Efficient Building Detection for Large-scale Urban Remote Sensing

Feiyu Qin, Tongcun Zuo, Xuejin Chen

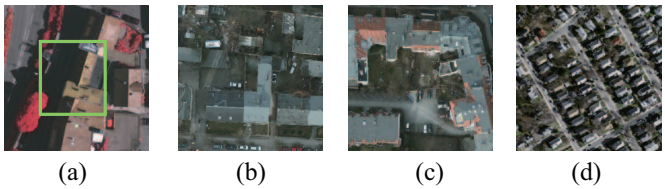
***Abstract*—Extracting buildings from remote sensing images plays an important role in urban applications (e.g., urban planning and digital city). However, this task is quite difficult due to the great diversity of buildings and similarities between buildings and other categories. Recent approaches have attempted to harness the capabilities of deep learning techniques for building extraction. In this paper, we propose a robust method which extracts buildings from large-scale remote sensing images efficiently. And we further build 3D models for extracted building areas. Learning low-level appearance information and high-level semantic information are equally important since buildings in remote images possess various scales and aspect ratios. Hence, in order to make full use of the information extracted from each layer, we propose a simple but effective hierarchical fusion operation which fuses the feature maps between channels stage by stage. By using the modified VGG16 network, we present a novel network named hierarchical fused fully convolution network(HF-FCN). The experiments on several available remote sensing image datasets show that our method achieves state-of-the-art performance. In addition, we combine the segmented building area and available corresponding Digital Surface Model (DSM) map to generate the 3D models of test scene, which as part of our application. (xuejin:Problem: Over-emphasized the buidling detection part, without clearly describe the scope of this paper and the relationship between detection and reconstruction.)**

***Index Terms*—building extraction, hierarchical fusion operation, Hierarchically Fused Fully Convolutional Network (HF-FCN), 3D city modelling**

I. Introduction

(xuejin:Is your goal reconstruction or building extraction? The introduction should explain the overall goal. What are the challenges for building modeling from remote sensing images? What kind of work has been done in the literature? Why do we focus on building detection? What are our contributions?)

BUILDING extraction, which aims to extract rooftop[1] in a large-scale remote sensing image, remains one of the main challenges have been studied for decades in the field of remote sensing. Moreover, automatic extraction of building rooftops from aerial and satellite imagery is an important step in many applications, such as: urban planing, automated map making, 3D city modeling, updating geographical dataset and military reconnaissance. It is particularly difficult to extract rooftop from remote sensing images at the pixel level because of the following three reasons: i) Density of the structures in the scene. A rural scene has low density but an urban scene has high density, with a suburban scene in between (medium density). ii) Shape of the structure. Buildings come in many shapes from simple rectangular blocks with flat roof to complex shapes with intricate, multi-based roof structure. iii) Image quality. Images vary in terms of contrast, resolution, and visibility [1]. Some remote sensing patches are shown in Fig. 1 (xuejin:do not use the figure no directly. using ref..) , which illustrate the challenges of building extraction task.



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

Fig. 1. Examples of remote sensing patches with different kinds of challenges. (a) Shadow occlusion in green frame. (b) Low inter-class differences. (c) High intra class variance. (d) A lot of tiny buildings close to each other.

In the past decades, many researchers made some experimental investigations to extract buildings automatically. In the early days, many knowledge-based methods were put forward by [1], [2], [3], [4], [5] whose basic ideas are derived from prior knowledge of buildings, for instance, buildings are closed polygons made up of some straight lines. Some others are energy based methods which mainly includes the variational level set evolution, improved snake model and graph cut [6], [7], [8].

In recent years, with the development of machine learning, many machine-learning techniques are gradually penetrating into the remote sensing domain. At first, some shallow networks were proposed for multiple object extraction [9], [10], [11], [12]. Afterwards, with increasing computer power, deep learning developed rapidly and introduced into the field of remote sensing. At the same time, some researchers tried Convolutional Neural Networks (CNNs) for aerial images classification and semantic pixel labelling [13], [14], [15], [16], [17].

In this work, a relatively simple, but very effective, manner is proposed and combined into a general CNN architecture for building extraction. We take full advantages of the low-level appearance information as well as high-level semantic information by the novel fusion operation in a way of stage by stage. Numerous experiments conducted on three remote sensing image datasets all obtain fairly good results. We further extend our work to the field of 3D modeling as the part of building detection. And it be easily intergrated into the pipeline of building reconstruction. Our technical contributions

[1]Because the data sets used in our article are high altitude remote sensing images which could be considered as the top views of the ground. Therefore, we do not distinguish the concepts of buildings and rooftops in the subsequent description.

are:

1) A effective hierarchical fusion operation which is specially designed for multi-scale building extraction is proposed. Combining with a trimmed VGG16 Net, a novel network is presented, named HF-FCN that can deal with the problems of different sizes, diverse appearance and mutual occlusion of buildings and etc.
2) HF-FCN is an end-to-end network that does not need any post processing. And the approach is significantly computationally efficient than existing techniques. Besides, the overall accuracy based on HF-FCN exceeds the state-of-art algorithms.
3) A extend exploration for building reconstruction of large-scale urban areas is studied. And the experiments reveal the method well preserve buildings details.

The remainder of this paper is organized as follows. Sec. II sums up the related works in the past. In Sec. III, we introduce the fusion operation and architecture of HF-FCN. The training steps are also presented. And in Sec. IV, a brief description of the dataset used for our task is provided. HF-FCN training strategies, details and its evaluation metrics are also described. In Sec. V,we display and analysis the experimental results. Extension in 3D building modeling are presented in Sec. VI. Finally, the conclusion is discussed in Sec. VII.

II. Related Work

Building extraction is one of the most fundamental problems in remote sensing domain, which has been studied for nearly 30 years. As time goes by, many research achievements have sprung up. We roughly divide these methods into three groups: one is based on the shape prior, another is based on the energy function and machine learning third. Here we briefly review some representative methods that have evolved in the past decades in the different groups respectively. (xuejin:More related work on city modeling/urban modeling. Maybe facade modeling.)

During early days, methods are mainly based on the hypothesis of prior knowledge. Huertas and Nevatia [1] assumed that buildings are rectangular or composed of rectangular components. Based on this, the approach detected lines and corners, traced object boundaries and used shadows to verify. Later, a system [2] for building detection and modelling was proposed with the assumption that the roofs were flat or symmetrical and walls were vertical. Using known ground height and detected rooftop, the reconstructed models could be soon obtained. Further, Noronha and Nosrati [3] transformed the line and intersection points of the image into a graph presentation, and turned the problem of polygon finding into the one that finding loops in the graph. However, it was still estimated on assumption that the buildings are polygonal. In addition, Izadi and Saeedi [4] presented a complete system for building detection and modelling. In the stage of building detection, a tree consisting of intersection points of lines was created and refined based on the found hypotheses. The sun azimuth and elevation angles were used to estimate the height with existing shadows afterwards. As the height of buildings estimated, the three-dimensional polygonal building models were built. In recent years, very high resolution (VHR) optical satellite imagery could be obtained easily. Hence, Wang et al [5] proposed an efficient method for automatic rectangular building extraction from VHR remote sensing images by detecting line segments and grouping lines based on path integrity and closed contour search. (xuejin:If there are two authors, say A and B proposed.... if more than two authors, say A et al. ..)

The aforementioned shape-based methods have a good performance in rural scenes with low density of buildings. Neverthless, there are several limitations of these methods. First, the shape-based methods inherently limited to handle buildings of arbitrary shapes. Second, they may failed to deal with complicated cases, for instance, buildings are close to each other, which thereby is hard to adapt to today's applications. Third, the algorithms using shadows to verify corners and estimate height are greatly limited to obvious shadows and sparse building environment.

Later, several energy-based methods in image segmentation domain have been applied in automatic rooftop extraction. Cote and Saeedi [6] employed corner detection as an initial estimate of the roof, and then refined with level set evolution. Peng et al [7] proposed an approach that segments remote sensing images into high objects, ground and shadow regions, with further refined by an improved snake model. The urban-region-detection problems were casted as one of multiple subgraph matching by Sirmacek and Unsalan [8]. They considered each SIFT keypoint as a vertex, neighborhood between vertexs as edge of the graph and formulated the problem of building detection in terms of graph cut.

Over the past decade, CNNs have achieved great success in the field of computer vision. There are significant amount of efforts on semantic pixel-level classification for extraction buildings in remote sensing. A shallow patch-based network was proposed by Mnih [9] which has only five layers with a 64 by 64 aerial patch as input. And the output of the network was processed by conditional random fields (CRFs). Afterwards, Satio et al. [10] applied two major strategies to improve the performance of the network. One was a channel-wise inhibited softmax (CIS) for getting a multi-label prediction result, the other was model averaging with spatial displacement (MA) for enhancing the prediction result. Alshehhi et al. [11] also adjusted the architecture of network proposed by Mnih through changing the kernel size of convolutional layers and replacing the fully connection layer of the last layer with the average pooling layer. Alternative post-processing strategies such as CRFs and multi-scales were used to improve the final prediction results. Some methods took advantage of the feature extraction capability of CNNs to generate feature descriptions of patches. Paisitkriangkrai et al. [13] made use of both the CNN and hand-craft extracted features, which were combined together to generate predicted labels of each patch. They also used CRFs as post-processing to get a sound result. Zhao et al. [12] proposed a method using edge information of VHR to guide semantic segmentation. Unlike [13], [17] put forward a multi-label pixelwise classification method using the feature vector extracted by a CNN to train a Support Vector Machine (SVM) for classification.

More recently, Long et al. [18] illustrated that Fully Convolutional Networks (FCN) could better handle the problem of multi-label pixel-wise classification. By up-sampling, final predicted result could be the same resolution of the input. Liu et al. [14] did a further research on the formulation proposed by Paisitkriangkrai [13] but used FCN as the branch of CNN and applied a higher-order CRFs as post-processing. Unlike traditional CRFs, the label consistency for the pixels within the same segment were enforced by higher-order CRFs. In order to reduce the information loss during pooling stage, SegNet [19] delivered pooling indices computed in the max-pooling to the decoder. It eliminated the need of learning during the up-sample stage while achieving good segmentation performance. The SegNet architecture was used by Audebert et al. [15] for semantic labeling of remote sensing and got better prediction results compared to the traditional methods. Later, Kampffmeyer at al. [16] proposed a novel idea that using CNN with missing data for urban land cover classification. The idea came from a modality hallucination architecture proposed by Hoffman et al. [20] which learned with side information during training stage.

In the filed of computer vision, the FCNs [18] were introduced as a powerful method for semantic segmentation and have achieved great performance. But, along with the deepening of network, the feature maps with lower resolution which causes the segmentation accuracy decline. In order to weaken the influence caused by pooling, Chen et al. [21] proposed a atrous convolution which enlarged the receptive field and reduced the number of pooling layers at the same time. Vemulapalli et al. [22] later extended the Deeplab [21] with a pairwise network and proposed a Gaussian Conditional Random Field Network for more continuous segmentation results. Afterwards, with the advent of the powerful networks such as ResNet [23], GoogLeNet [24] and their variants [25] [26] [27], a large amount of literature made use of these networks as their backbone for semantic segmentation. Zhao et al. [12] recently developed a pyramid pooling module following the ResNet [23] to get multi-scale feature maps and connected these feature maps with those which before pyramid pooling to create the final prediction. Zuo et al. [28] described a hierarchically fused fully convolutional network, which combined the feature maps from each group of VGG16 Net to generate the final prediction. In this paper, we extend the work of [28] to explore the effect of different layers of features on the final result. And comparing with other mainstream semantic segmentation networks to prove our method more suitable to the building detection task.

(xuejin:Need a paragraph to discuss recent semantic segmentation networks.)

(xuejin:Also cite our accv paper and describe the relationship/difference of this journal paper with it.)

Although above-mentioned CNN-based models have exceeded the traditional methods significantly, all of them lost important hierarchical features encoded in the CNNs. They usually apply the CNN features from the last layer to get a segmentation result. It may omit tiny objects during the process of pooling, and could not handle the situation when the size of buildings have great difference in distribution. Aiming at this case, a hierarchical fusion operation is proposed to combine features extracted from each convolutional layer to capture various information of input images. We will describle the details of our idea below.

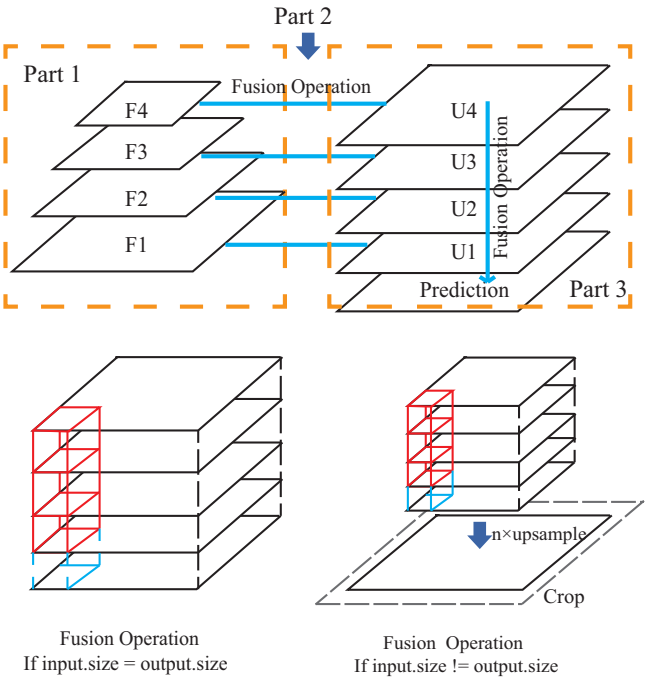


Fig. 2. The first line shows the overview of our network. The second row shows the details of fusion operation. The one on the left is a case where the input is equal to the output and the one on the right is the case of the input not equal to the output.

III. Hierarchically Fused Fully Convolutional Network

In this section, we introduce a novel operation for feature fusion, named hierarchical fusion operation and apply it to the common networks, VGG16 Net and ResNet. The overview diagram in Fig. 2 shows where the fusion operations take effect and how they work. Different from other networks for semantic segmentation, we apply the fusion operation twice to integrate information gradually. Our network consists of three parts. Part 1 is a backbone network whose role is to extract the features at different levels. In theory, arbitrary feature extraction network is applicable to the Part 1. The second part is a process of feature fusion in the first stage, which fuses the feature maps generated from each convolutional(conv) layer. Besides, Part 3 is feature fusion in the second stage. In the second stage of the fusion process, we take full advantage of the information extracted from the second part by learning the connection weights between upsampled feature maps.

(xuejin:Put overview here. Explain the main components of our methods.)

A. Network Architecture

Here, we illustrate our main idea using the VGG16 network as our backbone network, which have been proven to have better performance in experiments. Some modifications

are made to apply to our building extraction task including removing its fc layers and last pooling layer. The reasons of these changes are 1) The fc layer generates a fair number of parameters and takes up too much memory. 2) The existence of fc layer limits the size of input image. 3) After the last pooling layer, the resolution of the feature map is reduced to 1/32 of the input, which is too small to building extraction task. The details of our network using VGG16 Net as backbone network are shown in Fig. 3. The Level 1 in Fig. 3 is a trimmed VGG16 Net which regards as our Part 1, backbone network. (xuejin:Where do you define F1_1? Is it the layer of VGG16 or HF-FCN?)

In order to leverage the information extracted from different layers, we add the fusion branches on the backbone network. The branches between Level 1 and Level 2 in Fig. 3 form the second part of our network. The idea is similar to getting the response of scale functions of images when looking for the SIFT feature points. Unlike the feature descriptor of SIFT, we use the concatenated conv layers as our feature extractor. After getting the responses of different scale functions, the biggest response is selected between adjacent scales of each feature point. The selecting process is determined by weights learned from fusion operations in our network. From the perspective of neural networks, the fusion operations in first stage play a role of both feature compression and semantic information fusion in the same levels. They extracts the information from different scales of receptive field as well as diverse levels of semantics. In addition, the whole weights are learned from the network automatically indicating that network studies the connection relationship among feature maps of same resolution. (xuejin:Fig 1 is the introduction figure.) With the growing of the receptive field, the detailed information is captured by each conv layer from fine-grained to coarser while the semantic information captured from low level to high level. For the task of rooftop extraction, not only the details of the appearance of the buildings captured by shallow layers is needed, but also the line and corner extracted by middle layers and the high-level semantics which mainly come from deep layers are needed. Therefore, we extract various kinds of information by applying the fusion operations to the whole conv layers. The upsampled feature maps from different conv layers are shown in Fig. 4. The U1_1 in Fig. 4(b) means the upsampled feature maps from F1_1 which are feature maps generated from conv1_1 with small receptive field extracts low-level features like edges. In Fig. 4(c), the U1_2 looks like an over-segmentation which groups pixels with similar color or texture into a subregion. In the U2_1, as Fig. 4(d) shows, shape information is augmented. From the U3_3, we can see that regions with significantly varying appearance are merged into an integrated building by considering high-level features. In U4_3 and U5_3, more semantic information of rooftop is got, which can distinguish the rooftop and the roads with similar color and deal with the problem caused by shadow.

After getting the upsampled feature maps, we fuse them into a final prediction. This is the third part of our network. Since all the upsampled feature maps are fused, it is expected to achieve a boost in rooftop segmentation which is shown in Fig. 4(h). In this part, the fusion operation plays a role

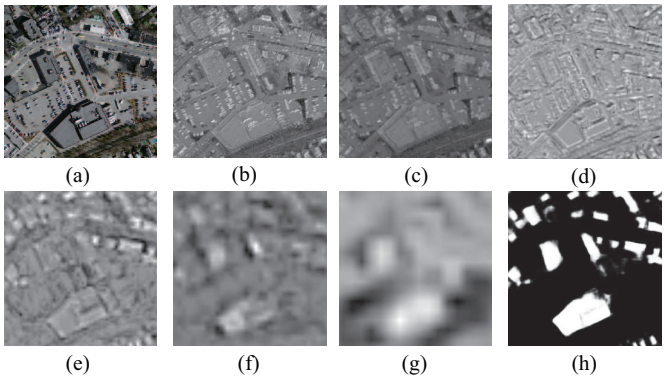


Fig. 4. (a) Input aerial image. (b-g) Feature maps of U1_1, U1_2, U2_2, U3_3, U4_3, U5_3, respectively. (h) Predicted label map.

of feature weighting. Our intention is learning a group of parameters to combine the upsampled feature maps which is similar to a process of feature selection. The expression of the formula is as follows:

$$y(i,j) = \sum_{n=1}^{N} w_n U_n(i,j) \tag{1}$$

where $y(i,j)$ is a point on the output, N is the number of upsampled feature maps and U_n is a upsampled feature map.

B. Network Training

The ground truth M in our dataset is labeled by 0 or 1 to indicate whether a pixel belongs to a roof or not. (xuejin:only roof? or part of the building including facades?) When a remote sensing image X is inputted into the network, the output is a prediction probability map $P(X;W)$ of roof, where W denotes all the parameters that learned by HF-FCN. Each pixel value in $P(X_i;W)$ means the probability of this pixel belongs to rooftop. We use the sigmoid cross-entropy loss function formulated as

$$L(W) = -\frac{1}{|I|}\sum_{i=1}^{|I|}[\tilde{m}_i \log P(X_i;W) + (1-\tilde{m}_i)\log(1-P(X_i;W))], \tag{2}$$

where $\tilde{m}_i$ is label of X_i, $|I|$ is the number of pixels in the input image X.

IV. Experiments

To verify the effectiveness of the proposed network, extensive experiments have been conducted on three remote sensing datasets. In this section, the experimental setup is described including details of datasets, training settings of HF-FCN and different criterion for evaluation.

A. Dataset Description

a) Massachusetts dataset: Massachusetts dataset consists of 151 aerial images of the Boston area which covers roughly 340 square kilometers. The resolution of each image is 1500×1500 with the spacial resolution of 1 meter per pixel. And the images are composed of red, green and blue channels. This dataset is built by Mnih while ground-truth is produced by Saito et al. The dataset is split into three parts, a training set

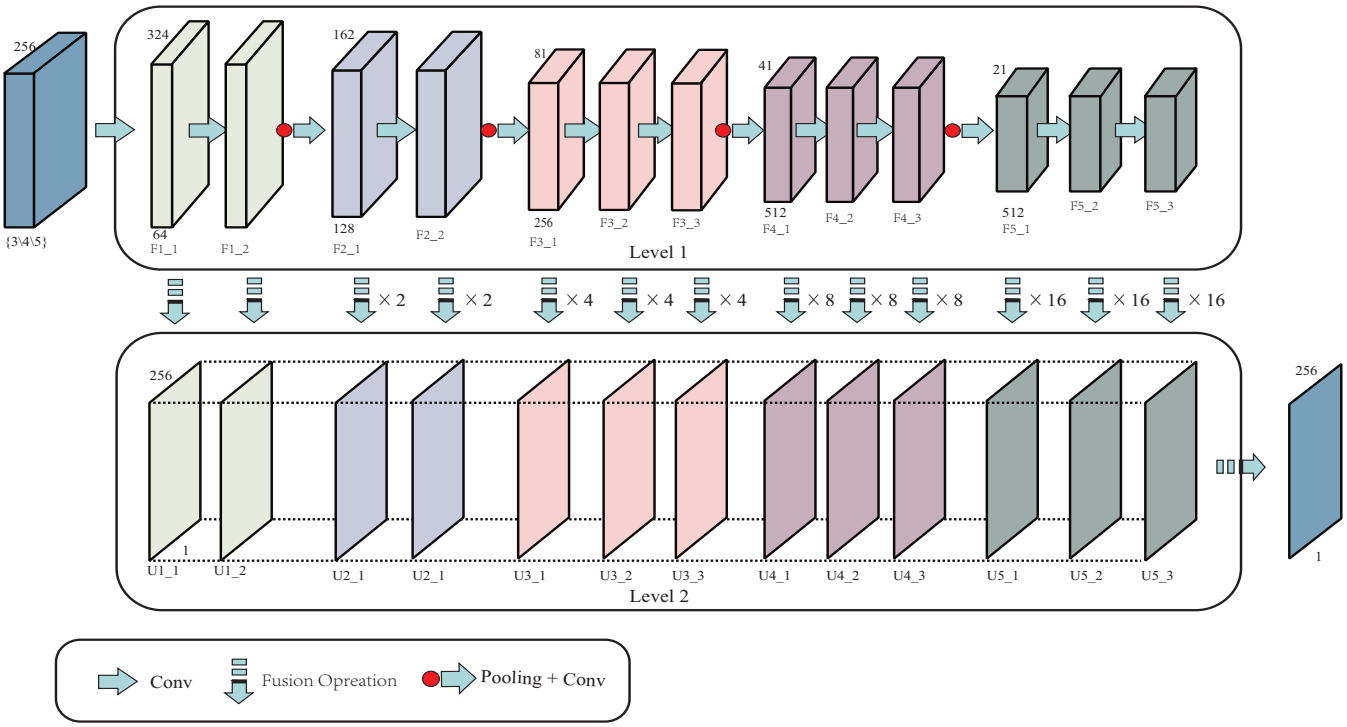


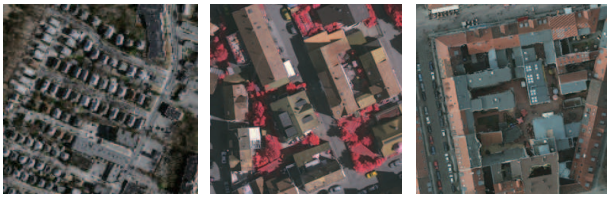
Fig. 3. Overall architecture of HF-FCN. The input of HF-FCN could be 3, 4 or 5 channels for RGB, DSM, nDSM. The backbone network is VGG16 network which contribute to the Level 1. The F1_1 in Level 1 indicates the feature maps generated by conv1_1. In Level 2, 13 upsampled feature maps are cropped to the same size of input. $\times 2$ next to the fusion operation means 2 times of upper sampling. U1_1 in Level 2 means the upsampled feature map of F1_1, and so forth. (xuejin:What is the different between our network with U-Net or other FCN networks?)

of 137 images, a test set of 10 images and validation set of 4 images. To train the network, we create a set of image tiles for training and validation. The detailed description is shown in Table [**?**].

b) Vaihingen dataset: Vaihingen dataset is captured over Vaihingen which is a relatively small village with many detached buildings and small multi story buildings in Germany. This dataset contains 16 labeled images whose spacial resolution is 9cm per pixel. It consists of near infra-red, red, green, blue imagery with corresponding normalized digital surface models (nDSMs) and row DSMs. The dataset is divided into training set, validation set, and test set which have 11 images, 2 images, and 3 images respectively. The same crop operations are done as the Massachusetts dataset.

c) Potsdam dataset: In the Potsdam dataset, there are 24 labeled images whose ground sampling distance is 5cm. This dataset shows a typical historic city with large building blocks. In order to grasp the global information of the building, the spacial resolution of the original image is reduced from 6000×6000 to 1500×1500. Each image in this dataset contains 5-channel information: red, green, yellow, DSM and nDSM. We split the dataset into training, validation and test sets in a proportion of $7:2:1$.

Data augmentation is made on the Vaihingen dataset and the Potsdam dataset. One reason is that methods using dataset $a)$ do not extend the data. Hence, to make a fair comparison with other methods, we also do not extend it. Another reason is that the data quantity of dataset $b)$ and $c)$ is not enough which may lead to inadequate training. Therefore, some measures of data augmentation are made in dataset $b)$ and $c)$ including data rotation and mirror flipping. Components of the datasets are listed in Table I. Meanwhile, some sampled patches of dataset a, b, c are shown in Fig. 5(xuejin:5).



(a) (b) (c)

Fig. 5. Sample patches on the three datasets (a) Massachusetts dataset (b) Vaihingen dataset (c) Potsdam dataset

TABLE I
COMPOSITION OF DATASET

	Masschusetts	Vaihingen	Potsdam
Labeled images	151	16	24
GSD	1m	9cm	5cm
Bands	R,G,B	IR,R,G,DSM	IR,R,G,B,DSM
Training images	137	11	17
Training patches	75938	115088	85000
Validation images	4	3	4
validation patches	2500	28376	25000
Test images	10	2	3

TABLE II
PARAMETERS FOR NETWORK TRAINING

	Massachusetts	Vaihigen	Potsdam
mini-batch size	18	15	15
initial learning rate	10^{-5}	10^{-6}	10^{-5}
test_interval	1000	1000	1000
training iteration	10000	10000	10000
momentum	0.9	0.9	0.9
clip_gradients	16000	10000	10000
weight_decay	0.02	0.005	0.005

B. Training Settings

HF-FCN is trained on dataset a firstly owing to large amounts of training data. The pre-trained VGG16 Net model is used to finetune our HF-FCN. We use the stochastic gradient descent algorithm with the learning rate divided by 10 for each 8000 iterations to train our network. The drop-out ratio is set to 0.5 which avoids overfitting. When the HF-FCN converges on the dataset a), we transfer it to the other datasets. All experiments in this paper are performed using the deep learning framework Caffe and trained on a single NVIDIA Titan 12GB GPU. Besides, the hyper-parameters are listed in Table II (xuejin:III).

C. Evaluation Metrics

Several evaluation metrics are adopted in our work. For dataset a), the most common metrics are correctness (precision) and completeness (recall). The stantdard (ρ=0) and relaxed (ρ=3) precision and recall scores are used to evaluate the prediction results. Here the relaxed precision means the predicted pixels are within ρ pixels of a true pixel while the relaxed recall is the true pixels are within ρ pixels of a predicted pixel. Moreover, the time cost is used to measure the efficiency of our HF-FCN. For dataset b) and c), we use correctness, completeness and F1 score as evaluation metrics.

$$completeness = \frac{TP}{TP + FN}, \tag{3}$$

$$correctness = \frac{TP}{TP + FP}, \tag{4}$$

$$F1\_score = 2 \cdot \frac{completeness \cdot correctness}{completeness + correctness} \tag{5}$$

where TP indicates the true positives, FP implies the false positive, TN means the true negatives and FN refers to the false negatives.

V. RESULTS AND DISCUSSION

In this section, the proposed method using dataset a, b, c are compared to the recent non-deep-learning algorithms, such as Minh-CNN [9], Satio-multi [10] and Context [15]. Furthermore, it is also compared with some recent deep-learning based approaches, including FCN [18], SegNet [19], Deeplab [21] and U-Net [29].(xuejin:What else?) Moreover for HF-FCN itself, we expect to investigate the effects of extracted information from different layers on the final prediction. Thus, some variants which combine different up-sampling feature maps from Level 2 are proposed with details shown in Fig. 6.

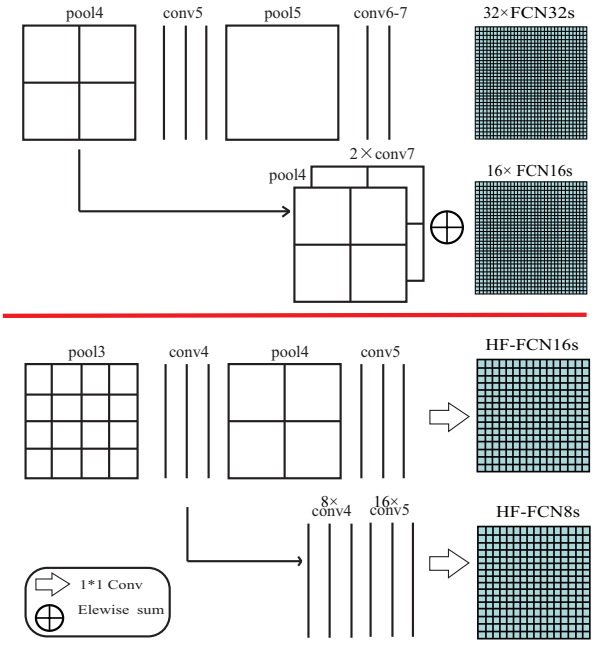


Fig. 6. HF-FCN variants. The feature maps generated from final group are fused into a coarse result, which is HF-FCN16s. The variant called HF-FCN8s concatenates the feature maps from the last 2 groups with the same fusion operation, and so on.

In Fig. 6, we also compare our variants with the FCN to illustrate the differences between these two types of variants. In addition, in order to find a better backbone network, we try the VGG16 Net and ResNet. The details are shown below. (xuejin:What about to change the backbone network?)

A. Massachusetts dataset

On the Massachusetts dataset, our method is compared to both the non-deep-learning algorithms and deep-learning based approaches. Table III(xuejin:4) present the quantitative analysis. A standard and relaxed precision and recall are amply to make a comparison. From the result, our method shows obvious superiority in terms of speed and precision. When comparing with SatiomultiMA&CIS, the standard and relaxed recall of our method are 5.5% and 1.3% higher than it. Meanwhile, the time cost is reduced from 67.84s to 1.07s and the speed is promoted about 63 times. These significant improvements demonstrate that HF-FCN achieves best performance in effectiveness and efficiency.

Extensive comparisons are made between HF-FCN and other mainstream methods in semantic segmentation domain. The quantitative and visual results are shown in Table III and Fig. 9, respectively. On the charts, we can see that our method better performance in speed and precision. And the details and integrity of the building are well preserved by our method.

To explore the effects of the feature maps generated from each conv layer on the final result, variants of HF-FCN which are counterpart of FCN are designed. Unlike FCN, a fusion operation rather than summation are leveraged to build our HF-FCN 16s, 8s, 4s. Fig. 6 shows the constrast diagram. The performance of these variants are shown in Fig. 7, Fig. 8

TABLE III
CORRECTNESS AT BREAKEVEN OF HF-FCN V.S. [9] [10] [11] ON MASSACHUSETTS TEST SET. COST TIME IS COMPUTED IN THE SAME COMPUTER WITH A SINGLE NVIDIA TITAN 12GB GPU

	Recall (ρ = 3)	Recall (ρ = 0)	Time (s)
Mnih-CNN [9]	0.9271	0.7661	8.70
Mnih-CNN+CRF [9]	0.9282	0.7638	26.60
Satio-multi-MA [10]	0.9503	0.7873	67.72
Satio-multi-MA&CIS [10]	0.9509	0.7872	67.84
Alshehhi-GAP+seg [11]	0.955	–	–
FCN_4s [18]	0.839	0.6147	4.20
SegNet [19]	0.7710	0.5675	2.39
U-Net [29]	0.9638	0.8357	3.165
DeepLab_V2 [21]	0.9620	0.7575	1.89
HF-FCN(VGG16 Net)	0.9643	0.8424	1.07
HF-FCN(ResNet)	0.9588	0.8175	2.42
HF-FCN16s	0.9330	0.7233	0.85
HF-FCN8s	0.9643	0.8171	0.93
HF-FCN4s	0.9632	0.8394	0.99

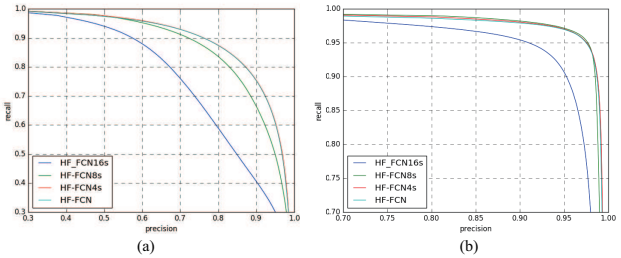


Fig. 7. The relaxed precision-recall curves from HF-FCN variants with two slack paramters. The biggest gap occurs between HF-FCN16s and HF-FCN8s, which indicates the most additional information coming from middle layers.

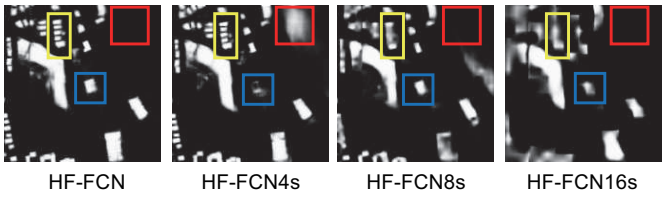


Fig. 8. Prediction results of HF-FCN, HF-FCN4s, HF-FCN8s and HF-FCN16s. The yellow box shows the continuous refinement of the tiny buildings. The red and blue boxes show the mutual promotion and contradiction between different layers.

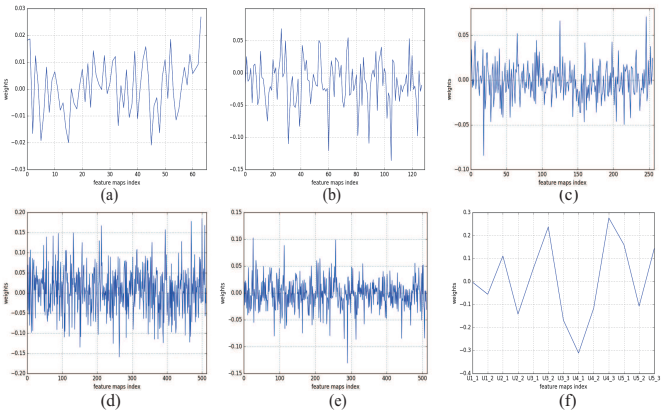


Fig. 10. (a) is weights learned by F1_1, (b) is weights learned by F2_1, (c) is weights learned by F3_1, (d) is weights learned by F4_1, (e) is weights learned by F5_1, (f) is weights learned by Level 2.

and Table III(xuejin:Figure 8, Figure 9 and Table V). From the disgrams, we get the following conclusions. First, the prediction result obtained from the last layer gets a coarse result, which loses much of location information that are mainly encoded in the shallow feature maps. Second, the largest gap presented between HF-FCN16s and HF-FCN8s about 9% in recall rates, it may suggest that the most information supplement to the HF-FCN is got in middle layers. Third, the PR curves of HF-FCN4s and HF-FCN almost coincide. It illustrates the low-level information has little effect on the prediction results. Forth, with the addition of the shallow feature map, the network is more distinct for the segmentation of tiny buildings, which solves the problem of easy adhesion to adjacent buildings. Since, all the conv layers contained useful hierarchical information that is critical to the final prediction.

In the end, we want to prove that our fusion operations learn the connections between features. Connection weights are shown in Fig. 10. The weights are not the same, which means that fusion operation have effect on feature combination. From the Fig. 10 (f), we can arrive at the conclusion that the different layers have virous effect on the final result. For example, the U1_1 has little effect on the prediction while the U3_2 and U4_3 have bigger role on the final prediction. It also in accordance with our experimental results that middle layers provide more information. (xuejin:Weight for what? to fuse feature map? The distribution does not make too much sense.)

B. *Vaihingen dataset*

On Vaihingen dataset, three experiments are undertaken to explore the effects of different inputs, diverse variants and various methods. We utilize three kinds of combinations of image channels as inputs. The inputs of the 3 channels are IR, R, G and adding the nDSM as the forth channel. Based on it, DSM is added and made up 5-channel input. We use three standards to make a more comprehensive evaluation. The evaluation results are shown in Table IV, which illustrates that 3-channel input performed better than the other. The Rec and Pre in Table IV means the recall and precision of prediction results. And F1 indicates the F1_score of results. The number in bold shows the best results in validation and test set. Corresponding visual results are shown in Fig. 13.

(xuejin:Do you compare with others?)

We compare with some other methods which use the same dataset. The detail comparison results are shown in Fig. 14. From a visual perspective, our method gets a much more refined roof region, both on continuity of labels and integrity of structural.

The results of diverse variants are shown in Fig. 11. The HF-FCN_1 in Fig. 11 indicates that the last conv layer in Level 2 does not use the previous trained model to initialize. And HF-FCN means that the whole layers use the pre-trained model to initialize. From the curves, the performance of HF-FCN exceeds the variants and gets a excellent result. Additionally, using the pre-trained weights of Level 2 has a significance in the final results.

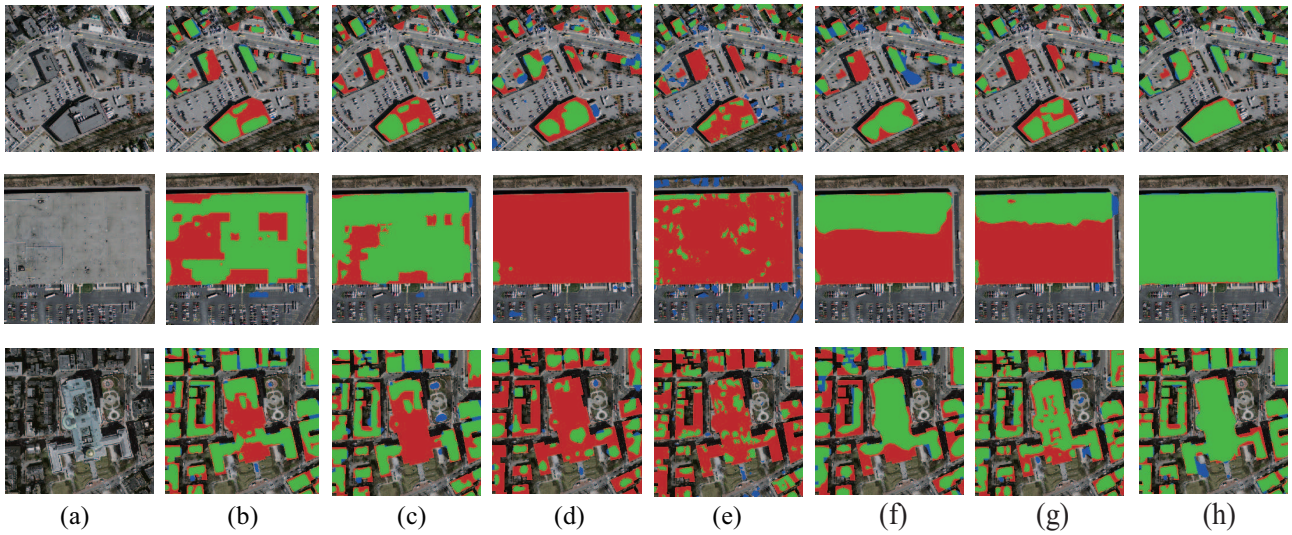


Fig. 9. (a) input images. (b) Results of Mnih-CNN+CRF. (c) Results of SatiomultiMA&CIS. (d) Results of FCN4s . (e) Results of SegNet. (f) Results of DeepLab_V2. (g) Results of U-Net. (h) Our results. TP are shown in green, FP are shown in blue and FN are in red.

TABLE IV
PERFORMANCE COMPARISON OF THE RESULTS OF DIFFERENT INPUTS ON VAIHIGEN DATA SET. (XUEJIN:WHAT ARE THE NUMBERS IN THE IMG COLUMN?)

	Img	3_in: IR, R, G			4_in: IR, R, G, nDSM			5_in: IR, R, G, DSM, nDSM		
		Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
Val	11	0.911	0.906	0.909	0.936	0.900	0.917	0.890	0.900	0.900
	28	0.94	0.875	0.906	0.96	0.792	0.868	0.952	0.823	0.883
	34	0.965	0.899	0.930	0.987	0.902	0.942	0.972	0.918	0.944
	Ave	0.939	0.894	**0.915**	0.961	0.865	0.909	0.939	0.880	0.907
Test	15	0.918	0.930	0.924	0.883	0.917	0.9	0.833	0.931	0.88
	30	0.921	0.929	0.926	0.931	0.827	0.876	0.875	0.877	0.876
	Ave	0.919	0.930	**0.925**	0.907	0.872	0.888	0.858	0.900	0.878

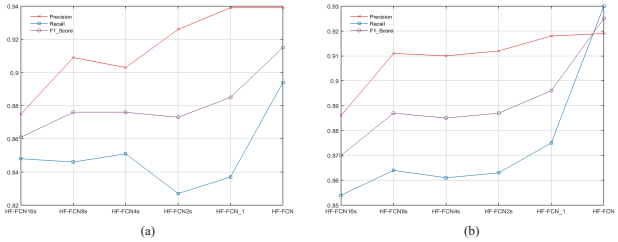


Fig. 11. Results of HF-FCN variants on Vaihingen dataset. (a) (b) shows the precision, recall and F1_score of validation set and test set of Vaihingen dataset respectively.(xuejin:Bigger font)

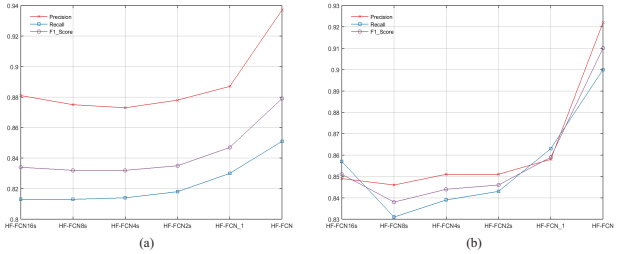


Fig. 12. Results of HF-FCN variants on Potsdam dataset. (a) (b) shows the precision, recall and F1_score of validation set and test set of Potsdam dataset respectively.

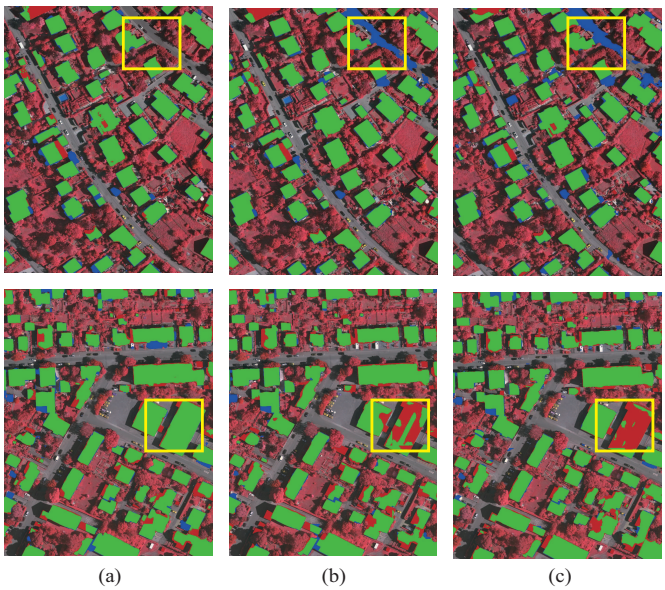


Fig. 13. Prediction results on Vaihingen dataset. (a) (b) (c) shows results of the 3-channel input, 4-channel input and 5-channel input of Vaihingen dataset respectively. Here, TP are shown in green, FP are shown in blue and FN are in red.

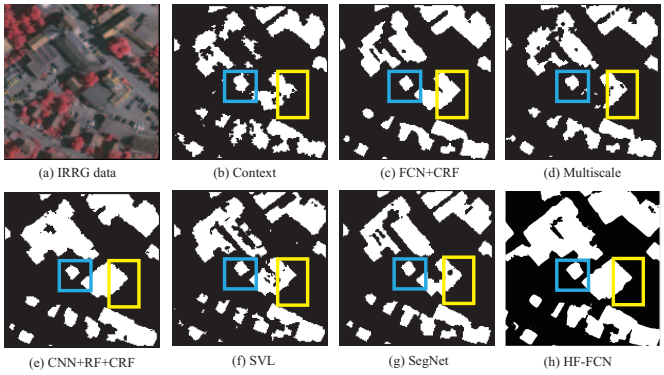


Fig. 14. Results of different methods. (a) is input image, (b)(d)(g) are results of [15], (c) is result of [30], (f) is result of [31], (g) is our result. The blue and yellow frames show some details between these methods.

C. Potsdam dataset

The same experiments are implemented on Potsdam dataset. First, We utilize DSM and IR information as extra inputs based on the RGB input. The specific quantitative evaluation and intuitive visual prediction results are shown in Table V and Fig. 15. In the validation process, the 4-channel input gets better overall performance. Meanwhile, the 5-channel input seems perform better in the course of testing. From the visual results, the 5-channel input network gets lower error detection rate which is shown on the image with small blue areas. And from the 3-channel input to 5-channel input, the F1 score increases from 0.879 to 0.891 on the validation set and increases 0.031 on the test set. It indicate that the other information of geographical feature have a certain effect on the final result.

We compare HF-FCN with other methods using the Potsdam dataset. Some qualitative results are shown in Fig. 16. From the figure, we can easily see that HF-FCN got more remarkable segmentation results. And edges and structure of buildings are preserved better.

As done on Vaihingen dataset, contrast experiments of HF-FCN variants are implemented. The performance curve of HF-FCN variants are shown in Fig. 12. The HF-FCN_1 in Fig. 12 indicates that the last conv layer in Level 2 does not use the previous trained model to initialize. And HF-FCN means that the whole layers use the pre-trained model to initialize. Initialization of parameters has a greater promotion on the final results.

VI. Application

The segmentation results are further used to 3D building reconstruction. We make use of the depth map and generate the point cloud of remote sensing images. After that, the 3D building reconstruction methods could applied to the generated point cloud. In this paper, the approach proposed by zhou [32] are used to generate the 3D models of buildings in the scene. Fig. 17 and Fig. 18 show the 3D models of Vaihingen and Potsdam dataset respectively. The details of a single building are also presented. From the figures, we can see the 3D models preserve the characteristics of buildings well whether the structure of the roof or the simplification of details.

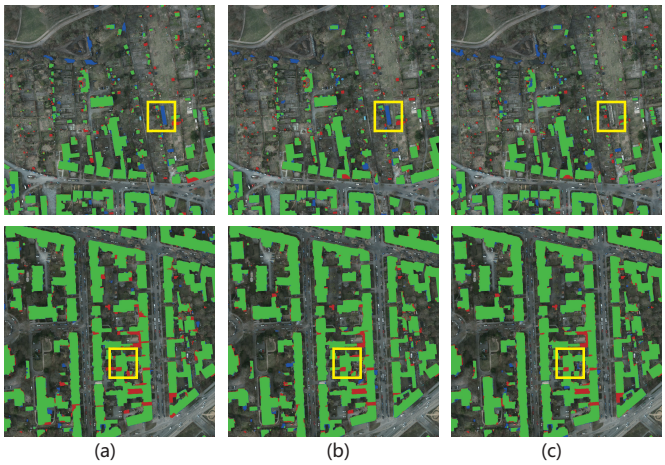


Fig. 15. Prediction results on potsdam dataset. (a) (b) (c) shows results of the 3-channel input, 4-channel input and 5-channel input of Vaihingen dataset respectively. Here, TP are shown in green, FP are shown in blue and FN are in red.

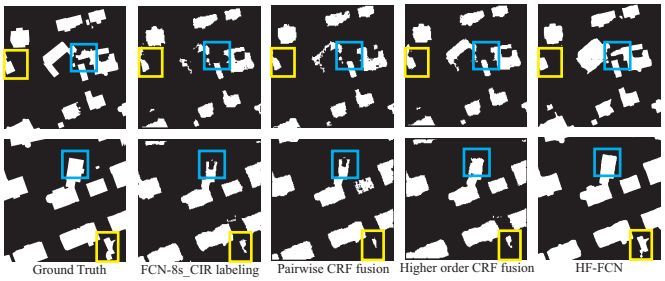


Fig. 16. Results of different methods. The second column is the results of using only the FCN with CIR. Pairwise CRF fusion shows the result of fusing FCN-8s_CIR with LiDAR data in a pairwise CRF. Higher-order CRF are used to generate the results shown in third column. Our results are shown in last colunm.

VII. Conclusion

In this paper, an efficient building detection approach is proposed and have a further application in building reconstruction. Using proposed feature fusion operations, a novel CNN architecture is presented for building extraction, named HF-FCN. Unlike previous non-deep-learning algorithms, our method relies on a CNN for feature extraction which is more robust and efficient to the different scales of buildings. On the other

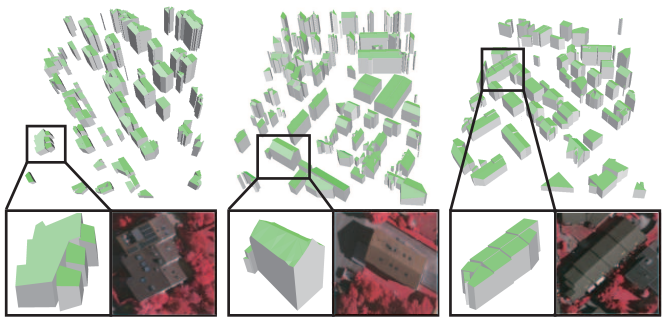


Fig. 17. The 3D modelling of Veihingen dataset. The single building model and its corresponding optical patch were shown together.

TABLE V
PERFORMANCE COMPARISON OF THE RESULTS OF DIFFERENT INPUTS ON POTSDAM DATA SET

	Img	3_in:RGB			4_in:RGB,IR			5_in:RGB,IR,nDSM		
		Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
Val	2_11	0.917	0.950	0.933	0.917	0.978	0.946	0.934	0.976	0.954
	4_10	0.937	0.945	0.941	0.926	0.943	0.936	0.947	0.946	0.946
	5_11	0.930	0.972	0.950	0.959	0.975	0.966	0.956	0.977	0.967
	7_10	0.964	0.536	0.689	0.950	0.590	0.728	0.939	0.554	0.697
	Average	0.937	0.851	0.879	0.937	**0.872**	**0.894**	**0.944**	0.864	0.891
Test	2_12	0.897	0.868	0.882	0.920	0.959	0.939	0.944	0.965	0.955
	6_7	0.894	0.902	0.898	0.915	0.909	0.912	0.901	0.918	0.909
	7_8	0.975	0.929	0.951	0.977	0.950	0.957	0.976	0.946	0.960
	Average	0.922	0.900	0.910	0.937	0.935	0.936	**0.940**	**0.943**	**0.941**

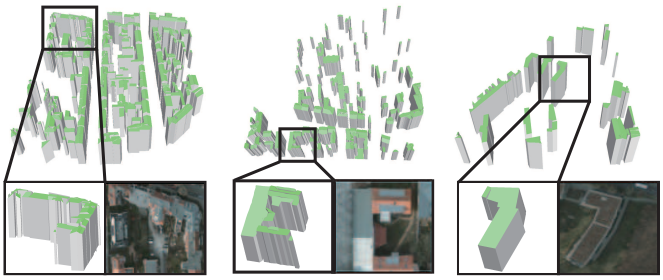


Fig. 18. The 3D modelling of Potsdam dataset. The single building model and its corresponding optical patch were shown together.

hand, distinct from the previous deeplearning based methods, we utilize the multi-scale inherent information within the CNN and refine the details by a fusion manner stage by stage. In addition, an application of 3D building reconstruction depend on the segmentation results is implemented. Compared to the existing 3D reconstruction methods, our proposed approach greatly accelerates the part of building extraction. Finally, our study suggests that even with the powerful semantic expressive ability of CNNs and their good robustness to scale, it is still critical to address multi-scale problems utilizing hierarchical feature maps encoded in CNNs.

REFERENCES

[1] A. Huertas and R. Nevatia, "Detecting buildings in aerial images," *Computer Vision, Graphics, and Image Processing*, vol. 41, no. 2, pp. 131–152, 1988.

[2] S. Noronha and R. Nevatia, "Detection and modeling of buildings from multiple aerial images," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, no. 5, pp. 501–518, 2001.

[3] M. S. Nosrati and P. Saeedi, "A novel approach for polygonal rooftop detection in satellite/aerial imageries," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*. IEEE, 2009, pp. 1709–1712.

[4] M. Izadi and P. Saeedi, "Three-dimensional polygonal building model estimation from single satellite images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 6, pp. 2254–2272, 2012.

[5] 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 Geoscience and Remote Sensing Letters*, vol. 12, no. 3, pp. 487–491, 2015.

[6] M. Cote and P. Saeedi, "Automatic rooftop extraction in nadir aerial imagery of suburban regions using corners and variational level set evolution," *IEEE transactions on geoscience and remote sensing*, vol. 51, no. 1, pp. 313–328, 2013.

[7] J. Peng, D. Zhang, and Y. Liu, "An improved snake model for building detection from urban aerial images," *Pattern Recognition Letters*, vol. 26, no. 5, pp. 587–595, 2005.

[8] B. Sirmacek and C. Unsalan, "Urban-area and building detection using sift keypoints and graph theory," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 4, pp. 1156–1167, 2009.

[9] V. Mnih, "Machine learning for aerial image labeling," Ph.D. dissertation, University of Toronto (Canada), 2013.

[10] S. Saito, T. Yamashita, and Y. Aoki, "Multiple object extraction from aerial imagery with convolutional neural networks," *Electronic Imaging*, vol. 2016, no. 10, pp. 1–9, 2016.

[11] R. Alshehhi, P. R. Marpu, W. L. Woon, and M. Dalla Mura, "Simultaneous extraction of roads and buildings in remote sensing imagery with convolutional neural networks," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 130, pp. 139–149, 2017.

[12] W. Zhao, S. Du, Q. Wang, and W. J. Emery, "Contextually guided very-high-resolution imagery classification with semantic segments," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 132, pp. 48–60, 2017.

[13] S. Paisitkriangkrai, J. Sherrah, P. Janney, V.-D. Hengel *et al.*, "Effective semantic pixel labelling with convolutional networks and conditional random fields," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 36–43.

[14] Y. Liu, S. Piramanayagam, S. T. Monteiro, and E. Saber, "Dense semantic labeling of very-high-resolution aerial imagery and lidar with fully-convolutional neural networks and higher-order crfs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 76–85.

[15] N. Audebert, A. Boulch, H. Randrianarivo, B. Le Saux, M. Ferecatu, S. Lefevre, and R. Marlet, "Deep learning for urban remote sensing," in *Urban Remote Sensing Event (JURSE), 2017 Joint*. IEEE, 2017, pp. 1–4.

[16] M. Kampffmeyer, A.-B. Salberg, and R. Jenssen, "Urban land cover classification with missing data using deep convolutional neural networks," *arXiv preprint arXiv:1709.07383*, 2017.

[17] Y. He, S. Mudur, and C. Poullis, "Multi-label pixelwise classification for reconstruction of large-scale urban areas," *arXiv preprint arXiv:1709.07368*, 2017.

[18] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[19] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for scene segmentation," *IEEE transactions on pattern analysis and machine intelligence*, 2017.

[20] J. Hoffman, S. Gupta, and T. Darrell, "Learning with side information through modality hallucination," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 826–834.

[21] 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," *arXiv preprint arXiv:1606.00915*, 2016.

[22] R. Vemulapalli, O. Tuzel, M.-Y. Liu, and R. Chellapa, "Gaussian conditional random field network for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3224–3233.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[24] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions,"

in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[25] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2818–2826.

[26] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning." in *AAAI*, 2017, pp. 4278–4284.

[27] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 5987–5995.

[28] T. Zuo, J. Feng, and X. Chen, "Hf-fcn: Hierarchically fused fully convolutional network for robust building extraction," in *Asian Conference on Computer Vision*. Springer, 2016, pp. 291–302.

[29] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.

[30] D. Marmanis, J. D. Wegner, S. Galliani, K. Schindler, M. Datcu, and U. Stilla, "Semantic segmentation of aerial images with an ensemble of cnss," *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2016*, vol. 3, pp. 473–480, 2016.

[31] M. Gerke, "Use of the stair vision library within the isprs 2d semantic labeling benchmark (vaihingen)," 01 2015.

[32] Q.-Y. Zhou and U. Neumann, "2.5 d building modeling with topology control," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 2489–2496.