



Image Segmentation On Satellite Images

¹Mr. K. Padmanaban, Y. Anusha

¹Assistant Professor, Dept Of MCA, Annamacharya Institute Of Technology &
Sciences, Tirupati, 517520, Andhra Pradesh, India.

²Post Graduate, Dept Of MCA, Annamacharya Institute Of Technology &
Sciences, Tirupati, 517520, Andhra Pradesh, India.

Abstract:-- The object of the study is in the area of image segmentation, specifically for satellite images, where YOLOv8 is used to detect and label certain regions in the images into three categories; Agriculture Lands, Water Bodies and Urban Buildings. These images of satellites are processed and divided into segments in order to study the use and management of land at a macro level. This division is useful in many ways including planning for cities, allocating resources and even in monitoring the effects of climate change. The framework used for the backend of the system is Flask while the frontend was developed in HTML, CSS and JavaScript. The system is accurate and timely in its operations for image categorization and partitioning of the satellite imagery useful for agriculture, city planning, and environmental conservation.

Keywords: YOLOv8, Flask framework, real-time classification.

I. INTRODUCTION

Picture division is the most common way of apportioning the satellite pictures into different regions according to the ideal highlights. Considering this, the fundamental motivation behind this task is to foster an improved satellite picture division framework taking on YOLOv8 to identify agribusiness lands, water bodies, and metropolitan structures. Applying profound learning, all the more explicitly object location, the course of division ought to be improved with the assistance of the framework created. The backend will depend on Cup to answer and serve this application, while the essential record of the frontend executed in HTML-CSS/JavaScript will permit clients to transfer satellite pictures and show results. It has expected use in regions like metropolitan and rustic turn of events, cultivating, or nature preservation where convenient aggregative data about the land use is exceptionally important.

The headways in both the remote detecting framework and progress in PC vision have upgraded the examination of satellite symbolism broadly across the various fields. One of the principal techniques for picture handling is division which is to parted a satellite picture to thusly break down it. Division in this review is restricted to three principal classes, for example, horticultural land, water bodies and structures of

metropolitan regions. With the assistance of YOLOv8 profound learning model the framework can precisely get these regions and features which assists in anticipating such exercises with preferring utilization of land, assignment of assets or in any event, for logical investigation of the climate.

Satellite picture division is extremely crucial to different government divisions, researcher, and city organizers since it supports the evaluation of land use at an enormous point of view. This congruity of expectation of land type is vital for the arrangement of satisfactory convenience for urbanization, apportion, water supply and control, and improvement of yield on the creation of harvests. Profound learning models when acquainted with remote detecting exercises have significantly improved the arrangement continuously with high accuracy. This is exacerbated by the way that YOLOv8, an inventive article identification model, gives an extremely quick and precise approach to characterizing the sorts of land thus making it truly reasonable for satellite symbolism investigation.

A. Scope of the Project

This project proposed to put to work innovative deep learning and computer vision technology to achieve this accomplishment. Thus the chief consideration is to categorize the satellite images into three broad classes, namely the Agricultural Lands, Water Bodies, and Urban Buildings. This system increases the efficiency and accuracy of the satellite image analysis with the help of YOLOv8, an advanced object detection and segmentation model. Therefore, A segmentation of the study area with the aid of GIS map is very important for aspects such as the planning of the land usage, environmental impact analysis and the allocation of the resources.

One significant advantage of using this system is that it offers an automated, real-time approach in processing the degree of satellite image; hence there will be no need for manual classification. Flask is used as the backend framework while HTML, CSS, and JavaScript that will be used to design the frontend of the project. It enabled efficient uploading of images, processing and visualization of the segmented images by the application's interactive interface. This solution is very

accurate and efficient and can be beneficial to government bodies, city policymakers, agriculturalists, and environmentalist.

Other highlighted enhancements for the future of the project include the augmenting of GIS with advanced spatial statistics facilities, utilization of machine learning in super resolution techniques, and the option for having real time active interfaces for the data. Besides, temporal analysis for instance through repeat image processing can reveal everything that has to do with land use change, urban sprawl, water resource exploitation and deforestation.

Consequently, the system's versatility allows it to be utilized in addressing and prediction of climatic change, disaster modeling and even in agriculture which involves a huge composition analysis of the land. That is why this project can be a valuable tool in any field that involves the analysis of satellite images and benefit from timely and accurate information. Keeping real-time classification and segmentation in its design also makes this platform useful in addressing the dynamic technological and the environment changes.

B. Problem Statement

Satellite image analysis may be a lengthy process and is a difficult task of segmenting the land types depending on the type of the application in the area including urban planning, land use, and conservation and agriculture among others. Conventional approaches in segmenting satellite images can take longer time or even need human interferences or utilize the older and less efficient techniques of algorithms. The absence of an automatic, real-time method of segmenting region through which one can classify these areas including agriculture lands, water bodies and urban buildings is obviously a complication in many fields. The objective of the present project is to design a system that can efficiently segment satellite images through the application of object detection and deep learning models like YOLOv8, build on its components for real-time analysis and extract useful insights from the given images.

II. LITERATURE SURVEY

The application of image segmentation is widely used in remote sensing and satellite image analysis especially in the area of land-use, environmental monitoring, and disaster management. Standard segmentation methods like focal level, region-growing and with clustering methods to select K-means and Fuzzy C means are used for relating land classification. However, these approaches fail when faced with the issue of change in the satellite images due to the change in lighting conditions, shadows, and spectral distortions which are major factors when working with the satellite images. SVM and the Random Forest classifiers were found to perform well but the process also involves heavy amount of feature extraction and preprocessing[1].

Image segmentation has been revolutionised throughout last few years through Deep Learning and especially through CNNs where U-Net and DeepLabV3+ are those architectures which are frequently used in remote sensing data semantic segmentation. However, these models require big set of annotated data and high computational resources, which results in problems with real-time application. The state-of-the-art object detection models, especially YOLO (You Only Look Once) has provided a good trade of between speed and accuracy for real-life use. The recent version presented is

YOLOv8 that has upgrades in detection, segmentation, and classification and therefore has the potential to serve as a model for satellite image segmentation[2].

A number of the published articles have shown that YOLO-based methods are very effective in distinguishing various types of the land cover including vegetation, water, and constructions. The findings indicate that YOLOv8 achieves higher classification speed and accuracy due to the improvements in the deep learning system's robustness as opposed to conventional methods. Flask has been used for backend processing and deployment and the web-based interactive front-end built with HTML, CSS, JavaScript for real-time visualization of segmented satellite imagery. This framework can be used for instances in monitoring agriculture produce, planning urban areas and implementing schemes for protection of environment[3].

Therefore, the proposed system helps to progress the area of research in remote sensing by developing a system that can scale in terms of efficiency [4]and operate in real-time for satellite image segmentation. This shows the[5] effectiveness of YOLOv8 in large scale mass endeavors involving the use[6] of land hence making it applicable in various fields and areas of use in technology.

III. METHODOLOGY

A. Proposed System

The general approach for developing the proposed system involves accurately segmenting satellite images into three major groups of land: Lands used for Agriculture, Water and Built-Up Areas or Urban Structures. To classify the land-use types from satellite images, the following system will use YOLOv8; an efficient object detection model. As such, it is significantly valuable for various applications, including the development of the city, the usage and allocation of resources in agriculture, and environmental purposes[7]

For the backend, the system uses the Flask framework: it is a lightweight API that is suitable for image processing and classification at a high speed. The frontend of the application was implemented using HTML, CSS, and JavaScript, in which the users can upload satellite images, and immediately get the segmented result. This paper describes how the system receives and analyzes images, uses YOLOv8 to segment them, and produces images with annotations that outline the lands of different categories.[8]

It's for this reason, other preprocessing methods like image normalization, image enhancement, and image denoising is first performed on images before feeding them into the YOLOv8. Further, the system is capable of online processing of the text and provides results in real-time, which is very crucial for the application where quick results are required such as after disaster or climate change impacts.[9]

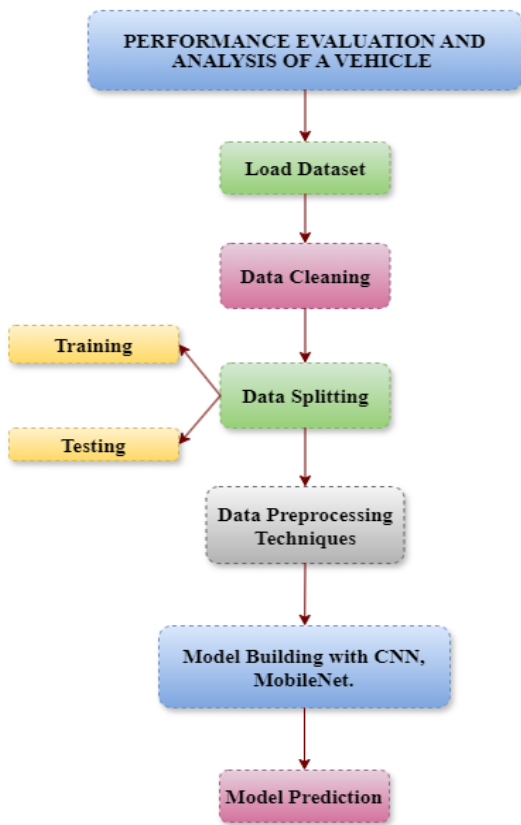


Figure 1 Project Flow

IV. IMPLEMENTATION

A. Dataset

There are more than 5000 satellite images in raw form obtained from platforms namely Google Earth, Sentinel-2, and Landsat 8. These images are grouped into three parts, namely; Agricultural Lands, Water Bodies, and Urban Buildings. All images are georeferenced and provided with geographic coordinates and augmented with additional attributes such as the NDVI, the land cover type, and the number of dwellings per hectare. This helps in the segmentation and classification of satellite imagery because it was designed to train deep learning models such as YOLOv8. The labeled data is therefore useful in segmentation of land use particularly useful in planning, identification of environmentally sensitive areas, and subsequently facilitating in proper usage of available agricultural tracts.[10]

B. Preprocessing Steps

a) Data Cleaning and Annotation:

To ensure data quality, duplicate and corrupted images were removed. Manual and semi-automated annotation of regions was performed using tools like **LabelImg** and **Roboflow**, ensuring precise class labels for accurate model training.

b) Image Resizing and Normalization

All images were resized to **640x640 pixels** to maintain consistency across the dataset. Pixel values were normalized to the **[0,1] range**, which enhances model efficiency by ensuring stable gradients during training.

c) Data Augmentation

To improve dataset diversity and prevent overfitting, augmentation techniques such as **rotation, flipping, brightness/contrast adjustment, and random cropping** were applied. These augmentations were implemented using **Albumentations** and **OpenCV**, effectively increasing the model's generalization ability.

d) Splitting the Dataset:

The dataset was divided into three subsets: **training (70%)**, **validation (20%)**, and **testing (10%)**, ensuring balanced data distribution for effective model generalization and evaluation.[11]

C. Model Training

YOLOv8 is from the family of YOLO, a real-time object detection model that is the latest in the series. Just as others in its family, the YOLOv8 is fast and accurate thus making it suitable for high-performance object detection. categorical object detection was solved using R-CNN models, YOLO models approach object detection as one regression problem which is a significant plus compared to multiple steps such as the region proposals.[12]

How YOLOv8 Works

1. Input Image Processing

- Simple pre-processing of the input image is done in the form of resizing the image to a certain fixed size of perhaps (640 \times 640) and normalizing the pixel intensities to a range of values such as [0, 1].
- YOLOv8 first segments the image into different cells and then passes the segmented image to a CNN in order to get the features of the image.[13]

2. Backbone Network

- YOLOv8 has CSP-Darknet (Cross Stage Partial Darknet) architecture for feature extraction in its design. This is a convolutional neural network and the layer i named is responsible of capturing the spatial pattern in an input image.
- In the backbone, the number of features is decreased in the spatial area, and the depth of feature maps is added through several layers.

3. Feature Pyramid Network (FPN) and Path Aggregation Network (PAN)

- YOLOv8 integrates **Feature Pyramid Network (FPN)** and **Path Aggregation Network (PAN)** to enhance multi-scale feature extraction, ensuring that both fine and coarse details are captured effectively.
- FPN extracts features from different hierarchical levels, helping in detecting objects of varying sizes. PAN further refines these features by improving spatial information flow, leading to better object localization.

4. Head and Detection Layers

- The detection **head** is responsible for generating bounding boxes and predicting class probabilities.
- YOLOv8 employs multiple scales, typically three, to enhance detection accuracy for small, medium, and large objects. These scales connect to detection

layers that output bounding box coordinates, an object confidence score, and class predictions.

5. Bounding Box Prediction

- YOLOv8 determines bounding box coordinates using anchor-based predictions and regression techniques.
- Each grid cell predicts multiple bounding boxes at different scales, considering anchor box dimensions.
- The bounding box is parameterized by:
 - $\text{tx}, \text{ty}, \text{t}_x, \text{t}_y$: Adjustments to the center coordinates relative to the grid cell.
 - $\text{tw}, \text{th}, \text{w}, \text{h}$: Scaling factors for width and height in relation to the anchor box.

6. Bounding Box Calculation

- If cxc_xcx and cyc_ycy represent the center of a grid cell, the predicted bounding box center bxb_xbx and byb_byb are derived as:

$$\begin{aligned} \text{bx} &= \text{cx} + \sigma(\text{tx}) \cdot \text{b_x} = \text{c_x} + \sigma(\text{tx}) \cdot \text{b_x} \\ \text{by} &= \text{cy} + \sigma(\text{ty}) \cdot \text{b_y} = \text{c_y} + \sigma(\text{ty}) \cdot \text{b_y} \end{aligned}$$

- The width bwb_wbw and height bhb_bhb are calculated as:

$$\text{bw} = \text{p_w} \cdot \text{e}^{\text{tw}} \quad \text{bh} = \text{p_h} \cdot \text{e}^{\text{th}}$$

where p_wp_wp and p_hp_hp are anchor box dimensions, and σ represents the sigmoid activation function, ensuring tx and ty remain between 0 and 1.

7. Objectness Score and Class Prediction

- Each predicted bounding box is assigned an **objectness score**, indicating the likelihood of an object being present in that box.
- This score is computed using a **logistic regression model**.
- For classification, YOLOv8 calculates individual scores for each class, determining the most probable class for the detected object.

8. Non-Maximum Suppression (NMS)

- To eliminate duplicate bounding boxes for the same object, **Non-Maximum Suppression (NMS)** is applied.
- This involves computing the **Intersection over Union (IoU)** between pairs of overlapping bounding boxes.
- If the **IoU** exceeds a predefined threshold (e.g., 0.5), the box with the lower confidence score is discarded, keeping only the most confident detection.

9. Loss Function

YOLOv8 optimizes its performance by minimizing three primary loss components:

- Localization Loss (IoU Loss):** Measures the discrepancy between predicted and actual bounding boxes.
- Confidence Loss:** Evaluates the accuracy of the model's objectness score predictions.
- Classification Loss:** Determines how accurately the predicted label matches the actual object class.

YOLOv8 Loss Function

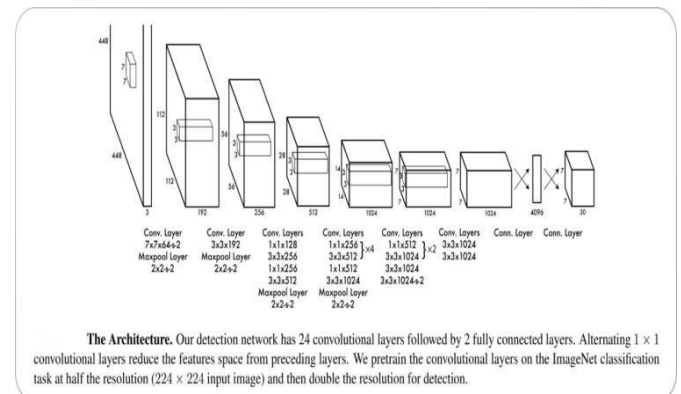
The total loss function combines these three components:

$$\begin{aligned} L_{\text{total}} &= \lambda_{\text{loc}} L_{\text{loc}} + \lambda_{\text{conf}} L_{\text{conf}} + \lambda_{\text{cls}} L_{\text{cls}} \\ &= \lambda_{\text{loc}} L_{\text{loc}} + \lambda_{\text{conf}} L_{\text{conf}} + \lambda_{\text{cls}} L_{\text{cls}} \end{aligned}$$

where:

- $\lambda_{\text{loc}}, \lambda_{\text{conf}}, \lambda_{\text{cls}}$ are weight factors for each loss component.
- Localization Loss** relies on IoU-based metrics.
- Confidence Loss** employs **binary cross-entropy** for object presence prediction.
- Classification Loss** applies **cross-entropy** for multi-class classification.

This combination ensures that YOLOv8 effectively balances detection accuracy, confidence estimation, and classification precision.



V. RESULTS

Class	Images	Instances	Box(P)	R	mAP50	mAP50-95)	Mask(P)	R	mAP50	mAP50-95)
all	36	171	0.523	0.441	0.47	0.311	0.537	0.456	0.464	0.332
Agriculture-lands	16	63	0.411	0.302	0.343	0.21	0.368	0.27	0.275	0.175
Water-	32	30	0.806	0.764	0.841	0.622	0.89	0.842	0.806	0.717
urban-building	11	70	0.354	0.257	0.226	0.103	0.354	0.257	0.21	0.102

Figure 4 :Results

This summary shows the validation results of a YOLO model after 100 training epochs. Each defect class (e.g., crazing, inclusion) has metrics like precision (P), recall (R), mAP@50, and mAP@50-95. The overall metrics indicate

moderate performance, with high precision and recall in some classes (e.g., patches, scratches) and lower values in others, suggesting varied accuracy across classes. Results are saved for further analysis.

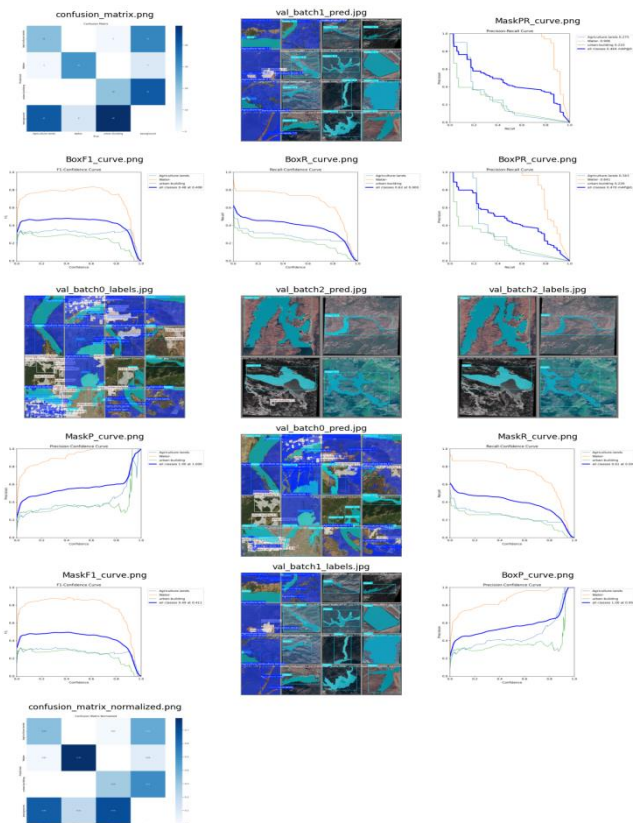


Figure 5 :Yolo v8 Performance

The learning curve for box loss illustrates the training and validation losses over 100 epochs. The training loss consistently declines, demonstrating effective model learning. However, the validation loss fluctuates and settles at a relatively higher level, suggesting potential challenges in generalization or slight overfitting. Adjusting hyperparameters may help in minimizing the validation loss.

VI. CONCLUSION

The study effectively applies an image segmentation technique using YOLOv8 to analyze satellite imagery, categorizing land into three primary types: agricultural areas, water bodies, and urban structures. The developed system efficiently processes satellite images, enabling precise segmentation and classification, which is beneficial for diverse applications such as urban planning, agricultural resource management, and environmental assessment. The backend operations are managed using Flask, while the frontend is built with HTML, CSS, and JavaScript, ensuring a seamless user experience with real-time image processing capabilities.

By dividing satellite images into distinct regions, the system supports data-driven insights, empowering policymakers, researchers, and urban planners to assess land utilization more effectively. Its high accuracy and quick processing make it a valuable tool for sustainable urban expansion, optimized agricultural planning, and monitoring climate change impacts on water bodies and urban landscapes. Moreover, the automation of image segmentation minimizes manual effort, enhances operational efficiency, and improves scalability for large-scale satellite data analysis.

VII. FUTURE SCOPE

The future scope of this project involves improving segmentation accuracy by incorporating advanced transformer-based vision models such as the Segment Anything Model (SAM) alongside YOLOv8 to enhance precision in satellite image classification. The integration of AI-powered change detection algorithms can facilitate the monitoring of environmental transformations over time, contributing to urban expansion analysis and disaster management. Additionally, leveraging cloud computing for large-scale, real-time image processing will enhance the system's scalability. The framework can also be expanded to identify additional land-use categories, including forests, wetlands, and barren lands. Moreover, the adoption of explainable AI (XAI) techniques will help in interpreting model predictions, increasing transparency and reliability in critical applications.

VIII. REFERENCES

- [1] K. G. Arun Depak, S. Saikrishnan, A. J. Sudhakar, and K. Kaviyarasan, "A Comprehensive Web Application for Chronic Kidney Disease Prediction with Cuisine-Centric Diet Recommendation," *International Conference on Self Sustainable Artificial Intelligence Systems, ICSSAS 2023 - Proceedings*, pp. 891–896, 2023, doi: 10.1109/ICSSAS57918.2023.10331627.
- [2] D. Dai and W. Yang, "Satellite image classification via two-layer sparse coding with biased image representation," *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 1, pp. 173–176, Jan. 2011, doi: 10.1109/LGRS.2010.2055033.
- [3] T. Wang *et al.*, "Geometric accuracy validation for ZY-3 satellite imagery," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 6, pp. 1168–1171, Jun. 2014, doi: 10.1109/LGRS.2013.2288918.
- [4] J. Song, J. H. Jeong, D. S. Park, H. H. Kim, D. C. Seo, and J. C. Ye, "Unsupervised Denoising for Satellite Imagery Using Wavelet Directional CycleGAN," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 8, pp. 6823–6839, Aug. 2021, doi: 10.1109/TGRS.2020.3025601.
- [5] M. Wu, C. Zhang, J. Liu, L. Zhou, and X. Li, "Towards Accurate High Resolution Satellite Image Semantic Segmentation," *IEEE Access*, vol. 7, pp. 55609–55619, 2019, doi: 10.1109/ACCESS.2019.2913442.
- [6] L. Shen, H. Tang, Y. Chen, A. Gong, J. Li, and W. Yi, "A semisupervised latent dirichlet allocation model for object-based classification of VHR panchromatic satellite images," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 4, pp. 863–867, Apr. 2014, doi: 10.1109/LGRS.2013.2280298.
- [7] H. Jia, K. Sun, W. Song, X. Peng, C. Lang, and Y. Li, "Multi-Strategy Emperor Penguin Optimizer for RGB Histogram-Based Color Satellite Image Segmentation Using Masi Entropy," *IEEE Access*, vol. 7, pp. 134448–134474, 2019, doi: 10.1109/ACCESS.2019.2942064.

- [8] Y. Zhou, J. Li, L. Feng, X. Zhang, and X. Hu, "Adaptive Scale Selection for Multiscale Segmentation of Satellite Images," *IEEE J Sel Top Appl Earth Obs Remote Sens*, vol. 10, no. 8, pp. 3641–3651, Aug. 2017, doi: 10.1109/JSTARS.2017.2693993.
- [9] K. Sasaki, T. Sekine, and W. Emery, "Enhancing the Detection of Coastal Marine Debris in Very High-Resolution Satellite Imagery via Unsupervised Domain Adaptation," *IEEE J Sel Top Appl Earth Obs Remote Sens*, vol. 17, pp. 6014–6028, 2024, doi: 10.1109/JSTARS.2024.3364165.
- [10] K. Segl and H. Kaufmann, "Detection of small objects from high-resolution panchromatic satellite imagery based on supervised image segmentation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 9, pp. 2080–2083, Sep. 2001, doi: 10.1109/36.951105.
- [11] L. Duan and X. Hu, "Multiscale Refinement Network for Water-Body Segmentation in High-Resolution Satellite Imagery," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 4, pp. 686–690, Apr. 2020, doi: 10.1109/LGRS.2019.2926412.
- [12] I. H. Lee and M. T. Mahmood, "Robust registration of cloudy satellite images using two-step segmentation," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 5, pp. 1121–1125, May 2015, doi: 10.1109/LGRS.2014.2385691.
- [13] M. Awad, K. Chehdi, and A. Nasri, "Multicomponent image segmentation using a genetic algorithm and artificial neural network," *IEEE Geoscience and Remote Sensing Letters*, vol. 4, no. 4, pp. 571–575, Oct. 2007, doi: 10.1109/LGRS.2007.903064.