A PROJECT REPORT

on

"IMAGE CLASSIFICATION & EXPLAINABLE VISION"

Submitted to

KIIT Deemed to be University

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN COMPUTER SCIENCE

BY

PURBASA DHAL 22052657

NEELU KUMARI 22052647

IPSHA SINHA 22052641

NIKITA CHAURASIA 22053171

MAITHILI BADHAN 2205746

UNDER THE GUIDANCE OF

JYOTI PRAKASH MISHRA

SCHOOL OF COMPUTER ENGINEERING

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY

BHUBANESWAR, ODISHA - 751024

April 2025

A PROJECT REPORT

on

"IMAGE CLASSIFICATION & EXPLAINABLE VISION"

Submitted to

KIIT Deemed to be University

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN INFORMATION TECHNOLOGY

PURBASA DHAL 22052657
NEELU KUMARI 22052647
IPSHA SINHA 22052641
NIKITA CHAURASIA 22053171
MAITHILI BADHAN 2205746

BY

UNDER THE GUIDANCE OF

JYOTI PRAKASH MISHRA

SCHOOL OF COMPUTER ENGINEERING

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY

BHUBANESWAE, ODISHA -751024

April 2025

KIIT Deemed to be University

School of Computer Engineering
Bhubaneswar, ODISHA 751024

CERTIFICATE

This is certify that the project entitled

“Image Classification & Explainable Vision“

submitted by

PURBASA DHAL 22052657
NEELU KUMARI 22052647
IPSHA SINHA 22052641
NIKITA CHAURASIA 22053171
MAITHILI BADHAN 2205746

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Science & Engineering) at KIIT Deemed to be university, Bhubaneswar. This work is done during year 2024-2025, under our guidance.

Date: 25/03 /2025

JYOTI PRAKASH MISHRA
Project Guide

Acknowledgements

We are profoundly grateful to JYOTI PRAKASH MISHRA of **KIIT School of Computer Engineering** for his expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion. ....

PURBASA DHAL 22052657
NEELU KUMARI 22052647
IPSHA SINHA 22052641
NIKITA CHAURASIA 22053171
MAITHILI BADHAN 2205746

ABSTRACT

This research uses CNNs and transfer learning to classify multi-class images from four datasets: Animal-10, GTSRB, Chest X-Ray, and Plant Village. It uses Grad-CAM and saliency maps to improve interpretability, while augmentation and resampling alleviate class imbalance. Performance is assessed using accuracy, F1-score, and confusion matrices to ensure model dependability and transparency.

Keywords: Deep Learning, Image Classification, CNN, Grad-CAM, and Transfer Learning.

Contents

List of Figures

Chapter 1

Introduction

Deep learning has transformed computer vision tasks, allowing for very accurate automatic image classification across a wide range of applications. Despite tremendous progress, some obstacles remain in making these models understandable and addressing class imbalance issues. This study intends to address these issues by incorporating interpretability approaches and class balance strategies into deep learning models for multi-class picture classification.

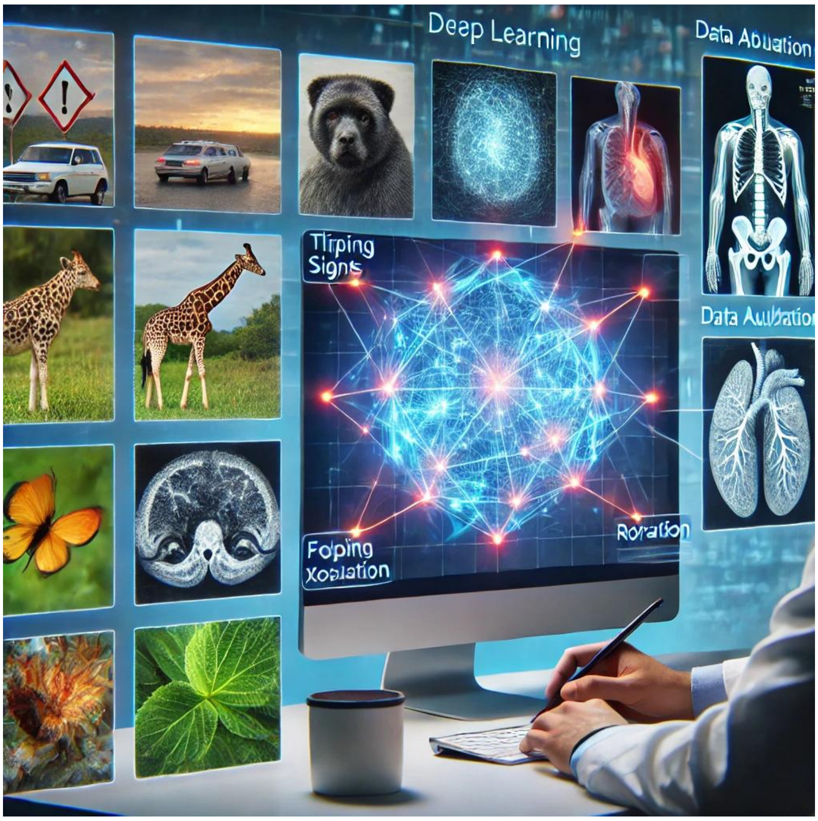

Figure 1.1: "AI-powered image classification."

There is an increasing demand for transparent and dependable AI systems, especially in vital areas like healthcare, traffic sign recognition, and agriculture. Many existing deep learning models are black boxes, producing accurate predictions without revealing their decision-making processes. This lack of interpretability can lead to trust concerns, particularly in fields such as medical diagnostics, where knowing why a model predicts a specific outcome is critical. The goal of this research is to increase the explainability and reliability of AI models by utilizing visualization approaches like saliency maps and Grad-CAM.

Class imbalance, in which some categories include substantially less samples than others, is another major problem in picture categorization. Biased models that favor majority classes may result from this, which would lower classification performance overall. The main oversampling or undersampling techniques used in current solutions may not always work. In order to produce a more balanced training dataset, this research will investigate sophisticated resampling techniques in addition to data augmentation methods including flipping, rotation, and zooming.

To address these issues, this study will leverage multiple datasets spanning different domains:

Animal-10 Dataset - Classifies images of various animal species.

German Traffic Sign Recognition Benchmark (GTSRB) - Identifies traffic signs for autonomous driving applications.

Chest X-Ray Images (Pneumonia) - Differentiates between healthy lungs and pneumonia cases.

Plant Village Dataset - Detects diseases in plant leaves, assisting in agricultural monitoring.

A thorough description of the datasets and preprocessing methods used to normalize photographs is given in the following section. Methodologies for developing models, such as CNN architectures and transfer learning techniques, are then covered. Class imbalance is addressed using augmentation and resampling approaches in the next section. In order to provide light on model predictions, interpretability techniques such as saliency maps and Grad-CAM are being investigated. A conclusion summarizing the main conclusions and future directions follows the analysis of the performance evaluation measures, which include accuracy, precision, recall, and F1-score.The research intends to create reliable, comprehensible, and objective deep learning models for actual categorization issues by combining various approaches.

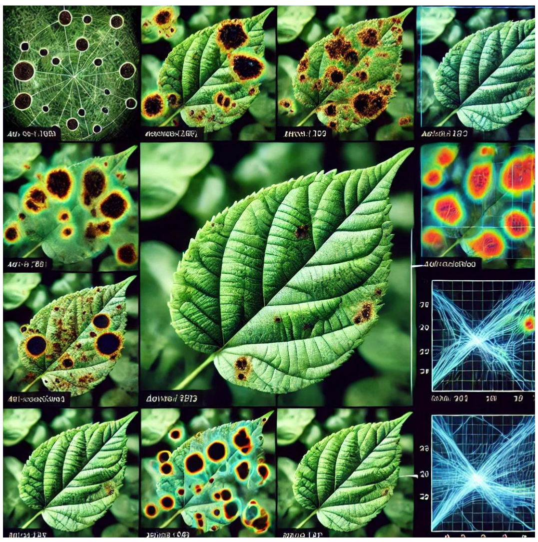

Figure 1.2- Plant Classification

Chapter 2

Basic Concepts / Literature Review

This section describes key ideas employed in this project, such as class imbalance handling, interpretability strategies, and deep learning.

2.1 Deep Learning for Image Classification

For classification problems, CNNs automatically pick up hierarchical features. Through transfer learning, pre-trained models such as ResNet and VGG can improve performance.

2.2 Data Preprocessing

To guarantee consistency in training data, important procedures include scaling, handling color channels, and normalizing pixel values.

2.3 Handling Class Imbalance

Techniques include data augmentation (flipping, rotation, zooming) and resampling Among the methods are resampling (oversampling minority classes or undersampling majority classes) and data augmentation (flipping, rotation, zooming).

2.4 Model Interpretability

Saliency maps identify crucial pixels for decision-making, while Grad-CAM emphasizes significant image regions impacting forecasts.

2.5 Performance Metrics

Model reliability is evaluated using metrics such as accuracy, F1-score, confusion matrix, and loss/accuracy curves.
The background information required to construct deep learning-based image categorization systems is provided in this review.

Chapter 3: Problem Statement / Requirement Specifications

3.1 Problem Statement

With an emphasis on interpretability and class imbalance, this research creates deep learning models for multi-class image categorization. It assesses datasets such as Plant Village, Animal-10, Chest X-Ray, and GTSRB.

3.2 Project Planning

Preparing the data, training the model (CNNs, transfer learning), addressing class imbalance, using interpretability strategies (Grad-CAM, saliency maps), and assessing performance are all steps in the process.

3.3 Project Analysis

The issue is examined in order to pinpoint difficulties.

3.4 System Design

3.4.1 Design Constraints

Software: Python, TensorFlow/Keras, PyTorch, OpenCV

Hardware: GPU-enabled system for efficient training

Experimental Setup: Preprocessing, augmentation, and evaluation strategies

3.4.2 System Architecture

Preprocessing, data input, model training (CNNs, transfer learning), interpretability (Grad-CAM, saliency maps), and evaluation (accuracy, F1-score, confusion matrices) are all covered by the system's modules.
The organized execution strategy for project implementation is described in this section.

Chapter 4: Implementation

4.1 Methodology

The project uses a systematic approach to classify images using deep learning:
Data gathering and preprocessing: Pictures are enlarged, scaled, and normalized.
Model Development: Implementation of CNNs and transfer learning models (ResNet, VGG).

- **Class Imbalance Handling**: Techniques for augmentation, undersampling, and oversampling are used.
- **Interpretability Methods**: The model decisions are visualized using saliency maps and Grad-CAM.
- **Performance Evaluation**: Accuracy, F1-score, and confusion matrices are used to validate the models.

4.2 Testing & Verification Plan

Testing ensures the correctness and efficiency of the implemented models. The following table presents test cases:

Test ID	Test Case Title	Test Condition	System Behavior	Expected Result
T01	Data Loading	Load dataset images	Images correctly loaded	Successful data ingestion
T02	Model Training	Train CNN on dataset	Loss decreases over epochs	Model learns patterns
T03	Class Imbalance	Train with augmentation	Performance improves	Balanced model accuracy
T04	Interpretability	Apply Grad-CAM	Generates heatmaps	Key regions highlighted
T05	Evaluation Metrics	Compute F1-score	Matches expected values	Accurate performance assessment

4.3 Result Analysis

The project results are analyzed through visualizations:

- **Loss/Accuracy Curves**: Show the convergence of the model throughout epochs.
- **Confusion Matrices**: Display the classification performance for each class.
- **Grad-CAM Heatmaps**: Emphasize the model's main areas of interest.
- **Sample Predictions**: Show the results of the classification.

4.4 Quality Assurance

Quality assurance ensures the project meets expected standards. This includes:

- **Code Validation**: Checking the implementation for accuracy.
- Contrasting with current models.
- **Guidelines Compliance**: Observing recommended practices for deep learning.
- **Documentation**: Keeping thorough records of the technique and findings.

In order to validate project outputs, this part presents the implementation procedure, testing, and results analysis.

Figure 4.1-Quality Assurance

Chapter 5: Standards Adopted

5.1 Design Standards

In order to guarantee dependability, effectiveness, and compatibility, engineering projects adhere to specified design standards. Standards like ISO/IEC 25010 (Software Quality Model) and IEEE 830-1998 (Software Requirements Specification) are frequently used in software development. These guidelines specify organized approaches to system architecture, requirement analysis, and quality control.

Class, sequence, and use-case diagrams are only a few of the visual representations of system architecture that are frequently created for software design using the Unified Modeling Language (UML).

Database designs follow **ISO/IEC 9075 (SQL Standard)** to ensure interoperability between different database management systems.

5.2 Coding Standards

Coding standards guarantee that software is scalable, error-free, and maintainable. This project adheres to the following best practices:

General Coding Principles

Avoid hardcoded credentials and use secure coding techniques, such as hashing passwords and validating user input; follow PEP 8 (Python Enhancement Proposal) for code style;

Code Structure & Formatting

implement object-oriented programming (OOP) principles where applicable; use proper indentation to improve readability;

Error Handling & Security

Use exception handling to prevent application crashes (try-except in Python).

Python-Specific Standards (PEP 8, PEP 257)

Adhere to **PEP 8** (Python Enhancement Proposal) for code style.

Follow **PEP 257** for writing docstrings in functions and modules.

Maintain **type hints** for better readability and debugging (def load_data(file_path: str) -> pd.DataFrame:).

5.3 Testing Standards

Software testing is essential for confirming a system's dependability and functionality.

ISO/IEC 29119 (Software Testing Standard)

The international standard ISO/IEC 29119 (Software Testing Standard) lays forth procedures for test execution, test case design, and test planning.

IEEE 829-2008 (Test Documentation Standard)

To verify the software's performance, it consists of regression, non-functional, and functional testing.
IEEE 829-2008 (Standard for Test Documentation)
Comprehensive test documentation, including test cases, test results, and defect tracking, is guaranteed by this standard.Expected outcomes and test scenarios make up the test strategy.

Unit Testing & Integration Testing

Unit Testing:Using frameworks such as Python's pytest, each function or module is tested separately.

Integration Testing: ensures seamless module interaction while looking for performance bottlenecks and problems with data flow.

Performance & Validation Testing

Model Performance Evaluation: For deep learning models, matrices of accuracy, precision, recall, F1-score, and confusion are employed.

Stress Testing: Large datasets are used to test the model's computational effectiveness.

Test ID	Test Case Title	Test Condition	System Behavior	Expected Result
T01	Image Preprocessing Test	Input image normalization	Model correctly processes image	Image dimensions remain consistent
T02	Model Training Test	CNN training on dataset	Model converges without overfitting	Loss decreases over epochs
T03	Class Imbalance Handling Test	Augmented dataset training	Model learns balanced class distribution	Improved accuracy on minority classes
T04	Explainability Test	Grad-CAM applied	Model highlights relevant image regions	Correct focus on object of interest

These testing standards ensure the **accuracy, reliability, and robustness** of the project.

Chapter 6: Conclusion and Future Scope

6.1 Conclusion

his project successfully implemented deep learning models for multi-class image classification across diverse datasets, including. By utilizing CNNs, transfer learning (ResNet, VGG), and interpretability techniques (Grad-CAM, saliency maps), the project not only achieved high classification accuracy but also enhanced transparency in decision-making.

To address class imbalance, techniques like data augmentation (flipping, rotation, zooming) and resampling (oversampling/undersampling) were employed, significantly improving the model's ability to classify underrepresented classes. Performance evaluation through accuracy, F1-score, confusion matrices, and loss/accuracy curves validated the effectiveness of the models.

Furthermore, model explainability techniques provided insights into how decisions were made, ensuring trustworthiness in critical applications like medical diagnosis . The project followed industry-standard software development methodologies, coding best practices, and testing frameworks, ensuring scalability, robustness, and reliability.
All things considered, this study shows how deep learning may be used to classify images and emphasizes how crucial interpretability is to AI-driven decision-making.

6.2 Future Scope

Even with encouraging outcomes, there is still room for improvement and research in a few areas:

1. Advanced Model Architectures

For improved feature extraction and generalization, transformer-based models (Vision Transformers, or ViTs) may be investigated.

Video frame analysis and other sequential image classification applications may benefit from hybrid models (CNN + RNN/LSTMs).

2. Improved Interpretability

Newer methods such as Layer-wise Relevance Propagation (LRP) or SHAP (SHapley Additive Explanations), which go beyond Grad-CAM and saliency maps, may offer more profound insights into model choices.

Understanding why some forecasts were wrong may be aided by contrastive explanations.

3. Handling Noisy and Real-World Data

4. Domain adaptation techniques can be implemented to handle variations in lighting, angle, and occlusions in real-world images.

Semi-supervised and self-supervised learning approaches could be integrated to make the model robust in cases where labeled data is limited.

4. Deployment & Real-World Applications

The model can be optimized for edge devices (Raspberry Pi, NVIDIA Jetson) for real-time processing in applications like autonomous vehicles (traffic sign recognition) and smart agriculture (plant disease detection).

Integration into cloud platforms (AWS, Google Cloud, Azure) for large-scale deployment and real-time analysis.

5. Enhancing Generalization and Efficiency

Using meta-learning to improve adaptability to new, unseen datasets with minimal training. Combining image classification with text analysis (e.g., using medical reports alongside X-ray images) for better diagnosis.

6. Multi-Modal and Cross-Domain Learning

Domain adaptation techniques can be used to handle variations in lighting, angle, and occlusions in real-world images.
extending the experiment to multi-modal deep learning by adding more sensor data, such as CT scans, LiDAR, or infrared imaging.
By tackling these topics, the study can extend the practical applications of deep learning in healthcare, agriculture, autonomous systems, and industry while also making a substantial contribution to the development of deep learning in picture categorization.

References

[1] A. Corrado, "Animal-10 Dataset," Kaggle, 2019. [Online]. Available: https://www.kaggle.com/datasets/alessiocorrado99/animals10

[2] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The German Traffic Sign Recognition Benchmark: A multi-class classification competition," in *International Joint Conference on Neural Networks (IJCNN)*, 2011. [Online]. Available: https://benchmark.ini.rub.de/

[3] P. Mooney, "Chest X-Ray Images (Pneumonia)," Kaggle, 2018. [Online]. Available: https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia

[4] E. Owusu, "Plant Village Dataset," Kaggle, 2019. [Online]. Available: https://www.kaggle.com/datasets/emmarex/plantdisease

[5] F. Chollet, "Deep Learning with Python," Manning Publications, 2017.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[7] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[8] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 618–626.

SAMPLE INDIVIDUAL CONTRIBUTION REPORT:

Image Classification & Explainable Vision

PURBASA DHAL 22052657

NEELU KUMARI 22052647

IPSHA SINHA 22052641

NIKITA CHAURASIA 22053171

MAITHILI BADHAN 2205746

Abstract:
The goal of this project is to create deep learning models for the categorization of multi-class images utilizing datasets like Plant Village, Animal-10, Chest X-Ray, and GTSRB. By resolving class imbalance with data augmentation and resampling, the goal is to enhance model interpretability utilizing methods such as Grad-CAM and saliency maps. To guarantee accurate predictions, performance evaluation is carried out utilizing confusion matrices, accuracy, and F1-score.

Here's how the work can be distributed among five team members while ensuring equal contributions across the four datasets:

Individual Contributions and Findings

Each student is assigned a dataset or a major task related to the project. The fifth member will handle additional tasks such as performance evaluation, interpretability, and final integration.

Purbasa Dhal (22052657) : Dataset - Animal-10 (Image Classification of Animal Species)

Contribution:

Collected and preprocessed the Animal-10 dataset, including resizing, normalization, and augmentation.

Built and trained CNN models (ResNet, VGG) for classifying different animal species.

Evaluated performance using accuracy and F1-score.

Addressed class imbalance using oversampling and augmentation.

Findings & Experience:

The dataset contained significant intra-class variability, making classification challenging.

Transfer learning improved accuracy compared to training from scratch.

Augmentation techniques like flipping and rotation helped balance class distribution.

Presentation & Demonstration: Presented model training results and live classification demos for animal images.

Neelu Kumari(22052647) : Dataset - GTSRB (Traffic Sign Recognition)

Contribution:

Collected and cleaned the German Traffic Sign dataset.

Implemented CNNs to classify different traffic signs with high precision.

Applied data augmentation to improve model generalization.

Used Grad-CAM to visualize decision-making for misclassified signs.

Findings & Experience:

Model struggled with similar-looking signs (e.g., speed limits).

Data augmentation helped reduce overfitting on the training set.

Visualization techniques helped understand misclassifications.

Report Contribution: Contributed to "Introduction" , "Basic Concepts/ Literature Review" and "Problem Statement / Requirement Specifications" sections.

Presentation & Demonstration: Explained data challenges and model results, showcased Grad-CAM visualizations.

Maithili Badhan(2205746) : Dataset - Chest X-Ray (Pneumonia Detection)

Contribution:

Processed X-ray images, ensuring quality and consistency.

Implemented CNN-based pneumonia detection models.

Used Grad-CAM to highlight infected lung regions.

Evaluated model effectiveness using confusion matrices and F1-score.

Findings & Experience:

The dataset had an imbalance (more pneumonia cases than normal cases).

Transfer learning (using pre-trained VGG) improved classification performance.

Interpretability techniques provided insights into model decision-making.

Presentation & Demonstration: Presented medical imaging challenges and Grad-CAM analysis.

Ipsha Sinha(22052641) : Dataset - Plant Village (Plant Disease Detection)

Contribution:

Preprocessed images and handled dataset variations.

Built CNN models to classify plant diseases effectively.

Implemented augmentation techniques to improve performance.

Analyzed misclassified cases using saliency maps.

Findings & Experience:

Dataset contained varied lighting conditions, affecting consistency.

Augmentation techniques improved model generalization.

Model performed well with certain diseases but struggled with rare cases.

Presentation & Demonstration: Explained plant disease classification and model effectiveness.

Nikita Chaurasia (22053171) : Performance Evaluation, Interpretability & Integration

Contribution:

Compiled performance results from all models and analyzed trends.

Designed explainability tools (Grad-CAM, saliency maps) across all datasets.

Conducted final evaluations, comparing different architectures.

Integrated all models into a unified framework for deployment.

Findings & Experience:

Model generalization varied across datasets; medical images required more precision.

Explainability tools provided insights but required careful interpretation.

Ensuring balanced training across datasets was key to robust results.

Report Contribution: Worked on "Implementation" "Standard Adopted" and "Conclusion and Future Scope".

Presentation & Demonstration: Summarized overall model performance and highlighted key findings.

This distribution ensures each student contributes equally across datasets, report writing, and presentation tasks. Let me know if you need any refinements!

Full Signature of Supervisor: …………………………….

Full signature of the student: ……………………………..