CS4490 2021-Summer Thesis

USING CONVOLUTION NEURAL NETWORK TO LABEL DIFFERENT PARTS OF REMOTE SENSING IMAGE

Ziqingqing Ye
Department of Computer Science
Western University

2021/08/11

Project supervisor:

Prof. Roberto Solis-Oba Department of Computer Science

Course instructor:

Prof. Nazim H.Madhavji

Contents

Glossary

ANN	Artificial Neural Network
DNN	Deep Neural Network
FCN	Full Convolution Network
ReLU	Rectified Linear Unit
Adam	Adaptive Moment Estimation
CNN	Convolution Neural Network

Abstract

The classification of remote sensing images plays a key role in several areas. For example, Adapt the measurements to the local conditions: land capability maps, vegetation cover maps and other maps can be obtained through remote sensing image classification, which can be used as the basic map for the next studies on the environment, land use and greening. Traditional remote sensing classification methods have the disadvantages of poor robustness, slow speed and when a particular data classification requires a particular algorithm, the accuracy is relatively low. In recent years, deep learning has achieved success in many fields. This method can capture the high-dimensional nonlinear characteristics of data, so it is of great significance to apply it to the classification of remote sensing.

Introduction

With the continuous development of remote sensing technology, the spatial information of high spatial resolution remote sensing images has become richer and more detailed [5]. At the same time, the complexity of high spatial resolution remote sensing images also puts forward higher requirements on the classification technology of remote sensing images.

Recent years, the classification algorithms often used in the field of building classification include Maximum Likelihood (ML), ISO clustering (ISO clustering), random forest (RF), support vector machine (SVM), and neural network (NN). However, these methods rely on spectral features and underutilize spatial features and are not suitable for remote sensing images' high spatial resolution with low spectral resolution. Nowadays, deep learning is already in use Speech recognition, image recognition, information retrieval and other fields have surpassed traditional machine learning algorithms. The strong extraction capabilities of image semantic segmentation algorithms for spectral and spatial features have led more scholars to introduce them to remote sensing images.

In classification, the current image semantic segmentation methods mainly include data-driven methods based on non-parametric conversion, Bayesian, Markov random fields, and conditional random fields, and these methods have relatively low segmentation efficiency and large computational complexity. Long et al. proposed a fully convolutional neural network (FCN) [4], which is a classical semantic segmentation network, and it improves the segmentation efficiency and reduces the computational complexity by discarding the fully connected layers. What is more, FCN-based image semantic segmentation algorithms are particularly outstanding in building extraction. For example, Zhang et al. proposed adaptive image segmentation and developed a multi-level classifier to further improve the accuracy of building extraction [8]; Zhao et al. used multi-scale images to construct multi-scale samples in order to fully excavates the spatial information in the remote sensing image [8]. However, at present, most of the image semantic segmentation models used for building extraction are slice-based network architectures. Compared with pixel-based end-to-end network architectures, this architecture lacks an overall understanding of the features in the sample and is less efficient [8]. Badrinarayanan et al. proposed the SegNet network [1]. This network is an end-to-end, pixel-based network architecture, more accurate outputs feature map can be obtained and more accurate classification results can be obtained even with limited training samples.

The main problem with this project is how to design a reasonable feature system and choose a suitable classification model to accurately and quickly grasp the quantity and distribution of rural construction land when facing more richer texture features in high spatial resolution remote sensing images. Regarding this question, I choose a powerful DNN called SegNet, which implements image segmentation by classifying each pixel in the image and identifying a category for each pixel. The second problem is the number of remote sensing image dataset is small, but its size of each image is large, my solution to this problem is to divide the original image into several small images to train the network.

An accurate and quick understanding of the quantity and distribution of land is of great significance to urban and rural planning, economical and intensive land use and sustainable development. Also, it is helpful to design and develop the application of deep learning model in building classification of high spatial resolution remote sensing images.

2. Background

2.1 ANN

Artificial Neural Network (ANN), short for Neural Network (NN). ANN is built like a human brain; the human brain is composed of hundreds of billions of cells called neurons [3]. As the "neural" part of their name implies, these are brain-inspired systems designed to imitate human learning. A neural network consists of input and output layers and a hidden layer that transforms the input into something that the output layer can use. The output of neural network depends on the network structure, network connection method, weight and activation function. Just as humans need rules and guidelines to obtain results or conclusions, artificial neural networks also use a set of learning rules called error back propagation (short for error back propagation) to improve their results. These are great tools for finding patterns that are too complex or too many for programmers to extract and teach machine recognition.

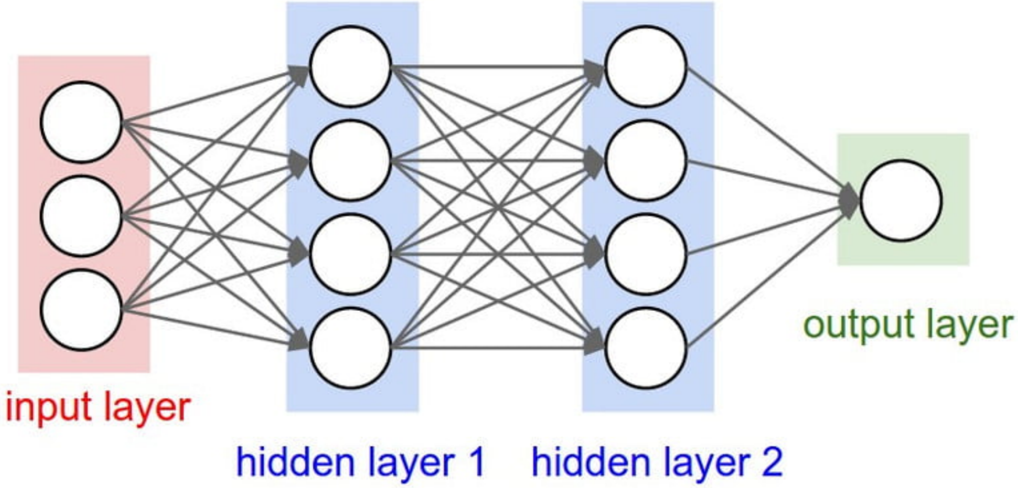


Figure 1. A regular 3-layers NN

source: *Convolutional Neural Networks for Visual Recognition*. CS231n convolutional neural networks for visual recognition. (n.d.). https://cs231n.github.io/convolutional networks/#norm.

As figure.1 shows, to get a rough idea of how to train deep learning neural networks. Imagine a production line -- after inputting the raw material (data set), it runs on the conveyor belt and generating different data records each time it is further stopped or further transferred. When the network is used for object recognition, the first layer can analyze the brightness of its pixels, and the next layer can use lines of similar pixels to identify the edges of the image. After that, the texture, shape, etc. can be seen on another layer. Then, the deep learning network will create a complex feature detector when the fourth or fifth level is reached. It can find that certain elements of the image (such as building, road, and tree) usually appear together.

Once completed, the researchers training the network can label the output and then use backpropagation to correct any mistakes they made. After a period of time, the network will be able to perform its own classification tasks without manual intervention every time.

The learning process includes forward propagation and backward propagation. Forward propagation is used to calculate the forward network, that is, to obtain the output result of a certain input information after the network calculation. Backpropagation is used to transfer errors layer by layer, modify the connection weights between neurons, so that the output of the network after calculating the input information can meet the expected error requirements.

Artificial neural networks have paved the way for the development of life-changing applications in all areas of the economy. The artificial intelligence platform based on ANN is changing the traditional way of doing things. The artificial intelligence platform simplifies transactions and makes services available to everyone. From translating websites into other languages, ordering products online, to communicating with chatbots to solve problems. All prices are negligible.

2.2 CNN

Convolutional neural networks have attracted much attention due to their unique advantages of local connections and parameter sharing. Generally, convolutional neural networks are composed of the following parts: input layer, convolutional layer, activation layer, pooling layer, fully connected layer, output layer, etc. Because this project uses a fully convolutional neural network, there is no fully connected layer in the convolutional neural network, so this paper mainly introduces the convolutional layer, the activation layer and the pooling layer.

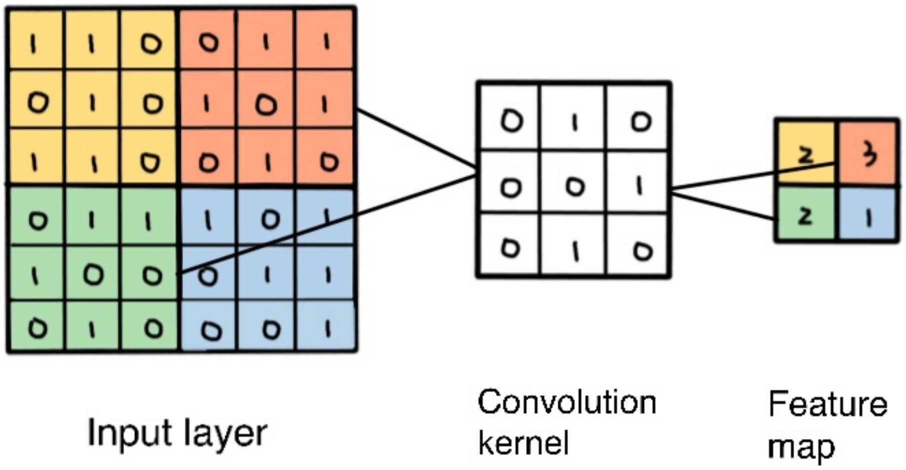
2.2.1 Why Convolutional Neural Network

By applying appropriate filters, ConvNet can successfully capture the spatial and temporal dependencies in the image. By reducing the number of parameters involved and reusing weights, the architecture is more suitable for image data sets. That is, the network can be trained to better understand the image's sophistication.

2.2.2 Convolutional layer.

The convolutional layer is composed of many small convolutional units. The convolution calculation is to extract the characteristics of the input image. Due to the large amount of information and the complexity of the image, a single convolutional layer can only obtain some superficial features, so more complex and obscure features can only be obtained by combining multiple convolutional layers. By adding layers, the architecture also adapts to advanced properties (High-Level features), giving a network which has an overall understanding of the images in the dataset. The operation has two results: one is that the dimensionality of the object is reduced compared to the input, and the other is that the dimensionality increases or remains the same. This is done by applying valid padding in case of the former or applying the same padding in case of the latter.

The backpropagation algorithm is the same between the convolutional neural network and the traditional neural network. They are both used to adjust and improve the parameters of each convolution unit. The convolution kernels sequentially pass the input features in a sliding window, and the convolution kernels determine the size of the receptive field. In the receptive field, matrix multiplication and summation are performed on the input layer and a bias term is added. Figure shows a schematic diagram of the convolution operation.



1	1	0	0	1	1
0	1	0	1	0	1
1	1	0	0	1	0
0	1	1	1	0	1
1	0	0	0	1	1
0	1	0	0	0	1

0	1	0
0	0	1
0	1	0

2	3
2	1

Figure 2. Convolution layer

2.2.3 ReLU function

Because the convolution operation has a single linear feature, it can only express the relatively simple connection between the input and the output. So, the activation function plays an important role in more complicated tasks and situations. It can control the results within a controllable range. An activation function is generally added after the convolutional layer, so that the convolutional neural network can face more and more complex tasks and situations due to the existence of the activation function, which greatly improves the modeling ability. This project selects ReLU activation Function, the ReLU activation function has the advantages of fast speed, convenient parameter adjustment, and generally no gradient explosion problem. Besides, the ReLU function is unilateral suppression and has a wide acceptance range. The mathematical expression of the ReLU activation function is

$$R(x)=\max(0, x)$$

The model curve is shown in the figure, the function is always 0 in the range of $(-\infty, 0)$, and it is a piecewise function of yx in the range of $(0, +\infty)$.

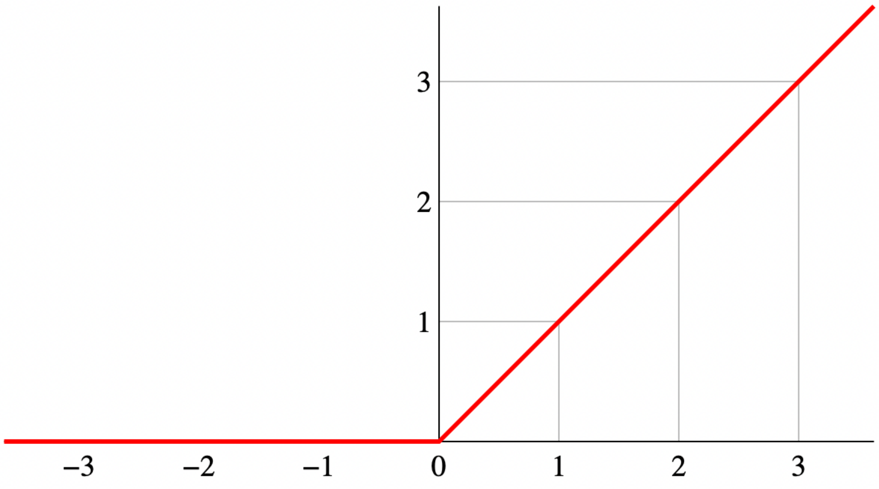


Figure 3. ReLU function

After the convolutional layer and ReLU function operation, all pixels in the image are given the information of the adjacent area, so it will cause the information to be repeated and occupy space. If continue calculate, it will not only reduce the performance of the algorithm, but also cause the loss of local feature information. Therefore, it is necessary to subsample the image to improve the performance and stability of the algorithm. This behavior is called pooling and there are two main types of pooling: maximum pooling and average pooling.

2.2.4 Pooling layer

Generally, a pooling layer is regularly added between successive Convolutional layers in the Convolutional Network architecture. Its function is to gradually reduce the space size of the representation to reduce the number of parameters' calculations in the network, thereby controlling over-fitting. The pooling layer works independently of each input depth segment and changes its space size through the MAX operation. The maximum pooling operation is to select the largest number of all values in the receptive field for output, and the average pooling layer operation is to select all the values in the receptive field and average the number for output. Figure shows the operation of max pooling. Figure 3 shows the most common down-sampling operation is max, which results in maximum pooling, shown here with a stride of 2. This means that each maximum has 4 numbers (small 2x2 squares).

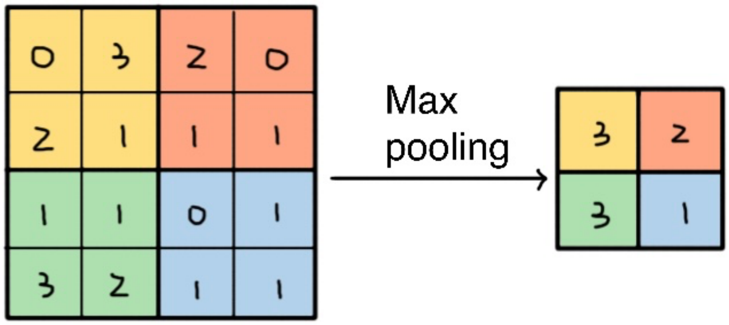


Figure 3 Pooling layer

2.3 Full Convolution Network

When it is necessary to classify images, traditional convolutional neural networks cannot classify each pixel. If you want to input the classified pixels into the network, you must select a certain range around the pixels to be classified, and the same is true for testing. The test value of the entire selected range needs to be used as the test result of the pixel. However, this method has certain disadvantages. Because the traditional convolutional neural network is constantly reusing the pixel, it needs to have a large the increase in the storage area and the amount of calculation also leads to a long training time, and the size of the selected image range also has a certain impact on the perceptual domain, which makes the acquired semantic information not rich enough, and the classification effect is also not good.

To cope with these problems, Long et al.6 proposed a fully convolutional neural network structure, including three core ideas: fully convolutional structure, deconvolution operation, and jump structure. Next, we will introduce these three core ideas in detail.

2.3.1Fully convolutional structure

This means that all fully connected layers in the network are replaced by convolutional layers, so that there is only a convolutional structure in the network. The full convolution structure allows the size of the input image in the convolutional neural network to be of any size, so that the required samples can be better obtained, and the model structure can be better adjusted.

2.3.2 Deconvolution operation.

The method of deconvolution operation can increase the size of the feature image that has become smaller after multiple convolutions in the network and restore it to the size of the original image. At the same time, different from the general upsampling method that can make the image size larger, back propagation can be used to make the deconvolution mode to keep the convolution kernel learning, so that the deconvolution operation can also perform feature learning. The image size after the deconvolution not only becomes larger, but the original pixel information can be predicted more accurately. Figure 4 shows upsampling via deconvolution, dark blue represents input and dark green represent the output.

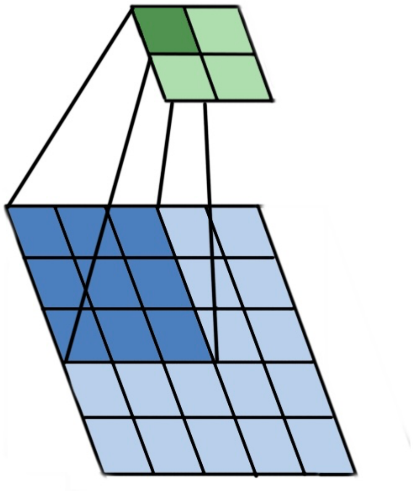


Figure 4. Deconvolution layer

The skip structure means that when the deconvolution method is used to restore the image size to the size of the original image, the relevant information of the shallow feature map and the deep feature map can be superimposed and fused to jointly predict the type of pixel. Due to the need to take multiple convolution operations on the input image in the network structure, the size of the finally obtained image will become very small. Although the finally obtained image contains deep and rich information, the size of the image is too large. Small, if the image obtained in this way is directly restored to the size of the input image through the deconvolution operation, then very inaccurate prediction results will be obtained. The skip connection from the previous layer provides the decoder layer with the information which is needed to create a precise boundary.

2.4 frameworks

The mainstream deep learning frameworks mainly include Caffe, Torch, Keras, MXNet, Tensorflow CNTK, Theano, etc. This paper uses U-Net model to detect remote sensing image changes, using Keras supported by Tensorflow as the framework.

Keras is an advanced deep learning API, which contains CNN, FCN, DBN and other network structure codes. This project is written in Pycharm software using Python language. Keras need to install one or more of Tensortlow, Theano, and Microsoft-CNTK before installation. Keras can not only use multiple computers to train the model at the same time, run on different systems, but also switch between CPU and GPU, which can effectively reduce the time required for model training.

3. Methodology

The sample set in this article is divided into image sample set and label sample set. The preprocessed image is used as an image sample, and ArcGIS 10.2 is used to draw the vector file needed for the label sample, and the obtained label sample is processed by One-hot encoding. Because the computer takes too long to process the entire remote sensing image, and the video memory and memory are limited, this project uses a sliding window method to crop the processed image sample and the corresponding label sample into a 256x256 png format image with the support of the python language. Image sample set and label sample set perform data enhancement operations such as flipping, rotating, and adding noise to the sample set. The sample set obtained after screening is divided into training samples, verification samples, and test samples. The training samples are used to debug the parameters of the model, and the verification samples are used to check the effect of the model. The sample set construction process is shown in the figure 5.

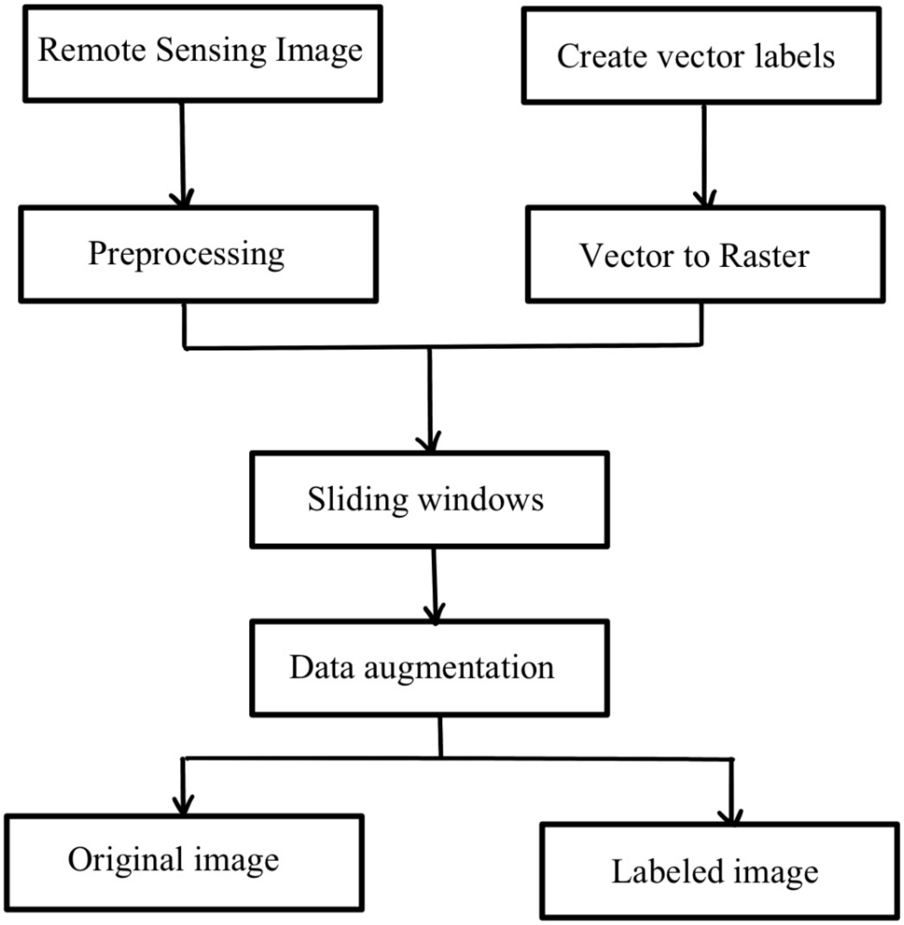


Figure 5. dataset

3.1 SegNet

The overall architecture is SegNet, a semantic segmentation model based on deep convolutional neural networks used in this project, as shown in Figure 6. The network model is mainly composed of an encoder network, a decoder network and a pixel-wise classification layer, and each convolutional layer is followed by a Batch Normalization layer and a ReLU activation function. The encoder network consists of 13 convolutional layers, corresponding to the first 13 convolutional layers in the VGG16 object classification network, and discarding the fully connected layers, which are conducive to storing higher-resolution feature maps at the deepest encoder output [2].

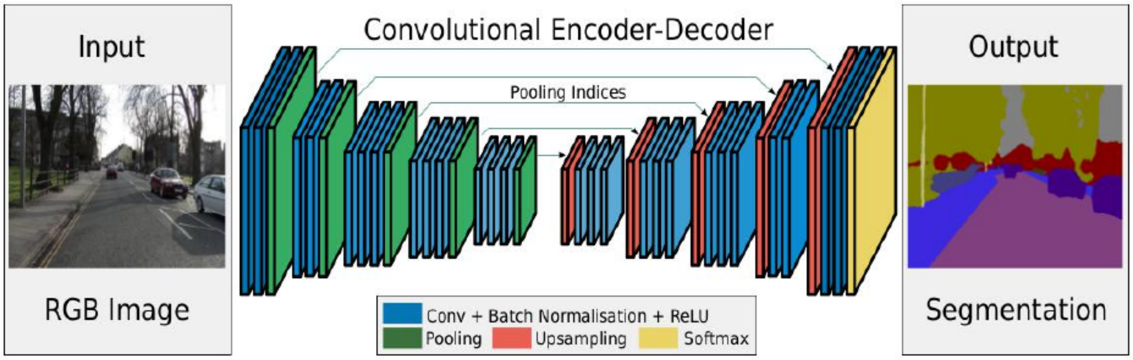


Figure 6. SegNet

Source: Fezan. (2019, October 24). *Understanding of semantic segmentation & How SEGNET model work to perform semantic segmentation*. Medium. https://medium.com/@fezancs/understanding-of-semantic-segmentation-how-segnet-model-work-to-perform-semantic-segmentation-5c426112e499.

Encoder. The encoder network and the filter bank are convolved to create a set of feature maps. Then normalize them in batches. Next, apply the rectified linear nonlinear element (ReLU) max (0, x), After that, use a 2x2 window with stride 2 (non-overlapping window) to perform maximum pooling, and use a factor of 2 to downsample the resulting output to achieve translation invariance for small spatial changes in the input image. Although multi-level maximum pooling and downsampling can provide higher translation invariance for reliable classification, the spatial resolution of the feature maps will be lost. Therefore, it is necessary to obtain and save the boundary information in the encoder feature maps before performing downsampling. The coding network converts high-dimensional vectors into low-dimensional vectors and performs low-dimensional extraction from high-dimensional objects. Although the coding network can capture more translation invariance features through multiple maximum pooling operations, it will also lose more important basis for segmentation such as the boundary information of the feature map. Therefore, during the pooling process, the maximum pooling index information is recorded at the same time, and the position of the maximum feature value is saved, and then the input feature map is picked up using the maximum pooling index information, so that the boundary information can be saved.

Decoder. The decoder network uses the stored maximum clustering index from the corresponding encoder feature map to map its input feature map. These feature maps are combined using filter banks that can be trained with the decoder to generate dense feature maps. Then apply the batch normalization step to each of these maps. The output of the high-dimensional object representation from the final decoder is passed to the trainable Softmax classifier. This soft-max can classify each pixel independently and the result of soft-max classifier is the probability of K channels, where k is the number of classes. The segmentation corresponds to the category with the highest probability of each pixel.

Skip connections. If only encoder and decoder layers are stacked together, there may be slight information loss. Therefore, the boundaries of the segmentation maps generated by the decoder could not be accurate. To compensate for the lost information, we allow the decoder to access the low-level features generated by the encoder layers. This is achieved by skip connections. The intermediate output of the encoder is added/combined with the intermediate input of the decoder at the corresponding position. The skip connection from the previous layer provides the decoder layer with the information which is needed to create a precise boundary.

Activation function layer. For deep convolutional neural network, a nonlinear activation function needs to be connected behind each hidden layer. ReLU is used as the activation function in SegNet network structure to enhance the recognition capability of features.

Upsampling layer. Restore the original size of the image and enlarge the features after image classification.

Soft-max layer. The classifier classifies each pixel separately, and its output is the probability that each pixel belongs to each classification. The classification with the maximum probability of each pixel is the classification of its predictive segmentation.

3.2 Dataset

Preparing the data set is the first step in the training segmentation model. First of all, we need to input RGB images and the corresponding segmented image. After producing the segmented images, store them in the testing or training folder. Then, create separate folders for segmentation images and input images. The file name of the input image and the corresponding segmentation image must match. The data set I adopted this time is the data provided by CCF big data and computing intelligence competition and it is short for CCF BDCI. (High-definition remote sensing image of a city in southern China). This is a relatively small data set, which contains 5 large-size RGB remote sensing images with labels (size range from 3000*3000 to 6000*6000), in which four types of objects are marked, vegetation (Marked as 1), buildings

(Marked as 2), water bodies (Marked as 3), roads (Marked as 4) and others (Marked as 0). Among them, cultivated land, forest and grassland are classified as vegetation. In order to better observe the labeling, visualized three training pictures as follows: blue water body, yellow house, green vegetation and brown road. Data details: http://www.datafountain.cn/#/competitions/270/data -intro.

Data augmentation. If there are few training pairs, the results may not be that accurate because the model may be overfitted. Regarding this problem, we can increase the size of the data set by applying random transformations to the images. For instance, change the color attributes of the input image, such as hue, saturation, brightness, etc. Moreover, apply transformations such as rotation, scaling, and flipping are also good strategies.

4. Result

The network is tested on two Ge-force GTX 1080 Ti GPU s in an Intel Xeon(R) CPU system with 32×2, 10 GHz cores. I implement the model using Keras, OpenCV and Scikit-learn mainly, which are known python library. Using OpenCV to detect the preprocess dataset and Scikit-learn to build transformer the label encoder.

The sample of original data is list following and the original image was clipped into 10000 sub image samples. After inferencing the label, the label of model output could not be visualized directly, which is like figure 8. So, it has to be transformed into RBG image. Then the final result is figure 9. I trained the data for 30 epochs, and batch size was 50. The optimizer I used is adam. We can find that as the train epoch grows, the loss curve is convergence, and the accuracy is growing.

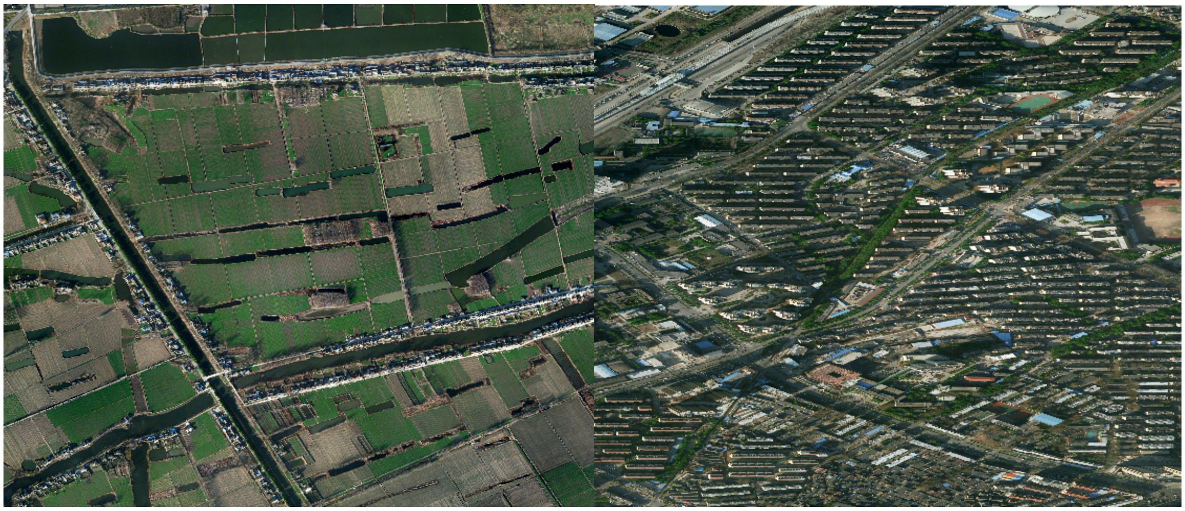


Figure 7. Original data

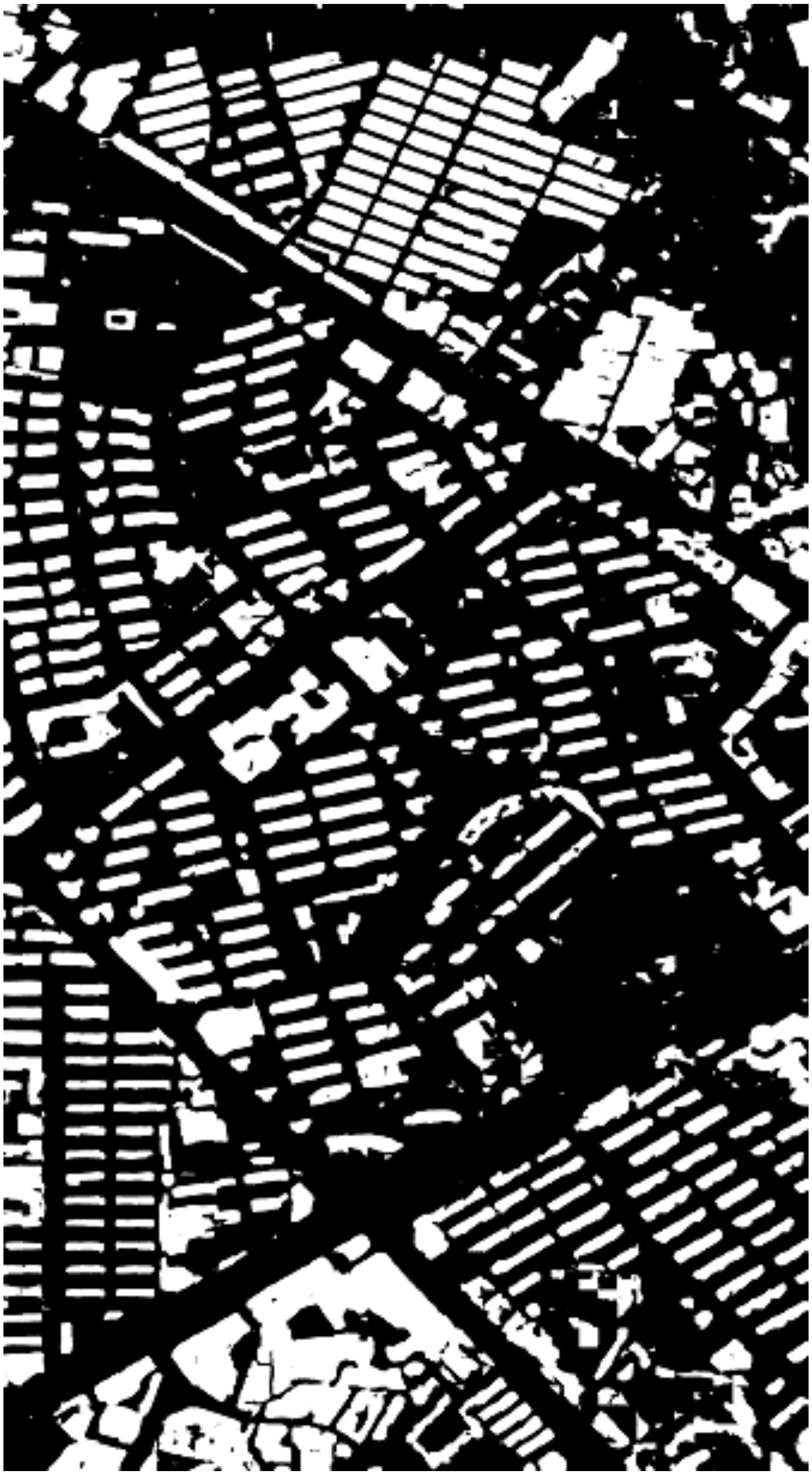


Figure 8. Mask

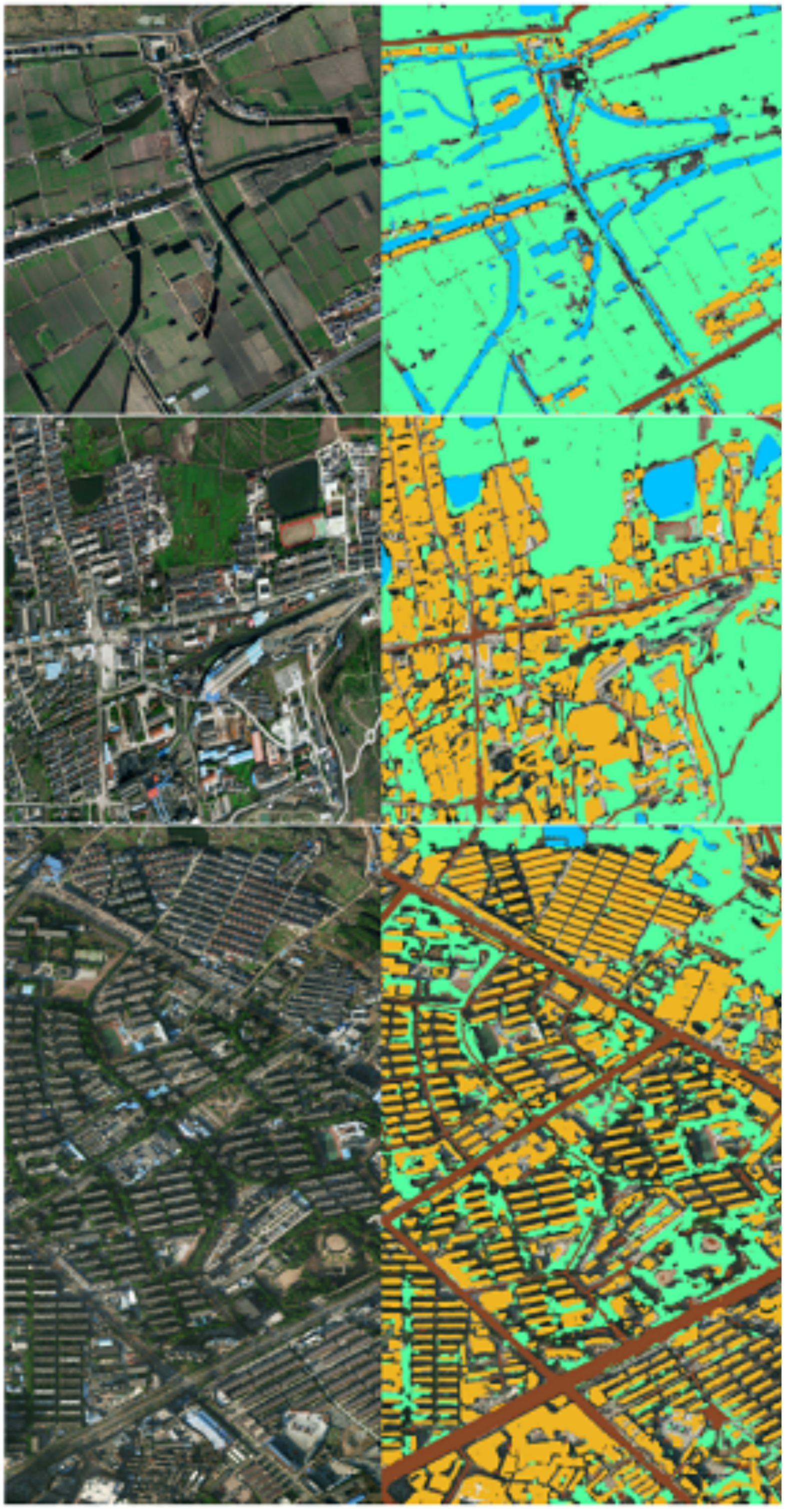


Figure 9. Final result

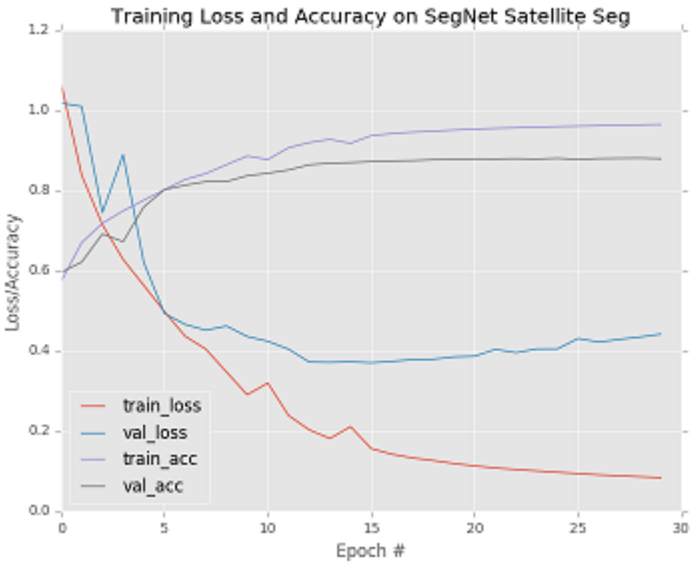


Figure 10. Training loss and accuracy

5. Conclusion

This project uses the data of a southern Chinese city as the data source, selects the image semantic segmentation algorithm SegNet based on deep convolutional neural network to extract rustic buildings in high spatial resolution remote sensing images, and clusters them with the maximum likelihood method ML and ISO. During the training of semantic segmentation model segnet based on deep convolution neural network, because the decoder of the model is a process of up sampling and convolution and only convoluted their corresponding feature maps, which reduces the training parameters and saves computing resources. As the number of iterations increases, the loss function decreases rapidly and gradually tends to be stable, the accuracy improves rapidly and tends to be stable, and the convergence speed is faster. Finally, an idealistic feature suitable for pattern classification is formed, which enhances the convergence and generalization ability of the model and improves the classification accuracy of the model.

6. Future Work

Semantic segmentation model based on deep learning has inestimable potential in the field of remote sensing image classification.

On the technical level, one of the challenges is the time required to train the network, which may require powerful computing power to complete highly sophisticated tasks

s. However, the biggest problem is that neural networks are "black boxes" where users enter data and receive responses. They can refine the answer, but he has no access to the exact decision-making process. Also, the data set used in this project is relatively simple. When there are more classified objects or smaller recognition targets, it will be more challenging and we can consider about more powerful and complex neural network structure, attention mechanism, transformer, etc. Semantic segmentation model based on deep learning is still a developing technology. There are still some deficiencies in this paper, such as the selection of network model and there is no consummate theoretical basis for the setting of training parameters. The parameters of model in experiment currently are selected by manually, this process could be automated in the future.

Acknowledgments

Supervised by Prof. Roberto Solis-Oba.

Reference

[1] Badrinarayanan, V., Kendal, A., & Cipolla, R. (2017). *SegNet: A deep Convolutional encoder-decoder architecture for image segmentation*. IEEE Xplore. https://ieeexplore.ieee.org/document/7803544.

[2] Fezan.(2019, October 24). *Understanding of semantic segmentation & How SEGNET model work to perform semantic segmentation*. Medium. https://medium.com/@fezancs/understanding-of-semantic-segmentation-how-segnet-model-work-to-perform-semantic-segmentation-5c426112e499.

[3] Frankenfield, J. (2021, May 19). *Artificial neural Network (ANN)*.Investopedia. https://www.investopedia.com/terms/a/artificial-neural-networks-ann.asp.

[4] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic. segmentation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/cvpr.2015.7298965

[5] Tan, kun, Zhang, Y., Du, Q., Du, P., Jin, X., & Li, J. (n.d.). *Page 733*. PE&RS November 2018 Full. https://www.asprs.org/a/publications/pers/2018journals/PERS_November_2018_2mY-5Ak/HTML/files/assets/basic-html/page67.html.

[6] Volpi, M., & Tuia, D. (2017). Dense semantic labeling of subdecimeter resolution images with convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, *55*(2), 881–893.

https://doi.org/10.1109/tgrs.2016.2616585

[7] Zhao, W., & Du, S. (2016). Learning multiscale and deep representations for classifying remotely sensed imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, *113*, 155–165.

https://doi.org/10.1016/j.isprsjprs.2016.01.004

[8] Zhang, X., & Du, S. (2019). Learning self-adaptive scales for extracting urban functional zones from very-high-resolution satellite images. *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*.

https://doi.org/10.1109/igarss.2019.8898975