27th International Conference on Computer and Information Technology (ICCIT)
December 20-22, 2024, Cox's Bazar, Bangladesh

Synergetic KDRF Classifier: A Random Forest-Based Hybrid Approach for Multi-Class Classification Task

Israt Jahan Khan[1], Md. Fahim Bin Amin[1], Kamrul Islam Shahin[2], Huu-Hoa Nguyen[3] and Dewan Md. Farid[1]
[1]Department of Computer Science and Engineering, United International University,
United City, Madani Avenue, Badda, Dhaka 1212, Bangladesh
[2]Software Engineering, The Maersk Mc-Kinney Moller Institute,
University of Southern Denmark, Campusvej 55, DK-5230 Odense M, Denmark
[3]College of Information and Communication Technology, Can Tho University
3/2 Street, Ninh Kieu District, Can Tho City, Vietnam
Email: {ikhan201142, mamin201158}@bscse.uiu.ac.bd, kish@mmmi.sdu.dk,
nhhoa@ctu.edu.vn, dewanfarid@cse.uiu.ac.bd

2024 27th International Conference on Computer and Information Technology (ICCIT) | 979-8-3315-1909-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICCIT64611.2024.11021799

***Abstract*—Multi-class classification presents a significant challenge in supervised machine learning, and it is frequently applied across various real-world domains. Random Forest (RF) stands out as a popular ensemble technique for multi-class classification tasks that utilizes attribute bagging to construct random trees and employs a majority-voting scheme for classification. However, the existing Random Forest approach often faces limitations in accuracy and efficiency, particularly for more critical tasks. In this study, we have addressed this gap by introducing a novel methodology that integrates the K-Nearest Neighbor (KNN) classifier and Decision Trees into the Random Forest framework to enhance performance. The proposed method has involved training a K-Nearest Neighbor and a Decision Tree (DT) classifier on the extracted features. The output probabilities from these classifiers have been used as meta-features, forming an enriched feature vector fed into a Random Forest as the meta-learner for final classification. Through extensive experimentation on 5 datasets, the proposed approach has demonstrated superior performance in accuracy and efficiency compared to the traditional method. By integrating Random Forest and K-Nearest Neighbor techniques, this research has offered a versatile solution applicable across diverse domains where precise multi-class classification is essential.**

Multi-class Classification; Decision Tree; K-Nearest Neighbor; Meta-Features; Random Forest

I. Introduction

Classification models such as K-Nearest Neighbors (KNN) and Decision Trees (DT) are fundamental tools in machine learning. These are widely used to categorize data into predefined classes [1]. However, despite their strengths, these models face significant limitations. KNN struggles with irrelevant features in high-dimensional spaces, while DTs are prone to overfitting, especially in large and complex datasets [2]. To address these issues, ensemble techniques such as Random Forests (RF) have been developed. RF constructs multiple decision trees on random subsets of the data and aggregates the predictions to improve accuracy and reduce overfitting [3], [4]. Traditional Random Forest (RF) is powerful and widely used but it faces several critical limitations when applied to complex or high-dimensional datasets. RF relies solely on Decision Trees, which can become biased toward specific features, particularly when the feature set is homogeneous or lacks diversity. As a result, it leads to overfitting. Additionally, RF treats all features equally during decision-making, which reduces accuracy in high-dimensional datasets with irrelevant or noisy features. It also struggles with imbalanced data and fails to capture local data patterns effectively, particularly in cases with complex class boundaries [1]. Despite its ensemble nature, RF requires careful hyperparameter tuning, and its inability to exploit feature interactions fully limits its performance in tasks such as disease diagnosis, customer feedback analysis, and object recognition [5], [6]. Addressing these shortcomings is crucial for improving the effectiveness of RF in real-world applications.

In response to this critical research gap in various classification tasks, this paper proposes a novel approach, named the "Synergetic KDRF Classifier," which combines 'K' for K-Nearest Neighbors (KNN), 'D' for Decision Trees (DT), and 'RF' for Random Forest (RF) to enhance multi-class classification performance for images. The integration of KNN and DT enables dynamic partitioning of the dataset and improves capturing local patterns and complex boundaries. The "Synergetic KDRF Classifier" enhances traditional Random Forest by using high-level features extracted from a fine-tuned MobileNetV2. These features train KNN (with k=5) and DT classifiers, whose output probabilities are used as meta-features, concatenated into an enriched feature vector, and then passed to a Random Forest for final predictions. The proposed Synergetic KDRF classifier overcomes many limitations of traditional Random Forest by integrating K-Nearest Neighbors and decision trees into the learning process. KNN introduces a distance-based decision-making layer that mitigates feature dependency and bias, while decision trees capture hierarchical relationships. This combination allows the model to better handle noisy data, as KNN smooths out local patterns, reducing the impact of outliers. Furthermore, the KDRF algorithm is less sensitive to hyperparameter tuning, because each component contributes different strengths. It makes the model more stable and less prone to overfitting.

979-8-3315-1909-4/24/$31.00 ©2024 IEEE

Authorized licensed use limited to: Southeast University. Downloaded on January 04,2026 at 04:01:19 UTC from IEEE Xplore. Restrictions apply.

By using Random Forest as a meta-learner on top of KNN and Decision Tree predictions, KDRF benefits from diverse feature representations.

II. Literature Review

Sun et al. [7] proposed an enhanced random forest algorithm that improves classification accuracy and decision tree diversity. Their approach evaluates trees across multiple datasets for reliability and quantifies diversity using cosine similarity between trees. The improved random forest shows superior performance, with higher classification accuracy, G-means, and out-of-bag (OOB) scores compared to five other methods. Lilhore et al. [8] have introduced a Hybrid Predictive Model (HPM) combining enhanced random forest with a support vector machine to improve accuracy and address data imbalance in HCV prediction, showing superior results over existing methods. Li et al. [9] have developed a Self-Attention Random Forest (SARF) model, achieving 97.16% accuracy and an AUC of 0.9713 for binary classification on the BreaKHis dataset by utilizing multi-scale fusion features. Jain et al. [10] proposed a "logically randomized forest" (LRF) algorithm to address performance issues in tree-based ensemble methods with scarce informative features, integrating feature-level engineering with stratified sampling. Evaluations have shown that LRF outperforms existing methods.

Srisuradetchai et al. [11] introduced Random Kernel K-Nearest Neighbors (RK-KNN) regression, which integrates kernel smoothing with bootstrap sampling. RK-KNN demonstrated superior performance over standard KNN and Random KNN (R-KNN) in terms of RMSE and R-squared. Karabadji et al. [12] have introduced GARF, a Genetic Algorithm optimizing Random Forest (RF) for improved accuracy and diversity. It has outperformed classical and evolutionary methods on UCI datasets, including IoT Botnet traffic detection. Das et al. [13] have proposed a hybrid model combining CNN with RF for BSL recognition. It has surpassed existing systems by leveraging transfer learning and RF algorithms with effective background elimination and validation using Ishara-Lipi datasets. Bravo et al. [14] have introduced a streamlined RF model for predicting energy supply and solar fraction in thermosiphon solar water heating systems. It has offered high accuracy and robustness with MAPE between 2.94% and 5.86%.

Pazhanikumar et at. [15] have improved satellite image classification using datasets SAT-4, SAT-6, and RSI-CB. It combines features extracted by VGG19 and ResNet50, and classifies them using a modified Random Forest with an empirical loss function. The method outperforms Decision Tree and K-Nearest Neighbors in precision, accuracy, F1-score, and recall. Zhang et al. [16] have improved forest land resource classification using Sentinel-2 satellite images by evaluating Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Random Forest (RF), Decision Trees (DT), and Multi-layer Perceptron (MLP) algorithms. It found that SVM achieved the highest overall accuracy (95.8%), outperforming other algorithms. The study demonstrated that incorporating texture features enhanced classification accuracy and provided efficient methods for forest resource management and utilization. Anyanwu et al. [17] have introduced a hyper-tuned ensemble Random Forest algorithm, employing Randomised Search Cross-Validation (RSCV) for precise detection of false Basic Safety Messages (BSMs). Their False BSM Detection System (FBDS) has effectively secured IoV against false safety measures and outperformed established ML algorithms. Thakur et al. [18] has introduced a modified Guided Regularized Random Forest (mGRRF) for feature selection, aiming to reduce bias in traditional RF-based methods. It has utilized permutation importance to select vital features and demonstrated improved human activity recognition accuracy, up to 98%, using multiple classifiers.

III. Methodology

A. Core Methods for Classification Tasks

1) Random Forest (RF): It's a popular machine learning algorithm for classification and regression, constructing multiple decision trees from random subsets of data and features, and aggregating their predictions for the final result [7], [10].

Random Forest Formulation: RF utilizes bagging, creating bootstrap samples for each tree from the training set $\{(X_1, y_1), (X_2, y_2), \ldots, (X_N, y_N)\}$:

$$\mathcal{D}_t = \{(X_1^t, y_1^t), \ldots, (X_N^t, y_N^t)\} \tag{1}$$

Tree Construction: For each bootstrap sample, a decision tree is built:

- **Node Splitting:** A random subset of m features is selected from p total features:

$$\mathcal{M}_t \subseteq \{1, 2, \ldots, p\}, \quad |\mathcal{M}_t| = m \tag{2}$$

- **Best Split Selection:** The best split is determined based on an impurity measure (e.g., Gini impurity or variance).

Prediction Aggregation: Predictions from all trees are aggregated:

- **Classification:** The final prediction is the majority vote:

$$\hat{y} = \text{mode}\{\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_T\} \tag{3}$$

- **Regression:** The final prediction is the average:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^{T} \hat{y}_t \tag{4}$$

Random Subspace Method: This ensures randomness and reduces overfitting by selecting a subset of features at each node.

2) K-Nearest Neighbor (KNN): It's a simple algorithm for classification and regression, predicting values based on the K nearest neighbors in the feature space [19].

Authorized licensed use limited to: Southeast University. Downloaded on January 04,2026 at 04:01:19 UTC from IEEE Xplore. Restrictions apply.

Key Formulas: **a. Euclidean Distance:** The distance between points X and Y is given by:

$$d(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{5}$$

b. Classification Prediction: The predicted class $\hat{y}$ is determined by majority vote:

$$\hat{y} = \text{mode}\{y_1, y_2, \ldots, y_K\} \tag{6}$$

c. Regression Prediction: The predicted value $\hat{y}$ is the mean of the K neighbors:

$$\hat{y} = \frac{1}{K}\sum_{i=1}^{K} y_i \tag{7}$$

3) Decision Tree (DT): It's a supervised learning algorithm for classification and regression tasks, modeling decisions through a tree-like structure based on feature questions. Classification Trees are used for categorical targets, predicting class labels. The common method for building DTs is the Classification And Regression Tree (CART) approach, utilizing Gini Impurity and Information Gain as key metrics.

a. Gini Index: Gini Impurity measures misclassification probability:

$$Gini = 1 - \sum_{i=1}^{K} p_i^2 \tag{8}$$

The weighted Gini after a split is:

$$Gini_{\text{split}} = \sum_{j=1}^{m} \frac{|D_j|}{|D|} Gini_j \tag{9}$$

b. Information Gain: Information Gain (IG) measures entropy reduction:

$$IG = H_{\text{parent}} - \sum_{i=1}^{m} \frac{|D_i|}{|D|} H_i \tag{10}$$

with entropy defined as:

$$H = -\sum_{j=1}^{K} p_j \log_2(p_j) \tag{11}$$

Post-pruning, particularly cost complexity pruning, helps reduce overfitting:

$$C_\alpha(T) = R(T) + \alpha|T| \tag{12}$$

where $R(T)$ is the misclassification rate and α balances size and accuracy.

B. The Proposed Synergetic KDRF Classifier

The "Synergetic KDRF Classifier" algorithm enhances multiclass classification task by integrating K-Nearest Neighbors (KNN), Decision Tree (DT), and Random Forest (RF) classifiers. The process starts with feature extraction using a fine-tuned MobileNetV2 model, which extracts high-level, discriminative features from dermoscopic images. KNN, with k=5 neighbors, and a DT classifier are then trained on these features to capture local patterns and complex decision boundaries. Instead of using these classifiers directly for final predictions, we use their output probabilities as meta-features. These probabilities are concatenated to form an enriched feature vector, which is then fed into a Random Forest with 200 trees. This Random Forest, which functions as a meta-learner, combines the strengths of KNN and DT, learning to optimally utilize their predictions and overcoming their individual limitations. This ensemble approach improves the model's performance and generalization by leveraging the complementary strengths of the base classifiers. During the testing phase, the model extracts features with MobileNetV2, predicts probabilities with KNN and DT, and performs the final classification using the Random Forest meta-learner. This method provides robust classification and reduces overfitting by utilizing a diverse set of classifiers and a meta-learning strategy. Our novel approach for the Random Forest algorithm has been detailed in Algorithm 1 and Figure 1 illustrates the workflow diagram of this algorithm.

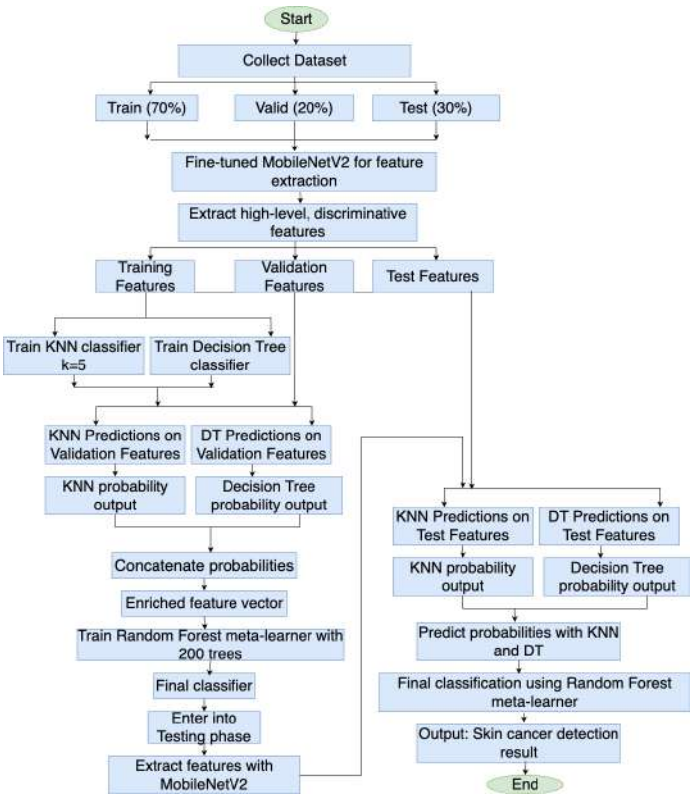


Fig. 1. Workflow diagram of Synergetic KDRF Classifier.

Authorized licensed use limited to: Southeast University. Downloaded on January 04,2026 at 04:01:19 UTC from IEEE Xplore. Restrictions apply.

IV. Results and Discussion

We have divided this section into five subsections: a) Dataset Overview, b) Experimental Setup, c) Code Pipeline, d) Results Discussion, and e) Reasons for Improved Performance of "Synergetic KDRF Classifier".

A. *Dataset Overview*

The dataset utilized in this research comprises 2D RGB images of 5 distinct types, sourced from the Roboflow website. It is divided into training, testing, and validation sets in a 70:10:20 ratio, respectively. To enhance the robustness of the model, several preprocessing techniques were applied, including zoom, rotation, horizontal and vertical flips, as well as brightness and contrast adjustments. Augmentation was performed to increase dataset variability and reduce the risk of overfitting. These preprocessing steps ensured the dataset's readiness for efficient and effective deep learning model training and evaluation. Table II provides key information about each dataset, along with a detailed analysis of individual performance metrics.

B. *Experimental setup*

The environment for the experiment is set up on a machine equipped with an NVIDIA GeForce RTX 3060 GPU, which has 12 GB of memory, Driver Version 560.81, and CUDA Version 12.6. The machine uses TensorFlow for deep learning tasks, including utilizing the MobileNetV2 architecture, and Scikit-learn for traditional machine learning classifiers such as Random Forest, KNN, and Decision Tree. Table I summarizes the infrastructure used in our derivation.

TABLE I
System environment setup for deep learning experiments.

Component	Details
GPU	NVIDIA GeForce RTX 3060
GPU Memory	12 GB
Driver Version	560.81
CUDA Version	12.6
Power Usage	11W / 170W (At the time of the NVIDIA-SMI output)
Fan Temperature	49°C
TensorFlow Version	Latest compatible with CUDA 12.6
Python Version	Python 3.12
Deep Learning Model	MobileNetV2 (with custom layers for classification)
ML Libraries	Scikit-learn (Random Forest, KNN, Decision Tree), TensorFlow, Keras, Numpy, Pandas, Matplotlib, Seaborn
Memory Usage (GPU)	10,571 MiB / 12,288 MiB
Process Running	Python script (PID: 1497), using CUDA for deep learning model training
Operating System	Linux

C. *Code Pipeline*

Figure 2 illustrates the coding pipeline diagram of our Python-based system. The Synergetic KDRF classifier uses a fine-tuned MobileNetV2 to extract features from input images, which are passed to K-Nearest Neighbors (KNN) and Decision Trees (DT) to predict class probabilities. These probability outputs are concatenated to form an enriched feature vector, combining the strengths of both classifiers. This enriched vector is subsequently passed to a Random Forest (RF) model, which acts as a meta-learner, using the combined information to make the final classification decision.

Algorithm 1 Synergetic KDRF Classifier

Input: Extracted features from CNN model (MobileNetV2). Corresponding labels for training, validation, and test sets.
Output: Final classification predictions on the test set.
Method:

1) Train KNN on Training Features:
   a) Initialize K-Nearest Neighbors (KNN) classifier.
   b) **For each** instance in the training set:
      i) Find the nearest neighbors for that instance.
      ii) Train the KNN model using the training data.
   c) **End For**
2) Train Decision Tree on Training Features:
   a) Initialize a Decision Tree classifier.
   b) **For each** instance in the training set:
      i) Build a decision tree using the training data.
   c) **End For**
3) Generate Predictions (Probabilities) on Validation Set:
   a) **For each** instance in the validation set:
      i) Use the trained KNN to predict probabilities.
      ii) Use the trained DT to predict probabilities.
   b) **End For**
4) Concatenate the probability predictions from both KNN and Decision Tree to form a combined feature set.
5) Train Random Forest on Combined Features:
   a) Initialize a Random Forest classifier with a predefined number of estimators.
   b) **For each** combined feature set:
      i) Train the Random Forest model using the combined features and validation labels.
   c) **End For**
6) Test Phase (Final Evaluation):
   a) **For each** instance in the test set:
      i) Predict probabilities using KNN.
      ii) Predict probabilities using Decision Tree.
      iii) Concatenate the test probabilities to form a combined feature set.
   b) **End For**
7) Use the trained Random Forest to predict final labels on the test features and calculate test accuracy by comparing the predictions with the true labels.

D. *Results Discussion*

Table II have demonstrated the improvement in performance metrics with the Synergetic KDRF Classifier compared to the standard Random Forest. The KDRF Classifier has achieved

Authorized licensed use limited to: Southeast University. Downloaded on January 04,2026 at 04:01:19 UTC from IEEE Xplore. Restrictions apply.

TABLE II
DATASET SUMMARY AND COMPARISON OF RF AND OTHER CUTTING EDGE ALGORITHMS VS. SYNERGETIC KDRF CLASSIFIER.

Dataset Name	Source	Total class	File Count	Remarks	Accuracy (only RF)	Accuracy (gcForest)	Accuracy (GBRF)	Accuracy (RF+SVM)	Performance Evaluation (Synergetic KDRF Classifier)		
									Accuracy	Precision	Recall
Skin Cancer Detection	Roboflow	7	9900	Focuses on detecting various skin cancer types.	85.02%	79.11%	83.88%	85.6%	**86.46%**	85.8%	85.1%
Diabetic Retinopathy Screening	Roboflow	5	2838	Assists in screening diabetic retinopathy stages.	79.2%	80.14%	81.92%	79.8%	**82.61%**	81.18%	81.3%
Renal Failure Analysis	Roboflow	4	9574	Aids in the identification of renal failure cases.	91.61%	95.76%	90.23%	91.2%	**95.7%**	94.8%	95.87%
Skin Disease	Roboflow	17	4537	Covers multiple skin diseases for diagnostic aid.	89.12%	88.8%	91.43%	89.3%	**94.12%**	91.4%	93.08%
Ocular Diseases	Roboflow	4	6009	Focuses on a collection of ocular diseases from eye images.	89.38%	91.79%	87.25%	89.7%	**92.8%**	93.6%	92.01%

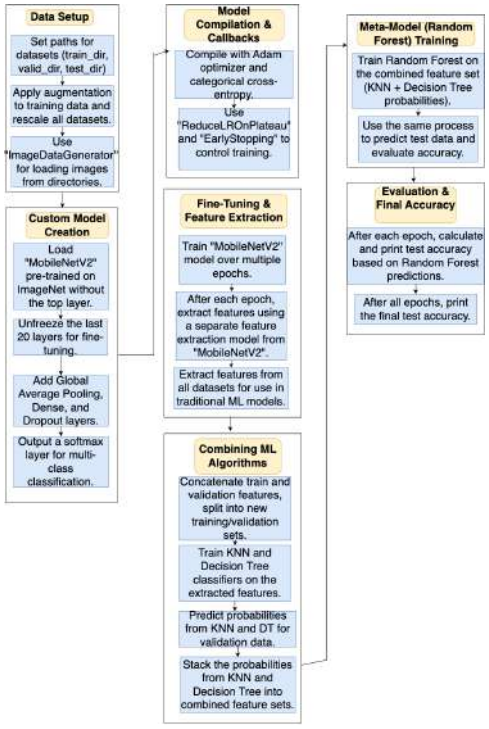


Fig. 2. Code Pipeline of Synergetic KDRF Classifier

test accuracies between 82.61% and 95.7%, surpassing RF's range of 79.2% to 91.61%. This improvement has been especially notable in Skin Cancer Detection and Ocular Diseases, where accuracy increases have ranged from 1.44% to 5%. Figures 3 to 12 compare the training and validation accuracy curves of GBRF, gcForest, RF+SVM, two traditional RF models, and our proposed model across each dataset. The KDRF classifier consistently outperforms all other methods.

The standard RF model has achieved accuracies from 79.2% to 91.61%, serving as a solid baseline. However, it has lacked the optimization capabilities of more advanced classifiers. The gcForest (Multi-Grained Cascade Forest) model has improved upon traditional RF, achieving accuracies between 79.11% and 95.76%. While it has captured more complex patterns through gradient boosting, it has still fallen short of KDRF's accuracy. The GBRF (Gradient Boosted RF) model has achieved accuracies ranging from 81.92% to 91.43%. Although it has combined gradient boosting with RF's robustness, it has not consistently outperformed the KDRF Classifier. The RF+SVM model has produced competitive accuracies from 79.8% to 89.7%, leveraging the strengths of both algorithms. Despite improvements over standard RF, it has not matched the performance of the Synergetic KDRF Classifier, particularly in datasets like Skin Cancer Detection and Ocular Diseases.

E. Reasons for Synergetic KDRF Classifier's Improved Performance

The Synergetic KDRF Classifier uses a pre-trained MobileNetV2 for efficient high-level feature extraction from images. It enhances computational efficiency and reduces model size for faster training. Instead of training RF on raw features, the improved model uses probability distributions from KNN and DT classifiers as meta-features that simplify the input space and reduce dimensionality and computational complexity. By stacking KNN and DT predictions into a feature vector for RF, the model reduces complexity and allows RF to focus on a smaller, more relevant feature set while efficiently leveraging the strengths of each classifier. Training RF on meta-features from KNN and DT is faster because they capture key patterns efficiently. Pre-trained models also reduce the need for extensive raw data training. The improved RF model fine-tunes parameters like the number of trees and maximum depth to balance performance and efficiency. It reduces computational overhead and provides high accuracy.

V. CONCLUSION

The Synergetic KDRF Classifier has consistently outperformed the Random Forest model across all datasets, with test accuracy improvements ranging from 1.44% to 5%. It

Authorized licensed use limited to: Southeast University. Downloaded on January 04,2026 at 04:01:19 UTC from IEEE Xplore. Restrictions apply.



Fig. 3. RF: Diabetic Screening



Fig. 4. KDRF: Diabetic Screening

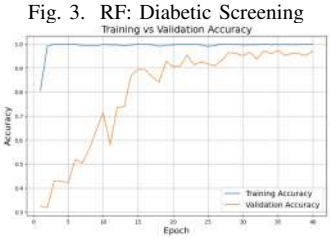


Fig. 5. RF: Skin Diseases



Fig. 6. KDRF: SkinDiseases

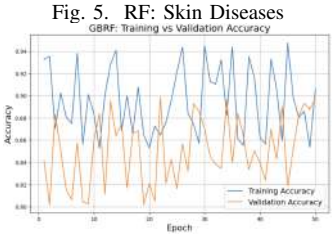


Fig. 7. GBRF: Renal Failure

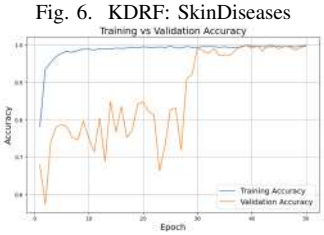


Fig. 8. KDRF: RenalFailure

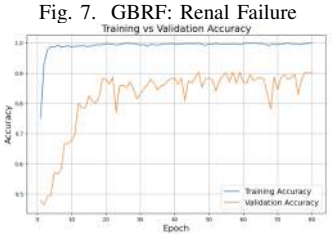


Fig. 9. gcForest: Ocular Diseases



Fig. 10. KDRF: Ocular Diseases

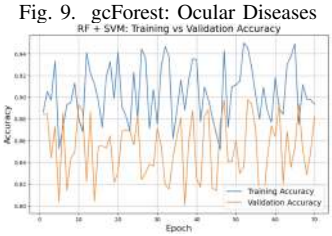


Fig. 11. RF+SVM: Skin Cancer



Fig. 12. KDRF: SkinCancer

and capture complex patterns in medical data.

has also achieved higher validation accuracy than the RF and other cutting-edge models which can highlight its superior performance and robustness. Overall, it has proven to be a reliable tool for real-world classification tasks. Also, it demonstrates dependability across various metrics. This study shows the effectiveness of the proposed method can overcome the limitations of RF. By integrating KNN with Random Forest, we have enhanced accuracy and efficiency in multi-class classification, which proves it to be a robust and adaptable solution across various datasets. In future work, we plan to integrate advanced deep learning models, including Transformer-based architectures and Graph Neural Networks with the Synergetic KDRF Classifier, to enhance performance

REFERENCES

[1] D. A. Shifman, I. Cohen, K. Huang, X. Xian, and G. Singer, "An adaptive machine learning algorithm for the resource-constrained classification problem," *Engineering Applications of Artificial Intelligence*, vol. 119, p. 105741, 2023.

[2] T. Lazebnik and S. Bunimovich-Mendrazitsky, "Decision tree post-pruning without loss of accuracy using the sat-pp algorithm with an empirical evaluation on clinical data," *Data & Knowledge Engineering*, vol. 145, p. 102173, 2023.

[3] M. Shi, W. Hu, M. Li, J. Zhang, X. Song, and W. Sun, "Ensemble regression based on polynomial regression-based decision tree and its application in the in-situ data of tunnel boring machine," *Mechanical Systems and Signal Processing*, vol. 188, p. 110022, 2023.

[4] M. Alrifaey, W. H. Lim, C. K. Ang, E. Natarajan, M. I. Solihin, M. R. M. Juhari, and S. S. Tiang, "Hybrid deep learning model for fault detection and classification of grid-connected photovoltaic system," *IEEE Access*, vol. 10, pp. 13 852–13 869, 2022.

[5] A. R. Zarei, M. R. Mahmoudi, and M. M. Moghimi, "Determining the most appropriate drought index using the random forest algorithm with an emphasis on agricultural drought," *Natural Hazards*, vol. 115, no. 1, pp. 923–946, 2023.

[6] M. Belgiu and L. Drăguţ, "Random forest in remote sensing: A review of applications and future directions," *ISPRS journal of photogrammetry and remote sensing*, vol. 114, pp. 24–31, 2016.

[7] Z. Sun, G. Wang, P. Li, H. Wang, M. Zhang, and X. Liang, "An improved random forest based on the classification accuracy and correlation measurement of decision trees," *Expert Systems with Applications*, vol. 237, p. 121549, 2024.

[8] U. K. Lilhore, P. Manoharan, J. K. Sandhu, S. Simaiya, S. Dalal, A. M. Baqasah, M. Alsafyani, R. Alroobaea, I. Keshta, and K. Raahemifar, "Hybrid model for precise hepatitis-c classification using improved random forest and svm method," *Scientific Reports*, vol. 13, no. 1, p. 12473, 2023.

[9] J. Li, J. Shi, J. Chen, Z. Du, and L. Huang, "Self-attention random forest for breast cancer image classification," *Frontiers in Oncology*, vol. 13, p. 1043463, 2023.

[10] N. Jain and P. K. Jana, "Lrf: A logically randomized forest algorithm for classification and regression problems," *Expert Systems with Applications*, vol. 213, p. 119225, 2023.

[11] P. Srisuradetchai and K. Suksrikran, "Random kernel k-nearest neighbors regression," *Frontiers in big Data*, vol. 7, p. 1402384, 2024.

[12] N. E. I. Karabadji, A. A. Korba, A. Assi, H. Seridi, S. Aridhi, and W. Dhifli, "Accuracy and diversity-aware multi-objective approach for random forest construction," *Expert Systems with Applications*, vol. 225, p. 120138, 2023.

[13] S. Das, M. S. Imtiaz, N. H. Neom, N. Siddique, and H. Wang, "A hybrid approach for bangla sign language recognition using deep transfer learning model with random forest classifier," *Expert Systems with Applications*, vol. 213, p. 118914, 2023.

[14] I. Lillo-Bravo, J. Vera-Medina, C. Fernandez-Peruchena, E. Perez-Aparicio, J. A. Lopez-Alvarez, and J. M. Delgado-Sanchez, "Random forest model to predict solar water heating system performance," *Renewable Energy*, vol. 216, p. 119086, 2023.

[15] K. Pazhanikumar and S. N. KuzhalVoiMozhi, "Remote sensing image classification using modified random forest with empirical loss function through crowd-sourced data," *Multimedia Tools and Applications*, vol. 83, no. 18, pp. 53 899–53 921, 2024.

[16] C. Zhang, Y. Liu, and N. Tie, "Forest land resource information acquisition with sentinel-2 image utilizing support vector machine, k-nearest neighbor, random forest, decision trees and multi-layer perceptron," *Forests*, vol. 14, no. 2, p. 254, 2023.

[17] G. O. Anyanwu, C. I. Nwakanma, J. M. Lee, and D.-S. Kim, "Novel hyper-tuned ensemble random forest algorithm for the detection of false basic safety messages in internet of vehicles," *ICT Express*, vol. 9, no. 1, pp. 122–129, 2023.

[18] D. Thakur and S. Biswas, "Permutation importance based modified guided regularized random forest in human activity recognition with smartphone," *Engineering Applications of Artificial Intelligence*, vol. 129, p. 107681, 2024.

[19] A. Huang, R. Xu, Y. Chen, and M. Guo, "Research on multi-label user classification of social media based on ml-knn algorithm," *Technological Forecasting and Social Change*, vol. 188, p. 122271, 2023.

Authorized licensed use limited to: Southeast University. Downloaded on January 04,2026 at 04:01:19 UTC from IEEE Xplore. Restrictions apply.