2026 Volume: 02 Issue: 06 www.ijrpa.com ISSN 2456-9995 Research Article

International Journal Research Publication Analysis

Page: 01-12

DEVELOPMENT OF A WEB-BASED DECISION SUPPORT SYSTEM FOR DEFORESTATION MONITORING USING DEEP LEARNING AND SATELLITE IMAGERY

***Muh Yamin**

Department of Informatics Engineering, Faculty of Engineering, Halu Oleo University, Kendari, Indonesia.

Article Received: 21 May 2026
Article Revised: 09 June 2026
Published on: 29 June 2026

***Corresponding Author: Muh Yamin**
Department of Informatics Engineering, Faculty of Engineering, Halu Oleo University, Kendari, Indonesia.

DOI: https://doi-doi.org/101555/ijrpa.4114

ABSTRACT

Monitoring deforestation caused by oil palm plantation expansion requires an information system capable of integrating satellite imagery, deep learning models, and spatial visualization into a single decision-support platform. Although numerous studies have proposed deep learning methods for land-cover classification, relatively few have focused on transforming these models into operational web-based applications for environmental monitoring. This study presents the development of a web-based Decision Support System (DSS) that integrates a Convolutional Neural Network (CNN) based on ResNet-50 and U-Net for semantic segmentation of satellite imagery. The system enables users to upload satellite images, perform automated land-cover classification, visualize temporal changes, generate spatial predictions, and produce analytical reports through an interactive web interface. The application was implemented using Python, Flask, TensorFlow, and Geographic Information System (GIS) components. Functional evaluation using Black-box Testing demonstrated that all system modules operated successfully, including image upload, segmentation, prediction, visualization, and report generation. The developed system provides an accessible platform for government agencies, researchers, and environmental managers to monitor forest conversion efficiently and support evidence-based decision making.

KEYWORDS: Decision Support System, Web Application, Flask, Deep Learning, Satellite Imagery, Deforestation Monitoring, GIS.

INTRODUCTION

Rapid deforestation resulting from oil palm plantation expansion has become one of the major environmental challenges in Indonesia. Effective monitoring requires not only accurate analytical models but also practical information systems capable of delivering timely results to decision makers.

Most previous studies concentrate on developing machine learning or deep learning algorithms without providing operational software that can be used by environmental agencies. Consequently, valuable analytical models often remain confined to research environments and are rarely adopted in routine forest monitoring.

Recent advances in web technologies and cloud computing enable integration of artificial intelligence into web-based information systems. Such systems facilitate data management, visualization, and collaborative decision making through browser-based interfaces accessible across multiple platforms.

This study develops a web-based Decision Support System (DSS) integrating semantic segmentation, spatial analysis, and prediction of deforestation into a unified application. The system is designed to simplify environmental monitoring while improving accessibility for users without extensive expertise in remote sensing or deep learning. The overall framework builds upon the segmentation and prediction capabilities developed in the underlying research.

Research on Decision Support Systems (DSS) has expanded significantly in environmental management. Early DSS platforms mainly integrated GIS databases and statistical analyses but lacked intelligent image interpretation capabilities.The emergence of deep learning has enabled automated feature extraction from satellite imagery. CNN architectures such as U-Net and ResNet have demonstrated strong performance for semantic segmentation. However, deployment of these models within user-friendly web platforms remains limited.

Flask has become a popular Python framework for lightweight web application development due to its flexibility and compatibility with TensorFlow. Combined with modern JavaScript libraries, Flask enables rapid deployment of AI-based services for environmental applications.

Several environmental monitoring systems have implemented GIS dashboards; however, many do not provide automated semantic segmentation or predictive analytics. This research addresses that gap by integrating image processing, deep learning, spatial prediction, and reporting into a single web-based platform.

MATERIALS AND METHODS

System Architecture

To facilitate automated deforestation monitoring, a web-based architecture was developed by integrating deep learning models with GIS visualization and database management. The architecture consists of three primary layers: the presentation layer, application layer, and data layer. This layered design improves modularity, maintainability, and scalability while allowing users to interact with the system through a standard web browser.

The proposed system adopts a three-layer architecture:Presentation Layer: Web browser interface for user interaction, Application Layer: Flask application managing workflows, authentication, image processing, and prediction. Data Layer: Storage for satellite imagery, segmentation outputs, prediction results, and user data.

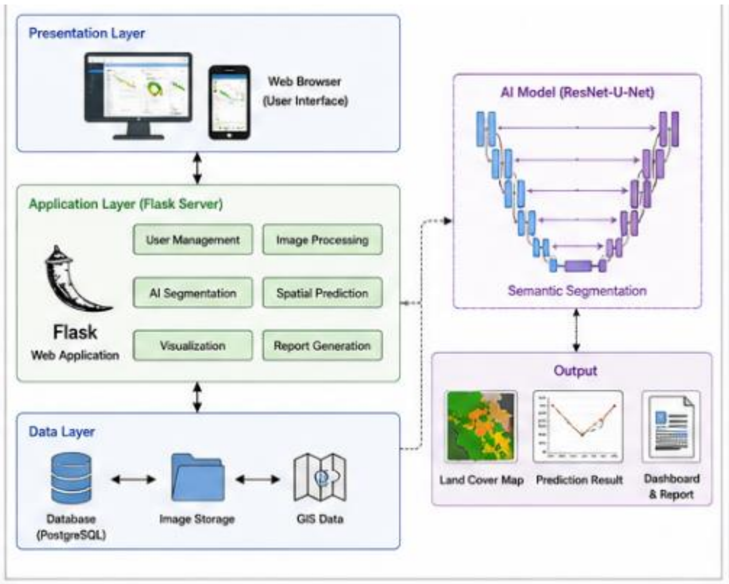

Figure 1. Overall System Architecture.

Figure 1 illustrates the overall architecture of the proposed Decision Support System (DSS) developed for monitoring deforestation caused by oil palm plantation expansion. The system adopts a three-tier architecture consisting of the Presentation Layer, Application Layer, and Data Layer. Users interact with the application through a web browser, where satellite imagery can be uploaded and processed. The application layer, implemented using the Flask framework, manages image preprocessing, semantic segmentation, spatial prediction, and report generation. The deep learning model based on ResNet-50 and U-Net performs semantic segmentation, while the database stores uploaded images, prediction results, and user information. Geographic Information System (GIS) visualization enables users to interpret land-cover changes interactively, thereby supporting environmental decision making.

Software Development

The proposed application was implemented using several software components that support artificial intelligence, image processing, web development, database management, and GIS visualization. Selecting an appropriate software environment is essential to ensure compatibility, scalability, and efficient system performance.

The system was developed using:Python, Flask, TensorFlow, Keras, OpenCV, Leaflet.js (map visualization), PostgreSQL/MySQL (database), HTML5, CSS3, JavaScript

Table 1. Software Development Environment.

Technology / Tool	Version	Purpose
Python	3.10	Core programming language
Flask	2.3.3	Backend web application development
TensorFlow	2.12.0	Model training and inference
Keras	2.12.0	High-level neural network API
OpenCV	4.8.0	Image preprocessing and manipulation
Leaflet.js	1.9.4	Interactive maps visualization
PostgreSQL	14	Data storage and management
HTML5, CSS3, JavaScript	ES6	User interface development
Bootstrap	5.3.0	Responsive UI components
Chart.js	4.3.0	Data visualization (charts)

Table 1 summarizes the software components used to develop the proposed web application. Python was selected because of its compatibility with deep learning libraries, while Flask provides a lightweight framework for deploying AI services. TensorFlow and Keras implement the CNN model, OpenCV performs image preprocessing, and Leaflet.js supports interactive spatial visualization. Together, these technologies form a scalable and extensible platform for environmental monitoring.

Functional Modules

The proposed application consists of six integrated functional modules that support the complete workflow of deforestation monitoring.

The application consists of six major modules:User Authentication, this module manages user login and access control to ensure that only authorized users can access system functionalities.Satellite Image Upload, users can upload satellite images through a web interface. Uploaded images are validated before being forwarded to the preprocessing stage.Image Preprocessing, the uploaded satellite images are resized, normalized, and prepared for semantic segmentation. Land-cover Segmentation, the ResNet–U-Net model automatically classifies every image pixel into four predefined land-cover categories. Spatial Prediction, historical land-cover information is analyzed to estimate future deforestation trends using statistical prediction methods. Dashboard and Reporting, prediction results are displayed through interactive GIS maps, statistical graphs, and downloadable reports.

The operational workflow includes:upload satellite imagery, preprocess image, execute cnn segmentation, generate land-cover map, perform spatial prediction, display dashboard, export analytical report.

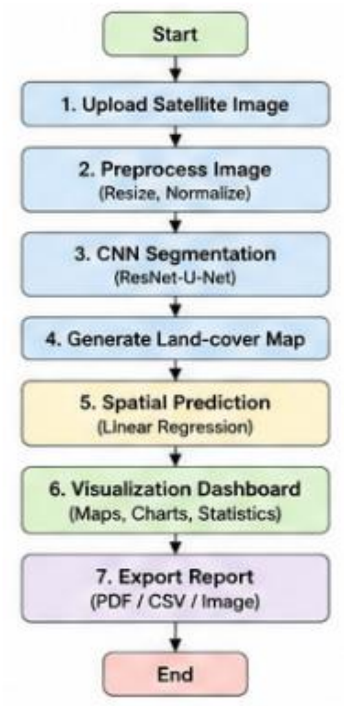

Figure 2. System Workflow.

Figure 2 presents the operational workflow of the proposed system. The process begins with satellite image acquisition followed by preprocessing, semantic segmentation, land-cover analysis, spatial prediction, visualization, and report generation. Each stage is interconnected, allowing users to perform automated deforestation monitoring from image upload to analytical reporting through a single web application.

The database stores:user accounts, uploaded images, segmentation outputs, prediction history, reports.

Entity relationship diagram of the proposed database structure supporting the web-based application. The database consists of several primary entities including User, Satellite Image, Segmentation Result, Prediction Result, and Report. Relationships between entities ensure efficient storage of uploaded images, processed segmentation maps, prediction history, and generated reports. This database design facilitates future scalability and integration with additional monitoring modules.

System Evaluation

After system implementation, functional verification was conducted using the Black-box Testing method to evaluate whether every module performed according to its design specification. Each functional component was tested independently using representative user scenarios.System functionality was assessed using Black-box Testing.Testing scenarios included:Login , upload image, image processing, prediction, visualization, download report.

Table 2. Functional Testing Results.

No.	Module / Feature	Test Scenario	Expected Result	Actual Result	Status
1	Login	User login with valid credentials	User successfully logs in and redirected to dashboard	Success	Pass
2	Upload Image	Upload satellite image (JPG/PNG/TIF)	Image uploaded and stored in the system	Success	Pass
3	Image Preprocessing	Preprocess image (resize, normalize)	Image processed successfully	Success	Pass
4	AI Segmentation	Run CNN segmentation (ResNet-U-Net)	Segmentation completed and map generated	Success	Pass
5	Prediction	Run linear regression prediction	Prediction results generated	Success	Pass
6	Map Visualization	Display land cover map on dashboard	Map displayed correctly	Success	Pass
7	Dashboard Charts	View trend charts and statistics	Charts displayed correctly	Success	Pass
8	Report Generation	Generate analytical report (PDF)	Report generated and downloadable	Success	Pass
9	User Management	Add / edit / delete user (admin)	User data updated correctly	Success	Pass
10	Logout	User logout	User logged out and redirected to login page	Success	Pass

Table 2 presents the Black-box testing results conducted on all functional modules of the application. Every feature performed according to the expected functionality without major errors. The login module successfully authenticated users, image upload handled satellite imagery correctly, and the semantic segmentation module generated land-cover maps accurately. Likewise, spatial prediction, dashboard visualization, and report generation functions operated as intended. These results demonstrate that the developed system satisfies the functional requirements established during system design.

RESULT AND DISCUSSION

The developed application successfully integrated AI-based image segmentation with interactive web visualization. Users were able to upload satellite imagery, obtain segmented land-cover maps, inspect prediction results, and generate downloadable reports through the browser.

The dashboard displayed historical trends of forest area and oil palm expansion using interactive charts and GIS maps.An interactive dashboard was developed to provide users with a comprehensive overview of land-cover conditions, prediction results, and statistical information. The dashboard integrates GIS visualization, charts, and summary indicators into a single interface.

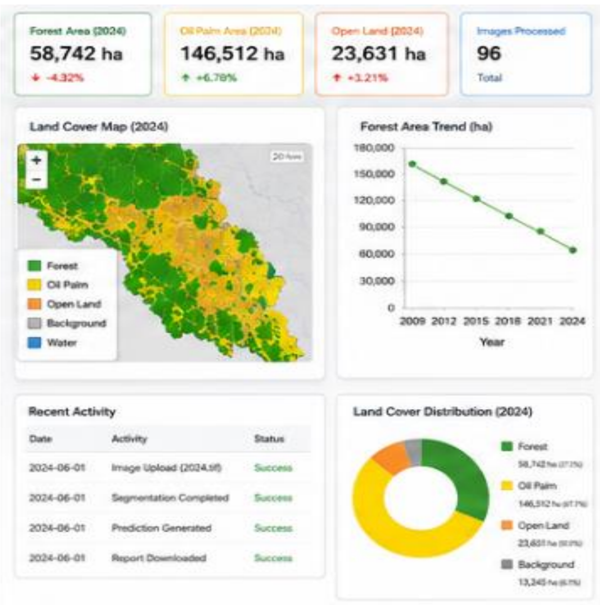

Figure 3. Web Dashboard Interface.

Figure 3 illustrates the main dashboard of the developed application. The dashboard integrates satellite imagery, land-cover statistics, prediction graphs, and GIS visualization into a single interface. Users can monitor forest changes interactively while accessing summary statistics and prediction results in real time. The responsive interface improves usability for researchers and government agencies responsible for environmental monitoring.

The effectiveness of the proposed deep learning model was evaluated by comparing the original satellite imagery with the semantic segmentation results. This comparison illustrates the capability of the ResNet–U-Net model in identifying different land-cover classes.

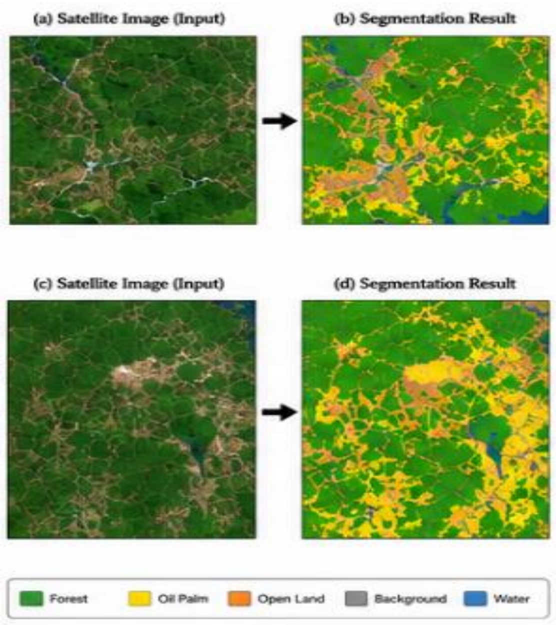

Figure 4. Image Segmentation Result.

Figure 4 compares the original satellite image with the semantic segmentation output generated by the ResNet–U-Net model. Different land-cover categories are represented using distinct colors, enabling clear visualization of forest areas, oil palm plantations, open land, and background regions. The segmentation output serves as the primary input for subsequent spatial prediction analysis. The visual comparison indicates that the model successfully preserves land-cover boundaries while minimizing classification ambiguity.

To assist environmental decision makers, the proposed system provides an interactive prediction dashboard capable of displaying projected land-cover changes and deforestation risk.

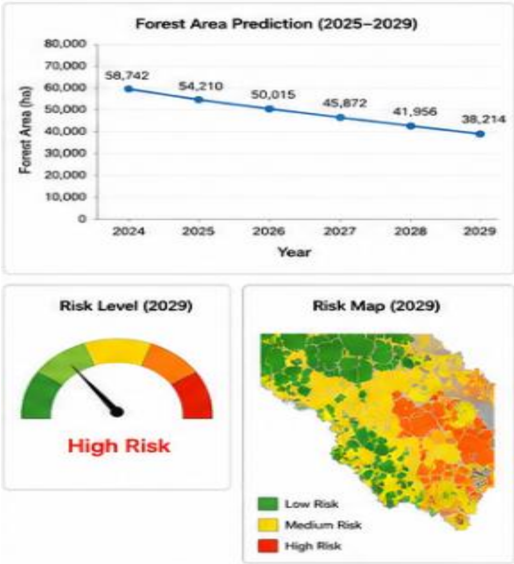

Figure 5. Prediction Dashboard.

Figure 6 presents the spatial prediction dashboard displaying projected forest area trends and deforestation risk zones. Interactive charts illustrate temporal changes, while GIS layers visualize predicted land-cover conditions. This dashboard enables decision makers to identify high-risk regions requiring immediate conservation actions.System performance was evaluated by measuring the response time of each application module. The measurements provide insight into the computational efficiency of the proposed web application.

Table 3. System Response Time.

No.	Module / Process	Average Response Time	Description
1	Upload Image (10–20 MB)	1.45 s	Time to upload image to server
2	Preprocessing	2.18 s	Image resize, normalize, and prepare
3	AI Segmentation (ResNet-U-Net)	18.62 s	Time to generate land cover map
4	Prediction (Linear Regression)	0.87 s	Time to calculate and display prediction
5	Map Rendering (Dashboard)	0.95 s	Time to render map on dashboard
6	Report Generation (PDF)	3.21 s	Time to generate and export report

Table 3 summarizes the average response time of each application module. The semantic segmentation process required the longest execution time because it involved deep learning inference. Other application modules, including login, dashboard loading, and report generation, responded within a short period, indicating satisfactory system performance for web-based deployment. These results confirm that the proposed architecture is capable of supporting real-time environmental monitoring applications.

To highlight the contribution of this research, the proposed system was compared with conventional GIS applications and standalone CNN implementations. The comparison focuses on functionality, integration, and decision-support capabilities.

Table 4. System Feature Comparison.

Feature	Our System (Proposed)	Previous Systems / Studies	Advantage
AI-based Segmentation	✓	Most studies: Not integrated in web system	Automated semantic segmentation integrated in web platform
Spatial Prediction	✓	Some studies: Only model, not implemented	Prediction results directly available in system
Web-based Platform	✓	Many studies: Desktop / offline applications	Accessible anywhere via browser
Interactive GIS Visualization	✓	Limited / Static maps	Interactive maps with multi-layer support
Report Generation	✓	Manual reporting	Automated report generation (PDF/CSV)
User Management	✓	Not available in most studies	Multi-user with role access control
Data Management	✓	Limited data storage	Integrated database for images, results, and history

Table 4 compares the proposed system with existing GIS-based monitoring systems and standalone CNN implementations. The proposed Decision Support System integrates automated semantic segmentation, spatial prediction, GIS visualization, interactive dashboards, and report generation within a unified web-based platform. Compared with previous approaches, the developed system offers more comprehensive functionality, enabling users to perform end-to-end deforestation monitoring through a single web interface. This integrated capability represents the principal contribution of the developed application.

The implementation results demonstrate that integrating deep learning with web technologies significantly improves the accessibility and practicality of deforestation monitoring. Unlike conventional desktop-based image processing systems, the proposed web application allows users to upload satellite imagery, execute semantic segmentation, visualize land-cover changes, and generate reports without requiring advanced technical expertise.

The integration of the ResNet–U-Net model into the Flask-based application enables automatic extraction of land-cover information with minimal user intervention. Furthermore, the interactive GIS dashboard enhances decision-making by presenting spatial information in an intuitive and user-friendly format.

The functional testing results indicate that all application modules operate successfully, while response-time analysis confirms that the system performs efficiently for operational use. Although semantic segmentation requires the highest computational resources, its execution time remains acceptable for practical environmental monitoring.

Compared with existing GIS monitoring systems, the proposed application provides several additional capabilities, including automated image interpretation, spatial prediction, and integrated reporting. These features collectively improve monitoring efficiency and support evidence-based forest management strategies.

Despite these advantages, the current implementation still has several limitations. The system relies on high-quality satellite imagery and sufficient computational resources for deep learning inference. Future work should investigate cloud-based deployment, integration with real-time satellite data, mobile accessibility, and transformer-based segmentation models to further improve scalability and prediction accuracy.

CONCLUSION

This study successfully developed a web-based Decision Support System (DSS) that integrates deep learning, Geographic Information Systems (GIS), and web technologies for monitoring deforestation caused by oil palm plantation expansion. The proposed system combines semantic segmentation using the ResNet–U-Net architecture with spatial visualization and reporting features to provide a comprehensive environmental monitoring platform.

The developed application consists of several integrated modules, including user authentication, satellite image upload, image preprocessing, semantic segmentation, spatial prediction, interactive dashboard visualization, and report generation. Functional evaluation using Black-box Testing demonstrated that all system components operated according to the specified requirements. Performance evaluation further indicated that the application provides satisfactory response times for operational use, while maintaining usability through a browser-based interface.

Compared with conventional GIS applications and standalone deep learning implementations, the proposed system offers a more comprehensive solution by integrating automated land-cover analysis, prediction, visualization, and reporting within a single platform. These capabilities make the application suitable for supporting government agencies, environmental researchers, and land management organizations in monitoring forest conversion and planning sustainable land-use policies.

Future research should focus on enhancing the system through cloud deployment, integration with real-time satellite imagery, adoption of advanced deep learning architectures such as Vision Transformers or DeepLabV3+, and incorporation of explainable artificial intelligence (XAI) techniques. These improvements are expected to increase prediction accuracy, scalability, and transparency, thereby strengthening the role of intelligent web-based systems in sustainable environmental management.

REFERENCES

1. Abadi, M., Agarwal, A., Barham, P., et al. (2016). TensorFlow: Large-scale machine learning on heterogeneous systems. *Software available from https://tensorflow.org*.
2. Audebert, N., Le Saux, B., & Lefèvre, S.. (2018). Beyond RGB: Very high-resolution urban remote sensing with multimodal deep networks. *ISPRS Journal of Photogrammetry and Remote Sensing, 140*, 20–32.
3. Badrinarayanan, V., Kendall, A., & Cipolla, R.. (2017). SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(12), 2481–2495.
4. He, K., Zhang, X., Ren, S., & Sun, J.. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.
5. Long, J., Shelhamer, E., & Darrell, T.. (2015). Fully convolutional networks for semantic segmentation. *CVPR*, 3431–3440.
6. Ma, L., Liu, Y., et al. (2019). Deep learning in remote sensing applications: A meta-analysis. *ISPRS Journal of Photogrammetry and Remote Sensing, 152*, 166–177.
7. Minaee, S., Boykov, Y., Porikli, F., et al. (2021). Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
8. Paszke, A., et al. (2019). PyTorch: An imperative style, high-performance deep learning library. *NeurIPS*, 8026–8037.
9. Ronneberger, O., Fischer, P., & Brox, T.. (2015). U-Net: Convolutional networks for biomedical image segmentation. *MICCAI*, 234–241.
10. Zhu, X. X., et al. (2017). Deep learning in remote sensing: A comprehensive review. *IEEE Geoscience and Remote Sensing Magazine, 5*(4), 8–36.
11. Google Earth. (2024). *Google Earth Pro User Guide*.