International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 31, No. 4, April, 2026, pp. 1323-1331
Journal homepage: http://iieta.org/journals/isi

Weather Image Classification Using a Modified CNN with EfficientNetB3

Nada Abdulkareem Hameed

Computer Science and Information Technology Department, Al-Mansour University College, Baghdad 10067, Iraq

Corresponding Author Email: nada.abdulkarim@muc.edu.iq

Copyright: ©2026 The author. This article is published by IIETA and is licensed under the CC BY 4.0 license (http://creativecommons.org/licenses/by/4.0/).

https://doi.org/10.18280/isi.310426

Received: 3 January 2026
Revised: 10 March 2026
Accepted: 20 April 2026
Available online: 30 April 2026

Keywords:
deep learning, Convolutional Neural Network, AI-based learning, convolution-based deep learning model

ABSTRACT

Identifying weather conditions from images plays a crucial role in computer vision and environmental information systems, particularly due to its practical applications in areas such as climate monitoring, intelligent transportation, and emergency disaster response. This study introduces an innovative Convolutional Neural Network (CNN) architecture built upon EfficientNetB3 for multi-class weather image classification. In addition to leveraging EfficientNetB3 as the primary feature extractor, the proposed approach incorporates batch normalization, a fully connected layer, and dropout in the final classification layer to enhance training stability and mitigate overfitting. These advancements align with the rapid progress in artificial intelligence. The dataset encompasses eleven distinct weather conditions. including dew frost, hail, lightning, rain, rainbow, rime, sandstorm, and snow. Additionally, all images were standardized to a resolution of 224 × 224 pixels and were subsequently divided into training, testing, and validation subsets with a distribution ratio of 60:20:20. After running a series of experiments, the suggested new model almost perfectly performed with a training accuracy of 0.9984, testing accuracy of 0.9345, and validation accuracy of 0.9375. Additionally, the macro averages for precision, recall, and F1-score were all 0.94, indicating that the model was consistent and reliable across various weather categories. Individually, some weather events like dew hail, lightning, and rainbows have been recognized with very high accuracy, whereas frost glaze, rime, rain, and snow have exhibited comparatively lower recognition performance. This is mainly because Truth is the last group with very similar visual characteristics. The experimental results demonstrate that the adapted EfficientNetB3 architecture offers a reliable and resource-efficient approach to classifying weather images within environmental monitoring systems.

1. INTRODUCTION

Classifying weather from images is an increasingly significant task within computer vision and environmental surveillance, as it facilitates the identification of weather conditions such as sunny, cloudy, rainy, and snowy. It is very useful for various applications such as intelligent transportation systems, disaster management, and climate monitoring [1].

The rapid advancement of artificial intelligence has led to significant progress, with deep learning techniques, especially Convolutional Neural Networks (CNNs), demonstrating exceptional performance in image classification tasks. This is mainly due to their capability of automatically detecting important features in images [2]. More recently, classification with Convolutional Neural Network (CNN)-based models in recent researches has attained a higher level of accuracy compared to learning techniques traditional machine., Yet, typical CNN models usually have their own demerits like consuming a lot of computation resources and yielding worse results when the data consists of complex and very different weather patterns. Researchers have developed advanced architectures, such as the EfficientNet layer, which improves performance by employing compound scaling across network depth, width, and input resolution [3].

EfficientNetB3 is best known for its capability to draw out features very well at the same time keeping a very high-level computational efficiency. Hence, this paper puts forward an altered CNN model building on top of EfficientNetB3 to raise the precision and resilience of weather image classification. Our suggested model is designed to address the problems of the existing methods and improve the reliability of the classification of weather images in real-life situations. Machine learning and neural learning techniques, including neural networks (NN), decision trees (DT), and random forests (RF), are widely used in weather forecasting. Recent studies aim to improve prediction accuracy and disaster preparedness. Islam et al. [4] proposed a hybrid machine learning model for predicting weather conditions, including rainfall, wind speed, and temperature based on data analysis [5]. The researchers presented a new approach to forecasting the weather based on an intelligent learning method. The model has the ability to adapt to change, including weather features such as temperature, wind flow, and humidity. Data was used that depends on the target area by taking advantage of information acquired through various sources. The model showed high accuracy of rainfall in multiple locations. The model successfully forecasted the precipitation very precisely even in areas with very different weather conditions. Rahman et al. [6] proposed a weather forecasting model based on machine learning techniques applied to the data set of twenty years from one weather station. The study contrasted the performance of Artificial Neural Networks (ANNs), RF, and K-Nearest Neighbor (KNN) algorithms. Multiple research papers have examined the application of machine learning and deep learning methods to weather forecasting and classification. Singh et al. [7] put forward several machine learning techniques such as Support Vector Machine (SVM), Artificial Neural Network (ANN), and Recurrent Neural Network (RNN]), among which Recurrent Deep Learning models that work on time series data were found to outperform others as they could effectively learn temporal dependencies. Nevertheless, the downside of their method was the high complexity of the model and limitations of the dataset, Papadimitriou et al. [8] presented a CNN-based technique for multiclass weather image classification, which resulted in a high accuracy of 98% using data augmentation methods. On the other hand, the model is highly dependent on the size of the dataset and needs further tuning for better generalization. In this paper, we introduce a CNN-based architecture primarily based on EfficientNetB3, which has been modified for the purpose of weather image classification. One of the key contributions of this research is that the authors made a tailor-made incorporation of EfficientNetB3 into a CNN architecture, which not only elevated the feature extraction ability and classification accuracy but at the same time decreased computational complexity, thus enhancing the model's effectiveness and its applicability in real-world weather classification scenarios. In order to remove such limitations, this article puts forward a novel CNN design based on EfficientNetB3 for the classification of weather images. Below is a short summary of the key contributions of this work:

1-Suggest a modified CNN based on EfficientNetB3: The paper presents a tailored deep learning model that combines EfficientNetB3 with extra layers to further enhance the capability of feature extraction for classifying weather images.

2-Novel integration of EfficientNetB3: In contrast to conventional methods, EfficientNetB3 is not just used but thoroughly reworked and incorporated into a CNN model to significantly boost the ability of the network to learn features and thus improve the classification results.

3-Improved classification accuracy: The proposed method outperforms the other methodologies in terms of accuracy as it is capable of learning the most making features from the complicated weather image patterns.

4-Reduced computational complexity: The design of the network is such that the computational expense is less than that of commonly used computational learning networks. Thus, our model is more efficient during both training and inference.

5-Better generalization ability: Our network is capable of taking care of weather condition changes, for example, changes in lighting, different seasons, as well as visually similar classes.

6-Suitability for real-world applications: The proposed solution is not only effective but also light and scalable, which makes it a viable option for live weather monitoring and classification system.

Weather classification, from the viewpoint of information systems, should be seen as a single computational task but rather as a vital component of meteorological information systems that assist in decision-making. To deliver real-time analysis, enhance situational awareness, and guide automated services like early warning issuance, emergency response, and resource management, these systems rely on precise and efficient data processing models.

2. LETERATURE REVIEW

Currently, more and more focus is put on the application of artificial intelligence and machine learning for environmental monitoring and managing natural disasters. These works emphasize the crucial role of data-oriented methods in not only enhancing the accuracy of predictions and the quality of decision-making but also in dealing with real-world complexities effectively. Tang et al. [1] put forward a machine learning-guided system for predicting medium- and long-term precipitation using data augmentation techniques. Their method showed that supplementing datasets can greatly enhance prediction accuracy. Though, it is mainly about the rainfall forecasting part and its direct use for weather image classification is limited.

Albahri et al. [2] conducted a systematic review of trustworthy artificial intelligence applications in natural disasters. Their study emphasized the importance of reliability, transparency, and robustness in AI systems used for disaster management. Although the review provides valuable insights into AI safety and trustworthiness, it does not specifically address weather image classification or intelligent learning architectures for visual data analysis. Similarly, Al Shafian and Hu [3] reviewed the integration of machine learning and remote sensing for post-disaster building damage assessment. Their work highlights the effectiveness of combining ML with remote sensing data for disaster analysis. However, the focus is primarily on structural damage detection rather than weather image classification, and it does not explore CNN-based intelligent learning models in detail.

Islam et al. [4] introduced a domain-adaptive machine learning approach for solar power prediction using remote sensing data. Their model showed an amazing ability to adapt to different locations without using source data. Though, the study did not consider weather classification from images.

Islam et al. [5] developed a location-agnostic Neural learning model for rainfall prediction. Their method aims at enhancing geographical adaptability in different regions without the necessity of local training data. Despite the fact that the model exhibits excellent performance for precipitation forecasting, it is mainly concentrated on numerical prediction and does not deal with image-based weather classification issues.

Rahman et al. [6] performed a comparative analysis of various machine learning models for weather forecasting. They found that the success of machine learning algorithms is mostly influenced by the nature of the dataset and the features chosen. Nevertheless, the research points out that conventional ML methods are not only incapable of dealing with complicated nonlinear weather patterns, but they are also less robust compared to computational learning approaches.

Singh et al. [7] proposed a computational learning approach for the classification of biomedical signals and proved the potential of CNN-based architectures for feature extraction from complicated data. As a result to this study, it has been scientifically demonstrated that AI-driven methods are extremely capable of identifying very complex patterns, even though this analysis was not focused on weather-related problems., which is a key component of image-based weather classification. Papadimitriou et al. [8] created a CNN-based model for classifying weather images, which led to a high accuracy by using very deep convolutional features. Nevertheless, the accuracy of their model is largely reliant on the quality of the dataset, and it does not consider efficiency optimization for real-time applications. Mittal and Sangwan [9] presented a deep neural network system for classifying weather images on a large scale. Although they managed to design a highly scalable and accurate method, this was also reliant on the availability of huge datasets and expensive computational resources. Gad and Hosahalli [10] compared different machine learning algorithms on weather data sets and found that although traditional ML methods may deliver acceptable results, they are less capable than intelligent learning methods when it comes to recognizing complex patterns. Jaseena and Kovoor [11] provided a comprehensive survey of intelligent weather forecasting models, highlighting that although many techniques achieve high predictive accuracy, most lack stability analysis and real-world robustness evaluation. Shrivastava et al. [12], Nagaraj and Kumar [13], and Lee et al. [14] explored computational learning models such as DNN, CNN-LSTM, and multi-scale architectures for temperature prediction. These studies have confirmed the effectiveness of using artificial intelligence-powered methods to grasp tricky and nonlinear weather patterns., but they also revealed challenges such as high computational cost and limited generalization across different regions. Shen et al. [15] proposed a multi-scale CNN-LSTM-attention model that improved prediction accuracy significantly. However, the complexity of hybrid architectures makes them less suitable for lightweight or real-time systems. Gupta and Goel [16] utilized deep reinforcement learning for weather image classification by employing pre-trained models, and managed to achieve a high level of accuracy. Similarly, Li and Luo [17] presented a ViT-based technique coupled with attention mechanisms, which further increased the classification performance. These tests demonstrate the effectiveness of techniques based on artificial intelligence in classifying complex and non-linear weather patterns. Moreover, Salim et al. [18], Ding et al. [19], and Riad et al. [20] have been working on CNN improvements with kernel optimization, stride learning, and structural design modifications, among other topics. In essence, these research works reveal the significance of CNN architectural optimization in enhancing feature extraction and efficiency, an aspect that is indeed helping to refine weather image classification models Classification of weather and their forecasting have gained more methods recently. Exact features of those methods, i.e., both pros and cons, can be reviewed by small tabulation in Table 1. Besides that, it serves as nice visual support for placing current methods on a map clearly. The comparison covers the area, the problem tackled, the main findings, the advantages, and the disadvantages of the methods.

This comparison highlights that existing methods suffer from a trade-off between accuracy, computational cost, and model complexity, motivating the need for more efficient architectures such as EfficientNetB. According to some latest investigations, architecturally advanced networks like Vision Transformers (ViT), ResNet, and DenseNet are capable of delivering excellent accuracy in classifying weather images. Nevertheless, these models usually consume a large amount of computation and demand extensive training datasets. EfficientNetB3, on the other hand, offers a better balance between accuracy and efficiency. It matches the performance level of other models; however, it has far fewer parameters and requires less computation. Therefore, it is a more appropriate option for real-time applications and resource-limited environments.

Table 1. Comparative analysis of existing methods for weather image classification and forecasting

Ref.	Method	Key Result	Limitation
[8]	Conventional Based Model	High accuracy in weather image classification	Sensitive to dataset size
[9]	Transfer Learning	Good performance with faster training	High computational cost
[10]	Machine Learning Models	Simple and acceptable results	Poor performance on complex data
[11]	Machine and Deep Learning Models	Comprehensive analysis of forecasting methods	Lack of stability evaluation
[12]	Deep Neural Network (DNN) Models	High accuracy in temperature prediction	Limited generalization
[13]	Deep Learning Models ((Long short-Time Memory (LSTM), Gated Recurrent Unit (GRU))	Strong performance in time-series forecasting	High computational cost
[14]	CNN, LSTM, DNN	Improved forecasting using time-interval data	Performance varies by dataset
[15]	Convolutional Neural Network-Long Short Time Memory (CNN-LSTM)	High forecasting accuracy	Very complex model
[17]	Vision Transformer (ViT)	Strong classification performance	Requires high computational resources

3. RESEARCH GAP

Currently, machine learning and deep learning-based methods are regarded as the main driving factors behind success in weather image classification and forecasting. Still, the application of these very modern techniques in real-world situations is restricted by a few issues: Firstly, conventional machine learning models depend mainly on hand-crafted features. Because of this, because of this characteristic, they are incapable of detecting even slight changes in weather images. Secondly, traditional CNN models are not only highly computationally costly, but they also necessitate very large training datasets to attain high levels of classification accuracy. Transfer learning methods have shown better results; Though, they are still highly dependent on the amount and quality of available data. Besides, advanced architectures like CNN-LSTM and ViT can produce high-quality classifications, their main disadvantages being the complexity of the models and A lot higher computational costs. On top of that, most of the current methods are not able to handle the normalization to new weather conditions, illumination changes, and the changing of seasons. So, there is an imperative need for a classification model having a very high degree of accuracy, low computational complexity, and strong generalization. Still, to meet these demands, a re-designed CNN architecture that relies on EfficientNetB3 as its backbone is presented in this paper. The proposed network is not only an excellent feature extractor but also achieves higher classification accuracy and needs less computational power at the same time. That means, it is capable of better balancing the trade-offs between accuracy, efficiency, and generalization in the weather image classification problem.

4. THEORETICAL BACKGROUND

4.1 Convolutional Neural Network layer architecture

Figure 1 illustrates the architecture of a CNN. It primarily comprises several types of layers, including convolutional layers, pooling layers, and fully connected layers, which are arranged sequentially as depicted in the figure.

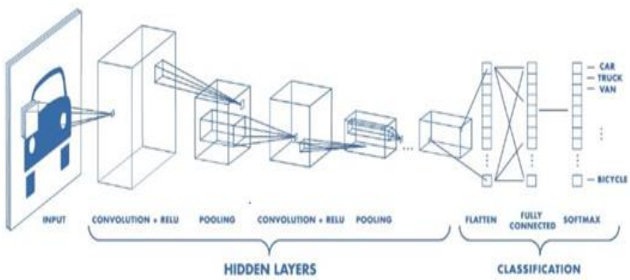

Figure 1. Architecture of Convolutional Neural Network (CNN)

4.2 Convolutional layer operation

When a computer processes an image, it essentially interprets a series of numbers organized in matrices. The smallest component of an image is called a pixel, and each pixel is assigned a numerical value that indicates its brightness level [8]. Figure 2 illustrates the pixels of an image, as explained by Adam Geitgey on Medium.

Figure 2. Image pixels

The kernel filter in convolutional layers plays a crucial role in enabling a network to identify and extract the most significant features from a given image [7]. As the filter traverses the entire image, it calculates the dot product between its elements and the corresponding image pixels at each position. This process generates the activation map, which a CNN utilizes to learn image features. The convolution operation is mathematically represented by Eq. (1), while Figure 3 illustrates the convolutional operation and the resulting activation map.

$$\text{Activation Map} = \text{Input} \times \text{Filter}$$

$$Activation\ Map(i,j) = \sum\_p \sum\_q Input(i-p, j-q) \cdot Filter(p,q) \tag{1}$$

A convolutional layer is defined by three primary parameters: the kernel size, stride length, and padding [9]. The kernel size determines the dimensions of the filter applied to the input feature map. The stride length specifies the distance the filter moves with each step as it scans the input image, affecting both the size and resolution of the resulting output feature map [10].

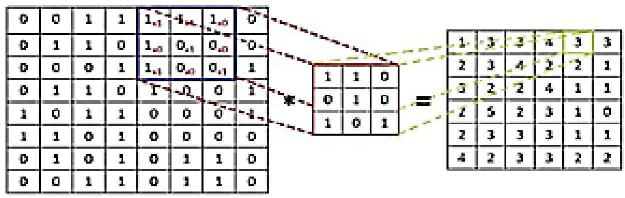

Figure 3. Convolutional operation

Padding adds zero values around the borders of the input feature map, ensuring the output feature map retains essential image details while maintaining consistent dimensions [11].

4.3 The pooling layer

The primary purpose of incorporating a pooling layer after a convolutional layer is to down sample the feature maps, thereby reducing computational complexity. Specifically, max-pooling is employed to retain significant features by selecting the maximum value within each localized region. Additionally, this process helps mitigate the risk of overfitting, making the model more robust. An illustration of max-pooling can be found in Figure 4 [12, 13].

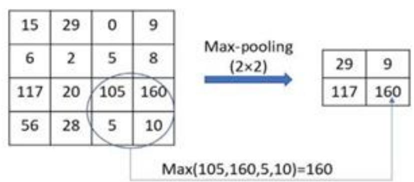

Figure 4. Pooling layer

4.4 Fully connected layer

The last part of the CNN design is the fully connected layer. That layer shapes to 1D vector the feature maps extracted from the previous layers and it also performs the classification task.

4.5 ReLU activation function

The most efficient and straightforward way to improve model training is by utilizing the ReLU activation function. ReLU is by far the most popular activation function for CNNs, largely due to its simplicity and speed of computation. Simply, it retains positive values and changes negative values to zero. So, training speeds up and the quality of outcomes is most of the time better too. ReLU operates by preserving positive values as they are, while converting all negative values to zero, making it particularly favorable for handling positive inputs. Moreover, it not only minimizes computational complexity but also expedites the overall training process of the model.

5. METHODOLOGY

Figure 5 shows the main system architecture that is being inquired. In essence, the entire process is divided into six principal steps: gathering the dataset, data initialization and data cleaning, dividing the dataset into training, validation, and testing parts, CNN model designing, model training, and model evaluation. Finally, the effectiveness of the designed model is measured using performance metrics.

5.1 Data initialization

This section includes an explanation and analysis of the dataset that is used in the proposed model, where the weather dataset is used as a data source [18]. Each image of the dataset depicts a different scene for natural phenomena like lightning, snow, and rain, and each image contains captured scenes for atmospheric conditions, as shown in Figure 6.

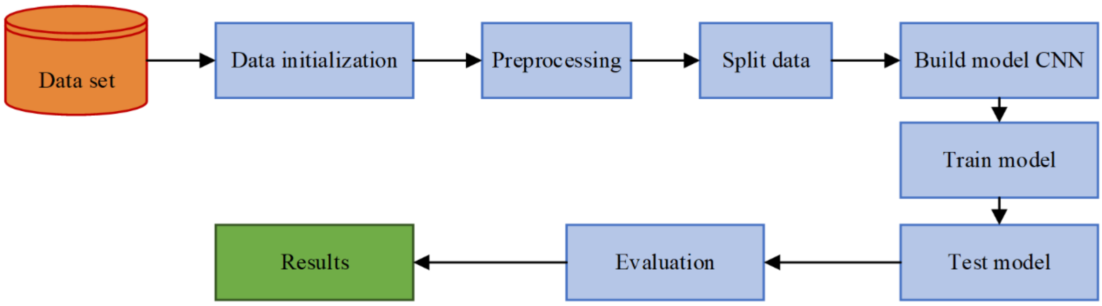

Figure 5. The proposed structure

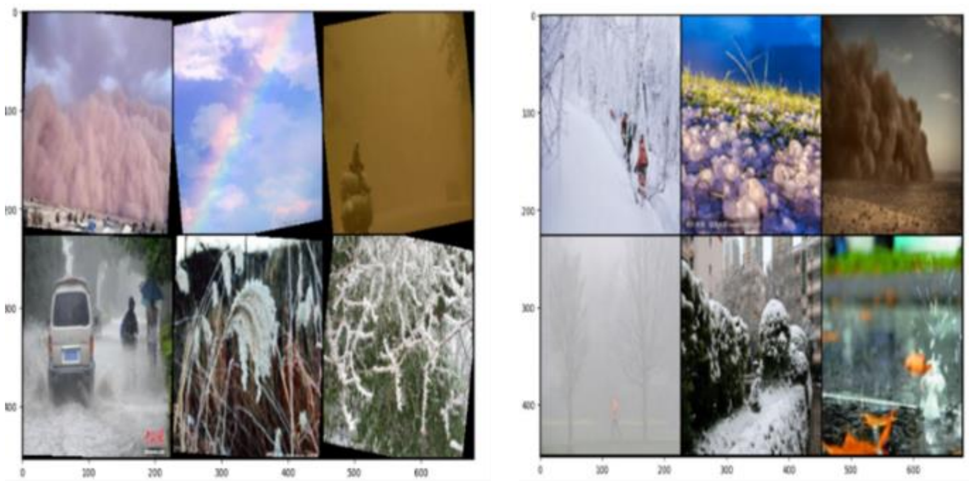

Figure 6. Dataset samples

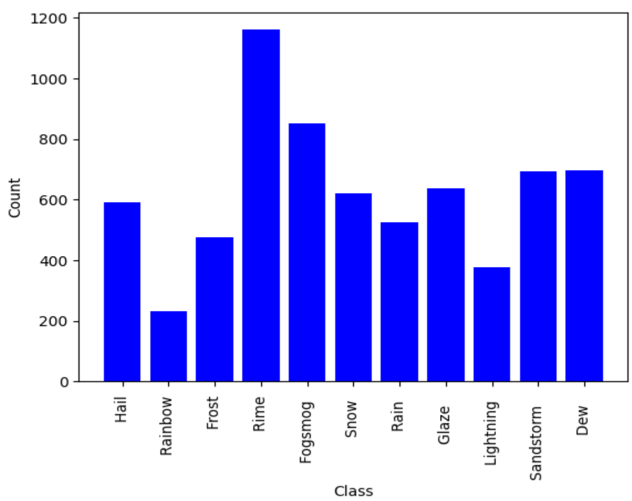

Figure 7. The count of class in dataset

First of all, the dataset comprises a total of eleven classes: Hail Rainbow Frost Rime Fog/Smog Snow Rain Glaze Lightning, Sandstorm and Dew. The breakdown of these classes and the corresponding number of images for each one are presented in Figure 7. The class that has the most pictures is rime, while the class that has the fewest pictures is rainbow [19].

5.2 Pre-processing

Preprocessing data in this manner is important to keep the data uniform and this is also one of the major factors in model training to achieve higher accuracy. For example, the images originally came in various file formats, the first step was to standardize all into a single format. In addition, the images were also all resized into the same size of 224 × 224 pixels. A few samples are displayed in Figure 8. Finally, the dataset was partitioned into training, testing and validation datasets in the ratio of 60:20:20, respectively [20].

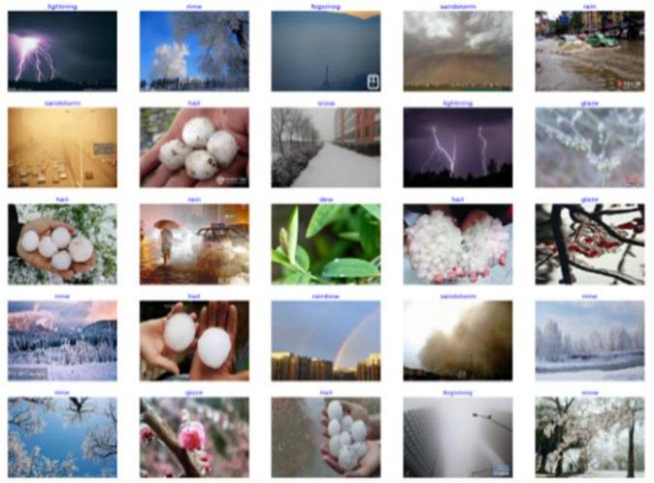

Figure 8. Samples after processing

5.3 Model structure

This paragraph includes building the proposed model using deep learning technology with the use of the EfficientNETB3 model after modifying the architecture by adding a set of normalization layers, followed by a connection layer, dropout layer, followed by a connection layer, as shown in Figure 9.

As depicted in Figure 9, the proposed model is built upon the EfficientNetB3 backbone, which is utilized for feature extraction. To enhance feature normalization and stabilize the learning process, a Batch Normalization layer is introduced immediately after the feature extraction stage. Subsequently, a fully connected (Dense) layer with 256 units is employed to capture high-level feature representations. Furthermore, a Dropout layer is incorporated prior to the final classification stage to reduce overfitting and improve the generalization capability of the model. The final classification is performed using a dense layer with 11 output neurons, corresponding to the number of target classes, followed by an activation function. For better transparency and reproducibility, Table 2 presents a comprehensive summary of the network architecture, including layer types, output dimensions, and the number of parameters. These additions clearly specify the modifications made to the baseline EfficientNetB3 model and address the limitations of the initial description.

Layer (type)	Output Shape	Param #
efficientnetb3 (Functional)	(None, 1536)	10,783,535
batch_normalization (Batch Normalization)	(None, 1536)	6,144
dense (Dense)	(None, 256)	393,472
dropout (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 11)	2,827

Total params: 11,185,978 (42.67 MB)
Trainable params: 11,095,603 (42.33 MB)
Non-trainable params: 90,375 (353.03 KB)

Figure 9. Convolutional Neural Network (CNN) algorithm structure of the proposed model

In this paper, our choice of EfficientNetB3 was based on its perfect trade-off between accuracy and computing power demands. Smaller versions (B0-B2) can only provide limited features whereas bigger versions (B4-B7) require a great deal more computing power. Furthermore, thanks to its compound scaling method, EfficientNetB3 has fewer parameters than oldies like VGG and ResNet but still delivers performance at a comparable level. More specially, the proposed model is based EfficientNetB3 as the backbone. Additionally, the Batch Normalization layer is used to normalize the output features. The output is then fed into a fully connected (Dense) layer with 256 neurons, followed by a Dropout layer with a rate of 0. 5 to protect against overfitting. At last, a Dense output layer with 11 neurons is the classification layer. In order to make things clearer and reproducible, we have inserted a new table (Table 2) near Figure 8 that outlines all the layers of the network, their output dimensions, and the number of parameters in details.

Table 2. Detailed architecture of the proposed Convolutional Neural Network (CNN) model based on EfficientNetB3

Layer No.	Layer Type	Output Shape	Parameters
1	EfficientNetB3	(None, 1536)	10,783,535
2	Batch Normalization	(None, 1536)	6,144
3	Dense (256)	(None, 256)	393,472
4	Dropout (0.5)	(None, 256)	0
5	Dense (Output = 11)	(None, 11)	2,827

6. RESULT AND TESTING

This part of the study reveals the test results from the proposed technique. The collected data are examined and the explanation of findings is aided by graphs and various performance metrics. The training and validation loss curves are shown in Figure 10. Here the red line is for training loss and green line is the validation loss. The training and validation accuracy curves are provided in Figure 11. The red line is for training accuracy and the green line is for validation accuracy. The confusion matrix in Figure 12 evaluates the proposed model comprehensively across the eleven weather classes: Dew, Fog/Smog Frost Glaze, Hail Lightning Rain, Rainbow Rime Sandstorm, and Snow. The diagonal elements of the confusion matrix stand for correctly classified weather conditions, revealing generally strong performance, with particularly high accuracy for classes like dew (70 correct out of 71), fog smog (60 perfect classifications), and rime (71 correct out of 72). However, the off-diagonal entries highlight areas where the modified CNN architecture exhibits confusion. For instance, frost is occasionally misclassified as glaze and more frequently as rime, suggesting potential similarities between these conditions. Similarly, rain instances are sometimes mistaken for dew or snow, and sandstorm predictions show some overlap with snow and rime. The model that was proposed showed high classification performance for the different categories of weather. Still, a few classes had virtually indistinguishable features which in some cases led to misclassification. Enhancements in distinguishing between these visually similar weather conditions will be the focus of the future work.

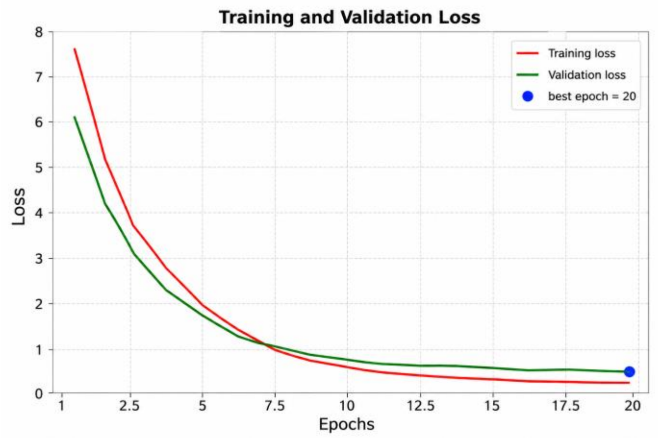

Figure 10. Training and validation loss

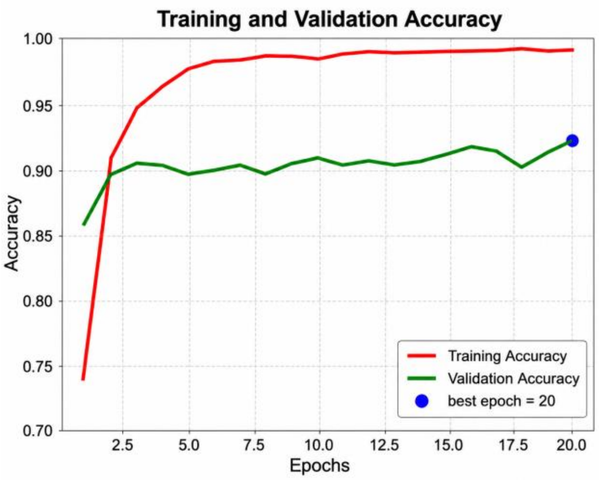

Figure 11. Training and validation accuracy

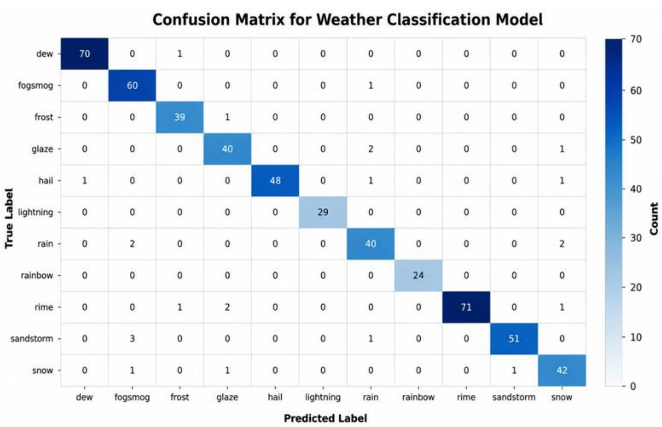

Figure 12. Confusion matrix

Table 4 is a per-class report that shows the Precision, Recall, F1-score, and Support for each category. Overall, the model demonstrates strong performance across most classes, with several achieving perfect or near-perfect scores. Notably, hail, lightning, and rain exhibit exceptional results with 1.00 Precision, Recall, and F1-score, indicating that the model flawlessly identifies these classes when they are present and rarely misclassifies other phenomena as these. Furthermore, dew also performs remarkably well with 0.99 for all three metrics, suggesting a highly reliable classification. While most other classes maintain high F1-scores above 0.90, frost, glaze, rain, rime, and snow show slightly lower, though still respectable, performance, indicating some room for minor improvement in distinguishing these particular categories.

Table 3 presents the loss and accuracy performance across various stages of the newly modified CNN model. The network demonstrates a remarkable training accuracy of 99.84% alongside a minimal loss value of 0.1814, highlighting its strong learning capability. Furthermore, the test and validation accuracies have improved to 93.45% and 93.75%, respectively, indicating that the proposed model is both robust and well-suited for generalizing to unseen data, ensuring reliable performance.

Table 3. Loss and accuracy across training, testing, and validation sets

	Loss	Accuracy
Train	0.1814	0.9984
Test	0.4332	0.9345
Validation	0.4479	0.9375

Table 4. Represents precision, recall, F1-score and support

	Precision	Recall	F1-Score	Support
Dew	0.99	0.99	0.99	71
Fog/Smog	0.91	0.98	0.94	61
Frost	0.95	0.87	0.91	45
Glaze	0.91	0.87	0.89	46
Hail	1.00	0.94	0.97	51
Lightning	1.00	1.00	1.00	29
Rain	0.89	0.91	0.90	44
Rainbow	1.00	1.00	1.00	24
Rime	0.86	0.95	0.90	75
Sandstorm	0.98	0.93	0.95	55
Snow	0.89	0.86	0.88	49

Table 5. Accuracy, macro average and weighted average

	Precision	Recall	F1-Score	Support
Accuracy			0.93	550
Macro average	0.94	0.94	0.94	550
Weighted average	0.94	0.93	0.93	550

Table 4 provides a detailed breakdown of how effectively various weather categories were classified using metrics such as precision, recall, F1-score, and support. In analyzing the results, the stationary model demonstrated strong performance, particularly in accurately classifying categories like lightning and rainbow, both achieving a perfect precision score of 1.00. Meanwhile, Fog/Smog and glaze received a precision score of 0.91, which, although slightly lower, still reflects a very high level of accuracy.

Table 5 offers an aggregated view of the model's performance, presenting overall Accuracy, Macro Average (avg), and Weighted Average (avg) for the entire dataset of 550 samples. The model achieves an impressive overall Accuracy of 0.93, signifying that 93% of all predictions were correct. Both the Macro Average and Weighted Average for Precision, Recall, and F1-score are consistently high, ranging from 0.93 to 0.94. The Macro Average, which treats all classes equally, reinforces the notion of balanced performance across the various categories, while the Weighted Average, which considers the support (number of instances) for each class, further confirms the robust overall effectiveness of the proposed model. These strong aggregate metrics, combined with the detailed per-class scores, suggest that the model is well-suited for its classification task, showing both high overall correctness and commendable performance across individual categories.

7. CONCLUSION

This study proposed a novel CNN model based on the modification of EfficientNetB3 architecture for weather image classification. It was experimentally demonstrated that the model achieved excellent performance in identifying various weather conditions. Incorporating Batch Normalization, Dense, and Dropout layers together not only stabilized the model but also enhanced its ability to generalize. The proposed architecture offered a compromise between classification accuracy and computational efficiency thereby making it a viable option for the implementation of weather monitoring. In the future, we intend to test our designed model on bigger datasets that will be more geographically diverse. Besides, we will be concentrating on adjusting the model so that it will be compatible with edge devices and mobiles for it to be used in live environment monitoring and disaster prevention systems.

REFERENCES

[1] Tang, T.T., Jiao, D.L., Chen, T., Gui, G. (2022). Medium-and long-term precipitation forecasting method based on data augmentation and machine learning algorithms. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 15: 1000-1011. https://doi.org/10.1109/JSTARS.2022.3140442

[2] Albahri, A.S., Khaleel, Y.L., Habeeb, M.A., Ismael, R.D., et al. (2024). A systematic review of trustworthy artificial intelligence applications in natural disasters. Computers and Electrical Engineering, 118: 109409. https://doi.org/10.1016/j.compeleceng.2024.109409

[3] Al Shafian, S., Hu, D. (2024). Integrating machine learning and remote sensing in disaster management: A decadal review of post-disaster building damage assessment. Buildings, 14(8): 2344. https://doi.org/10.3390/buildings14082344

[4] Islam, M.S., Hasan, A.S.M.J., Rahman, M.S., Yousuf, J., Sajol, M.S.I., Tumpa, F.A. (2023). Location agnostic source-free domain adaptive learning to predict solar power generation. In 2023 IEEE International Conference on Energy Technologies for Future Grids (ETFG), Wollongong, Australia, pp. 1-6. https://doi.org/10.1109/ETFG55873.2023.10407265

[5] Islam, M.S., Rahman, M.S., Haque, M.S.U., Tumpa, F.A., Hossain, M.S.B., Arabi, A.A. (2023). Location agnostic adaptive rain precipitation prediction using deep learning. In 2023 IEEE 9th International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE), Thiruvananthapuram, India, pp. 148-153. https://doi.org/10.1109/WIECON-ECE60392.2023.10456479

[6] Rahman, M.S., Tumpa, F.A., Islam, M.S., Al Arabi, A., Hossain, M.S.B., Haque, M.S.U. (2023). Comparative evaluation of weather forecasting using machine learning models. In 2023 26th International Conference on Computer and Information Technology (ICCIT), Cox's Bazar, Bangladesh, pp. 1-6. https://doi.org/10.1109/ICCIT60459.2023.10441077

[7] Singh, S.A., Meitei, T.G., Majumder, S. (2020). Short PCG classification based on deep learning. In Deep Learning Techniques for Biomedical and Health Informatics, pp. 141-164. https://doi.org/10.1016/B978-0-12-819061-6.00006-9

[8] Papadimitriou, O., Kanavos, A., Mylonas, P., Maragoudakis, M. (2023). Advancing weather image classification using deep convolutional neural networks. In 2023 18th International Workshop on Semantic and Social Media Adaptation & Personalization (SMAP), Limassol, Cyprus, pp. 1-6. https://doi.org/10.1109/SMAP59435.2023.10255190

[9] Mittal, S., Sangwan, O.P. (2023). Classifying weather images using deep neural networks for large scale datasets. International Journal of Advanced Computer Science and Applications, 14(1): 337-343. https://doi.org/10.14569/IJACSA.2023.0140136

[10] Gad, I., Hosahalli, D. (2022). A comparative study of prediction and classification models on NCDC weather data. International Journal of Computers and Applications, 44(5): 414-425. https://doi.org/10.1080/1206212X.2020.1766769

[11] Jaseena, K.U., Kovoor, B.C. (2022). Deterministic weather forecasting models based on intelligent predictors: A survey. Journal of King Saud University-Computer and Information Sciences, 34(6): 3393-3412. https://doi.org/10.1016/j.jksuci.2020.09.009

[12] Shrivastava, V.K., Shrivastava, A., Sharma, N., Mohanty, S.N., Pattanaik, C.R. (2023). Deep learning model for temperature prediction: A case study in New Delhi. Journal of Forecasting, 42(6): 1445-1460. https://doi.org/10.1002/for.2966

[13] Nagaraj, R., Kumar, L.S. (2023). Univariate deep learning models for prediction of daily average temperature and relative humidity: The case study of Chennai, India. Journal of Earth System Science, 132: 100. https://doi.org/10.1007/s12040-023-02122-0

[14] Lee, S., Lee, Y.S., Son, Y. (2020). Forecasting daily temperatures with different time interval data using deep neural networks. Applied Sciences, 10(5): 1609. https://doi.org/10.3390/app10051609

[15] Shen, J.J., Wu, W.Y., Xu, Q.Y. (2024). Accurate prediction of temperature indicators in eastern China using a multi-scale CNN-LSTM-attention model. arXiv preprint arXiv:2412.07997. https://doi.org/10.48550/arXiv.2412.07997

[16] Gupta, A., Goel, S. (2023). Deep learning empowered weather image classification for accurate analysis. In 2023 IEEE 2nd International Conference on Industrial Electronics: Developments & Applications (ICIDeA), Imphal, India, pp. 567-572. https://doi.org/10.1109/ICIDeA59866.2023.10295216

[17] Li, J., Luo, X.P. (2023). A study of weather-image classification combining VIT and a dual enhanced-attention module. Electronics, 12(5): 1213. https://doi.org/10.3390/electronics12051213

[18] Salim, S.S.K., Msallam, M.M., Olewi, H.I. (2024). Design novel CNN architecture to protect personal devices from unauthorized access using face recognition. Nanotechnology Perceptions, 20(3): 82-93. https://doi.org/10.62441/nano-ntp.v20i3.7

[19] Ding, C., Li, Y., Xia, Y., Zhang, L., Zhang, Y.N. (2018).

Automatic kernel size determination for deep neural networks based hyperspectral image classification. Remote Sensing, 10(3): 415. https://doi.org/10.3390/rs10030415

[20] Riad, R., Teboul, O., Grangier, D., Zeghidour, N. (2022). Learning strides in convolutional neural networks. arXiv preprint arXiv:2202.01653. https://doi.org/10.48550/arXiv.2202.01653