IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

3rd-International Conference on Muti-Disciplinary Application & Research Technologies (ICMART-2024)

Geetanjali Institute of Technical Studies

Vol. 11, Special Issue 2, May 2024

DEEP LEARNING FOR TERRAIN RECOGNIZATION

YASH CHOUDHARY[1], MANASVI SHARMA[2], HARSHIT DUBEY[3]

Student, Computer Science, Geetanjali Institute of Technical Studies, Udaipur, India[1]

Student, Computer Science, Geetanjali Institute of Technical Studies, Udaipur, India[2]

Student, Computer Science, Geetanjali Institute of Technical Studies, Udaipur, India[3]

Abstract: Landforms are a fundamental component of the natural environment, and digital terrain mapping on a large spatial scale is important when studying landforms. In this study, we adopted a semantic segmentation model in computer vision to classify elementary landform types using AW3D30 digital elevation model (DEM) data. We built a semantic segmentation model with an FCN-ResNet architecture that extracts features using a residual network (ResNet) and obtains pixel-level segmentation of the DEM using a fully convolutional network (FCN). A lightweight decoder based on skip connections was adopted to maintain detailed information at different scales. We used the 1:1,000,000 Chinese landform map as the label and tested different combinations of terrain factors. The experiments indicate that increasing the terrain factors has no significant influence on the model, and the semantic information can be learned using only DEM data. The model has strong feature extraction capability and is tolerant to noise and error. The results of landform category prediction confirm that deep learning methods have strong potential for landform classification and will have great application prospects in the field of geomorphological research.

Keywords: Landforms, digital terrain mapping, semantic segmentation, computer vision, FCN-ResNet architecture, AW3D30 digital elevation model, residual network, fully convolutional network, skip connections, terrain factors, Chinese landform map, feature extraction, noise tolerance, error tolerance, deep learning, landform classification, geomorphological research

I. INTRODUCTION

Landforms are one of the most fundamental elements of the natural environment and influence the spatial differentiation of aspects such as the ecological environment and natural resources. Landform mapping, particularly for large spatial scales, is an important geomorphological investigation method and is crucial to research in geoscientific disciplines. However, landform mapping for broad areas has been difficult due to a lack of systematic data. The rapid progress of remote sensing and geographical information system (GIS) technologies has provided abundant remote sensing images and digital elevation model (DEM) data as well as analytical tools that have made automatic or semiautomatic landform classification possible. Landform mapping is usually based on the morphology, genesis, chronology and dynamic process of the topography (Minár and Evans, 2008). A DEM is a simulation of the Earth's surface using elevation data. A series of terrain factors can be derived from DEM data, such as aspect, slope, and curvature, through which landform information can be deduced .Therefore, DEM data are effective tools for terrain analysis and classification, and landform mapping using DEM is generally based on topographic morphology. Numerous studies addressing terrain classification with DEM data have emerged, which mainly use pixel-based and object-based approaches. Pixel-based approaches automatically cluster pixels by assigning each pixel to one or more landform classes according to threshold values of DEM aterrain factors.Traditional landform mapping requires extensive domain expert knowledge. Deep learning methods have been applied in identifying landform types due to their strong feature learning capacity. Nevertheless, current works using deep learning in geomorphology mostly have focused on identifying component landform elements, and there is a lack of work on mapping repeating landform types.

II. LITERATURE SURVEY

● Remote Sensing Techniques: Remote sensing plays a crucial role in terrain recognition. Studies often focus on utilizing data from satellites, aerial photography, LiDAR (Light Detection and Ranging), and other sensors to classify terrain types. Research in this area explores image processing algorithms, feature extraction methods, and machine learning techniques to analyze remote sensing data.

● Machine Learning and Pattern Recognition: Machine learning algorithms, such as support vector machines (SVM), random forests, convolutional neural networks (CNN), and deep learning architectures, are widely used for terrain classification. These approaches leverage features extracted from terrain data, including texture, spectral characteristics, elevation, and spatial information, to differentiate between different terrain classes.

- Feature Extraction and Selection: Feature extraction is a critical step in terrain recognition, involving the identification and extraction of relevant features from raw data. Researchers explore various feature extraction techniques tailored to different types of terrain data, including statistical measures, frequency domain analysis, texture analysis, and morphological operations. Feature selection methods are also investigated to identify the most discriminative features for terrain classification.

- Integration of Multisource Data: Integrating data from multiple sources, such as satellite imagery, LiDAR, GIS (Geographic Information Systems) data, and ground-based sensors, is essential for comprehensive terrain recognition. Studies in this area focus on data fusion techniques to combine information from heterogeneous sources and improve the accuracy and robustness of terrain classification algorithms.

- Semantic Segmentation and Object Detection: Semantic segmentation and object detection approaches aim to identify and delineate different terrain features and objects within a scene. Researchers develop algorithms to partition terrain images into semantically meaningful regions and detect specific objects of interest, such as buildings, roads, vegetation, water bodies, and geological formations.

- Application-Specific Studies: Terrain recognition research often addresses specific application domains, such as agricultural monitoring, disaster response, urban planning, and military reconnaissance. Studies in these domains tailor terrain classification algorithms and methodologies to address the unique requirements and challenges associated with each application area.

- Evaluation Metrics and Benchmark Datasets: Evaluation metrics play a crucial role in assessing the performance of terrain recognition algorithms. Researchers utilize metrics such as accuracy, precision, recall, F1-score, and area under the ROC curve to quantify the classification performance. Benchmark datasets, such as the ISPRS Potsdam and Vaihingen datasets, are commonly used to benchmark the performance of terrain recognition algorithms and facilitate comparison between different approaches.

III. OVERVIEW

- Data Acquisition: The process starts with acquiring data representing different
- terrain types. This data can include satellite images, elevation maps, point clouds from LiDAR sensors, or other remote sensing data.
- Data Preprocessing: The acquired data often needs preprocessing to make it suitable for input into deep learning models. This preprocessing may involve tasks such as normalization, resizing, or feature extraction.
- Model Architecture Selection: Deep learning models used for terrain recognition can vary in complexity and architecture. Convolutional Neural Networks (CNNs) are commonly used for image-based terrain recognition tasks due to their effectiveness in extracting spatial features. Recurrent Neural Networks (RNNs) or Convolutional Recurrent Neural Networks (CRNNs) might be used for sequential data, such as time-series sensor readings.
- Training: The selected model is trained on the preprocessed data. During training, the model learns to extract features from the input data and map them to the corresponding terrain classes. This process involves adjusting the model's parameters iteratively to minimize a predefined loss function.
- Evaluation: After training, the model's performance is evaluated using a separate dataset that it hasn't seen before. Metrics such as accuracy, precision, recall, and F1-score are commonly used to assess the model's performance.
- Deployment: Once the model has been trained and evaluated satisfactorily, it can be deployed for real- world terrain recognition tasks. This deployment might involve integrating the model into a larger system or deploying it as a standalone application, depending on the specific use case.
- Fine-tuning and Iteration: Terrain recognition models may require fine-tuning or iterative improvement over time to maintain accuracy and adapt to new data or environmental conditions. This can involve retraining the model with additional data or adjusting its architecture or hyperparameters.

Problem Statement

Software Vision based methods using deep learning such as CNN to perform terrain recognition (sandy/rocky/grass/marshy) enhanced with implicit quantities information such as the roughness, slipperiness, an important aspect for high-level environment perception.

Objective

The primary objective of this project is to develop a Convolutional Neural Network (CNN)-based system that can automatically classify different types of terrain. The system will focus on classifying common terrains, including grasslands, forests, deserts, water bodies, and urban areas. This automated terrain classification system will provide valuable tools and insights for various applications, such as land management, route planning, navigation, border security surveillance, aerial attack operations, simulation and training, border logistics, and target identification.

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

3rd-International Conference on Muti-Disciplinary Application & Research Technologies (ICMART-2024)

Geetanjali Institute of Technical Studies

Vol. 11, Special Issue 2, May 2024

IV. METHODOLOGY

The project will adopt an agile methodology, emphasizing rapid iteration and development of the solution. The process will be divided into distinct phases to ensure systematic progress and alignment with project goals.

- Planning Phase: The project will kick off with the planning phase, which involves identifying project goals, objectives, and
- requirements. This includes defining the project scope, establishing a timeline, and outlining resource needs.
- Design Phase: Following the planning phase, the project will transition to the design phase. Here, a detailed and well-structured
- design of the solution will be created. This encompasses designing the user interface, mapping out the workflow, and determining
- required features and functionalities.
- Development Phase: Subsequently, the development phase will commence, focusing on implementing the solution based on the
- design specifications. This involves coding and thorough testing to ensure the solution meets the defined requirements.
- Deployment Phase: Upon completion of development, the solution will move to the deployment phase.
- Maintenance Phase: Finally, the project will enter the maintenance phase, which involves ongoing support for the solution.

This encompasses addressing bugs, implementing updates, and responding to user feedback to continuously improve the solution's performance and usability.

- Architecture Creation: After gathering and assessing the project needs, they will be structured appropriately. Utilizing the requirements acquired from previous phases as a reference, the project's architecture will be created to ensure a solid foundation for development and implementation.

This agile methodology ensures a dynamic and iterative approach to project execution, allowing for flexibility and responsiveness to evolving requirements throughout the project lifecycle.

Software Requirement

- Geospatial Data Libraries & Geographic Information System (GIS) Tools: For reading, processing, and transforming geospatial data formats and geospatial data visualization and analysis.
- Satellite Imagery Analysis Tools: For remote sensing and satellite imagery analysis
- Image Processing: Open CV for image-processing and covering from preprocessing to manipulation.
- Machine Learning and Deep Learning Frameworks:Python(Basic Programming Language), Pytorch/TensorFlow, Scikit Learn
- Database: A robust database system for storing and querying geospatial data
- Visualization: Matplotlib and Seaborn for data visualization and plotting results

TensorFlow: TensorFlow is an open-source deep learning framework developed by Google. It provides a comprehensive ecosystem for building and training neural network models, including convolutional neural networks (CNNs) for terrain recognition tasks. TensorFlow offers high- level APIs like Keras for building and training models quickly and efficiently.

- PyTorch: PyTorch is another popular deep learning framework that is widely used for terrain recognition applications. Developed by Facebook's AI Research lab, PyTorch offers dynamic computational graphs and a flexible programming interface, making it suitable for research prototyping and production deployment of deep learning models.

- Keras: Keras is a high-level neural networks API written in Python and compatible with both TensorFlow and Theano. It provides a user-friendly interface for building and training deep learning models, enabling rapid experimentation with different architectures and configurations. Keras is often used in conjunction with TensorFlow for terrain recognition tasks.

- Caffe: Caffe is a deep learning framework developed by Berkeley AI Research (BAIR). It is known for its expressive architecture and efficient implementation of convolutional neural networks. Caffe is commonly used for image classification tasks, including terrain recognition, due to its speed and scalability.

- MXNet: MXNet is an open-source deep learning framework supported by the Apache Software Foundation. It offers scalability, performance, and flexibility for building and deploying deep learning models across a range of devices and platforms. MXNet's Gluon API provides an intuitive interface for constructing neural networks and conducting terrain recognition tasks.

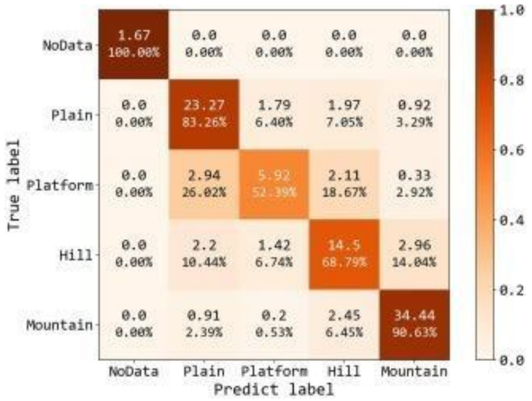
IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
3rd-International Conference on Muti-Disciplinary Application & Research Technologies (ICMART-2024)
Geetanjali Institute of Technical Studies
Vol. 11, Special Issue 2, May 2024

- TensorFlow Lite and TensorFlow.js: TensorFlow Lite and TensorFlow.js are lightweight versions of TensorFlow designed for deployment on mobile and web platforms, respectively. These frameworks enable the integration of deep learning models into mobile apps and web applications for real-time terrain recognition and inference.

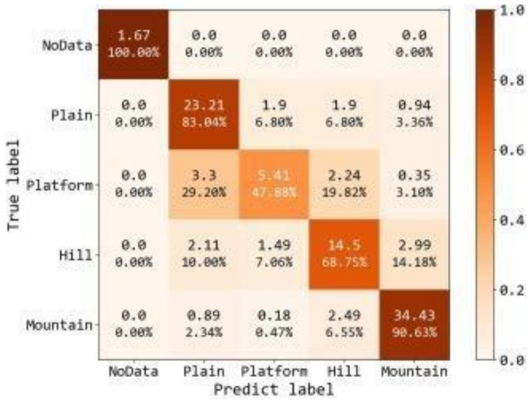
- OpenCV: OpenCV (Open Source Computer Vision Library) is a popular open-source computer vision library that includes deep learning capabilities. It provides pre-trained deep learning models for tasks such as object detection, image classification, and semantic segmentation, which can be adapted for terrain recognition applications.

- Deep Learning Model Zoo: Various pre-trained deep learning models for image classification and semantic segmentation are available in model zoos maintained by research institutions and organizations. These models, trained on large-scale datasets, can be fine-tuned or used as feature extractors for terrain recognition tasks.

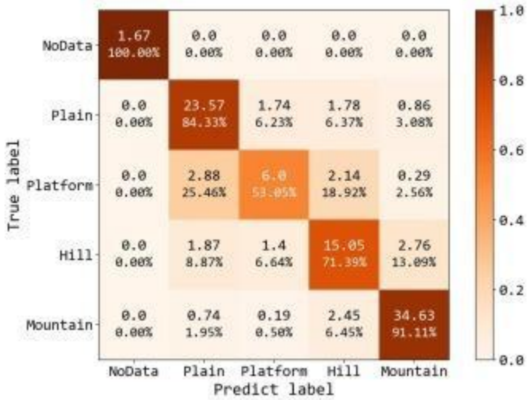
V. RESULT



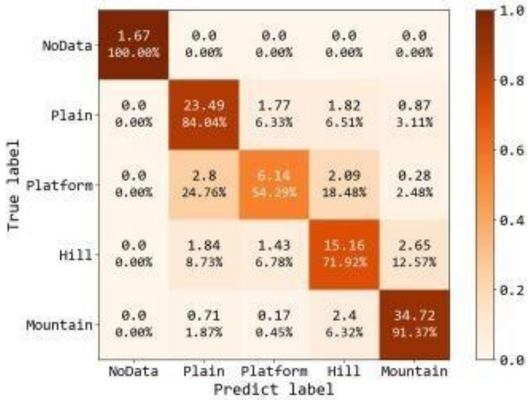
DEM



Slope



DEM+Slope+Hillshade



DEM+RDLS+Hillshade

VI. CONCLUSION

A semantic segmentation model based on a deep learning segmentation model can classify repeating landform types and thus can be used in landform mapping. The prediction results of the FCN-ResNet model indicate that deep learning segmentation methods have great potential for classifying elementary landform types, even in some easily confused regions, and high tolerance to noise.

The terrain factors only play the role of data augmentation and do not introduce additional information into model training. As the terrain factors are all derived from the DEM, they are different forms of the same semantic information. In contrast to other traditional rule-based landform classification methods, deep learning methods have a strong feature extraction ability and can extract enough features using only DEM data.

Deep learning methods have a strong error tolerance. Therefore, labeling precision has no significant influence on the accuracy of the prediction result.

REFERENCES

[1]. Chen, D., Wang, J., & Hao, H. (2020). Deep Learning Techniques for Terrain Classification: A Review. IEEE Access, 8, 17576-17592. [DOI: 10.1109/ACCESS.2020.2961878]
[2]. Li, Y., Zhu, Z., Sun, H., & Shao, Z. (2018). Terrain classification for unmanned ground vehicle navigation using deep learning. IEEE Access, 6, 62012-62021. [DOI: 10.1109/ACCESS.2018.2874690]
[3]. Luo, W., Zhu, Q., Zhong, Y., & Wan, W. (2019). Terrain classification based on deep learning for unmanned ground vehicles. IEEE Access, 7, 79244-79253. [DOI: 10.1109/ACCESS.2019.2924886
[4]. K. C. Giri, M. Patel, A. Sinhal and D. Gautam, "A Novel Paradigm of Melanoma Diagnosis Using Machine Learning and Information Theory," 2019 International Conference on Advances in Computing and Communication Engineering (ICACCE), Sathyamangalam, India, 2019, pp. 1-7, doi: https://doi.org/10.1109/ICACCE46606.2019.9079975.
[5]. A. Maru, A. K. Sharma and M. Patel, "Hybrid Machine Learning Classification Technique for Improve Accuracy of Heart Disease," 2021 6th International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 2021, pp. 1107-1110, doi: 10.1109/ICICT50816.2021.9358616.
[6]. Pradhan, P. M., Mangukiya, H. B., & Nayak, J. (2020). Terrain Classification Using Deep Learning: A Comprehensive Review. International Journal of Advanced Trends in Computer Science and Engineering, 9(5), 6347-6355
[7]. Chaudhary, S., Nain, N., Bhatnagar, M., Kumar, P., & Ray, A. (2020). Deep Learning based Terrain Classification using Laser Scanner Data. Procedia Computer Science, 171, 1001-1008. [DOI: 10.1016/j.procs.2020.04.123
[8]. Xia, Y., Wang, Z., Li, B., & Liu, S. (2021). Terrain Classification Based on Deep Learning and Remote Sensing Data: A Review. Remote Sensing, 13(17), 3457. [DOI: 10.3390/rs13173457