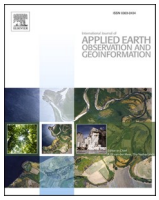
International Journal of Applied Earth Observations and Geoinformation 100 (2021) 102341

Contents lists available at ScienceDirect

International Journal of Applied Earth Observations and Geoinformation

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



Adaboost-like End-to-End multiple lightweight U-nets for road extraction from optical remote sensing images

Ziyi Chen [a,d], Cheng Wang [b], Jonathan Li [c], Wentao Fan [a], Jixiang Du [a], Bineng Zhong [d,*]

[a] Department of Computer Science and Technology, Fujian Key Laboratory of Big Data Intelligence and Security, Xiamen Key Laboratory of Computer Vision and Pattern Recognition, Huaqiao University, China
[b] School of Information Science and Engineering, Xiamen University, China
[c] Department of Geography and Environmental Management, University of Waterloo, Waterloo, Canada
[d] Department of Computer Science, Guangxi Normal University, Guilin 541004, China

ARTICLE INFO

Keywords:
Road extraction
Remote sensing image
Semantic segmentation

ABSTRACT

Road extraction from optical remote sensing images has many important application scenarios, such as navigation, automatic driving and road network planning, etc. Current deep learning based models have achieved great successes in road extraction. Most deep learning models improve abilities rely on using deeper layers, resulting to the obese of the trained model. Besides, the training of a deep model is also difficult, and may be easy to fall into over fitting. Thus, this paper studies to improve the performance through combining multiple lightweight models. However, in fact multiple isolated lightweight models may perform worse than a deeper and larger model. The reason is that those models are trained isolated. To solve the above problem, we propose an Adaboost-like End-To-End Multiple Lightweight U-Nets model (AEML U-Nets) for road extraction. Our model consists of multiple lightweight U-Net parts. Each output of prior U-Net is as the input of next U-Net. We design our model as multiple-objective optimization problem to jointly train all the U-Nets. The approach is tested on two open datasets (LRSNY and Massachusetts) and Shaoshan dataset. Experimental results prove that our model has better performance compared with other state-of-the-art semantic segmentation methods.

1. Introduction

Road extraction has become a crucial technique in many daily application scenarios, such as navigation, road network update, road network planning, automatic driving and intelligent transportation, etc. Compared with traditional methods for road area labeling (such as manually or GPS based methods), remote sensing image based methods are much more balance in the economy and labeling accuracy. Manually labeling is with hard manual burden, while GPS based road extraction methods usually loss the road detail information, such as road width, road edge, etc.

Although road extraction from optical remote sensing images is such meaningful, it still faces many challenges such as complex background, occlusions, etc. Thus, road extraction from remote sensing images is a hot study area and have attracted many researchers' attention. Currently, most state-of-the-art road extraction methods are deep learning model based (Liu et al. 2018, Guo and Wang 2020, Tao et al. 2019, Lu et al. 2019, Gao et al. 2019, Liu et al. 2019, Abdollahi et al. 2020, Yang et al. 2019). It is found that the deeper a model is, the better performance it will achieve once the model is well trained. So most state-of-the-art road extraction deep learning models are with deep layers and large scale of parameters. A model with deeper layers and higher complexity may achieve better performance, but it may also means the harder for training and easier for over-fitting, asking to take actions (such as drop out, pooling) for preventing over fitting problems(Xie et al. 2016). To alleviate the training difficulty, this paper aims at using a lightweight model to achieve the same good or even better performance. A simple idea is to combine multiple lightweight models to improve the performance. However, the cruel fact is that a model with deeper layers and larger scale of parameters usually gets better performance than the performance of multiple lightweight models' combination. The reason lays on two aspects. (1) multiple lightweight models are isolatedly trained and combined with simple rules, resulting to unable to release the largest power of models' combination. (2) The multiple models has

* Corresponding author.
E-mail addresses: chenziyihq@hqu.edu.cn (Z. Chen), cwang@xmu.edu.cn (C. Wang), junli@uwaterloo.ca (J. Li), fwt@hqu.edu.cn (W. Fan), jxdu@hqu.edu.cn (J. Du), bnzhong@gxnu.edu.cn (B. Zhong).

https://doi.org/10.1016/j.jag.2021.102341
Received 12 March 2021; Received in revised form 29 March 2021; Accepted 6 April 2021
0303-2434/© 2021 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

no relationship of stepwise enhancement, which is the key idea of Adaboost.

To solve the above two problems, aiming at combining lightweight deep models to achieve better performance than deeper and larger models, this paper proposes an Adaboost-like combination of multiple lightweight U-Nets with end-to-end training. To make multiple U-Nets be Adaboost-like combination and has relationship of stepwise enhancement, each output of prior U-Net is used as the input of next U-Net. To make the multiple U-Nets be jointly training, we design the loss function as a multiple-objective optimization problem. We test our model on three datasets. One is the open dataset LRSNY (Chen, 2020). The second dataset is the Shaoshan dataset, which is not publicly available. The third dataset is Massachusetts Road dataset (Mnih, 2013), which is a publicly open dataset for road extraction from remote sensing images. In the experiments, our model shows an impressive and satisfactory results. We also compared with many other state-of-the-art models, the better comparison results prove the effectiveness of our AEML U-Nets. We also analyze the effectiveness of multiple U-Nets and Adaboost-like strategies.

The contributions of this paper lie on:

(1) We propose an Adaboost-like combination strategy of multiple U-Nets.
(2) We design multiple U-Nets' combination as multiple-objective optimization problem, thus we jointly train multiple U-Nets.

The rest of this paper is organized as follows. Section 2 reviews related works. We give a detail introduction about AEML U-Nets in Section 3. Section 4 illustrates the results. Section 5 shows the analysis. Finally, we give a conclusion in Section 6.

2. Related works

2.1. Adaboost

Ensemble methods have a prosperity time before the great breakages of deep learning methods, such as Adaboost (Li and Lei Wang 2008, Zhu et al. 2006, Hu et al. 2008, Zhang and Zhang 2008, Lv and Nevatia 2006, Viola and Jones 2001, Zhang and Yang 2018, Zhang et al. 2020, Chen et al. 2021). The main idea of Adaboost is to train a set of weak classifiers and then through a rule (such as linear combination) to combine those weak classifiers together as one, thus make weak classifiers become a strong good classifier (Li and Lei Wang 2008). Adaboost have been proved be quite effective in many studies. Li et al proposed SVM-based component classifiers with Adaboost and achieved better performance compared with standard SVM (Li and Lei Wang 2008). Hu et al proposed an Adaboost-based algorithm for network intrusion detection (Hu et al. 2008). In the experiments, they proved that Adaboost-based method can obtain better detection performance compared with other non-Adaboost-based methods. Viola et al proposed an Adaboost based fast and robust face detector and presented quite good results (Viola and Jones 2001). Zhang proposed a novel Adaboost framework with robust threshold and structural optimization for regression (Zhang and Yang 2018). They tested the model on UCI benchmarks and showed state-of-the-art results. Zhang et al. proposed a small target recognition model through combining CNN and Adaboost (Zhang et al. 2020). In their experiments, the accuracy was improved about 20%. Chen et al. used AdaBoost-KNN for emotion classification for dynamic emotion recognition in human-robot interaction (Chen et al. 2021). Their experimental results demonstrated the dynamic emotion understanding ability of robots in human-robot interaction.

2.2. Multiple-Objective optimization

It has been found that a learning paradigm in which data from multiple tasks is used with the hope to obtain superior performance over learning each task independently (Caruana 1997). Since then, multi-task learning has attracted attentions of large amounts of researchers (Zhou et al. 2011, Rosenbaum et al. 2017, Rudd et al. 2016, Misra et al. 2016, Liu et al. 2017, Shen et al. 2017). In multi-task learning, multiple tasks are jointly trained and sharing inductive bias between them (Volpia and Tuiab 2018). In essentially, multi-task learning is a multiple-objective problem. The different tasks can be divided into two types. The first type is that different tasks are conflict and it needs a trade-off between different tasks. The second type is that different tasks have no conflict, thus it does not need a trade-off consideration among different tasks. Multi-task learning has many meaningful application scenario and shows its great power in many researches. Ranjan et al designed a joint multi-task learning CNN network for face detection (Ranjan et al. 2017). In their network, they combined face detection, localization, pose estimation and gender recognition together and simultaneously ran the above different works as a multi-task project. In their extensive experiments, they proved that their model performed impressive good results compared with many other competitive methods. Rad et al. used multi-task learning strategy for super-resolution from a single image, they illustrated good performance in their experiments (Rad et al. 2020). Liu et al designed gas classification and concentration estimation as multi-task learning in a LSTM network, in which different tasks shared basic features (Liu et al. 2020). The strategy was proved be effective for performance improvement. Liu et al. proposed a hierarchical clustering multi-task learning method for joint human action grouping and recognition (Liu et al. 2017). They showed breakthroughs in the experimental results. Shen et al. used multi-task deep learning for object skeleton extraction in natural images and obtained good results (Shen et al. 2017).

2.3. Road extraction from remote sensing images

Area extraction and centerline extraction are the two major aspects of works for road extraction from remote sensing images (Cheng et al., 2017a, 2017b, Zhang et al. 2017). Given a remote sensing image, area extraction based methods will automatically label out all the pixels which belong to roads (Tao et al. 2019, Zhang et al. 2017, Maboudi et al. 2018, Sghaier and Lepage 2016, Alshehhi et al. 2017, Coulibaly et al. 2017, Lv et al. 2017, Li et al. 2016, Yin et al. 2016, Poullis 2014, Chen et al., 2020a, 2020b, Ren et al. 2020). While in a centerline extraction based method, it will figure out the road skeleton through algorithms (Guo and Wang 2020, Liu et al. 2019, Cheng et al., 2017a, 2017b, Zang et al. 2017, Cheng et al. 2016, Zang et al. 2016, Hui et al. 2016, Courtrai and Lefèvre 2016, Liu et al. 2016, Shi et al., 2014a, 2014b).

As roads have outstanding shape feature compared with other ground targets, the morphological features are utilized for road extraction (Cheng et al., 2017a, 2017b, Sghaier and Lepage 2016, Alshehhi et al. 2017, Shi et al., 2014a, 2014b). Due to the development of machine learning methods, e.g. Sparse Representation, Support Vector Machine (SVM), Tensor-Voting and Hough Forest, etc. based methods, many researchers used machine learning methods combining with artificial designed features for road extraction from remote sensing images and obtained many achievements (Li et al. 2016, Poullis 2014). Morphological feature based road extraction methods usually only focus extraction strategy on morphological features such as curves, lines, etc. Thus, Morphological feature based methods may suffer from wrong extractions where have similar morphological features with roads. Under complex background of remote sensing images, only relying on morphological features for road extraction is forceless.

To enforce the feature power during road extraction, traditional artificial designed features are spring up and combined with machine learning methods. Poullis et al. used a framework called Tensor-Cuts which did not need any threshold for pre-processing. The framework was especially suitable for the extraction of linear features which are the major features of roads, thus they achieved good results in the experiments (Poullis 2014). Lv et al. proposed a road area extraction method,

in which they proposed a combination feature containing color, local entropy and HSC features (Lv et al. 2017). The feature extraction procedure in their method was adaptive and sparsity. In the experiments, they obtained satisfactory results. Movaghati et al. combined PF (particle filtering) with EKF (extended kalman filtering) for road extraction and obtained quite good performance in the experiments (Movaghati et al. 2010). In their method, they focused their attention on fixing the continuations of roads when the extraction was break by obstacles or junctions.

Morphological features and traditional artificial designed features are also combined to promote the road extraction performance. Leninisha et al. used geometric active deformable model to extraction road network in remote sensing images (Leninisha and Vani 2015). In their method, they used Water Flow to extraction different shape types of road junctions. After the extraction of road junctions, they combined the junctions to figure out the final road network. They tested in images with high resolution. And they achieved good results on the tests. Ziems et al. proposed a road databases verification through combining ten road detection methods (Ziems et al. 2017). In their approach, different method was applied to different detection scenario. The verification results was obtained after the stage by stage verification through all the ten methods. In their experiment, they proved their method can achieve state-of-the-art performance and was flexible due to the adaptation of verification modules' number. Xiao et al. proposed a road detection method through fusing the data of cameras and LiDARs (Xiao et al. 2017). In their method, they used a novel conditional random field model to fuse data obtained from different sources. To classify the pixels in image and points in LiDAR data, they used booted decision tree. Through a probabilistic way of integrations, they successfully fused image and LiDAR data to get a good road detection result.

For traditional artificial designed feature based machine learning methods, they usually need to consider the sensibility of model parameters. As using artificial designed features, the generalization ability is unstable.

Except traditional artificial designed feature based machine learning methods, recently, deep convolutional neural networks (CNN) have brought a large amount of showy and excellent breakthroughs in various tasks of compute vision (He et al. 2016, Krizhevsky et al. 2012, Redmon and Farhadi 2017, Ren et al. 2015, Taigman et al. 2014, Simonyan and Zisserman 2015, Szegedy et al. 2015, He et al. 2020, Huang et al. 2017). Meishvili et al. proposed a CNN based model which used images with very low resolution and audio information for face super-resolution reconstruction (Meishvili et al. 2020). Their experimental results exhibited quite impressive performance. Chen et al. proposed a visual tracking network which made the anchors be free (Chen et al., 2020a, 2020b). They verified their model on large amount of datasets and presented impressive performance.

CNN based methods have also illustrated its great power and excellent performances in road extraction from remote sensing images (Liu et al. 2018, Guo and Wang 2020, Tao et al. 2019, Lu et al. 2019, Gao et al. 2019, Liu et al. 2019, Abdollahi et al. 2020, Yang et al. 2019). Currently, the CNN based methods usually can achieve the best performance compared with artificial designed feature based methods. Zhang et al. proposed a U-Net based model which combined residual structure together for road extraction (Zhang et al. 2017). They tested their model on remote sensing images and proved that their method can obtain better results comparing with other state-of-the-arts. Chen et al. combined Dirichlet Mixture Models (DMM) and CNN model together to make the road extraction be a stage by stage framework (Chen et al., 2020a, 2020b). They firstly used unsupervised DMM to obtain the coarse extraction result, then a CNN model is applied for precisely classification. They tested a large dataset and achieved impressive performance compared with other methods. Alshehhi et al. used superpixel strategy to instead pixel strategy, thus can effectively use contextual structures. Besides, they combined texture features for road classification and proposed a shortest approach to deal with the discontinuous problems. In their experiments, they showed quite good results (Alshehhi et al. 2017).

Although CNN based methods have led a great step forward for road extraction from remote sensing image, several problems still need to be concerned. Current deep learning based road extraction methods are rarely considering using multi-objective learning, which has been proved can improve a model's performance. Also, the combination strategy of multiple models is simple (usually use simple linear combination) and the multiple models are usually isolatedly trained. The models' combination and combined learning strategies still need to be studied. Considering to solve the above problems, we propose this paper.

3. Materials and methods

In this section, we first illustrate the datasets used in this paper. Then, we give an introduction about our model architecture. Third, we give a detail presentation about model designs. Finally, we illustrate the multiple-objective learning and Adaboost-like combination of our model.

3.1. Datasets

In this paper, we use third datasets for performance evaluation. The first dataset is LRSNY (Large Road Segmentation Dataset from Optical Remote Sensing Images of New York) (Chen, 2020), which is a publicly open dataset and can be obtained from the website: ftp://154.85.52.76/LRSNY/. The images in LRSNY are optical remote sensing images with a resolution of 0.5 m. 716, 220 and 432 images are contained in the training, validation and test sets, respectively. The original images include two versions with different image sizes: 1000 × 1000 and 256 × 256. The images has a resolution of 256 × 256 are the smaller version of images with a resolution of 1000 × 1000. In our experiments, we use images with a resolution of 256 × 256. Fig. 1 shows several sample images of training, validation and test sets in LRSNY dataset.

The second dataset used in our experiment is Shaoshan dataset (Chen et al., 2020a, 2020b). The dataset is not publicly available due to the copyright problem. The Shaoshan dataset is a 11125 × 7918 Pleiades optical image of part Shaoshan (in China) with a resolution of 0.5 m. We follow the doings in our prior work and cut the large image into 49 pieces, including 29 training images and 20 test images. To fit our model input size, we further divide the images into smaller ones with a resolution of 256 × 256 for both training and test images. Fig. 2 shows several training, validation and testing images in Shaoshan dataset. It should be note that we generate 256 × 256 training images with a part of overlapping, generating 14,580 images for training finally. In detail, we generate the training images having overlapping with neighbor images of 10 pixels in both row and column directions. We divide the original test images into 256 × 256 pieces without overlappings, obtaining 456 test images. We randomly select 4400 images from training images for validation during training. It should be note that the Shaoshan dataset only label the visible road areas and omit the road areas which are occluded, resulting to breakages of labeled roads in visual.

The third dataset used in our experiment is Massachusetts Road dataset (Mnih, 2013), which is a publicly open dataset for road extraction from remote sensing images. The Massachusetts Road dataset contains 1108 training images, 14 validation images and 49 test images. Each original image in Massachusetts is 1500 × 1500. In our experiment, we divide the original training, validation and test images into 256 × 256 without overlappings, generating 27,700 training images, 350 validation images and 1225 test images, respectively. Fig. 3 shows several sample image of training, validation and test images in Massachusetts Road dataset.

3.2. Model architecture

Fig. 4 shows the model architecture of our Adaboost-like multiple

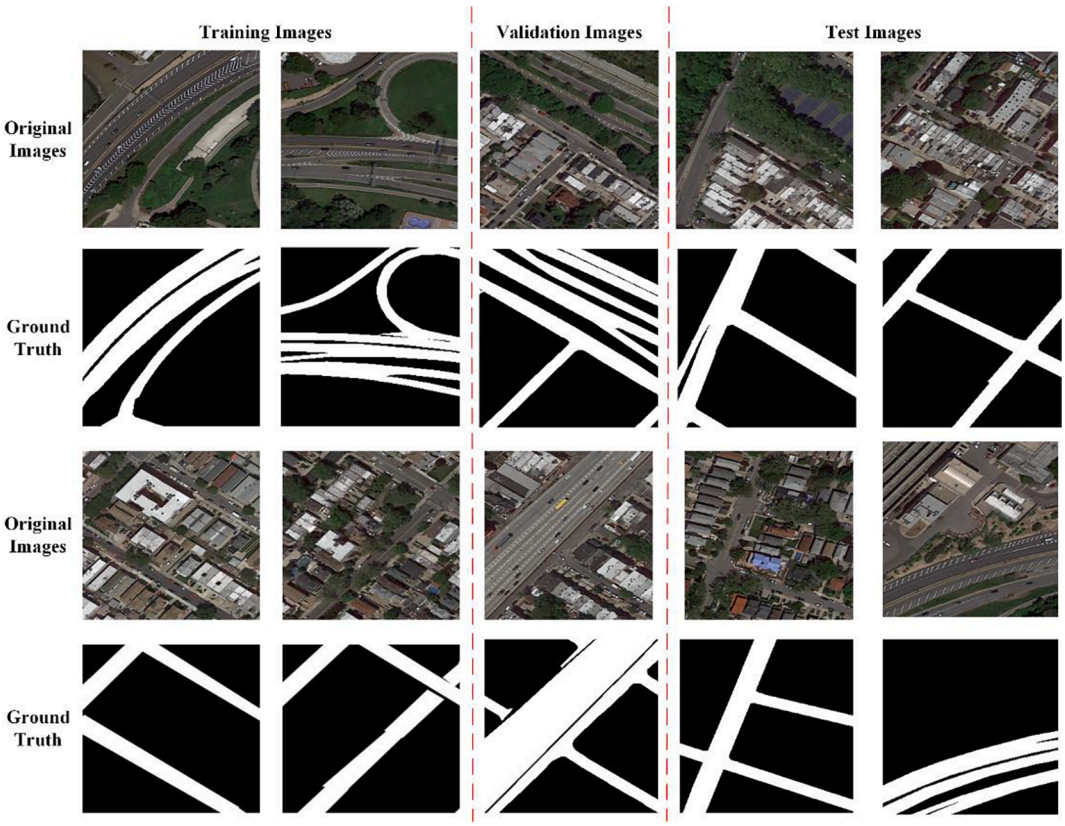


Fig. 1. Sample images of training, validation and test sets in LRSNY dataset.

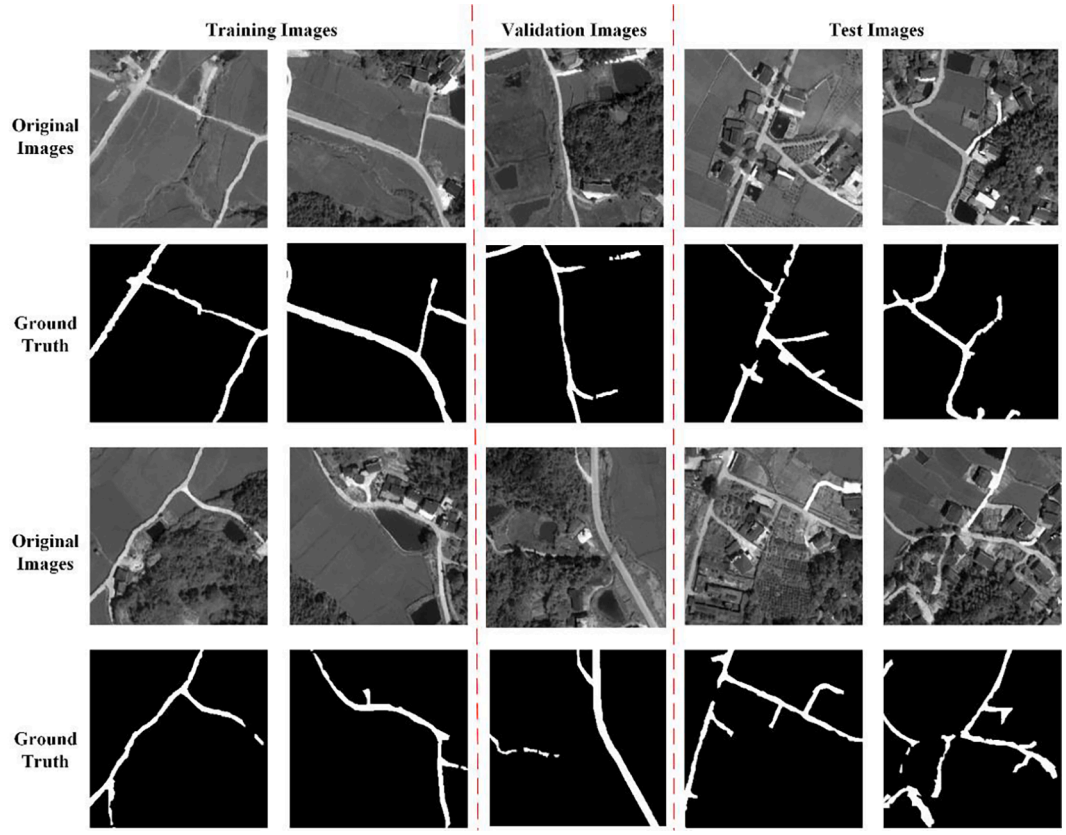


Fig. 2. Sample images of training, validation and test sets in Shaoshan dataset.

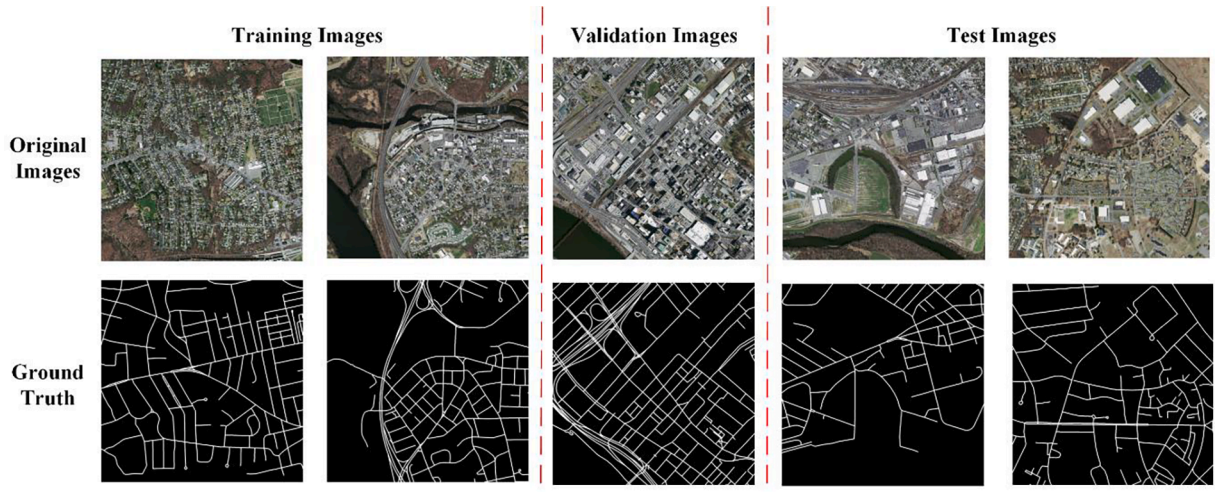


Fig. 3. Sample images of training, validation and test sets in Massachusetts Road dataset.

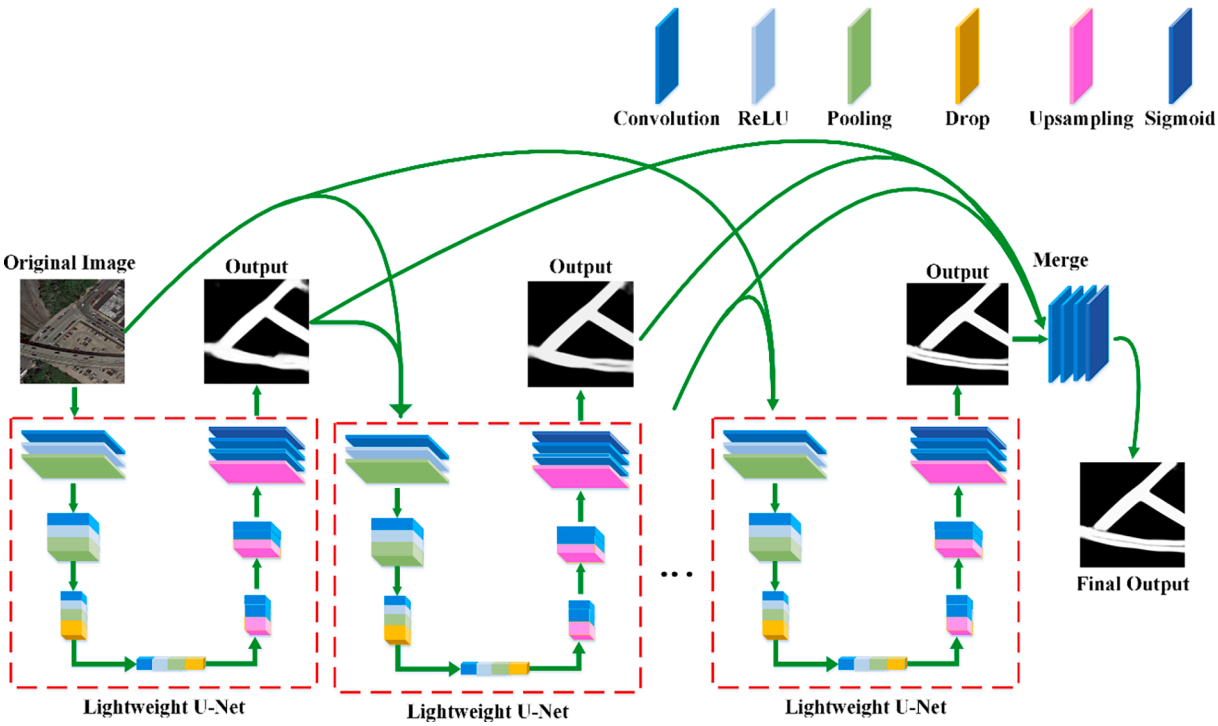


Fig. 4. The model architecture of our Adaboost-like multiple lightweight U-Nets. Our model contains multiple lightweight U-Nets. The output of prior U-Net is combined with initial input image as the input of next U-Net, except the first U-Net.

lightweight U-Nets for road extraction from remote sensing images. First, we design a small and lightweight U-Net. Then, we combine multiple lightweight U-Nets stage by stage. It should be note that all the U-Nets have the same network structure. To make next U-Net have strong relationship with prior U-Net, the output of prior U-Net is used as the input of next U-Net. Furthermore, to maintain the standalone ability of road extraction, the initial image is also used as the input of each U-Net. Thus, except first U-Net, the output of prior U-Net and the initial image are concatenated as the input of next U-Net.

To make our model as Adaboost-like, i.e. combining multiple weak classifiers together as a strong and better classifier, we merge all the outputs of U-Nets through concatenation, convolution and sigmoid operations. After the merge stage, we can get the final result. It should be note that we jointly train multiple U-Nets during training stage through multiple-objective learning method. Thus, our model is an end-to-end training model.

3.3. Detail structure of lightweight U-Net

Fig. 5 shows the detail model structure of lightweight U-Net used in our method. In our single lightweight U-Net, it is divided into two parts: encoding part and decoding part. The encoding part consists of four operation groups. In the first two operation groups, two convolutions, two ReLU and one pooling are used. In the last two operation groups, the drop out operation is added into each operation group. Other operations in the last two operation groups are just as same as the operations in the first two operation groups.

In the decoding part, there are four groups of upsampling, convolution and ReLU operations. After each upsampling operation group, a concatenation operation group is followed. In each concatenation group,

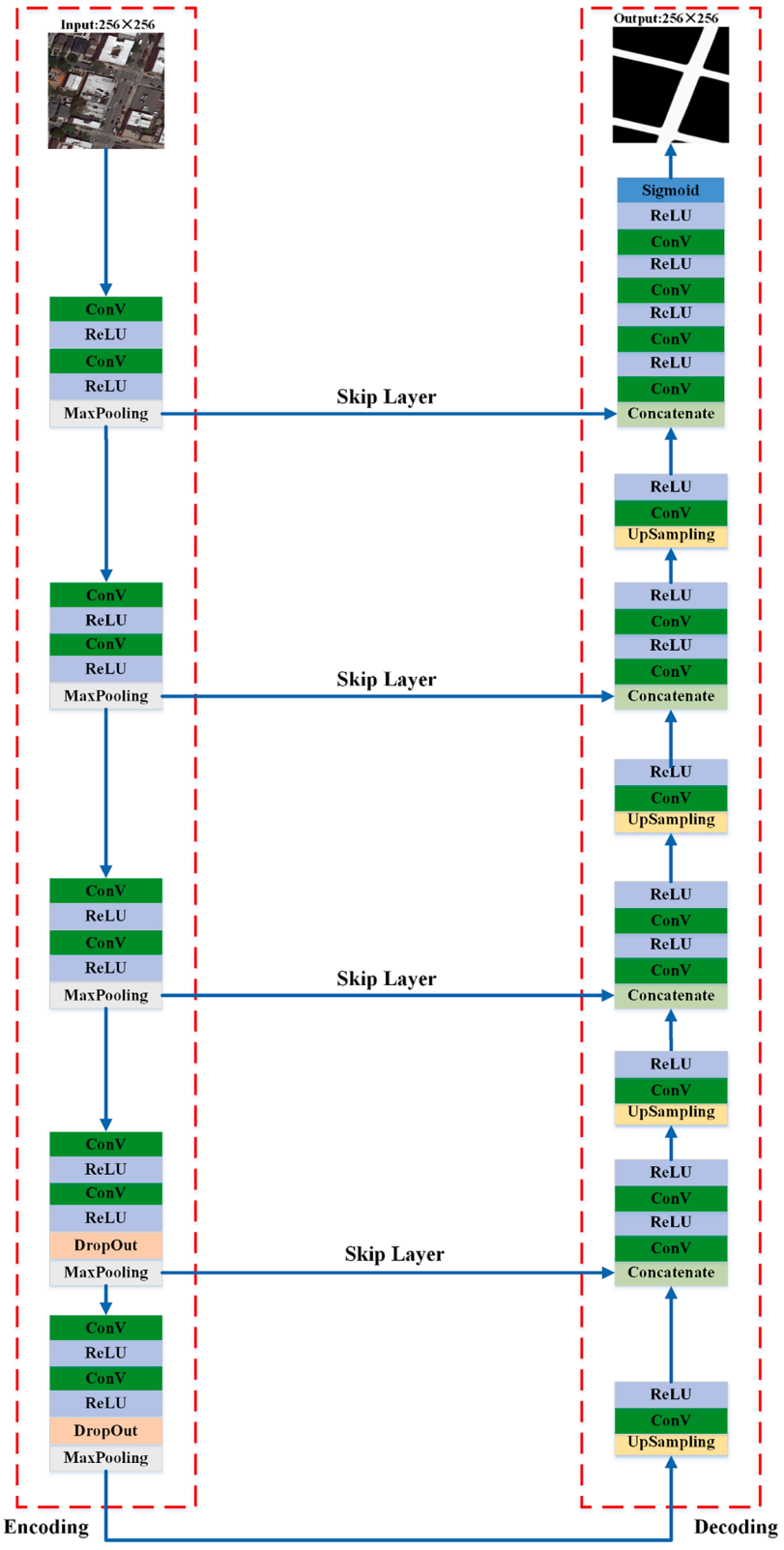


Fig. 5. The model structure of lightweight U-Net used in our method.

one concatenation layer, two convolutional layers and two ReLU layers are included. In the final concatenation group, a sigmoid layer is followed to obtain the final output results.

Table 1 shows the detail network parameters in our lightweight U-Nets. For the first U-Net, its input size is 256 × 256 × 3. Differently, the input size of other U-Nets is 256 × 256 × 6. In the encoding part, the convolutional kernel size is 3 × 3, the pooling kernel size is set as 2 × 2 and the dropout rate is set at 0.5. The convolutional kernel numbers for all the convolutional layers are 32, 32, 64, 64, 128, 128, 256, 256, 512 and 512, respectively. Thus, the outputs of four operation groups in the encoding part are 128 × 128 × 32, 64 × 64 × 64, 32 × 32 × 256, 16 × 16 × 512, respectively.

Table 1
The detail network architecture of lightweight U-Net.

	Operation couple	Convolution Filter	Stride	Output size
Input				256 × 256 × 3 (or 256 × 256 × 6)
Encoding	[Conv, ReLU, Conv, ReLU, MaxPooling] × 3	[3 × 3, 3 × 3, 2 × 2] × 3	[1, 1, 1] × 3	[128 × 128 × 32] [64 × 64 × 64] [32 × 32 × 128]
	[Conv, ReLU, Conv, ReLU, DropOut, MaxPooling] × 1	[3 × 3, 3 × 3, 0.5, 2 × 2] × 1	[1, 1] × 1	[16 × 16 × 256]
	[Conv, ReLU, Conv, ReLU, DropOut] × 1	[3 × 3, 3 × 3, 0.5] × 1	[1, 1] × 1	[16 × 16 × 512]
Decoding	[Up, Conv, ReLU, Concat, Conv, ReLU, Conv, ReLu] × 4	[2 × 2, 2 × 2, 2 × 2, 2 × 2] × 4	[1, 1] × 4	[32 × 32 × 256] [64 × 64 × 128] [128 × 128 × 64] [256 × 256 × 32]
	[Conv, Conv, Sigmoid]	[1 × 1, 1 × 1, 1 × 1]	[1, 1, 1]	[256 × 256 × 3]

In the decoding part, the convolutional kernel size is 2 × 2 for all the convolutional layers. Besides, the kernel size is also 2 × 2 for all the upsampling layers. For the final sigmoid layer, the kernel size is 1 × 1. The convolutional kernel numbers for all the convolutional layers are 256, 256, 128, 128, 64, 64, 32, 32, 3 and 3, respectively. Thus, the outputs for the four upsampling operation groups are 32 × 32 × 256, 64 × 64 × 128, 128 × 128 × 64, 256 × 256 × 32, respectively. The convolutional kernel numbers of the final two convolution layers are both 3. We use a similar structure of U-Net as (Ronneberger et al. 2015), the differences between our lightweight U-Net and the original U-Net are the filter number in different layers. Our filter number in each layer are much less than the original U-Net, thus we call our single U-Net as lightweight U-Net.

3.4. Multiple-objective learning

In our paper, we follow the multiple-objective learning in (Sener and Koltun 2018). For our multiple U-Nets, we consider each U-Net as one task, and denote training images and their corresponding labeling images as $(X_i, y_i), i \in M$, where M represents the number of training images. Multiple U-Nets can be considered as a multi-task learning problem over an input dataset X and a corresponding tasks' objective set $\{y_t\}, t \in T$, where T is the task number. Denote the $[x_i, y_{i1}, y_{i2}, \cdots, y_{iT}]$ as a couple of input image x_i and the corresponding T task labels $[y_{i1}, y_{i2}, \cdots, y_{iT}]$. For each task, we consider a parametric hypothesis mapping between X and y_t, which can be represented as:

$$f_t(X, W_t) : X \rightarrow y_t \tag{1}$$

where W_t is the mapping parameters. The loss function of task t can be represented as follow:

$$\mathscr{L}_t(X, W_t) = y_t - f_t(X, W_t) \tag{2}$$

where $\mathscr{L}_t(X, W_t)$ represents the loss value of task t over data$X$ with parameter W_t. For task t, the task objective function can be written as follow:

$$\min \mathscr{L}_t(X, W_t). \tag{3}$$

For all the T tasks' objective function, we want to minimize the total loss of T tasks. Thus, for a multiple-task learning with T tasks, the objective function can be written as follow:

$$\min \sum_{t=1}^{T} q^t \mathscr{L}_t(X, W_t), \tag{4}$$

where q^trepresents the weight of loss in task t.

Through the above functions, we can make our multiple-U-Nets' training as multiple-objective learning.

3.5. Adaboost-like U-Nets

In this paper, we follow the Adaboost in (Hu et al. 2008). Denote n as the number of classifiers which will be combined together using Adaboost. Then, an iteration with K steps will be applied for updating the weight of each classifier. For step k of tth classifier, the training error will be calculated, which can be denote as ε_{kt}. Then, according to ε_{kt}, the combination weight of tth classifier will be updated, which can be written as follow:

$$q^{kt} = \frac{1}{2} \ln\left(\frac{1 - \varepsilon_{kt}}{\varepsilon_{kt}}\right) \tag{5}$$

In our Adaboost-like U-Nets, we do not update the weights of each U-Net according to Eq. (5). Instead, we use convolution concatenation to automatically learn the combination weights for all the U-Nets, as shown in Fig. 6.

4. Results

In this section, we first introduce the details about implementations. Then, we introduce the evaluation criteria used in the experiments. Finally, we present and analyze the experimental results on the test datasets.

4.1. Experimental implementation detail

We train our model on a computer with Intel® Core™ i9-9900X 3.5 GHz and 128 GB memories. The computer has two GPUs, which type is RTX 2080 Ti with 11 GB GPU memories. During training and test, we use only one GPU. When training our model, we set the training epoch as 200 and the learning rate as 0.0001. Our training batch size is 2. After training, we save the model with minimum loss within 200 epochs.

Our implementation is based on Python, Tensorflow (Abadi et al. 2016) and Keras. To further strengthen the training stage and avoid the over fitting problem of model training, we utilize the data augmentation for training images. The rotation, zoom, shift, shear and flip operations are all used in our training data augmentation. In our experiment, the rotation range is set at 0.9, the width shift range and height shift range

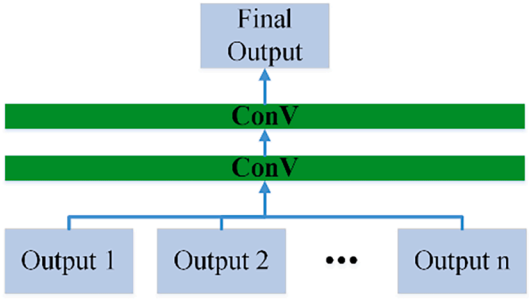


Fig. 6. Automatically learning for combination weights of multiple U-Nets.

are set at 0.1. And the shear range and zoom range are also set at 0.1.

4.2. Evaluation criteria

To comprehensively evaluate the performance of models, we use five evaluation criteria which are widely used for evaluating road segmentation performance. The first to fourth criteria are recall, precision, IoU and F-Score (Zang et al. 2017), the representations are as follows:

$$\begin{aligned} \text{recall} &= \frac{TP}{TP+FN} \\ \text{precision} &= \frac{TP}{TP+FP} \\ \text{IoU} &= \frac{TP}{TP+FN+FP} \\ \text{F - } Score &= \frac{2*precision*recall}{precision+recall}, \end{aligned} \tag{6}$$

where TP, FN and FP denote true positive, false negative and false positive, respectively.

The fifth evaluation criterion is PRI (Probabilistic Rand Index) (Hu et al. 2018), which can be computed as follow:

$$\text{PRI}\left(S_{seg}, S_{gt}\right) = \frac{1}{C_n^2}\sum_{i}\sum_{j(i\neq j)}[\psi(l_i = l_j \& l_i^{'} = l_j^{'}) + \psi(l_i \neq l_j \& l_i^{'} \neq l_j^{'})], \tag{7}$$

where ψ is a discrimination function, l_i and l_j are the labels of S_{seg}, $l_i^{'}$ and $l_j^{'}$ are the labels of S_{gt}, c_n is the total pixel numbers of S_{seg}.

4.3. Experimental results and analysis

4.3.1. Experimental results on LRSNY dataset

In this section, we first exhibit the comparisons among our model and other seven state-of-the-art semantic segmentation methods on LRSNY and Shaoshan datasets. The seven state-of-the-art semantic segmentation methods include the original U-Net(Ronneberger et al. 2015), SegNet (Badrinarayanan et al. 2017), PSPNet-50 (Zhao et al. 2017), Residual U-Net(Zhang et al. 2018), DeepLabV3(Chen et al. 2018), DANet (Fu et al. 2019) and PSPNet-101 (Zhao et al. 2017). For each compared method, we obtain the source code from the original author or from the the GitHub. During model training, the training settings are just as same as the settings in our model training.

Table 2 shows the Recall, Precision, IoU and F-Score comparison results among our AEML U-Nets and other seven state-of-the-art semantic segmentation methods tested on LRSNY dataset. In this experiment, we use the results obtained by 3 U-Nets for comparison, which achieves 0.88215 in IoU score. For other compared methods, the IoU scores are about 0.838, 0.856, 0.865, 0.827, 0.85, 0.86 and 0.871, respectively. It is obviously that our model achieves best IoU score in the experiment, which is about 5.25%, 3.1%, 1.98%, 6.6%, 3.8%, 2.6% and 1.3% higher than other seven state-of-the-art methods, respectively. In F-Score, our result also achieves the highest score among all the compared methods, which is about 0.938. The performance convincingly demonstrate the good performance of our AEML U-Nets. Besides, Table 2 also shows the model parameter numbers for all the models. Our AEML U-Nets only uses about 20.9 $\times$ 10^6 parameters to obtain better performance than PSPNet-101 which has about 65.7 $\times$ 10^6 parameters. Except SegNet and Residual U-Net, all the parameter scales of other models are much larger than the parameter scale of our model. The parameter scales prove the effectiveness of our model from another aspect, as our model uses less parameters than most of other compared models while obtaining better performance. The SegNet and Residual U-Net seem to obtain worse performance due to their much smaller parameter scales.

Table 2
The comparison results among our method and other seven state-of-the-art segmentation methods tested on the LRSNY dataset.

Method	Recall	Precision	IoU	F-Score	Parameters (10^6)
U-Net	0.8836	0.9582	0.8379	0.9118	31
SegNet	0.91233	0.93219	0.85555	0.92215	0.93
PSPNet-50	0.91221	0.94351	0.86497	0.9276	46.77
Residual U-Net	0.90218	0.90899	0.82744	0.90558	4.36
DeepLabV3	0.90588	0.9323	0.84996	0.9189	41.25
DANet	0.90504	0.94521	0.85993	0.92469	71.4
PSPNet-101	0.9291	0.9327	0.87073	0.9309	65.7
AEML U-Nets (3 U-Nets)	0.94069	0.93411	**0.88215**	**0.93739**	20.9

Fig. 7 shows the PRI comparison among our AEML U-Nets and other seven state-of-the-art methods tested on LRSNY dataset. The original U-Net, SegNet, PSPNet-50, Residual U-Net, DeepLabV3, DANet, PSPNet-101 and our AEML U-Nets obtain the PRI values of 0.941, 0.9548, 0.95816, 0.9455, 0.9531, 0.9565, 0.9593 and 0.9628, respectively. Our AEML U-Nets improves the performance about 0.0218, 0.008, 0.0464, 0.0173, 0.0097, 0.0063, 0035 PRI scores compared with other methods. It is obvious that AEML U-Nets obtains highest PRI score among all the compared methods. The experimental results further prove the superior performance of our AEML U-Nets. Fig. 8 exhibits several examples of visual test results of AEML U-Nets and other seven compared methods. The 1–10 columns are the original images, ground truth, visual results of original U-Net, PSPNet-50, SegNet, Residual U-Net, DeepLabV3, DANet, PSPNet-101 and AEML U-Nets, respectively. From Fig. 8 we can see that our AEML U-Nets obtains better visual results in the tested example images. The results of our method are more consistent, smooth and possess better road properties in vision.

4.3.2. Experimental results on Shaoshan dataset

To further verify the superior performance of our AEML U-Nets, we also compare AEML U-Nets with other methods on Shaoshan dataset. The compared methods include (Zhang et al. 2017, Chen et al., 2020a, 2020b, Zang et al. 2017, Zhao et al. 2017, Sachin et al. 2018). As the results of other methods have shown in our prior paper (Chen et al., 2020a, 2020b), thus we just follow the results in (Chen et al., 2020a, 2020b). Table 3 shows the comparison results, from which we can see our AEML U-Nets obtains better performance than other compared methods. The IoU scores of Zang et al., ResidualUnet, PSPNet, ESPNet and Chen et al are 0.5963, 0.6970, 0.6615, 0.6795, 0.7159, respectively. Our AEML U-Nets can achieve as high as 0.75085 score in IoU score, which is about 4% higher than our prior method. And the 0.75085 IoU score is also much higher than other four methods, which is about 25.8%, 7.6%, 13.4%, 10.5% higher than the results of Zang, Residual U-Net, PSPNet and ESPNet, respectively. The experimental results prove the satisfactory performance of our method once again.

Fig. 9 shows several road extraction visual results on Shaoshan dataset by our method. The first, second and third rows are the original images, ground truth and road extraction results by our method, respectively. It can be seen that our method obtains quite good visual road extraction results on Shaoshan dataset.

4.3.3. Experimental results on Massachusetts road dataset

In the third experiment, we verify the performance of our AEML U-Nets on Massachusetts Road dataset. In this experiment, we also compared with other methods including the original U-Net(Ronneberger et al. 2015), SegNet (Badrinarayanan et al. 2017), PSPNet-50 (Zhao et al. 2017), Residual U-Net(Zhang et al. 2018), DeepLabV3(Chen et al. 2018), DANet (Fu et al. 2019) and PSPNet-101 (Zhao et al. 2017). The parameter settings for training models have been illustrated in our prior section. As we finally has 27,700 training images, we set our steps per epoch as 27700, epochs as 20 and validation step as 350. Table 4 shows the comparison results among our method and other seven state-of-the-art segmentation methods tested on the Massachusetts Road dataset.

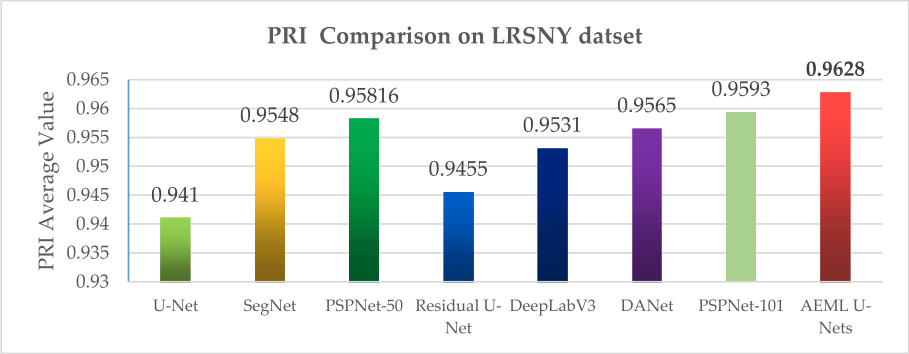


Fig. 7. The PRI comparison among AEML U-Nets and other seven state-of-the-art methods on LRSNY dataset.

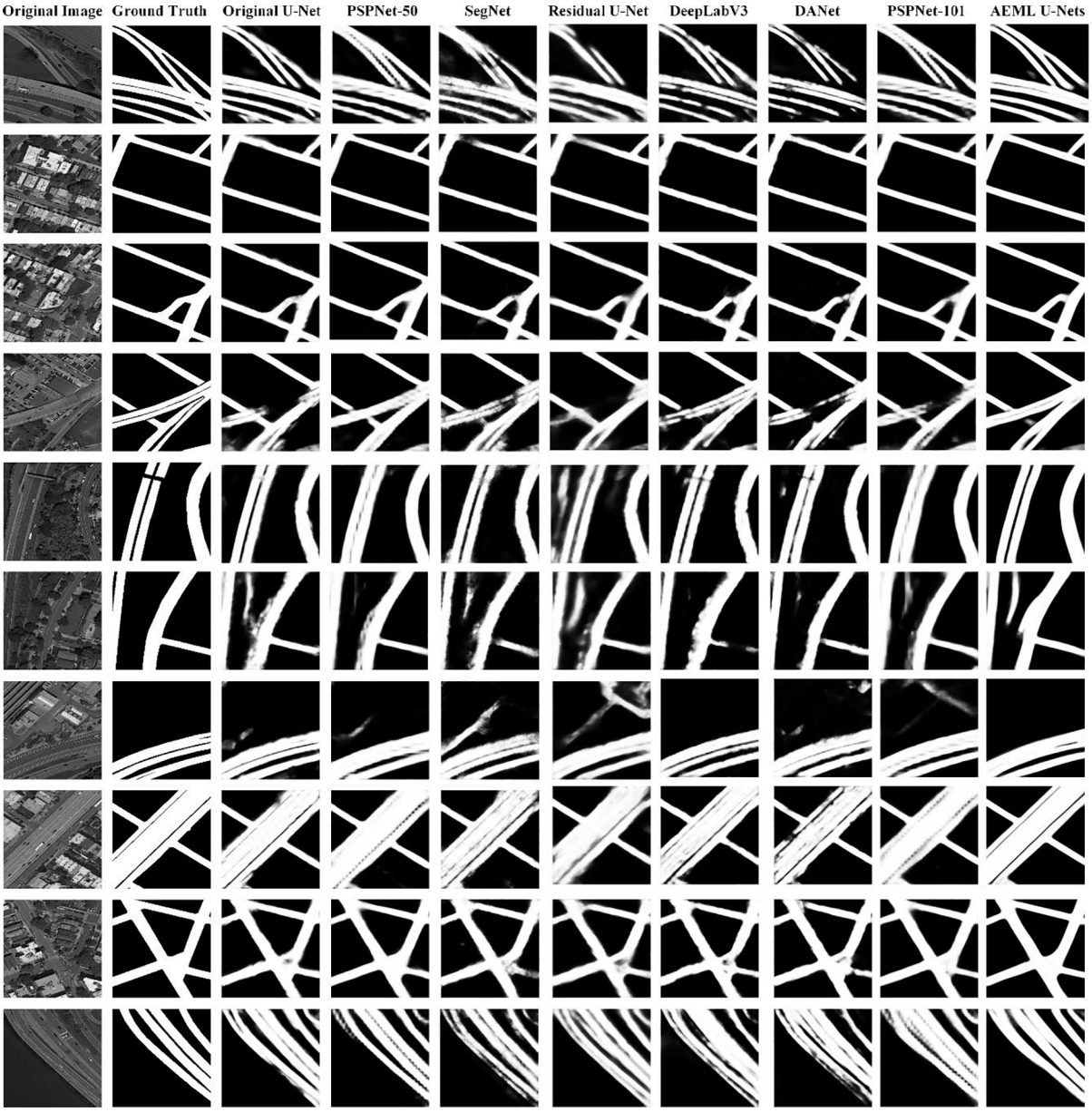


Fig. 8. The examples of visual comparison results on LRSNY dataset among AEML U-Nets and other seven methods. The 1–10 columns are the original images, ground truth, results of original U-Net, PSPNet-50, SegNet, Residual U-Net, DeepLabV3, DANet, PSPNet-101 and AEML U-Nets, respectively.

Table 3
The comparison among (Zang et al. 2017), (Zhang et al. 2017), (Zhao et al. 2017), (Sachin et al. 2018) (Chen et al., 2020a, 2020b) and our method on Shaoshan dataset.

Method	Precision	Recall	IoU
Zang et al. (Zang et al. 2017)	0.7786	0.7135	0.5963
ResidualUnet(Zhang et al. 2017)	0.7454	0.9149	0.6970
PSPNet(Zhao et al. 2017)	0.6888	0.9434	0.6615
ESPNet(Sachin et al. 2018)	0.7431	0.8882	0.6795
Chen et al. (Chen et al., 2020a, 2020b)	0.8247	0.8443	0.7159
AEML U-Nets (3 U-Nets)	0.86493	0.88841	**0.75085**

Our method and other seven methods achieve IoU scores of 0.64779, 0.59888, 0.62477, 0.6271, 0.64271, 0.6141, 0.6334 and 0.62297, respectively. Our method achieves the highest IoU score among all the compared methods on Massachusetts Road dataset. For detail, our method promote the IoU score about 8.1%, 3.7%, 3.3%, 0.8%, 5.5%, 2.3% and 4% compared with the original U-Net, SegNet, PSPNet-50, Residual U-Net, DeepLabV3, DANet and PSPNet-101, respectively. For F-Score, our method achieves as high as 0.78625, which is the highest score among all the compared methods. The experimental result proves the superior performance of our method compared with the seven state-of-the-art methods.

Fig. 10 shows several road extraction visual results on Massachusetts Road dataset. The 1–10 columns are the original images, ground truth, results of original U-Net, PSPNet-50, SegNet, Residual U-Net, DeepLabV3, DANet, PSPNet-101 and AEML U-Nets, respectively. In Fig. 10, the results of our method shows better results compared with other method in visual. The results of our method obtain clearer results in the extraction details, which proves the satisfactory results of our method.

5. Discussion

5.1. Analysis about multiple U-Net strategy

To verify the effectiveness of multiple U-Nets strategy in our model, we examine the performance of our model with different lightweight U-Net number, ranging from 1 to 3. We test on LRSNY dataset. Table 5 shows the comparison results among our method with 1, 2 and 3 lightweight U-Nets tested on LRSNY dataset. In Table 5, the IoU scores of AEML U-Nets with 1 U-Net, 2 U-Nets and 3 U-Nets achieve 0.8626, 0.88071 and 0.88215, respectively. From Table 5 we can see that the IoU performance is increased when the number of lightweight U-Net structure is increased. This phenomenon shows the effectiveness of our multiple lightweight strategy. It should be noted that we do not use 4 or more lightweight U-Nets as the performance seems not improved, on which our future work will focus.

Table 4
The comparison results among our method and other seven state-of-the-art segmentation methods tested on the Massachusetts Road dataset.

Method	Recall	Precision	IoU	F-Score
U-Net	0.73964	0.75886	0.59888	0.74913
SegNet	0.72053	**0.82459**	0.62477	0.76905
PSPNet-50	0.76261	0.77921	0.6271	0.77082
Residual U-Net	**0.79688**	0.76862	0.64271	0.7825
DeepLabV3	0.73984	0.78322	0.6141	0.76092
DANet	0.74218	0.81209	0.6334	0.77556
PSPNet-101	0.72838	0.81149	0.62297	0.76769
AEML U-Nets (3 U-Nets)	0.76332	0.81061	**0.64779**	**0.78625**

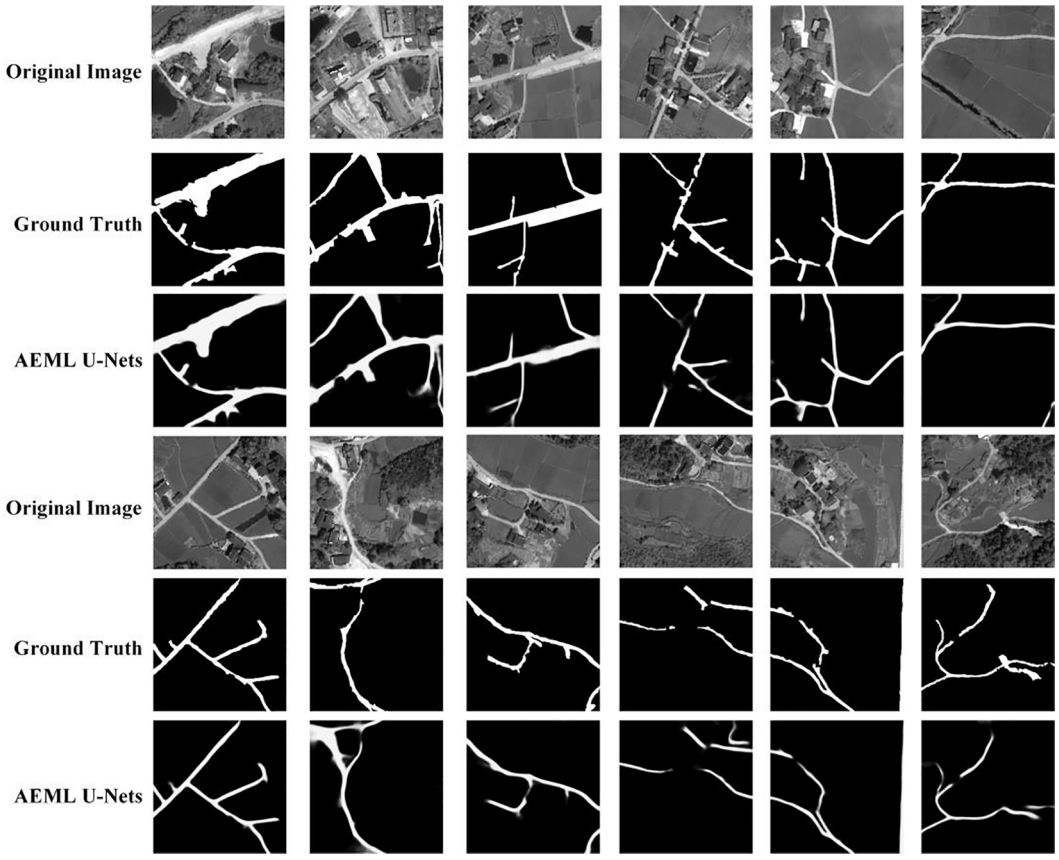


Fig. 9. Visual road extraction results in Shaoshan dataset by our AEML U-Nets. The first row, second row and third row are the original images, ground truth and road extraction results by AEML U-Nets, respectively.

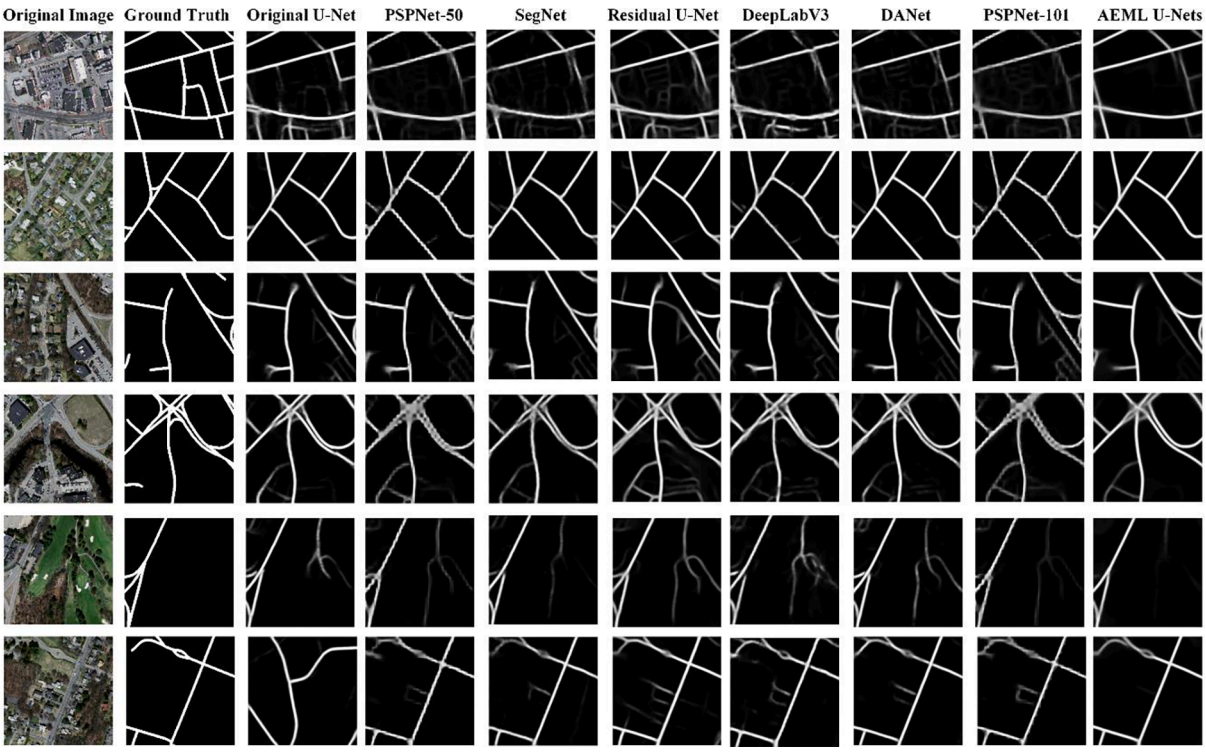


Fig. 10. The examples of visual comparison results on Massachusetts Road dataset among AEML U-Nets and other seven methods. The 1–10 columns are the original images, ground truth, results of original U-Net, PSPNet-50, SegNet, Residual U-Net, DeepLabV3, DANet, PSPNet-101 and AEML U-Nets, respectively.

Table 5
The comparison among our method using 1, 2 and 3 U-Nets on LRSNY dataset.

Method	Recall	Precision	IoU
AEML U-Nets (1 U-Net)	0.94008	0.91279	0.8626
AEML U-Nets (2 U-Nets)	0.93506	0.93809	0.88071
AEML U-Nets (3 U-Nets)	0.94069	0.93411	**0.88215**

5.2. Analysis about Adaboost-like combination strategy

To verify the effectiveness of Adaboost-like strategy, we also compare the outputs of multiple lightweight U-Nets. In this experiment, we also use our model with three multiple lightweight U-Nets. Thus, there will be four output results, i.e. outputs of first U-Net, second U-Net, third U-Net and final combination. Besides, the experiment is implemented on LRSNY dataset. Table 6 shows the comparison results. The four outputs obtain the IoU scores of 0.85904, 0.87662, 0.87772 and 0.88215, respectively. It is obviously that the output of third U-Net is better than the output of second U-Net, and the output of second U-Net is better than the output of first U-Net. The output of final combination result is the best among all the outputs, which proves the effectiveness of our Adaboost-like strategy combination of multiple lightweight U-Nets. Due to combinations through convolutional layers, the combination weights can be learned jointly through the training of whole model.

Table 6
The comparison among outputs of first U-Net, second U-Net, third U-Net and final combination tested on LRSNY dataset.

Method	Recall	Precision	IoU
First U-Net	0.93897	0.90984	0.85904
Second U-Net	0.93877	0.92978	0.87662
Third U-Net	0.94123	0.92861	0.87772
AEML U-Nets (3 U-Nets)	0.94069	0.93411	**0.88215**

5.3. Analysis about false extractions

From the IoU analysis of the above experiments, we know that our approach still fails to extract right road areas under some situations. To make an investigation of wrong extractions, we visualize our extractions through labeling wrong extractions with red and right extractions with green on the original images. The right extracted road areas will be paint with green color and the wrong extracted road areas will be paint with red color. Fig. 11 shows several serious wrong extraction examples of our method on LRSNY dataset. From Fig. 11, we can see that the serious wrong extractions majorly occur in areas which are occluded by trees or the road areas near a car parking lot. The reason for wrong extractions of areas with serious occlusion may be that the situation is out of the limition of logical reasoning ability of our model. For the wrong extractions near a car parking lot, we think the reason may be there exists negative training samples like roads in road parking lots. The classification ability about road areas in or out a car parking lot is still need to be improved.

5.4. Analysis about stability of our method

In this section, we analysis the stability of our method on LRSNY dataset. To verify the stability of our method's performance, we repeat training our model five times and compute the average IoU performance and standard deviation. For comparison, we also repeat training and test five times of other six state-of-the-art methods on LRSNY. Fig. 12 shows the average IoU performance comparisons of different methods tested on LRSNY dataset. In Fig. 12, our method obtains average IoU score of 0.871162, which is the highest average IoU score among all the compared methods. For the original U-Net, SegNet, PSPNet-50, Residual U-Net, DeepLabV3, DANet and PSPNet-101, they achieve average IoU scores of 0.844078, 0.857044, 0.867278, 0.838208, 0.8605675, 0.859528, and 0.869118, respectively. Besides, our model also achieves highest score in a single training epoch, which is 0.88215. The PSPNet and DANet seems more stable as their standard deviations are small.

Fig. 11. Several wrong extraction examples of our method on LRSNY dataset. Green represents the right extraction and Red represents the wrong extraction. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

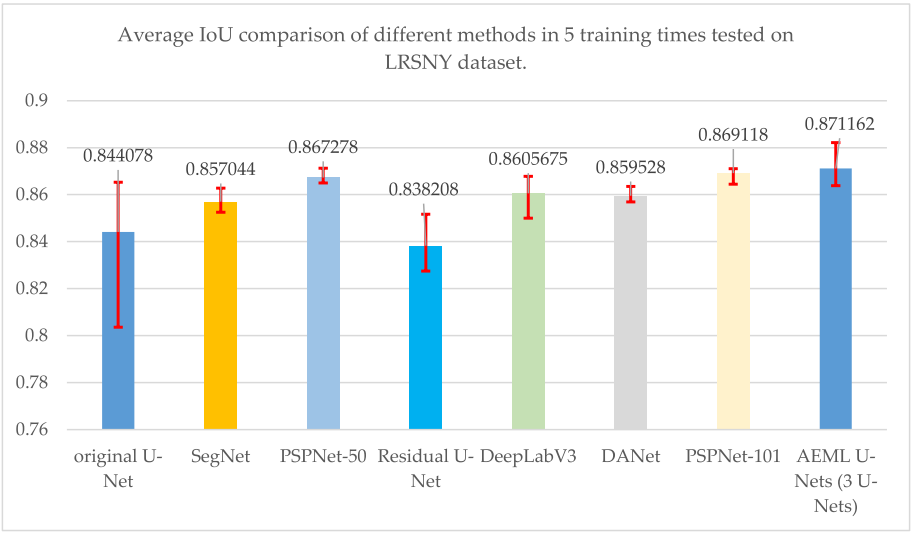


Fig. 12. Average IoU performance comparison of different methods tested on LRSNY dataset.

However, our average IoU is higher than PSPNet and DANet. The worst performance of our method in five training times is still good, which IoU score is as high as 0.86379. From this experiment, we prove that our method can achieve better performance than other six state-of-the-art methods tested on LRSNY. The performance stability of our method is satisfactory.

6. Conclusions

This paper proposed an Adaboost-like multiple lightweight U-Nets for road extraction from optical remote sensing images. To enhance the model's segmentation ability, we used multiple lightweight U-Nets. To solve the combination of multiple U-Nets, we proposed Adaboost-like strategy combination of multiple U-Nets. Finally, we used multiple-objective learning strategy to jointly train the multiple lightweight U-Nets. We tested our model on three datasets: LRSNY, Shaoshan and Massachusetts. The quantitative analysis and visual exhibitions all proved that our method can achieve state-of-the-art performance. Compared with other state-of-the-art methods, our model obtained better performances. In LRSNY dataset, our model achieved IoU score as high as about 0.8825. In Shaoshan dataset, our model achieved IoU score as high as about 0.75. In Massachusetts Road dataset, our model achieved IoU score as high as 0.6477. Finally, we also proved the effectiveness of our multiple U-Nets strategy and Adaboost-like joint strategy.

In our future work, we will focus on the combination of multiple kinds of models with Adaboost-like strategy and multi-objective learning.

CRediT authorship contribution statement

Ziyi Chen: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing - original draft, Writing - review & editing, Visualization, Project administration, Funding acquisition. **Cheng Wang:** Conceptualization. **Jonathan Li:** Investigation, Supervision. **Wentao Fan:** Formal analysis, Writing - review & editing. **Jixiang Du:** Resources. **Bineng Zhong:** Formal analysis, Writing - review & editing, Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This study was financially supported by National Natural Science Foundation of China (No. 62001175), Natural Science Foundation of Fujian Province (No.2019J01081), United National Natural Science Foundation of China (No.U1605254), National Natural Science Foundation of China (No. 6187606, 61972167 and 61673186), and the Special National Key Research and Development Plan (No. 2019YFC1604705).

References

Liu, Y., Yao, J., Lu, X., Xia, M., Wang, X., Liu, Y., 2018. Roadnet: Learning to comprehensively analyze road networks in complex urban scenes from high-resolution remotely sensed images. IEEE Trans. Geosci. Remote Sens. 57 (4), 2043–2056. https://doi.org/10.1109/TGRS.2018.2870871.

Guo, Q., Wang, Z., 2020. A Self-Supervised Learning Framework for Road Centerline Extraction From High-Resolution Remote Sensing Images. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 13, 4451–4461. https://doi.org/10.1109/JSTARS.2020.3014242.

Tao, C., Qi, J., Li, Y., Wang, H., Li, H., 2019. Spatial information inference net: Road extraction using road-specific contextual information. ISPRS J. Photogramm. Remote Sens. 158, 155–166. https://doi.org/10.1109/IGARSS.2019.8900507.

Lu, X., et al., 2019. Multi-scale and multi-task deep learning framework for automatic road extraction. IEEE Trans. Geosci. Remote Sens. 57 (11), 9362–9377. https://doi.org/10.1109/TGRS.2019.2926397.

Gao, L., Song, W., Dai, J., Chen, Y., 2019. Road extraction from high-resolution remote sensing imagery using refined deep residual convolutional neural network. Remote Sens. 11 (5), 552. https://doi.org/10.3390/rs11050552.

Liu, R., et al., 2019. Multiscale road centerlines extraction from high-resolution aerial imagery. Neurocomputing 329, 384–396. https://doi.org/10.1016/j.neucom.2018.10.036.

Abdollahi, A., Pradhan, B., Alamri, A., 2020. VNet: An End-to-End Fully Convolutional Neural Network for Road Extraction from High-Resolution Remote Sensing Data. IEEE Access 8, 179424–179436. https://doi.org/10.1109/ACCESS.2020.3026658.

Yang, X., Li, X., Ye, Y., Lau, R.Y., Zhang, X., Huang, X., 2019. Road detection and centerline extraction via deep recurrent convolutional neural network u-net. IEEE Trans. Geosci. Remote Sens. 57 (9), 7209–7220. https://doi.org/10.1109/TGRS.2019.2912301.

Xie, L., Wang, J., Wei, Z., Wang, M., Tian, Q., 2016. "Disturblabel: Regularizing cnn on the loss layer," Paper presented at the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA, June, p. 2016.

Chen Z. LRSNY [Online] Available: ftp://154.85.52.76/LRSNY/.

Mnih V. Machine Learning for Aerial Image Labeling [Online] Available: http://www.cs.toronto.edu/~vmnih/data/.

Li, X., Lei Wang, E.S., 2008, AdaBoost with SVM-based component classifiers, Eng. Appl. Artif. Intell., 21, 5, 785–795. http://doi.org/10.1016/j.engappai.2007.07.001.

Zhu, J., Arbor, A., Hastie, T., 2006. Multi-class AdaBoost. Stats & Its Interface 2 (3), 349–360. https://doi.org/10.4310/SII.2009.V2.N3.A8.

Hu, W., Hu, W., Maybank, S., 2008. AdaBoost-Based Algorithm for Network Intrusion Detection. IEEE Trans. Cybern. 38 (2), 577–583. https://doi.org/10.1109/TSMCB.2007.914695.

Zhang, C.X., Zhang, J.S., 2008. RotBoost: A technique for combining Rotation Forest and AdaBoost. Pattern Recogn. Lett. 29 (10), 1524–1536. https://doi.org/10.1016/j.patrec.2008.03.006.

Lv, F., Nevatia, R., 2006. Recognition and Segmentation of 3-D Human Action Using HMM and Multi-class AdaBoost. Paper presented at the ECCV 2006: Computer Vision Graz, Austria, May, 2006.

Viola, P., Jones, M., 2001. Fast and Robust Classification using Asymmetric AdaBoost and a Detector Cascade. Paper presented at the Advances in Neural Information Processing Systems, 14 (NIPS 2001).

Zhang, P.-B., Yang, Z., 2018. A Novel AdaBoost Framework With Robust Threshold and Structural Optimization. IEEE Trans. Cybern. 48, 64–76. https://doi.org/10.1109/TCYB.2016.2623900.

Zhang, F., Wang, Y., Ni, J., Zhou, Y., Hu, W., 2020. SAR Target Small Sample Recognition Based on CNN Cascaded Features and AdaBoost Rotation Forest. IEEE Geosci. Remote Sens. Lett. 17, 1008–1012. https://doi.org/10.1109/LGRS.2019.2939156.

Chen, L., Li, M., Su, W., Wu, M., Hirota, K., Pedrycz, W., 2021. Adaptive Feature Selection-Based AdaBoost-KNN With Direct Optimization for Dynamic Emotion Recognition in HumaŽ Robot Interaction. IEEE Trans. Emerg. Top. Comput. Intell. 5, 205–213. https://doi.org/10.1109/TETCI.2019.2909930.

Caruana, R., 1997. Multitask learning. Mach. Learn. 28 (1), 41–75. https://doi.org/10.1023/A:1007379606734.

Zhou, J., Chen, J., Ye, J., 2011. Clustered multi-task learning via alternating structure optimization. Paper presented at the Advances in neural information processing systems 24, 2011.

Rosenbaum, C., Klinger, T., Riemer, M., 2017. Routing networks: Adaptive selection of non-linear functions for multi-task learning. Paper presented at the ICLR 2018, Vancouver, BC, Canada, May, 2018.

Rudd, E.M., Günther, M., Boult, T.E., 2016. Moon: A mixed objective optimization network for the recognition of facial attributes. Paper presented at the Computer Vision – ECCV 2016 Amsterdam, The Netherlands, October, 2016.

Misra, I., Shrivastava, A., Gupta, A., Hebert, M., 2016. Cross-stitch networks for multi-task learning. Paper presented at the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, June 2016.

Liu, A., Su, Y., Nie, W., Kankanhalli, M., 2017. Hierarchical Clustering Multi-Task Learning for Joint Human Action Grouping and Recognition. IEEE Trans. Pattern Anal. Mach. Intell. 39, 102–114. https://doi.org/10.1109/TPAMI.2016.2537337.

Shen, W., Zhao, K., Jiang, Y., Wang, Y., Bai, X., Yuille, A., 2017. DeepSkeleton: Learning Multi-Task Scale-Associated Deep Side Outputs for Object Skeleton Extraction in Natural Images. IEEE Trans. Image Process. 26, 5298–5311. https://doi.org/10.1109/TIP.2017.2735182.

Volpia, M., Tuiab, D., 2018. Deep multi-task learning for a geographically-regularized semantic segmentation of aerial images. ISPRS J. Photogramm. Remote Sens. 144, 48–60. https://doi.org/10.1016/j.isprsjprs.2018.06.007.

Ranjan, R., Patel, V.M., Chellappa, R., 2017. Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. IEEE Trans. Pattern Anal. Mach. Intell. 41 (1), 121–135. https://doi.org/10.1109/TPAMI.2017.2781233.

Rad, M.S., et al., 2020. Benefiting from multitask learning to improve single image super-resolution. Neurocomputing 398, 304–313. https://doi.org/10.1016/J.NEUCOM.2019.07.107.

Liu, H., Li, Q., Gu, Y., 2020. A multi-task learning framework for gas detection and concentration estimation. Neurocomputing 416, 28–37. https://doi.org/10.1016/j.neucom.2020.01.051.

Cheng, G., Wang, Y., Xu, S., Wang, H., Xiang, S., Pan, C., 2017a. Automatic Road Detection and Centerline Extraction via Cascaded End-to-End Convolutional Neural Network. IEEE Trans. Geosci. Remote Sens. 55 (6), 3322–3337. https://doi.org/10.1109/TGRS.2017.2669341.

Zhang, Z., Liu, Q., Wang, Y., 2017. Road Extraction by Deep Residual U-Net. IEEE Geosci. Remote Sens. Lett. 15 (5), 749–753. https://doi.org/10.1109/LGRS.2018.2802944.

Maboudi, M., Amini, J., Malihi, S., Hahn, M., 2018. Integrating fuzzy object based image analysis and ant colony optimization for road extraction from remotely sensed images. ISPRS J. Photogramm. Remote Sens. 138, 151–163. https://doi.org/10.1016/J.ISPRSJPRS.2017.11.014.

Sghaier, M.O., Lepage, R., 2016. Road Extraction From Very High Resolution Remote Sensing Optical Images Based on Texture Analysis and Beamlet Transform. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 9 (5), 1946–1958. https://doi.org/10.1109/JSTARS.2015.2449296.

Alshehhi, R., Marpu, P.R., Wei, L.W., Mura, M.D., 2017. Simultaneous extraction of roads and buildings in remote sensing imagery with convolutional neural networks. ISPRS J. Photogramm. Remote Sens. 130, 139–149. https://doi.org/10.1016/J.ISPRSJPRS.2017.05.002.

Coulibaly, I., Spiric, N., Lepage, R., St-Jacques, M., 2017. Semiautomatic Road Extraction From VHR Images Based on Multiscale and Spectral Angle in Case of Earthquake. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 11 (1), 238–248. https://doi.org/10.1109/JSTARS.2017.2760282.

Lv, Z., Jia, Y., Zhang, Q., Chen, Y., 2017. An Adaptive Multifeature Sparsity-Based Model for Semiautomatic Road Extraction From High-Resolution Satellite Images in Urban Areas. IEEE Geosci. Remote Sens. Lett. 14 (8), 1238–1242. https://doi.org/10.1109/LGRS.2017.2704120.

Li, M., Stein, A., Bijker, W., Zhan, Q., 2016. Region-based urban road extraction from VHR satellite images using Binary Partition Tree. Int. J. Appl. Earth Obs. Geoinf. 44, 217–225. https://doi.org/10.1016/j.jag.2015.09.005.

Yin, D., Du, S., Wang, S., Guo, Z., 2016. A Direction-Guided Ant Colony Optimization Method for Extraction of Urban Road Information From Very-High-Resolution Images. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 8 (10), 4785–4794. https://doi.org/10.1109/JSTARS.2015.2477097.

Poullis, C., 2014. Tensor-Cuts: A simultaneous multi-type feature extractor and classifier and its application to road extraction from satellite images. ISPRS J. Photogramm. Remote Sens. 95 (95), 93–108. https://doi.org/10.1016/J.ISPRSJPRS.2014.06.006.

Chen, Z., Fan, W., Zhong, B., Li, J., Du, J., Wang, C., 2020a. Corse-to-fine road extraction based on local Dirichlet mixture models and multiscale-high-order deep learning. IEEE Trans. Intell. Transp. Syst. 21 (10), 4283–4293. https://doi.org/10.1109/TITS.2019.2939536.

Ren, Y., Yu, Y., Guan, H., 2020. DA-CapsUNet: A Dual-Attention Capsule U-Net for Road Extraction from Remote Sensing Imagery. Remote Sensing 12 (18), 2866. https://doi.org/10.3390/rs12182866.

Cheng, G., Zhu, F., Xiang, S., Pan, C., 2017b. Road Centerline Extraction via Semisupervised Segmentation and Multidirection Nonmaximum Suppression. IEEE Geosci. Remote Sens. Lett. 13 (4), 545–549. https://doi.org/10.1109/LGRS.2016.2524025.

Zang, Y., Wang, C., Yu, Y., Luo, L., Yang, K., Li, J., 2017. Joint Enhancing Filtering for Road Network Extraction. IEEE Trans. Geosci. Remote Sens. 55 (3), 1511–1525. https://doi.org/10.1109/TGRS.2016.2626378.

Cheng, G., Zhu, F., Xiang, S., Wang, Y., Pan, C., 2016. Accurate urban road centerline extraction from VHR imagery via multiscale segmentation and tensor voting. Neurocomputing vol. 205, no. C, 407–420. https://doi.org/10.1016/j.neucom.2016.04.026.

Zang, Y., Wang, C., Cao, L., Yu, Y., Li, J., 2016. Road Network Extraction via Aperiodic Directional Structure Measurement. IEEE Trans. Geosci. Remote Sens. 54 (6), 3322–3335. https://doi.org/10.1109/TGRS.2016.2514602.

Hui, Z., Hu, Y., Jin, S., Yao, Z.Y., 2016. Road centerline extraction from airborne LiDAR point cloud based on hierarchical fusion and optimization. ISPRS J. Photogramm. Remote Sens. 118, 22–36. https://doi.org/10.1016/J.ISPRSJPRS.2016.04.003.

Courtrai, L., Lefèvre, S., 2016. Morphological Path Filtering at the Region Scale for Efficient and Robust Road Network Extraction from Satellite Imagery. Pattern Recogn. Lett. 83, 195–204. https://doi.org/10.1016/j.patrec.2016.05.014.

Liu, R., Song, J., Miao, Q., Xu, P., Xue, Q., 2016. Road centerlines extraction from high resolution images based on an improved directional segmentation and road probability. Neurocomputing 212, 88–95. https://doi.org/10.1016/j.neucom.2016.03.095.

Shi, W., Miao, Z., Debayle, J., 2014a. An Integrated Method for Urban Main-Road Centerline Extraction From Optical Remotely Sensed Imagery. IEEE Trans. Geosci. Remote Sens. 52 (6), 3359–3372. https://doi.org/10.1109/TGRS.2013.2272593.

Shi, W., Miao, Z., Wang, Q., Zhang, H., 2014b. Spectral–Spatial Classification and Shape Features for Urban Road Centerline Extraction. IEEE Geosci. Remote Sens. Lett. 11 (4), 788–792. https://doi.org/10.1109/LGRS.2013.2279034.

Movaghati, S., Moghaddamjoo, A., Tavakoli, A., 2010. Road Extraction From Satellite Images Using Particle Filtering and Extended Kalman Filtering. IEEE Trans. Geosci. Remote Sens. 48 (7), 2807–2817. https://doi.org/10.1109/TGRS.2010.2041783.

Leninisha, S., Vani, K., 2015. Water flow based geometric active deformable model for road network. ISPRS J. Photogramm. Remote Sens. 102, 140–147. https://doi.org/10.1016/J.ISPRSJPRS.2015.01.013.

Ziems, M., Rottensteiner, F., Heipke, C., 2017. Verification of road databases using multiple road models. ISPRS J. Photogramm. Remote Sens. 130, 44–62. https://doi.org/10.1016/J.ISPRSJPRS.2017.05.005.

Xiao, L., Wang, R., Dai, B., Fang, Y., Liu, D., Wu, T., 2017. Hybrid conditional random field based camera-LIDAR fusion for road detection. Inf. Sci. 432, 543–558. https://doi.org/10.1016/j.ins.2017.04.048.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep Residual Learning for Image Recognition. Paper presented at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, June 2016.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. "ImageNet classification with deep convolutional neural networks," Paper presented at the NIPS'12. Proceedings of the 25th International Conference on Neural Information Processing Systems.

Redmon, J., Farhadi, A., 2017. YOLO9000: Better, Faster, Stronger. Paper presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu HI, USA, July 2017.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. 39 (6), 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031.

Taigman, Y., Yang, M., Ranzato, M.A., Wolf, L., 2014. DeepFace: Closing the Gap to Human-Level Performance in Face Verification. Paper presented at the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus OH, USA, June 2014.

Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition. Paper presented at the Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9.

Szegedy, C., et al., 2015. Going deeper with convolutions. Paper presented at the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), MA, USA, June Boston 2015.

He, K., Gkioxari, G., Dollar, P., Girshick, R., 2020. Mask R-CNN. IEEE Trans. Pattern Anal. Mach. Intell. 42 (2), 386–397. https://doi.org/10.1109/TPAMI.2018.2844175.

Huang, G., Liu, Z., Maaten, L.V.D., Weinberger, K.Q., 2017. Densely Connected Convolutional Networks. Paper presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu HI, USA, July 2017.

Meishvili, G., Jenni, S., Favaro, P., 2020. Learning to Have an Ear for Face Super-Resolution. Paper presented at the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle WA, USA, June 2020.

Chen, Z., Zhong, B., Li, G., Zhang, S., Ji, R., 2020b. Siamese Box Adaptive Network for Visual Tracking. Paper presented at the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle WA, USA, June 2020.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. Paper presented at the Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, Munich Germany, October 2015.

Sener, O., Koltun, V., 2018. Multi-task learning as multi-objective optimization. Paper presented at the NeurIPS 2018, Montréal CANADA Dec 2018.

Abadi, M., et al., November 2 2016, TensorFlow: A System for Large-Scale Machine Learning. Savannah, GA, USA, p. 21.

Hu, C., Fan, W., Du, J., Zeng, Y., 2018. Model-Based segmentation of image data using spatially constrained mixture models. Neurocomputing 283, 214–227. https://doi.org/10.1016/j.neucom.2017.12.033.

Badrinarayanan, V., Kendall, A., Cipolla, R., 2017. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 39 (12), 2481–2495. https://doi.org/10.1109/TPAMI.2016.2644615.

Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J., 2017. Pyramid Scene Parsing Network. In: Paper presented at the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu HI, USA, July, p. 2017.

Zhang, Z., Liu, Q., Wang, Y., 2018. Road extraction by deep residual u-net. IEEE Geosci. Remote Sens. Lett. 15 (5), 749–753. https://doi.org/10.1109/LGRS.2018.2802944.

Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2018. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. IEEE Trans. Pattern Anal. Mach. Intell. 40 (4), 834–848. https://doi.org/10.1109/TPAMI.2017.2699184.

Fu, J., et al., 2019. Dual attention network for scene segmentation. Paper presented at the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach CA, USA, June 2019.

Sachin, M., Mohammad, R., Anat, C., Linda, S., Hannaneh, H., 2018. ESPNet: Efficient Spatial Pyramid of Dilated Convolutions for Semantic Segmentation. Paper presented at the Computer Vision – ECCV 2018, Munich Germany, September 2018.