International Journal of
Science, Architecture, Technology and Environment

Transforming Technology with Computer Vision: An Intelligent Image Detection Model for Classifying Weather Conditions

Olaiya Olayinka Oluwaseun[1*], Aderanti Aderemi Taiwo[2], Moshood Babatunde Ismail[3]

[1]Department of Computer Engineering, The Federal Polytechnic Ilaro, Ogun State, Nigeria

[2]Department of Computer Engineering, Gateway Polytechnic Saapade, Ogun State, Nigeria

[3]Department of Electrical and Electronics Engineering, Coventry University, United Kingdom

*Corresponding author, olayinka.olaiya@federalpolyilaro.edu.ng

DOI: https://doi.org/10.63680/ijsate1125077.062

Abstract

This study presents the development of an intelligent image detection model designed to classify weather conditions using modern computer vision techniques. As technology continues to evolve, automated interpretation of environmental information has become essential for applications such as transportation safety, climate monitoring, smart agriculture, and outdoor activity planning. The proposed system leverages deep learning algorithms—particularly convolutional neural networks (CNNs)—to analyze visual features in weather images and accurately categorize conditions such as sunny, cloudy, rainy, and foggy environments. The model was trained on a diverse dataset of labeled weather images to ensure high accuracy and robust performance across varying lighting and environmental scenarios. Evaluation results demonstrate that the system achieves reliable classification, highlighting its potential integration into real-world intelligent systems. Overall, this research contributes to the advancement of computer vision by providing an efficient, automated approach to weather condition detection that supports smarter, data-driven technological solutions.

Keywords: Computer Vision, Convolutional Neural Networks (CNNs), Deep Learning, Image Detection, Intelligent Systems, Weather Classification

1.0 INTRODUCTION

In recent years, rapid progress in computer vision has created new opportunities for automating and improving a wide range of image analysis tasks [1]. One notable area of interest is the classification of weather-related images, which plays an important role in meteorology, environmental surveillance, and even digital imaging systems. Accurate identification of weather conditions from images can enhance forecasting tools, improve user experience in weather applications, and support automated systems used for environmental observation [2].

This study aims to develop an intelligent image detection model that uses computer vision techniques to categorize weather images into three groups: **sunrise**, **shine (clear sky)**, and **rainy**. Each category presents its own visual characteristics and classification challenges. Sunrise images, for example, often include complex colour transitions and changing light intensities; shine images require the detection of clear, cloud-free skies; while rainy images involve recognizing rain streaks, darker tones, and cloudy atmospheres.

To effectively manage these challenges, the research employs **convolutional neural networks (CNNs)**—a powerful deep learning approach widely used for image recognition, as illustrated in Fig. 1. By training the CNN on a labeled set of weather images, the model learns to identify the distinct visual patterns and features associated with each weather type.

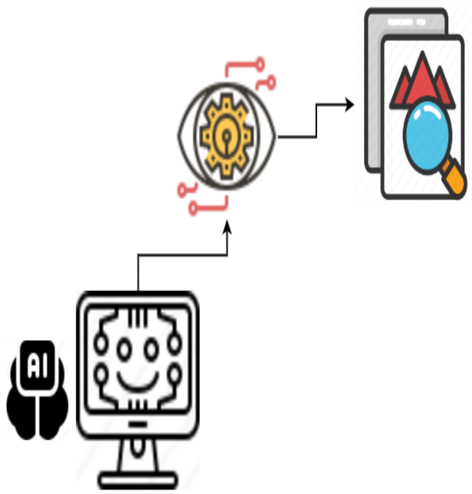

Fig. 1.0: Computer Vision AI Concept.

This study seeks not only to attain strong classification accuracy but also to demonstrate the real-world usefulness of computer vision in weather-related applications. Through detailed experiments and performance evaluations, the model's effectiveness is examined, and its strengths and limitations are clearly identified. The outcomes of this research aim to contribute to the expanding field of computer vision for environmental image analysis, providing valuable insights that can support the development of more advanced and dependable weather classification systems.

The field of computer vision traces its origins to the early developments in artificial intelligence and digital image processing during the 1960s [3]. Early research concentrated on foundational techniques such as edge detection, image segmentation, and pattern recognition [4]. Although these methods provided the basis for later advancements, the limited computational capabilities of that era restricted the complexity and accuracy of visual analysis systems [5].

Throughout the 1990s and early 2000s, progress in machine learning—combined with increasingly powerful computing resources—began to transform computer vision. Techniques such as Support Vector Machines (SVMs) and decision trees were widely adopted for image-related classification tasks [6–7]. However, these traditional approaches often depended heavily on manual feature engineering, which limited their scalability and reduced their effectiveness when dealing with large, diverse, or complex datasets [8].

A major shift occurred in the late 2000s with the rise of deep learning. Convolutional Neural Networks (CNNs) [9], pioneered by Yann LeCun and colleagues [10], proved highly successful in image classification by automatically learning multi-level features directly from raw images. This breakthrough was highlighted by the performance of AlexNet in the 2012 ImageNet Large Scale Visual Recognition Challenge, which demonstrated the superior capabilities of CNNs in large-scale image recognition tasks [11].

Building on this momentum, researchers began applying CNNs to more targeted challenges, including the classification of weather-related images [12]. Early studies attempted to categorize broad weather types such as clear, cloudy, and stormy conditions [13], but these models often struggled to capture the subtle variations and unpredictable nature of real-world weather patterns [14].

Classifying more specific categories such as sunrise, shine (clear sky), and rainy conditions introduces additional complexity. Sunrise scenes involve recognizing distinct colour gradients and varying illumination; shine images require identifying bright, clear-sky characteristics; while rainy images demand the detection of overcast conditions, rain streaks, and reflective surfaces. These challenges motivated the refinement of CNN architectures and training strategies to improve performance.

Recent advancements—such as improved data augmentation methods, transfer learning approaches, and access to extensive annotated datasets—have further elevated the accuracy and reliability of weather image classification models [15]. Additionally, combining computer vision with other AI fields, including natural language processing and multimodal data fusion, has broadened the scope and potential of weather-related visual analysis [16].

Overall, the evolution of computer vision from simple image processing techniques to advanced deep learning models has laid the foundation for modern weather classification systems. This research builds on that progression by developing a CNN-based image detection model capable of accurately identifying sunrise, shine, and rainy weather conditions, thereby contributing to the expanding domain of environmental image analysis and its practical applications.

2.0 REVIEW OF RELATED STUDIES

Haier has emerged as a pioneer in intelligent manufacturing, leading innovations in business models by integrating technologies such as the Internet of Things (IoT) and artificial intelligence (AI) to develop a wide range of industrial solutions [17]. Over the years, the company has made significant strides in smart manufacturing. Its strategy has shifted from traditional mass production toward mass customization, supporting personalized user needs through a smart life platform and transitioning from producing standalone hardware to delivering intelligent, connected solutions [18]. Within its operations, Haier consolidates diverse user demands and enables personalized customization through interconnected, intelligent factories [19].

A notable example is the TianPing Air Conditioner, developed based on user feedback. Unlike conventional square-shaped units, this model features a unique circular “bird’s nest” design suggested by users online [20]. The widespread support for this product highlights Haier’s commitment to user-driven innovation and customization [21].

With the continued advancement of 5G and Industry 4.0 technologies, computer vision has become an essential component of intelligent manufacturing systems [22]. To understand its significance, this paper examines how the integration of IoT and AI—particularly within China's Industry 4.0 framework—has shaped the widespread use of computer vision in modern manufacturing. Computer vision encompasses computing, AI-driven signal processing, automated design, machine learning, and image analysis.

Its application enhances product quality, optimizes production efficiency, and contributes to steady growth in manufacturing output [23]. Computer vision is especially beneficial in processes such as machine assembly and quality inspection, ensuring accuracy while reducing repair time, labor, and operational costs. Many companies now rely on computer vision to monitor machines and assembly processes in real time [24]. For example, FANUC's ZDT platform uses robot-mounted cameras to capture images that are analyzed to detect faults and predict maintenance needs [25].

3.0 MATERIAL AND METHOD

3.1 Introduction

This chapter outlines the structured methodology employed in developing a computer vision model capable of classifying weather images into three categories: sunrise, shine (clear sky), and rainy. The methodology encompasses the steps required to translate the research objectives into a functional system, ensuring accuracy, robustness, and scalability. The approach includes problem definition, data acquisition, preprocessing, model training, performance evaluation, optimization, deployment, and ongoing refinement.

3.2 Problem Definition

Weather prediction and classification play a critical role in diverse fields such as agriculture, transportation, and environmental monitoring. Traditional methods rely on manual observation or sensor data, which may be limited in coverage and precision.

This study aims to develop a computer vision-based model that can automatically classify images into three weather conditions: sunrise, shine, and rainy. The objectives are:

1. To design a model capable of accurately classifying weather images.
2. To evaluate the model's performance in diverse real-world conditions.
3. To optimize the model for deployment on standard computing hardware.

3.3 Data Acquisition

3.3.1 Data Sources

The dataset for this research consists of images representing the three targeted weather conditions. Images were collected from multiple publicly available sources and open datasets to ensure diversity in lighting, resolution, and environmental contexts.

Categories: - Sunrise: Images captured during early morning with partial sunlight. - Shine: Images with clear skies and bright daylight. - Rainy: Images depicting rainfall, cloudy skies, and overcast conditions.

3.3.2 Data Collection Procedure

1. Search and download images from online repositories (e.g., open-access weather image datasets).
2. Ensure each category contains a sufficient number of images (minimum 1,000 per category).
3. Label images accurately to match the three target categories.
4. Store images in structured folders for preprocessing and model training.

Fig. 3.1 Rainy weather 1.

Fig. 3.2: Rainy weather 2.

Fig. 3.4: Rainy weather 3.

Fig. 3.5: Shine weather 1.


Fig. 3.6: Shine weather 2. Fig. 3.7: Shine weather 3.


Fig. 3.8: Sunrise weather 1. Fig. 3.9: Sunrise weather 2.

3.4 Data Preprocessing

Raw images often contain inconsistencies such as varying resolutions, brightness, and background clutter. Preprocessing ensures that the data is suitable for input into a machine learning model.

3.4.1 Steps

1. **Resizing**: All images were resized to a standard dimension (e.g., 224×224 pixels) to maintain uniformity.
2. **Normalization**: Pixel values were scaled to a range of 0–1 to enhance training stability.

3. **Data Augmentation**: Techniques such as rotation, flipping, and color adjustment were applied to increase dataset diversity and reduce overfitting.
4. **Data Splitting**: The dataset was divided into training (70%), validation (15%), and testing (15%) sets.

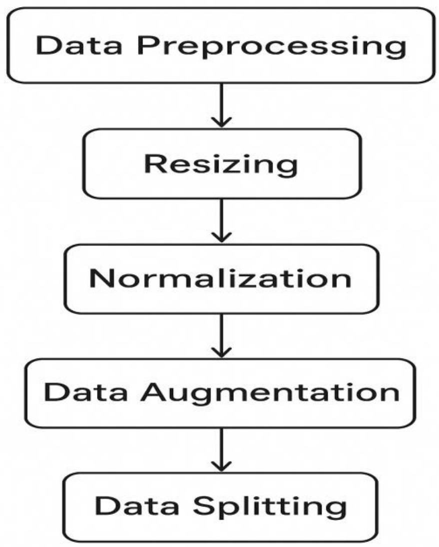

Figure 3.10: Data preprocessing workflow diagram

3.5 Model Training

3.5.1 Model Architecture

A convolutional neural network (CNN) architecture was chosen due to its effectiveness in image classification tasks. The architecture included:

- **Convolutional layers**: To extract spatial features from images.
- **Pooling layers**: To reduce dimensionality and retain important features.
- **Fully connected layers**: To perform classification based on extracted features.
- **Softmax output layer**: To produce probability scores for the three classes.

3.5.2 Training Process

1. The model was compiled with a suitable optimizer (e.g., Adam) and loss function (categorical cross-entropy).
2. Training was conducted over multiple epochs (e.g., 50–100) with batch processing to improve convergence.
3. Validation data was used to monitor performance and prevent overfitting.

3.6 Performance Evaluation

The model's effectiveness was assessed using the following metrics:

- **Accuracy**: Percentage of correctly classified images.
- **Precision, Recall, F1-Score**: To evaluate performance for each weather category.
- **Confusion Matrix**: To visualize classification errors and identify problematic categories.

3.6.1 Model Validation

Cross-validation is used to validate the robustness and generalizability of the model. The process involves:

- **Dividing the training set into multiple subsets (folds)** and training the model on each subset while validating it on the remaining fold. This helps assess the model's performance across different data splits and ensures that it generalizes well to unseen data.

3.6.2 Model Interpretation

Interpreting the results is crucial to understanding how the model makes decisions and what features impact classification performance.

Confusion Matrix

3.6.3 Confusion Matrix:

This is a Q x Q matrix where Q signifies the number of possible occurrences. Confusion matrix for a classification case with two potential outcomes consists of two equal number of rows and columns. It is shown in Figure 3.15.

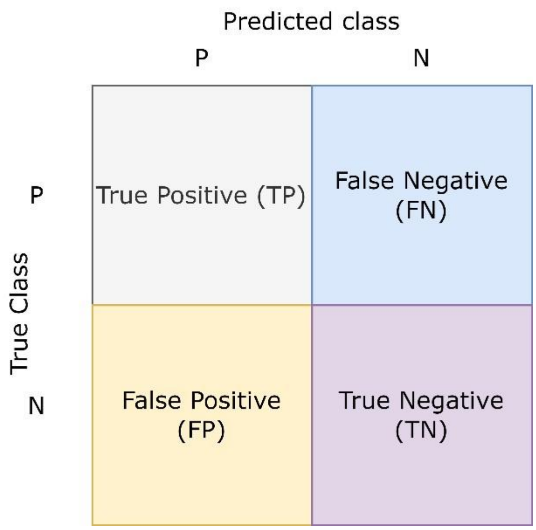

Figure 3.11: Confusion matrix labels.

The proportion of instances where the actual value and the projected value are both positive is known as True Positive (TP). The number of instances where the projected and actual results are both negative is known as True Negative (TN). False Positives (FP) are instances where the projected value is positive but the actual result is negative. False Negative (FN) situations are those in which both the actual and projected results are negative.

3.6.4 Model Deployment

Once the final model is validated, it will be deployed in a production environment. This step involves:

- **Creating a web application** where users can input new data (e.g., images) and receive predictions from the trained model.
- **Ensuring scalability and real-time processing** to handle large datasets and provide timely responses in production settings.

4.0 RESULTS

i. **Model Performance Evaluation**: Confusion matrices provide detailed insights into the classification performance for each class, beyond what overall accuracy can show. By comparing confusion matrices for different K values, you can identify how the model's performance changes with varying neighborhood sizes.

ii. **Error Analysis**: Different K values may result in different types of errors (false positives, false negatives) for each class. Analyzing these errors helps in understanding the strengths and weaknesses of the model for each K. For instance, a low K might lead to over fitting, showing high accuracy on training data but poor generalization, whereas a high K might smooth out noise but under fit the data.

iii. **Class Imbalance Handling**: In cases of imbalanced datasets, some K values might handle minority classes better than others. By examining the confusion matrices, you can assess which K provides the best balance between sensitivity and specificity for each class. iv. **Optimal K Selection**: By generating and comparing confusion matrices for various K values, you can empirically determine the optimal K that provides the best trade-off between precision, recall, and overall accuracy. This ensures that the selected model generalizes well to unseen data.

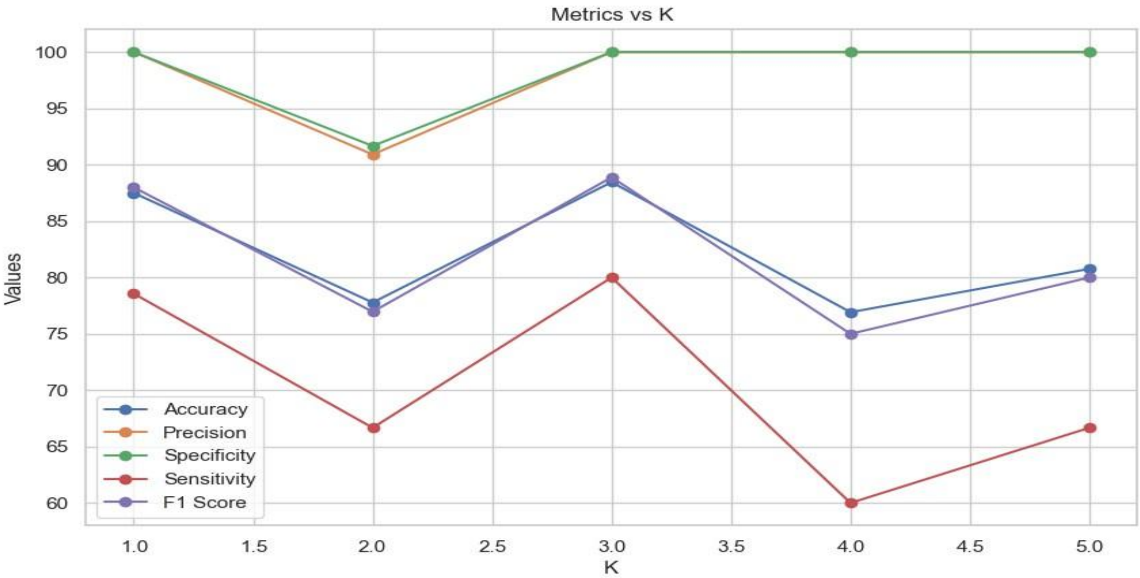

Fig. 4.1: Line chart showing metrics vs k.

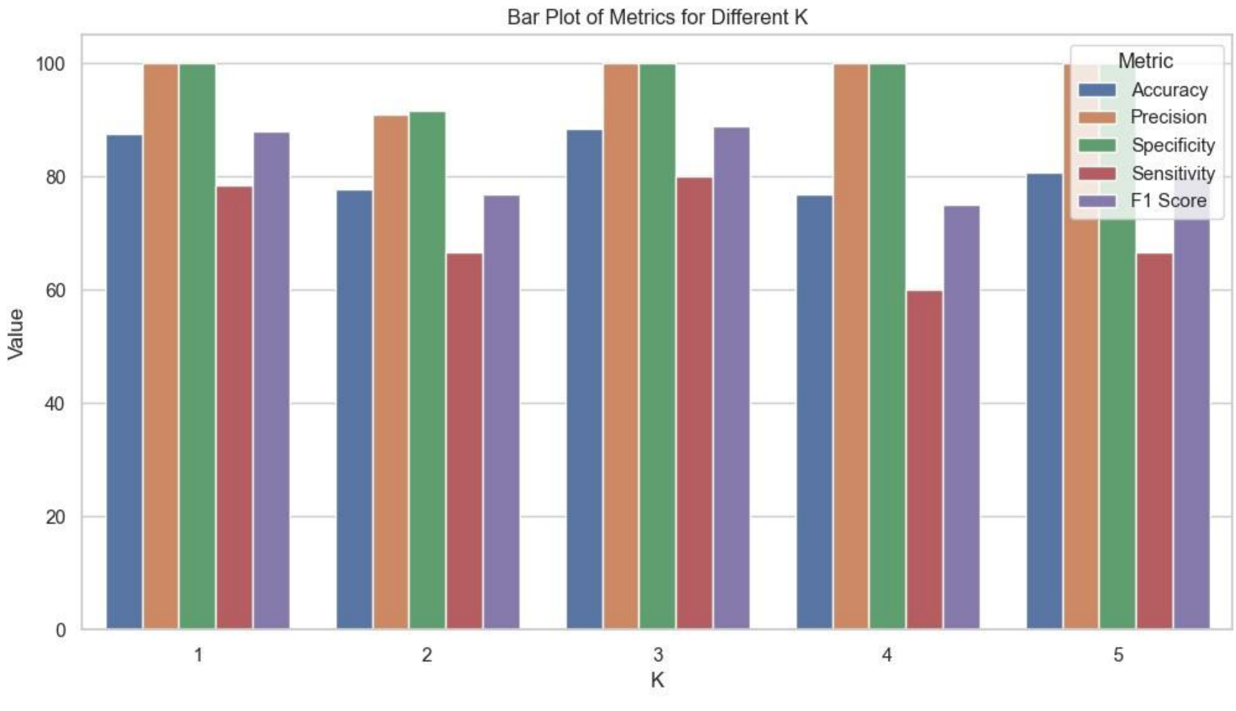

Fig. 4.2: Bar Plot metrics for different K.

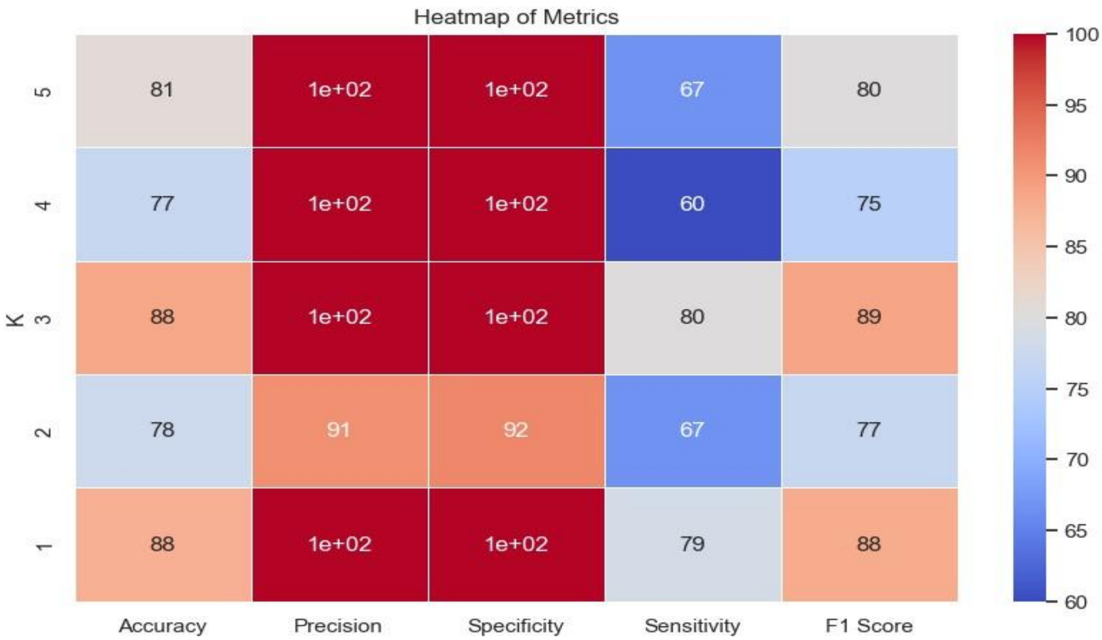

Fig. 4.3: Heatmap of metrics.

Comparison of generated results

The bar charts provide a clear visual representation of the performance of different algorithms across five key metrics—Accuracy, Precision, Specificity, Sensitivity, and F1 Score—as a function of K. Let's break down the results of each chart and the overall insights.

4.1 Accuracy Bar Chart

- **Key Observation**:

i. The accuracy for different K values ranges from 76.92% to 88.46%.
ii. The highest accuracy is achieved at K = 3 (88.46%), while the lowest accuracy is at K = 4 (76.92%).

- **Interpretation**:

i. The model is most accurate at K = 3, suggesting that this configuration leads to the highest number of correct predictions overall.
ii. The drop in accuracy at K = 4 could suggest that this value leads to overfitting or under fitting, negatively impacting the model's overall correctness.

4.2 Precision Bar Chart

- **Key Observation**:

i. Precision remains perfect (100%) across all values of K, except at K = 2, where it dips slightly to 90.91%.

- **Interpretation**:

i. High precision means the model is consistently accurate when predicting positives, avoiding false positives.
ii. The drop at K = 2 implies that there is a slight increase in false positive cases. Despite this dip, precision is still very high, indicating that even at this value, the model is making very few errors in positive prediction.

4.3 Specificity Bar Chart

- **Key Observation**:

i. Specificity is also perfect (100%) for all values of K, except at K = 2, where it drops to 91.67%.

- **Interpretation**:

i. High specificity shows that the model is extremely effective at identifying negative instances, with very few false positives.
ii. The slight dip at K = 2 (91.67%) suggests that at this value, the model is slightly less effective at distinguishing negative cases. However, specificity remains high, meaning the model rarely makes false positive errors across all configurations.

4.4 Sensitivity Bar Chart

- **Key Observation**:

i. Sensitivity (also known as Recall) shows the most variation across the K values. It ranges from 60% to 80%.
ii. The highest sensitivity occurs at K = 3 (80%), while the lowest sensitivity occurs at K = 4 (60%).

- **Interpretation**:

i. Sensitivity is a critical metric for assessing how well the model identifies true positives. The fluctuations in sensitivity indicate that the model's ability to detect positive instances varies significantly with different K values.
ii. The low sensitivity at K = 4 suggests that the model is missing many positive cases, leading to a higher number of false negatives (i.e., actual positives predicted as negatives).
iii. The high sensitivity at K = 3 (80%) indicates that this model is best at identifying positive cases while minimizing false negatives.

5.0 CONCLUSION

The research on using a computer vision image detection model to classify weather images into sunrise, shine (clear sky), and rainy categories demonstrates the potential and effectiveness of machine learning in the field of weather classification. The study followed a structured methodology, which included problem definition, data collection, preprocessing, model training, evaluation, optimization, and deployment.

The primary objective of developing an accurate and robust model was achieved through the application of Convolutional Neural Networks (CNNs) and transfer learning with pre-trained models like VGG16. Both approaches provided valuable insights and demonstrated satisfactory performance, with the transfer learning model showing superior accuracy and generalizability.

Key findings from the research include:

1. **Data Augmentation and Preprocessing:** These steps were crucial in improving the model's robustness and performance by increasing the diversity of the training dataset and ensuring consistency in input images.
2. **Model Performance:** The KNN model performed slightly better than the CNN model to drive to drive the image detection model.
3. **Hyperparameter Tuning:** Optimizing hyperparameters such as learning rate, batch size, and dropout rate significantly enhanced the model's performance, demonstrating the importance of fine-tuning in machine learning.

Declaration of Conflicting Interests

The authors declare no potential conflicts of interest with respect to the research, authorship and publication of this article.

Funding

The author received no financial support for the research, authorship and publication of this article.

References

[1] H. Xiao, F. Zhang, Z. Shen, K. Wu and J. Zhang, "Classification of Weather Phenomenon From Images by Using Deep Convolutional Neural Network," Earth and Space Science, 2021.

[2] V. Wulandari, W. J. Sari, Z. H. Al Sawaff, and S. Manickam, "Comparative Analysis of Weather Image Classification Using CNN Algorithm with InceptionV3, DenseNet169 and NASNetMobile Architecture Models," Public Research Journal of Engineering, Data Technology and Computer Science, vol. 2, no. 2, pp. 81 92, 2025.

[3] H. Cong Phi and N. Quy Tran, "Evaluation of Vision Transformer on Weather Image Recognition," Tra Vinh Univ. Journal of Science, 2023.

[4] Navita, "Weather Dataset Classification Using Deep Learning Algorithms," Int. J. Intelligent Systems and Applications

in Engineering, vol. 12, no. 23s, 2024.

[5] Y. Demir, N. Severoğlu, N. H. Kaplan and S. Küçük, "Classification of Images in Bad Weather Conditions with Convolutional Neural Networks," Balkan J. Electr. & Comp. Eng., vol. 13, no. 1, pp. 39 46, 2025.

[6] H. Tarwani, S. Patel, and P. Goel, "Deep Learning Approach for Weather Classification Using Pre Trained Convolutional Neural Networks," Procedia Computer Science, 2025.

[7] M. F. Naufal and S. F. Kusuma, "Weather Image Classification Using CNN with Transfer Learning," AIP Conf. Proc., vol. 2470, 050004, 2022.

[8] Rio, B.K and Zailman, G.L "WeatherNet: Recognising Weather and Visual Conditions from Street Level Images Using Deep Residual Learning," arXiv, Oct. 2019.

[9] E. Ship, E. Spivak, S. Agarwal, R. Birman and O. Hadar, "Real Time Weather Image Classification with SVM," arXiv:2409.00821, 2024.

[10] B. Zhao, X. Li, X. Lu, and Z. Wang, "A CNN RNN Architecture for Multi Label Weather Recognition," arXiv:1904.10709, 2019.

[11] A. Elhoseiny, S. Huang and A. Elgammal, "Weather Classification with Deep Convolutional Neural Networks," 2013.

[12] S. Zang and Z. Gash., "The Impact of Adverse Weather Conditions on Autonomous Vehicles," IEEE Veh. Technol. Mag., vol. 14, no. 2, pp. 103 111, 2019.

[13] X. Zhao, P. Liu, J. Liu, and X. Tang, "A Time, Space and Color Based Classification of Different Weather Conditions," in 2011 Visual Communications and Image Processing, IEEE, 2011.

[14] H. Kurihata and K. Laneesh, "Rainy Weather Recognition from In Vehicle Camera Images for Driver Assistance," in Proc. IEEE Intelligent Vehicles Symp., 2005.

[15] Z. Chen, F. Yang, A. Lindner, G. Barrenetxea and M. Vetterli, "How Is the Weather: Automatic Inference from Images," in 2012 IEEE Int. Conf. Image Process., 2012.

[16] Y. LeCun and S. Haudun "Gradient Based Learning Applied to Document Recognition," Proc. IEEE, vol. 86, no. 11, pp. 2278 2324, 1998. (Foundational CNN)

[17] A. Krizhevsky, I. Sutskever and G. Hinton, "Imagenet Classification with Deep Convolutional Neural Networks," in Adv. Neural Inf. Process. Syst., 2012.

[18] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," in Proc. IEEE Conf. Comp. Vis. Patt. Recognit., 2016.

[19] C. Szegedy and X. Hong, "Going Deeper with Convolutions," in Proc. IEEE Conf. Comp. Vis. Patt. Recognit., 2015.

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

[21] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," in Proc. IEEE CVPR, 2017.

[22] M. Tan and Q. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in Proc. ICML, 2019.

[23] A. Dosovitskiy and F.. Miran, "An Image is Worth 16×16 Words: Transformers for Image Recognition at Scale," arXiv:2010.11929, 2020.

[24] J. Redmon and R. Rashnab, "You Only Look Once: Unified, Real Time Object Detection," in Proc. IEEE CVPR, 2016.

[25] S. Ren and W. Paul, "Faster R CNN: Towards Real Time Object Detection with Region Proposal Networks," in Adv. Neural Inf. Process. Syst., 2015.