Vrije Universiteit Amsterdam

Universiteit van Amsterdam

Master Thesis

Combining Machine and Human Intelligence for Object Recognition in Satellite Images to Support Land Cover Management in Africa's Drylands

---

Author: Shuqi Yan (VU:2631023/UVA:12482617)

1st supervisor: Anna Bon
2nd reader: Hans Akkerman

A thesis submitted in fulfillment of the requirements for

the joint UvA-VU Master of Science degree in Computer Science

August 18, 2020

"I am the master of my fate, I am the captain of my soul"
from Invictus*, by William Ernest Henley*

Abstract

Land use and land cover management of Africa's Drylands can benefit from automated, Artificial Intelligence-based remote sensing techniques. However, for the proper interpretation of the satellite images of new, underexplored regions, contextual knowledge from local experts is often required. This research sets out to design a system that can save time and manpower to efficiently investigate land use. This is done using public satellite images a data source and object identification as the main technique. A combination of human intelligence and machine learning techniques is used to optimize object identification. Evaluation of the proposed prototype by a group of test users yields new requirements for refinement and further development and deployment of the system for rural Africa. The research outcomes show that the combination of machine learning and human intelligence is an adequate method to achieve results with remote sensing of images of new, underexplored regions.

Contents

List of Figures

LIST OF FIGURES

List of Tables

LIST OF TABLES

1

Introduction

1.1 Context

Land use and land cover management is a real world problem that many countries and regions are facing, especially for the regions with vast land area. The traditional method for land cover management is manually achieved by humans, which wastes plenty of time and labor, as well as it is difficult to observe the up to date information and change of a region. Developing regions, such as some regions in Asia and Africa have problems such as land degradation, low soil fertility, deforestation or negative impacts caused by climate change, which requires more understanding for proper land use and land cover management. Many researches have proposed solutions to this problem, but there is no study to propose land management methods that combine machine and human intelligence. Therefore, this paper proposes a method that combines machine learning methods and local expert knowledge. The purpose is to use remote sensing images to solve the problem of unclear target area segmentation or unknown object identity without field work.

The method proposed in this study needs to train a machine learning algorithm model (a method familiar to professional computer science researchers) to achieve remote sensing image interpretation, combined with the knowledge of local users (users may not be familiar with machine learning algorithms, but have professional knowledge in the field of image analysis). The user's domain knowledge is indigenous knowledge, which is not only theoretical knowledge learned in the literature, but learned from life experience. Combining this kind of human knowledge with machine knowledge is a brand new research method, and conducting field experiments locally will get better results, but this is not applicable to this research, so this research is carried out in the Netherlands as the experimental area. Knowledge itself is not universal, and research in different fields needs to use differ-

ent domain knowledge (this is generally composed of ontology). In order to quickly build domain knowledge, the basic domain knowledge used in this article is implemented from the database of Corine Land Cover. Due to the non-universal nature of domain knowledge, relevant ontology knowledge needs to be supplemented for object recognition in different regions.
Before entering the environment under development, the research and test users (from the Netherlands) tested the research method and discussed its feasibility.
With the combined use of various technologies, including satellite imagery, object recognition technology, and machine learning, there is great potential for achieving remote land management efficiently, and this is not trivial for Africa or other rural areas

1.2 Objective

From the perspective of the time cost and labor required for research, it is unrealistic to investigate land use by the government. It is also not feasible for local people to spend time conducting land surveys because it will affect their daily life and work.
Therefore, the objective of this research is to implement a system that can save time and manpower to efficiently investigate land use needs to be developed. Using public satellite images as a data source and object identification as the main technique to analyze images is suitable for this purpose. In order to obtain accurate results, the combined usage of technologies can be necessary.

1.3 Research Question

According to the objective presented in the former parts, this research aims to answer the following question: Is it possible to combine machine learning and (local) human intelligence for a better interpretation of complex objects of satellite images in a low resource context?
The question can be stated by answering two sub research questions:
Sub-RQ1: How can we design a promising approach for the interpretation of new objects on satellite images based on various techniques to identify image patterns?
Sub-RQ2: How can we include (local) expert knowledge to make a proper improvements to the models and obtain better results for the interpretation of land use?

1.4 Research Method

In order to answer the questions raised in this study, feasible solutions need to be considered. This research needs the help of local experts' knowledge, but if all the object recognition parts depend on the participation and guidance of experts, a lot of research time will be wasted. Therefore, this study proposes a method that can use both machine intelligence and human intelligence to achieve high efficiency and high accuracy. The system mainly includes two parts, one uses machine intelligence, semantic segmentation of remote sensing images is made according to machine learning algorithms, and the other part uses expert knowledge to implement the interaction between experts and machines by building user interfaces, thus adjust the prediction results generated by machine learning algorithm. Further, the system is supposed to generate more accurate land use information. The modified information is used to supplement the model training data, and further train the model to improve the accuracy of the model.

2

Background and Related Work

2.1 Remote Sensing and Object Recognition

2.1.1 Remote Sensing Overview

Remote sensing, in general, is a technology to monitor the surface of the earth through various sensors. Remotely-sensed data/information is, in theory, produced by remote sensing sensors, which can be classified as passive and active sensors. As defined in (1), passive sensors are "those which sense natural radiations, either reflected or emitted from the earth", such as photographic and thermal, while active sensors are "the sensors which produce their own electromagnetic radiation", such as LiDAR, radar, x-ray and etc. Remote sensing is also classified as optical and microwave (1).

Optical remote sensing mainly refers to the collection of detectable solar radiation through sensors, such as visible, near-infrared, mid-infrared and thermal infrared bands, which are mainly reflected, refracted or scattered through the surface of the earth, and further generates satellite images by sensors above the surface. The surface of the earth is covered with various materials, such as vegetation, water bodies, oceans, forests, buildings, roads, etc. Different objects have different abilities to reflect or refract solar radiation, and respond to different wavelengths of sunlight. This special characteristic provides the possibility of object recognition based on the information in the remote sensing image. Recent technologies fuse different combinations of the bands to analyze when recognizing different objects. For example, in the spectrum of Landsat 8 OLI in Figure 2.1, the combination of band 4 , 3, 2 (Red, Green, Blue) fusion becomes the closest natural color image, and the combination of band 5, 4, 3 (NIR, Red, Green) generates false color images, which can better identify vegetation. Other more fusion of bands and corresponded usage description is shown in Figure 2.2.

Sensor	Band	Band Range/μm	Signal to Noise Ratio	Spatial Resolution/m
OLI	1-COASTAL/ AEROSOL	0.43-0.45	130	30
	2-Blue	0.45-0.51	130	30
	3-Green	0.53-0.59	100	30
	4-Red	0.64-0.67	90	30
	5-NIR	0.85-0.88	90	30
	6-SWIR1	1.57-1.65	100	30
	7-SWIR2	2.11-2.29	100	30
	8-PAN	0.50-0.68	80	15
	9-Cirrus	1.36-1.38	50	30
TIRS	10-TIR	10.60-11.19	0.4K	100
	11-TIR	11.50-12.51	0.4K	100

Figure 2.1: Landsat-8 Band Description

R. G. B	Type/Usage
4. 3. 2	Natrual Color
7. 6. 4	False Color (urban)
5. 4. 3	Color Infrared (vegetation)
6. 5. 2	Agriculture
7. 6. 5	Atmospheric Penetration
5. 6. 2	Healthy Vegetation
5. 6. 4	Land/Water
7. 5. 3	Natural with Atmopheric Removal
7. 5. 4	Shortwave Infrared
6. 5. 4	Vegetation Analysis

Figure 2.2: Band Combination and Usage

However, optical remote sensing is largely affected by factors such as the atmosphere and humidity. For example, the formation of clouds will cover the objects on the ground. Therefore, in many cases, the application of optical remote sensing needs to perform preprocessing such as cloud removal of the image, which limits its application to a certain extent. Microwave remote sensing, since its detection of the surface is not affected by time, light, and atmospheric factors, makes it a helpful source of remote sensing data, and it is also used in conjunction with optical remote sensing data in some applications (2). There are many types of Earth observation satellites widely used worldwide. Low-resolution sensors, such as MODIS, with spatial resolution of 1000 meters. Multi-spectral medium-resolution sensors, such as Landsat-7 ETM+ and Landsat-8 OLI, with resolution of 30 meters. Hyper-spatial sensor, such as QUICKBIRD and RAPID EYE, whose spatial resolution can reach around 5 meters. More information of various sensors can be find in Figure 2.3.

Type	Sensor	Spatial/m	Spectrual/#
Coarse Resolution	AVHRR	1000	4
	MODIS	250-1000	36/7
Multi Spectral	Landsat-7 ETM+	30	8
	Landsat-8 OLI	30	8
Hyper-Spatial	QUICKBIRD	0.61-2.44	4
	RAPID EYE	6.5	5
	WORLDVIEW	0.55	1

Figure 2.3: Sensors Description

2.1.2 Remote Sensing and Land Cover

Information on land cover and land use is of great value for strengthening land management and planning. In recent years, with the continuous development of remote sensing technology, the spatial resolution of images is also getting higher and higher, which makes remotely sensed data have the opportunity to provide more contributions in the field of land cover detection research. The spatial resolution of remote sensing data has a significant impact on the identification of land cover. The types of objects that can be identified under different spatial resolution images vary greatly. For example, the minimum spatial

resolution for identifying forest land cover is only required to be 20 to 1 km, while the identification of specific vegetation types requires a spatial resolution of 0.1 to 2.0m, thus, the choice of corresponding sensors is also different. (3)

The sensors mainly include three types: coarse spatial resolution sensors, medium spatial resolution sensors, and high spatial resolution sensors. Since the resolution of the coarse spatial resolution sensor is above 250m, it does not make much contribution in the field of object recognition, but it can be applied to the acquisition of information on a wide range of land profiles. The MODIS sensor with a resolution of 250m can be used to study some plant or crop species in a fixed research area, such as the identification of the main species of crops in Central Asia.(4) Medium spatial resolution sensors, including the earliest Landsat project. Among various medium spatial resolution sensors, the Landsat Thematic Mapper (TM) sensor has been widely used for identifying vegetation, soil types, etc. due to its higher spectral and spatial resolution. (5). The remotely sensed data from Landsat has also been pointed out to be beneficial for crop-based object-based classification. (4) High spatial resolution images make it possible to identify specific objects in the landscape. For example, using the high-resolution sensor Quickbird, an object recognition study was performed on a region in Strasbourg, France.(6) Vegetation, water, road, and house with orange roofing tiles were effectively recognized.

Further, some studies have combined data of medium spatial resolution and high spatial resolution to identify crop species in specific regions, such as West Africa. For example, in the (2) study, Landsat, RapidEye, and TerraSAR- X remote sensing data is combined to analyze the types of rainfed crops. This type of research is valuable in the field of land management.

2.1.3 Image Interpretation

Image interpretation, in essence, is defined to automatically extract semantic information from the given image. In the field of computer vision, image interpretation leads a possible way to transfer imagery data into information that can achieve further machine understanding and processing. However, the semantic gap problem is always faced by image interpretation researchers (7), which causes mismatching between the knowledge from users and the automatically extracted information from an image. Ontology has been introduced in a lot of research in order to address the problem, since it provides a formal, unambiguous and uniform method to express text content. Studer et al. (8) stated that: "An ontology is a formal, explicit specification of a shared conceptualization." Hence, ontologies, every of which composed of a set of concepts, is widely used to describe regions

in a more formal and generic way of a specified imagery.
In the field of region-based image interpretation, two main steps are taken into account in most of the approaches, which are region building and region labeling. Region creation means to segment the image into several homogeneous and continuous regions, then characterize a set of low-level descriptors to each of the regions. While the main purpose of region labeling is to provide every region with the best-matched concept(s) of ontology. Ontology-based region characterization sets every single region with a semantic label, thus provides a higher possibility to recognize objects correctly.(6)

2.1.4 Object Recognition Applications

Object recognition is a main application in the domain of image interpretation systems. Considering about land cover and land use researches, the basic technique is classification of objects on earth surface. There are mainly two types of object classification approaches, which are region-based classification and pixel-based approaches, according to that, various image analysis applications have been developed. From the perspective of effectiveness, region-based classification methods have been proved to be better than pixel-based ones for high-resolution image processing.(9) While pixel-based semantic segmentation methods, with the participate of machine learning algorithms, also stepped into the public view, since it makes the maximum use of the computing power of machines, hence saved most of the human power.
For region-based segmentation, the basic idea is to implement edge detection for each region and segment images according to the edges. Traditional edge detection method, watershed algorithm, was a popular methodology in the earlier period, which is always combined with other algorithms nowadays and widely used for image segmentation. For instance, (10) mentioned a methodology that combined watershed and spectral methods and improved the performance to a higher level by using watershed algorithm to detect the basic regions, and cluster micro-regions using spectral method. In (11), a watershed-based method combined with machine learning algorithms, namely a fuzzy supervised classification procedure and a genetic algorithm, has been proposed. In the methodology, machine learning is used to construct the elevation map of the watershed paradigm and adjust the segmentation parameters. A watershed based image segmentation algorithm is proposed in (12), which implemented edge techniques to preprocess the image and get the estimated gradient for further region segmentation, as well as using a region adjacency graph and a bottom-up hierarchy to generate the final segmentation result. There are also methods

that implemented both watershed algorithm and convolutional neural network (CNN), watershed for coarse-grained segmentation and CNN for fine-grained segmentation, such as it mentioned in (13).

Pixel-based segmentation is mainly about using a slide-window on the image, checking per pixel when sliding, while locating each pixel a named object label. Pixel-based segmentation can also be expressed as a color-based method. Applications of pixel-based segmentation are mostly about distinguishing an object from the background, such as face recognition, person feature recognition and etc. For the purpose of distinguishing target object from the background, traditional methods always transform the original image to a binary format, then construct a gradient image, and recognize the part where the gradient value is significantly different as the foreground, but the performance of the method is limited since the pixels of background can also be changed in a large range. (14) proposed a new method, which is called Pixel-Based Adaptive Segmenter (PBAS), based on traditional methods but used control system theory. PBAS is always adjusting parameters for each pixel during runtime, which achieved impressive performance on the Change Detection Challenge(15). Pixel-based segmentation can also perform as a color-based way to be used in other fields, such as medical field. With the help of machine learning algorithm, K-means clustering technique, pixels of brain images are clustered based on the color differences, thus this method can be used to detect the tumor in the brain.(16) In addition, semantic image segmentation based on CNN is also a pixel-level classification, such as deep convolutional networks (DCNNs).(17) In recent years, this segmentation method has also shown its advanced performance in image segmentation tasks, and performed impressively in medical and remote sensing fields.

2.2 Machine Learning

2.2.1 K-Means

K-means algorithm is a commonly used clustering algorithm. The basic idea of the algorithm is to randomly select k samples as the center point in a given sample set, and then divide the sample set into k clusters according to the distance between points in the sample set and each center point, and each cluster is a category. The K-means algorithm is suitable for many fields, such as medical care, pattern recognition, traffic images, image processing, etc. (18)

In the field of brain tumor recognition, a color-based K-means image segmentation method is proposed in (16). This method combines K-means clustering and histogram-clustering

to perform tumor recognition on the converted grayscale image. The automatic classification and recognition of pedestrians is of great significance to the field of automatic driving technology and intelligent vehicle development. The accuracy of the classification results determines the maturity and safety of automatic driving technology and can reduce the occurrence of accidents. To address this task, a method based on K-means and random decision forest is proposed. (19) This method combines k-means and a radial basis function to transform the data into a smaller and more relevant set, and then merges the random decision forest algorithm for classification. The experimental results prove that the classification result of this method is very satisfactory, and the accuracy of identifying pedestrians reaches 97.37%.

In this study, the k-means algorithm was also used in the image segmentation, but the classification results were not very satisfactory, so other algorithms were tried such as the CNN algorithm that is going to be mentioned in the following section.

2.2.2 Convolutional Neural Network

With the rise of deep learning, convolutional neural networks are increasingly used in various research fields including image recognition. The special structure of sparse connection and weight sharing of convolutional neural network not only greatly reduces the computational complexity of the model, but also its rotation invariance and scaling invariance make it robust. At present, Google has achieved an amazing accuracy of 96.9% on the ImageNet dataset using convolutional neural networks(13). At the same time, the application of convolutional neural networks in the classification field of satellite images can better solve the problems of noise caused by different satellite distances from the ground, different shooting angles of remote sensing equipment and atmospheric multi-spectral scattering.

The CNN algorithm is a deep learning algorithm that simulates the connections and working methods between neurons in the cerebral cortex. This algorithm was gradually developed from the MP model (20) proposed by psychologist McCulloch and mathematical logic scientist Pitts in the earliest period and became the CNN algorithm model today. The CNN model uses a single neuron as the basic processing unit. The input signal generates the output signal through the neuron. The neuron processes the signal by introducing an appropriate activation function. The sigmoid function, ReLu function, and tanh function are more commonly used.

The basic structure of CNN is evolved from Multilayer Perceptron (MLP), which consists of three parts: input layer, hidden layer and output layer. The hidden layer can contain 1 to n layers, see Figure 2.4 for details. The signal is connected to all the neurons in the

adjacent hidden layer or output layer through the input layer, and the signal transmission direction is from the input layer to the hidden layer to the output layer. Different input signals and neurons may have different weight values. In the classification algorithm, the number of neurons in the input layer should correspond to the number of input feature values, the number of neurons in the output layer corresponds to the number of classified categories, the number of hidden layers and the number of neurons in each layer can be set according to specific circumstances.

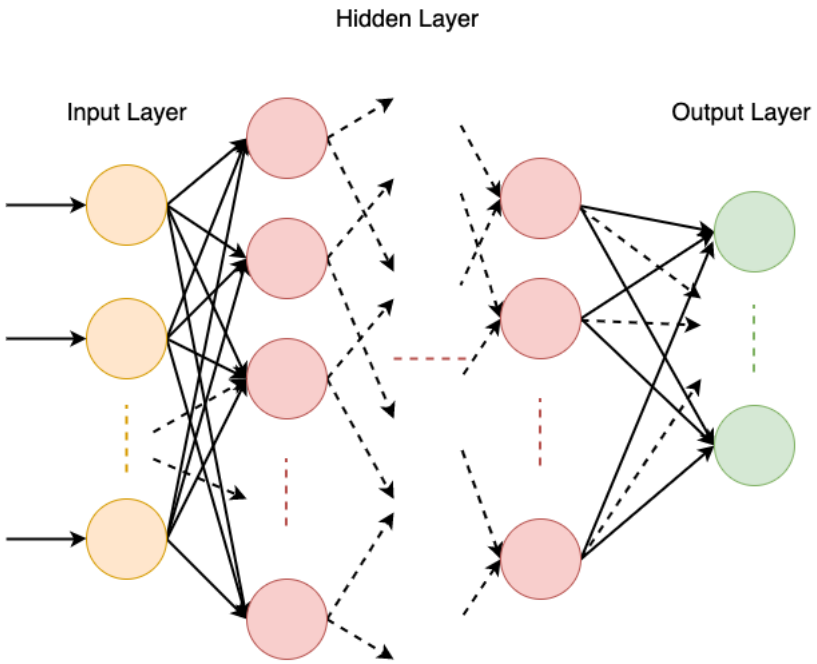

Figure 2.4: MLP

The complete CNN structure is composed of an input layer, a convolutional layer, a sampling layer, a fully connected layer, and an output layer, where the convolutional layer and the sampling layer are generally arranged in an alternating manner. Each convolution layer is composed of three parts, namely convolution part, pooling part and nonlinear activation function layer. CNN uses the convolution operation of the convolutional layer, such as using a 5*5 filter, to extract different features of the input layer by layer, and to extract more advanced features as the number of layers increases. Sampling is performed by the pooling operation to improve the calculation speed, and finally passed to the activation function to obtain the output value of each neuron. Check details in Figure 2.5 (21).

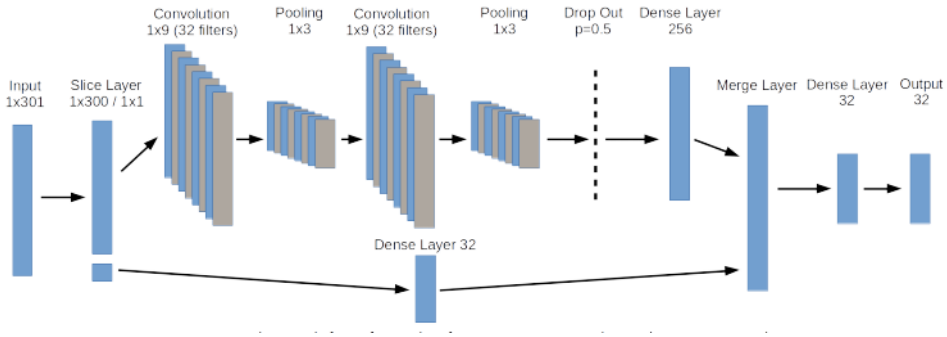

Figure 2.5: CNN

2.2.3 Applications of CNN

Due to the advantages and features mentioned in the previous section, the achievements of the CNN algorithm and its derived series of models in many research fields have received high attention from public.

As the resolution of remote sensing images improves, more details on the surface of the earth are recorded, which poses challenges to traditional image segmentation processing techniques. Deep Convolutional Networks (DCNNs) based on the CNN model have been proposed, applied and solved advanced computer vision problems, such as the semantic segmentation of remote sensing images, and played an indispensable role in various fields such as land use, land cover, urban construction, forestry and agriculture. DCNNs use a large number of training set samples to train deep learning models to improve the accuracy of prediction results, thereby achieving accurate pixel-level segmentation (semantic segmentation) of remotely sensed images. The Deeplab model proposed according to the fully convolutional network structure has achieved good results in the field of close-up image processing, such as the application of Google's pedestrian detection using the Deeplab model. In terms of semantic segmentation, some studies have proposed the method of bringing together DCNNs and probabilistic graphical models, and some studies have proposed to combine DCNNs with its multi-layer features to improve segmentation accuracy. The former research has achieved performance improvements in three aspects of the basic "DeepLab" system, namely processing speed, prediction accuracy, and simplicity of system construction (17). The latter was improved on the original "DeepLabv3" model, its segmentation performance evaluation was performed in three directions, namely pixel accuracy (PA), mean pixel accuracy (MPA), and mean intersection over union (MIoU). The experimental results showed that the performance of the model has been improved in

three aspects (22).

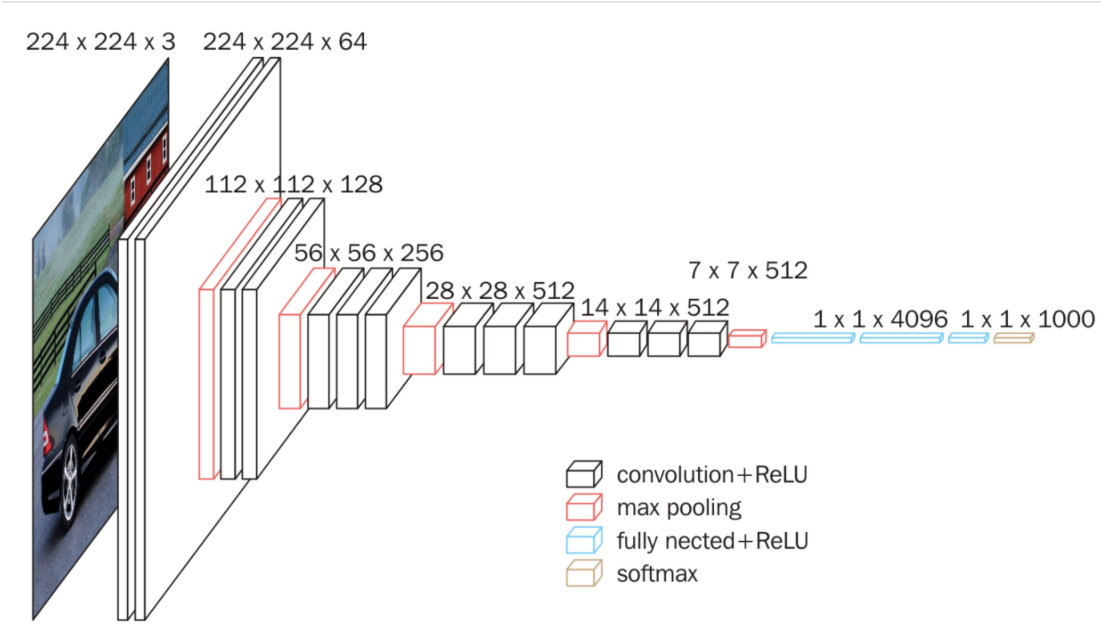

Figure 2.6: VGG16

Furthermore, the region-based convolutional neural network (R-CNN) model has also been rapidly developed in the field of object recognition. In recent years, R-CNN has been continuously optimized, the Fast R-CNN and Faster R-CNN algorithms have been generated to further improve the accuracy of object recognition. The advantage of the Faster R-CNN model is that it includes a Region Proposal Network (RPN) that is used to generate high-quality suggested region frames. Advantages of the model is that it can share the full-image convolution features during object detection and shortening time spent in this period (23). The overall structure of Faster R-CNN combines the two models of Fast R-CNN and RPN. RPN is responsible for the detection of the region, while Fast R-CNN is responsible for learning the characteristics of the region and classifying the region. This integrated model with clear division of labor greatly improves the effectiveness of the algorithm. Fast R-CNN contains two output layers, one to predict the category of the object, and the other to optimize the coordinates of the proposed object to obtain a more accurate target position (24). Faster R-CNN has three main models, namely ZF (small) model, VGG_CNN_M_1024 (medium) model, and VGG16 (large) model. Although the VGG16 model requires a larger GPU, its advantage is that the depth of the model is deeper, and it achieves better results in feature extraction, thereby improving detection accuracy. VGG16 contains a total of

13 convolutional layers, 3 fully connected layers, and 5 pooling layers. Among them, the convolutional layer and the fully connected layer have weight coefficients, also known as weight layers. VGG16 model and its structure are as shown in Figure 2.6 and Figure 2.7. In (23), R-CNN, Fast R-CNN, Faster R-CNN are compared, and the experimental results prove that Faster R-CNN greatly improves the recognition speed and accuracy. In the aircraft recognition task, compared with R-CNN, Faster R-CNN improved the recognition accuracy from 77.10% to 96.67%, and the recognition time of a single image was reduced from 13.40s to 0.14s.

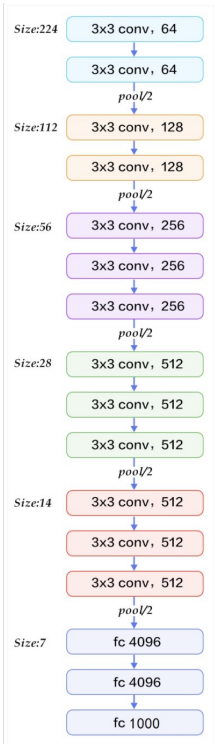

Figure 2.7: VGG16 Structure

2.3 Knowledge-Based Classification

Knowledge-based classification is essentially a method to solve the problem that generated during the region labelling part of object recognition. The expression of the same semantics may have different forms of text writing because of the context and context it appears, or in some cases there may be different cases of upper and lower case uses, resulting in the ambiguity of the semantics of the text. As mentioned in the previous sections, ontology was introduced as a formal and unified concept definition method to solve the ambiguity

of classification names.

The main step of ontology-based object classification is to assign each object to a concept in the corresponding knowledge domain. Such as mentioned in (25) and (26), SIGMA and Schema are two systems that provide knowledge for satellite imagery interpretation. However, the truth is, there are rarely any perfectly matched knowledge bases for the domain people are working with, so building a knowledge base especially for the needed knowledge domain is a challenging task. An ontology-based supervised learning system is presented in (27), named OntoPic, which is developed based on ontologies from DAML+OIL and DL reasoner for better results during image retrieval. In (28), a methodology is proposed for object learning and recognition. It combined both machine learning and knowledge representation, based on an existed visual concept ontology with basic concepts for remotely sensed field. An idea that worth mentioning is that machine learning algorithm is used for learning the visual concept in the paper, expect which, knowledge acquisition and object classification are the other two main parts of the method.

Ontology presents not only the concepts of objects, but also the relations between each concept. When considering the usage of ontology in real world cases, it is necessary to take both concepts and relations into account. In the use cases of ontology, (6) mainly pays attention to the application of ontology concepts in label objects, and (7) believes that the utilization of relations between ontology concepts can also effectively improve the semantic gap problem. However, the second use case is mainly used for structural recognition, so this study focused more on the first use case.

In use cases for land cover recognition, according to the different spatial resolution of the remotely sensed data, the important thing is to build a proper knowledge base that can also be reused. In (9), a few nomenclatures have been introduced for different urban area mapping scales. For instance, coarse-grained recognition of the land cover (e.g. urban fabric, airports, water bodies, etc) only requests a scale between 1:100,000 to 1:50,000 to map urban area, while fine-grained recognition for objects (e.g. housewater surfaces, bare soil, etc) which is able to achieve due to the advent of high spatial resolution images, a scale of 1:5,000 is needed. At different scales of the mapping, the proper names of objects (concepts) for one application can also be a lot different, since at some scale a few objects might not be able to be recognized. Therefore, according to various requirements of use cases, multiple kinds of data sets are selected, thus lead to the diverse choices of the nomenclatures, such as Corine Land Cover nomenclature (for 30m spatial resolution). Furthermore, two levels are considered in object recognition, one is the recognition of a single object, such as a house, another more difficult level is aggregate object recognition,

like block which includes several houses, gardens, and roads. In (9) a knowledge-based region labeling research is applied in a district of Marseille (a city in France). It made a fine-grained recognition of single objects, for example, orange house. The research proved that the proposed approach generates accurate enough results, as well as the knowledge base, that built in line with the methodology, is reusable in other districts. For this study, the finding is of great reference.

In this research, the main focus is on the recognition of a single object, but considering the spatial resolution of public remote sensing data, sufficiently fine-grained recognition may not be achievable.

3

Materials and Methods

3.1 Materials

3.1.1 Dataset

Reliable remote sensing data sources currently accepted by the public include the aforementioned QUICKBIRD and RAPID EYE (high spatial resolution sensors), Landsat-7 and Landsat-8 (medium spatial resolution sensors), as well as MODIS (Low spatial resolution sensors). Satellite sensors with high spatial resolution can take close-range, high-precision photos of surface objects, thereby achieving high-accuracy recognition of objects. However, there is currently no platform that provides free high-spatial resolution data sources, so this study selected the highest-resolution Landsat-7 and Landsat-8 data sets among the free remote sensing image sources, which can reach a spatial resolution of 30 meters. Landsat-7 includes 7 bands from TM1 to TM7, Landsat-8 includes 9 bands that Operational Land Imager (OLI) records and 2 bands that generates from Thermal Infrared Sensor (TIRS). In this study, Landsat-7's TM1 (blue), TM2 (green), TM3 (red) bands, and Landsat-8's 2 (blue), 3 (green), and 4 (red) bands were mainly used for natural color fusion.
The storage format of the Landsat series of remote sensing images is GeoTiff, which stores the number of pixels containing in the image in x and y directions, the number of image bands, the geographic coordinate information of the dataset, and projection information. This information is necessary in the subsequent processing of remotely sensed data.

3.1.2 Ground Truth

In addition to the selection of the dataset, the most important part is to find the "ground truth" that can be relied on during the data preprocessing process to ensure the accuracy of constructing the training set. Since the accuracy of the training set directly affects the

prediction results of the machine learning algorithm, both the naming system and the land use mapping data need to be accurate.

For the choice of naming rules, according to the method in (9), the existing domain ontology knowledge can be used and build a knowledge base suitable for the system, or use the existing object nomenclature mentioned in (9), such as Corine Land Cover, Spot Thema nomenclature, and BDCarto IGN. Constructing a new naming system using ontology knowledge requires expert knowledge in related fields, otherwise the knowledge base may lack universality and reliability, also, there is currently no unified evaluation standard for domain ontology knowledge, which makes the construction of domain ontology knowledge bases a more competitive challenge. Since the dataset selected in this study comes from Landsat dataset, and the Corine Land Cover nomenclature constructed for the Landsat data set (30 meters spatial resolution) just matches the system's requirements for naming rules, this object nomenclature was adopted.

Figure 3.1: Corine Land Cover Nomenclature

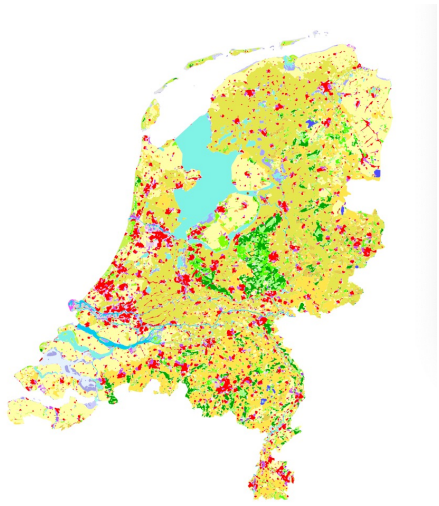

Figure 3.2: Corine Land Cover Map

Corine Land Cover nomenclature stipulates the names of the specifications of all land cover objects at a spatial resolution of 30 meters, including urban fabric, industrial, commercial and transport units, mine, dump and construction sites, artificial non-agricultural vegetated areas, and stored them as a map style in the form of numbers. (29) All category names are shown in Figure 3.1. This naming rule guides the naming system of this study and makes it proceed under a standardized situation. At the same time, the Corine study

also provided land use mapping data for the Netherlands as shown in Figure 3.2, which included all the land use types and regional mapping information for the Netherlands in 2018. This information was used to generate the training set for this study.

3.2 Methods

This study uses a method that combines machine intelligence and human intelligence. Machine intelligence uses machine learning algorithms to train the selected model (the U-net model in this article) and adjust the parameters to achieve a relatively high accuracy of image semantic segmentation results. Since this method is not combined with other algorithms, the ideal prediction result cannot be achieved at the machine processing level. In order to improve the above-mentioned shortcomings, the research introduces the human intelligence part, which aims to achieve the desired results through expert knowledge. By building a user interface, this study attempts to improve the accuracy of object recognition results with the help of human intelligence. The detailed research methods will be elaborated below.

3.2.1 U-Net

The first step of the method in this study is to use machine learning algorithms to semantically segment the preprocessed images.

First, the k-means algorithm is adopted. However, because this algorithm is used for color-based image segmentation, while for satellite images, different objects may contain pixels of multiple colors, or there are situations where the colors of different objects constitutions are similar with each other, thus caused not ideal result of k-means segmentation.

Finally, the research uses the CNN-based VGG16 model. The difference between VGG16 applied in Faster R-CNN and VGG16 used in the research method in this article is that the former aims at object detection, while the latter aims at image segmentation and classification. The VGG model can be divided into 6 configuration types according to the number of convolution layers and the size of the convolution kernel, namely type A, A-LRN, B, C, D, and E. VGG16 uses a D-type configuration, including 13 convolutional layers, 5 pooling layers, and 3 fully connected layers. As shown in Figure 3.3, the convolution kernel size of each convolutional layer of VGG16 is 3*3. The 13 convolutional layers include two convolutional layers with 64 channels, two with 128 channels, three with 256 channels, and six with 512 channels. The stride parameter in the convolutional layer is 1, and the padding parameter is set to same padding, so as to keep the convolution-processed

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Figure 3.3: VGG Configuration

image the same size as before the processing. The filter size of the pooling layer is 2*2, the stride parameter is 2, and the pooling method of max pooling is used. The advantage of max pooling is that it only retains the strongest feature value to reduce the problem of overfitting and at the same time ensure the position and rotation invariance of the feature. The VGG16 model structure mainly adopts the form of alternately stacking convolutional layers and pooling layers, which facilitates the construction of a deeper network structure. All in all, this study uses VGG16 as the basic model because of its simplicity and deep network structure.

The semantic segmentation model used in this paper is the U-net model based on VGG16. The structure of this model is similar to a letter "U" shape, as shown in Figure 3.4, so it is called the U-net model. The U-net model consists of two parts, one is the encoder and the other is the decoder. The encoder part is the left half of the figure. Its structure

is similar to the traditional VGG16 model, which contains a series of convolutional layers and pooling layers. Encoder is responsible for the convolution operation (blue arrow) and pooling operation (red arrow) of the input, to achieve the feature extraction of the image and reduce the dimension. The right part of the figure is the decoder part. The purpose of the upsampling operation (green arrow) is to restore the dimension of the feature map (blue and white box). The up-sampling method of U-net is different from that of FCN, which stitches the features together in the channel dimension and generates a new feature map. Skip-connection (grey arrow) also belongs to the decoder part, which aims to merge the features. At the end, a 1*1 convolution operation (cyan arrow) is used to generate the output.

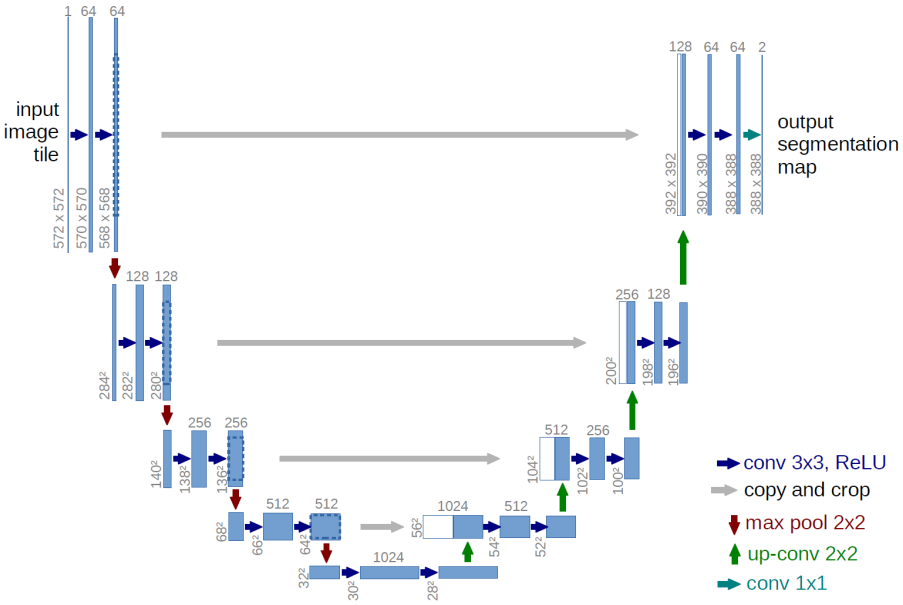

Figure 3.4: U-net Structure

3.2.2 Expert Knowledge

The purpose of this study is to construct a reusable and universal object recognition method. Therefore, expert knowledge is added to the method, which aims to use human intelligence to correct the predicted results to improve accuracy. Considering that a given machine learning model is applicable to a limited set of data sets, but changing the structure of the model or retraining the model according to the data set will take a lot of time, thus a method that can avoid repeated training and reorganization of the model needs to be developed. In addition, sometimes it is difficult to find the appropriate "great truth" data

that the training set needs to rely on, which is also one of the problems this article tries to solve.

The embodiment of the part of the expert knowledge in the entire system is mainly for modifying the label and shape of the segmented result through the interaction with the database and the user interface. The back end of the system stores the prediction results generated by machine learning in the database, and stores it as a geojson format that the front end can read and process. The front end displays the predicted results on the map through the corresponding geographic coordinate information in the form of marked polygon collection. Experts are able to click on each polygon through the web page and change its corresponding label and coordinates. The modified result will be stored in the database via API. This operation has improved the accuracy and effect of segmentation, as well as answered the Sub-RQ2 that needs to be answered in this study:

Which part of the whole procedure should expert knowledge be utilized to make a proper improvement?

3.2.3 User Interfaces

The user interface, as an essential part of the system, supports the participation of expert knowledge in this research. There are three main standards for the design of the front end of the system: 1. Easy for the user to operate; 2. The simplicity and readability of the interface; 3. The aesthetics.

At the beginning, the research tried to use JS and Mapbox to build a user interface. The advantage is that Mapbox contains many built-in components, which greatly improves the efficiency of development. However, because this development method is difficult to integrate with other third-party libraries or existing projects, Vue.js, which is a progressive framework for building user interfaces, is used as an interface development framework.

Web development based on Vue.js includes three parts, namely HTML, JS and CSS parts. The HTML part controls the surface according to the components that build the interface, such as buttons and pictures. The JS part is equivalent to the "brain" of the user interface, and is responsible for guiding the interaction between components, such as mouse click trigger events. The CSS part mainly writes the style of each component, such as color and shape. The three parts together build the entire user interface. In addition, the research also built an API through Flask. The function of the API is to connect the user interface with the data processed by the backend.

The map of the interface is built using the map component in the Leaflet component library. The prediction results are displayed in the form of polygon collections of different

colors on the map through geojson format files. The "hover" event of the mouse triggers the display of the properties of each polygon, namely id and label. The mouse click event realizes the popup of the selection box, and then the user can select and modify the label of the corresponding polygon in the drop-down menu, and change the position of the point by dragging the set of points constituting the polygon, further changes the shape of the polygon. All changes can be sent back to the database through the API after clicking the save button.

4

Experiment and Results

In order to intuitively explain and verify the feasibility of the method proposed above, this chapter will use the Dutch remote sensing datasets taken by Landsat-7 and Landsat-8 for experiments. In the experiment, only the red, green, and blue bands of the multiple bands of the Landsat series of remote sensing data were used. The image set contains 45 types of objects that can be recognized at a spatial resolution of 30 meters. All the recognition target objects correspond to geographical concepts, such as continuous urban fabric, port areas, airports, etc., which are defined by the Corine Land Cover nomenclature.
Since the land cover mapping map used in this study is data for the Netherlands, the accuracy of the labeling of regions can be guaranteed, so the accuracy of the training set labeling can also be guaranteed. It should be noted that the land use database contains the data of the Netherlands in 2018, thus the corresponding remote sensing data also needs to be captured in 2018. The semantic segmentation of images is done by U-net model.

4.1 Preprocessing

The raw dataset is a collection of images taken over the Netherlands by Landsat-7 and Landsat-8 satellite sensors in 2018. It contains 30 sets of raw images in GeoTiff format (each group contains three images that is blue, red, and green band separately). The pixels contain in each image is around 7700*7800.
The first step of preprocessing is to use the QGIS software to fuse the three bands of each group of images to obtain 30 natural-color remote sensing images, the format and information of which are not changed. Subsequent processing of natural color pictures need to call Geospatial Data Abstraction Library (GDAL), a computer software library, to read information in GeoTiff format images, including GetGeoTransform (geographic

location information of raster data), RasterXSize, RasterYSize (the number of pixels in x, y directions of images), GetProjection (projection information) and RasterCount (band number) and etc.

The second step of preprocessing is to segment the image. During the experiment, we first tried to divide the picture into 256*256. A single image was divided into about 961 sub-pictures, which caused too large size of the dataset and difficult to be managed. The category of objects contained in each picture is relatively small. Considering that a single image contains fewer object categories, which may affect the prediction result of machine learning, the size of the segmented image is changed to 512*512.

The third step is data augmentation. The method of expanding the training set is mainly through operations such as rotating, color changing, and cropping the image, as shown in Figure 4.1 and Figure 4.2. One of the benefits of this operation is that more data can be generated to train the model when the training set has insufficient data. Another benefit is that this operation can effectively improve the prediction accuracy. After learning the different angle transformation and color transformation of the image, the model can reduce the influence of factors such as angle and color on the prediction result.

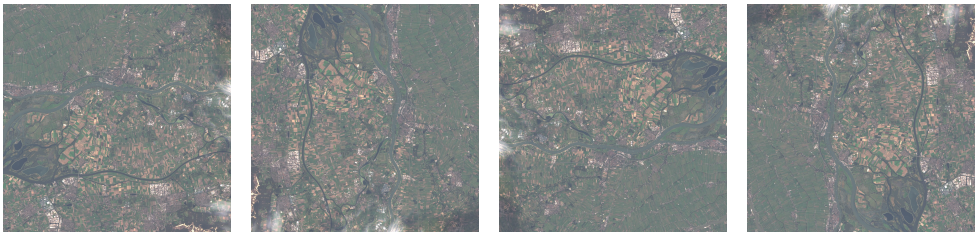

Figure 4.1: Rotated Image

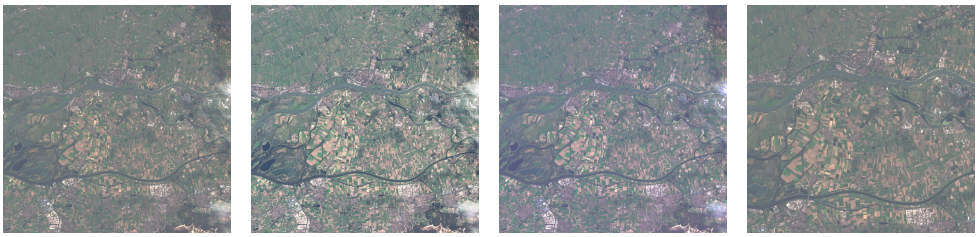

Figure 4.2: Color Changed and Cropped Image

The last step is to label the image. The basis for labeling is the map marker mapping data of the Netherlands in 2018 provided by the known Corine Land Cover (CLC) data source. The information of the CLC data set is composed of labels and polygon sets (one-to-many

mapping form). Labeling is mainly to determine whether each pixel belongs to a certain polygon, so as to mark the pixel as a label corresponding to the polygon. However, because the applied CLC dataset only includes the land cover map of the Netherlands, the parts outside the Dutch land have no corresponding labels and cannot be used as training set data. Therefore, the marked image and the original image pairs need to be selected. In the process of performing this step, it was found that the efficiency of judging pixel by pixel is low, which causes the program to run slowly. Therefore, according to the classification of each pixel point, the intersection between the image to be marked and the polygon in the polygon set is taken, thereby improving the calculation efficiency. The comparison between the original image and the labeled image is as shown in Figure 4.3 and Figure 4.4.

Figure 4.3: Original Image

Figure 4.4: Labeled Image

After the above four steps, the data preprocessing is completed. The result of preprocessing is the natural color image set after cropping, and the corresponding labeled dataset. The training data set contains a total of 534 pairs of 512 * 512 pixels size of original images and labeled images.

4.2 Model Training

Before using the U-net model, it is necessary to configure the environment. The reason why the vgg_unet model in the keras neural network library is selected for the experiment is because keras is open source and supports rapid experiments. Keras relies on TensorFlow, CNTK, or Theano as the backend. Tensorflow was configured in the experiment. Then, in order to call the unet model, the keras_segmentation module needs to be installed. The advantage of choosing keras library is that the built-in model in the module can be used

directly, which saves the time of rebuilding the model and determining the feasibility of the model. At the same time, the models in keras are proven and reliable open source models, which also help to obtain satisfactory segmentation results.
The vgg_unet model used in this experiment is an unet segmentation model constructed based on VGG16. The advantage of using VGG16 as the basic model is that it has fewer layers, so the training speed is relatively fast. The training of the vgg_unet model is mainly for the training of its check_points, which is a directory to save all model weights. Therefore, the training of check_point is the training of model weights.

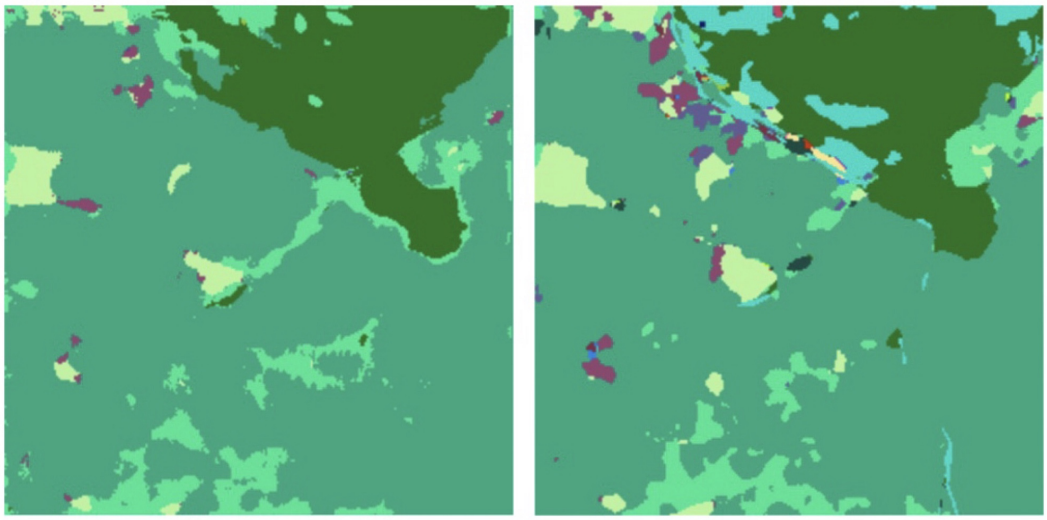

Figure 4.5: Segmentation Result

The ratio of training set, validation set and test set is 0.6:0.2:0.2. The role of the training set is to fit the model. The purpose of setting the validation set is to use it when implementing cross-validation on the model. The test set is used to test the accuracy of the model. The input training set is a set of 320 pairs of images obtained during data preprocessing, x_train is 320 original images, and y_train is 320 pixels labeled images. While validation set and test set each contains 107 pairs of images. The effect of model training is mainly improved by adjusting the number of iteration epochs. As the number of epochs increases, the image features extracted by the model are more detailed and precise, thereby improving the accuracy of segmentation. By plotting the segmentation results under different iteration epoch times, as shown in Figure 4.5 (the left one is result after 5 epochs, right one is after 40 epochs), it can be observed that as the number of epoch increases, the segmentation accuracy has been significantly improved. In addition,

according to the observation of the accuracy and loss data, the conclusion can be drawn, that is, as the number of epochs increases, the accuracy increases and the loss decreases. It should be noted that the ideal training result is obtained when the loss tends to be constant.

4.3 Model Evaluation

In order to determine whether the trained model can be used in the system, in other words, to judge its reliability and accuracy, evaluating the model is a necessary experimental step. The evaluation of the model mainly has three metrics, first, observe the loss curve, second, observe the pixel accuracy, and finally, observe the mean intersection over union.

4.3.1 Loss Curve

By visualizing the loss curve of the model, the performance of the model can be classified into three categories, namely underfit, overfit and good fit. The training loss is the loss value calculated by the set loss function, and the CNN loss function is the cross entropy. Cross entropy is the logarithm of the likelihood that the model outputs the correct label. Cross entropy has a certain relationship with the accuracy of the model.
The under-fitted model shows that the training loss decreases slowly as the number of epochs increases, and the curve is relatively flat. Underfitting indicates that the model's learning effect is not good, and the features of the data are not well captured. One obvious characteristic of underfitting is that after many iterations, the loss of training still has a downward trend, rather than tending to be stable. Overfitting is mainly due to the excessive training of the model, which makes the model learn some noise and random fluctuations, but often leads to inaccurate prediction results. The overfitting loss curve shows that when the training loss gradually decreases and stabilizes, it begins to show an upward trend. The underfitting and overfitting of the model are due to problems in the training process, resulting in unsatisfactory training results. The characteristic of the loss curve that achieves good fit is that in the first few rounds of training, the rate of decline in training loss is relatively fast, and after a certain period of training, the rate of loss loss slows down and gradually tends to remain unchanged. Good fit is the ideal state that model training tries to achieve.
Figure 4.6 plots the loss curve of the vgg_unet model of this experiment. The x-axis represents the number of epochs, and the y-axis represents the corresponding loss value. It can be observed from the figure that after 45 epochs training, the loss value stabilizes at

around 0.0795 and the curve tends to be flat, no longer falling or rising. This phenomenon indicates that the model has reached a good fit, which means the model is well trained and is reliable to be used in the further research.

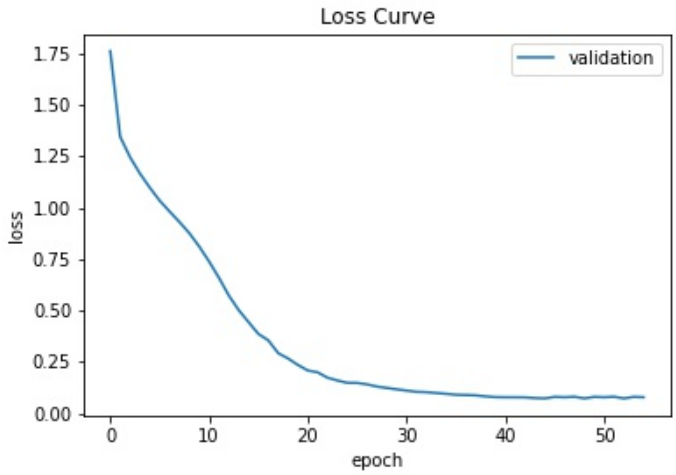

Figure 4.6: Loss Curve

4.3.2 Pixel Accuracy

Cross entropy mainly reflects the stability of the model, which can represent the accuracy of the model to a certain extent, but this is not absolute. While the calculation method of the model accuracy rate is achieved by judging pixel-by-pixel whether the classification is labeled correctly, in other words, the quotient of the number of pixels correctly labeled and the total number of pixels. In some specific cases, the loss increases and the accuracy rate may decrease. For example, in the case where the accuracy of the model remains very high, if an error occurs, the accuracy drop will be small, but the loss may become very high. Therefore, in the classification problem, the measurement of the reliability of the model depends more on the value of the calculated pixel accuracy. The calculation formula of Mean Pixel Accuracy (MPA) is shown as follows:

$$MPA = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k} p_{ij}},$$

In the formula, k is the kth image of the whole training set, p_{ii} represents the number of correctly labeled pixels, while p_{ij} represents the total number of the pixels in the kth

image.

In order to ensure the credibility of the calculated accuracy rate of the model, k-folds cross-validation method is applied in the experiment. The k-folds cross-validation method divides the data set into k folds according to the specified training set and verification set ratio (as we mentioned in former section, the ratio is 6:2, thus the k is supposed to be 4), and then randomly takes one of the folds as the verification set, and the remaining k-1 folds as the training set, repeat the above step k times, obtain k accuracy values, and take the average of them. The final obtained value is considered to be the most reliable model pixel accuracy rate.

Figure 4.7 plots the curve of pixel accuracy of the model during the learning process. The x-axis represents the number of epochs, and the y-axis represents the corresponding pixel accuracy value. It can be observed from the figure that the value of pixel accuracy gradually increases as the number of epochs increases. When the epoch reaches 45 times, from the change of the slope of the curve, it can be found that the highest point is stable at about 97.40%, no longer rising or falling, which represents that the model has reached the highest pixel accuracy.

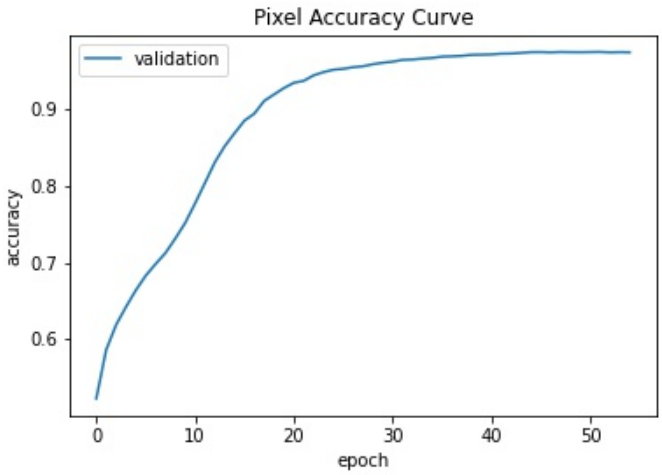

Figure 4.7: Pixel Accuracy

4.3.3 Mean Intersection over Union

Since the mean pixel accuracy is calculated on a pixel-by-pixel basis, when the pixel accuracy value is high, it can be considered that the model has a high prediction accuracy rate for the pixel category. However, the accuracy of single pixel labeling cannot fully represent

the accuracy of the model's semantic segmentation effect. Therefore, it is necessary to introduce a standard that is often used to evaluate the accuracy of the semantic segmentation model, that is, the mean intersection over union (MIoU). The purpose of calculating the intersection over union (IoU) is to find the quotient of the intersection and union of the predicted polygon and the ground truth polygon. The formula for calculating mean intersection over union is shown as follows:

$$MIoU = \frac{1}{k+1}\sum_{i=0}^{k}\frac{p_{ii}}{\sum_{j=0}^{k}p_{ij} + \sum_{j=0}^{k}p_{ji} - P_{ii}}.$$

In the formula, k is the kth image of the whole training set, p_{ii} represents the number of pixels in the intersection of predicted polygons and ground truth polygons, while p_{ij} and p_{ji} represents the number of the pixels in the predicted polygons and ground truth polygons separately. The p_{ij} + p_{ji} - p_{ii} represents the union of predicted polygons and ground truth polygons.

In the experiment, the threshold of MIoU is set to 0.5. Once the MIoU score exceeds the threshold, the result is considered to be effective segmentation, and the model can be thought to have a reliable segmentation ability. After the observation of loss and accuracy curves, the final most accurate unet model got a MIoU score which reached 0.68.

4.4 Interface Experiment

The constructed user interface (UI) is shown in Figure 4.8.

The map of the UI is the map component in the Vue Leaflet component library.

The upper left corner of the UI is the file upload component. By clicking the "select" button, users can upload image files, the format of the file is limited to GeoTiff. Because the GeoTiff format cannot be displayed directly in the browser, the uploaded image needs to be converted to the PNG format through the back-end processing before it can be displayed on the map. In order to determine the geographic location of the picture, the geographic information of the geotiff file needs to be read and processed. Then, through the API, the geographic coordinates of the two points on the upper left and lower right are returned to the front end to define the boundary of the picture, further ensure that the picture is displayed in the correct position on the map.

The multiple colored polygons displayed above the PNG file in the UI are the result of semantic segmentation through the unet model in the back end. As shown in Figure 4.9,

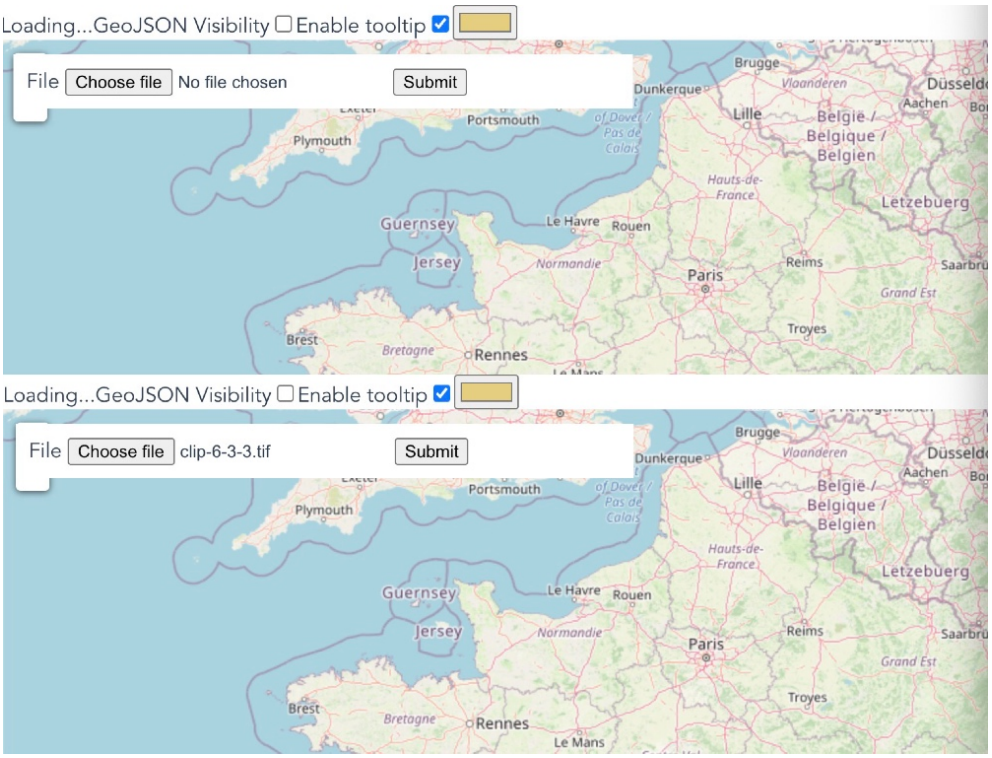

Figure 4.8: UI

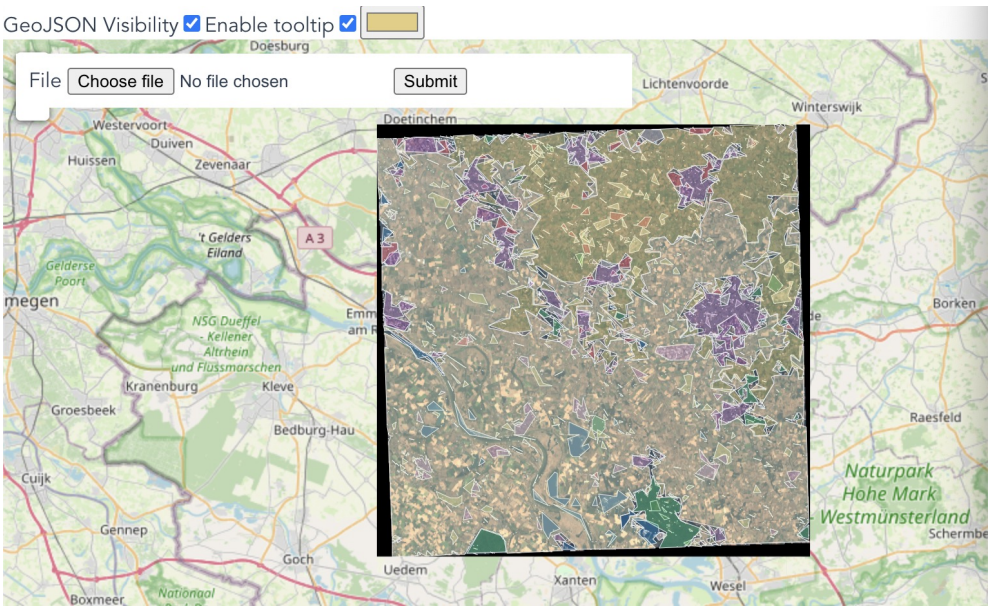

Figure 4.9: Polygons

each polygon represents a segment of the segmentation result, and each color corresponds to a specific label. When hovering over a polygon, the tooltip component will display the label number and id of the polygon. When the mouse clicks on a polygon, the selection box will pop up, as shown in Figure 4.10. The user can select and change the label of the polygon in the drop-down menu, and can also change the shape of the polygon by dragging the points around the polygon. The original polygons are composed of a large number of points, which makes it difficult to read and visualize the polygon data (the large amount of data leads to more reading time and causes web page jams), and is not simple for the user's manual operation (the large number of points will make manual changes to be time-consuming). Therefore, the final decision is to simplify the polygons before visualizing it. All the changes need to be saved to the database through the API by clicking the "save" button.

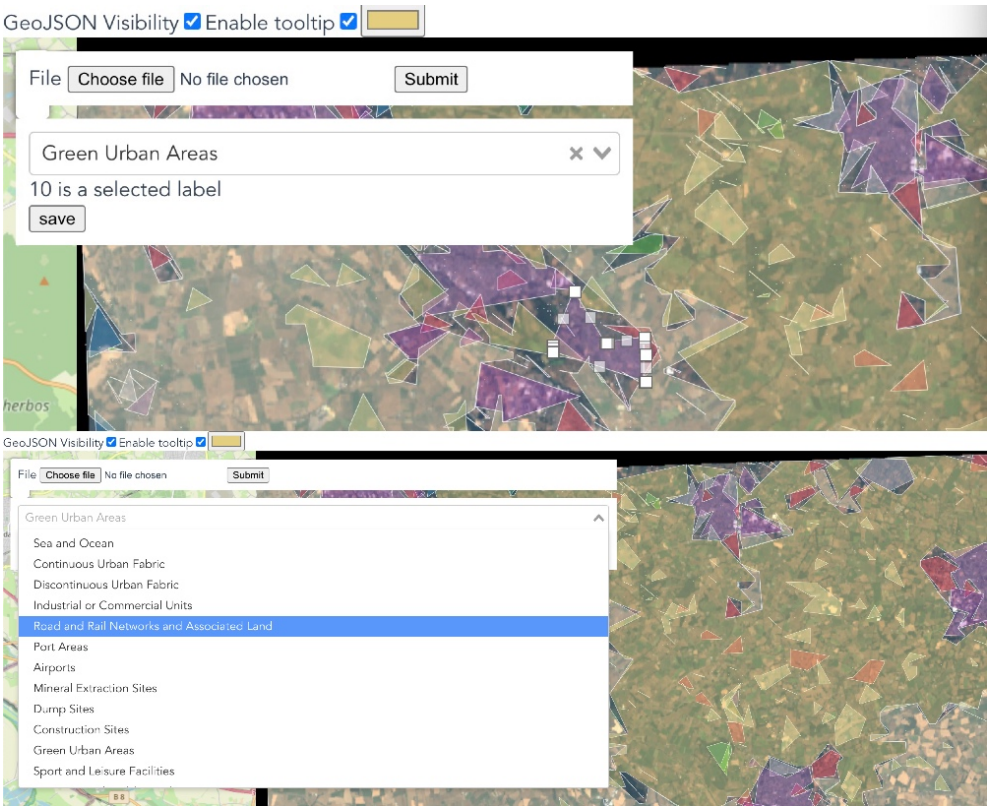

Figure 4.10: Selection Box

It should be noted that the original image and the segmentation result are overlapped and placed on the UI. The transparency of the segmentation result is set to 0.3. This design is to enable experts to observe the original image through the polygon. Experts are able

to use their professional knowledge and experience to judge the type and shape of each area of the original image, as well as make further corrections for the segmentation results manually. Thus the accuracy of the segmentation results reaches nearly 100%.

4.5 Further Training

This section explains the further modeling training part of the research. Since the final goal of this research is to implement the whole system to Africa's dryland, even though the model has achieved relatively accurate result, when applying to different regions, the predicting result can still be not precise enough. Thus, we need to use the modified data from previous steps to further train the machine learning model and improve the accuracy. This step keeps the model accuracy within an acceptable range.

5

Analysis

This chapter will summarize the analysis and discussion of the research. The analysis part mainly includes two subsections. The machine learning algorithm subsection is the subjective analysis of the experimental results obtained in the last chapter, combined with some findings in the experiment, trying to put forward suggestions for improvement of the project. In order to test the versatility and stability of the system, and get some suggestions for user interface improvements, this system has been tested by 10 random users. In the UI test subsection, some suggestions for improving the user interface will be put forward based on the feedback of the users.

5.1 Machine Learning Algorithm

From the analysis of the experimental results, it is clear that the pixel accuracy of the model is very high, which shows that the model has a relatively strong ability to classify individual pixels. However, the MIoU score of the model is relatively low, and through a preliminary analysis of the MIoU calculation formula, it can be considered that the model's ability to recognize the edges of objects is weak. This section attempts to find out the reasons for the low MIoU score through a comprehensive and detailed analysis of the segmentation results, and proposes improvements for the the system.

First, the segmented image is analyzed. Obviously, it can be found that there are some isolated pixels in the semantic segmentation results, that is, these pixels and the pixels distributed around them do not belong to the same category. The following part will analyze the causes of this situation from two perspectives. In the first case, the pixels are correctly marked. If the marking is accurate, there are many possible situations, for example, independent buildings (houses, etc.) in the farmland. In the second case, the pixel marking

is incorrect. This situation may correspond to two sub-cases. The first sub-case, that is, the model's ability to classify pixels is not strong enough, which is a normal phenomenon, since the pixel accuracy of the model has not reached 100%. The second sub-case, that is, there is some noise in the original image, which leads to wrong labeling. The noise problem can be addressed by implementing Gaussian filtering operations during the data preprocessing stage. Gaussian filtering is mainly to smooth the image, which is very effective for suppressing the noise that follows the normal distribution. In addition, dilation and erosion operations are also very helpful for noise processing, but these two operations are mainly applied to binary images for finding their edges, which is not suitable in this case. In short, due to the generation of isolated pixels, it is easy to cause the MIoU score to be low, which can be improved by removing noise.

Then, the parameters and weight values of the model are analyzed. Since the model has only trained and adjusted the weights in the training process and it has achieved a relatively high pixel accuracy, other parameters of the model have not been further fine tuning. In order to further improve the performance of the model, the adjustment of other parameters should also be considered. For example, the dropout parameter, by setting its size, makes the model automatically drop some features during the training process, which can effectively prevent the model from overfitting. It is also very effective to rescale the pixels, dividing the value of each pixel by 255, thereby reducing the scale and increasing the computation speed. In addition, in the experiment, cross-validation is only used to determine the accuracy of the model, but cross-validation can actually be used to adjust the parameters to improve the segmentation effect. The results can also be improved by replacing the optimizer, etc.

Further, replacing the model can also effectively improve the segmentation effect. The main reason for choosing VGG16 as the basic model in this study is that VGG16 has sufficient depth and fast training speed. Because the training of the model is done locally, the training speed is limited by the CPU and GPU, so that a model with a relatively small amount of calculation needs to be selected. If it is not affected by the factors of CPU and GPU, the use of some other models should be considered. For example, the ResNet model is characterized by low computational complexity, and as the network depth increases, the training effect generally continues to rise. Therefore, the deeper the ResNet network structure, the better the segmentation results. In addition, the SegNet semantic segmentation model based on VGG16 is also a model specifically applied to semantic segmentation tasks. Its encoder part is the same as the convolutional layer of VGG16. The decoder uses the

max-pooling indices received from the corresponding encoder to input non-linear upsampling of feature maps. Rebuilding the decoder part also has the opportunity to improve the model performance.
Finally, combining with other algorithms to improve the accuracy of segmentation can be tried. Because the MIoU score of this study is low, in order to improve this, we should consider combining a relatively good boundary recognition algorithm. Watershed algorithm is a widely used algorithm for detecting edges, which has achieved satisfactory results in many boundary recognition tasks. It may be considered to apply the watershed algorithm to first detect the coarse-grained edges of the image, and then use the CNN algorithm to further correct the boundary. Applying this method requires the same processing on the training set to ensure the accuracy of the model.

5.2 User Interface

In order to explore whether the front-end interface of the system can meet the needs of users, the study randomly selected 10 users to test the interface. These ten users include five related major users and five unrelated major users, and the users are all between 22 and 55 years old. The purpose of randomly selecting users in this way is to get comprehensive and meaningful feedback. The setting of the user's age range is to take into account that the system should maintain user-friendliness for users of different ages, and can not only meet the needs of relatively young or old users. While the meaning of selecting users from different majors is that the system needs both technical suggestions from professionals, as well as advice from general users. The content of the test is mainly to allow users to operate and use the UI without understanding the system in advance. The purpose of this task is to test whether the user interface is easy to operate and concise to help users understand the various functions of the system.
The research summarizes the advantages and disadvantages of the user interface based on user feedback, which will be described in detail below.

1. Advantages:

   (i) The layout of each component on the interface is simple and clear, easy to understand and use.

   (ii) The interface is beautiful, the use of maps and the selection of color of labeled polygons are reasonable. Users can make a preliminary judgment on the land

type based on the color of the polygon, such as green representing forest or vegetation-related categories.

(iii) The labeled polygon is transparent, this design is very helpful for reshape polygon.

2. Disadvantages:

(i) After uploading an image, there is no indication of the progress of the image upload, so it is impossible to intuitively understand whether the image was uploaded successfully.

(ii) The predicted polygons are not directly connected, which means there are some gaps between the polygons, so part of the land is not covered, which makes manual adjustment tasks more numerous.

(iii) The transparency setting is not low enough, which makes it difficult to see the original image corresponding to some polygons.

(iv) The colors of various components on the interface are mainly white, if they are designed into some different styles, UI will be more beautiful.

(v) Occasionally, there are webpage stutters during exploration.

In general, users believe that the system has achieved the goals set by expectations, and also meet the needs of simplicity in the design of the webpage, and have a certain aesthetics. However, the design of some details can be further improved. Based on the above feedback, the research attempts to propose some solutions that can improve the interface in the future.

The improvement plan is as follows:

(i) Add a popup component, which will pop up after clicking the "Submit" button and display the upload progress in the form of a percentage number. After the progress reaches 100%, the "Upload Successful" prompt will be displayed.

(ii) Reduce the transparency of the polygon to 0.2 to ensure that the user can clearly see the original picture.

(iii) Design the style of the component according to the characteristics and functions of each component, change the color, shape, etc. For example, the color corresponding to the category can be displayed in front of each category name in the drop-down menu, thereby providing users with a more intuitive sensory experience.

(iv) Due to the huge number of points that make up the original polygon after prediction, which is not conducive to manual operation and display, the system uses a polygon simplification algorithm. The advantage of this algorithm is to reduce the number of points that make up the polygon, but it also causes some gaps between the polygons. Research has not yet come up with a better way to address these two problems at the same time, in the future a better solution is expected to be proposed.

(v) The stuttering phenomenon can be improved by increasing the processing efficiency of the CPU and GPU. Stuttering may also be caused by the computer's cache, which can also be improved by clearing the cache.

6

Discussion

According to the experience in the experiment and the analysis of the results, this chapter will give a general discussion of the system.

The U-net model trained in the experiment has achieved relatively good results, but it can still be improved. By analyzing the whole back end of the system, to further improve the segmentation performance, a few ideas are mentioned. First, make more effort in the data preprocessing period. Second, implement fine tuning with other parameters may also help. Third, rebuild or restructure the CNN model. Finally, combine the unet model with other algorithms. The methods proposed above have the opportunity to achieve breakthroughs in improving MIoU scores.

The user interface also reached our expectation, but based on the feedback from user test, some details can be further improved. Such as the style of components, the gaps between polygons, the transparency of labeled polygons, the stutters during exploration, etc..

In conclusion, the system reached our expectation and the performance is satisfying, but improvements can still be made on both the front end and the back end.

The further work of this research is to implement this system to a specific rural area, such as Africa. The challenge of it is that we need to expand the domain knowledge base and architecture, which requires ontology data from local experts. Also, the modified data needs to be applied, as training data, to the system, the accuracy of the further trained model needs to be evaluated.

7

Conclusion

The purpose of this study is to develop a universal object recognition and segmentation method combining machine intelligence and human intelligence. At the same time, this method is expected to solve the problem of low efficiency and a lot of manpower consumption of the traditional method. Based on this research, the two proposed sub-questions attempt to be answered, which are:

Sub-RQ1: How can we design a promising approach for the interpretation of new objects on satellite images based on various techniques to identify image patterns?

Sub-RQ2: How can we include (local) expert knowledge to make a proper improvements to the models and obtain better results for the interpretation of land use?

In the process of designing and developing the system, the research was divided into two parts. The first part is to realize the participation of machine intelligence in the system through machine learning algorithms. The deep learning algorithm used is CNN, the basic model is VGG16, and the semantic segmentation model is a U-net model built on VGG16. The pixel accuracy of the U-net model reached 97.4%, and the MIoU score reached 0.68. The experimental results meet the basic needs, but still need to be further improved. The second part is to realize the participation of human intelligence in the system by building a front-end customer interface. The interface has achieved the expected effect, but based on user test feedback, the details of the UI can be further improved.

The study answered two sub-questions, namely:

Sub-RQ1: Experiments need to use both deep learning model and front-end technology to achieve better segmentation accuracy.

Sub-RQ2: Expert knowledge has improved the segmentation results by interacting with the front end.

In summary, the research achieved the basic purpose and found an object recognition and

segmentation technology that is superior to traditional methods. In the next step of research, we will improve from both the algorithm and the UI sides, hoping that the system can perform better in the future.

References

[1] V. Jayaraman Ranganath R. Navalgund and P. S. Roy. **Remote sensing applications: An overview**. *CURRENT SCIENCE*, **93**:1747–1766, 12 2007. 5

[2] M. Thiel T. Landmann G. Forkuor, C. Conrad and B. Barry. **Evaluating the sequential masking classification approach for improving crop discrimination in the Sudanian Savanna of West Africa**. *Computers and Electronics in Agriculture*, **118**:380–389, 10 2015. 7, 8

[3] John Rogan and Dongmei Chen. **Remote Sensing technology for mapping and monitoring land-cover and land-use change**. *Progress in Planning - PROG PLANN*, **61**:301–325, 5 2004. 8

[4] G. Duveiller F. Low. **Defining the spatial resolution requirements for crop identification using optical remote sensing**. *Remote Sensing*, **6**:9034–9063, 9 2014. 8

[5] J.R. Jensen. *Remote Sensing of the Environment: an Earth Resource Perspective*. Pearson, 5 2006. 8

[6] Germain Forestier Cedric Wemmert Pierre Gancarski et al. Nicolas Durand, Sebastien Derivaux. **Ontology-based Object Recognition for Remote Sensing Image Interpretation**. *IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2007)*, pages 472–479, 10 2007. 8, 9, 16

[7] I. Bloch C. Hudelot, J. Atif. **Fuzzy spatial relation ontology for image interpretation**. *Fuzzy Sets and Systems*, **159**:1929–1951, 8 2008. 8, 16

[8] R. Benjamins R. Studer and D. Fensel. **Knowledge engineering: Principles and methods**. *Data Knowledge Engineering*, **25**:161–197, 5 1998. 8

[9] Cédric Wemmert Pierre Gançarski Germain Forestier, Anne Puissant. **Knowledge-based region labeling for remote sensing image interpretation**. *Computers, Environment and Urban Systems*, **36**:470 – 480, 1 2012. 9, 16, 17, 20

[10] F. C. Monteiro and A. Campilho. **Watershed Framework to Region-based Image Segmentation**. *ICPR 2008 19th International Conference on Pattern Recognition*, **1**, 12 2008. 9

[11] Cédric Wemmert Sébastien Derivaux, Germain Forestier and Sébastien Lefèvre. **Supervised image segmentation using watershed transform, fuzzy classification and evolutionary computation**. *Pattern Recognition Letters*, **31**:2364–2374, 11 2010. 9

[12] Nicos Maglaveras Kostas Haris, Serafim N. Efstratiadis and IEEE Aggelos K. Katsaggelos, Fellow. **Hybrid Image Segmentation Using Watersheds and Fast Region Merging**. *IEEE Transactions on Image Processing*, **7**:1684–1699, 12 1998. 9

[13] ZHANG Risheng; ZHU Guibin; ZHANG Yanqin; CHEN Weijing. **Method of Satellite Images Region Segmentation and Recognition Based on CNN and Gradient Watershed Algorithm**. *Infrared Technology*, **39**:1114–1119, 12 2017. 10, 11

[14] Gerhard Rigoll Martin Hofmann, Philipp Tiefenbacher. **Background Segmentation with Feedback: The Pixel-Based Adaptive Segmenter**. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, **1**:38–43, 6 2012. 10

[15] J. Konrad P.-M. Jodoin, F. Porikli and P. Ishwar. **Changedetection.net: A New Change Detection Benchmark Dataset**. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 6 2012. 10

[16] Ming-Ni Wu ; Chia-Chen Lin ; Chin-Chen Chang. **Brain Tumor Detection Using Color-Based K-Means Clustering Segmentation**. *IEEE Comput Soc*, **2**:245–250, 12 2007. 10

[17] Iasonas Kokkinos Kevin Murphy-Alan L. Yuille Liang-Chieh Chen, George Papandreou. **Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs**. *CoRR. arXive*, 12 2014. 10, 13

[18] Khumanthem Manglem Nameirakpam Dhanachandra and Yambem Jina Chanu. **Image Segmentation using K -means Clustering Algorithm and Subtractive Clustering Algorithm**. *Eleventh International Multi-Conference on Information Processing-2015 (IMCIP-2015)*, **54**:764 – 771, 2015. 10

[19] Francisco A. R. Alencar; Carlos Massera Filho; Diego Gomes; Denis F. Wolf. **Pedestrian classification using K-means and Random Decision Forests**. *2014 Joint Conference on Robotics: SBR-LARS Robotics Symposium and Robocontrol*, pages 103–108, 10 2014. 11

[20] 11

[21] DONG Jun ZHOU Fei-Yan, JIN Lin-Peng. **Review of Convolutional Neural Network**. *CHINESE JOURNAL OF COMPUTERS*, **40**:1229–1251, 7 2017. 12

[22] Wang Wei Cai Shuo, Hu Hangtao. **Semantic Segmentation of High-Resolution Remote Sensing Image Based on Deep Convolutional Network**. *Journal of Signal Processing*, **35**:2010–2016, 12 2019. 14

[23] WANG Zhaohai ZHONG Yanfei DONG Huaping ZHOU Songtao CHENG Buyi WANG Jinchuan, TAN Xicheng. **Faster R-CNN Deep Learning Network Based Object Recognition of Remote Sensing Image**. *Journal of Geo-information Science*, **20**:1500–1508, 2018. 14, 15

[24] Ross Girshick. **Fast R-CNN**. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 12 2015. 14

[25] T. Matsuyama and V.-S. Hwang. *SIGMA - A Knowledge-Based Aerial Image Understanding System*. Plenum Press New York USA, 1990. 16

[26] 16

[27] Otthein Herzog Jean-Pierre Schober, Thorsten Hermes. *Content-based Image Retrieval by Ontology-based Object Recognition*. 01 2004. 16

[28] N. Maillot, M. Thonnat, and C. Hudelot. *Ontology based object learning and recognition : Application to image retrieval*. 2004. 16

[29] J. Feranec M. Bossard and J. Otahel. **Corine Land Cover—Technical Guide**. 2000. 20