PATCH-ORIENTED SURFACE TEXTURE IDENTIFICATION IN REMOTE SENSING IMAGES LEVERAGING VISION TRANSFORMERS

E. Saivarshith[1], A. Nagarani[2], Rekha Gangula[3*], Chitte Ganesh[1], Kusuma Vishnu Vardhan[1], C Shashidhar[1]

[1]UG Student, [2]Assistant Professor, [3]Associate Professor and Head, [1,2,3]Department of Computer Science and Engineering (AI&ML)

[1,2,3]Vaagdevi Engineering College, Bollikunta, Warangal, 506005, Telangana, India

*Correspondence: Rekha Gangula (gangularekha@gmail.com)

To Cite this Article
E. Saivarshith, A. Nagarani, Rekha Gangula, Chitte Ganesh, Kusuma Vishnu Vardhan, C Shashidhar, "PATCH-ORIENTED SURFACE TEXTURE IDENTIFICATION IN REMOTE SENSING IMAGES LEVERAGING VISION TRANSFORMERS", Journal of Science Engineering Technology and Management Science, Vol. 03, Issue 04, April 2026,pp: 367-376, DOI: http://doi.org/10.64771/jsetms.2026.v03.i04.pp367-376
Submitted: 28-02-2026 Accepted: 01-04-2026 Published: 09-04-2026

ABSTRACT

Texture classification has become a significant area of research in computer vision–based industrial inspection, where accurate surface analysis is essential for maintaining product quality and detecting manufacturing defects. In earlier practices, inspection was carried out manually by human experts who visually identified defects such as cracks, holes, and foreign objects. However, these methods were time-consuming, inconsistent, and prone to human error. Later, traditional automated approaches based on classical image processing techniques were introduced, but their reliance on handcrafted features and simple classifiers limited their ability to capture complex texture patterns. With the rapid growth of industrial datasets in terms of size and diversity, there is an increasing need for intelligent systems capable of extracting meaningful features and performing accurate multi-class classification. To address these challenges, this study proposes a transfer learning–based texture classification framework that combines deep feature extraction with machine learning techniques. A pretrained Visual Geometry Group 19 (VGG19) network is employed to extract high-level visual features from texture images. These features are then used to train multiple classifiers, including Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), Support Vector Classifier (SVC), and Extra Trees (ET). Experimental results show that the proposed VGG19 combined with the Extra Trees classifier achieves the highest classification accuracy of 97.31%, effectively distinguishing between texture categories such as good, hole, and objects. The results demonstrate that integrating deep learning–based feature extraction with ensemble machine learning methods significantly improves the reliability and efficiency of industrial texture classification systems.

Key words: Texture Classification, Transfer Learning, VGG19, Deep Feature Extraction, Extra Trees Classifier, Industrial Inspection, Ensemble Learning, Multi-class Classification.

1.INTRODUCTION

Remote sensing (RS) refers to the process of acquiring information about objects or areas without direct physical contact, typically using satellites, aircraft, or unmanned aerial vehicles (UAVs). It is

widely applied in fields such as geological surveys, environmental monitoring, oil exploration, traffic management, earthquake prediction, and water resource management. With advancements in satellite sensor technology, remote-sensing imagery has achieved significant improvements in both spatial and temporal resolution, enabling more detailed observation of the Earth's surface, as illustrated in Figure 1. Satellites like MODIS provide thermal data with high temporal resolution (1 km × 1 km), but they are limited by low spatial resolution [1]. In contrast, Landsat offers finer spatial detail (100–200 m) but with comparatively low temporal resolution.

Recent generations of satellites have enhanced both spectral and spatial capabilities; for instance, IKONOS-2 delivers 4-band multispectral images with spatial resolutions ranging from 2.5 to 4 m. Unmanned aerial vehicles (UAVs) have emerged as an effective alternative for remote-sensing data acquisition and have experienced rapid growth in recent years. They are widely used in applications such as fire detection, surveillance mapping, and landslide monitoring [2]. UAVs offer several advantages over traditional satellite and aerial imaging systems. They are easier to deploy for rapid monitoring, assessment, and mapping tasks. Operating at lower altitudes than piloted aircraft, they can capture images with centimeter-level spatial resolution. Additionally, UAVs can operate whenever weather conditions permit, improving temporal resolution. However, as spatial resolution increases, the likelihood of noise and outliers in the captured images also rises.

Figure 1: Remote sensing scene classification.

Recent studies have explored the use of convolutional neural networks (CNNs) for classifying UAV-based imagery [3]. CNN models have also been applied to classify digital surface models alongside UAV images [4], while other approaches integrate CNNs with object-based image analysis (OBIA) for land cover classification using Multiview data. Furthermore, two-branch neural network architectures have been proposed to assign multiple class labels to UAV imagery [5].

2.LITERATURE SURVEY

Bazi, et al. [6] proposed a remote-sensing scene-classification method based on vision transformers. These types of networks, which are now recognized as state-of-the-art models in natural language processing, do not rely on convolution layers as in standard convolutional neural networks (CNNs). Instead, they use multihead attention mechanisms as the main building block to derive long-range contextual relation between pixels in images. In a first step, the images under analysis are divided into patches, then converted to sequence by flattening and embedding. Carrilho, et al. [7] reviewed, they summarized the most important advancements in the last 5 years and focus mostly on machine learning-based approaches. They also outline the most promising avenues of research in the future.

Wang, et al. [8] developed and found that transformers need more parameters than CNNs. Additionally, further research is also needed regarding inference speed to improve transformers' performance. It was determined that the most common application scenes for transformers in our database are urban, farmland, and water bodies. They also found that transformers are employed in the natural sciences such as agriculture and environmental protection rather than the humanities or economics. Finally, this work summarizes the analysis results of transformers in remote sensing obtained during the research process and provides a perspective on future directions of development. Wang, et al. [9] introduced the fundamental concepts of transformers and highlight the first successful Vision Transformer (ViT). Building on the ViT, they reviewed subsequent improvements and optimizations introduced for image classification tasks. They then compare the strengths and limitations of these transformer-based models against classic CNNs through experiments. Finally, they explored key challenges and potential future directions for image classification transformers.

Aleissaee, et al. [10] reviewed the remote sensing community and has also witnessed an increased exploration of vision transformers for a diverse set of tasks. Although a number of surveys have focused on transformers in computer vision in general, to the best of their knowledge they are the first to present a systematic review of recent advances based on transformers in remote sensing. Their survey covers more than 60 recent transformer-based methods for different remote sensing problems in sub-areas of remote sensing: very high-resolution (VHR), hyperspectral (HSI) and synthetic aperture radar (SAR) imagery. They concluded the survey by discussing different challenges and open issues of transformers in remote sensing. Tombe, et al. [11] presented a comprehensive review of the developments of various computer vision methods in remote sensing. There is currently an increase of remote sensing datasets with diverse scene semantics; this renders computer vision methods challenging to characterize the scene images for accurate scene classification effectively. This paper presented technology breakthroughs in deep learning and discusses their artificial intelligence open-source software implementation framework capabilities. Further, this paper discusses the open gaps/opportunities that need to be addressed by remote sensing communities.

Mao, et al. [12] presented this work comprehensively comparing the entire YOLO family, highlighting key innovations and their practical implications. They also discuss the challenges, including dataset limitations, domain generalization, and computational constraints, proposing future solutions such as synthetic data generation, federated learning, and edge AI deployment. By bridging the gap between academic advancements and industrial applications, this review is a practical guide for selecting and optimizing YOLO models for fabric inspection, paving the way for intelligent quality control systems. Zhang, et al. [13] proposed a new remote sensing scene classification method, Remote Sensing Transformer (TRS), a powerful "pure CNNs → Convolution + Transformer → pure Transformers" structure. First, they integrate self-attention into ResNet in a novel way, using our proposed Multi-Head Self-Attention layer instead of 3 × 3 spatial revolutions in the bottleneck. Then they connect multiple pure Transformer encoders to further improve the representation learning performance completely depending on attention. Finally, use a linear classifier for classification. They train our model on four public remote sensing scene datasets: UC-Merced, AID, NWPU-RESISC45, and OPTIMAL-31. The experimental results show that TRS exceeds the state-of-the-art methods and achieves higher accuracy.

Li, et al. [14] investigated unexplored ideas for remote sensing image captioning task, using a novel patch-level region-aware module with a multi-label framework. Due to an overhead perspective and a significantly larger scale in RSIs, a patch-level region-aware module is designed to filter the redundant information in the RSI scene, which benefits the Transformer-based decoder by attaining improved image perception. Technically, the trainable multi-label classifier capitalizes on semantic

features as supplementary to the region-aware features. Moreover, modeling the inner relations of inputs is essential for understanding the RSI. Zhang, et al. [15] proposed a multiple hierarchical cross-scale Transformer model that efficiently combines the Transformer model with CNNs and is specifically designed for complex remote sensing scene classification. Firstly, a feature pyramid network with attention aggregation extracts the multi-scale base features. Then, these base features are fed into the proposed multi-scale channel Transformer (MSCT) module to derive the global features with channel-wise attention. Additionally, the base features are also fed into the proposed hierarchical cross-scale Transformer (HCST) module, which can obtain multi-level cross-scale representations.

3. PROPOSED SYSTEM

The proposed methodology follows a structured framework for developing an intelligent image analysis system capable of performing multi-class texture classification in industrial environments, as illustrated in figure 2. The architecture is divided into three functional domains: secure user interfaces, a deep feature extraction pipeline, and a distributed prediction server. This workflow ensures efficient data handling, rigorous model evaluation, and real-time prediction for automated texture analysis.

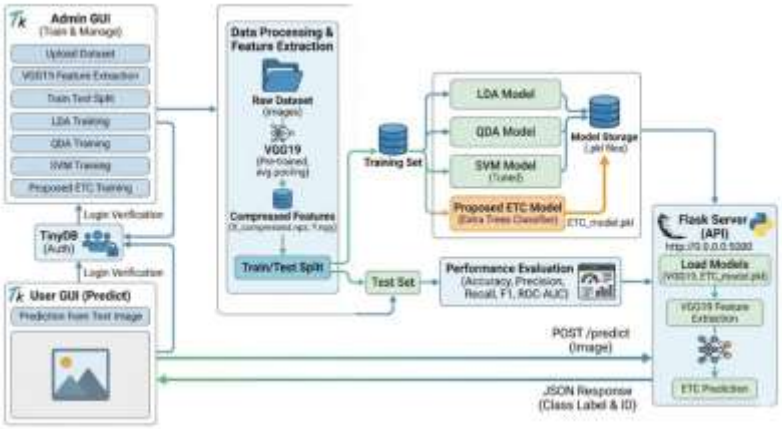


Figure 2: System architecture.

User Interface and Authentication (Tkinter GUI & TinyDB)

- **Secure Access Control:** The system integrates a TinyDB database to manage user authentication. It uses SHA-256 password hashing to verify Admin and User roles, ensuring secure access to the dashboard.
- **Interactive Control Hub:** Developed using Tkinter, the Graphical User Interface (GUI) serves as the primary interaction point. The Admin GUI facilitates critical workflow stages such as dataset uploading, VGG19 feature extraction, and model training. The User GUI provides a dedicated portal for uploading test images for server-based prediction.
- **Visualization:** Users can visualize real-time classification results, confusion matrices, and detailed performance metrics directly within the application window.

Dataset Input and Preprocessing

- **Structured Data Acquisition:** The system utilizes a structured dataset of industrial texture images (such as the TILDA-400) categorized into classes representing surface conditions like normal surfaces (good), holes, or foreign objects.

- **Image Standardization:** To ensure uniform dimensions for the neural network, all images are resized to 224×224 resolution. They are converted into multidimensional arrays and normalized to optimize the efficiency of the subsequent feature extraction process.

Deep Feature Extraction (VGG19)

- **Transfer Learning Backbone:** A pretrained VGG19 convolutional neural network serves as the backbone for capturing high-level visual patterns.
- **Dimensionality Reduction:** The final fully connected layers are truncated, allowing the convolutional base to act as a pure feature extractor. Global Average Pooling is applied to the output to generate compact, 512-dimensional numerical feature vectors representing structural and spatial patterns.
- **Data Persistence:** Extracted features are stored as compressed NumPy files (X_compressed.npz) to allow for rapid model iteration without re-processing raw images.

Train–Test Data Splitting

- **Partitioning:** Feature vectors are partitioned into distinct training and testing subsets (typically an 80/20 split).
- **Validation:** The training set allows classifiers to learn specific texture patterns, while the testing set provides an unbiased evaluation of predictive power. Shuffled indices are saved to ensure reproducibility across different training sessions.

Classification Models (Baseline vs. Proposed)

- **Existing Baseline Models:** The system benchmarks performance using three classical classifiers:
  - **LDA:** Focuses on maximizing linear separability between classes.
  - **QDA:** Models complex, quadratic decision boundaries for non-linear data.
  - **SVM:** Utilizes an RBF kernel to identify the optimal hyperplane for texture separation.
- **Proposed Ensemble Model (ETC):** The Extra Trees Classifier is implemented as the core ensemble engine. It constructs 200 randomized decision trees and aggregates their results. By using randomized feature selection, the model effectively reduces overfitting and improves accuracy over baseline methods.

Performance Evaluation and Result Analysis

- **Metric Suite:** Both baseline and proposed models are rigorously analyzed using Accuracy, Precision, Recall, and F1-score.
- **Visual Diagnostics:** The system generates Confusion Matrices and ROC Curves to identify the most reliable classifier for specific industrial surface types.

Flask Server Deployment and Prediction

- **Distributed Architecture:** A Flask-based REST API acts as the inference engine. It loads the trained ETC_model.pkl and the VGG19 backbone to handle remote prediction requests on port 5000.

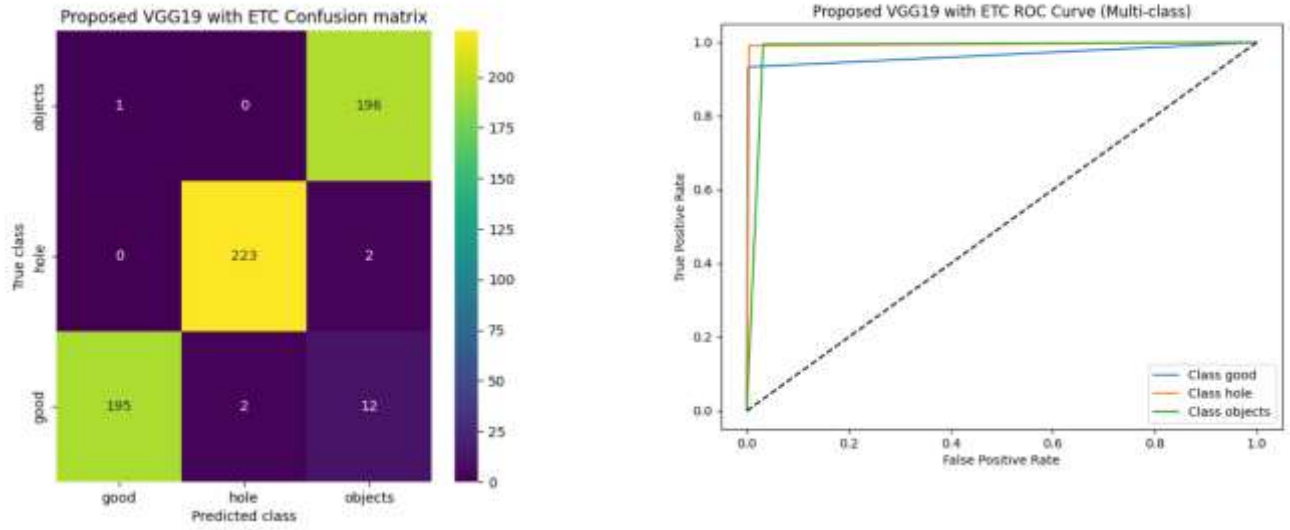
- **Real-time Prediction:** When a new image is submitted via the User GUI, it undergoes the VGG19 feature extraction pipeline. The predicted class label is sent back as a JSON response, overlaid on the image using OpenCV, and displayed in the UI for rapid identification of defects.

4. RESULTS ANALYSIS

The results of the study demonstrate the effectiveness of the proposed texture classification framework in analyzing industrial surface images. The system processes the input dataset through preprocessing, deep feature extraction, and machine learning classification stages to generate accurate predictions. Experimental results are obtained by training multiple classifiers and evaluating their performance using standard evaluation metrics. The results provide insights into how effectively the classifiers distinguish between different texture categories such as normal surfaces, holes, and foreign objects. Performance comparisons among the implemented models highlight the strengths and limitations of each classifier. These outcomes help determine the most reliable model for accurate industrial texture classification.

Figure 3 illustrates the feature extraction stage of the texture classification system where deep features are generated from the uploaded image dataset. This stage represents the completion of image preprocessing followed by the extraction of high-level visual features using the pretrained VGG19 convolutional neural network. During this process, each image is passed through the convolutional layers of the network to capture important texture patterns and structural characteristics. The extracted features are transformed into numerical feature vectors that represent the visual information contained in the images. The output displayed in this interface confirms that preprocessing and feature extraction have been successfully completed. These generated feature vectors serve as the input data for the subsequent machine learning classifiers including LDA, QDA, SVC, and ET for performing multi-class texture classification.

Figure 3: VGG19 Feature Extraction Output Display



(a) (b)

Figure 4: Performance Evaluation of Existing ET Model (a) Confusion Matrix, (b) Multi-Class ROC Curve.

Figure 4 (a) illustrates the confusion matrix obtained for the proposed classification approach that integrates VGG19 feature extraction with the ET. The confusion matrix represents the relationship between the actual texture categories and the predicted categories generated by the classifier. It provides detailed information about the correct classifications and the misclassifications among the texture classes such as good, hole, and objects. This representation highlights the ability of the proposed model to correctly distinguish between different surface texture patterns present in the dataset. The distribution of values in the matrix indicates improved classification accuracy compared to the baseline models. This evaluation helps demonstrate the effectiveness of combining deep feature extraction with an ensemble learning classifier for multi-class texture classification.

Figure 4 (b) depicts the multi-class ROC curve generated for the proposed VGG19 with ET model. The ROC curve illustrates the relationship between the true positive rate and the false positive rate for each texture class across different classification thresholds. Separate curves are plotted for each category to evaluate how effectively the proposed model discriminates between the classes. The curves approaching the upper-left region of the graph indicate strong classification capability and improved predictive performance. This graphical representation highlights the robustness of the proposed approach in identifying texture categories with higher accuracy. The ROC analysis complements the confusion matrix by providing a visual assessment of the classifier's discriminative performance across multiple classes.

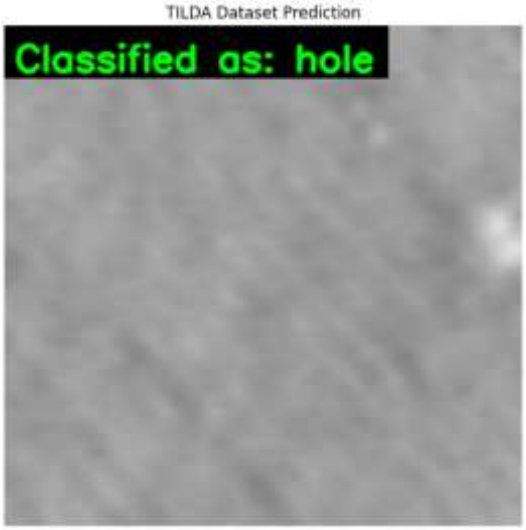


Figure 5: Prediction Result – Classified as Hole

Figure 5 illustrates the final prediction result obtained from the trained texture classification framework using a test image from the TILDA dataset. During this stage, the input image is processed through the same preprocessing and deep feature extraction pipeline using the VGG19 model. The extracted feature vector is then provided to the trained classifier, specifically the ET, which determines the most appropriate texture category for the given image. The predicted output is displayed directly on the image to clearly indicate the classification result generated by the system. This visual representation confirms the ability of the trained model to analyze unseen images and correctly identify the texture class. The prediction stage demonstrates the practical applicability of the developed framework for automated texture recognition and surface defect identification.

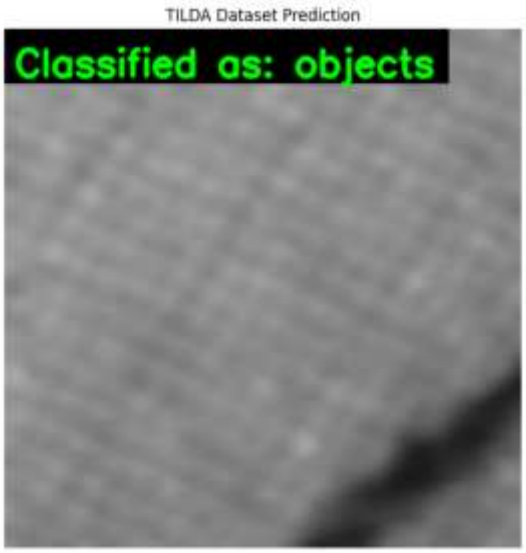


Figure 6: Prediction Result – Classified as Objects

Figure 6 illustrates another prediction result generated by the trained texture classification framework using a test image from the TILDA dataset. In this stage, the input image is processed through the preprocessing pipeline followed by deep feature extraction using the VGG19 convolutional neural network. The extracted feature vector is then provided to the trained ET, which analyses the visual patterns present in the image to determine the appropriate texture category. The predicted class label is displayed directly on the image to clearly indicate the classification outcome. This result

demonstrates the capability of the trained model to identify different texture conditions present in unseen images. The prediction output confirms the effectiveness of the integrated feature extraction and machine learning classification approach for automated texture recognition.

4.1 Comparative Analysis

The comparative analysis evaluates the performance of different machine learning classifiers used in the texture classification framework. In this study, several classifiers including LDA, QDA, SVC, and the proposed ET are implemented and compared. Each model is trained using the deep feature vectors extracted through the VGG19 transfer learning model. The comparison is performed using standard evaluation metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. This analysis helps identify the strengths and limitations of each classifier in distinguishing between different texture categories. By comparing the results, the most effective model for multi-class texture classification can be determined. The comparative study therefore provides insights into the performance improvements achieved by the proposed approach.

The comparative analysis results presented in the table 1 highlight the performance differences among the implemented machine learning classifiers. The LDA achieved an accuracy of 90.97%, demonstrating strong performance in classifying the texture categories with balanced precision, recall, and F-score values. The QDA produced a comparatively lower accuracy of 48.34%, indicating that this model struggled to effectively distinguish between the texture classes in the dataset. The SVC showed moderate performance with an accuracy of 75.44%, providing better classification capability than QDA but still lower than LDA. In contrast, the proposed ET achieved the highest accuracy of 97.31%, outperforming all other models in terms of precision, recall, and F-score.

Table 1**:** Comparative Performance Analysis of Machine Learning Classifiers for Texture Classification.

Model	Accuracy (%)	Precision (%)	Recall (%)	F-Score (%)
LDA	90.97	91.10	91.19	90.87
QDA	48.34	50.39	63.68	41.80
SVC	75.44	75.80	76.31	75.17
ET	97.31	97.31	97.30	97.24

5. Conclusion

This research presents an intelligent texture classification framework for analyzing industrial surface images using transfer learning and machine learning techniques. The system integrates VGG19-based deep feature extraction with multiple classifiers including LDA, QDA, SVC, and ET. Experimental results demonstrate that the proposed approach significantly improves classification performance compared to traditional classifiers. Among the evaluated models, LDA achieved an accuracy of 90.97%, SVC obtained 75.44%, and QDA produced 48.34% accuracy. The proposed VGG19 with ET model achieved the highest classification accuracy of 97.31%, along with improved precision, recall, and F-score values. The use of deep feature extraction combined with an ensemble classifier enables more effective identification of texture categories such as good, hole, and objects. The experimental results confirm that the proposed framework provides reliable and accurate texture classification suitable for automated industrial inspection applications.

REFERENCES

[1] Hu, Q.; Wu, W.; Xia, T.; Yu, Q.; Yang, P.; Li, Z.; Song, Q. Exploring the use of google earth imagery and object-based methods in land use/cover mapping. Remote Sens. 2013, 5, 6026–6042.

[2] Toth, C.; Jóźków, G. Remote sensing platforms and sensors: A survey. ISPRS J. Photogramm. Remote Sens. 2016, 115, 22–36.

[3] Hoogendoorn, S.P.; Van Zuylen, H.J.; Schreuder, M.; Gorte, B.; Vosselman, G. Microscopic traffic data collection by remote sensing. Transp. Res. Rec. 2003, 1855, 121–128.

[4] Valavanis, K.P. Advances in Unmanned Aerial Vehicles: State of the Art and the Road to Autonomy; Springer Science & Business Media: Berlin, Germany, 2008; ISBN 978-1-4020-6114-1.

[5] Sheppard, C.; Rahnemoonfar, M. Real-time scene understanding for UAV imagery based on deep convolutional neural networks. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 2243–2246.

[6] Purmani, S. S. R. (2025). Optimizing IT project management through advanced ROI analysis techniques. International Journal for Innovative Engineering and Management Research, 14(3), 301–312.

[7] Patel, S., & Patyrykin, K. (2025). Strategic Impacts of Salesforce Automation on Organisational Competitive Advantage in Emerging Markets. Journal of Posthumanism, 5(12), 357–372. https://doi.org/10.63332/joph.v5i12.3782

[8] Vasagam, M., Kumar, A., & Garg, A. (2026). Learning Execution Plan Embeddings for Multi-Dimensional Query Resource Prediction. IEEE Access.

[9] Kalae, U. K. (2021). Enhancing data analytics and reporting efficiency using Power BI and SQL in cloud computing environments. Journal of Computational Analysis and Applications, 29(6), 2021. https://doi.org/10.48047/jocaaa.2021.29.06.48

[10] Poojari, R. Frameworks for Data Management and Lineage in Large-Scale Healthcare Data Systems.

[11] Reddy, S. K. R. (2025). Tailoring Loyalty Rewards Systems across Industries: Cloud vs On-Prem Solutions. International Journal of All Research Education and Scientific Methods (IJARESM).

[12] Mao, M.; Hong, M. YOLO Object Detection for Real-Time Fabric Defect Inspection in the Textile Industry: A Review of YOLOv1 to YOLOv11. Sensors 2025, 25, 2270. https://doi.org/10.3390/s25072270.

[13] S. M. K. P. (2025). Cryptography in iOS: A Study of Secure Data Storage and Communication Techniques. International Journal on Science and Technology, 16(1). https://doi.org/10.71097/ijsat.v16.i1.1403

[14] Gaddam, S. From Fixed Specifications to Self-Adapting Systems: A Machine Learning Perspective on Software Engineering.

[15] Babburi, S. Privacy-Preserving Collaborative Framework with Auditable Federated Learning.