling of interactions between pixels, which can integrate more elaborate terms to preserve the sharp boundary. Therefore, graph models can be utilized to enhance the semantic segmentation results from CNN, which has the ability to capture fine-grained details.

A graph model is a probabilistic model that encodes a distribution based on a graph-based representation. In a graph model, conditional dependencies are expressed between random variables. There are two categories of graphical representations of distributions, Bayesian networks and Markov random field (MRF), which are distinguished by their encoded

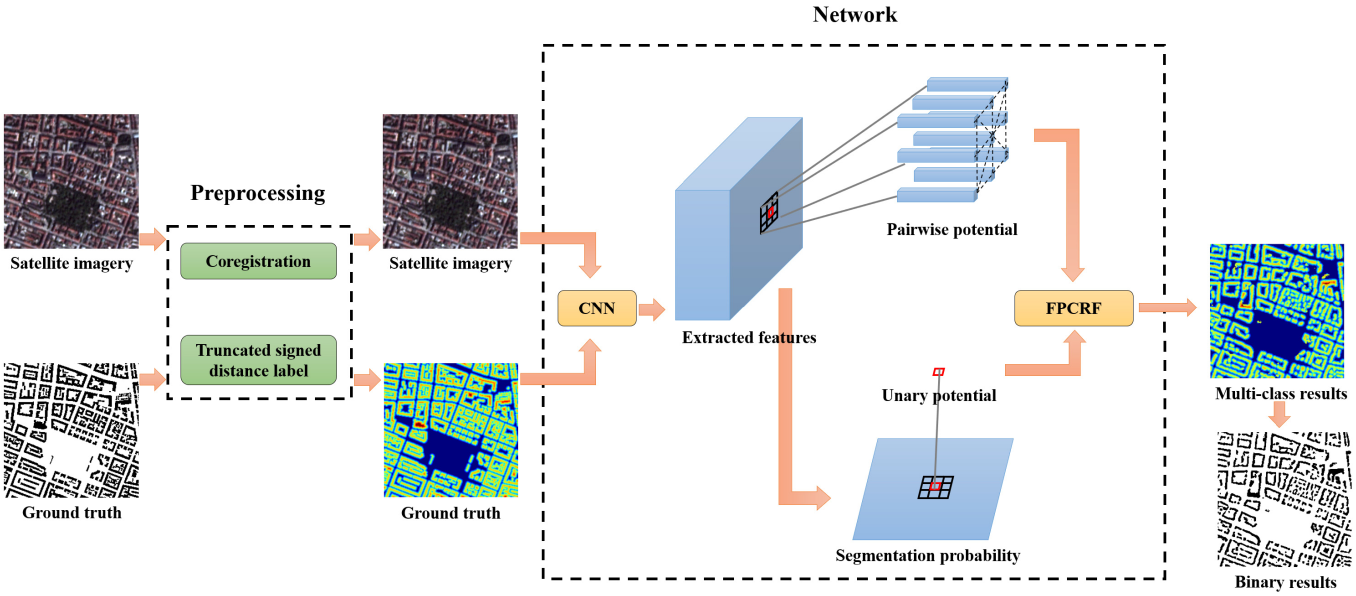


Fig. 1. Flowchart of the proposed approach.

set of independence and induced factorization of the distribution. In the Bayesian networks, the network structure of the model is based on a directed acyclic graph, where the joint distribution is represented as a product of conditional distributions. MRF is an undirected graph, which is described by random variables with a Markov property. In the Markov property, only the present state contributes to the conditional probability distribution of future states of the process. CRF is a notable variant of MRF, in which each random variable is conditioned upon some global observations. FullCRF [40] is a notable example of CRF, which is regarded as a recurrent neural network (RNN) that forms a part of a deep network for end-to-end training. However, FullCRF is based on a complex data structure and does not allow efficient GPU computation. Recently, there are some researches focused on the improvement of CRF. The work in [41] proposes to use bilateral convolution layers (BCLs) built inside CNN architectures for efficient CRF inference, where the receptive field of filters could change. ConvCRF [42] is a recently proposed CRF algorithm that adds a conditional independence assumption to supplement FullCRF, and such an adjustment reduces the complexity of the pairwise potential. A recent example is pixel-adaptive convolution (PAC)-CRF [43], propose a PAC for efficient inference of CRF to alleviate the computation, whose filter weights depends on a spatially varying kernel utilizing local pixel features.

Some researchers have tried to implement both CNN models and graph models for building footprint generation. The results have shown that combining graph models and CNN models can lead to better results, especially along the boundaries of buildings. In [44], MRF is integrated as a postprocessing stage after the training of CNN, which has ameliorated the final building footprint generation map. The CRF is exploited in [45] and [46] to smooth the final pixel labeling results from CNN, which can respect the edges present in the imagery. However, the graph models are exploited only as postprocessing steps in these studies. In [47], the FullCRF is plugged in at the end of the FCN for end-to-end training, which has preserved sharp boundaries, but requires longer training time and greater efforts to find optimal parameters.

III. Methodology

In this section, the proposed building footprint generation framework is first described. Then, we introduce the proposed FPCRF, which has a designed pairwise potential term for complete feature learning and efficient computation. The experiment design for detailed investigation of FPCRF parameters is provided in Section IV-C.

A. Proposed Building Footprint Generation Framework

The building footprint generation in this article is actually a semantic segmentation task in the computer vision field. Recently, CNN has achieved great success in semantic segmentation tasks, as it is able to learn a strong feature representation instead of hand-crafted features. However, there are also some problems with CNN models, such as limited spatial accuracy, nonsharp boundaries, and so on. Parallel with CNN models, graph models, which enable interactions between pixels to be modeled, have also been shown to be effective methods to improve semantic segmentation results. For example, sharp boundaries and fine-grained details can be preserved by graph models. In order to harness the strengths of both models, we propose to integrate CNN and a graph model in the framework of building footprint generation. However, it should be noted that although the results could be improved by simply including graph models after learning from CNN, an end-to-end training scheme that fully integrates the graph models with CNN is preferred in our research. The end-to-end approach can provide more replicable and stable building footprint maps, especially for large scale applications. In this regard, we propose to utilize FPCRF as the graph model in the end-to-end framework, as it is superior to other graph models in terms of computation efficiency and completeness in feature learning.

In our proposed approach, CNN and FPCRF are integrated in an end-to-end framework, where the gradients are propagated through the entire pipeline. In this case, CNN and FPCRF can coadapt and therefore produce the optimal output. Fig. 1 shows the overall architecture of the proposed approach. It has two major components: CNN and FPCRF. The output of the CNN consists of two parts. One output is the segmentation probability obtained from the last softmax layer of CNN, which predicts labels for pixels. This segmentation probability obtained from CNN is utilized as the unary potential [40]. The other output is extracted features from CNN, which encodes each pixel as a fixed-length vector representation (i.e., embedding). This feature embedding is used for pairwise potential calculation, which encourages assigning similar labels to pixels with similar properties. The FPCRF component is utilized as the graph model to complement the results obtained from CNN. FPCRF takes the patch of feature embedding and unary potential as input and models their spatial correlations. The final output from FPCRF is the marginal distribution of each pixel, which represents the different class label when the patch embedding is given.

B. Data Preprocessing

Since the ground truth of the building footprint is generated using OSM with different data sources from satellite images, the inconsistencies between data sets need to be resolved by the preprocessing steps, including coregistration and truncated signed distance labels.

1) Coregistration: One inconsistency is the misalignment between OSM building footprints and satellite imagery, which is caused by different projections and accuracy levels from data sources. This misalignment leads to inaccurate training samples, which need to be corrected. In this regard, we make an assumption that after translation the building footprint from OSM will be aligned with satellite imagery content within a local neighborhood [48]. Between the building footprint and gradient magnitude of satellite imagery, a cross correlation is calculated, where the maximum of the cross correlation corresponds to the estimated alignment location. In this regard, the offsets in both row and column direction can be derived, which are corresponding to the translation coefficients. An example of satellite imagery overlaid with the OSM building footprint is presented in Fig. 2(a). There are noticeable misalignments between the building footprint and the satellite imagery. The local neighborhood size is selected as 7. Fig. 2(b) illustrates the coregistration result.

2) Truncated Signed Distance Label: In order to incorporate both semantic information and geometric properties of the buildings during training [19], the distances from pixels to the boundaries of buildings are extracted as output representations. In our experiment, the signed distance from a pixel to its closest point on the boundary is calculated with positive values indicating building interior and negative indicating building exterior. Then we truncate the distance at a given threshold to only incorporate the pixels closest to the border [19]. Finally, the distance values are categorized into a number of class labels [19]. The advantage of this truncated signed distance mask is that the location of the boundary and implicit geometric properties of each pixel can be captured. In addition, different buildings can be distinguished based on their between distance and labels.

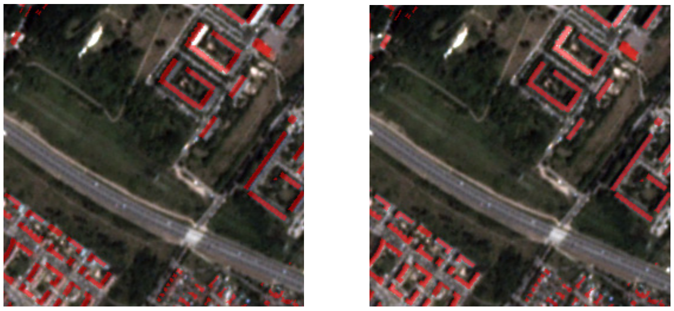
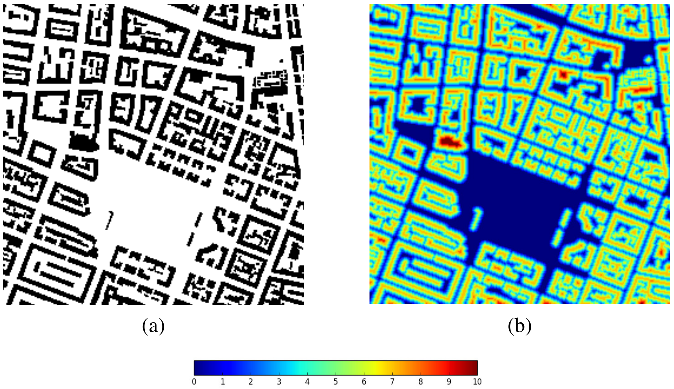

(a) (b)

Fig. 2. (a) Before coregistration. (b) After coregistration.



(a) (b)

(c)

Fig. 3. (a) Binary label. (b) Truncated signed distance label. (c) Colorbar for the class label.

Given that J is the set of pixels on the object boundary and L_l is the set of pixels with class label l, the truncated distance $D(i)$ for every pixel i is calculated as

$$D(i) = \delta_p \min(\min_{\forall j \in J} d_{eu}(i, j), T)$$
$$\delta_p = \begin{cases} 1 & \text{if } p \in L_{\text{building}} \\ -1 & \text{if } p \in L_{\text{non-building}} \end{cases} \tag{1}$$

with $d_{\text{eu}}(i, j)$ being the Euclidean distance between pixels i and j and T is the truncated threshold. The sign function δ_p is used to weigh the pixel distances to represent whether the pixels are inside or outside the building masks. To facilitate training, the continuous distance values are then uniformly quantized.

In this article, we use 11 classes with the labels $L = \{0, 1, 2, \ldots, 10\}$. Class 5 represents the building boundary and when the class label is greater than 5, this pixel belongs to the building. Similarly, the non-building pixel has a class label smaller than 5. Fig. 3 illustrates the binary label and truncated signed-distance label of a building footprint, which are used in the network training. Based on the raw output (multiclass) from a trained network, we simply select a threshold to classify the class labels as a final binary building footprint result: a pixel is considered as building if $l >= 5$; otherwise it is considered as non-building when $l < 5$.

C. FPCRF

An image can be regarded as a graph, where every pixel is a vertex, and there are edges between each pair of pixels. FPCRF provides a probabilistic model for an image that is both local and modular.

In FPCRF, the joint probability for the random variables is implied as functions over cliques

$$P(X = x \mid I) = \frac{1}{Z(I)} \exp\left(- \sum_{c \in C_G} \phi_c(X_c \mid I) \right) \tag{2}$$

where X is a field defined over a set of variables $\{X_1, \ldots, X_N\}$ with N being the number of pixels, where the domain of each variable is a set of labels $L = \{l_1, l_2, \ldots, l_c\}$ with c being the number of classes. The expression $G = (V, \varepsilon)$ denotes a graph where $V = \{X_1, X_2, \ldots, X_N\}$. The term $I = \{I_1, I_2, \ldots, I_N\}$ is a global observation (image). The term ϕ_c is a potential induced by the clique C_G (each two vertices are linked) in the graph G. The function $Z(I) = \sum \exp(-\sum_{c \in C_G} \phi_c(X_c \mid I))$ is a partition function. The energy of a labeling is $E(x \mid I) = \sum_{c \in C_G} \phi_c(X_c \mid I)$. Gibbs distribution is a probability distribution that measures a system with a certain state as a function of that state's energy. CRF explicitly gives a representation of the conditional independence between nodes of a graph. CRF and Gibbs distribution are proved to be equivalent with regard to the same graph from the Hammersley–Clifford theorem [49], which indicates that when the Gibbs distribution is given, the conditional independence specified by the corresponding CRF will be satisfied by all of the Gibbs joint probability distributions. Therefore, the Gibbs distribution characterized by FPCRF can thus be expressed as

$$P(X = x \mid I) = \frac{1}{Z(I)} \exp(-E(x \mid I)). \tag{3}$$

In order to take: 1) the interactions between pixels and 2) the approximation inference into consideration during learning, the Gibbs energy is expressed as

$$E(x \mid I) = \sum_{i \le N} \psi_u(x_i \mid I) + \sum_{i \ne j \le N} \psi_p(x_i, x_j \mid I) \tag{4}$$

and i and j range from 1 to N. The term $\psi_u(x_i \mid I)$ is the unary potential, which is independent for each pixel. Unary potential is a distribution over the label assignment x_i from the classifier. The term $\psi_p(x_i, x_j \mid I)$ is a pairwise potential function that is determined based on the compatibility among pairs of pixels. This pairwise potential term can overcome the drawbacks of the noisy and inconsistent labeling from the unary potential alone.

In FPCRF, the pairwise potential $\psi_p(x_i, x_j \mid I)$ is defined by the expression as follows:

$$\psi_p(x_i, x_j \mid I) = \mu(x_i, x_j) \underbrace{\sum_{m=1}^{M} w^{(m)} k^{(m)}(f_i, f_j)}_{k(f_i, f_j)} \tag{5}$$

where $w^{(m)}$ are learnable parameters, and M is the number of kernels, which is determined by the selected kernels. The terms f_i and f_j are feature vectors for pixels i and j and may depend on the input image I. The function $\mu(x_i, x_j)$ is the compatibility transformation and captures the compatibility between labels x_i and x_j.

However, FullCRF and ConvCRF only use shallow features—the color and position of the pixel for kernels in pairwise potential term, which have not fully harnessed the complete features extracted from CNN. In this regard, we propose FPCRF as a graph model to be exploited in the building footprint generation framework.

Inspired by the fact that ConvCRF is based on localized processing, we design a pairwise potential term with localized constraints in FPCRF that allows complete feature learning. The kernel utilized for pairwise potential in FPCRF is a Gaussian kernel, which is defined by the feature vectors $f_1, \ldots, f_B$, where B is the number of feature vector types. The kernel $k^{(m)}$ is defined as

$$k^{(m)}(f_i, f_j) = \exp\left(- \sum_{b=1}^{B} \frac{|f_{b,i} - f_{b,j}|^2}{2\theta_b^2} \right) \tag{6}$$

where θ_b is a learnable parameter.

The labeling of the random field is derived by the maximum *a posteriori* (MAP) method

$$x^* = \operatorname{argmax}_{x \in L^N} P(X = x \mid I). \tag{7}$$

The most probable label x can be yielded by the minimization of the Gibbs energy in FPCRF. However, the exact minimization is intractable. In this regard, the mean-field inference is utilized for the approximation of FPCRF distribution. A distribution $Q(X)$ that tries to minimize the KL-divergence $D(Q||P)$ from exact distribution $P(X)$ is computed by the mean-field approximation

$$D(Q||P) = \sum_{x} Q(x) \log\left(\frac{P(x)}{Q(x)} \right) \tag{8}$$

where the approximated distribution $Q(X)$ can be represented as a product of independent marginal distributions

$$Q(X) = \prod_{i} Q_i(X_i). \tag{9}$$

The combined message passing result Q of all kernels is expressed as

$$\begin{aligned} &Q_i(x_i = l) \\ &\quad = \frac{1}{Z_i} \exp\Bigg\{ - \psi_u(x_i \mid I) - \sum_{l' \in L} \mu(l, l') \\ &\qquad\qquad \times \sum_{m=1}^{M} w^{(m)} \sum_{d_{ma}(i,j) < r} k^{(m)}(f_i, f_j) Q_j(l') \Bigg\}. \end{aligned} \tag{10}$$

TABLE I

STEPS OF THE MEAN-FIELD ALGORITHMS IN FPCRF

Mean field approximation in FPCRF	
1. Initialize Q	$Q_i(x_i) = \frac{1}{Z_i} exp\{-\psi_u(x_i \mid I)\}$
2. while not converged	
3. $\tilde{Q}_i^{(m)}(l) \leftarrow \sum_{i \neq j} k^{(m)}(f_i, f_j) Q_j(l)$ for all m	Message passing from all X_j to all X_i
4. $\check{Q}_i(x_i) \leftarrow \sum_{m=1} w^{(m)} \tilde{Q}_i^{(m)}(l)$	Weighting filtering outputs
5. $\hat{Q}_i(x_i) \leftarrow \sum_{l' \in L} \mu(l, l') \check{Q}_i(l)$	Compatibility transformation
6. $Q_i(x_i) \leftarrow -\psi_u(x_i \mid I) - \hat{Q}_i(x_i)$	Adding unary potentials
7. $Q_i(x_i) \leftarrow \frac{1}{Z_i} exp(-Q_i(x_i))$	Normalization
8. end while	

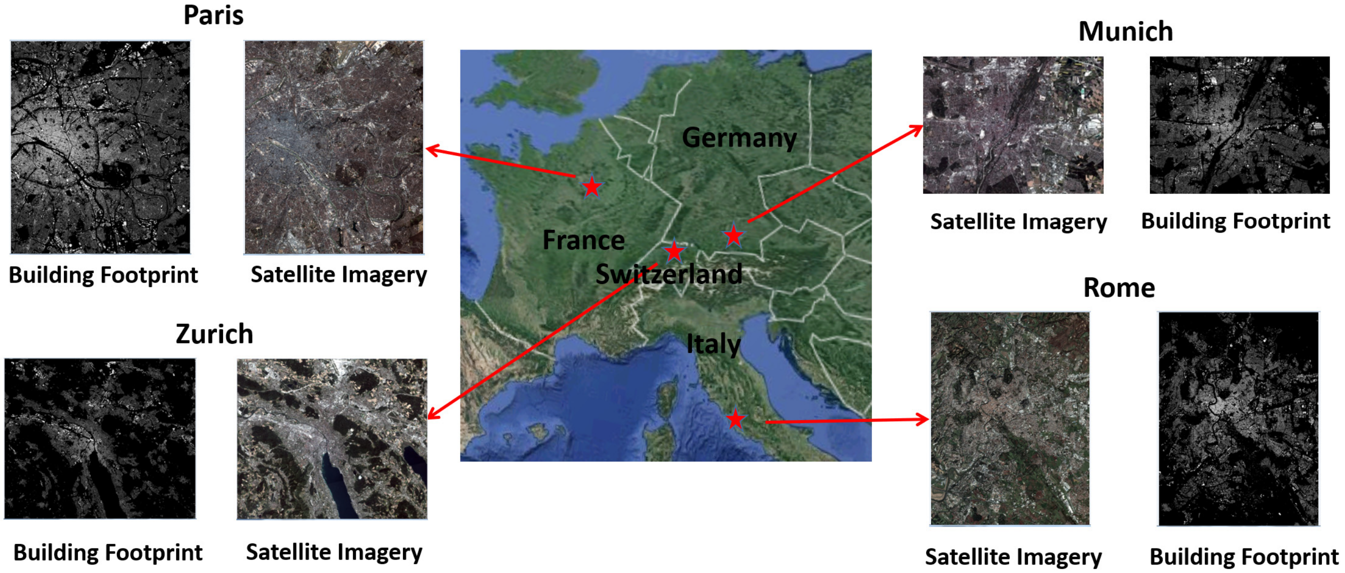


Fig. 4. True color Planetscope satellite images and building footprint of Munich, Rome, Paris, and Zurich.

The steps of the mean-field algorithms are presented in Table I.

The steps of the mean-field inference algorithm of FPCRF are reformulated as a network layer, where the error differentials in each layer with respect to its inputs are sent to previous layers by back propagation during training [40]. FPCRF exploits a 1 × 1 filter to assign the different penalties for all different pairs of labels.

To implement the efficient computation of the convolution, the input is first tiled into the specific shapes, which are related to the filter size r. An efficient message passing operation in FPCRF can be implemented analogously to 2-D-convolution [42]. Then, the message passing step is reformulated to be a convolution with a truncated Gaussian kernel.

IV. EXPERIMENTS

A. Study Area and Data Set

In this article, the study sites cover four cities (see Fig. 4): 1) Munich, Germany; 2) Rome, Italy; 3) Paris, France; and 4) Zurich, Switzerland. We use Planetscope satellite imagery [50] with three bands—red, green, blue (RGB)—and 3-m spatial resolution to validate our proposed method. The imagery is processed using a 256 × 256 sliding window. The corresponding building footprint (stored as polygon shape files) is downloaded from OSM, where the detailed building footprints around these four cities are publicly released. Some patches are mismatched, which result from the time difference between OSM building footprints and satellite imagery. For example, a building might appear in the OSM building footprint, while it is missing in the corresponding satellite imagery, or vice versa. To limit such patches, we have manually selected 3000 pairs of proper patches. The selected pairs are then separated into two parts, where 80% of the sample patches are used for training the network and 20% are used for model validation.

TABLE II

ACCURACY OF DIFFERENT FEATURE EXTRACTORS COMBINED WITH FPCRF

Methods	Overall accuracy	F1 score	IOU
FC-DenseNet + FPCRF	**0.9297**	**0.6698**	**0.5046**
FCN-8s + FPCRF	0.9248	0.6340	0.4642
U-Net+ FPCRF	0.8927	0.6278	0.4575

B. Experiment Setup

In this article, all networks were investigated within a Pytorch framework on an NVIDIA Titan X GPU with 12 GB of memory [51]. For all networks, a stochastic gradient descent (SGD) optimizer with a learning rate of 0.0001 was utilized

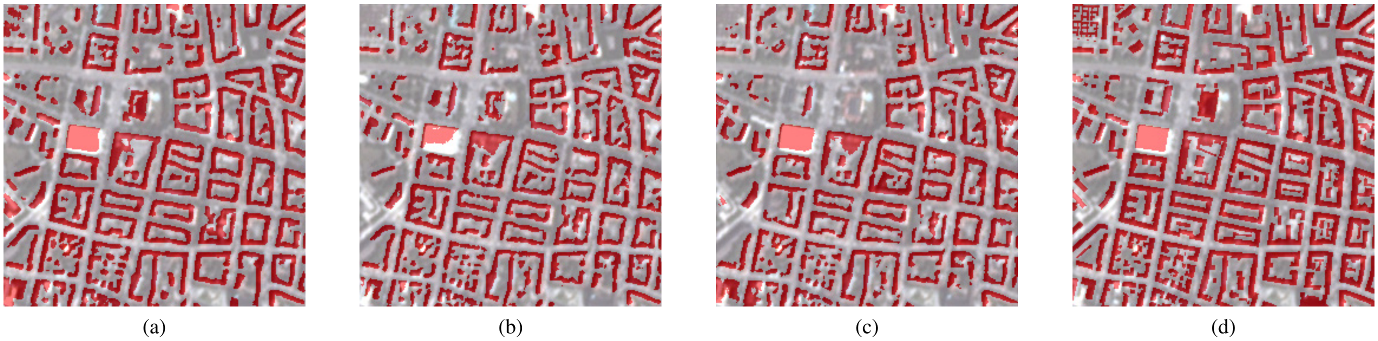


Fig. 5. Predicted results (in red) obtained from different feature extractors. (a) FC-DenseNet. (b) FCN-8s. (c) U-Net combined with FC-DenseNet. (d) Ground truth.

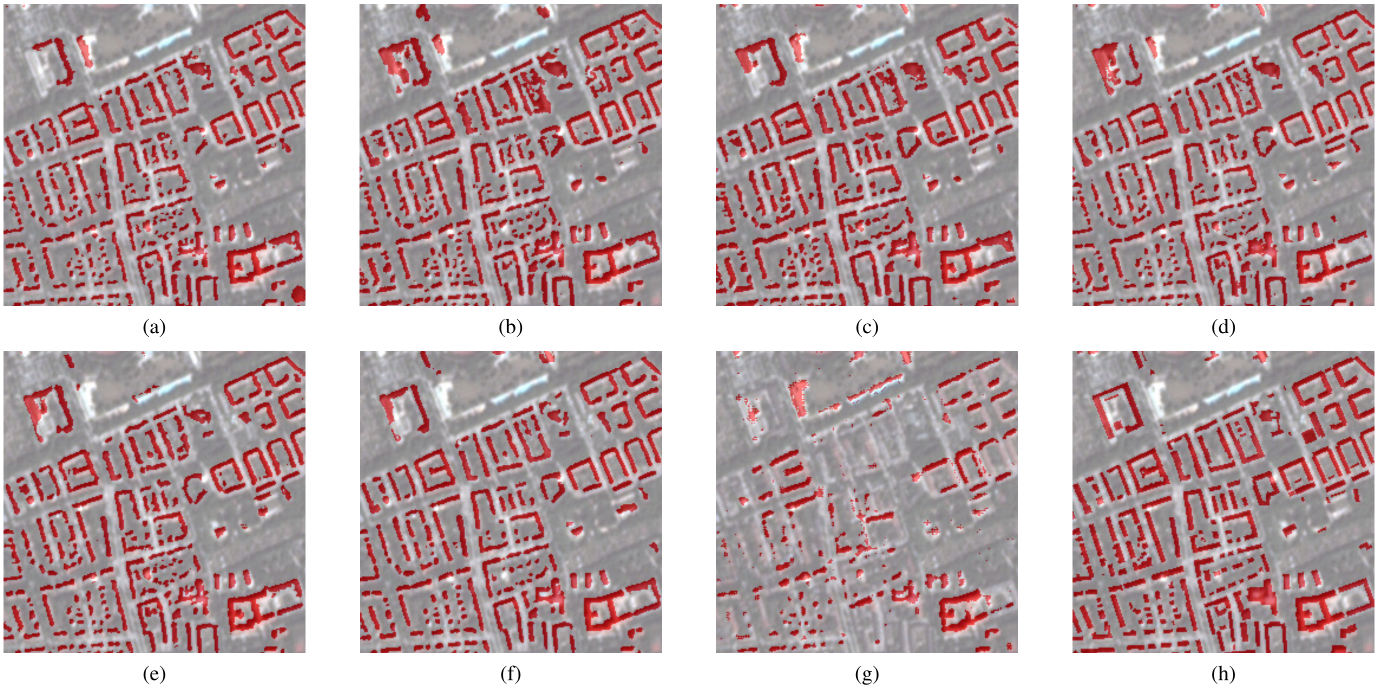


Fig. 6. Predicted results (in red) obtained from FC-DenseNet combined with FPCRF from different kernels. (a) $a + s$. (b) $a + s +$ fd. (c) $s +$ fd. (d) $a +$ fd. (e) fs. (f) fd. (g) fc. (h) Ground truth. (a: appearance kernel, s: smooth kernel, fd: feature difference kernel, fs: feature spatial kernel, and fc: feature cosine kernel.)

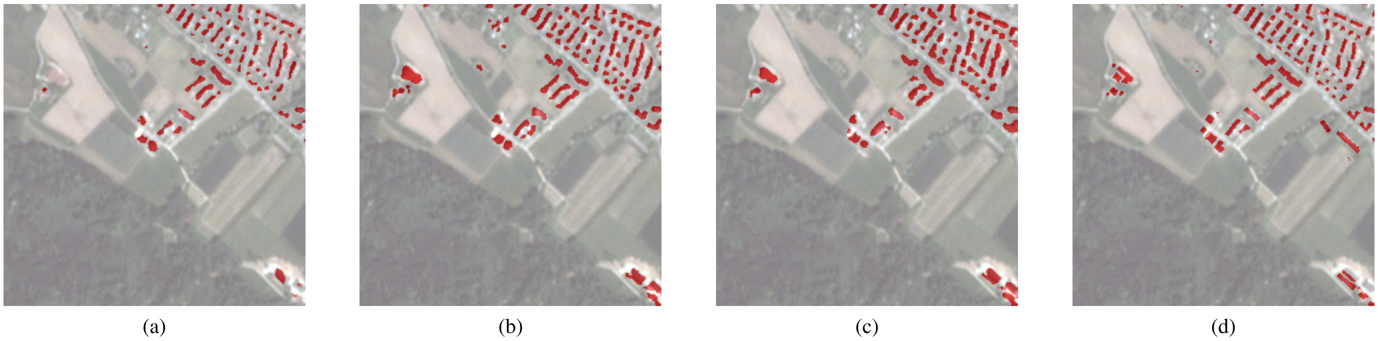


Fig. 7. Predicted results (in red) obtained from FC-DenseNet combined with FPCRF within different filter size. (a) FC-DenseNet + FPCRF ($r = 5$). (b) FC-DenseNet + FPCRF ($r = 7$). (c) FC-DenseNet + FPCRF ($r = 9$). (d) Ground truth.

and negative log likelihood loss (NLLLoss) was taken as loss function. The batch size of all networks was 4.

In our proposed end-to-end approach, CNN and FPCRF are two vital parts in the framework, where the CNN component acts as a feature extractor, and the FPCRF models their pixel correlations by using pairwise potential. Hence, we first investigate which CNN model has better feature extraction capability. Then, the feature kernels that are taken in pairwise

TABLE III

ACCURACY OF FC-DENSENET COMBINED WITH FPCRF FROM DIFFERENT KERNELS. (A: APPEARANCE KERNEL, S: SMOOTH KERNEL, FD: FEATURE DIFFERENCE KERNEL, FS: FEATURE SPATIAL KERNEL, AND FC: FEATURE COSINE KERNEL)

Methods	Overall accuracy	F1 score	IOU
FC-DenseNet + FPCRF (a+s)	0.9075	0.6653	0.4986
FC-DenseNet + FPCRF (a+s+fd)	0.9166	0.6682	0.5018
FC-DenseNet + FPCRF (s+fd)	0.9013	0.6660	0.4991
FC-DenseNet + FPCRF (a+fd)	0.9212	0.6685	0.5013
FC-DenseNet + FPCRF (fs)	0.9275	0.6673	0.5006
FC-DenseNet + FPCRF (fd)	**0.9297**	**0.6698**	**0.5046**
FC-DenseNet + FPCRF (fc)	0.7888	0.4521	0.2921

TABLE IV

ACCURACY OF FC-DENSENET COMBINED WITH FPCRF WITHIN DIFFERENT FILTER SIZE. *r* IS FILTER SIZE

Methods	Overall accuracy	F1 score	IOU
FC-DenseNet + FPCRF (r=5)	0.9121	0.6665	0.4985
FC-DenseNet + FPCRF (r=7)	**0.9297**	**0.6698**	**0.5046**
FC-DenseNet + FPCRF (r=9)	0.9142	0.6670	0.4993

Fig. 8. Aerial imagery in ISPRS data set (spatial resolution: 5 cm).

potential calculation of FPCRF are also carefully studied to find the optimal feature embedding. Moreover, the sensitivity of the filter size r, being the only hyperparameter of FPCRF, is analyzed. Additionally, to prove the superiority of our proposed framework, we train the following networks for comparison.

1) FCN-8s is based on VGG16 as the encoder and an upsampling layer and convolutional layer as the decoder.
2) ResNet-DUC, which has [3, 4, 6, 3, 3, 6, 4, 3] convolutional layers in each ResNet block.
3) SegNet, which attaches a reversed VGG16 as a decoder to the encoder.
4) U-Net, which has a depth of five with a feature channel in each depth [64, 128, 256, 512, 1024].
5) ENet, which consists of five stages, where the first three stages act as the encoder, while the last two stages belong to the decoder.
6) cwGAN-gp which also has five depth U-Net in the generator.

Fig. 9. WorldView 3 imagery in Dstl data set (spatial resolution: 1.24 m).

Fig. 10. Aerial imagery in Inria data set (spatial resolution: 30 cm).

7) FC-DenseNet, with each dense block having [5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5] convolutional layers.
8) PSPNet starts off with a standard feature extraction network (ResNet101).
9) DeepLabv3+ utilizes the Xception model [52] as the feature extractor.

V. RESULTS

The three metrics in the following experiments selected to evaluate the results are overall accuracy, F1 score, and intersection over union (IoU), which are used widely to evaluate building footprint generation results:

$$\text{Overall accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \tag{11}$$

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{12}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{13}$$

$$\text{F1 score} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}} \tag{14}$$

$$\text{IoU} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \tag{15}$$

where TP is the number of building pixels correctly detected, and FN denotes the missed building pixels. FP and TN are the

TABLE V

NUMBERS OF TRAINING AND VALIDATION PATCHES OF THREE ADDITIONAL DATA SETS

Dataset	Number of training patches	Number of validation patches
ISPRS benchmark data (spatial resolution: 5cm)	16000	3573
Dstl Kaggle dataset (spatial resolution: 1.24m)	2312	578
Inria Aerial Image Labeling data (spatial resolution: 30cm)	50540	14440

TABLE VI

COMPARISON OF ACCURACY INDEXES AMONG DIFFERENT MODELS OF PLANETSCOPE DATA SET (SPATIAL RESOLUTION: 3 m)

Models	Overall accuracy	F1 score	IoU
ResNet-DUC	0.7976	0.4593	0.2981
SegNet	0.8263	0.5597	0.3886
ENet	0.8379	0.5831	0.4115
U-Net	0.8435	0.6054	0.4341
FCN-8s	0.8505	0.6292	0.4590
cwGAN-gp	0.8453	0.6339	0.4641
PSPNet	0.8395	0.5948	0.4233
DeepLabv3+	0.8742	0.6592	0.4901
FC-DenseNet	0.8718	0.6556	0.4877
FC-DenseNet+FullCRF	0.8913	0.6580	0.4903
FC-DenseNet+FPCRF	**0.9297**	**0.6698**	**0.5046**

TABLE VII

COMPARISON OF ACCURACY INDEXES AMONG DIFFERENT MODELS OF ISPRS DATA SET (SPATIAL RESOLUTION: 5 cm)

Models	Overall accuracy	F1 score	IoU
ResNet-DUC	0.7475	0.6766	0.5051
SegNet	0.8948	0.8511	0.7408
ENet	0.7711	0.7764	0.6110
U-Net	0.8892	0.8392	0.7229
FCN-8s	0.8617	0.7986	0.6647
cwGAN-gp	0.8926	0.8504	0.7397
PSPNet	0.9141	0.9144	0.8682
DeepLabv3+	0.8995	0.9086	0.8325
FC-DenseNet	0.9186	0.9182	0.8789
FC-DenseNet+FullCRF	0.9298	0.9232	0.8826
FC-DenseNet+FPCRF	**0.9315**	**0.9358**	**0.8974**

numbers of non-building pixels in the ground reference, but detected as buildings and non-buildings in the result, respectively. The F1 score indicates a balance between precision and recall.

A. Feature Extractor Combined With FPCRF

Fig. 5 and Table II list the results of the different CNN models combined with FPCRF. The results of FC-DenseNet combined with FPCRF are more accurate than the other two CNN models combined with FPCRF. This is due to the superiority of FC-DenseNet, which extends the DenseNet architecture to FCN for semantic segmentation. In the DenseNet block, through feature reuse, there are shorter connections within the layers close to the input or output, which strengthen the learning of the discriminated features. Moreover, features are combined by iterative concatenation, which contributes to the improved flow of information. In addition, a standard skip connection between the encoder and decoder is used to pass higher resolution information, which can help the encoder recover spatially detailed information from the decoder.

TABLE VIII

COMPARISON OF ACCURACY INDEXES AMONG DIFFERENT MODELS OF DSTL DATA SET (SPATIAL RESOLUTION: 1.24 m)

Models	Overall accuracy	F1 score	IoU
ResNet-DUC	0.8923	0.5184	0.3499
SegNet	0.9240	0.6050	0.4337
ENet	0.9127	0.6890	0.5189
U-Net	0.9485	0.7576	0.5887
FCN-8s	0.9447	0.7467	0.5779
cwGAN-gp	0.9412	0.7291	0.5732
PSPNet	0.9379	0.6926	0.5297
DeepLabv3+	0.9602	0.7578	0.6100
FC-DenseNet	0.9507	0.7602	0.5928
FC-DenseNet+FullCRF	0.9598	0.7697	0.6034
FC-DenseNet+FPCRF	**0.9604**	**0.7821**	**0.6176**

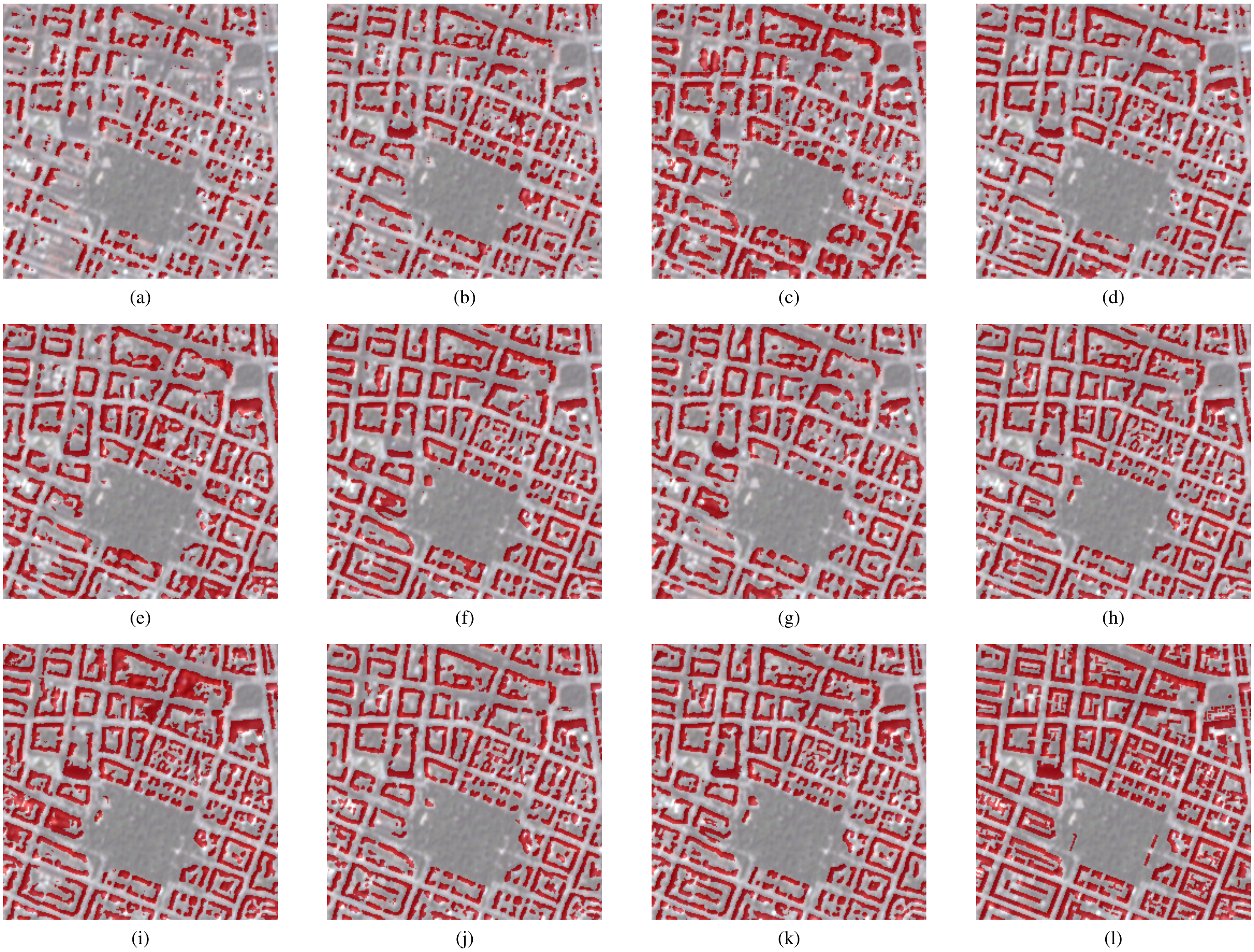
TABLE IX

COMPARISON OF ACCURACY INDEXES AMONG DIFFERENT MODELS OF INRIA DATA SET (SPATIAL RESOLUTION: 30 cm)

Models	Overall accuracy	F1 score	IoU
ResNet-DUC	0.8704	0.7395	0.6097
SegNet	0.8826	0.7845	0.6455
ENet	0.8972	0.8001	0.6669
U-Net	0.9018	0.8027	0.6704
FCN-8s	0.9169	0.8192	0.6837
cwGAN-gp	0.9387	0.8371	0.7198
PSPNet	0.8960	0.7951	0.6599
DeepLabv3+	0.9498	0.8551	0.7299
FC-DenseNet	0.9426	0.8536	0.7258
FC-DenseNet+FullCRF	0.9485	0.8605	0.7312
FC-DenseNet+FPCRF	**0.9581**	**0.8765**	**0.7479**

B. Kernel Selection in FPCRF

FullCRF and ConvCRF only utilize the pairwise potentials from shallow features, which include only appearance and smooth Gaussian kernels. In the implementation of ConvCRF, the unary potential is obtained from CNN, and only the smooth kernel and appearance kernel are utilized for the calculation of the pairwise potential term. FPCRF is able to reduce the complexity of the pairwise potential greatly, which makes the exact message passing and complete feature learning possible. In this regard, we can use the features extracted from CNN models to calculate pairwise potentials, which may facilitate training. The results for the FC-DenseNet combined with FPCRF from the different kernels $k^{(m)}(f_i, f_j)$ are presented in Table III and Fig. 6. The appearance kernel (a) and the smooth kernel (s) are the same as FullCRF and ConvCRF. The feature difference kernel (fd) represents the CNN extracted feature difference calculated with a Gaussian function, and the feature spatial kernel (fs) is the feature difference combined with position difference calculated with a Gaussian function. In the feature cosine kernel (fc), the cosine distance between



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

(e) (f) (g) (h)

(i) (j) (k) (l)

Fig. 11. Predicted results (in red) obtained from (a) ResNet-Duc, (b) SegNet, (c) ENet, (d) U-Net, (e) FCN-8s, (f) cwGAN-gp, (g) PSPNet, (h) DeepLabv3+, (i) FC-DenseNet, (j) FC-DenseNet + FullCRF, (k) FC-DenseNet + FPCRF, and (l) ground truth from Planetscope data set (spatial resolution: 3 m).

feature vectors is implemented as pairwise potential [53]. The detailed formulas of the different kernels are listed in the following.

1) *Appearance Kernel (a):*

$$k^{(a)}(f_i, f_j) = \exp\left(-\frac{|f_{p,i} - f_{p,j}|^2}{2\theta_\alpha^2} - \frac{|f_{I,i} - f_{I,j}|^2}{2\theta_\beta^2}\right) \tag{16}$$

where f_p is the feature of position, f_I is the feature of color, θ_α and θ_β are learnable parameters.

2) *Smooth Kernel (s):*

$$k^{(s)}(f_i, f_j) = \exp\left(-\frac{|f_{p,i} - f_{p,j}|^2}{2\theta_\gamma^2}\right) \tag{17}$$

where θ_γ is a learnable parameter.

3) *Feature Difference Kernel (fd):*

$$k^{(f_d)}(f_i, f_j) = \exp\left(-\frac{|f_{f,i} - f_{f,j}|^2}{2\theta_\delta^2}\right) \tag{18}$$

where f_f is the feature extracted from CNN and θ_δ is a learnable parameter.

4) *Feature Spatial Kernel (fs):*

$$k^{(f_s)}(f_i, f_j) = \exp\left(-\frac{|f_{f,i} - f_{f,j}|^2}{2\theta_\zeta^2} - \frac{|f_{p,i} - f_{p,j}|^2}{2\theta_\eta^2}\right) \tag{19}$$

where θ_ζ and θ_η are learnable parameters.

5) *Feature Cosine Kernel (fc):*

$$k^{(f_c)}(f_i, f_j) = \left(1 - \frac{|f_{f,i} \cdot f_{f,j}|^2}{\| f_{f,i} \| \| f_{f,j} \|}\right). \tag{20}$$

"FC-DenseNet + FPCRF ($a + s$)" is corresponding to the "ConvCRF," which means that unary potential is the segmentation probability obtained from FC-DenseNet, but for the calculation of the pairwise potential term only the smooth kernel and appearance kernel are utilized. It should be noted that in our proposed method "FC-DenseNet + FPCRF (fd)," FC-DenseNet not only provide the segmentation probability as unary potential, but also extracts features for the calculation

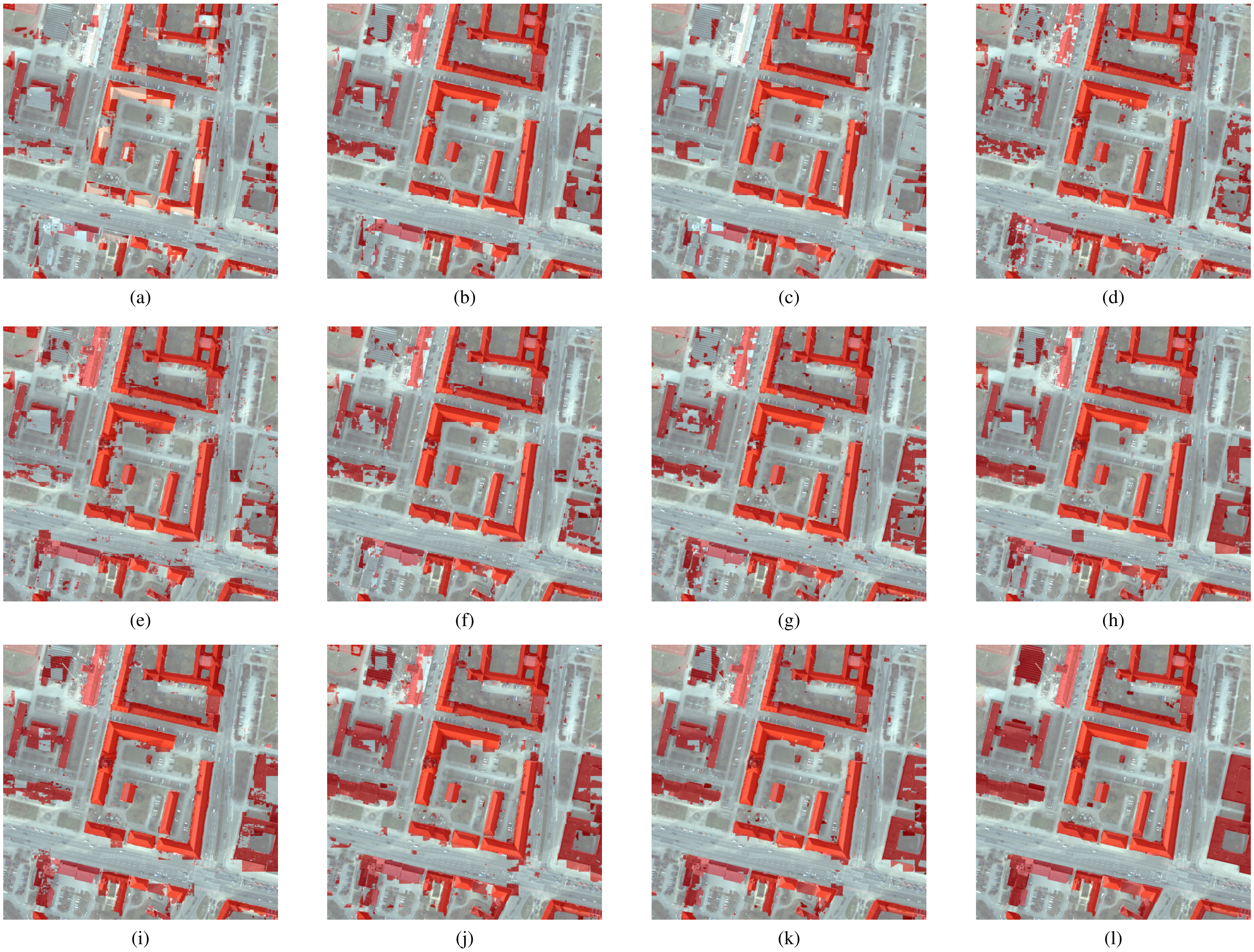


Fig. 12. Predicted results (in red) obtained from (a) ResNet-Duc, (b) SegNet, (c) ENet, (d) U-Net, (e) FCN-8s, (f) cwGAN-gp, (g) PSPNet, (h) DeepLabv3+, (i) FC-DenseNet, (j) FC-DenseNet + FullCRF, (k) FC-DenseNet + FPCRF, and (l) ground truth from ISPRS data set (spatial resolution: 5 cm).

of the pairwise potential term. FC-DenseNet combined with FPCRF using the feature difference kernel (fd) outperforms other kernels in terms of their high F1 score and IoU. There are several reasons for this. The smooth kernel (s), which removes small isolated regions, is not useful in our case. Since the spatial resolution of satellite imagery is coarse, we can preserve isolated small buildings by removing smooth kernel. The feature spatial kernel (fs) controls the degree of nearness that neighboring pixels having similar features may belong to the same class. However, since we have already used filter size to add a locality by filter size, we want the pixels within the filter to contribute equally to the centered pixel. In addition, the appearance kernel (a) has not shown any improvements to the results. This may result from the fact that the RGB information in the appearance kernel (a) is not sufficient to distinguish the buildings from other non-building areas (sometimes roads and buildings have similar RGB information). The feature cosine kernel (fc) shows very low accuracy, which can be explained by the fact that a Gaussian function in feature difference (fd) can remove the noise, but cosine distance can be largely affected by the noise. In this case, when the cosine distance between feature vectors is implemented as a pairwise potential, the final results will suffer from great instability.

C. Hyperparameter Analysis in FPCRF

The hyperparameter filter size r in FPCRF implies that the pairwise potential is zero when the Manhattan distance between the pairs of pixels exceeds r. In order to better understand the influence of the various filter sizes r for building footprint generation, the visual results of FC-DenseNet combined with FPCRF within different filter size r, as well as their accuracy indexes, are shown and compared in Fig. 7 and Table IV. From the visual results, we can observe that when the filter size is not optimal, there are more non-building areas wrongly detected as building areas, and some small buildings are not detected. This can be explained by the fact that filter size is related to the quantity of the most useful neighboring pixels, which contributes to the improvement of the segmentation results.

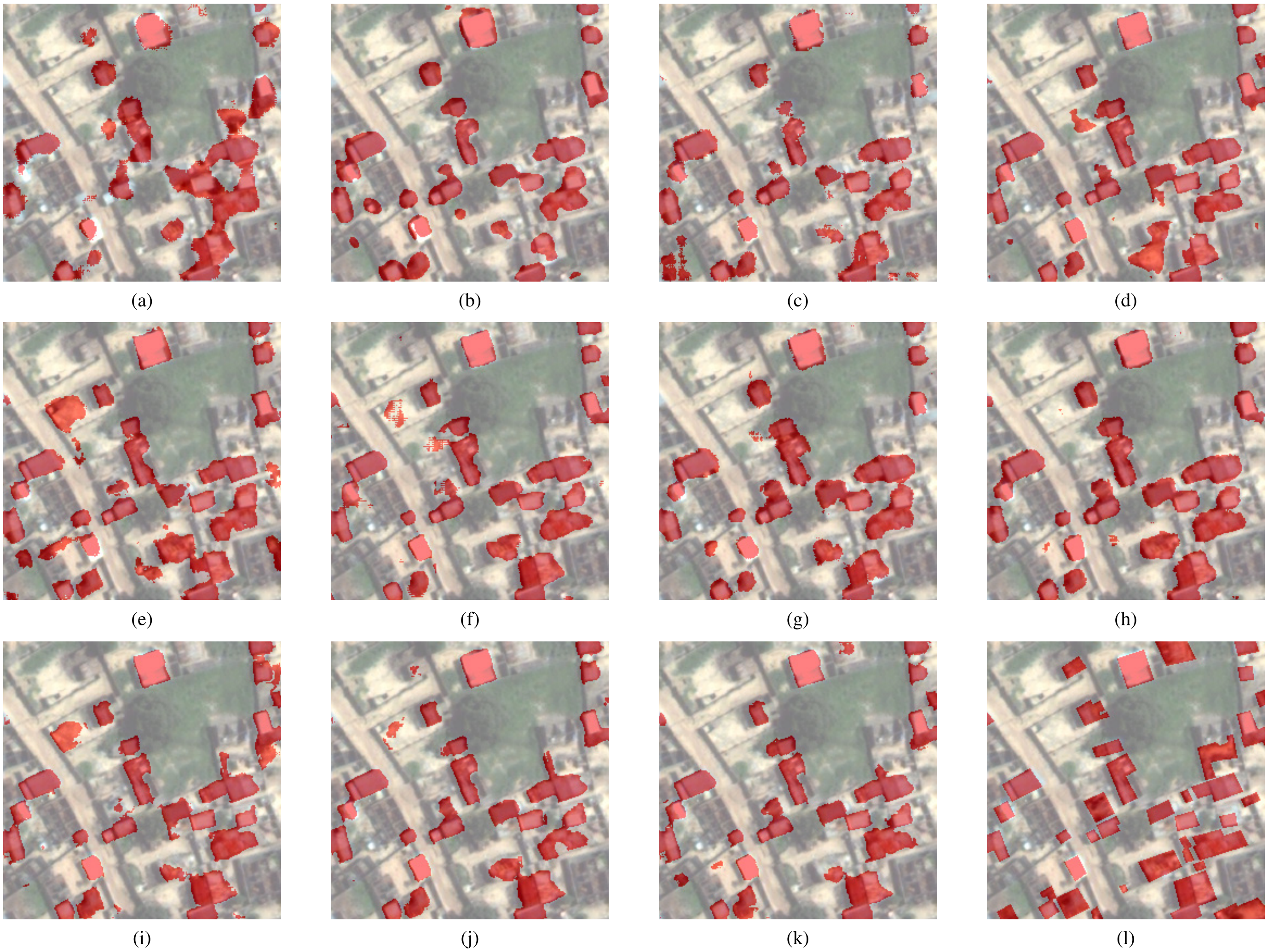


Fig. 13. Predicted results (in red) obtained from (a) ResNet-Duc, (b) SegNet, (c) ENet, (d) U-Net, (e) FCN-8s, (f) cwGAN-gp, (g) PSPNet, (h) DeepLabv3+, (i) FC-DenseNet, (j) FC-DenseNet + FullCRF, (k) FC-DenseNet + FPCRF, and (l) ground truth from Dstl data set (spatial resolution: 1.24 m).

VI. Discussion

A. Additional Data Sets

Another three data sets, ISPRS Benchmark data, Dstl Kaggle data set, and Inria Aerial Image Labeling data are used to test the performance and characteristics of the different networks for building footprint generation.

The first data set is ISPRS Benchmark data [54], shown in Fig. 8. The data set covers the city of Potsdam, which contains 38 aerial images with pixel size 6000 × 6000 and four channels: RGB and near-infrared bands with 5-cm spatial resolution. The corresponding ground truth is also available from the ISPRS benchmark data, which includes six categories. In this article, we take the building class as building and other five classes as non-building; traditional natural color aerial imagery is utilized. The images 7-07, 7-08, 7-09, 7-10, 7-11, 7-12, and 7-13 are used as the validation set, and the remaining images are exploited for training.

Dstl Kaggle data set [55] is the second data set, which provides 57 satellite images with a region of 1 km × 1 km in both 3-band RGB and 16-band multispectral formats. Here, we use three-band images with the spatial resolution 1.24 m. In this data set, ten different classes have been labeled within some images. In this research, the pixels of building are from building class, and those of non-building are the remaining pixels. Ten satellite images with pixel size 3348 × 3348, which has corresponding building class in the ground truth, are exploited for this experiment, which includes eight images with ID (6100-2-3, 6100-1-2, 6100-3-1, 6110-4-0, 6120-2-0, 6120-2-2, 6140-1-2, 6140-3-1) for training, and two images with ID (6100-1-3, 6100-2-2) for validation. Fig. 9 illustrates one satellite imagery sample.

The third data set is Inria Aerial Image Labeling data [33]. This data set contains 360 aerial images of size 5000 × 5000 (at a 30-cm spatial resolution), which have three bands: RGB. In this article, 36 tiles of aerial imagery and their corresponding ground truth (building and non-building) are selected for each of the following five regions: Austin, Chicago, Kitsap County, Western Tyrol and Vienna, where dissimilar urban settlements are covered. The sample data are shown in Fig. 10. To split the training set and test set, we used the first eight images of every city for validation.

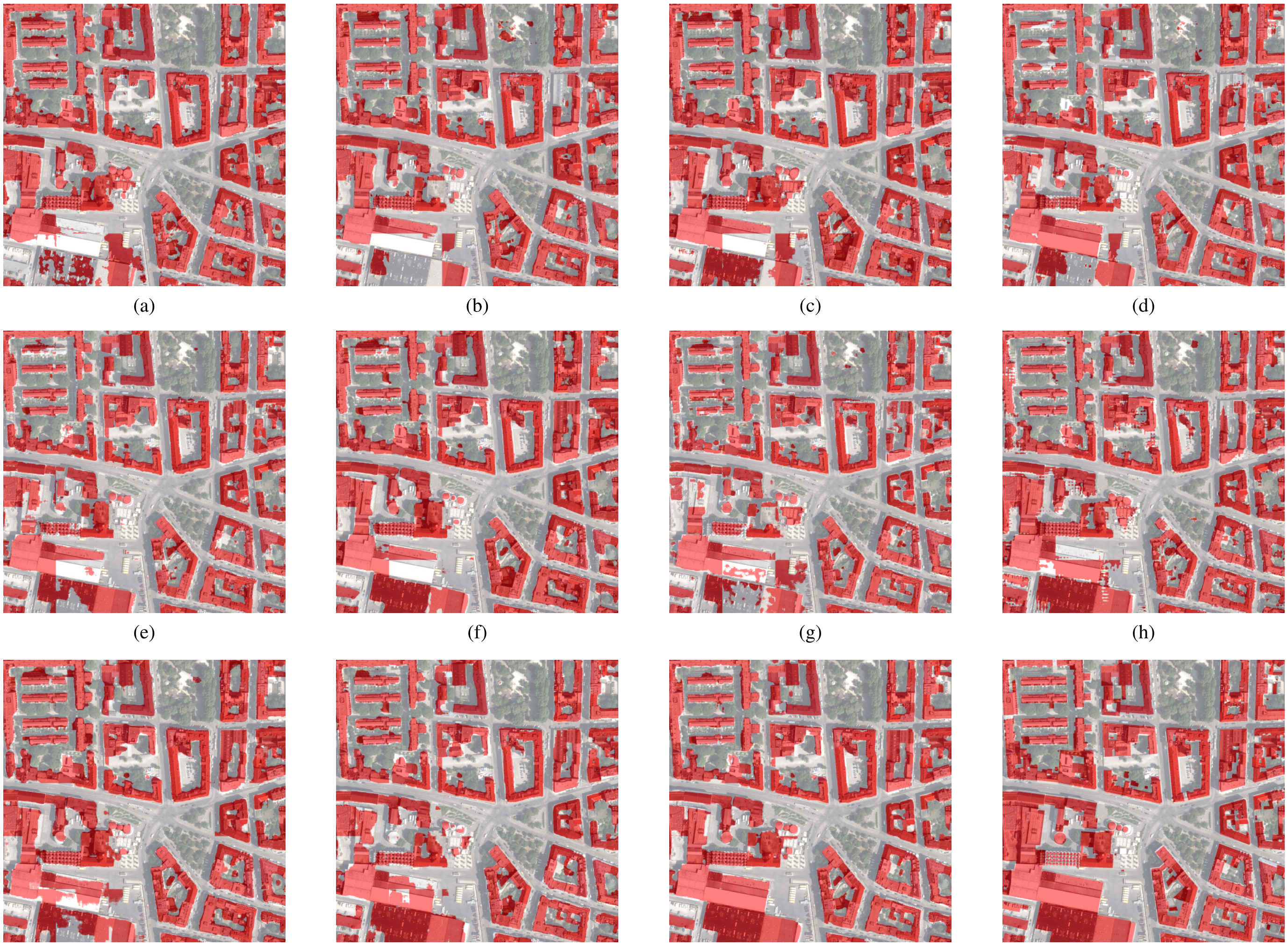


Fig. 14. Predicted results (in red) obtained from (a) ResNet-Duc, (b) SegNet, (c) ENet, (d) U-Net, (e) FCN-8s, (f) cwGAN-gp, (g) PSPNet, (h) DeepLabv3+, (i) FC-DenseNet, (j) FC-DenseNet + FullCRF, (k) FC-DenseNet + FPCRF, and (l) ground truth from Inria data set (spatial resolution: 30 cm).

In order to get more training data, satellite imagery and their corresponding ground truth from Dstl Kaggle data set are cut into small patches of size 256 × 256 pixels with overlap of 64. However, since the numbers of samples from ISPRS benchmark data and Inria Aerial Image Labeling data are enough for network training, aerial imagery and their corresponding ground truth from both data sets are cut into nonoverlapping patches with size 256 × 256 pixels. The numbers of training and validation patches for the additional three data sets are listed in Table V.

B. Comparison With Other Models

In this article, several popular semantic segmentation neural networks from four different data sets were also investigated for comparisons of the proposed method. Their performance in building footprint generation such as accuracy indexes is presented in Tables VI–IX. Moreover, the visual results of different networks are also illustrated in Figs. 11–14. The training and inference time costs of the different methods from Planetscope data set are listed in Fig. 15, where the training time measures the whole training patches for 100 epochs, and inference time refers to the time cost for each patch.

DeepLabv3+ and PSPNet, which are the state-of-art networks for semantic segmentation tasks in computer vision, achieved satisfactory accuracy. These two networks are multiscale processing techniques, which not only allow the refinement of details, but also retain high-level semantic information. They can also take global structure into consideration when making local predictions. ENet is highly superior with respect to both training time and inference time, due to its specific architectures. First, the decoder uses max-pooling indices to produce sparse upsampled maps, which can reduce training time requirements. The input size can also be reduced heavily by the first two blocks, which adopt only a small number of features. Moreover, in the first stage, a max-pooling operation is performed in parallel with a strided convolution, and the resulting feature maps are concatenated, which speeds

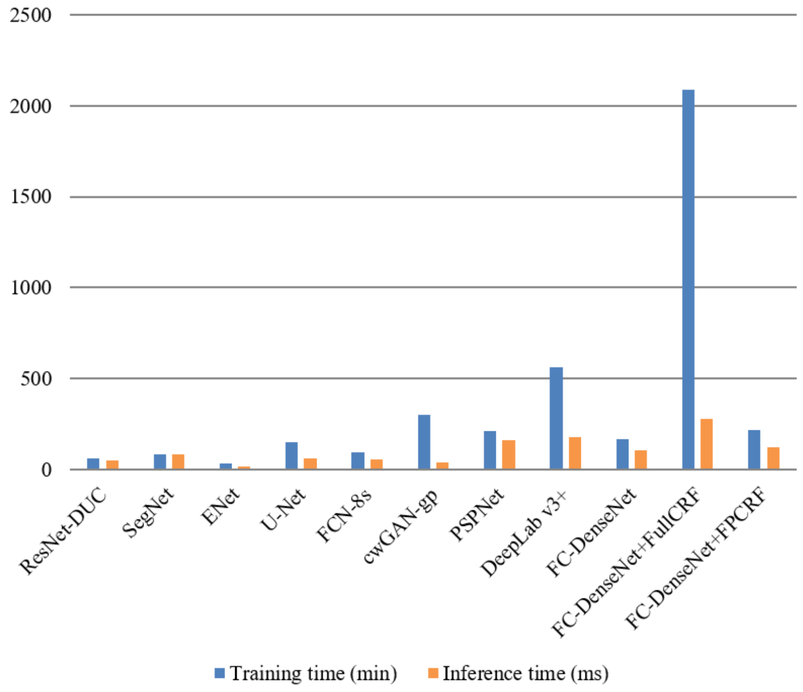


Fig. 15. Comparison of training time and inference time among different models from Planetscope data set (spatial resolution: 3 m).

up inference process of the initial block. Compared to other CNN models, cwGAN-gp, which is a newly proposed network, also shows promising results for building footprint generation. The generator of cwGAN-gp exploits skip connection, which is helpful for retaining the boundary of the buildings. Moreover, the generator and discriminator of the GAN are both improved by the min-max game. However, the difficulty of training of GAN also leads to the longest training time among all the CNN models. Among all CNN models, FC-DenseNet is a superior network with respect to the numerical accuracy and visual results. On one hand, feature maps produced from different layers are concatenated in the DenseNet block, which can improve variation in the input of subsequent layers. On the other hand, high-frequency information can be transferred by a standard skip connection between the encoder and the decoder, which contributes to the recovery of spatial details.

The architectures of the network, such as the feature extractor, decoder, and skip connection, have different significance when applied with satellite imagery of diverse spatial resolution. On one hand, for the higher spatial resolution satellite imagery (ISPRS data set), the feature extractor is rather important. For instance, the accuracy indexes of PSPNet are much higher than those of DeepLabv3+, which means that the ResNet101 in PSPNet has a better feature extraction capability than the Xception in DeepLabv3+. On the other hand, the decoder plays an important role in other data sets, including lower spatial resolution satellite imagery. DeepLabv3+ achieves much better results than PSPNet when applied in lower spatial resolution satellite imagery (Planetscope data set, Dstl data set, and Inria data set), This is owing to the decoder module on top of the encoder output in DeepLabv3+, which contributes to sharper segmentation results. The skip connection in the networks (e.g., U-Net) is also vital to lower spatial resolution satellite imagery, as it is able to concatenate feature maps from both low-level and high-level layers. Hence, it can create a more efficient path for information propagation. However, it consumes more training and inference time, due to the fact that the feature maps from the encoder are transferred and concatenated to the decoder.

However, there are still some problems with CNN-based results such as weak boundaries and coarse pixel-level prediction. Therefore, graph models can be implemented to overcome the drawbacks of exploiting CNN for building footprint generation. CRF is a popular graph model with widespread success in solving semantic segmentation problems. The CRF inference can be used as a postprocessing step, which is not integrated with the training of the CNN. However, in this case, the strength of CRF cannot be fully harnessed. Therefore, we adopt an end-to-end deep learning network to produce sharp boundaries and fine-grained segmentation. FullCRF and FPCRF are combined with CNN models in one unified framework. When connected with CRF-based graph models, the results can be improved as wrongly detected non-building pixels are removed. FC-DenseNet combined with FPCRF has achieved higher IoU and F1 scores than that combined with FullCRF, and can also better preserve the details and sharper boundaries. Moreover, FPCRF can substantially reduce the time needed for the training and inference stages. This superiority can be attributed to two reasons. First, FPCRF uses exact message passing, which avoids the approximation errors resulted from the permutohedral lattice approximation [56] in FullCRF. Second, localized processing in FPCRF can implement the feature learning more efficiently.

VII. Conclusion

Considering that there are weak boundary and coarse pixel-level label predictions in CNN-based results, we have proposed an end-to-end building footprint generation framework integrating CNN and a graph model in this article. Moreover, a number of state-of-the-art CNN models for semantic segmentation are selected to generate building footprints from high-resolution RS images for comparison. The effectiveness of CNN models and the proposed end-to-end CNN-graph model building footprint generation approach is validated on four different data sets: 1) Planetscope satellite imagery of the cities of Munich, Paris, Rome, and Zurich; 2) aerial imagery of the City of Potsdam (North Germany) from ISPRS benchmark data; 3) WorldView3 satellite imagery from Dstl Kaggle data set; and 4) aerial imagery of the city of Austin, Chicago, Kitsap County, Western Tyrol, and Vienna from Inria Aerial Image Labeling data. The experimental results show that building footprint generation based on CNN-graph model-based methods can obtain more accurate results than CNN-based methods alone. Furthermore, FPCRF as the graph model in our proposed framework is effective in producing sharp boundaries and fine-grained segmentation results. On one hand, the completeness of the buildings can be preserved. On the other hand, some non-buildings, which are wrongly detected as buildings by CNN models, can be removed by graph models. Thus, we believe the proposed CNN-graph model method will be of practical value for the monitoring of fast-growing urban areas. In the future, we plan to extend our work to instance segmentation. More types of graph models will also be investigated.

Acknowledgment

The authors thank the Gauss Centre for Supercomputing (GCS) e.V. by providing computing time on the GCS Supercomputer SuperMUC at the Leibniz Supercomputing Center (LRZ) and on the supercomputer JURECA at Forschungszentrum Jülich. They also thank Planet and the Defence Science and Technology Laboratory (DSTL) for providing the data sets.

References

[1] C. Akinlar and C. Topal, “EDLines: A real-time line segment detector with a false detection control,” *Pattern Recognit. Lett.*, vol. 32, no. 13, pp. 1633–1642, Oct. 2011.

[2] J. Wang, X. Yang, X. Qin, X. Ye, and Q. Qin, “An efficient approach for automatic rectangular building extraction from very high resolution optical satellite imagery,” *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 3, pp. 487–491, Mar. 2015.

[3] X. Huang and L. Zhang, “A multidirectional and multiscale morphological index for automatic building extraction from multispectral GeoEye-1 imagery,” *Photogrammetric Eng. Remote Sens.*, vol. 77, no. 7, pp. 721–732, Jul. 2011.

[4] L. Zhang, X. Huang, B. Huang, and P. Li, “A pixel shape index coupled with spectral information for classification of high spatial resolution remotely sensed imagery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 10, pp. 2950–2961, Oct. 2006.

[5] X. X. Zhu *et al.*, “Deep learning in remote sensing: A comprehensive review and list of resources,” *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.

[6] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, “Deep feature extraction and classification of hyperspectral images based on convolutional neural networks,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 6232–6251, Oct. 2016.

[7] W. Liao, F. Van Coillie, L. Gao, L. Li, B. Zhang, and J. Chanussot, “Deep learning for fusion of APEX hyperspectral and full-waveform LiDAR remote sensing data for tree species mapping,” *IEEE Access*, vol. 6, pp. 68716–68729, 2018.

[8] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[9] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.

[10] V. Badrinarayanan, A. Kendall, and R. Cipolla, “SegNet: A deep convolutional encoder-decoder architecture for image segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[11] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[12] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, “ENet: A deep neural network architecture for real-time semantic segmentation,” 2016, *arXiv:1606.02147*. [Online]. Available: http://arxiv.org/abs/1606.02147

[13] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.

[14] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.

[15] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, “Encoder-decoder with atrous separable convolution for semantic image segmentation,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 801–818.

[16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[17] G. Wu *et al.*, “Automatic building segmentation of aerial imagery using multi-constraint fully convolutional networks,” *Remote Sens.*, vol. 10, no. 3, p. 407, 2018.

[18] Y. Xu, L. Wu, Z. Xie, and Z. Chen, “Building extraction in very high resolution remote sensing imagery using deep learning and guided filters,” *Remote Sens.*, vol. 10, no. 1, p. 144, 2018.

[19] B. Bischke, P. Helber, J. Folz, D. Borth, and A. Dengel, “Multi-task learning for segmentation of building footprints with deep neural networks,” 2017, *arXiv:1709.05932*. [Online]. Available: http://arxiv.org/abs/1709.05932

[20] Y. Shi, Q. Li, and X. X. Zhu, “Building footprint generation using improved generative adversarial networks,” *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 4, pp. 603–607, Apr. 2019.

[21] P. Kaiser, J. D. Wegner, A. Lucchi, M. Jaggi, T. Hofmann, and K. Schindler, “Learning aerial image segmentation from online maps,” *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6054–6068, Nov. 2017.

[22] K. Bittner, S. Cui, and P. Reinartz, “Building extraction from remote sensing data using fully convolutional networks,” *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 42, pp. 481–486, May 2017.

[23] P. Wang *et al.*, “Understanding convolution for semantic segmentation,” in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1451–1460.

[24] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[25] 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.*, vol. 140, pp. 20–32, Jun. 2018.

[26] M. Volpi and D. Tuia, "Deep multi-task learning for a geographically-regularized semantic segmentation of aerial images," *ISPRS J. Photogramm. Remote Sens.*, vol. 144, pp. 48–60, Oct. 2018.

[27] L. Mou, Y. Hua, and X. Xiang Zhu, "A relation-augmented fully convolutional network for semantic segmentation in aerial scenes," 2019, *arXiv:1904.05730*. [Online]. Available: http://arxiv.org/abs/1904.05730

[28] L. Mou, L. Bruzzone, and X. X. Zhu, "Learning spectral–spatial–temporal features via a recurrent convolutional neural network for change detection in multispectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 924–935, Feb. 2019.

[29] D. Marcos, M. Volpi, B. Kellenberger, and D. Tuia, "Land cover mapping at very high resolution with rotation equivariant CNNs: Towards small yet accurate models," *ISPRS J. Photogramm. Remote Sens.*, vol. 145, pp. 96–107, Nov. 2018.

[30] B. Le Saux, A. Beaupere, A. Boulch, J. Brossard, A. Manier, and G. Villemin, "Railway detection: From filtering to segmentation networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2018, pp. 4819–4822.

[31] J. E. Vargas-Muñoz, S. Lobry, A. X. Falcão, and D. Tuia, "Correcting rural building annotations in OpenStreetMap using convolutional neural networks," *ISPRS J. Photogramm. Remote Sens.*, vol. 147, pp. 283–293, Jan. 2019.

[32] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Convolutional neural networks for large-scale remote-sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 645–657, Feb. 2017.

[33] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Can semantic labeling methods generalize to any city? The inria aerial image labeling benchmark," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2017, pp. 3226–3229.

[34] K. Bittner, F. Adam, S. Cui, M. Korner, and P. Reinartz, "Building footprint extraction from VHR remote sensing images combined with normalized DSMs using fused fully convolutional networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 8, pp. 2615–2629, Aug. 2018.

[35] D. Marmanis, K. Schindler, J. D. Wegner, S. Galliani, M. Datcu, and U. Stilla, "Classification with an edge: Improving semantic image segmentation with boundary detection," *ISPRS J. Photogramm. Remote Sens.*, vol. 135, pp. 158–172, Jan. 2018.

[36] H. L. Yang, J. Yuan, D. Lunga, M. Laverdiere, A. Rose, and B. Bhaduri, "Building extraction at scale using convolutional neural network: Mapping of the united states," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 8, pp. 2600–2614, Aug. 2018.

[37] S. Ji, S. Wei, and M. Lu, "Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 574–586, Jan. 2019.

[38] X. Li, X. Yao, and Y. Fang, "Building-a-nets: Robust building extraction from high-resolution remote sensing images with adversarial networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 10, pp. 3680–3687, Oct. 2018.

[39] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, "A review on deep learning techniques applied to semantic segmentation," 2017, *arXiv:1704.06857*. [Online]. Available: http://arxiv.org/abs/1704.06857

[40] S. Zheng *et al.*, "Conditional random fields as recurrent neural networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1529–1537.

[41] V. Jampani, M. Kiefel, and P. V. Gehler, "Learning sparse high dimensional filters: Image filtering, dense CRFs and bilateral neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4452–4461.

[42] M. T. T. Teichmann and R. Cipolla, "Convolutional CRFs for semantic segmentation," 2018, *arXiv:1805.04777*. [Online]. Available: http://arxiv.org/abs/1805.04777

[43] H. Su, V. Jampani, D. Sun, O. Gallo, E. Learned-Miller, and J. Kautz, "Pixel-adaptive convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11166–11175.

[44] M. Vakalopoulou, K. Karantzalos, N. Komodakis, and N. Paragios, "Building detection in very high resolution multispectral data with deep learning features," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2015, pp. 1873–1876.

[45] S. Paisitkriangkrai, J. Sherrah, P. Janney, and A. Van-Den Hengel, "Effective semantic pixel labelling with convolutional networks and conditional random fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2015, pp. 36–43.

[46] S. Paisitkriangkrai, J. Sherrah, P. Janney, and A. van den Hengel, "Semantic labeling of aerial and satellite imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 7, pp. 2868–2881, Jul. 2016.

[47] X. Zhuo, F. Fraundorfer, F. Kurz, and P. Reinartz, "Optimization of OpenStreetMap building footprints based on semantic information of oblique UAV images," *Remote Sens.*, vol. 10, no. 4, p. 624, 2018.

[48] J. Yuan and A. M. Cheriyadat, "Learning to count buildings in diverse aerial scenes," in *Proc. 22nd ACM Int. Conf. Adv. Geographic Inf. Syst. (SIGSPATIAL)*. New York, NY, USA: ACM, 2014, pp. 271–280.

[49] C. Sutton, "An introduction to conditional random fields," *Found. Trends Mach. Learn.*, vol. 4, no. 4, pp. 267–373, 2012.

[50] *Planetscope*. Accessed: Apr. 1, 2018. [Online]. Available: https://www.planet.com

[51] D. Krause, "JUWELS: Modular Tier-0/1 supercomputer at the Jülich supercomputing centre," *J. Large-Scale Res. Facilities*, vol. 5, Feb. 2019, Art. no. A135, doi: 10.17815/jlsrf-5-171.

[52] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1251–1258.

[53] Y. Li and W. Ping, "Cancer metastasis detection with neural conditional random field," 2018, *arXiv:1806.07064*. [Online]. Available: http://arxiv.org/abs/1806.07064

[54] *Isprs*. Accessed: Dec. 1, 2018. [Online]. Available: http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html

[55] *Dstl-Kaggle*. Accessed: Dec. 15, 2018. [Online]. Available: https://www.kaggle.com/c/dstl-satellite-imagery-feature-detection

[56] A. Adams, J. Baek, and M. A. Davis, "Fast high-dimensional filtering using the permutohedral lattice," *Comput. Graph. Forum*, vol. 29, no. 2, pp. 753–762, May 2010.

Qingyu Li received the bachelor's degree in remote sensing science and technology from Wuhan University, Wuhan, China, in 2015, and the master's degree in Earth oriented space science and technology (ESPACE) from Technische Universität München (TUM), Munich, Germany, in 2018. She is pursuing the Ph.D. degree with the German Aerospace Center (DLR), Wessling, Germany, and TUM.

Her research interests include deep learning, remote sensing mapping, and remote sensing applications.

Yilei Shi (Member, IEEE) received the Dipl.-Ing. degree in mechanical engineering and the Dr.-Ing. degree in signal processing from Technische Universität München (TUM), Munich, Germany, in 2010 and 2019, respectively.

He is a Senior Scientist with the Chair of Remote Sensing Technology, TUM. His research interests include fast solver and parallel computing for large-scale problems, high-performance computing and computational intelligence, advanced methods on SAR and InSAR processing, machine learning and deep learning for variety of data sources, such as SAR, optical images, and medical images, and PDE-related numerical modeling and computing.

Xin Huang (Senior Member, IEEE) received the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2009, working with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS).

He is a Luojia Distinguished Professor with Wuhan University, where he teaches remote sensing, photogrammetry, image interpretation, and so on. He is the Founder and Director of the School of Remote Sensing and Information Engineering, Institute of Remote Sensing Information Processing (IRSIP), Wuhan University. He has published more than 140 peer-reviewed articles (SCI papers) in the international journals. His research interest includes remote sensing image processing methods and applications.

Dr. Huang was supported by the National Program for Support of Top-notch Young Professionals in 2017, the China National Science Fund for Excellent Young Scholars in 2015, and the New Century Excellent Talents in University from the Ministry of Education of China in 2011. He was a recipient of the Boeing Award for the Best Paper in Image Analysis and Interpretation from the American Society for Photogrammetry and Remote Sensing (ASPRS) in 2010, the second place recipient for the John I. Davidson President's Award from ASPRS in 2018, and was a recipient of the National Excellent Doctoral Dissertation Award of China in 2012. In 2011, he was recognized by the IEEE Geoscience and Remote Sensing Society (GRSS) as the Best Reviewer of the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS. He was the winner of the IEEE GRSS 2014 Data Fusion Contest. He was a Lead Guest Editor of the special issue for the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING in May 2015 and August 2019, the *Journal of Applied Remote Sensing* in October 2016, *Photogrammetric Engineering and Remote Sensing* in November 2018, and *Remote Sensing* in November 2019. He was an Associate Editor of the *Photogrammetric Engineering and Remote Sensing* from 2016 to 2019. He has been serving as an Associate Editor for the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS since 2014 and the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING since 2018. He has also been an Editorial Board Member of the *Remote Sensing of Environment* since 2019 and *Remote Sensing* (an open access journal from MDPI) since 2018.

Xiao Xiang Zhu (Senior Member, IEEE) received the M.Sc., Dr.-Ing., and Habilitation degrees in signal processing from the Technical University of Munich (TUM), Munich, Germany, in 2008, 2011, and 2013, respectively.

She is the Professor for Signal Processing in Earth Observation, Technical University of Munich (TUM), and the Head of the Department "EO Data Science," Remote Sensing Technology Institute, German Aerospace Center (DLR). Since 2019, she has been coordinating with the Munich Data Science Research School (www.mu-ds.de). She is also leading the Helmholtz Artificial Intelligence Cooperation Unit (HAICU)–Research Field "Aeronautics, Space and Transport. She was a Guest Scientist or Visiting Professor with the Italian National Research Council (CNR-IREA), Naples, Italy, Fudan University, Shanghai, China, the University of Tokyo, Tokyo, Japan, and the University of California at Los Angeles, Los Angeles, CA, USA, in 2009 and 2014–2016, respectively. Her main research interests are remote sensing and Earth observation, signal processing, machine learning, and data science, with a special application focus on global urban mapping.

Dr. Zhu is a member of the young academy (Junge Akademie/Junges Kolleg) at the Berlin-Brandenburg Academy of Sciences and Humanities, the German National Academy of Sciences Leopoldina, and the Bavarian Academy of Sciences and Humanities. She is an Associate Editor of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING.