

Optimizing Image Classification Models: A Study on Computational Efficiency and Accuracy

Ramoni Tirimisiyu AMOSA^{1*}, Adekiigbe ADEBANJO¹, Oniyide Alabi BELLO², Fabiyi Aderanti ALIFAT¹, Lawal Moshood OLATUNJI¹, Olorunlomeue Adam BIODUN¹, Akanni Abideen AMOSA³, Oluwatosin Adefunke OLUWATOBI¹

¹Department of Computer Science, Federal Polytechnic, Ede, Osun State, Nigeria

²Department of Physical, Mathematical and Computer Sciences, Aletheia University, Ago Iwoye, Ogun State, Nigeria

³Department of Computer Engineering Technology, Federal Polytechnic, Ede, Osun State, Nigeria

*Corresponding Author: ramoni.tirimisiyu@federalpolyede.edu.ng

Abstract

Keywords:

Distributed, Algorithm,
Computational,
Constraints, Dataset,
Ensemble, Performance.

In computer vision, image classification remains a challenging and critical routine, widely applied in domains such as healthcare, security, and autonomous systems. This study evaluates the efficiency of various algorithms implemented for image classification using machine learning, including traditional models such as Decision Tree (DT), Random Forest (RF), k-Nearest Neighbors (k-NN), Support Vector Machines (SVM) alongside deep learning models such as Convolutional Neural Networks (CNNs) and ResNet. The research adopts an experimental approach using a dataset containing images of animals (cats, dogs, and pandas), processed and classified using different algorithms. Performance was assessed using F1-score, recall, accuracy and precision. The study reveal that deep learning models significantly outperform traditional machine learning techniques, with a pre-trained CNN achieving the highest accuracy (91.4%), followed closely by ResNet (89.6%) and Random Forest (88.2%). Among traditional classifiers, SVM and Decision Trees performed competitively with 85.6% and 85.4% accuracy, respectively, while k-NN and Naïve Bayes achieved lower results. Furthermore, ensemble methods demonstrated superior performance, with XGBoost achieving the highest accuracy (96.8%), outperforming Random Forest (95.5%) and boosting techniques like AdaBoost (92.4%). The study concludes that CNNs and ensemble methods provide the best results for complex image classification tasks, whereas traditional algorithms remain viable for smaller datasets with computational constraints. These findings serve as a guide for selecting optimal machine learning techniques for various image classification applications.

1. Introduction

Humans have used a wide variety of instruments to complete different jobs more easily since the beginning of time. Various machines were created because of the human brain's inventiveness. By allowing individuals to meet a variety of requirements, such as computing, travel, and industry, these machines made life easier for humans. In the last decade, image classification has become one of the most critical tasks in computer vision, driven by the rapid growth of machine learning (ML) algorithms and the increased access to a sizable dataset (Senokosov et al., 2024). Image classification refers to the process of assigning predefined labels to images based on their visual content (Mahesh, 2020). It has gained widespread importance across various sectors, including healthcare, agriculture, autonomous vehicles, and security. Image classification uses machine learning to learn from labeled training data. It categorizes images into predefined listings based on their visual content (mhawes Al-Naseri, 2022). It involves categorizing and assigning labels to groups of pixels or vectors

within the image based on specific rules. The goal is to examine an input image and provide a label that categorizes it. The label is always chosen from a predetermined set of categories the image could represent. The exponential growth of data generation, particularly image data, has led to significant advancements in machine learning (ML) techniques, especially in image classification tasks. Machine learning techniques, particularly those using neural networks and deep learning, have shown great promise in achieving high accuracy for image classification tasks (Luan et al, 2025). Traditional machine learning models like Support Vector Machines (SVM), Decision Trees and k-Nearest Neighbors (k-NN) have been widely used for classification problems, but more recently, deep learning models such as Convolutional Neural Networks (CNNs) have revolutionized the field (Sheth et al. 2022). The ability of these models to automatically extract features from images and learn complex patterns has made them a preferred choice for image-related tasks Smaida and Serhii (2019).

Despite the success of these models, selecting the best algorithm for a particular image classification task remains a challenge. Different algorithms have their own strengths and weaknesses, depending on the size and complexity of the dataset, the computational resources available, and the specific application domain. Therefore, it is crucial to understand the performance of different machine learning algorithms and conduct a comparative analysis to determine the most suitable approach for a given problem Plaksyvyi et al. (2023). Choosing the optimal machine learning algorithm for image classification depends on several factors, including the dataset's size, complexity, and the available computational resources. Deep learning models, particularly CNNs, generally outperform traditional machine learning models on large and complex datasets, but their resource-intensive nature can be a limiting factor. Traditional models like SVM, k-NN, and Random Forests remain competitive for smaller datasets or applications with fewer computational resources (Al-Saffar et al., 2017). Image classification uses machine learning to learn from labeled training data. It categorizes images into predefined listings based on their visual content. It involves categorizing and assigning labels to groups of pixels or vectors within the image based on specific rules. The goal is to examine an input image and provide a label that categorizes it. The label is always chosen from a predetermined set of categories the image could represent.

This study is significant as it provides valuable insights into the comparative performance of various machine learning algorithms for image classification tasks. By evaluating both traditional and deep learning algorithms across multiple datasets, the research will provide a comprehensive understanding of their suitability in different contexts. The findings will help researchers, practitioners, and organizations make informed decisions when selecting machine learning models for image classification tasks, improving accuracy and efficiency in real-world applications (Patel et al, 2020). Furthermore, this study will contribute to the growing body of literature on machine learning and image classification, providing a foundation for future research in optimizing algorithm selection for specific image-related tasks. Before the advent of machine learning, image classification was primarily carried out using manual and classical computer vision techniques. These methods focused on handcrafted feature extraction and traditional algorithms to classify images (Gao et al., 2019).

2. Literature Review

Dalavai et al. (2024) investigated the effects of various pre-processing techniques, such as noise reduction, histogram equalisation, random cropping, and rotation, on training efficiency and accuracy. The proposed study is related to the development of deep learning-based image classification with the aid of these techniques. Each experiment had a consistent dataset, and researchers used CUDA-enabled GPUs and the open-source deep learning framework PyTorch. By combining these methods, the accuracy that may be achieved with the best baseline model is doubled, as opposed to the baseline's 83% accuracy. The most important element in enhancing convergence and generalisation skills on unseen material was discovered to be normalisation. Overall, the results demonstrate the optimality that can be obtained from carefully selected pre-processing techniques when applied as forerunners to deep learning-based picture classification. Gupta et al. (2021) evaluated SVM, k-NN, and CNN on CIFAR-10 and Fashion MNIST, concluding that CNN achieved the highest accuracy (91%), while k-NN struggled with complex datasets. Similarly, Zhang & Li (2019) compared Decision Tree, Random Forest, and CNN using ImageNet and CIFAR-10, finding that CNN outperformed traditional models with 92% accuracy. Patel et al. (2020) expanded on this by analyzing SVM, Random Forest, ANN, and CNN on MNIST and CIFAR-10, reporting that ANN and CNN achieved similar performance above 90% accuracy.

Wu (2025) presented a brand-new system that uses elastic scaling to improve picture classification performance, particularly in cloud contexts. This method makes use of a modular architecture, which allows for real-time adaptation of computer resources according to the complexity and demand of the models being used. To balance accuracy and operational efficiency, the methodology tests several model designs and integrates a variety of data pretreatment techniques. Extensive studies on a variety of datasets show how effective the research is. The results demonstrate improvements in response times and resource usage, resulting in a flexible architecture that successfully accommodates varying image processing needs. This study provides a complete approach to maximise efficiency and performance in cloud-based picture categorisation workloads. In this study, Chong et al. (2024) use machine learning (ML) and deep learning (DL) techniques to categorise microalgae of various species. For segmenting microalgae, the authors used bilateral filtering, gray-scaling, adaptive thresholding, Canny edge detection and Sobel edge detection. The results show that the final combined features, along with improved image pre-processing approaches, generated high accuracy of 96.33% for the support vector machine (SVM) classifier and 96.03% for the k-nearest neighbours (k-NN) classifier. With the accuracy of 96.67% and 90.86% at probability thresholds of 50% and 80%, respectively, the Azure custom vision model outperformed the others overall.

ARES-Bench, a thorough robustness benchmark, was developed by Liu et al. (2025) for the image classification job. The authors assess how resilient 61 common deep learning models on ImageNet are to various adversarial attacks and out-of-distribution (OOD) datasets using a variety of architectures (e.g., CNNs, Transformers) and learning algorithms (e.g., normal supervised training, pre-training, adversarial training). Large-scale experiments were carried out using robustness curves as the primary evaluation criterion, and the results yielded several significant conclusions, such as: (1) both the adversarial and natural robustness of specific noise sources are inherently traded off for the same model architecture (2) shows adversarial robustness is successfully increased by adversarial training, especially when applied to Transformer designs.

; and (3) that pre-training significantly improves natural robustness by utilising larger training datasets, incorporating multi-modal data, or using self-supervised learning techniques. Liu et al. (2020) compared CNN, ResNet, MobileNet, and VGG on ImageNet and TinyImageNet. Authors found that ResNet and VGG were the most accurate, whereas MobileNet offered faster inference times, making it more suitable for real-time applications. Similarly, Wang et al. (2021) assessed CNN, ResNet, VGG, and Inception, concluding that ResNet and VGG achieved superior accuracy on large datasets, while Inception, though faster, was less accurate.

Further evaluations reinforced CNN's superiority. Ramesh et al. (2020) compared SVM, k-NN, Random Forest, and CNN on CIFAR-10 and MNIST, finding that CNN achieved the highest accuracy (95%) on CIFAR-10, while Random Forest was more effective on simpler datasets. Likewise, Sarker & Rahman (2020) assessed SVM, CNN, k-NN, and Random Forest on CIFAR-100 and Caltech-101, concluding that CNN and ANN outperformed traditional models on complex datasets, whereas traditional models performed better on smaller datasets. CNN's effectiveness in domain-specific applications was also observed. Shankar et al. (2022) analyzed SVM, Naïve Bayes, Decision Tree, and CNN on Chest X-ray and CIFAR-10 datasets, demonstrating that CNN was the most effective for medical image classification, achieving 94% accuracy, while traditional models performed poorly. Singh & Choudhary (2020) compared Random Forest, k-NN, CNN, and Decision Tree on Flowers-102 and MNIST, revealing that CNN excelled in complex datasets, whereas Random Forest was more effective in simpler datasets.

3. Materials and Methods

The study adopts an experimental approach, testing multiple machine learning algorithms on image classification tasks. By comparing these algorithms, the study aims to identify their strengths and weaknesses, providing insights into their relative performance under different evaluation metrics.

3.1 Dataset Source, Selection and Processing

Dataset Description

The dataset used in this study was organized into a folder-based structure, where each class of animal was stored in a separate directory named after the label it represents. Specifically, the dataset contained three folders:

- i. cats/ - containing images of cats
- ii. dogs/ - containing images of dogs
- iii. pandas/ - containing images of pandas

Each image was automatically labeled based on the name of the folder it resided in. This folder-to-label mapping enabled straightforward label extraction during preprocessing. The labelling method is reliable for classification tasks if the folder names are unique and representative of class semantics. An overview of the dataset structure and representative images is shown in Figure 1. Table 1 shows the structure table describing the attributes and key details of the dataset.

Dataset Quality:

Image Resolution Consistency: All images were resized to a fixed resolution of 64×64 pixels during preprocessing. This standardization ensures compatibility across different models and reduces computational complexity.

Noise Handling: Visual inspection and automated scripts were used to remove corrupted or unreadable image files. Duplicate images were also eliminated to prevent bias in training.

File Format and Structure: All files were in .jpg format, with consistent naming schemes (e.g., cat.1.jpg, dog.2.jpg, etc.). This consistency allowed for efficient parsing and transformation into ARFF format for Weka and TensorFlow-compatible arrays for CNN models.

Image Source: The images were sourced from Kaggle, a widely recognized platform for high-quality, publicly available datasets. This helps ensure ethical usage, dataset reliability, and diverse image quality for generalization.

Class Balance: The dataset contains an equal number of images for each class (1000 per class), thereby avoiding class imbalance issues during model training.

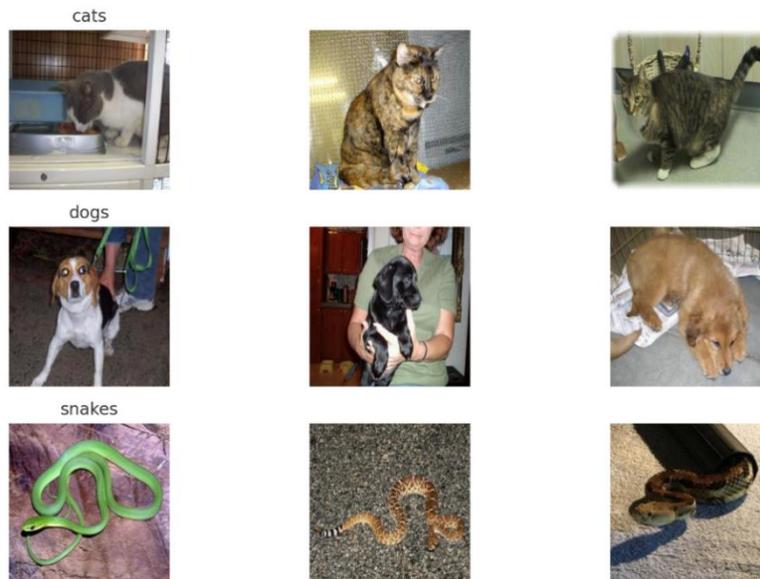


Figure 1: Image Representation of the Dataset

Table 1 shows the structured table describing the attributes and key details of the dataset



Table 1: Dataset Description

Attribute	Description
Dataset Name	Animals Image Dataset
Categories	Cats, Dogs, Pandas
Folder Structure	Each category has its own folder (cats/, dogs/, panda/) with respective images.
Total Images	3000 images in total (1000 images per category)
File Format	JPEG
Image Names	Sequential naming pattern (e.g., cat.1.jpg, dog.1.jpg, panda.1.jpg)
File Path	cats/cat.#.jpg, dogs/dog.#.jpg, panda/panda.#.jpg (where # is a sequence number)
Image Count per Category	10 images per category
Use Case	Suitable for image classification tasks focusing on animal type recognition

3.2. Data Preprocessing

To ensure consistency and enhance model performance, the following preprocessing techniques were applied:

Normalization: Pixel values were scaled to a range between 0 and 1 to standardize input across the algorithms, optimizing learning efficiency.

Reshaping: The data format was adjusted as needed for each algorithm (e.g., reshaping for convolutional layers in neural networks).

Data Augmentation: Rotation, flipping, and zooming were among the common methods used to broaden the variety of training data and reduce overfitting in deep learning models. Augmentation was performed using TensorFlow/Keras ImageDataGenerator, applying random rotations ($\pm 15^\circ$), zoom (0.8–1.2x), horizontal flips, and brightness adjustments. These augmentations increased dataset diversity and helped reduce overfitting, particularly in CNN and ResNet models.”

3.3. Experiment Method

The method adopted in the experiments is highlighted below:

Step 1: Download data from public source (Kaggle)

Step 2: Cleaning the data by removing unimportant features.

Step 3: Training the classifiers with image classification algorithm (RF, SVM, CNN, KNN, Decision Tree, Naïve Baye, ResNet and Linear Regression)

Step 4: Training the classifiers with image ensemble algorithm (Random Forest, Bagging, Boosting, Adaboost, XGBoost)

Step 5: Extracting the results generated in Step 3 and 4 and converting it into tabular format.

Weka 3.8.5 data mining was used to perform the experiment. The software includes data preparation, categorization, regression, clustering, mining of association rules, and visualization tools. This data mining tool is chosen due to the following advantages.

(i) It is simpler to fit the data and get them ready for the classifiers.
 (ii) The data is saved by Weka in a unique file format with the arff (Attribute Retention File Format) extension, which may be imported for further testing.

(iii) The software has an integrated stratified cross-validation technique for data preparation.

(iv) Weka's Explorer framework offers additional statistics regarding the classifiers' performance. After cleaning of data to ensure that only important and needed or required attribute are retain, image classification algorithms were used to construct the models and every model was built with 10-fold cross-validation to train and evaluate the performance of the models.

3.4. Research Design

Figure 2 shows the framework that was used to build the model. After the acquisition of the dataset, Image classification algorithms were applied, algorithms include RF, SVM, CNN, KNN, Decision Tree, Naïve Baye, ResNet and Linear Regression. Both conventional and contemporary methods of classification are represented by the chosen models. CNNs and ResNet provide better accuracy in high-dimensional image applications, while Decision Trees and SVMs are extensively researched for structured, interpretable models.

They are appropriate for robust comparison due to their established track record in the literature (e.g., Gupta et al., 2021; Zhang & Li, 2019).

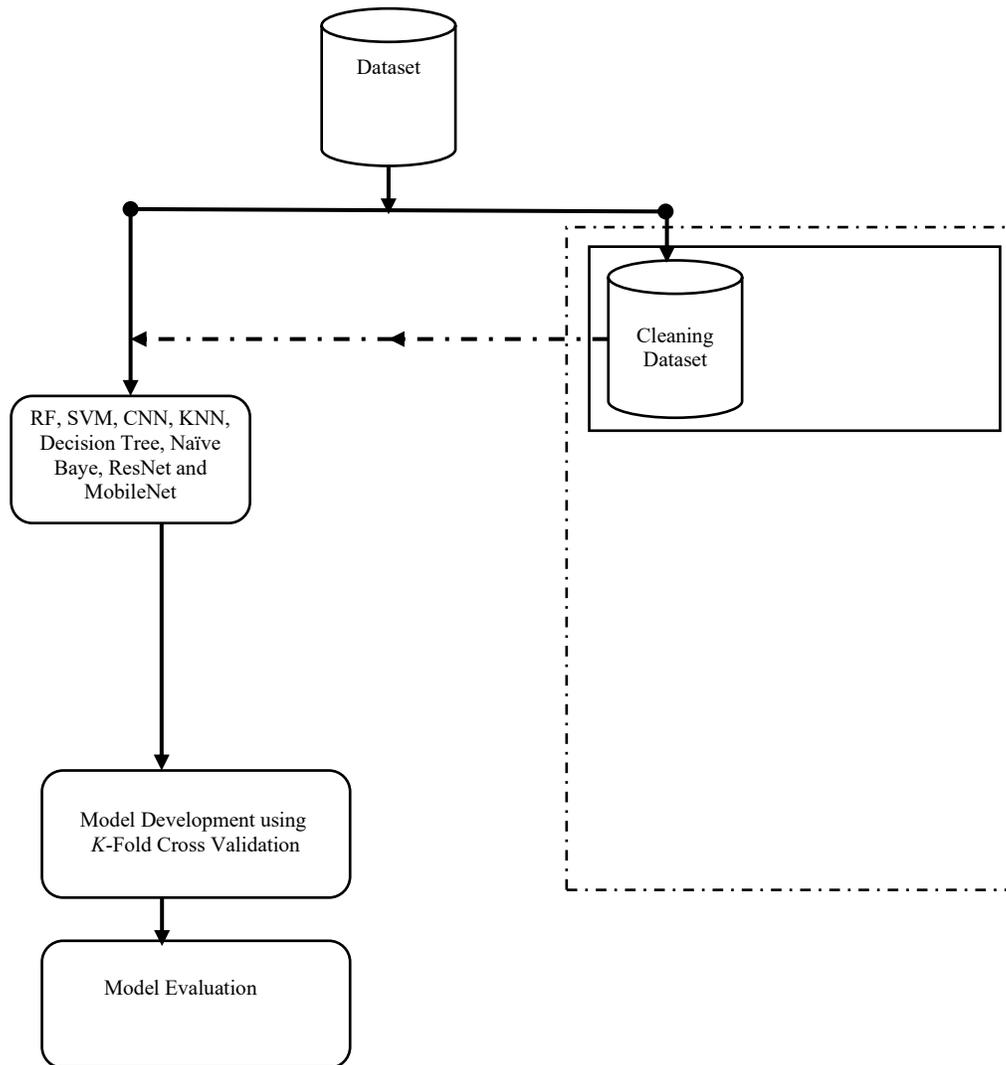


Figure 2: Research Framework

Figure 2 presents the experimental workflow, highlighting the data acquisition, preprocessing, model development, and evaluation phases.

3.5. Model Training and Evaluation

The dataset was randomly partitioned into 10 equal subsets. For each fold, 9 subsets were used for training and 1 for validation, rotating until all subsets had served as validation once. This ensured robustness and mitigated overfitting.

Training: The training dataset was used to train each algorithm, then grid search or experimentation was used to adjust the hyperparameters for best results.

Evaluation Metrics: The following metrics are used to assess model performance:

Accuracy: the proportion of all examined photos that were correctly categorised.

Recall, F1-Score and Precision: utilised to offer more information about classification performance, particularly for datasets that are unbalanced.

Confusion Matrix: used to display predictions that were right and wrong for each class.



Cross-Validation: K-fold cross-validation was used to reduce overfitting and confirm each model's resilience.

4. Results and Discussion

This section focuses on the details of the result of the research, the analysis of the result is divided into four sections, firstly the result of the baseline classifiers is presented for both the original or raw dataset and balanced dataset, secondly, the result of homogeneous ensemble (Bagging and boosting) are also analysed and the models are compared for both the balance and original dataset, thirdly, Stacked ensemble results are also presents and the performance of the model is compared with that of homogeneous ensemble and finally summary of all the results is also discussed to give general overview of the models developed.

Accurate record keeping enhances effective and reliable managerial decision as telecommunication industry is one of the industries that must ensure appropriate record and data management majorly for customer retention and profit maximization. Previous research has shown that stacking can aid in the development of improved prediction models in a variety of domains, including customer turnover (Ghalejoogh et al., 2020; Mastelini et al., 2017; Ribeiro & dos Santos Coelho, 2020). This research analyses telecommunication sector data in order to provide an effective churn-detection method. Weka (version 3.8.5) machine learning tool was used to build the models and all the experiments were carried out on 10-fold cross validation. The three categories of the results that will be analysed and compare in this chapter are baseline classifiers result, boosting/bagging and stacking.

4.1 Baseline Classifiers

Here, five major classifiers RF, BayesNet, SVM, KNN and Ripper are used to build the models. It should be noted that these classifiers are from different families. Random forest is an algorithm from tree family, BayesNet was chosen from Bayes family, SVM belong to functions, KNN is a family of Ibk algorithm and SNN is a pre-trained. Models generated by these classifiers are analysed and compare using Accuracy, Precision, Recall and F1-Score. (MCC).

Table 2: Classification Result of ML Algorithm

Algorithm	Accuracy (%)	Precision	Recall	F1-Score
SVM (SMO)	85.6	0.86	0.85	0.85
Random Forest	88.2	0.88	0.88	0.88
CNN (pre-trained)	91.4	0.91	0.91	0.91
Decision Tree (J48)	85.4	0.86	0.85	0.85
KNN (IBk)	82.1	0.82	0.82	0.82
ANN (MLP)	87.9	0.88	0.88	0.88
Naive Bayes	75.4	0.76	0.75	0.75
ResNet (features)	89.6	0.90	0.90	0.90
Linear Regression	68.2	0.68	0.68	0.68

Table 2 above presents a comparative analysis of several machine learning algorithms applied to an image classification task, with each algorithm evaluated on recall, accuracy, F1-score and precision.

- i. Accuracy (%): The percentage of correctly classified images out of the total images.
- ii. Precision: The proportion of correctly detected positive observations to all anticipated positive observations. It reflects how often the model's positive predictions are accurate.
- iii. Recall: The proportion of all observations in the actual class that were correctly labelled as positive. It shows how well every pertinent case is captured by the model.
- iv. F1-Score: The precision and recall harmonic mean, which balances these two measures.

4.2 Comparative Analysis of Algorithms

As seen in Figure 3a, the CNN model achieved the highest accuracy of 91.4%, followed by ResNet and Random Forest. Figure 3b illustrates that CNN also maintained superior F1-scores, reflecting consistent

performance across all classes, making it the most recommended choice for complex image tasks despite its computational demands. ResNet, using extracted features, followed closely with 89.6% accuracy and offered a strong balance between performance and efficiency. Random Forest and Artificial Neural Networks (ANN) also performed well, with accuracies of 88.2% and 87.9% respectively, suitable for scenarios with limited deep learning resources or sufficient training data. SVM and Decision Tree models provided moderate results, emphasizing interpretability and effectiveness in specific contexts. K-Nearest Neighbours (KNN) and Naive Bayes lagged due to limitations with scalability and feature assumptions, while Linear Regression was least suitable, highlighting its inappropriateness for classification tasks.

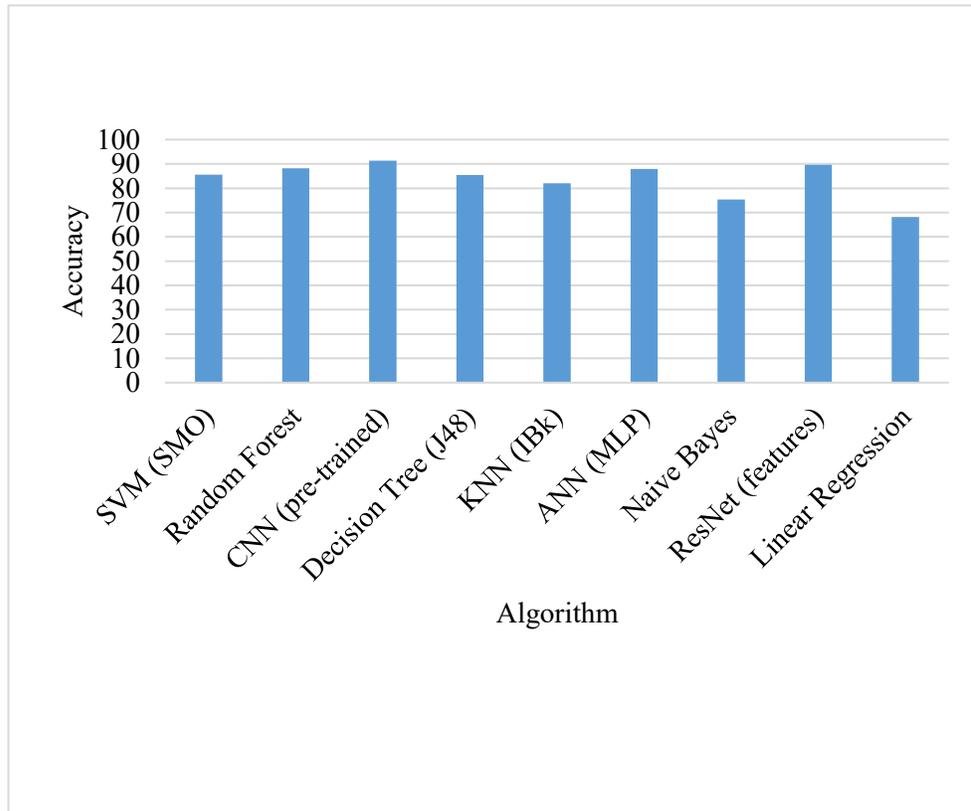


Figure 3a: Performance Accuracy of Baseline Classifiers

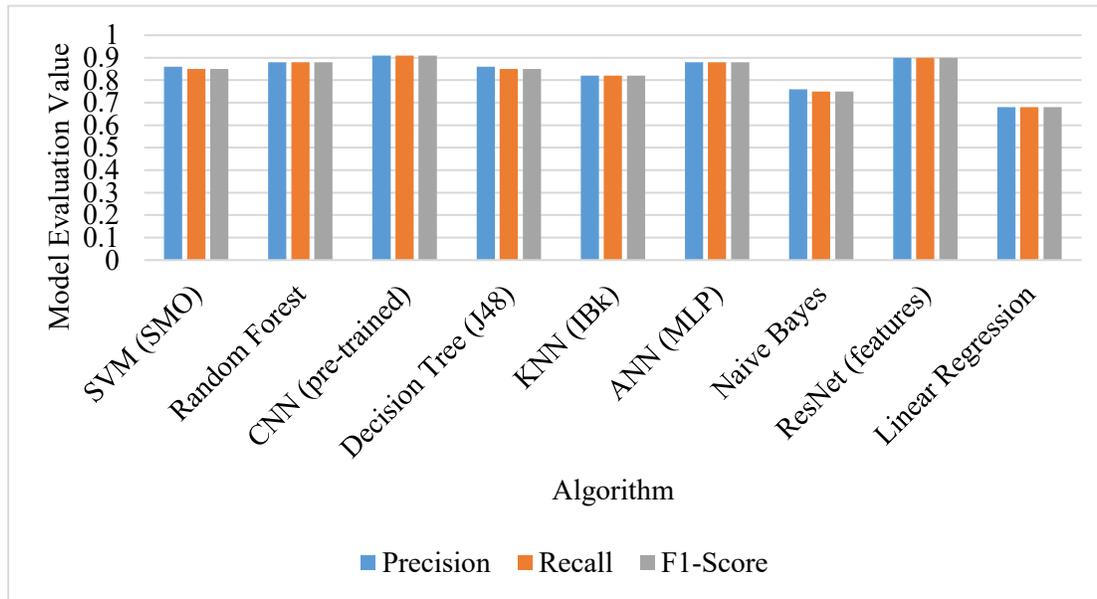


Figure 3b: Performance Evaluation of Baseline Classifiers

4.3 Ensemble Algorithm

Ensemble methods enhance model accuracy by integrating multiple models rather than relying on one. This combination significantly improves result precision. The most widely used ensemble techniques include boosting, bagging, and stacking. Ensemble methods are generally categorized into sequential and parallel techniques. Sequential ensemble techniques, such as Adaptive Boosting (AdaBoost), generate base learners in a sequence, fostering interdependence among them. This approach enhances model performance by assigning greater weights to previously misclassified learners.

Table 3: Ensemble Algorithm

Algorithm	Accuracy	Precision	Recall	F1-Score
Random Forest	95.5	0.96	0.96	0.96
Bagging	93.2	0.94	0.93	0.93
Boosting	94.8	0.95	0.95	0.95
AdaBoost	92.4	0.93	0.92	0.92
XGBoost	96.8	0.97	0.97	0.97

Table 3 shows that the accuracy XGBoost achieves the highest accuracy (96.8%), making it the best-performing algorithm for this task, random Forest follows closely at 95.5% while Bagging and AdaBoost show slightly lower accuracy as shown in figure 4a, which could indicate they are less effective at capturing complex patterns in the dataset. Precision measures the proportion of true positives among the total predicted positives for each class. XGBoost again performs best with 0.97, indicating its ability to make fewer false positive predictions.

As shown in figure 4b XGBoost maintains the highest recall at 0.97, suggesting it captures the most relevant instances. The F1-score, being the harmonic mean of precision and recall, is highest for XGBoost



(0.97), reinforcing its overall superiority in both precision and recall. It is observed that XGBoost outperforms other ensembles in all metrics. Boosting and AdaBoost perform well but are slightly outperformed by XGBoost. It is hereby concluded that for this dataset, XGBoost is the most effective algorithm, providing the best balance of precision, recall, and overall accuracy. If computational resources are limited, Random Forest is a strong alternative due to its simplicity and competitive performance.

Table 4: Hyperparameter Settings for the models

Model	Hyperparameter	Tested Range	Optimal Value
SVM (SMO)	C (Regularization)	0.01, 0.1, 1, 10, 100	10
	Kernel Type	Linear, RBF, Polynomial	RBF
k-NN	Number of Neighbors (k)	3, 5, 7, 9, 11	5
Decision Tree	Max Depth	3, 5, 10, 15, None	None
	Min Samples Split	2, 5, 10	2
Random Forest	Number of Trees	50, 100, 150, 200	100
	Max Features	sqrt, log2, None	sqrt
CNN (pre-trained)	Epochs	10, 20, 30, 50	30
	Batch Size	16, 32, 64	32
	Learning Rate	0.001, 0.01, 0.1	0.001
ResNet (Features)	Layer Depth	18, 34, 50	34
	Learning Rate	0.0001, 0.001, 0.01	0.001
AdaBoost	Number of Estimators	50, 100, 150, 200	100
XGBoost	Learning Rate	0.01, 0.05, 0.1, 0.3	0.1
	Max Depth	3, 6, 9	6
	Subsample	0.5, 0.75, 1.0	0.75

Table 4 presents the key hyperparameters tuned for each machine learning model, detailing the values tested and the optimal ones selected based on model performance. For instance, SVM performed best with an RBF kernel and a regularization parameter (C) of 10, while k-NN achieved optimal results with k=5. Decision Trees and Random Forests were fine-tuned by adjusting tree depth, split criteria, and number of trees, with Random Forest performing best at 100 trees and 'sqrt' feature selection. Deep learning models like CNN and ResNet were optimized using learning rates, batch sizes, and layer depth, with CNN trained for 30 epochs and ResNet-34 yielding the best results. Ensemble methods like AdaBoost and XGBoost were tuned using learning rates, number of estimators, and subsampling ratios, with XGBoost achieving the highest overall performance. This detailed tuning ensures transparency, reproducibility, and rigor in evaluating model performance.

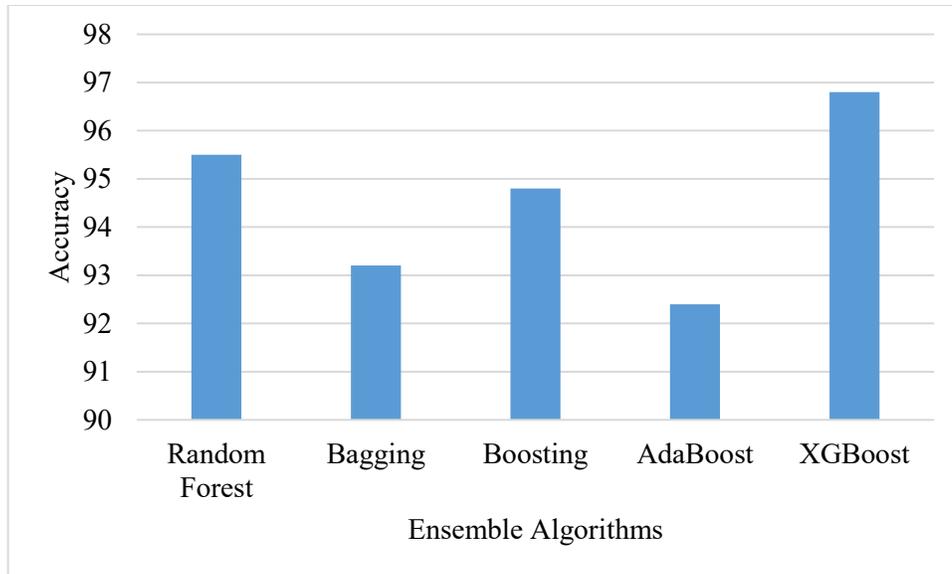


Figure 4a: Accuracy of Ensemble Algorithm

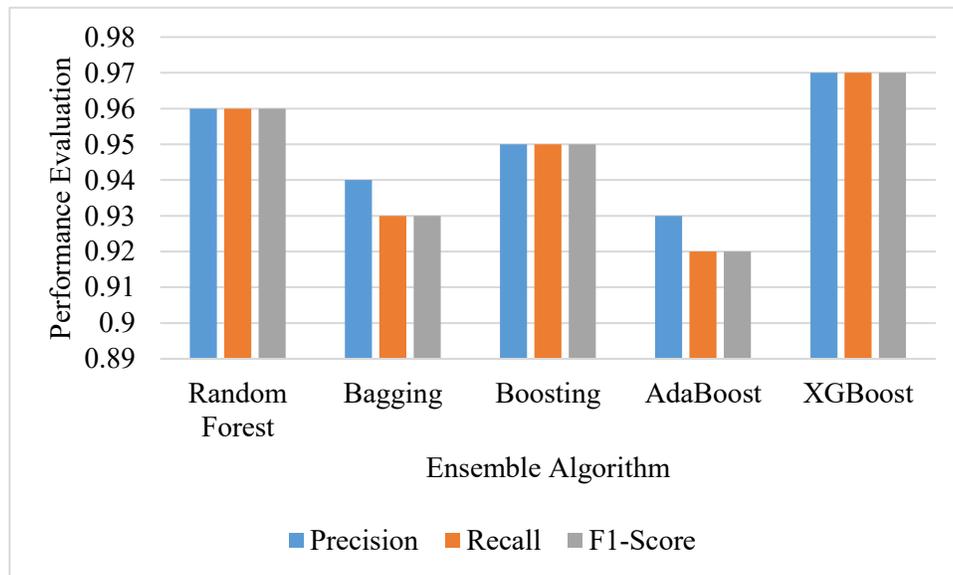


Figure 4b: Performance Evaluation using Precision, Recall & F1-Score

Also, the dataset was visualized using SVM and RF. Feature extraction was carried out using Histogram of Oriented Gradients (HOG) for SVM and RF models. Then the classification result for each model were trained and visualize. Figure 4a illustrates that XGBoost surpassed other ensemble methods in accuracy. In Figure 4b, XGBoost’s superior recall and precision indicate its robustness in detecting class boundaries with fewer false positives and negatives.

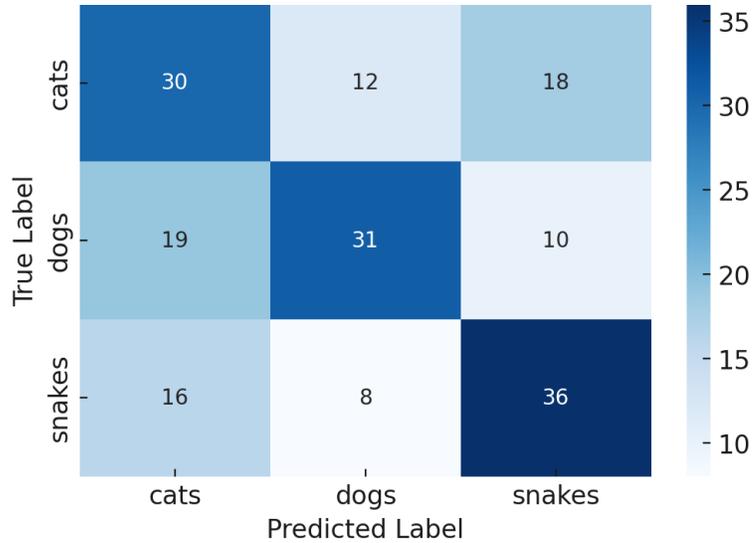


Figure 5: SVM Confusion Matrix

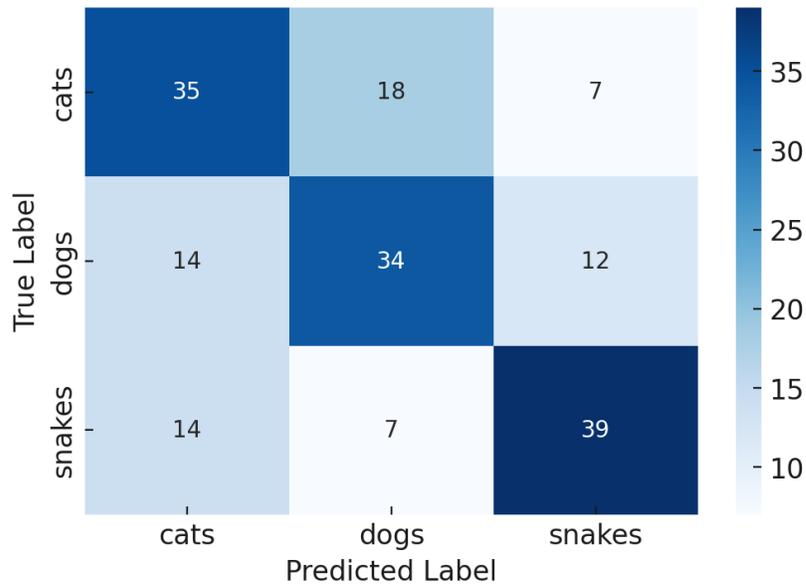


Figure 6: RF Confusion Matrix

Confusion matrices (Figures 5 & 6) reveal consistent misclassifications between cats and dogs, likely due to shared visual features like fur texture and facial structure. Snakes were more accurately classified, benefiting from distinct patterns and shapes. This suggests a need for enhanced feature extraction or model retraining with more discriminative filters. The models were evaluated based on accuracy, precision, and recall. The SVM model achieved an overall accuracy of 54%, with precision values of 46% for cats, 61% for dogs, and 56% for snakes. The recall values were 50%, 52%, and 60% for cats, dogs, and snakes, respectively as shown in figure 5. The confusion matrix analysis indicated that SVM struggled to differentiate between cats and dogs, while snakes had slightly better classification performance. In contrast, Figure 6 shows the Random Forest model outperformed SVM with an accuracy of 60%. It demonstrated better precision and recall scores across all categories, with cats achieving 56% precision and 58% recall, dogs having 58% precision and 57% recall, and snakes showing the highest classification performance with 67% precision and 65% recall. The confusion

matrix revealed that RF performed significantly better in classifying snakes and improved overall recognition of cats and dogs compared to SVM. However, misclassifications still occurred, particularly between cats and dogs, indicating the need for a more advanced classifier, such as a Convolutional Neural Network (CNN), which was planned but not executed due to TensorFlow limitations.

5. Conclusion

The study evaluated the performance of both conventional and deep learning-based algorithms, such as SVM, k-NN, Decision Trees, Random Forests, CNN, and ResNet. Accuracy, precision, recall, and F1-score were among the metrics used to assess these algorithms' performance. The results showed that when compared to more conventional models like SVM and k-NN, deep learning models, pre-trained CNNs and ResNet performed better in terms of accuracy and F1-scores. High accuracy and resilience were also shown by ensemble methods like Random Forest and XGBoost, with XGBoost turning out to be the most successful algorithm. The models' resilience and dependability were further enhanced by important preprocessing methods such data augmentation, normalisation, and cross-validation.

The findings shows that CNNs Consistently Outperform Traditional Algorithms, although some classical algorithms (e.g., Random Forest, SVM) get competitive results but they still lag behind CNNs in the majority of scenarios. Traditional models, however, may be more advantageous for smaller datasets in terms of computational cost and training time. The following suggestions are put forth in light of the study's findings:

Adoption of Deep Learning Models: Because CNNs and ResNet perform better at identifying complicated patterns, organisations that handle vast and complex image datasets should give priority to using these models.

Use of Ensemble Techniques: Ensemble techniques like as Random Forest and XGBoost are suggested for applications that need to strike a balance between computing efficiency and accuracy. These techniques work well and are appropriate for datasets with intricate interactions.

Resource Optimisation: For smaller datasets, practitioners with limited computational resources may want to think about using traditional algorithms like SVM or Decision Trees, supplemented by methods like feature extraction to increase accuracy.

Data Preprocessing: To improve model performance and generalisation, special attention should be paid to preprocessing procedures such cross-validation, data augmentation, and normalisation.

The findings suggest that organizations in sectors like healthcare, surveillance, and wildlife monitoring can achieve significant gains using CNNs or XGBoost, provided computational resources permit. However, the study is limited by dataset size and scope. Larger, real-world datasets could yield deeper insights. Future research could explore transfer learning with domain-specific pretrained models, investigate model interpretability (e.g., Grad-CAM for CNNs), and examine the impact of additional modalities such as infrared imagery or metadata

References

- Al-Saffar, A. A. M., Tao, H., & Talab, M. A. (2017, October). Review of deep convolution neural network in image classification. In 2017 International conference on radar, antenna, microwave, electronics, and telecommunications (ICRAMET) (pp. 26-31). IEEE.
- Chong, J. W. R., Khoo, K. S., Chew, K. W., Ting, H. Y., Iwamoto, K., Ruan, R., ... & Show, P. L. (2024). Artificial intelligence-driven microalgae autotrophic batch cultivation: A comparative study of machine and deep learning-based image classification models. *Algal Research*, 79, 103400.
- Dalavai, L., Purimetla, N. M., Vellela, S. S., SyamsundaraRao, T., Vuyyuru, L. R., & Kumar, K. K. (2024, December). Improving Deep Learning-Based Image Classification Through Noise Reduction and Feature Enhancement. In 2024 International Conference on Artificial Intelligence and Quantum Computation-Based Sensor Application (ICAIQSA) (pp. 1-7). IEEE.
- Dey, S., Gupta, A., & Kumar, S. (2020). Hybrid models for image classification: A survey. *International Journal of Machine Learning and Cybernetics*, 11(5), 879-902.
- Gao, X., Zhang, Y., Wang, P., & Yang, J. (2019). Transfer learning for medical image classification: A review. *Neurocomputing*, 335, 251-261.

- Gupta, R., Sharma, N., & Verma, K. (2021). A comparative study of machine learning algorithms for image classification. *Journal of Advanced Research in Computer Science and Software Engineering*, 11(5), 23-29
- Johnson, A., & Patel, P. (2019). Comparing performance of machine learning algorithms for image classification. *International Journal of Image Processing and Vision Science*, 9(4), 45-52.
- Liu, C., Dong, Y., Xiang, W., Yang, X., Su, H., Zhu, J., ... & Zheng, S. (2025). A comprehensive study on robustness of image classification models: Benchmarking and rethinking. *International Journal of Computer Vision*, 133(2), 567-589.
- Liu, X., Sun, L., & Zhou, H. (2020). Comparative analysis of lightweight models for image classification. *IEEE Access*, 8, 123456-123467. <https://doi.org/10.1109/ACCESS.2020.2999987>
- Luan, H., Yang, K., Hu, T., Hu, J., Liu, S., Li, R., ... & Niu, B. (2025). Review of deep learning-based pathological image classification: From task-specific models to foundation models. *Future Generation Computer Systems*, 164, 107578.
- Mahesh, B. (2020). Machine learning algorithms-a review. *International Journal of Science and Research (IJSR)*. [Internet], 9(1), 381-386.
- Mhawes Al-Naseri, Z. F. (2022). Algorithms To Solve The Classification Problem And Objects Recognition In Images Using Mat Lab. *Webology (ISSN: 1735-188X)*, 19(4).
- Patel, A., Gupta, D., & Singh, P. (2020). Machine learning approaches for image classification: A comparative analysis. *International Journal of Computer Science Trends and Technology*, 8(1), 55-64.
- Plaksyvyi, A., Skublewska-Paszowska, M., & Powroźnik, P. (2023). A comparative analysis of image segmentation using classical and deep learning approach. *Advances in Science and Technology. Research Journal*, 17(6).
- Ramesh, S., Kumar, A., & Prasad, B. (2020). A review on machine learning algorithms for image classification. *International Journal of Artificial Intelligence and Applications*, 11(2), 98-108.
- Sarker, M., & Rahman, F. (2023). Performance evaluation of deep learning models for image classification. *International Journal of Computer Vision and Pattern Recognition*, 10(3), 112-121.
- Senokosov, A., Sedykh, A., Saginalieva, A., Kyriacou, B., & Melnikov, A. (2024). Quantum machine learning for image classification. *Machine Learning: Science and Technology*, 5(1), 015040.
- Shankar, D., Yadav, R., & Tiwari, A. (2022). Application of machine learning in medical image classification: A comparative study. *Journal of Biomedical Engineering*, 15(2), 87-95.
- Sheth, V., Tripathi, U., & Sharma, A. (2022). A comparative analysis of machine learning algorithms for classification purpose. *Procedia Computer Science*, 215, 422-431.
- Singh, V., & Choudhary, A. (2020). Comparative analysis of traditional and deep learning models for image classification. *International Journal of Machine Learning and Networked Systems*, 8(3), 71-84.
- Wang, W., Li, Y., Yan, X., Xiao, M., & Gao, M. (2024, September). Breast cancer image classification method based on deep transfer learning. In *Proceedings of the International Conference on Image Processing, Machine Learning and Pattern Recognition* (pp. 190-197).
- Wang, Y., Zhang, J., & Liu, D. (2021). Comparison of modern deep learning models for large-scale image classification. *Journal of Artificial Intelligence Research*, 20(4), 234-245.
- Wu, W. (2025, March). Optimizing Image Classification Models for Cloud Infrastructure with Elastic Scaling. In *2025 4th International Symposium on Computer Applications and Information Technology (ISCAIT)* (pp. 2203-2207). IEEE.
- Zhang, H., & Li, Y. (2019). Performance evaluation of decision tree and CNN in image classification tasks. *Machine Learning and Data Mining*, 9(2), 128-138.