


Article

Model Construction and System Design of Natural Grassland-Type Recognition Based on Deep Learning

Yangjing Xiu ^{1,†}, Jing Ge ^{1,2,*}, Mengjing Hou ¹ , Qisheng Feng ¹, Tiangang Liang ¹, Rui Guo ¹, Jigui Chen ³ and Qing Wang ³

- ¹ State Key Laboratory of Herbage Improvement and Grassland Agro-Ecosystems, Key Laboratory of Grassland Livestock Industry Innovation, Ministry of Agriculture and Rural Affairs, Engineering Research Center of Grassland Industry, Ministry of Education, College of Pastoral Agriculture Science and Technology, Lanzhou University, Lanzhou 730000, China
 - ² Key Laboratory of Western China's Environmental Systems (Ministry of Education), College of Earth and Environmental Sciences, Lanzhou University, Lanzhou 730000, China
 - ³ Grassland Station of Menyuan Hui Autonomous County, Haibei 812200, China
- * Correspondence: gejing@lzu.edu.cn
- † These authors contributed equally to this work.

Abstract: As an essential basic function of grassland resource surveys, grassland-type recognition is of great importance in both theoretical research and practical applications. For a long time, grassland-type recognition has mainly relied on two methods: manual recognition and remote sensing recognition. Among them, manual recognition is time-consuming and laborious, and easily affected by the level of expertise of the investigator, whereas remote sensing recognition is limited by the spatial resolution of satellite images, and is not suitable for use in field surveys. In recent years, deep learning techniques have been widely used in the image recognition field, but the application of deep learning in the field of grassland-type recognition needs to be further explored. Based on a large number of field and web-crawled grassland images, grassland-type recognition models are constructed using the PyTorch deep learning framework. During model construction, a large amount of knowledge learned by the VGG-19 model on the ImageNet dataset is transferred to the task of grassland-type recognition by the transfer learning method. By comparing the performances of models with different initial learning rates and whether or not data augmentation is used, an optimal grassland-type recognition model is established. Based on the optimal model, grassland resource-type map, and meteorological data, PyQt5 is used to design and develop a grassland-type recognition system that uses user-uploaded grassland images and the images' location information to comprehensively recognize grassland types. The results of this study showed that: (1) When the initial learning rate was set to 0.01, the model recognition accuracy was better than that of the models using initial learning rates of 0.1, 0.05, 0.005, and 0.001. Setting a reasonable initial learning rate helps the model quickly reach optimal performance and can effectively avoid variations in the model. (2) Data augmentation increases the diversity of data, reducing the overfitting of the model; recognition accuracies of the models constructed using the augmented data can be improved by 3.07–4.88%. (3) When the initial learning rate was 0.01, modeling with augmented data and with a training epoch = 30, the model performance reached its peak—the TOP1 accuracy of the model was 78.32% and the TOP5 accuracy of the model was 91.27%. (4) Among the 18 grassland types, the recognition accuracy of each grassland type reached over 70.00%, and the probability of misclassification among most of the grassland types was less than 5.00%. (5) The grassland-type recognition system incorporates two reference grassland types to further improve the accuracy of grassland-type recognition; the accuracy of the two reference grassland types was 72.82% and 75.01%, respectively. The recognition system has the advantages of convenient information acquisition, good visualization, easy operation, and high stability, which provides a new approach for the intelligent recognition of grassland types using grassland images taken in a field survey.



Citation: Xiu, Y.; Ge, J.; Hou, M.; Feng, Q.; Liang, T.; Guo, R.; Chen, J.; Wang, Q. Model Construction and System Design of Natural Grassland-Type Recognition Based on Deep Learning. *Remote Sens.* **2023**, *15*, 1045. <https://doi.org/10.3390/rs15041045>

Academic Editor: Dino Ienco

Received: 14 December 2022

Revised: 2 February 2023

Accepted: 11 February 2023

Published: 14 February 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: grassland-type recognition; field grassland images; PyTorch; transfer learning; recognition system

1. Introduction

China's grassland resources are rich and diverse, with a total area of $265 \times 10^4 \text{ km}^2$ (https://mnr.gov.cn/dt/ywbb/202108/t20210826_2678340.html, accessed on 13 February 2023). As the largest terrestrial ecosystem in China [1,2], grassland ecosystems have a variety of ecological functions, such as climate regulation, water conservation, soil and water conservation, carbon and nitrogen fixation, and biodiversity maintenance. At the same time, grasslands also provide a large number of production and living resources for human beings, such as livestock products, plant resources, and tourism resources, which are important for the sustainable development of human society [3,4]. Since China's reforms and opening-up in the 1980s, the utilization of grassland resources has been increasing, and the irresponsible utilization phenomena of grassland, such as over-grazing and over-cultivation, have existed for a long time in large areas [5,6]. By the beginning of this century, the Chinese government had identified the problem and developed a series of grassland conservation projects and policies; these grassland conservation measures have effectively curbed the degradation of natural grassland in China and significantly improved the grassland ecosystem in most areas of China [7,8].

As a basic tool for monitoring and evaluating grasslands, the grassland resource survey provides a comprehensive and systematic understanding of the basic conditions of grasslands through the investigation of variables, such as natural resources, grassland types, areas, spatial distribution ranges, and production and management status, etc., [7]. Assessing and developing reasonable and effective grassland use and conservation measures helps maintain the health and stability of grassland ecosystems and promotes the sustainable use and development of grassland natural resources [9]. Among them, grassland-type recognition, as an important part of the grassland resource survey, is an important basic function in understanding and studying the natural and economic characteristics of grassland resources. Grassland-type recognition helps to understand the principles under which grassland occurs, the development of grassland, and how to rationalize the use, development, and cultivation of grassland resources [10].

Currently, the main methods of grassland-type recognition are manual recognition and remote sensing image recognition. Manual recognition mainly relies on the experience of professionals, through a combination of visual recognition and geographic location information to arrive at a comprehensive judgment. This method requires that investigators have a high level of expertise and training on the techniques required, so there are a series of disadvantages, such as low efficiency, high costs in money and time, and difficulty with information sharing. In addition, due to the different knowledge bases of investigators, there is some inevitable subjectivity in the recognition process. Remote sensing image recognition mainly uses the spectral signatures of different types of grassland and the difference in sensitivity to vegetation indices to construct a classification decision method for grassland types that can be used to complete classification and recognition [11]. This method has strengths such as easy data acquisition, wide coverage, and suitability for grassland-type classification at a large scale. For example, Guo et al. (2011) used 23 MODIS NDVI time series data in 2009 to complete the recognition of grassland types in northern Tibet using an unsupervised classification method, which proved the feasibility of using remote sensing images for grassland-type recognition in northern Tibet [12]. Sun et al. (2012) used Landsat TM imagery and a decision tree classification method, based on the wave characteristics of different grassland types, and used NDVI data to finally complete the remote sensing recognition of grassland types in the Yarlung Tsangpo River source area [13]. Although existing studies have confirmed the advantages of remote sensing image recognition in large-scale grassland classification, the recognition accuracy based

on the remote sensing method is greatly influenced by the spatial resolution of satellite images. Low and medium-resolution remote sensing images are not suitable for the refined recognition of grassland typing, and high-resolution satellite images are usually costly to use, and the preprocessing is complicated and time-consuming. Additionally, the acquisition of satellite images often has a certain lag and cannot achieve real-time recognition of grassland types. Therefore, this method is not suitable for use in field surveys.

Some existing studies have shown that deep learning technology applied to remote sensing classification works [14,15]. With the continuous development of computer and image processing technologies, deep learning technology can also be effectively applied to the classification and recognition of images. Compared with traditional machine learning-based image recognition, the advantage of deep learning is that it eliminates the step of extracting image features artificially, with the algorithm automatically conducting feature extraction, which reduces human interference, greatly reduces workload, and improves recognition efficiency based on ensuring recognition accuracy [16]. Sun et al. (2017) completed the recognition of 100 ornamental plants based on the improved ResNet model, and the result showed that the recognition accuracy reached 91.78%; Liu et al. (2019) completed the image recognition of 103 chrysanthemum species based on the VGG-16 model with a verified accuracy of 89.43%; Li et al. (2022) completed the recognition of 24 typical desert plant species based on the RegNetX_8GF model, and the result showed that the recognition accuracy reached 78.33%; Mu et al. (2022) integrated the feature pyramid network (FPN) and ResNeXt model to complete the image recognition of nine weeds in farmland, and the verified accuracy was >95% [17–20]. Although deep learning algorithms have made good progress in the field of plant image classification and recognition, deep learning techniques are still in the preliminary exploration stage in the field of grassland-type recognition research. In addition, most of the existing plant image classification and recognition studies only focus on the predicted results of the model and lack other auxiliary judgment criteria. For classification and recognition studies with a large number of categories and a high degree of similarity between categories (e.g., grassland-type recognition), it is difficult to guarantee the accuracy of recognition for the image type. Therefore, research into the image recognition of grassland types based on deep learning algorithms needs to be further explored, and in particular, a comprehensive classification and recognition method for grassland types suitable for grassland resource field surveys is urgently required.

Considering the above factors, this study is based on a large amount of grassland image data taken in the field during the growing seasons of 2018–2021, with the grassland image data augmented by web crawling and image transformation. This data was then processed using the PyTorch deep learning framework, the VGG-19 model, and the transfer learning method in order to construct a grassland-type recognition model and to evaluate the accuracy of the model for each grassland type. On this basis, we designed and developed a grassland-type recognition system that uses the recognition results of the model, grassland resource-type map, and meteorological data to comprehensively identify grassland images uploaded by users, providing a new approach and technical support for the intelligent recognition of grassland types in grassland field surveys conveniently and reliably.

2. Materials and Methods

2.1. Principles of Grassland Classification

Currently, there are two major systems for classifying grassland in China: the habitat classification system and the comprehensive sequential classification system [21–23]. This study used the grassland habitat classification system, which classifies natural grassland in China into 18 major categories, which is also the number adopted nationally in the principles of grassland classification established by the Chinese Ministry of Agriculture in 1987. This system first classifies natural grassland into zonal grassland and non-zonal grassland. Zonal grasslands are divided into 16 categories according to moisture and heat conditions and types of grassland vegetation (heat conditions and moisture conditions

are divided according to climatic heat zones and Ivanov wetness index, respectively; grassland vegetation is divided into five types: meadow, steppe, desert, wetland, and shrub): temperate meadow steppe, temperate steppe, temperate desert steppe, temperate steppe desert, temperate desert, mountain meadow, alpine meadow steppe, alpine steppe, alpine meadow, alpine desert, alpine desert steppe, warm tussock, warm shrub tussock, tropical tussock, tropical shrub tussock, and savanna. Non-zonal grasslands, which are classified according to moisture condition and grassland vegetation type, are divided into two categories: wetland and lowland meadows [24].

2.2. Grassland Image Data Acquisition and Preprocessing

2.2.1. Data Acquisition

The acquisition of grassland image data included both field photography and web crawling. The field photography data were collected from July to September of 2018–2021, and these data were photographed at an appropriate distance by digital camera or cell phone. The acquired images are required to have grassland as the main subject, show the full view of grassland, and have simple backgrounds. The images also need to be clear with few interfering objects (examples are shown in Figure 1). After collating and analyzing the data, a total of 3307 images taken in the field were obtained, covering the 18 types of grassland mentioned in Section 2.1. Since the deep learning method relies heavily on a large amount of labeled data for training, a further 4096 images of grassland were obtained by web crawling, involving 16 types of grassland, with the aim of improving the model's prediction accuracy. In total, 7403 images of grassland were obtained. The data from field photography and web crawling were uploaded to the Grassland Resource Monitoring and Intelligent Analysis System, where experienced experts identified the grassland types of the images in the background.

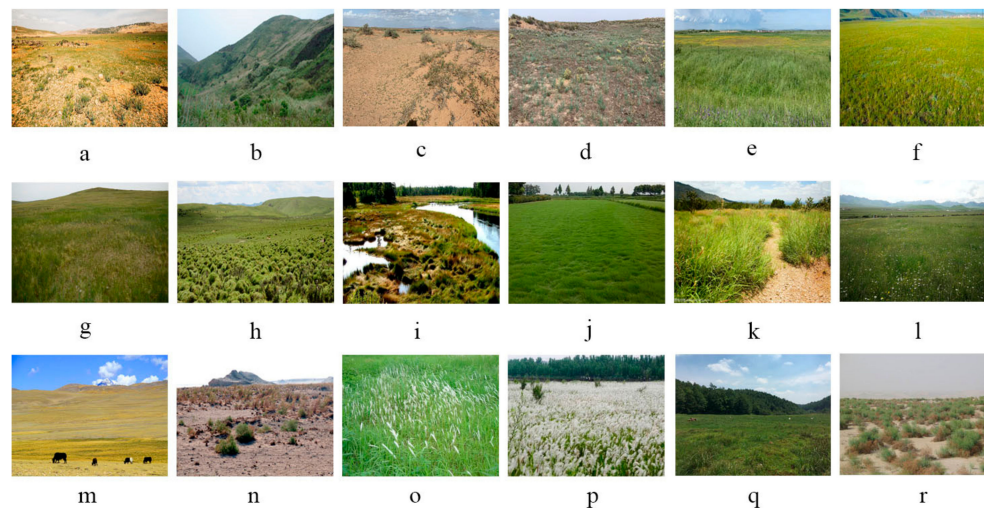


Figure 1. Grassland image data (examples): (a) alpine desert steppe, (b) mountain meadow, (c) temperate steppe desert, (d) temperate desert steppe, (e) temperate meadow steppe, (f) tropical tussock, (g) temperate steppe, (h) alpine meadow, (i) wetland, (j) lowland meadow, (k) savanna, (l) alpine meadow steppe, (m) alpine steppe, (n) alpine desert, (o) warm tussock, (p) warm shrub tussock, (q) tropical shrub tussock, and (r) temperate desert.

2.2.2. Data Cleaning and Segmentation

Data cleaning aims to address problems in the original data by filling in missing values, smoothing noisy data, removing abnormal values, etc., and optimizing the data structure [25]. Of these, removing abnormal values is the easiest to use. In the study, we manually screened all acquired grassland image data and removed data that did not meet the model training criteria. Finally, 400 images of each type of grassland were retained, for a total of 7200 images of the 18 grassland types.

In the cleaned grassland image data, each grassland type was randomly allocated to the training set, the validation set, and the test set in a ratio of 8:1:1. The training set had a total of 5760 images (320 images of each grassland type), the validation set a total of 720 images (40 images of each grassland type), and the test set a total of 720 images (40 images of each grassland type). The training set was used for creating the grassland-type recognition model, the validation set was used to adjust the hyperparameters of the model during training and to evaluate the overall recognition accuracy of the model, and the test set was used to evaluate the recognition accuracy of the optimal model for each grassland type.

2.2.3. Data Augmentation

Judging that the data set of grassland images that had been obtained was still not large enough for a deep learning algorithm and aiming to improve the recognition accuracy of the model, this study used data augmentation to supplement the grassland images in the training set [26]. Using the transforms tool in the torchvision image processing library, when data were inputted into the model, images were first adjusted to 256×256 pixels and then randomly cropped to 224×224 pixels. This step is equivalent to expanding the data set $32 \times 32 = 1024$ times. Then, images were randomly flipped horizontally, flipped vertically, rotated, and transformed in brightness, contrast ratio, and saturation (an example is shown in Figure 2). The above image transformation operations effectively complement the grassland image training set, which can reduce overfitting in model training and improve the generalization performance of the model. This study compared the differences between modeling with the original data and modeling with the augmented data to quantify the positive impact of data augmentation on model recognition accuracy. For the original data without augmentation, the images needed to be centrally cropped to 224×224 pixels to meet the algorithm's input image pixel requirements.

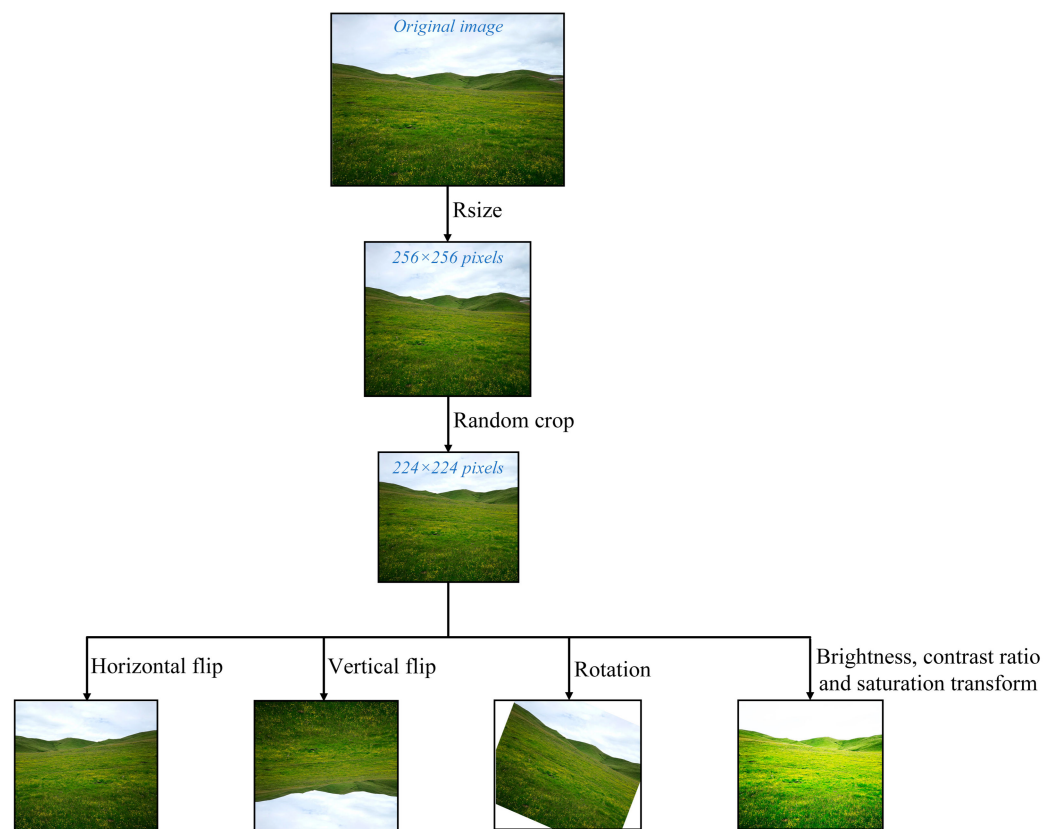


Figure 2. Examples of image augmentation.

2.3. Reference Grassland Types

Because of the large number of grassland types involved in this study and the high similarity between some grassland types, we chose two spatial distribution maps of grassland types as auxiliary judgments for the recognition results of the grassland-type recognition model: one is the reference grassland type 1 based on the Chinese grassland resource-type map, and the other is the reference grassland type 2 based on the classification of grassland according to meteorological data.

2.3.1. Chinese Grassland Resource-Type Map

The Chinese grassland resource-type map was downloaded from the Resource and Environmental Science and Data Center of the Chinese Academy of Science (<https://www.resdc.cn>, accessed on 16 March 2022) as 1:1,000,000 Chinese grassland resource-type vector data. This data was obtained by digitizing the results of the field survey of Chinese grassland resources from 1978 to 1985 [27]. The grassland classification principle of this data also adopts the grassland habitat classification system, and its collation and digitization process adopt the method of multiple checks by validators. The data are of good quality, reliable accuracy, and accurately reflect the grassland types in China in the 1980s. However, nearly 40 years have passed since the development of the Chinese grassland resource-type map, and some grassland types have inevitably changed during this period; this data cannot accurately reflect the natural grassland types in China in recent years, so it is only used as an auxiliary judgment for the recognition results of the grassland-type recognition model constructed in this study.

2.3.2. Classification of Grassland Based on Meteorological Data

Grassland types are significantly influenced by meteorological factors [28]; therefore, the classification of grassland based on meteorological data was chosen as the second reference for grassland types in this study (the specific classification principles are shown in Table 1). The meteorological data used included average monthly temperatures, the average temperature of the hottest month, annual precipitation, the average annual temperature, the annual relative humidity, and $>0\text{ }^{\circ}\text{C}$ annual cumulative temperature for each year from 2018 to 2021, where the data for average monthly temperatures, the average temperature of the hottest month, annual precipitation, the average annual temperature, and the annual relative humidity were downloaded from the National Earth System Science Data Center Shared Services Platform (<http://www.geodata.cn>, accessed on 16 March 2022), with a spatial resolution of 1 km. The data for the $>0\text{ }^{\circ}\text{C}$ annual cumulative temperature was calculated in ArcGIS 10.2 software based on the data of average monthly temperature from 2018 to 2021. The Ivanov wetness index was calculated from annual precipitation, average annual temperatures, and annual relative humidity, using the formula [27]:

$$K = \frac{r}{E_0} = \frac{r}{[0.0018 \cdot (25 + t)^2 \cdot (100 - f)]} \quad (1)$$

where K is the Ivanov wetness index, r is the annual precipitation (mm), E_0 is the evaporation force, t is the average annual temperature ($^{\circ}\text{C}$), and f is the annual relative humidity (%).

Table 1. Meteorological classification indicators for grassland types.

Steppe Type	Average Temperature of the Hottest Month/°C	>0 °C Annual Cumulative Temperature/°C	Ivanov Wetness Index
Temperate meadow steppe	10–22	1500–3900	0.60–1.00
Temperate steppe	10–22	1500–3900	0.30–0.60
Temperate desert steppe	10–22	1500–3900	0.20–0.30
Temperate steppe desert	10–22	1500–3900	0.13–0.20
Temperate desert	10–22	1500–3900	0–0.13
Mountain meadow	10–22	1500–3900	>1.00
Alpine meadow steppe	6–10	0–1500	0.60–1.00
Alpine steppe	6–10	0–1500	0.30–0.60
Alpine meadow	6–10	0–1500	>1.00
Alpine desert	6–10	0–1500	0–0.13
Alpine desert steppe	6–10	0–1500	0.20–0.30
Warm tussock	22–26	3900–4800	>1.00
Warm shrub tussock	22–26	3900–4800	>1.00
Tropical tussock	26–28	>4800	>1.00
Tropical shrub tussock	26–28	>4800	>1.00
Savanna	26–28	>4800	>1.00
Wetland	—	—	—
Lowland meadow	—	—	—

The above classification indicators are referenced from the “Rangeland Resources of China” prepared by the Bureau of Animal Husbandry and Veterinary Medicine of the Ministry of Agriculture of the People’s Republic of China [24].

To avoid encountering climate anomalies in a single year, all meteorological classification indicators were averaged from 2018 to 2021.

2.4. Construction and Evaluation of Grassland-Type Recognition Model

2.4.1. Deep Learning Framework

The framework chosen for this study was PyTorch. PyTorch is a Python version of Torch, an open-source neural network framework released by Facebook Inc. (San Francisco, CA, USA). in 2017. Torch is a very classical tensor library for manipulating multidimensional matrix data, which works well in deep learning and other math-intensive applications; however, because Torch uses Lua as its programming language, it has a limited audience. PyTorch is a deep learning framework using Python as the programming language, with a well-supported system and usable interface. PyTorch is not a simple encapsulation of Torch’s Python interface but refactors all modules on the tensor and adds automatic derivation functions and has become the most popular dynamic neural network today [29]. PyTorch’s design contains three abstract levels: tensor, variable, and module, which are tightly linked and arranged sequentially; besides that, there are no other complex and abstract concepts [30]. PyTorch’s code is simple and easy to understand; the amount of source code is only about one-tenth of TensorFlow, making it easier for users to read. In addition, because PyTorch’s object-oriented design is inherited from Torch, it continues to be flexible and easy to use, especially the design of the interface, API, and other parts are excellent. PyTorch is also more in line with the logical thinking of users, allowing them to focus on their own ideas as much as possible during the development process without being bound by too many frameworks. PyTorch can also support GPU acceleration, which is significantly faster than mainstream frameworks such as Caffe and Keras under the same algorithmic conditions [30].

The basic framework of PyTorch consists of four parts: the application layer, the optimization layer, the network construction layer, and the data storage layer. The application layer contains two modules, Dataset and Dataloader, which are responsible for reading data and loading data, respectively. The optimization layer is responsible for optimizing the network to move closer to the target network, where Autograd is responsible for the

differentiation and Optimizer is responsible for optimizing the network parameters. The core of the network construction layer is the nn module, where Containers are the main body of the nn module, including two parts—the network layer and the functional module. The network layer mainly includes the convolution layer, the pooling layer, and the fully connected layer. The functional module contains a large number of functions. Init is responsible for the initialization of various parameters, and the parameter is responsible for the management of parameters in the network. In addition, there are many common network models integrated into the nn module that can be easily invoked. The data storage layer consists of the Tensor module and the Storage module, which define the data structure and the storage structure for PyTorch operations [31].

2.4.2. VGG-19 Pretrained Model and Transfer Learning

VGG-Net is a convolutional neural network developed by the Visual Geometry Group of Oxford with the participation of Google DeepMind. VGG-Net successfully constructs a 16–19-layer convolutional neural network by repeatedly using 3×3 small convolutional kernels and 2×2 maximum pooling layers. This approach effectively enhances the depth of the network and improves the effectiveness of the neural network to a certain extent while ensuring the same perceptual field [32]. VGG-Net is often used for image feature extraction because of its simple network structure and good generalization performance, which transfers well to other image data. The trained model parameters of this network have been made open source on its website and can be used for training image classification tasks. It is very widely used because it provides good initialized weights. Research shows that a greater number of layers of a convolutional neural network benefit the extraction of image features, thus being more helpful for image recognition [33]. As the neural network with the deepest number of layers in VGG-Net, VGG-19 can obtain better recognition effects with the same data set. Therefore, in this study, the VGG-19 pretrained model was chosen to construct the grassland-type recognition model. The structure of VGG-19 is shown in Figure 3; it contains 16 convolutional layers and 3 fully connected layers. The convolutional layers are divided into five segments, with each segment containing two to four convolutional layers, and a max-pooling layer connected at the end of the segment. The number of convolutional kernels in each segment is the same, and the closer to a fully connected layer, the more convolutional kernels there are [34].

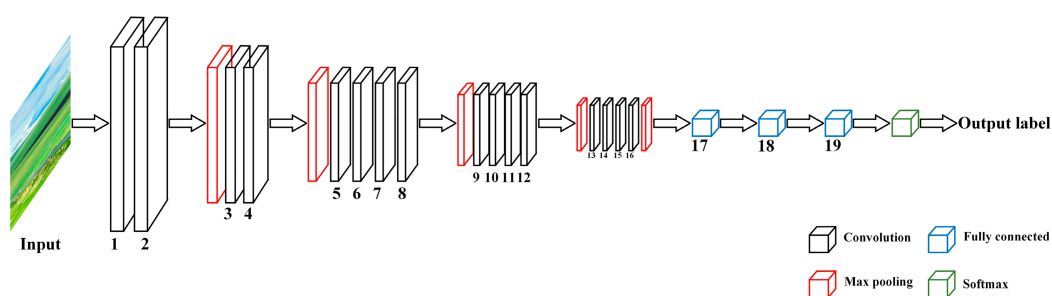


Figure 3. VGG-19 structure. Numbers 1–16 are the convolutional layers, numbers 17–19 are the fully connected layers.

VGG-19 is a pretrained model based on the large image dataset, ImageNet, which can be used for the classification and recognition of grassland types by transferring the large amount of knowledge learned from the ImageNet dataset. There are two main transfer learning methods: feature transfer and parameter transfer. Feature transfer firstly removes the last layer of the pretraining model, treats the whole pretraining model as a feature extractor, makes the image stop at the specified layer after passing through the input layer and forward propagation, and extracts feature vectors obtained before that as features of input images. Parameter transfer means fixing most of the layers of the pretrained model (directly using the weight parameters of the original model) and only initializing the

remaining few layers for the new classification and recognition task [35]. The first transfer learning method is adopted in this study. Transfer learning with VGG-19 is accomplished by reconstructing fully connected layers and fine-tuning them. This method can not only significantly reduce training time but can also effectively reduce the overfitting of the model in the training set and improve the accuracy of the model.

2.4.3. Setting of Main Training Parameters

In this study, the Adam optimizer is used for the optimization of the model, the weight decay is set to the default value of 0.0005, the learning rate is dynamically adjusted using the gradient decay method and combined with experience through a strategy of halving the learning rate every three epochs when the loss decreases, and dynamic monitoring of loss and accuracy during training is conducted using Meta's visualization tool, Visdom. With a reasonable initial learning rate, this method can shorten the training time of the model to reach the optimal solution quickly and effectively avoid variations at the same time [36,37]. The formula is expressed as:

$$lr = lr_0 \times 0.5^{(n/3)} \quad (2)$$

where lr_0 is the initial learning rate, lr is the learning rate after decay, and n is the number of training epochs.

The loss is calculated by the cross-entropy loss function, which is a common loss function used in multi-classification problems to measure the difference between the true value and the predicted value of the model. The formula is expressed as:

$$Loss = - \sum_{i=1}^n y_i \log y'_i \quad (3)$$

where y_i is the true distribution of the i th sample,

y'_i is the model output's distribution of the i th sample, and n is the number of categories.

The batch size is the number of samples fed into the model at each training of the neural network. A large batch size usually leads to faster convergence of the network; however, due to the limitation of memory resources, an overly-large batch size may lead to a lack of memory or the program kernel crashing [38]. Considering the hardware performance and training time, the batch size was set to 64 for both the training set and the validation set. The number of epochs was the number of times the entire training set was fed into the network for training. The maximum training epoch was preset to 60.

2.4.4. Model Preservation

As with TensorFlow, Keras, and other frameworks, PyTorch provides two ways to save a model: either the parameters of the model can be saved or the complete model can be saved. Both of these methods can be implemented by calling the pickle serialization method. In this study, we used the method of saving the model's parameters. As PyTorch's officially recommended model-saving method, it has the advantage of saving storage space and having a fast loading speed. It can also effectively avoid errors and anomalies after the model is adjusted or refactored.

2.4.5. Accuracy Evaluation

The overall recognition accuracy of the grassland-type recognition model was evaluated using the data in the validation set using two evaluation indicators, TOP1 and TOP5. These are monitored on Visdom, where TOP1 indicates the accuracy of the category with the highest prediction result being the same as the real category, and TOP5 indicates the accuracy of the top five categories of predicted results that contain the true category. The

recognition accuracy of each grassland type is evaluated using the data in the test set. The formula used to calculate this is:

$$Accuracy = \frac{m}{n} \quad (4)$$

where *Accuracy* is the recognition accuracy of each grassland type, *m* is the number of correctly predicted samples of a single grassland type in the test set, and *n* is the total number of forecasting samples of a single grassland type in the test set (*n* = 40).

2.5. Design of Grassland-Type Recognition System

Since the deep learning framework used in this study is PyTorch, we used object-oriented programming in Python to write a natural grassland-type recognition system, and we integrated the functions of this system to facilitate the operation and ease of use for users. Currently, the mainstream third-party libraries for program system development in the Python programming language are wxPython, Tkinter, PyQt5, etc., and the library we chose to use in this study was PyQt5. PyQt5 is a Python interface based on Digia's Qt5, which inherits the powerful features of Qt5 with more than 600 classes, 6000 functions and methods, and can run on Windows, Mac OS, and many other platforms. In addition, because of its powerful API, it is widely used in the development of program systems [39].

2.5.1. Design of System Architecture

The grassland-type recognition system consists of four main parts: the front-end UI interface, the information database, the image recognition system, and the image information acquisition system. The program architecture is shown in Figure 4. The front-end UI interface is responsible for the interaction between users and the system, such as loading a model, selecting and displaying images, displaying recognition results, etc. The information database holds text and image explanations for each grassland type, the grassland resource-type map, and the meteorological data (including the average temperature of the hottest month, >0 °C annual cumulative temperature, Ivanov's wetness index, annual precipitation, average annual temperature, and annual relative humidity). The image recognition system is mainly responsible for calling the saved recognition model, classifying and identifying images selected by users, and returning the recognition results. The image information acquisition system is responsible for extracting attribute information for selected images (including size, time, location, latitude, longitude, and altitude) and then extracting the grassland type and meteorological data of the corresponding location from the database.

2.5.2. Design of System Functions

The front-end UI interface includes four functions: model loading, selection and display of images, display of recognition results and image information, and UI controls. A model is loaded by calling the `model.load_state_dict` loading function. Image selection uses the directory tree and file list built by connecting the `QTreeView` control to the `QListView` control. Image display uses the `QLabel` component. The `QTextEdit` component is used to display recognition results or image information. There are five function controls in the main window of the system: "Load model", "Identify", "Clear", "More info", and "Introduction". Clicking the "Load model" button will call the file dialog box (`QFileDialog.getOpenFileName`) to obtain the path of the .pth file and pass the selected file's path to the load function as a parameter. Clicking the "Identify" button will invoke both the image recognition and information acquisition methods defined in the system to recognize selected images and extract attribute information from the images, and it will then display these pieces of information in `QTextEdit`. Clicking the "Clear" button will call the `setVisible(False)` method of `QLabel` and the `clear` method of `QTextEdit` to clear the display of images and recognition results and image information, respectively. Clicking the "More info" button will open a secondary window containing both reference grassland types and meteorological data. Clicking the "Introduction" button will open a secondary window containing text and image explanations for each grassland type.

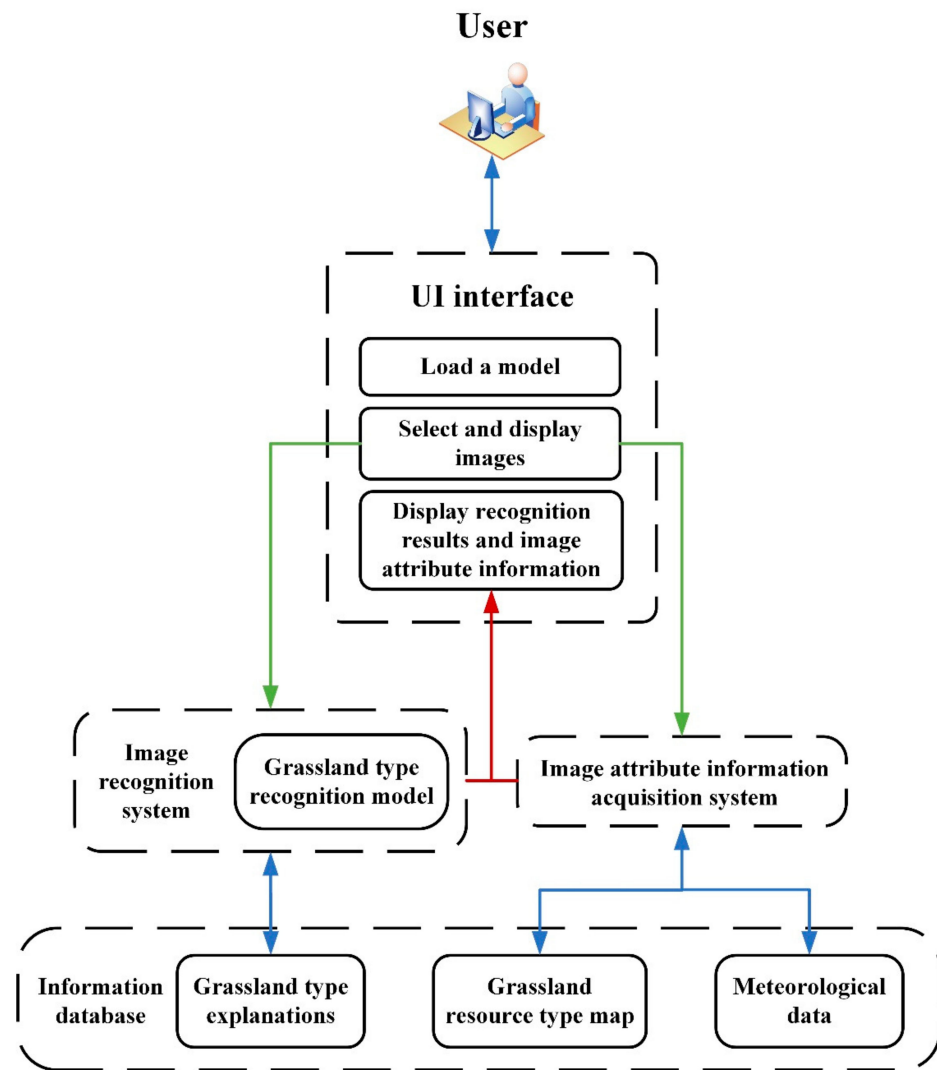


Figure 4. Architecture design of the grassland-type recognition system.

The information database is a MySQL database built with the QtSql module [40]. The text and image data for introducing each grassland type are stored directly in the database table. Create Fishnet in the data management tools of ArcGIS 10.2 is used to divide the national basic vector data into $1\text{km} \times 1\text{km}$ grids, create label points and add latitude and longitude values of each point, and then Extract Multi Values to Points in the spatial analysis module is used to extract the values of the grassland resource-type map and meteorological data at each label point to the corresponding point before the results are finally exported to an Excel sheet and saved to the MySQL database. A connection is established between the database and the front-end UI interface, and data queries are performed with the help of the addDatabase method of the QSqlDatabase class, displaying query results in the QTableView widget of the GUI through the QSqlTableModel class of the UI operational layer. In terms of the database table's design, the use cases are combined with the needs of the system to query and display data conveniently and effectively. The specific structure fields of the table are shown in Table 2.

Table 2. Design of the database storage table structure.

Field	Field Meaning	Field Type	Field Properties
Id	Number	INT	Primary key, NOT FULL
Longitude	Longitude of grid center point	FLOAT	NOT FULL
Latitude	Latitude of grid center point	FLOAT	NOT FULL
Hottest_temp	Average temperature of the hottest month	FLOAT	NOT FULL
Accumulated_temp	>0 °C annual cumulative temperature	FLOAT	NOT FULL
Wetness	Ivanov wetness index	FLOAT	NOT FULL
Precipitation	Annual precipitation	FLOAT	NOT FULL
Average_temp	Annual average temperature	FLOAT	NOT FULL
Humidity	Annual relative humidity	FLOAT	NOT FULL
Class_1	Reference grassland type 1	VARCHAR2	/
Class_2	Reference grassland type 2	VARCHAR2	NOT FULL
Image_loca	Storage path of image explanations	VARCHAR2	NOT FULL
Description	Text explanations	VARCHAR2	NOT FULL

Reference grassland type 1 is the reference grassland type obtained based on the grassland resource-type map, and reference grassland type 2 is the reference grassland type obtained based on the classification of meteorological data.

Image recognition firstly requires loading the trained grassland-type recognition model and calling model.eval, reading the user-selected images with the PIL method, then passing the images as parameters to the torch.max function to obtain the recognition results. Image information acquisition is conducted with the help of Python's exifread library to obtain EXIF of the images, then the geopy library is used to decode EXIF to obtain the attribute information of the images. A Pandas' DataFrame is used to load the data from the .xlsx file and find the row of the grid to which the longitude and latitude of each image belong, then the values in this row are used for the location of where the image was taken.

2.6. Technical Route

Firstly, a series of preprocessing steps, such as data cleaning, are performed on the collected grassland image data, and then based on the PyTorch deep learning framework, a grassland-type recognition model is constructed using the VGG-19 pretraining model and the transfer learning method, then the image recognition performance of the model is comprehensively evaluated. Finally, the optimal grassland-type recognition model, grassland resource-type map, and meteorological data are integrated into the grassland-type recognition system, and on this basis, the system architecture and functions are designed and implemented. The technology roadmap is shown in Figure 5.

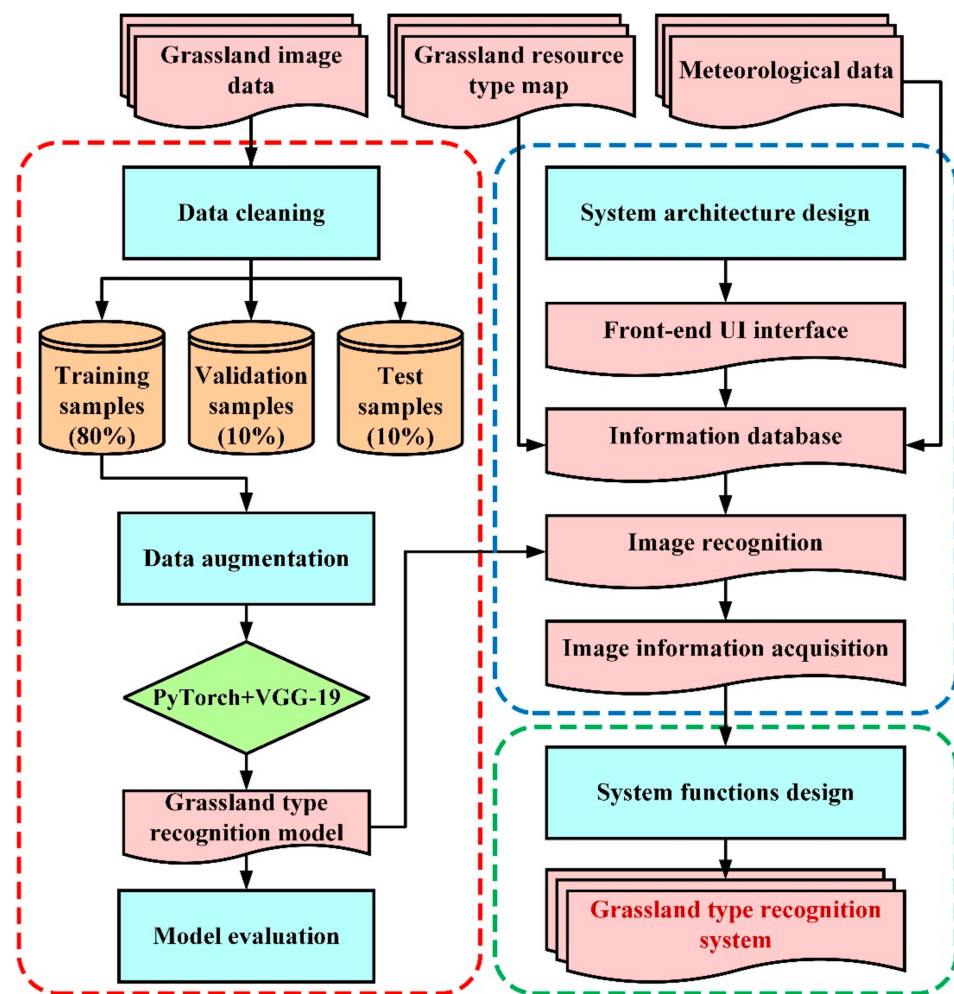


Figure 5. Technology roadmap.

3. Results and Analysis

3.1. Model Construction and Accuracy Evaluation

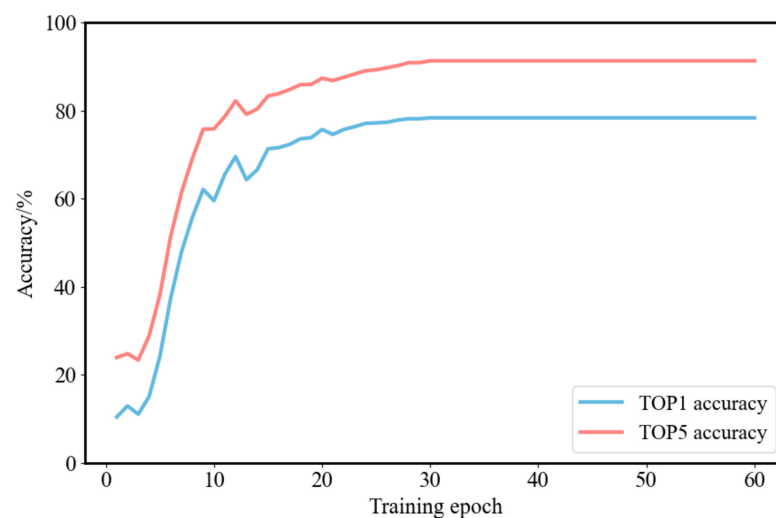
To construct grassland-type recognition models, both unaugmented data and augmented data are used as the input data while combining different initial learning rates (0.1, 0.05, 0.01, 0.005, and 0.001). The result of the accuracy evaluation of ten models is shown in Table 3. Among them, model 6 (using data augmentation and a learning rate of 0.01) has the highest classification accuracy with TOP1 and TOP5 accuracy reaching 78.32% and 91.27%, respectively. Model 5, model 4, model 3, model 8, model 7, model 10, model 9, and model 2 are ranked 2nd to 9th in terms of accuracy with TOP1 accuracy rates of 73.45%, 71.61%, 66.73%, 57.68%, 53.46%, 28.75%, 24.92%, and 19.13%, respectively, and TOP5 accuracy rates of 85.73%, 84.84%, 79.53%, 71.06%, 66.62%, 41.73%, 37.90%, and 32.12%, respectively. Model 1 has the lowest accuracy with TOP1 and TOP5 accuracies of only 16.06% and 29.08%, respectively.

Table 3. Comparison of classification accuracies for grassland-type recognition models.

Model	Whether or Not Data Augmentation Is Used	Learning Rate	TOP1 Accuracy %	TOP5 Accuracy %
1	No	0.1	16.06	29.08
2	Yes		19.13	32.12
3	No	0.05	66.73	79.53
4	Yes		71.61	84.84
5	No	0.01	73.45	85.73
6	Yes		78.32	91.27
7	No	0.005	53.46	66.62
8	Yes		57.68	71.06
9	No	0.001	24.92	37.90
10	Yes		28.75	41.73

3.2. Recognition Accuracy of Optimal Model

The best recognition result of the model is achieved with an initial learning rate of 0.01 and using the data augmentation method (model 6). Figure 6 shows the change in model recognition accuracy with the increase in training epoch. This shows that both TOP1 and TOP5 accuracies of the model's validation set increase with each increase in training epoch. When the training epoch = 30, the TOP1 and TOP5 accuracies both reach their maxima, 78.32%, and 91.27%, respectively; thereafter, as the training epoch continues to increase, the TOP1 and TOP5 accuracies no longer increase.

**Figure 6.** Change in trend for TOP1/TOP5 accuracy for the grassland-type recognition model.

3.3. Analysis of Factors Influencing Model Performance

3.3.1. Impact of Learning Rate on Model Performance

By observing the results of the ten models (Table 3), it is clear that different learning rates have a large impact on the performance of the models. Taking model 1, model 3, model 5, model 7, and model 9 (without data augmentation) as examples, the trends in model accuracies and losses are shown in Figure 7a. When the initial learning rate was set to 0.1, after 60 epochs of training, the model accuracy was only 16.06%, and the corresponding loss was 0.259, both of which show significant variations. When the initial learning rate was set to 0.05, the model accuracy showed an overall increasing trend but fluctuated significantly. The maximum accuracy was 66.73% within 60 epochs, and the corresponding loss was 0.056. When the initial learning rate was set to 0.005, the model accuracy grew smoothly without too many fluctuations; however, the maximum accuracy

within 60 epochs was only 53.46%, and the corresponding loss was 0.101. When the initial learning rate was set to 0.001, increases in the model's accuracy and decreases in the loss were both slow, with the model accuracy being only 24.92% at the end of the 60th epoch and the corresponding loss being 0.197. When the initial learning rate was set to 0.01, the overall training effect of the model was good, with accuracy growing rapidly and fluctuating less in the first period, the growth rate of accuracy slowed down and remained stable in the second period, and the loss showed an overall change in the opposite direction. Eventually, the accuracy reached a maximum of 73.45% in the 41st epoch, with a corresponding loss of 0.015. Thereafter, as the number of epochs continued to increase, the accuracy and loss no longer changed.

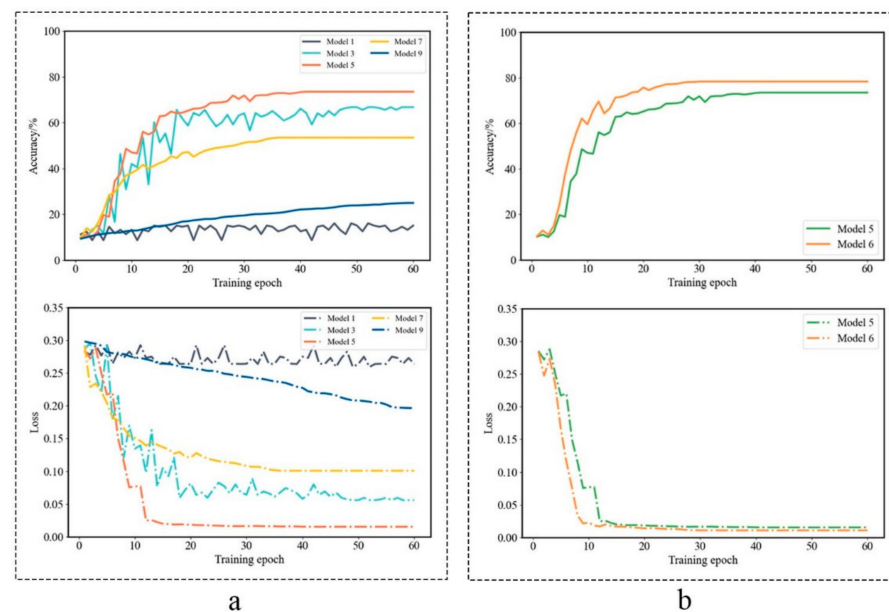


Figure 7. Change in accuracy/loss for the grassland-type recognition model: (a) Under different initial learning rates—the initial learning rates used in model 1, model 3, model 5, model 7, and model 9 are 0.1, 0.05, 0.01, 0.005, and 0.001, respectively. (b) With and without data augmentation—model 5 is without augmented data; model 6 reflects data augmentation.

Since this study adopts the transfer learning method, which makes each layer of the model learn a large amount of knowledge at the early stages of training, the excessive initial learning rate makes it very easy to skip the optimal solution, thus making it difficult to increase the accuracy of the model, which will also be accompanied by periodic variations. However, if the initial learning rate is set too small, the training time will be greatly prolonged and the model will easily fall into the local optimal solution, thus stalling the training. In summary, only by setting a reasonable initial learning rate can it help to shorten the training time of the model to reach the optimal solution quickly, and this can also effectively avoid variation. Using model 2, model 4, model 6, model 8, and model 10 as examples, we reach the same conclusion.

3.3.2. Impact of Data Augmentation on Model Performance

Observing the results of ten models (Table 3), we found that the accuracies of the models with data augmentation increased to differing degrees from those without data augmentation at the same learning rate. Taking model 5 and model 6 as examples, the change trends in model accuracy and loss are shown in Figure 7b. Figure 7b shows that the overall trends of the two are consistent, but the recognition accuracy of model 6 fluctuates less and has higher accuracy and lower loss in the same epoch. For the model without data augmentation, the accuracy was 73.45%, and the corresponding loss was 0.015. For the model with data augmentation, the accuracy was 78.32%, and the corresponding loss was

0.011. The differences between the two were 4.87% and 0.004, respectively. This shows that the use of data augmentation can increase the diversity of data to a certain extent and reduce overfitting in model training.

3.4. Recognition Accuracy of Each Grassland Type

The recognition results of each grassland type and confusion matrix based on the test set are shown in Figure 8, which shows that there are large differences in recognition accuracy and misclassification for different grassland types. The type with the highest recognition accuracy was an alpine meadow, reaching 95.00%, followed by temperate meadow steppe and mountain meadow, both reaching 92.50%, and the type with the lowest recognition accuracy was warm tussock with an accuracy of 70.00%. Misclassifications were more likely to occur among highly similar grassland types, for example, 12.50% of temperate desert images were misclassified as temperate steppe desert, 10.00% of the temperate desert steppe were misclassified as temperate steppe desert, 10.00% of alpine meadow steppe were misclassified as alpine meadow, but the probabilities of misclassification were low or even zero among the grassland types with large differences.

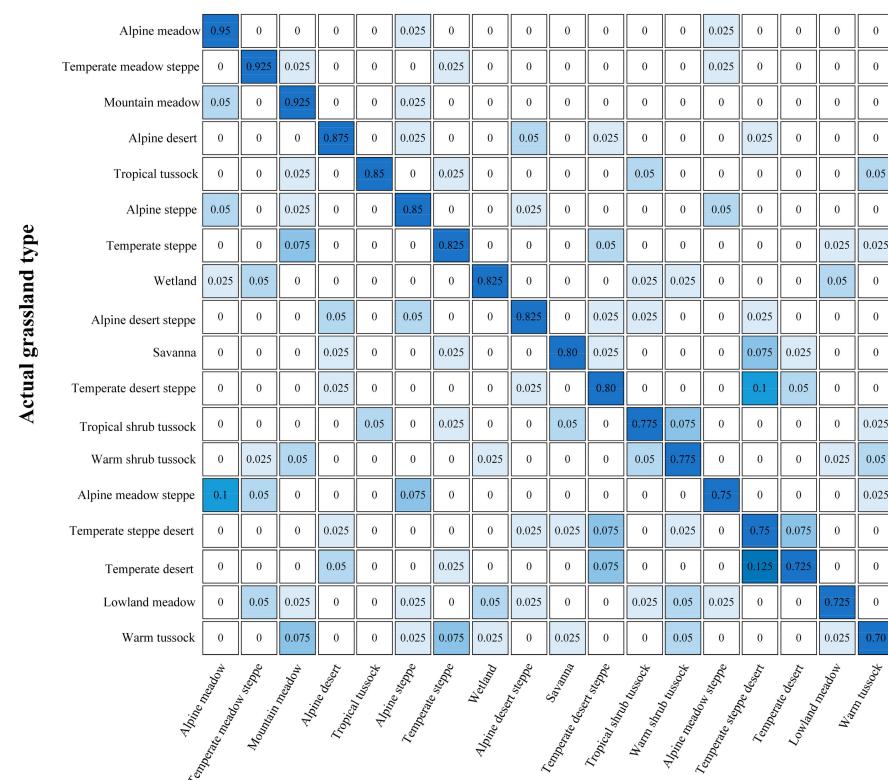


Figure 8. Confusion matrix for the recognition results for each grassland type.

The reasons for the above discrepancies may be due to the data quality of the training set for each grassland type being different, such as image resolution, feature salience, etc. All of these factors will affect the final recognition abilities of the model. In addition, due to the high degree of similarity among some grassland types (such as temperate desert, temperate steppe desert, etc.), it is sometimes difficult to distinguish even for professionals; therefore, the model is more likely to confuse and misclassify during the recognition of these types.

3.5. Implementation of the System

3.5.1. Implementation of System Functions

The system interface is shown in Figure 9. Double-clicking the icon enters the main interface of the system. First, clicking the “Load model” button will load the recognition

model into the system, then the target location and target image can be chosen by clicking the directory tree and file list, and a preview of the image will be displayed in the main interface. Clicking the “Identify” button after selecting the image will use the image recognition model to identify the grassland type the image belongs to and obtain the image attributes (size, time, location, latitude, longitude, and altitude), and then display the results (TOP5) and attributes in the main interface. Clicking the “Introduction” button will open a secondary window containing text and image explanations of the grassland types corresponding to the recognition results. Since the apparent similarity between some grassland types is high, it is difficult to distinguish them only by the recognition model, so users can click the “More info” button to open another secondary window to view the meteorological data and reference grassland types of the corresponding location of the image. This method provides users with other reference results in addition to the model recognition results, which can help in making a comprehensive judgment of the image and thus improve the recognition accuracy. Clicking the “Clear” button will clear the recognition results, image attribute information, and image display in the main interface. In addition, the recognition results, attribute information in the main interface, and the content of two secondary windows can be copied, which is convenient for users to extract information. In addition, in terms of system logic, when the user does not operate the software properly, an error will be raised, the process will be terminated, and a prompt window will be opened to inform the user how to use the software correctly, which will effectively avoid abnormal conditions, such as the program crashing in actual use (Figure 10).

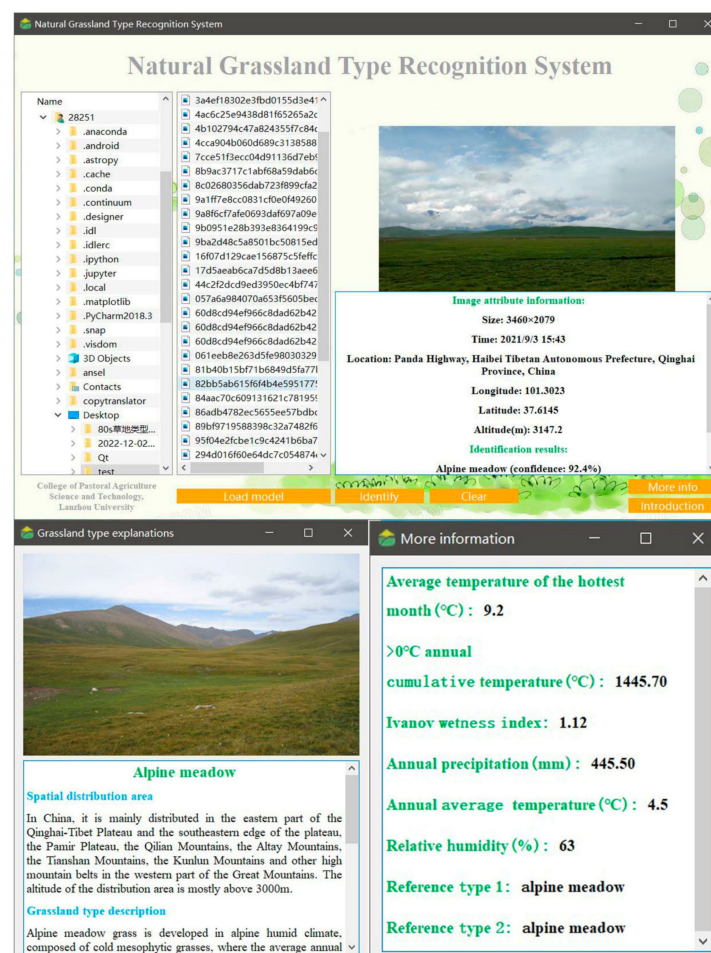


Figure 9. The user interface for the grassland-type recognition system. Reference type 1 is the reference grassland type obtained based on the grassland resource-type map, and reference type 2 is the reference grassland type obtained based on the classification of meteorological data.

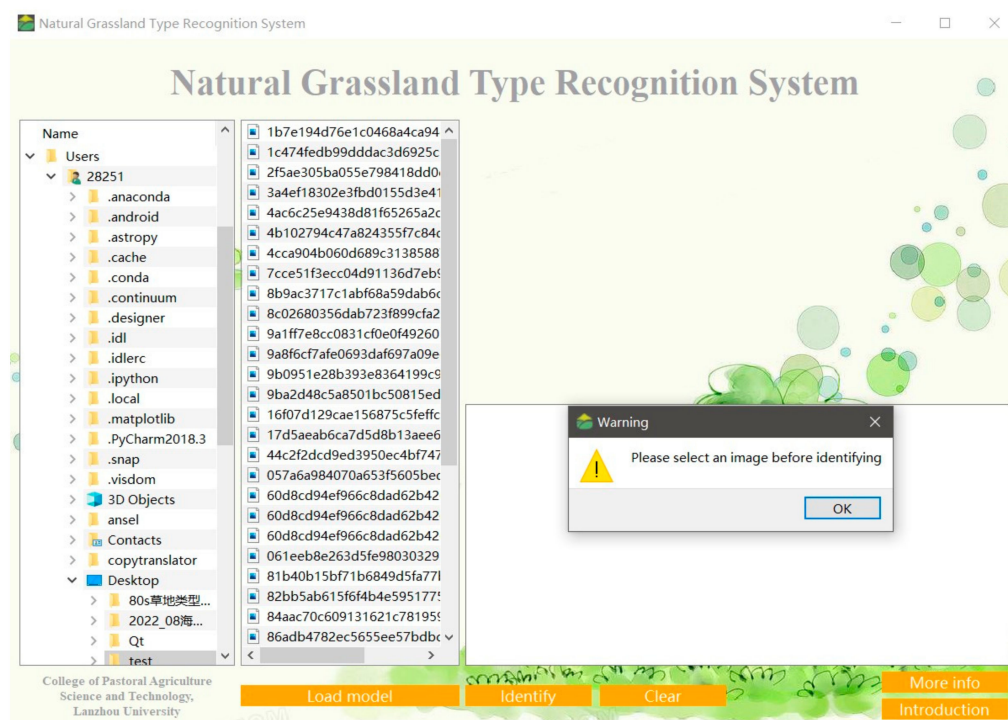


Figure 10. Prompt window example in the grassland-type recognition system.

3.5.2. Comparison of Prediction Results between the Model's TOP1 and Two Reference Grassland Types

We identified the images with location information in the test set (320 images in total) one by one, and the prediction results are shown in Table 4. We found that at least one of the three results was consistent with the true category in 98.44% of the images (No. 1–No. 7). Among them, the situation of all three results was correct: No.1 accounted for 42.81%, the situation of model's TOP1 prediction result, and one of the two reference grassland types were correct; No. 2 and No. 3 accounted for 30.32%, with these two situations accounting for 73.13%, i.e., nearly three-quarters of the total. Therefore, it is recommended to first refer to the result that is consistent with the model's prediction result when the system provides two different results. In addition, the accuracies for the model's TOP1 prediction result, the reference grassland type 1, and the reference grassland type 2 were 78.44%, 72.82%, and 75.01%, respectively. Therefore, when the system provides three different results, it is recommended to refer to the model's prediction result in priority, followed by the reference grassland type 2, and finally the reference grassland type 1. If the user is still unable to make an accurate judgment at that time, please ask an expert for advice.

Table 4. Statistics for the system's prediction results for grassland images.

Number	Whether the Model's TOP1 Prediction Result Is the Same as the True Category	Whether the Reference Grassland Type 1 Is the Same as the True Category	Whether the Reference Grassland Type 2 Is the Same as the True Category	Percentage/%
1	Yes	Yes	Yes	42.81
2	Yes	No	Yes	15.94
3	Yes	Yes	No	14.38
4	Yes	No	No	5.31
5	No	Yes	Yes	11.88
6	No	No	Yes	4.38
7	No	Yes	No	3.75
8	No	No	No	1.56

Reference grassland type 1 is the reference grassland type obtained based on the grassland resource-type map, and reference grassland type 2 is the reference grassland type obtained based on the classification of meteorological data.

3.5.3. Evaluation of System Stability

The operating hardware of the system running the software was an AMD Ryzen R5-4600H running at 3.0 GHz but overclocked to 4.0 GHz, 16 GB RAM, a GTX 1650Ti graphics card, and the Windows 10 operating system. The system's CPU utilization and memory usage for each step are shown in Table 5. The system started with 0% CPU utilization and 0.38% memory usage, and loading the model and selecting an image did not consume additional CPU or memory. The CPU utilization and memory usage during image recognition reached their highest at 8.60% and 14.70%, respectively, but the CPU cache was quickly released after the recognition was completed, and the utilization dropped to 0%, but the memory usage did not change much as it then remained at around 14%. CPU utilization was 1.00% when viewing the grassland-type explanations and 1.50% when viewing more information, both would also release the CPU cache quickly after the information was loaded. During the entire use process, the response time of the system was fast, with no obvious lag. The above situation shows that the system is relatively stable and can meet most of the usage requirements.

Table 5. CPU and memory utilization at each stage of system usage.

Operation Step	CPU Utilization/%	Memory Usage/%
Start	0	0.38
Load the model	0	0.38
Select an image	0	0.38
Recognize	8.60	14.70
View explanations	1.00	14.00
View more information	1.50	14.00

4. Discussion

4.1. Shortcomings and Prospects of Grassland-Type Recognition Model

Since deep learning relies heavily on a large amount of labeled data, this study supplements image data by web crawling and data augmentation based on grassland image data taken in the field, since this can avoid model overfitting and improve model recognition performance to a certain extent [41,42]. After parameter adjustment and comparison analysis of the models, the TOP1 accuracy of the constructed optimal grassland-type recognition model was 78.32%, and the TOP5 accuracy was 91.27%. The recognition accuracy for each of the 18 grassland types reached over 70.00%, and the overall recognition effect of the model was good, which can meet the recognition requirements in most cases; however, the model still has some shortcomings. Firstly, although this study adopted a transfer learning approach and supplemented the image data with web crawling and data augmentation, which do improve the recognition effect of the model, for deep learning algorithms, the amount of current training samples is still not large enough, which limits the recognition performance of the model to some extent. Secondly, the high similarity among some of the grassland types makes the model misclassify easily when recognizing these grassland types. Thirdly, the field photography data were collected from July to September, but the grassland phenotype may change with the temporal condition, and the model cannot guarantee the recognition accuracy of grassland images collected in other seasons. Therefore, in future research, we need to continue to obtain more grassland image data by field photography and consider combining generative adversarial networks and other methods to supplement and expand the samples to further improve the recognition accuracy of the existing model. The existing model needs to be optimized and improved to reduce the misclassification of similar grassland types by setting weight decay for some categories, etc. In addition, we must collect images of the 18 grassland types in various seasons to enable the use of the recognition model in different seasons, which will pose new challenges for the grassland-type recognition task in the future.

4.2. Advantages and Disadvantages of the Grassland-Type Recognition System

To facilitate the use of a grassland-type recognition model for users, this study also designs the grassland-type recognition system based on PyQt5. The system simplifies model calls and standardizes the process of grassland-type recognition, which can effectively avoid program exceptions that occur during use. At the same time, the system can analyze the relevant parameters returned by the model, which helps to show the recognition results to users more clearly and easily [43]. In addition, the system also integrates a module to assist in judging the grassland types to which the images belong based on the grassland resource-type map and meteorological data, which will provide users with references in addition to the model recognition results and will help improve the accuracy of grassland-type recognition. However, the grassland-type recognition system does not support use over the internet or on cell phones yet, since the software currently has a limited usage scope. Therefore, in the future, it will be necessary to publish the recognition system on the web with accessibility for cell phones to expand the user population and make grassland-type recognition more convenient and make it possible to identify grassland types in field surveys simply by taking images with a digital camera or cell phone.

5. Conclusions

The continuous development of deep learning techniques provides new approaches for image recognition in many fields; this study is a practical attempt to recognize grassland types in this context. Compared with traditional manual recognition and remote sensing recognition, the deep learning-based grassland image recognition method can not only achieve real-time and fast recognition but also significantly reduce the difficulty and complexity of grassland recognition, which simplifies the recognition process and improves efficiency. Using the PyTorch deep learning framework, this study constructed the natural grassland-type recognition models based on the transfer learning method and the VGG-19 model as well as comparing the effects of different initial learning rates and the use of data augmentation on model recognition performance. The main results were: (1) Different initial learning rates have a large impact on model performance. When the impact of data augmentation was not considered, the model accuracies corresponding to initial learning rates of 0.1, 0.05, 0.01, 0.005, and 0.001 were 16.06%, 66.73%, 73.45%, 53.46%, and 24.92%, respectively. (2) Data augmentation has a positive impact on model performance. The model accuracies with data augmentation increased by 3.07%, 4.88%, 4.87%, 4.22%, and 3.83% over the model accuracies without data augmentation at initial learning rates of 0.1, 0.05, 0.01, 0.005, and 0.001, respectively. (3) The accuracy of the grassland-type recognition model increased with the increase in training epoch, and the model performance reached its optimum with training epoch = 30, after which it remained stable and unchanged. At this time, the TOP1 accuracy of the optimal model was 78.32%, and the TOP5 accuracy was 91.27%. (4) The recognition accuracy of each grassland type was above 70.00%, misclassifications mainly occurred among the grassland types with high similarity, and the probabilities of misclassification among most of the grassland types were less than 5.00%. (5) Using the optimal grassland-type recognition model, this study designed a grassland-type recognition system through PyQt5. Two reference grassland types based on the Chinese grassland resource-type map and meteorological data are integrated into the system for auxiliary judgments of model recognition results, and the prediction accuracy of reference grassland type 2 was 2.19% higher than that of reference grassland type 1. This helps to improve the accuracy of grassland-type recognition.

We will continue to optimize and improve the existing model and system for grassland-type recognition to provide a new approach and technical support to carry out grassland field surveys easily and reliably.

Author Contributions: Y.X. and J.G.: conceptualization, data curation, formal analysis, software, methodology, visualization, writing, and investigation; M.H.: visualization, polishing, and investigation; Q.F. and T.L.: conceptualization, polishing, data resources; R.G.: data collation, investigation; J.C. and Q.W.: investigation. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the China Postdoctoral Science Foundation (2022M721441), the Major Consulting Research Project of the Chinese Academy of Engineering (2022-HZ-9, 2022-XY-139), the Fundamental Research Funds for the Central Universities, Lanzhou University (lzujbky-2022-sp13), and the China Agriculture Research System of MOF (Ministry of Finance), MARA (Ministry of Agriculture and Rural Affairs), and the 111 Project (B12002).

Data Availability Statement: Not applicable.

Acknowledgments: We acknowledge the reviewers and academic editors for their positive and constructive comments and suggestions.

Conflicts of Interest: All co-authors declare that there are no conflicts of interest.

References

1. Piao, S.; Fang, J.; Ciais, P.; Peylin, P.; Huang, Y.; Sitch, S.; Wang, T. The carbon balance of terrestrial ecosystems in China. *Nature* **2009**, *458*, 1009–1013. [\[CrossRef\]](#) [\[PubMed\]](#)
2. Hou, M.; Ge, J.; Xiu, Y.; Meng, B.; Liu, J.; Feng, Q.; Liang, T. The urgent need to develop a new grassland map in China: Based on the consistency and accuracy of ten land cover products. *Sci. China Life Sci.* **2022**, *66*, 385–405. [\[CrossRef\]](#) [\[PubMed\]](#)
3. O'Mara, F.P. The role of grasslands in food security and climate change. *Ann. Bot.* **2012**, *110*, 1263–1270. [\[CrossRef\]](#)
4. Zheng, X.; Zhang, J.; Cao, S. Net value of grassland ecosystem services in mainland China. *Land Use Policy* **2018**, *79*, 94–101. [\[CrossRef\]](#)
5. Akiyama, T.; Kawamura, K. Grassland degradation in China: Methods of monitoring, management and restoration. *Grassl. Sci.* **2007**, *53*, 1–17. [\[CrossRef\]](#)
6. Zhou, W.; Yang, H.; Huang, L.; Chen, C.; Lin, X.; Hu, Z.; Li, J. Grassland degradation remote sensing monitoring and driving factors quantitative assessment in China from 1982 to 2010. *Ecol. Indic.* **2017**, *83*, 303–313. [\[CrossRef\]](#)
7. Kang, L.; Han, X.; Zhang, Z.; Sun, O. Grassland ecosystems in China: Review of current knowledge and research advancement. *Philos. Trans. R. Soc. B Biol. Sci.* **2007**, *362*, 997–1008. [\[CrossRef\]](#)
8. Yan, L.; Zhou, G.; Zhang, F. Effects of different grazing intensities on grassland production in China: A meta-analysis. *PLoS ONE* **2013**, *8*, e81466. [\[CrossRef\]](#)
9. Cai, H.; Yang, X.; Xu, X. Human-induced grassland degradation/restoration in the central Tibetan Plateau: The effects of ecological protection and restoration projects. *Ecol. Eng.* **2015**, *83*, 112–119. [\[CrossRef\]](#)
10. Zhang, Q.; Buyantuev, A.; Fang, X.; Han, P.; Li, A.; Li, F.Y.; Liang, C.; Liu, Q.; Ma, Q.; Niu, J.; et al. Ecology and sustainability of the Inner Mongolian Grassland: Looking back and moving forward. *Landsc. Ecol.* **2020**, *35*, 2413–2432. [\[CrossRef\]](#)
11. Xu, D.; Chen, B.; Shen, B.; Wang, X.; Yan, Y.; Xu, L.; Xin, X. The classification of grassland types based on object-based image analysis with multisource data. *Rangel. Ecol. Manag.* **2019**, *72*, 318–326. [\[CrossRef\]](#)
12. Guo, F.; Fan, J.; Bian, J.; Liu, F.; Zhang, H. Grassland types identification based on time-series MODIS NDVI data in northern Tibet. *Remote Sens. Technol. Appl.* **2011**, *26*, 821–826. [\[CrossRef\]](#)
13. Sun, M.; Shen, W.; Xie, M.; Li, H.; Gao, F. The identification of grassland types in the source region of the Yarlung Zangbo River based on spectral features. *Remote Sens. Nat. Resour.* **2012**, *24*, 83–89. [\[CrossRef\]](#)
14. Alem, A.; Kumar, S. Deep Learning Models Performance Evaluations for Remote Sensed Image Classification. *IEEE Access* **2022**, *10*, 111784–111793. [\[CrossRef\]](#)
15. Cheng, G.; Xie, X.; Han, J.; Guo, L.; Xia, G.-S. Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 3735–3756. [\[CrossRef\]](#)
16. Sheehan, S.; Song, Y.S. Deep learning for population genetic inference. *PLoS Comput. Biol.* **2016**, *12*, e1004845. [\[CrossRef\]](#)
17. Sun, Y.; Liu, Y.; Wang, G.; Zhang, H. Deep learning for plant identification in natural environment. *Comput. Intell. Neurosci.* **2017**, *2017*, 7361042. [\[CrossRef\]](#) [\[PubMed\]](#)
18. Liu, Z.; Wang, J.; Tian, Y.; Dai, S. Deep learning for image-based large-flowered chrysanthemum cultivar recognition. *Plant Methods* **2019**, *15*, 146. [\[CrossRef\]](#)
19. Li, J.; Sun, S.; Jiang, H.; Tian, Y.; Xu, X. Image recognition and empirical application of desert plant species based on convolutional neural network. *J. Arid. Land* **2022**, *14*, 1440–1455. [\[CrossRef\]](#)
20. Mu, Y.; Feng, R.; Ni, R.; Li, J.; Luo, T.; Liu, T.; Li, X.; Gong, H.; Guo, Y.; Sun, Y.; et al. A Faster R-CNN-Based Model for the Identification of Weed Seedling. *Agronomy* **2022**, *12*, 2867. [\[CrossRef\]](#)
21. Lin, H.L.; Feng, Q.S.; Liang, T.G.; Ren, J.Z. Modelling global-scale potential grassland changes in spatio-temporal patterns to global climate change. *Int. J. Sustain. Dev. World Ecol.* **2013**, *20*, 83–96. [\[CrossRef\]](#)
22. Lin, H.; Zhang, Y. Evaluation of six methods to predict grassland net primary productivity along an altitudinal gradient in the Alxa Rangeland, Western Inner Mongolia, China. *Grassl. Sci.* **2013**, *59*, 100–110. [\[CrossRef\]](#)

23. Jin, Y.; Yang, X.; Qiu, J.; Li, J.; Gao, T.; Wu, Q.; Zhao, F.; Ma, H.; Yu, H.; Xu, B. Remote sensing-based biomass estimation and its spatio-temporal variations in temperate grassland, Northern China. *Remote Sens.* **2014**, *6*, 1496–1513. [\[CrossRef\]](#)
24. DAHV (Department of Animal Husbandry and Veterinary, the Ministry of Agriculture of the People's Republic of China); NAHVS (National Animal Husbandry and Veterinary Service, the Ministry of Agriculture of the People's Republic of China). *Rangeland Resources of China*; China Science and Technology Press: Beijing, China, 1996; pp. 147–339.
25. Shen, J.-J.; Chang, C.-C.; Li, Y.-C. Combined association rules for dealing with missing values. *J. Inf. Sci.* **2007**, *33*, 468–480. [\[CrossRef\]](#)
26. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 60. [\[CrossRef\]](#)
27. Su, D. The compilation and study of the grassland resource map of China on the scale of 1:1,000,000. *J. Nat. Resour.* **1996**, *11*, 75–83. [\[CrossRef\]](#)
28. Ren, J.Z.; Hu, Z.Z.; Zhao, J.; Zhang, D.G.; Hou, F.J.; Lin, H.L.; Mu, X.D. A grassland classification system and its application in China. *Rangel. J.* **2008**, *30*, 199–209. [\[CrossRef\]](#)
29. Laporte, F.; Dambre, J.; Bienstman, P. Highly parallel simulation and optimization of photonic circuits in time and frequency domain based on the deep-learning framework PyTorch. *Sci. Rep.* **2019**, *9*, 5918. [\[CrossRef\]](#)
30. Ketkar, N.; Moolayil, J. Introduction to pytorch. In *Deep Learning with Python*; Apress: Berkeley, CA, USA, 2021; pp. 27–91.
31. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 8026–8037.
32. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556. [\[CrossRef\]](#)
33. Lee, H.; Grosse, R.; Ranganath, R.; Ng, A.Y. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, QC, Canada, 14–18 June 2009; pp. 609–616. [\[CrossRef\]](#)
34. Han, B.; Du, J.; Jia, Y.; Zhu, H. Zero-Watermarking Algorithm for Medical Image Based on VGG19 Deep Convolution Neural Network. *J. Health Eng.* **2021**, *2021*, 5551520. [\[CrossRef\]](#) [\[PubMed\]](#)
35. Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering. IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [\[CrossRef\]](#)
36. Jiang, W.; Peng, J.; Geyan, Y. Research on adaptive learning rate algorithm in deep learning. *J. Huazhong Univ. Sci. Technol. (Nat. Sci. Ed.)* **2019**, *47*, 79–83. [\[CrossRef\]](#)
37. Petrovska, B.; Atanasova-Pacemska, T.; Corizzo, R.; Mignone, P.; Lameski, P.; Zdravevski, E. Aerial scene classification through fine-tuning with adaptive learning rates and label smoothing. *Appl. Sci.* **2020**, *10*, 5792. [\[CrossRef\]](#)
38. Khan, A.; Sohail, A.; Zahoora, U.; Qureshi, A.S. A survey of the recent architectures of deep convolutional neural networks. *Artif. Intell. Rev.* **2020**, *53*, 5455–5516. [\[CrossRef\]](#)
39. Li, M.; Zhou, Z.-H. Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples. *IEEE Trans. Syst. Man, Cybern.-Part A Syst. Humans* **2007**, *37*, 1088–1098. [\[CrossRef\]](#)
40. Foster, E.C.; Godbole, S. Overview of MySQL. In *Database Systems*; Apress: Berkeley, CA, USA, 2016; pp. 451–460.
41. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [\[CrossRef\]](#)
42. Tian, L.; Fan, C.; Ming, Y.; Jin, Y. Stacked PCA network (SPCANet): An effective deep learning for face recognition. In Proceedings of the 2015 IEEE International Conference on Digital Signal Processing (DSP), Singapore, 21–24 July 2015; pp. 1039–1043. [\[CrossRef\]](#)
43. Omer, R.; Fu, L. An automatic image recognition system for winter road surface condition classification. In Proceedings of the 13th International IEEE Conference on Intelligent Transportation Systems, Funchal, Portugal, 19–22 September 2010; pp. 1375–1379. [\[CrossRef\]](#)

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.