iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 Issue: IV Month of publication: April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79475

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Real-Time Satellite Image Segmentation Using YOLOv8

Mrs. R. Prathyusha[1], G. Humsika[2], P. Saketh[3], S. Yashwant Raj Reddy[4]
[1]Assistant Professor , [2, 3, 4]Students, Department of Computer Engineering, Methodist College of Engineering and Technology Abids, Hyderabad, Telangana, 500001, India.

***Abstract:** The project focuses on real-time satellite image segmentation using YOLOv8 for detecting and classifying land regions into three categories: Agriculture Lands, Water Bodies, and Urban Buildings. By leveraging advanced deep learning and object detection models, the system automates segmentation, providing higher accuracy and speed compared to traditional methods. The backend is developed using Flask, while the frontend uses HTML, CSS, and JavaScript to offer a user-friendly web interface for uploading images and visualizing results. This solution can significantly contribute to urban planning, environmental monitoring, and resource management, enabling timely insights into land usage for sustainable development.*
***Keywords:** YOLOv8, Satellite Image Segmentation, Flask Framework, Real-time Classification, Sustainable Land Monitoring.*

I. INTRODUCTION

Satellite image segmentation is an important method used to analyze Earth observation data. In this project, we develop a real-time segmentation system of automatically identifying features such as farmland, water bodies, and urban areas using YOLOv8. As YOLOv8 is a highly advanced deep learning model for object detection, it ensures that the segmentation process takes place much faster and more accurately than traditional methods. A Flask-based backend handles image processing and model inference, while a web interface built using HTML, CSS, and JavaScript allows users to upload images and visualize the results easily. HTML, CSS, and JavaScript, is an interface where users can upload satellite images to view the output. Such a system is highly beneficial in regions such as agriculture, urban development, and environmental studies, where fast and accurate land use information is of utmost significance.With the advancement of satellite technology and the production of high-resolution images, there is an increasing need for intelligent systems that can analyze these images efficiently. Conventional systems tend to fail because satellite images can be quite different in terms of quality, lighting, and landscape. This project aims to address these issues by developing a contemporary segmentation system that can efficiently classify different land use types in different landscapes.

The power of the system largely lies in the YOLOv8, which is capable of fast and accurate detection even for large datasets. When integrated with a Flask server, the system can analyze the uploaded images seamlessly and provide the segmented result without any significant delay. The system is user-friendly, even for those who are not conversant with machine learning and satellite image analysis.This project also has the potential for expansion. In the future, the system could handle additional classes, multispectral or hyperspectral images, or additional data such as NDVI maps or elevation data. These additions would allow the segmentation to be much more detailed and helpful, particularly in regions where visual information is not sufficient on its own.

The model is trained carefully to achieve reliable performance by using techniques such as data augmentation, transfer learning, and fine-tuning of parameters. To evaluate how well the model performs, metrics like IoU, precision, recall, and F1-score are used, providing a clear understanding of its accuracy and effectiveness.

In terms of deployment, the system is efficient and scalable. The Flask server can be integrated with cloud services such as AWS, Azure, or Google Cloud to handle large amounts of data. The workloads and processing of satellite images can be done faster using GPUs. The lightweight frontend ensures smooth interaction with the user, even when using a slow internet connection. Overall, the project offers a very useful and friendly tool that can be used to support research, industry, and government activities in the area of geospatial analysis.

Another key aspect of the project is ensuring that the system is designed to perform effectively under different real-world conditions. Satellite images often change due to factors such as climate conditions, seasonal variations, sensor types, and geographical differences, making the analysis more challenging. Due to such differences, developing a segmentation system that can handle all scenarios effectively can be challenging. To overcome this, the dataset used for training purposes needs to have a range of images that vary depending on the climate, terrain, and lighting conditions.As the system is further developed, more data sets from international satellite libraries can also be integrated to further enhance the accuracy of the system.

The goal of this project is to design a system that appears simple, intuitive, and useful. By providing a browser-based interface, this project eliminates such challenges and enables anyone, from students to agricultural officers, to analyze satellite images with ease. Uploading an image, performing segmentation, and displaying results can all be accomplished in a matter of clicks. Even for organizations with limited budgets, this system can be an inexpensive and effective solution for land use change detection or environmental analysis without the need for expensive computational infrastructure. For example, agricultural departments can utilize this system to estimate crop area, detect regions that might be drought-affected, or detect regions where land clearing is being done illegally. Urban planners can analyze construction trends, track population growth, and forecast regions for future development. Environmental departments might utilize the segmentation output to track regions where water bodies are shrinking, measure deforestation levels, or detect regions that are likely to be affected by flooding. With the addition of analytical modules and automation capabilities, the software can develop from a basic segmentation tool into a full-fledged intelligence system that enables data-driven planning and decision-making. These functionalities would greatly enhance its utility for both the public and private sectors.

Finally, the long-term maintenance and development of the model are crucial to ensuring that the system remains efficient and effective. The technology for satellite imaging is also constantly advancing, with new sensors, resolutions, and data formats emerging every year. To stay current, the segmentation model would need to be updated periodically with new data, and its underlying architecture may need to be upgraded to accommodate more sophisticated functionalities. Community suggestions can also help shape the future course of development, for instance, users may suggest new segmentation options, drone image support, or faster processing speeds. With a modular architecture, these developments can be achieved without having to rebuild the entire system. Over time, the project can potentially mature into a widely used platform for geospatial analysis and analysis-related applications in various sectors around the world.

II. MOTIVATION

This project is motivated by the growing need to process satellite images quickly, accurately, and on a large scale.With global populations growing and environmental challenges becoming more urgent, it is essential to keep track of how land is being used, how vegetation is changing, and how water resources and urban areas are developing over time. Relying on manual interpretation is no longer practical because it takes too long, introduces human errors, and cannot handle the massive volume of modern satellite data. This is why automated semantic segmentation tools have become so important. By classifying satellite images into clear categories such as farmland, water bodies, forest cover, barren land, and urban zones, these systems provide valuable information for planning and monitoring activities. These insights play important role in areas such as agriculture, urban development, disaster management, and environmental monitoring.Among the available models, YOLOv8 is especially effective due to its high speed and reliable performance in both detection and segmentation tasks. By using YOLOv8 for satellite image analysis, the system can handle large volumes of data efficiently while still delivering accurate results. This makes it possible to automate tasks like tracking city growth, detecting deforestation, monitoring crop conditions, and identifying regions facing water shortages. The ability to analyze this information quickly helps organizations respond to challenges faster and make better decisions.Furthermore, automating these processes reduces the workload for analysts and minimizes the chances of human error or misinterpretation. It also ensures that large-scale geographic changes can be monitored in a consistent and timely manner, something that manual methods often fail to achieve. As a result, the system becomes a dependable tool for long-term environmental and developmental studies. In addition, automated segmentation allows organizations to respond more quickly to sudden changes such as floods, wildfires, or drought conditions. It supports better planning by providing continuous and up-to-date insights without requiring constant human intervention. Overall, this level of automation makes the entire analysis process smoother, more efficient, and far more practical for real-world applications.

III. WORKING PRINCIPLE

Satellite image segmentation plays a crucial role in modern geospatial intelligence. Without advanced deep learning techniques, most land-use analysis systems would struggle to classify and detect features within high-resolution imagery. Automated segmentation enables faster interpretation, improves accuracy, and supports decision-making in critical areas such as agriculture, urban development, and environmental changes.

1) Developing the Satellite Image Acquisition and UploadInterface

A dedicated front-end module allows users to upload satellite images in common formats. This module ensures that the system receives high-quality inputs suitable for segmentation and further analysis.

2) Implementing the Preprocessing Pipeline for Image Standardization

Each uploaded image undergoes preprocessing techniques like image resizing, normalization and colour-space alignment. This ensures compatibility with the YOLOv8 architecture and optimal inference results.

3) Constructing a Multilevel Segmentation and Detection Model Using YOLOv8

Through feature extraction, bounding-box prediction, and mask generation, the model identifies agricultural lands, water bodies, urban buildings, and other land-cover categories. The deep learning framework evaluates multilevel spatial relationships in the image to generate accurate segmentation outputs.

4) Applying Post-Processing and Relevance Filtering forSegmentation Results

After inference, segmentation masks are refined using thresholding, contour enhancement, and class-based filtering. This ensures that only the most relevant and accurate land-use regions appear in the final visual

Finally, all modules image acquisition, preprocessing, multilevel segmentation, and post-processseamlessly integrated within a Flask-based backend and a user-friendly web interface to create a cohesive and efficient geospatial analysis system. The Flask server manages all core operations, including receiving uploaded satellite imagery, executing preprocessing algorithms, running YOLOv8 inference, and returning the processed segmentation masks to the client. This modular architecture ensures high scalability, allowing the system can run both on local machines and cloud-based platforms. On the front-end, an interactive interface built with HTML, CSS, and JavaScript enables users to preview images, visualize segmentation overlays, and interact with detection outputs in real time. Additional functionalities such as region-area measurement tools, downloadable mask layers, map-based coordinate visualization, and temporal comparison modules (for analyzing changes across historical imagery) can be incorporated to extend system capabilities.

IV. LITERATURESURVEY

Krizhevsky, Sutskever, and Hinton (2012) introduced AlexNet, a deep learning model that brought a major breakthrough in computer vision by showing that neural networks can automatically learn usefulfeatures like edges, textures, and shapes of objects; their use of GPU-based training made it possible to train large models efficiently, while techniques like ReLU activation, dropout, and data augmentation improved accuracy and reduced overfitting, making it a foundation for many modern detection and segmentation models [1].

Simonyan and Zisserman (2014) developed the VGG network, which demonstrated that increasing network depth improves performance, and by using small 3×3 convolution filters repeatedly, the model was able to capture fine details like textures and edges more effectively while keeping the architecture simple and easy to use as a backbone in many applications [2].

Long, Shelhamer, and Darrell (2015) introduced Fully Convolutional Networks (FCN), which changed segmentation by enabling pixel-level predictions instead of classifying the whole image, and by using skip connections, the model combined high-level and low-level features to produce more accurate boundaries and detailed segmentation maps [4].

Zhang, Ren, and Sun (2016) presented ResNet, which solved the problem of vanishing gradients in deep networks by introducing skip connections that allow information to pass directly across layers, making it possible to train very deep models that can extract both fine details and global context effectively [5].

Cheng et al. (2018) proposed a hierarchical CNN approach for hyperspectral image classification, where features from different levels are combined to better understand both spatial and spectral information, helping in handling complex data with many spectral bands and improving classification accuracy [6].

Zhu et al. (2018) introduced Generative Adversarial Networks (GANs) for hyperspectral image classification, where a generator creates realistic data samples and a discriminator evaluates them, allowing the model to learn better representations even when labeled data is limited, thus improving generalization and robustness [7].

Barthakur and Sarma (2019) showed that CNN-based semantic segmentation performs much better than traditional image-processing techniques like clustering and edge detection, as it can handle real-world challenges such as shadows, seasonal variations, and overlapping land patterns, producing cleaner segmentation outputs and reducing classification errors [8].

Zhou et al. (2019) developed a stacked autoencoder model that learns compact and meaningful features from hyperspectral data without requiring large amounts of labeled data, making the system more efficient and suitable for real-time applications while maintaining good accuracy [9].

Wurm et al. (2019) applied deep learning combined with transfer learning has been used to identify slum areas in satellite images, demonstrating that pre-trained models can be fine-tuned for new tasks even with limited data.

This approach is particularly effective in recognizing complex and densely populated urban regions with irregular structures. [10].

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

In 2024, researchers introduced transformer-based models for satellite image segmentation, which analyze the entire image and capture relationships between distant regions, allowing better understanding of large-scale patterns and improving segmentation accuracy in high-resolution images with mixed land-use areas and complex backgrounds [11].

In 2025, advanced models such as improved YOLO versions and self-supervised learning techniques were developed, which increased processing speed and reduced dependency on labeled datasets by allowing models to learn patterns automatically, making satellite image analysis faster, scalable, and more effective for real-time applications like disaster monitoring, environmental analysis, and urban planning [12].

V. SYSTEM ARCHITECTURE

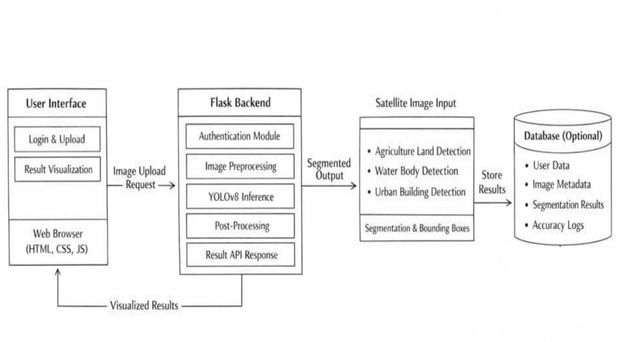


Fig1:-System Architecture

The proposed system solution uses a web-based client–server setup to analyze satellite images and automatically identify different land-cover types using deep learning.

The framework integrates a user interface, a backend processing server, a satellite image analysis module, and an optional database for storing results and metadata.

The workflow begins with the user interface, which is accessed through a web browser built using HTML, CSS, and JavaScript. This interface allows users to securely log in and upload satellite images for analysis. Once the image is uploaded, a request is sent to the backend server for processing. The interface also provides result visualization, enabling users to view segmentation outputs and detected objects in an intuitive format.

The backend of the system is built using a Flask server that handles all major processing tasks. Initially, a login module checks the user's credentials to ensure secure access. Once the user is authenticated, the uploaded image is processed by applying steps such as resizing and normalization to make it suitable for analysis. After this preparation, the image is fed into the YOLOv8 model, which carries out object detection and segmentation on the satellite data.

During the inference stage, the model identifies key land-cover features such as agricultural land, water bodies, and urban buildings. The detected regions are represented using segmentation masks and bounding boxes. After inference, a post-processing step refines the predictions by filtering low-confidence detections.

The processed results are returned to the frontend through an API response, where they are visually presented to the user. This allows users to easily interpret the detected features and segmentation outputs directly within the web interface.

Optionally, the system includes a database component for storing relevant information. The database maintains user data, image metadata, segmentation outputs, and accuracy logs. This storage capability supports future analysis, performance tracking, and reproducibility of results.

Overall, this architecture provides a scalable andefficient solution for automated satellite image analysis. By combining a web-based interface with deep learning inference and optional data storage, the system enables users to easily upload satellite images, perform land-cover detection, and visualize results in a streamlined workflow

VI. METHODOLOGY

The proposed system for satellite image segmentation and classificationfollowsa well-defined methodology designedto ensurehigh accuracy,robustness, andscalability.The key steps in the methodology are described below:

1) Data Collection

Satellite imagery and hyperspectral datasets are collected from reliable sources such as ISPRS, NASA, Sentinel missions, and Google Earth Engine. The datasets include 1185abelled ground-truth annotations necessary for training and evaluating the models. Efforts are made to include a diverse range of land-cover types to improve the generalization capability of the system across different geographic regions.

2) Data Preprocessing

Raw satellite images are to ensure they are suitable with deep learning models. This involves steps like resizing the images, normalizing pixel values, and reducing noise to improve quality. To make the model more accurate data augmentation methods such as rotation, flipping, and scaling are applied, which also help in expanding the dataset and minimizing overfitting. In the case of hyperspectral images, the spectral bands are properly organized, and dimensionality reduction techniques are used when necessary to handle the large number of channels efficiently.

3) Feature Extraction

Deep learning methods are used to extract meaningful spectral and spatial features from the imagery. Convolutional neural networks (CNNs) and hierarchical feature learning layers capture multi-level spatial information, while stacked autoencoders (SAEs) and generative adversarial networks (GANs) learn compact and discriminative representations from hyperspectral data. This combination allows the system to effectively characterize the complexity of satellite images and capture subtle variations in land-cover types.

4) Model Selection

Appropriate deep learning architectures are chosen based on the dataset characteristics and segmentation requirements:

- FCN or U-Net for pixel-level semantic segmentation
- ResNet for deep residual feature learning
- R-CNN variants for object detection
- GAN-based models for hyperspectral classification

The choice of model depends on the target land-cover categories, dataset type, and desired output resolution.

5) Model Training

Selected models are trained using high-performance GPU environments. Optimization algorithms such as Adam and SGD are employed, while loss functions like crossentropy and Dice loss help manage class imbalances and improve segmentation accuracy. Key performance metricsincluding accuracy, precision, recall, F1-score, and Intersection-over-Union (IoU)are tracked to ensure proper convergence and to prevent overfitting.

6) Testing and Validation

The trained models are evaluated on separate test datasets to assess their generalization capabilities. Performance is compared against classical machine learning approaches and traditional image-processing techniques. Both quantitative metrics and qualitative visual assessments are used to validate improvements in segmentation precision and robustness.

7) Results and Visualization

Final outputs are visualized as classified and segmented maps highlighting different land-cover types. Geographic Information System (GIS) tools are used to overlay these results on satellite imagery for clear spatial interpretation. These visualizations is useful for environment study, urban planning, disaster control, and agriculture.

VII. DATASET

The dataset used in this study consists of satellite images representing different land cover regions. The images capture various geographical patterns including agricultural areas, water bodies, and urban infrastructure. These images are used to train and evaluate deep learning models for automated land cover detection and segmentation.

Each image in the dataset is provided in RGB format and stored in JPEG format, making it suitable for processing with deep learning frameworks.

The dataset includes diverse spatial patterns and textures that help the model learn different land characteristics present in satellite imagery.

To enable supervised learning, all images are annotated with polygon-based segmentation labels. The annotation files contain normalized coordinate values that represent the boundary of each object present in the image. These annotations follow the YOLO segmentation labeling format, where each label file corresponds to a specific image and contains the class identifier along with the polygon coordinates describing the object region.

The dataset focuses on three major land cover categories:

Agriculture Land,Water Bodies,Urban Buildings.

Each image may contain multiple instances of these classes, allowing the model to learn complex spatial relationships between different land regions. The annotations help the model accurately detect and segment different land cover types within the satellite imagery.

Before training the deep learning model, the dataset undergoes preprocessing steps such as image resizing, normalization, and augmentation. These preprocessing techniques help improve model performance and enhance generalization during training.

This dataset provides a diverse representation of land cover regions and is suitable for developing and evaluating deep learning–based satellite image segmentation systems.

Fig2:-Data Set Images

VIII. TECHNOLOGIES USED

The satellite image segmentation system is built using different deep learning models and tools. Convolutional Neural Networks (CNNs), including Alex Net, VGG, and ResNet, are used to extract important features from satellite images. These models help in identifying both basic patterns and more complex information in the data. Fully Convolutional Networks (FCNs) are used for detailed pixel-level segmentation, allowing accurate identification of different land-cover areas.Transfer learning is applied to improve performance and reduce training time, especially when there is limited data available. Generative Adversarial Networks (GANs) are used to create additional data samples, which helps improve model accuracy. Stacked Autoencoders (SAEs) are used to learn important features from hyperspectral images while reducing unnecessary data and maintaining useful information.

Hierarchical CNN FeaturesThese deep-learning libraries facilitate model development, training, evaluation, and deployment on high-performance systems. Remote Sensing & GIS Tools: Software platforms are used for satellite image preprocessing, visualization, and geospatial mapping of segmentationresults. Image Processing Techniques: The system uses steps like resizing, normalization, noise removal, and data augmentation. GPUs are used to speed up training and handle large data quickly.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

IX. RESULT

USER INTERFACE PAGE:

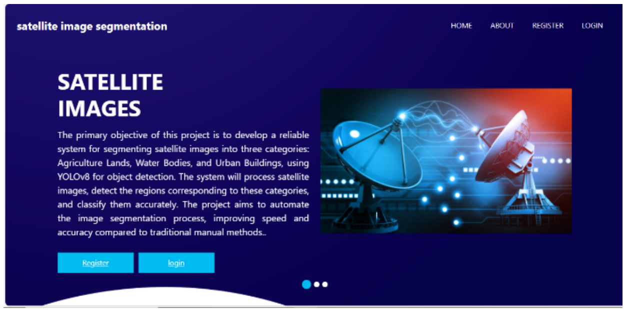


Fig3: Home Page

ABOUT PAGE :

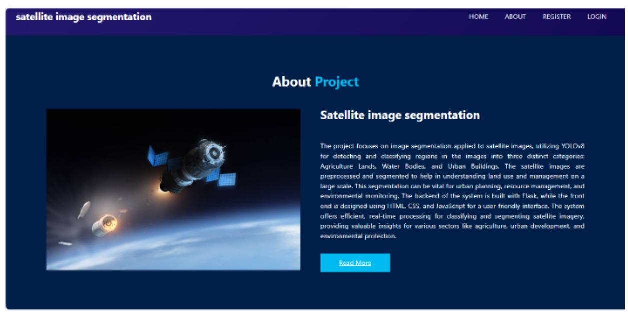


Fig4:YOLOv8-based satellite image classification into land categories

REGISTER PAGE:

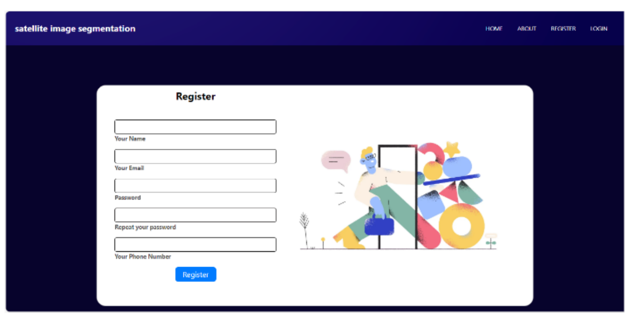


Fig5: User Registration Page

LOGIN PAGE:

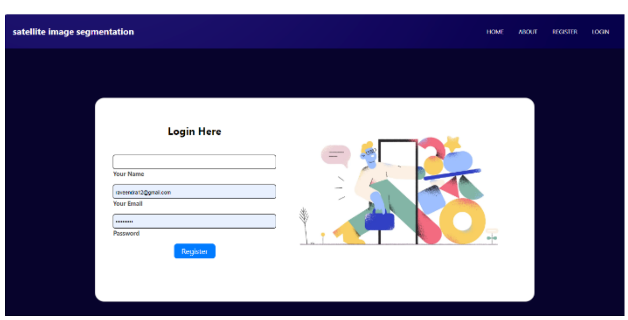


Fig6: User Login Page

PREDICTION PAGE

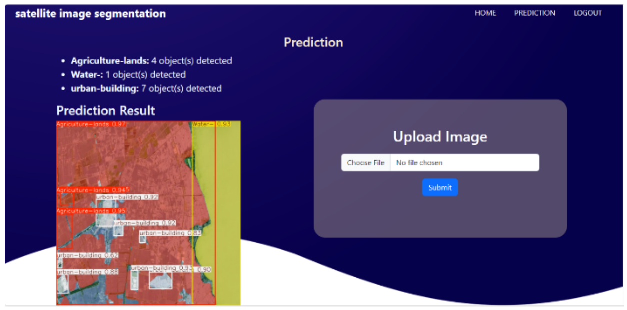


Fig7:Prediction and Segmentation Page

X. FUTURE SCOPE

The future scope of this project includes integrating the segmentation tool with other technologies like geographic mapping and satellite monitoring systems to enhance its real-world applications. Additionally, expanding the system to handle a wider variety of land categories and larger datasets will increase its utility. By incorporating more advanced machine learning models and optimizing for edge devices, the system can be deployed in diverse environments. Furthermore, integrating predictive analytics to forecast land use changes and trends could further improve the tool's applicability for strategic planning and resource

Moreover, improving the system's accuracy through continuous model training and data updates will make it more reliable for real-time applications. The integration of cloud-based platforms can also enable faster processing, storage, and easy accessibility of large geospatial datasets. These integrations will make the system more scalable, efficient, and suitable for large-scale environmental monitoring and decision-making.

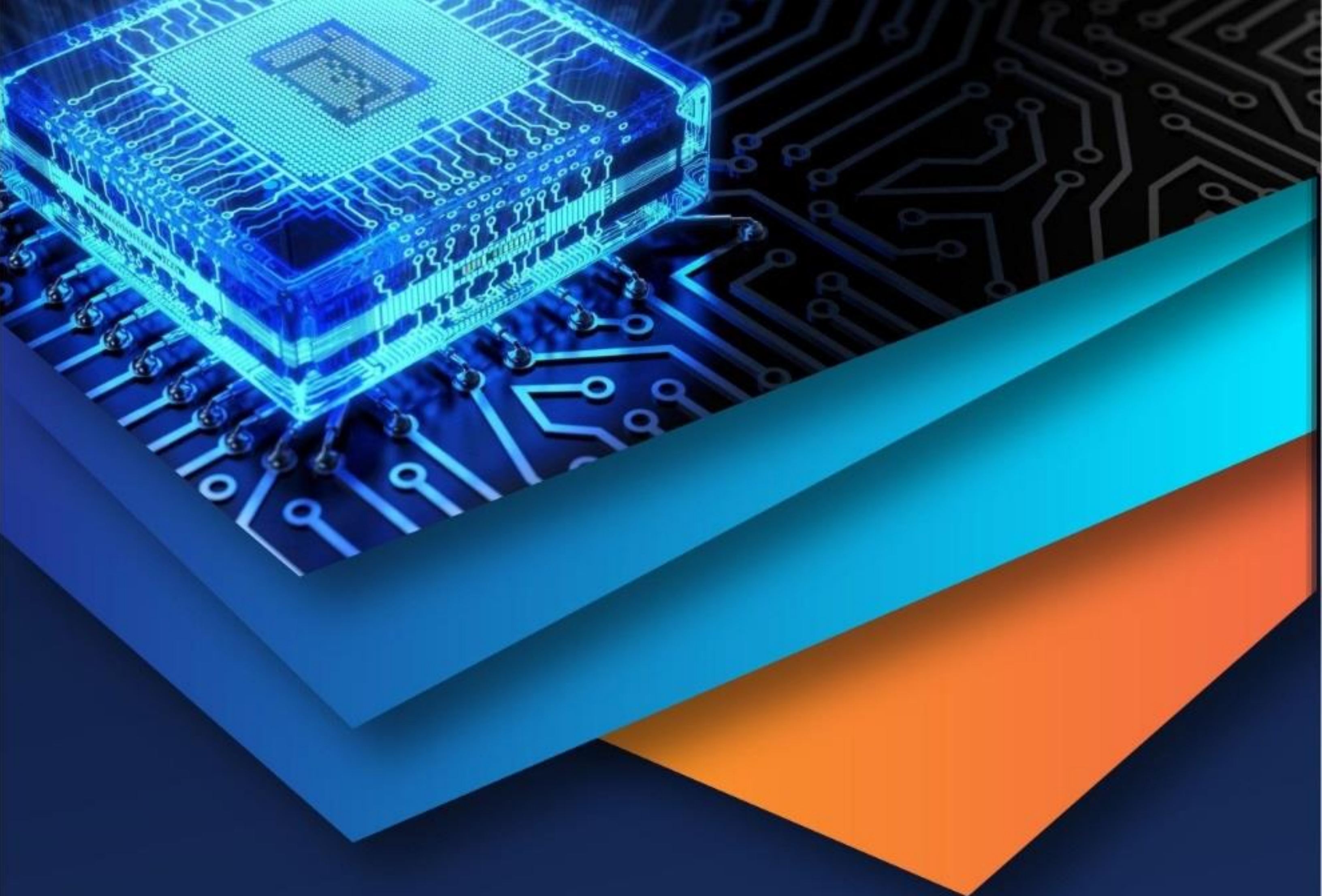
XI. CONCLUSION

The Satellite Image Segmentation System successfully transforms complex satellite imagery into clear, actionable land-use maps. By harnessing the power of YOLOv8, the system accurately identifies key features such as vegetation, water bodies, agricultural fields, bare land, and urban areas, even across diverse geographical and atmospheric conditions. With a fast and reliable backend, a clean and intuitive web interface, and full support for georeferenced inputs and outputs, users can simply upload an image and receive professional-quality results in seconds. The entire solution is designed to work both on a local computer and on the cloud, making it accessible to individuals, organizations, and governments a like.More than just a prototype, this system delivers a practical, ready-to-usetool that brings advanced artificial intelligence to real-world challenges in environmental monitoring,urban planning, agriculture, and disaster response.It empowers userswhether experts or beginnersto understand and manage our changing planet

REFERENCES

[1] M. Barthakur and K.K. Sarma, Deep Learning Based Semantic Segmentation Applied to Satellite Image, In book: Data Visualization and Knowledge Engineering, pp.-79-108, Springer, 2019.
[2] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, pp. 1097-1105, 2012.
[3] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, arXiv:1409.15, 2014.
[4] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition pp. 770-778, 2016.
[5] R. Girshick, J. Donahue, T. Darrell, J. Malik Rich feature hierarchies for accurate object detection and semantic segmentation,In: CVPR,2014.
[6] G. Cheng, Z. Li, J. Han, X. Yao and L. Guo, Exploring Hierarchical Convolutional Features for Hyperspectral Image Classification, in IEEE Transactions on Geoscience and Remote Sensing, vol. 56, no. 11, pp. 6712-6722, Nov. 2018.
[7] P. Zhou, J. Han, G. Cheng and B. Zhang, Learning Compact and Discriminative Stacked Autoencoder for Hyperspectral Image Classification, in IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 7, pp. 4823-4833, July 2019.
[8] M. Wurma, T. Starkb, X. X. Zhubc,M. Weigandad, H. Taubenbocka, Semantic segmentation of slums in satellite images using transfer learning on fully convolutional neural networks, ISPRS Journal of Photogrammetry and Remote Sensing, Elsevier, vol. 150, pp. 59-69 April 2019,
[9] J. Long, E. Shelhamer and T. Darrell, Fully Convolutional Network for Semantic Segmentation, In CPVR, 2015.
[10] L. Zhu, Y. Chen, P. Ghamisi and J. A. Benediktsson, Generative Adversarial Networks for Hyperspectral Image Classification, IEEE Transaction on Geoscience and Remote Sensing, VOL. 56, NO. 9, pp. 5046-5062, 2018.
[11] Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. arXiv:2004.10934.
[12] Jocher, G., Chaurasia, A., & Qiu, J. (2023). YOLOv8: Next-generation vision AI. Ultralytics. Retrieved from https://docs.ultralytics.com

[13] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 3431–3440.
[14] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. Lecture Notes in Computer Science.
[15] Zhang, L., & Ma, J. (2021). Remote sensing image segmentation based on deep learning: A survey. Information Fusion, 73, 145–168.
[16] Ma, L., Liu, Y., Zhang, X., Ye, Y., Yin, G., & Johnson, B. A. (2019). Deep learning in remote sensing applications: A meta-analysis and review.
[17] Kamavisdar, P., Saluja, S., & Agrawal, S. (2013). A survey on image classification approaches and techniques. International Journal of Advanced Research in Computer and Communication Engineering
[18] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
[19] Chollet, F. (2017). Deep Learning with Python.
[20] Singh, A. (1989). Digital change detection techniques using remotely-sensed data. International Journal of Remote Sensing, 10(6), 989–1003.
[21] Tuia, D., Persello, C., & Bruzzone, L. (2016). Domain adaptation for the classification of remote sensing data: An overview of recent advances. IEEE Geoscience and Remote Sensing Magazine, 4(2), 41–57.
[22] Zhang, Z., Liu, Q., & Wang, Y. (2018). Road extraction by deep residual U-Net. IEEE Geoscience and Remote Sensing Letters, 15(5), 749–753.
[23] Audebert, N., Le Saux, B., & Lefèvre, S. (2017). Semantic segmentation of Earth observation data using multimodal and multi-scale deep networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 1–9.
[24] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556.
[25] Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. International Conference on Learning Representations (ICLR



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY