BALKAN JOURNAL OF ELECTRICAL & COMPUTER ENGINEERING, Vol. 13, No. 1, March 2025

Research Article

Classification of Images in Bad Weather Conditions with Convolutional Neural Networks

Yasin Demir, Nagihan Severoglu, Nur Huseyin Kaplan, Sefa Kucuk

***Abstract*—Weather conditions are one of the major factors significantly influencing the daily lives of individuals. Unfavorable weather conditions adversely affect their lives and directly impede the progress of the subsequent image-processing steps necessary for real-world vision tasks such as object detection and autonomous driving. For this reason, the correct classification of the weather conditions is of great importance. Although traditional classification methods achieve high accuracy in various tasks, they cannot achieve the same success in classifying weather conditions. In this paper, we propose a novel convolutional neural network (CNN) framework for the classification of weather conditions with high accuracy. The proposed network outperforms the existing methods with 95.50% accuracy for a classification problem with five different scenarios.**

***Index Terms*—Multi-class classification, deep learning, convolutional neural networks, weather classification**

I. Introduction

WEATHER conditions influence outdoor imaging systems, leading to low-contrast and reduced image visibility. It can also directly affect the operation of many real-world visual systems, such as autonomous vehicles, intelligent driver assistance systems, and outdoor video analysis. Most research in computer vision is based on the assumption that the weather is clear in the processed images. However, one of the most critical issues in developing these systems is their poor performance in adverse weather conditions such as rain, snow, fog, and haze [1], [2]. Therefore, weather classification applications have great importance in providing more reliable and better visibility of imagery.

Over the past decades, the authors have generally focused on the single-class weather recognition problem in which they try to determine whether an image belongs to a particular category or not [3], [4], [5], [6]. It is a fact that single-class classification tasks are unable to provide a comprehensive description of weather conditions. On the other hand, some of them deal with two-class recognition problems, e.g., sunny and cloudy weather [7], [8]. To the best of our knowledge, despite its numerous application areas, a limited number of multi-class weather classification studies [9], [10] have been carried out.

With the rapid development of machine learning, learning-based models have been widely used in classification problems. Furthermore, collecting large data sets has become more accessible in recent years, owing to the progress in image acquisition systems and their increased accessibility. These developments have facilitated the training process of learning-based networks. Weather classification is a multi-class classification problem for which numerous learning-based methods have been proposed for such problems. Generally speaking, learning-based approaches are roughly divided into four main groups [11]. The first group, supervised learning methods, utilize information from labeled training data to predict output classes [12]. The second category, unsupervised learning methods, is applied when the training data lacks labeling. In such cases, these methods classify based on certain features and inferences from the available data [13]. The third group, semi-supervised learning methods, are between supervised and unsupervised techniques as they use a combination of labeled and unlabelled data for training, and the amount of unlabelled data is higher in this techniques [14]. The last category, reinforcement learning methods, estimates the consequences of system actions in environments that lead from one situation to another through rewards and punishments scheme [15]. These learning-based methods provide significant advantages in terms of flexibility in a wide range of applications.

Weather conditions classification holds crucial importance for various computer vision applications in outdoor surveillance systems, robotic vision, and driver assistance systems, to name a few. It can play a vital role in deciding which pre-processing steps to execute for an application. For example, computer vision processes suffer from hazy environments, and dehazing methods deal with these circumstances to improve the visual quality of images. Detecting a hazy environment during autonomous driving will allow the proper functioning of required pre-processing steps.

In this paper, we propose an effective convolutional neural network (CNN) framework to classify images captured in adverse weather scenarios. By doing this, we attempt to address multi-class weather classification problems for the benefit of further image processing algorithms such as dehazing, defogging, and low-light image enhancement methods. To summarize, the main contributions of this paper can be summarized as follows:

- As part of this work, we have compiled a dataset from several publicly available datasets for those interested in further research on this task.
- We propose a novel multi-label CNN-based weather classification network that can accurately categorize poor weather images, including haze, snow, and rain.
- Low-light illuminations degrade the performance of vision applications. Considering this fact, we aim to extend

Yasin DEMİR, **Nagihan SEVEROGLU**, **Nur Hüseyin KAPLAN**, and **Sefa KÜÇÜK** are with the Department of Electrical and Electronics Engineering, Erzurum Technical University, Erzurum, 25050, TURKEY, e-mails: yasin.demir@erzurum.edu.tr, nagihan.severoglu@erzurum.edu.tr, huseyin.kaplan@erzurum.edu.tr, sefa.kucuk@erzurum.edu.tr.

*Corresponding author.

Manuscript received Jan 04,2024; revised Feb 17, 2025.

DOI: 10.17694/bajece.1415025

the applicability of the proposed model by adding a new category to the dataset called *low light class* for bad weather conditions.
- We have created a normal class devoid of any adverse weather conditions. By doing this, we have tried completely separating bad weather conditions from clear ones.

The rest of the paper is organized as follows. Section II gives the most common deep learning methods used specifically in image classification tasks. Section III details the proposed architecture for classifying different weather scenarios. Section IV presents the experimental results, and the conclusion is given in Section V.

II. Related work

Many deep learning (DL) models have been proposed for image classification including convolutional neural networks, dynamic Bayesian networks, autoencoders, and restricted Boltzmann machine models [16]. In this section, we first provide a brief overview of commonly used deep learning models for image classification. Then, we review the image-based weather detection and classification works in the literature.

A. Convolutional Neural Networks

Convolutional neural networks (CNNs) are one of the most widely used DL models, generally consisting of convolution layers, pooling layers, and fully connected layers. LeNet-5, a leading model, is composed of two convolutional layers, two fully connected layers, multiple pooling layers, and a Gaussian connection layer. With large-scale datasets and significant advances in computational capabilities, more advanced networks have been proposed, such as AlexNet [17], which leverages the ImageNet dataset [18]. AlexNet is structured with five convolutional layers and three subsequent fully connected layers. VGGNet [19], another inspiring model, has been proposed to achieve better performance by increasing the depth of the network while reducing the number of model parameters. It has also introduced innovations such as modular networks, smaller convolution, and multi-scale training. In contrast to previous approaches, the Network in Network (NIN) [20] model adopts a combination of multi-layer perceptron and convolution, resulting in a more complex micro-neural network structure than the traditional convolutional layers.

B. Dynamic Bayesian Networks

Bayesian Networks (BNs) play a significant role in various applications, including anomaly detection, classification, and clustering. BNs provide a better efficient representation of the joint probability distribution over a group of random variables [21]. Dynamic Bayesian Networks (DBNs), a specialized variant of BNs, recursively capture the dynamics of the system in a time-dependent fashion [22]. In a DBN, the first layer is referred to as the input layer, the middle layer is the hidden layer, and the final layer is designated as the output layer [23].

C. Autoencoders

Autoencoders serve as a technique for extracting principal components within large data distributions [24]. Due to its adaptable network structure that can be customized for various domains, it has the ability to create deeper networks. The autoencoder stands out as one of the most effective preprocessing techniques for image classification. Sparse autoencoder is a commonly used deep learning approach for automatically extracting features from unlabeled data [25]. Since deep learning applications are not robust against noisy data, pre-training with noisy data is necessary. To cope with these circumstances, denoising autoencoder structures [26] have been proposed. In the denoising autoencoder, the input is distorted by adding random noise. The model is then trained to generate predictions for the original, uncorrupted data. Deep Wavelet Autoencoder is an autoencoder architecture that has gained interest in recent years [27]. It integrates concepts from wavelet transforms into its design, employing these transforms within the network's operations to enhance its capacity for acquiring hierarchical and multi-scale data representations.

D. Restricted Boltzman Machine

Restricted Boltzmann Machine (RBM) is frequently used as a feature extractor in image classification. RBM shares parametrization with the layers of the deep belief network and is therefore considered the building block of the deep belief network. This model was first introduced under the name Harmonium [28]. Recently, many deep learning algorithms have been proposed using the RBM model [29], [30]. Model structures generally consist of two layers: the visible layer and the hidden layer. While RBMs may not effectively represent certain distributions, they demonstrate the capability to represent any discrete distribution when an adequate number of hidden layers are employed [31]. This characteristic renders RBMs one of the most suitable types of deep networks for feature extraction in unsupervised learning applications.

E. Image-based Weather Detection and Classification

Weather classification involves the identification of different weather conditions from a single image, which is a challenging task due to the diversity of weather phenomena and the lack of distinctive features. Early attempts are based on handcrafted features and traditional machine-learning techniques. For example, Kurihata $et\ al.$ defined raindrop characteristics using image features from principal component analysis (PCA) that represent the essential characters of raindrops [4]. Roser and Moosmann proposed a classification method on single color images based on Support Vector Machines (SVM) using some features of the image such as contrast, minimum brightness, sharpness, and color [7]; The authors of [5] proposed an algorithm based on Real Adaboost that combines three features: histogram of gradient amplitude (HGA), histogram of HSV color space and road information.

These works are pioneers in the task of weather classification, but can only recognize rainy weather, while applications are limited due to fixed target scenes. Over the last decades,

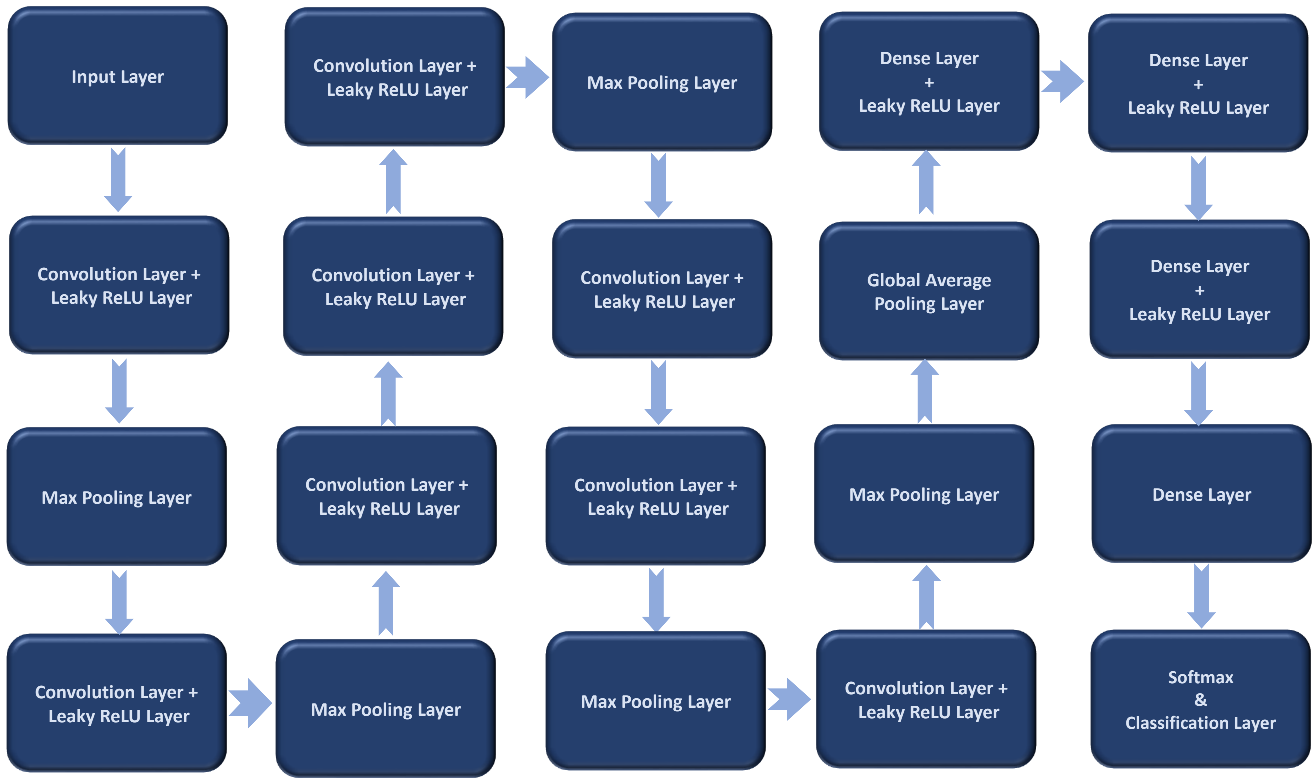


Fig. 1. Illustration of the proposed method.

researchers have focused on two-class weather classification to estimate weather conditions from images. For example; the authors of [32] use a correlation between weather and scene illumination to categorize classes; the authors of [8] proposed a collaborative learning framework that uses some specific image features such as sky, shadow, and haze to identify images as sunny or cloudy; the authors of [6] used the bag of words method for feature extraction from images and multiple kernel learning (MKL) to classify images into three class; Elhoseiny $et\ al.$ introduced a CNN architecture for two-class weather classification [33]; Kang $et\ al.$ [34] utilized a deep learning-based method to classify weather images into one of four classes: hazy, rainy, snowy, and other. They have achieved better performance than traditional methods by using the GoogleNet deep CNN model. Zhao and Wu [35] proposed a weather forecast classifier for four classes (rainy, snowy, sunny, and foggy) using the CNN method to extract high-dimensional features from images. The images were preprocessed with Mask R-CNN to improve classification performance.

III. The Proposed Framework

Here, we give the details of the proposed network. CNNs are widely used in applications such as object recognition [36] and classification [37], predominantly owing to their superior classification accuracy. This paper proposes a CNN-based structure as shown in Fig. 1.

A. Convolution layer

The main purpose of the layer is to obtain the filtered image by moving the filters of certain sizes over the entire image. The dimensions of these filters are generally chosen as 3×3, 5×5, and 7×7. This process results in an output image with higher-level features in a hierarchical manner. We set the filter sizes to 3×3 and 5×5 in convolution layers.

B. Pooling layer

Reducing the number of hyper-parameters is crucial to prevent the model from memorizing the training data and to alleviate the computational burden. To do this, CNNs utilize pooling layers. Similar to the convolution operation, the pooling process is also carried out via specific filters. These filters execute maximum, minimum, and averaging operations with certain window sizes. In the proposed model, we set the size of the pooling layers as 2×2.

C. Leaky ReLU

Activation functions play a crucial role in introducing non-linearities to CNNs. Different activation functions, including Linear, Tanh, ReLU, Swish, and Leaky ReLU (LReLU), have been employed in numerous studies. To investigate the influence of activation functions on the proposed model, we employ different activation functions and present the corresponding training accuracy in Table I. As seen from Table, we have achieved the highest training accuracy with LReLU. Thus, we choose LReLU as the activation function. LReLU, a variant of

TABLE I
ABLATION STUDY ON ACTIVATION FUNCTIONS

Activation function	Training accuracy(%)
Linear	93.12
Tanh	93.44
ReLU	96.18
Swish	96.88
LReLU	**97.69**

the classical ReLU, also has a small slope for negative inputs. LReLU addresses the issue of neurons struggling to learn after entering the negative range, thanks to its small slope. It is also known that although it is slower than classical ReLU, it demonstrates better performance [38]. Mathematically, LReLU is defined as

$$f(x) = \begin{cases} 0.01x & x < 0 \\ x & x \geq 0 \end{cases} \tag{1}$$

and visually it looks as shown in Fig. 2.

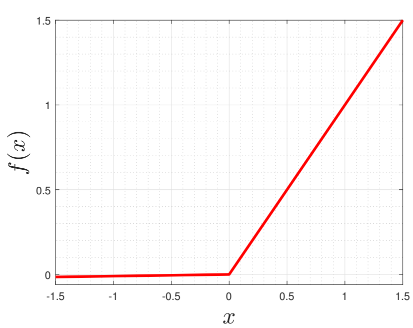


Fig. 2. Graph of the LReLU activation function.

D. Fully connected layer

The fully connected layer utilizes a weight matrix for each neuron to perform a linear transformation to the input. Through this process, all possible connections between layers ensure that the entries of the input influence every entry of the output. Typically, the fully connected layer is used at the end of CNN models to optimize classification scores.

E. Classification layer

As its name implies, the classification layer is used in DL networks for classification tasks. The output dimension of the classification layer is equivalent to the number of classified objects. Among the classifiers used in CNNs, *softmax* is the most commonly used and highly effective one. Softmax generates a probability output value within the range of 0 to 1 for each object to be classified. Softmax assigns the output as the class of the item for a probability value close to 1.

TABLE II
DETAILS OF THE CONSTRUCTED DATASET

Image class	# of image	Dataset
Hazy	340	[39]-[40]
Rainy	381	[41]-[42]
Snowy	430	[43]
Low-light	444	[44]
Normal	500	[45]
Test	200	40 of each class

IV. EXPERIMENTS

In this section, we first provide the procedure for creating the dataset and then give the implementation details. Finally, we present the evaluation metrics and experimental results in turn.

A. Dataset

For the multi-class weather classification experiment, we identified five different weather-related attributes: hazy, rainy, snowy, low-light, and normal. We choose images for individual classes from various datasets that are publicly accessible. To enhance the effectiveness of the proposed model, we introduce challenging scene images to diversify the dataset. For example, the model should recognize an image taken in snowy conditions as snowy weather and classify an image containing snow but not snowfall as representing normal weather. In doing so, we allow for the examination of weather conditions that affect the visibility of objects in the scene. Considering these conditions, we have constructed the dataset as follows:

- *Hazy images:* We select the images for this class from the O-Haze [39] dataset and the hazy weather [40] dataset of road images captured in foggy weather.
- *Rainy images:* Images are chosen from the datasets given in [41] and [42].
- *Snowy images:* We pick snowy images from the Snow100k [43].
- *Low-light images:* We randomly select low-light images from the ExDark [44], which contains images taken in various low-light conditions.
- *Normal images:* Finally, normal images without any adverse weather conditions have been chosen from the Part2 Subset [45].

When determining images within each class from the datasets, we consider the presence of only one feature in a single image. For example, we disregard hazy information present in a snowy image. Detailed information on the dataset is presented in Table II. The last row of Table II corresponds to the number of test images, which includes 40 randomly selected images for each class.

B. Implementation results

It is common knowledge that fine-tuning the hyperparameters is crucial for achieving high performance in DL methods. We have conducted several experiments to demonstrate the

TABLE III
ABLATION STUDY ON MODEL TRAINING WITH K-FOLD

# of fold	Average accuracy (%)
5	89
10	91.5

validity of the proposed CNN model. The test set has been created from unseen images not used in the training phase. We set the batch size to 32, and the number of epochs to 200. With these parameters, we use the k-fold cross-validation technique to evaluate the performance of the proposed model. We set k to 5 and 10, and have obtained the results given in Table III. We have achieved a training accuracy of 97.88% and a test accuracy of 95.50% using the model parameters that provide the highest classification accuracy among the k-folding results. Moreover, we present the confusion matrix in Fig. 3 to evaluate the performance of the proposed model on the test set. As can be seen from Fig. 3, the proposed model demonstrates a classification performance exceeding 97% for all classes, except for the Hazy class. Fig. 4 illustrates a few examples where we give some failure cases from different k-fold validation results.

It is apparent from Fig. 4 that hazy weather tends to be misclassified as normal. We believe that this tendency arises from the insufficient learning of the airlight in hazy images. We note from Fig. 3 and Fig. 4 that the hazy class is the most challenging weather condition to classify among the five weather conditions.

The proposed CNN architecture was modeled using the open-source TensorFlow library in version 3.9.13 of the Python programming language [46]. Experimental operations such as training and testing were carried out on a personal computer with 11$^{\text{th}}$ Generation Intel® Core$^{\text{TM}}$ i7-11800H CPU and NVIDIA Geforce RTX 3060 GPU.

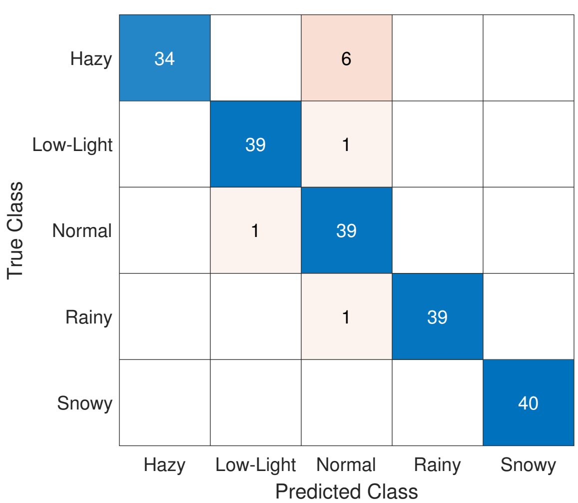


Fig. 3. Confusion matrix of five-class classification results.

C. Comparison with related methods

We have chosen four assessment criteria to quantitatively measure the performance of the compared methods. The first evaluation criterion is accuracy, which is popular in multi-class classification. Accuracy is defined as the ratio of correctly predicted data to the total amount of data and is calculated as follows.

$$\text{accuracy} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}} \tag{2}$$

Here, $\mathrm{TP}, \mathrm{TN}, \mathrm{FP},$ and FN represent true positive, true negative, false positive, and false negative, respectively. Using Eq. (2), we calculate the accuracy of the proposed model as 0.955. Although accuracy is a useful metric, it may not be sufficient when evaluating datasets characterized by uneven distributions or unbiased data. Therefore, precision has been used for a more comprehensive evaluation. It is particularly employed in scenarios where the cost of making a false positive prediction is high. Mathematically, the precision metric is computed as follows.

$$\text{precision} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{3}$$

We get the precision as 0.9646 by Eq. (3). Another assessment metric is recall, which becomes particularly crucial when the cost of predicting false negatives is substantial. The recall is defined by the following expression.

$$\text{recall} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{4}$$

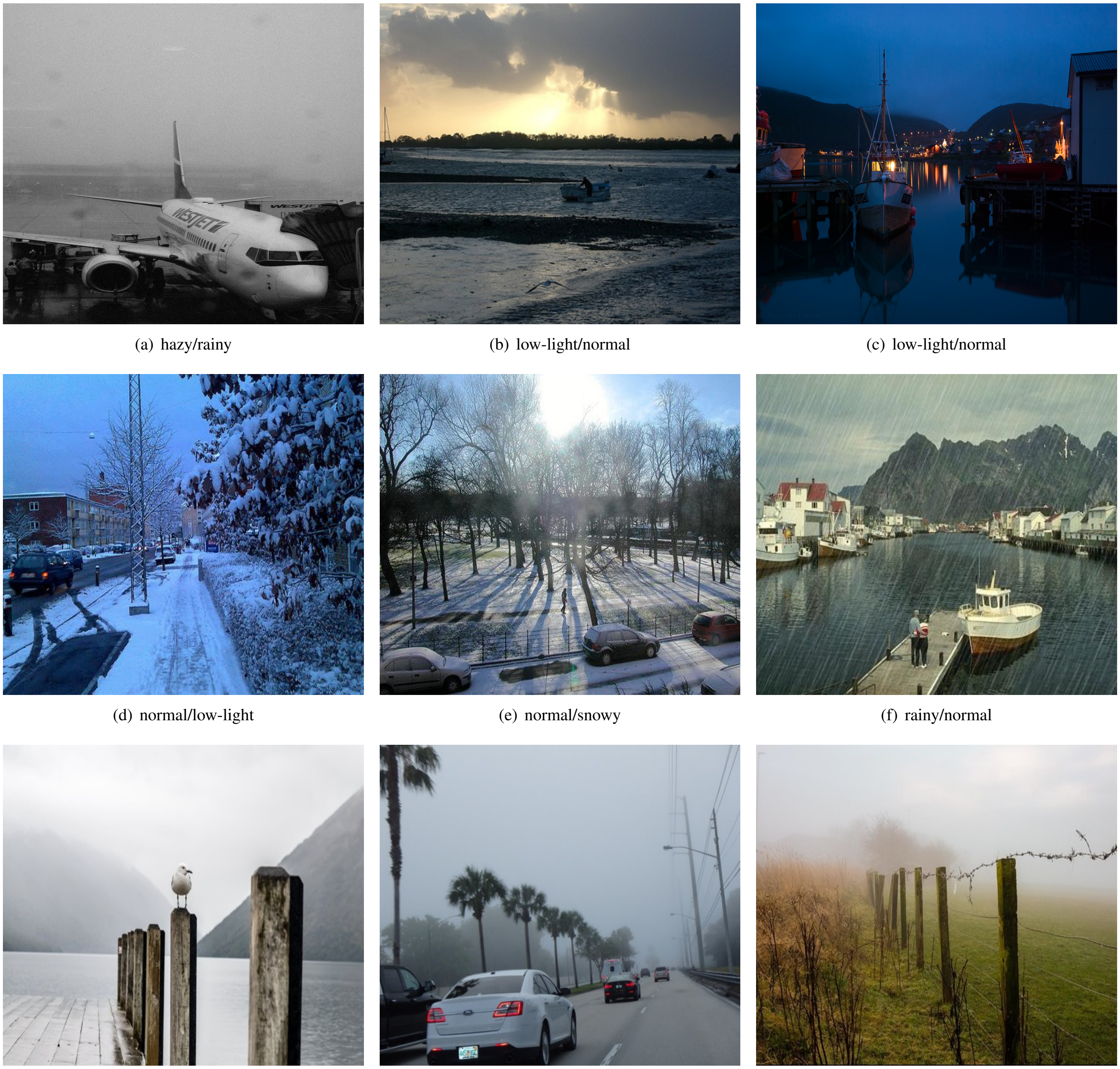
The recall value is evaluated as 0.9845. The last performance discriminator is the F1 score, which comprehensively evaluates every aspect of prediction success on a dataset, as it considers all error costs. The F_1 score [49] is calculated as in Eq. (5). The F_1 score for the proposed model is computed as 0.9744.

$$\mathrm{F}_1 = 2 \times \left(\frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}\right) \tag{5}$$

We compare the performance of the proposed method with several competing methods, including AlexNet [17], VGGNet [19], ResNet101 [33], ML-KNN [47], SRN [48] and CNN-RNN [9]. We adopt overall precision (OP), overall recall (OR), and overall F1 (OF1) as evaluation metrics, and tabulate the results in Table IV. As highlighted in Table IV, the proposed model outperforms the compared methods in all metrics, except for the rainy class. Moreover, we achieve superior performance in the newly introduced classes, namely low-light and normal.

V. CONCLUSION

In this paper, we have proposed a CNN framework for multi-label weather classification tasks. We have identified five different classes for weather conditions. To train the proposed network model, we have created a new training and test set by selecting images suitable for five classes from public datasets. On the constructed dataset, we have demonstrated the effectiveness of the proposed method and proved that it yields



(a) hazy/rainy (b) low-light/normal (c) low-light/normal

(d) normal/low-light (e) normal/snowy (f) rainy/normal

(g) hazy/normal (h) hazy/normal (i) hazy/normal

Fig. 4. Examples of some failure cases. The captions of the sub-figures represent the true class/predicted class.

TABLE IV
PRECISION/RECALL COMPARISONS WITH THE STATE-OF-THE-ART METHODS. WE HIGHLIGHT THE BEST AND THE SECOND-BEST RESULTS IN BOLD AND ITALIC, RESPECTIVELY

Method	Hazy	Snowy	Rainy	Low-light	Normal	OP	OR	OF1
AlexNet [17]	0.735/0.890	0.784/0.685	0.876/0.905	-	-	0.9007	0.8668	0.8834
VGGNet [19]	0.867/0.728	0.814/0.701	0.887/0.931	-	-	0.9087	0.8494	0.8780
ResNet101 [33]	0.841/0.855	0.776/0.882	0.947/0.938	-	-	0.8876	0.8861	0.8868
ML-KNN [47]	0.819/0.834	0.794/0.736	0.918/0.934	-	-	0.9138	0.8766	0.8948
SRN [48]	-	-	-	-	-	0.8988	0.865	0.8816
CNN-RNN [9]	0.856/0.861	0.856/0.758	0.894/0.938	-	-	*0.9263*	*0.8946*	*0.9135*
Proposed method	1/0.850	1/1	1/0.975	0.975/0.975	0.829/0.975	**0.9646**	**0.9845**	**0.9744**

quantitatively superior results compared to other competing algorithms. The proposed model, designed to predict weather conditions for specified classes, is expected to improve the utilization of further image enhancement algorithms. In this way, it will be possible to prevent applications such as object detection and target tracking against the disruptive effect of adverse weather conditions.

We are aware that the proposed framework has limitations in classifying multiple weather types within a single image. In further studies, we focus on labeling multi-weather types in a single image to provide a more comprehensive description of weather conditions.

Acknowledgment

The source code and pre-trained model, along with the constructed dataset, are available at https://spars.erzurum.edu.tr.

References

[1] S. Zang, M. Ding, D. Smith, P. Tyler, T. Rakotoarivelo, and M. A. Kaafar, "The impact of adverse weather conditions on autonomous vehicles: How rain, snow, fog, and hail affect the performance of a self-driving car," *IEEE vehicular technology magazine*, vol. 14, no. 2, pp. 103–111, 2019.

[2] 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 (VCIP)*. IEEE, 2011, pp. 1–4.

[3] Q. Li, Y. Kong, and S.-m. Xia, "A method of weather recognition based on outdoor images," in *2014 International Conference on Computer Vision Theory and Applications (VISAPP)*, vol. 2. IEEE, 2014, pp. 510–516.

[4] H. Kurihata, T. Takahashi, I. Ide, Y. Mekada, H. Murase, Y. Tamatsu, and T. Miyahara, "Rainy weather recognition from in-vehicle camera images for driver assistance," in *IEEE Proceedings. Intelligent Vehicles Symposium, 2005.* IEEE, 2005, pp. 205–210.

[5] X. Yan, Y. Luo, and X. Zheng, "Weather recognition based on images captured by vision system in vehicle," in *Advances in Neural Networks–ISNN 2009: 6th International Symposium on Neural Networks, ISNN 2009 Wuhan, China, May 26-29, 2009 Proceedings, Part III 6*. Springer, 2009, pp. 390–398.

[6] Z. Chen, F. Yang, A. Lindner, G. Barrenetxea, and M. Vetterli, "How is the weather: Automatic inference from images," in *2012 19th IEEE International conference on image processing*. IEEE, 2012, pp. 1853–1856.

[7] M. Roser and F. Moosmann, "Classification of weather situations on single color images," in *2008 IEEE intelligent vehicles symposium*. IEEE, 2008, pp. 798–803.

[8] C. Lu, D. Lin, J. Jia, and C.-K. Tang, "Two-class weather classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3718–3725.

[9] B. Zhao, X. Li, X. Lu, and Z. Wang, "A cnn–rnn architecture for multi-label weather recognition," *Neurocomputing*, vol. 322, pp. 47–57, 2018.

[10] S. Chen, T. Shu, H. Zhao, and Y. Y. Tang, "Mask-cnn-transformer for real-time multi-label weather recognition," *Knowledge-Based Systems*, vol. 278, p. 110881, 2023.

[11] R. Saravanan and P. Sujatha, "A state of art techniques on machine learning algorithms: a perspective of supervised learning approaches in data classification," in *2018 Second international conference on intelligent computing and control systems (ICICCS)*. IEEE, 2018, pp. 945–949.

[12] M. Cord and P. Cunningham, Eds., *Machine Learning Techniques for Multimedia - Case Studies on Organization and Retrieval*. Springer, 2008.

[13] M. A. T. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 24, no. 3, pp. 381–396, 2002.

[14] P. Thomas, "Review of" semi-supervised learning" by o. chapelle, b. schölkopf, and a. zien, eds. london, uk, mit press, 2006," *IEEE Transactions on Neural Networks*, vol. 20, no. 3, pp. 542–542, 2009.

[15] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *Journal of artificial intelligence research*, vol. 4, pp. 237–285, 1996.

[16] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[18] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[20] M. Lin, Q. Chen, and S. Yan, "Network in network. arxiv 2013," *arXiv preprint arXiv:1312.4400*, 2013.

[21] B. Mihaljević, C. Bielza, and P. Larrañaga, "Bayesian networks for interpretable machine learning and optimization," *Neurocomputing*, vol. 456, pp. 648–665, 2021.

[22] P. Sarshar, J. Radianti, and J. J. Gonzalez, "Modeling panic in ship fire evacuation using dynamic bayesian network," in *Third International Conference on Innovative Computing Technology (INTECH 2013)*. IEEE, 2013, pp. 301–307.

[23] P. Wang, E. Fan, and P. Wang, "Comparative analysis of image classification algorithms based on traditional machine learning and deep learning," *Pattern Recognition Letters*, vol. 141, pp. 61–67, 2021.

[24] C.-Y. Liou, W.-C. Cheng, J.-W. Liou, and D.-R. Liou, "Autoencoder for words," *Neurocomputing*, vol. 139, pp. 84–96, 2014.

[25] Y. Xiao, J. Wu, Z. Lin, and X. Zhao, "A semi-supervised deep learning method based on stacked sparse auto-encoder for cancer prediction using rna-seq data," *Computer methods and programs in biomedicine*, vol. 166, pp. 99–105, 2018.

[26] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, and L. Bottou, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion." *Journal of machine learning research*, vol. 11, no. 12, 2010.

[27] P. Kumar Mallick, S. H. Ryu, S. K. Satapathy, S. Mishra, G. N. Nguyen, and P. Tiwari, "Brain mri image classification for cancer detection using deep wavelet autoencoder-based deep neural network," *IEEE Access*, vol. 7, pp. 46 278–46 287, 2019.

[28] P. Smolensky *et al.*, "Information processing in dynamical systems: Foundations of harmony theory," 1986.

[29] L.-W. Kim, "Deepx: Deep learning accelerator for restricted boltzmann machine artificial neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 5, pp. 1441–1453, 2018.

[30] C. L. P. Chen, C.-Y. Zhang, L. Chen, and M. Gan, "Fuzzy restricted boltzmann machine for the enhancement of deep learning," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 6, pp. 2163–2173, 2015.

[31] N. Le Roux and Y. Bengio, "Representational power of restricted boltzmann machines and deep belief networks," *Neural computation*, vol. 20, no. 6, pp. 1631–1649, 2008.

[32] L. Shen and P. Tan, "Photometric stereo and weather estimation using internet images," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 1850–1857.

[33] M. Elhoseiny, S. Huang, and A. Elgammal, "Weather classification with deep convolutional neural networks," in *2015 IEEE international conference on image processing (ICIP)*. IEEE, 2015, pp. 3349–3353.

[34] L.-W. Kang, K.-L. Chou, and R.-H. Fu, "Deep learning-based weather image recognition," in *2018 International Symposium on Computer, Consumer and Control (IS3C)*. IEEE, 2018, pp. 384–387.

[35] X. Zhao and C. Wu, "Weather classification based on convolutional neural networks," in *2021 International Conference on Wireless Communications and Smart Grid (ICWCSG)*. IEEE, 2021, pp. 293–296.

[36] J. M. Gandarias, A. J. Garcia-Cerezo, and J. M. Gómez-de Gabriel, "Cnn-based methods for object recognition with high-resolution tactile sensors," *IEEE Sensors Journal*, vol. 19, no. 16, pp. 6872–6882, 2019.

[37] R. Vaddi and P. Manoharan, "Hyperspectral image classification using cnn with spectral and spatial features integration," *Infrared Physics & Technology*, vol. 107, p. 103296, 2020.

[38] J. Xu, Z. Li, B. Du, M. Zhang, and J. Liu, "Reluplex made more practical: Leaky relu," in *2020 IEEE Symposium on Computers and communications (ISCC)*. IEEE, 2020, pp. 1–7.

[39] C. O. Ancuti, C. Ancuti, R. Timofte, and C. De Vleeschouwer, "O-haze: a dehazing benchmark with real hazy and haze-free outdoor images," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 754–762.

[40] H. Halmaoui, A. Cord, and N. Hautière, "Contrast restoration of road images taken in foggy weather," in *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*. IEEE, 2011, pp. 2057–2063.

[41] G. Viyaj, "Keras-multiclass-image-classification," https://github.com/vijayg15/Keras-MultiClass-Image-Classification/tree/master, 2021.

[42] X. Fu, J. Huang, X. Ding, Y. Liao, and J. Paisley, "Clearing the skies: A deep network architecture for single-image rain removal," *IEEE Transactions on Image Processing*, vol. 26, no. 6, pp. 2944–2956, 2017.

[43] Y.-F. Liu, D.-W. Jaw, S.-C. Huang, and J.-N. Hwang, "Desnownet: Context-aware deep network for snow removal," *IEEE Transactions on Image Processing*, vol. 27, no. 6, pp. 3064–3073, 2018.

[44] Y. P. Loh and C. S. Chan, "Getting to know low-light images with the exclusively dark dataset," *Computer Vision and Image Understanding*, vol. 178, pp. 30–42, 2019.

[45] J. Cai, S. Gu, and L. Zhang, "Learning a deep single image contrast enhancer from multi-exposure images," *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 2049–2062, 2018.

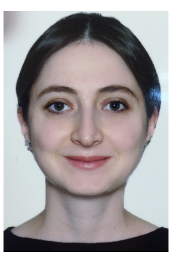
[46] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[47] M.-L. Zhang and Z.-H. Zhou, "Ml-knn: A lazy learning approach to multi-label learning," *Pattern recognition*, vol. 40, no. 7, pp. 2038–2048, 2007.

[48] F. Zhu, H. Li, W. Ouyang, N. Yu, and X. Wang, "Learning spatial regularization with image-level supervisions for multi-label image classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5513–5522.

[49] P. Flach and M. Kull, "Precision-recall-gain curves: Pr analysis done right," *Advances in neural information processing systems*, vol. 28, 2015.

Yasin DEMİR received his B. Sc., degree in Electrical and Electronics Engineering from Karadeniz Technical University, Turkey. He received his M.Sc., degree in Electrical and Electronics Engineering from Erzurum Technical University, Turkey. He is a Research Assistant at the Electrical and Electronics Engineering Department at Erzurum Technical University, Erzurum, Turkey. His primary research interests include digital signal and image processing.



Nagihan SEVEROGLU received the B.Sc. and M.Sc. degrees in Electrical and Electronics Engineering from Atatürk University, Erzurum, Turkey, in 2012 and 2014, respectively, and the Ph.D. degree in Electrical and Electronics Engineering from Gazi University, Ankara, Turkey, in 2020. Her main research interest has been biomedical engineering during her B.Sc. studies, whereas she has focused on signal processing for power system applications during her Ph.D. research. She is currently an Assistant Professor at the Department of Electrical and Electronics Engineering at Erzurum Technical University, Erzurum, Turkey. Her research interests include signal processing, deep learning techniques, and image processing.

Nur Huseyin KAPLAN received his B. Sc., M. Sc, and Ph.D degrees in Electronics and Telecommunication Engineering from Istanbul Technical University in 2005, 2008 and in 2015, respectively. He worked for TURKSAT A.Ş. as a specialist between 2005-2015. Since 2024, he is a Professor at Electrical and Electronic Engineering Department, Erzurum Technical University, Erzurum, Turkiye, where he taughts undergraduate/graduate level courses about signal and image processing. His primary research interests include digital signal and image processing.

Sefa KUCUK received the B.Sc degree from the Department of Electrical and Electronics Engineering, Atatürk University, Erzurum, Turkey, in 2012, the M.Sc and Ph.D. degrees from the Department of Electrical and Electronics Engineering, Hacettepe University, Ankara, Turkey in 2015 and 2021, respectively. He is currently an Assistant Professor with the Electrical and Electronic Engineering Department, Erzurum Technical University, Erzurum, Turkey. His research interests include computer vision, hyperspectral imaging, image processing, inverse problems, sparse theory, and machine learning.