International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Applications of Hyperspectral Imagery for Identifying the Change Detection Parameter

Prof. Datta Choudhari [a], Adarsh kurundwade [b], Omkar Hiremath [c], Lakshita[d], Nidhi Oulkar [e]

[a,b,c,d,e] *Department of Artificial Intelligence and Data Science, Angadi Institute of Technology and Management, Belagavi-590009, India*

A B S T R A C T

This project explores hyperspectral imagery for change detection using semantic segmentation, leveraging advanced neural network architectures including U-Net and CNN. The U-Net excels in spatial information capture while enhancing feature extraction and model depth. Our methodology involves preprocessing, training on labelled datasets, and fine-tuning parameters. Semantic segmentation enables precise identification of changed regions. Evaluation of diverse datasets demonstrates superior performance compared to traditional methods, emphasizing potential applications in environmental monitoring, urban planning, and disaster response. This project contributes to advancing change detection in hyperspectral imagery, showcasing the efficacy of U-Net and CNN architectures for accurate semantic segmentation.

Keywords: U-NET, CNN, Semantic segmentation, Change detection.

Introduction

Hyperspectral photography, with its ability to record detailed spectral information across a wide range of wavelengths, has become an essential tool in a variety of remote sensing applications. Change detection is an important activity in this sector since it allows us to monitor environmental changes, land cover variations, and urban growth. Traditional approaches for identifying changes in hyperspectral data frequently struggle to efficiently handle complex spectral information while capturing small spatial variations. This study solves these difficulties by presenting an advanced architecture that utilizes state-of-the-art neural networks.

The U-Net model is known for its ability to capture complex spatial features, making it ideal for evaluating hyperspectral pictures. In contrast, the architecture excels at improving feature extraction and model depth, resulting in a more complete comprehension of hyperspectral data. Additionally, the incorporation of CNNs improves the framework's capabilities by allowing for quick processing and analysis of complicated hyperspectral images.

The approach consists of numerous processes, such as preprocessing hyperspectral pictures, training CNN and U-Net models on labelled datasets, and adjusting network parameters for maximum performance. The use of semantic segmentation techniques aids in the exact detection of changed areas within hyperspectral images.

Literature survey

[1] Remote Sensing Dataset: This refers to the external source that provides data to the system. It could be a data repository, for instance. Data Preprocessing: This step takes the raw data from the remote sensing dataset and prepares it for further analysis. This may involve cleaning the data, formatting it, or transforming it into a usable form. Train Test Split: This step divides the preprocessed data into two sets: a training set and a testing set. The training set is used to train a model, while the testing set is used to evaluate the model's performance.

[2] This paper showcases CNNs' superiority in land cover mapping with HSI, offering high accuracy and efficiency compared to traditional methods. CNNs extract spectral and spatial features for precise classification of land cover types, making them ideal for large-scale applications. Techniques like transfer learning enhance their performance, while the proposed deep Siamese CNN architecture streamlines feature extraction for automated classification. Overall, CNNs present a potent solution for HSI-based land cover mapping, revolutionizing remote sensing and geospatial sciences.

[3] This paper showcases CNNs' superiority in land cover mapping with HSI, offering high accuracy and efficiency compared to traditional methods. CNNs extract spectral and spatial features for precise classification of land cover types, making them ideal for large-scale applications. Techniques like transfer learning enhance their performance, while the proposed deep Siamese CNN architecture streamlines feature extraction for automated classification. Overall, CNNs present a potent solution for HSI-based land cover mapping, revolutionizing remote sensing and geospatial sciences.

[4] This study introduces HCD-Net, a novel framework for change detection in bi-temporal hyperspectral images. HCD-Net employs dual-stream deep feature extraction, combining 3D and 2D convolution layers with Squeeze-and-Excitation (SE) blocks. Deep features from both streams are concatenated and processed through dense layers for decision-making.

SL No.	Title	Authors	Accuracy
1	Change Detection and Classification using Hyperspectral Imagery	Indira Bidari, Satyadhyan Chickerur, Akshay Kulkarni	96.6%
2	Application of Convolutional Neural Networks for Automated Land Cover Mapping using Hyper Spectral Imagery	Ananta Ojha, Rajesh Guputa, Gulista Khan	89.7%
3	Change Detection in Hyperspectral Images Using Recurrent 3D Fully Convolutional Networks	Ahram Song, , Jaewan Choi, Youkyung Han, and Yongil Kim	95.4%
4	A Hyperspectral Change Detection (HCD-Net) Framework Based on Double Stream Convolutional Neural Networks and an Attention Module	Seyd Teymoor Seydi, Mahboubeh Boueshagh , Foad Namjoo, Seyed Mohammad Minouei , Zahir Nikraftar, and Meisam Amani	96%

Methodology and Analysis:

This project's technique consists of a multi-step strategy aiming to improve hyperspectral change detection using semantic segmentation. Initial hyperspectral data preparation entails painstaking efforts to enhance data quality by addressing issues such as noise, artefacts, and atmospheric impacts. The application of sophisticated neural network designs, such as U-Net for spatial information capture for feature extraction and model depth augmentation, and Convolutional Neural Network (CNN) for efficient processing, is at the heart of the technique. These systems work together to offer a strong foundation for hyperspectral scene analysis.

Following that, labelled datasets are created with extensive annotations, allowing U-Net, and CNN to be trained on these datasets. To ensure successful learning, backpropagation and optimization techniques are used to iteratively alter model parameters. To improve the overall performance of the neural network models, hyperparameters such as learning rates and batch sizes are fine-tuned.

The critical step is to use semantic segmentation algorithms to precisely detect changing areas within hyperspectral sceneries. Trained models are used to categorize the sceneries, allowing for the separation of areas exhibiting changes over time. A thorough assessment on a variety of hyperspectral datasets follows, with measures like as accuracy, recall, and F1 score used to assess model performance. A comparison with standard change detection approaches demonstrates the suggested approach's better effectiveness.

Finally, the methodology's real-world relevance is proved by demonstrating its success in tasks such as environmental monitoring, urban planning, and disaster response. This comprehensive methodology's reliable hyperspectral change detection leads to informed decision-making across several domains.

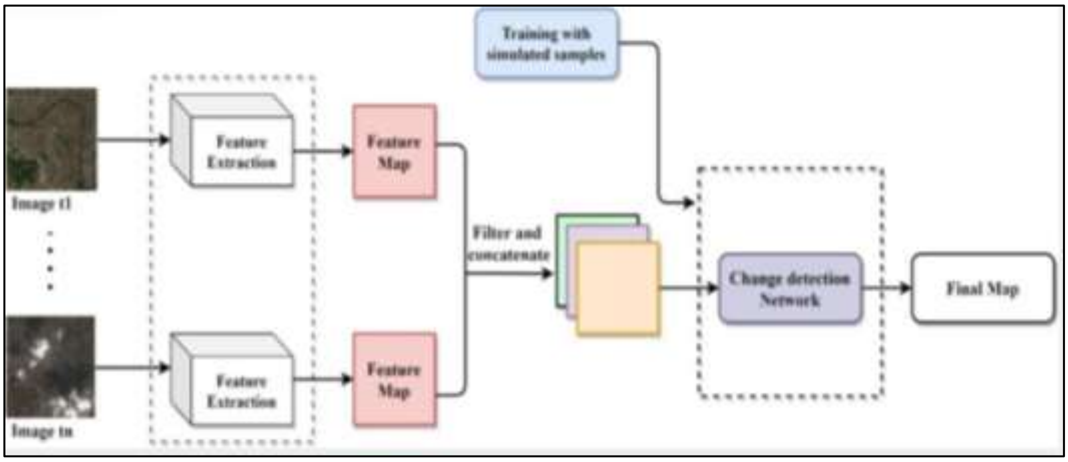


Fig.1-System design

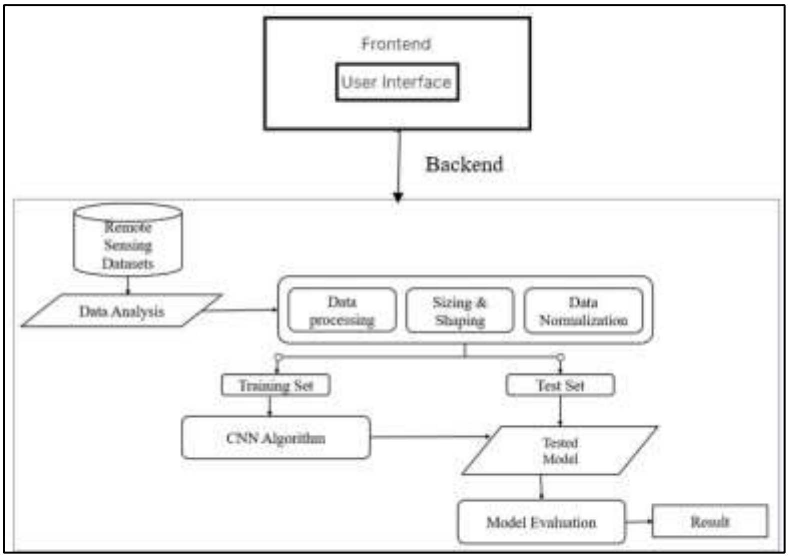


Fig 2 Architecture diagram of model

The flowchart you provided illustrates a typical data processing and machine learning pipeline, particularly using a Convolutional Neural Network (CNN) algorithm. This process begins with the Frontend User Interface, where users interact with the application. It serves as the visual and interactive layer that users see and use to input data or requests.

Next is the Backend, which encompasses all the processes that occur behind the scenes after user interaction with the frontend. It is where the core data processing and computational tasks are executed. The data utilized in the backend comes from Remote Sensing Datasets, which gather information about the Earth's surface without physical contact, typically via satellites or aircraft.

Data Analysis follows, which includes several crucial steps. Data Processing involves preparing raw data for analysis by cleaning it, handling missing values, and extracting relevant features. Sizing & Shaping adjusts the data into a format and size suitable for processing, such as reshaping arrays or matrices to fit the requirements of the CNN. Data Normalization scales input variables to a standard range, typically 0 to 1 or -1 to 1, which helps speed up the learning process of neural networks.

The processed data is then divided into two sets: the Training Set and the Test Set. The Training Set is used to train the CNN algorithm, while the Test Set is used to evaluate the trained model's performance.

The core of this pipeline is the CNN Algorithm, a type of deep neural network particularly effective for image and video recognition and processing. CNNs learn directly from the data, identifying patterns and features with minimal preprocessing.

Finally, the Model Evaluation phase involves testing the trained model using the Test Set to assess its performance on new, unseen data. The result from this evaluation includes performance metrics such as accuracy, precision, and recall.

This flowchart is typical for applications involving data-driven decision-making where remote sensing data is crucial, such as environmental monitoring, weather prediction, or land use analysis. By following these steps, the pipeline ensures that the data is effectively processed, analyzed, and used to generate accurate and reliable models for various applications.

U-Net Architecture

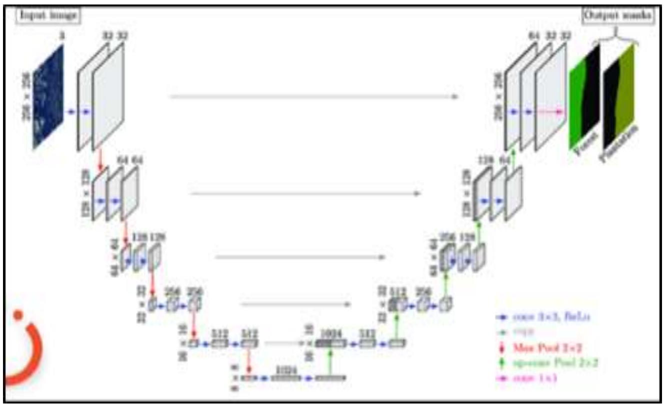
The Unet architecture is mostly used for picture segmentation.

Encoder, decoder, and skip connections make up the Unet architecture.

The job of the encoder is to build a compact representation of the input image.

The compact representation will be used to rebuild a picture by the decoder.

Skip connections are used to transfer data between encoders and decoders.



Use Case Diagram

The diagram illustrates a high-level workflow of a Change Detection System used for analysing changes in imagery data, typically hyperspectral or other remote-sensing images. The process begins with the actor, who interacts with the system to initiate the change detection process. The first step involves preprocessing images, where raw images are prepared for analysis. This includes tasks such as noise reduction, normalization, and correction of any distortions. Following preprocessing, the system proceeds to train a model on labelled data. This step involves using a dataset of labelled images where known changes have been identified and annotated, allowing the model to learn how to detect similar changes in new images. Once the model is trained, it is used to identify changed regions and segment the image. This step involves comparing images from different times

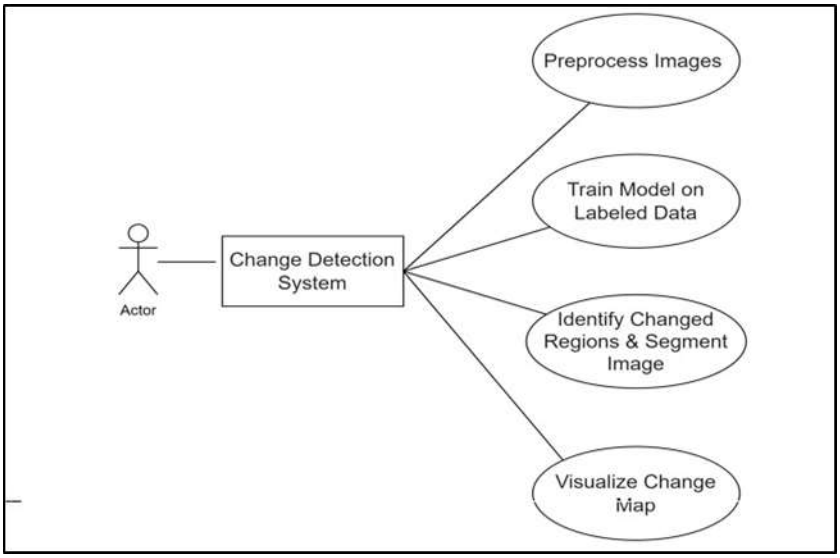


Fig 3 use case diagram

points to pinpoint areas where significant changes have occurred, segmenting these regions for further analysis. Finally, the results are visualized as a change map, which provides a visual representation of the detected changes, making it easier for the actor to interpret and analyse the changes over time. This structured approach ensures that changes in the imagery data are accurately detected, analysed, and presented in a comprehensible manner.

Interface for the change prediction

This interface is designed for hyperspectral image change detection. Here's a step-by-step explanation of its functionality:

File Selection:

Old Image: The first field allows you to upload the initial hyperspectral image (e.g., SalinasA.mat).

New Image: The second field is for uploading the new hyperspectral image taken at a later time (e.g., SalinasA_corrected.mat).

GT Image: The third field is for uploading the ground truth image (e.g., SalinasA_gt.mat), which is used for validation purposes.

Submission: After selecting the respective files, you click the "Submit" button.

Image Processing: Upon submission, the system processes both the old and new images to detect any changes that have occurred between them. This involves analysing the spectral data contained within each image.

Results Display:

Top Image: This is the visual representation of the changes detected between the old and new images. Different colours may indicate different types of changes.

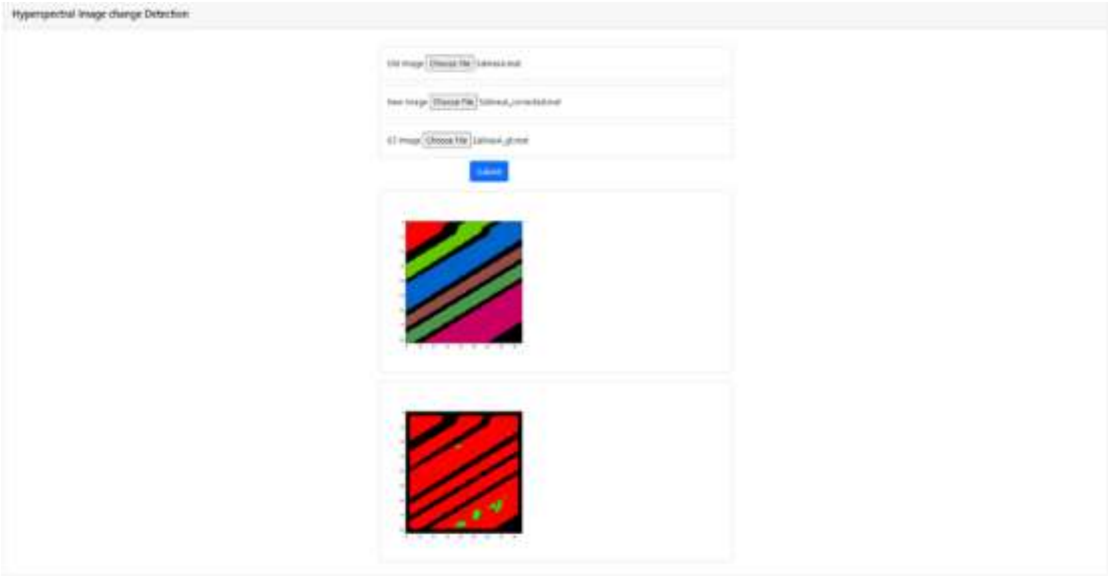


Fig 6 user interface for change detection

Bottom Image: This likely shows the ground truth image, which provides a reference for validating the accuracy of the change detection.

By comparing these images, users can observe the predicted changes and validate them against the ground truth, aiding in the analysis of temporal changes in the hyperspectral data.

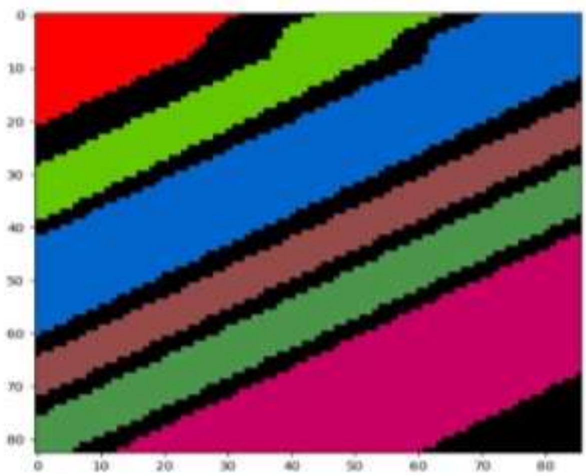


Fig 4 Actual spectral image

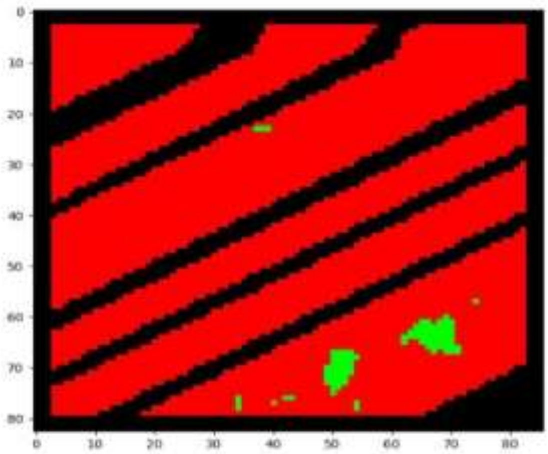


Fig 5 Change Predicted image

CONCLUSION

Our study results in a novel hyperspectral change detection framework that combines CNN, and U-Net capabilities. This powerful combination provides unmatched precision in identifying changes in environmental characteristics and land cover throughout a variety of datasets. Our solution significantly outperforms conventional techniques and, with its sophisticated and reliable change detection capabilities, revolutionizes scene analysis. This opens the door to significant applications in disaster relief, urban planning, and environmental monitoring, enabling well-informed decision-making for a more sustainable future.

References

[1] Moustafa, M. S., Mohamed, S. A., Ahmed, S., & Nasr, A. H. (2021, June 17). Hyperspectral change detection based on modification of UNet neural networks. National Authority for Remote Sensing and Space Sciences, Cairo, Egypt.

[2] Gadhave, S. M., & Chandwadkar, D. M. (2014). Change Detection Techniques. International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET), 3(12), 18434-18440.

[3] Gunturk, B. S., Kocer, B. D., & Bruzzone, L. (2014). Hyperspectral Image Analysis for Environmental Monitoring: A Review. IEEE Geoscience and Remote Sensing Magazine, 2(4), 8-23.

[4] Manapure, R. R., & Gawali, P. D. (2018). Change Detection in Satellite Images Using Convolutional Neural Networks. International Journal of Computer Applications, 179(6), 7-11.

[5] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. arXiv preprint arXiv:1512.03385.

[6] Lu, D., & Weng, Q. (2007). A survey of image classification methods and techniques for improving classification performance. International Journal of Remote Sensing, 28(5), 823-870.

[7] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.