Bias-Aware Data Quality Enhancement for Forest Fire Detection in AI-Based Remote Sensing

Vishal Krishna Singh, Soumendu Chakraborty, *Senior Member, IEEE*, Raman Singh, Rajkumar Singh Rathore, *Senior Member, IEEE*, and Weiwei Jiang, *Senior Member, IEEE*

***Abstract*—Bias, including preconceived notions embedded within the algorithm, selection in dataset collection, imbalances within the training data, impose severe restrictions on the algorithm's performance for accurate detection of forest fires. The variability of forest landscapes, environmental conditions across different geographic regions exacerbates these challenges. Considering the importance of tailored feature extraction, debiasing strategies in improving model performance for critical applications like forest fire detection, a novel approach using dual autoencoder architectures is proposed to independently learn fire-specific, nonfire-specific features. The proposed method effectively addresses visual biases in the datasets, utilizes superimposed image synthesis to create overlay images that enhance feature representation. The proposed integrated framework, validated on the publicly available *Wildfire*, *DFire* datasets, shows significant improvements in detection accuracy, robustness. The proposed approach outperforms existing methods with an *accuracy* of 80%, 76% on the *Wildfire*, *DFire* datasets, respectively, achieving lower *false positive rates*, *false negative rates* compared to the state-of-the-art methods.**

***Index Terms*—Autoencoders, bias, bias mitigation, forest fire detection, and transfer learning.**

I. Introduction

BIAS in machine learning algorithms can be a major issue especially in the case of forest fire detection as it may lead to skewed outcomes in classification tasks. The issue of bias originates from a set of complex factors, such as a preconceived notion programmed in algorithms, selection bias in datasets and imbalances of training data. Consider the case of an algorithm, which was predominantly trained using images of a certain type of forest or in a certain lighting environment. Such an algorithm will have a high false alarm rate (FAR) while detecting fire in diverse environments or lighting scenarios. Besides, algorithmic biases [1], where models replicate the constraints and assumptions of their design, restrict the models ability to make general predictions in diverse environments. Temporal biases, introduced due to the past data, hinder the ability of the model to accommodate the change in the current changes in the forest ecosystem or climatic conditions [2]. The consequences of such bias are far-reaching as there is a possibility of the underdetection of real fires, particularly the ones with weak or unusual features. These challenges are added to by the heterogeneity of forest landscapes and the environmental conditions in geographic regions; a model that is biased toward a particular set of features of a region may not work elsewhere, as it may fail to sense fires in different vegetation types or under unusual weather conditions.

Bias mitigation in image-based early forest fire detection is a widely addressed problem. One of the prominent solution suggests increasing and diversifying training data so as to be able to capture a broader range of geographical locations, environment, and time variations to directly overcome selection bias and promote the ability of the model to be generalized to forest ecosystems [3]. Also, methods like resampling [4] and reweighting are aimed at solving the problem of imbalance by increased attention to the underrepresented samples. Ensemble learning-based methods, such as in [5], utilize several models and are driven by bias mitigation in individual models. Recent years have seen increase in sophisticated deep-learning (DL) architectures, such as DenseNet201 [6], ResNet50 [6], and InceptionV3 [6]. These methods have shown steep proliferation in detection accuracy by representing the most relevant features for early fire detection and minimizing confusion with visually similar events like smoke or fog.

Existing literature on forest fire detection presents some interesting solutions aimed at improving detection accuracy in rough terrain. For example, the authors in [7], combined visible light images in the *YCrCb* color space with infrared images to outline the areas of flames. Another study in [8] presents the idea of a lightweight and real-time solution by enriching data and features within a YOLO DL model. The seminal work in [9] introduces *FireXnet*, a simplified structure aimed at effective and accurate wildfire detection; its lightweight architecture allows it to permit a high level of accuracy with minimal training and testing time. Addressing the detection and alleviation of low-level bias, the authors in [10] proposed an autoencoder to produce universal representations that can be used in models to detect key characteristics across many categories. Though this method

Received 12 December 2025; revised 15 February 2026; accepted 25 February 2026. Date of publication 3 March 2026; date of current version 20 March 2026. This work was supported by Uttar Pradesh Council of Science and Technology under the grant in-aid scheme for the project having project ID 908 and project title 'Fairness in Machine Learning for Accurate Forest Fire Inference.' *(Corresponding author: Vishal Krishna Singh.)*

Vishal Krishna Singh is with the School of Computer Science and Electronics Engineering, University of Essex, CO4 3SQ Colchester Campus, U.K. (e-mail: v.k.singh@essex.ac.uk).

Soumendu Chakraborty and Raman Singh are with the Department of Computer Science, Indian Institute of Information Technology Lucknow, Lucknow, Uttar Pradesh 226002, India.

Rajkumar Singh Rathore is with the Department of Computer Science, School of Technologies, Cardiff Metropolitan University, CF5 2YB Cardiff, U.K.

Weiwei Jiang is with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China.

Digital Object Identifier 10.1109/JSTARS.2026.3669360

deals with low-level biases, there are high-level biases that form due to complex interaction and misinterpretation of semantics, high-level biases, which is very crucial when seeking to classify data; ignoring high-level biases may impair the workable model performance. An alternate approach, proposed in [11], aimed at minimizing the FAR in autonomous aerial vehicle (AEV) detection. The method makes use of pulse-Doppler radar to identify AEVs through a dual-headed convolutional neural network (CNN) and nonmaximum suppression. Despite of significantly reduced FAR, the model could not easily adapt to new settings, raising questions on the model flexibility, adaptability, and resilience in quickly changing harsh terrain of forests.

Clearly, existing debiasing techniques can be categorized into two major groups: 1) those that demand explicit bias labels at training; and 2) those that do not demand such labels and attain debiasing. Methods, such as in [12], are based on weighted loss computation, while some methods on statistically or dynamically computed weighted samples are [13]. In addition, ensemble methods, such as in [14], and adversarial debiasing [15] are all part of explicit bias mitigation techniques. Example of implicit bias mitigation includes the restricted capacity model [16] and gradient starvation [17] models. An interesting parallel approach on distributed and intelligent learning presents a recent trend for developing bias-conscious environmental models [18]. The federated domain generalization method [19] provides improved privacy-preserving learning in heterogeneous domains; however, its effectiveness tends to decline in place of strong domain shifts and data imbalance thus generating biased generalization. The application of edge AI to Earth observation [20], enables near-source wildfire detection in the presence of sparse data and resource-scarce conditions; however, the lightweight and small-sample learning structures are usually undermined by the bias in the representation and reduced robustness. The recent SatCooper framework [21] enhances cooperative satellite inference by maximizing the tradeoff between latency and accuracy, this technique, however, depends on accurate system-state estimation and does not specifically deal with the issue of data bias or skewed distributions. In another recent work, the authors in [22] used socio-technical studies of the Agentic AI to emphasize the necessity of fairness and accountability and highlight persistent challenges in the detection and alleviation of bias in autonomous systems.

However, such approaches often fail to specify the features that proliferate bias. Most of the current methods of bias-mitigation in image-classification have average performance when dealing with bias on the higher classical level due to reduced ability to adapt to diverse settings. Accurate distinction of the complex factors of the environment, such as fog, and smoke, along with complex interaction with environmental factors, are the main challenge faced in accurate forest fire detection. These methods have limited classification accuracy and high FAR, especially when it comes to a multidimensional and multifaceted scenario of forest fire detection. Conventional approaches are unable to portray minor differences among various types of fires and their peculiarity in different settings providing false identifications and false alarms. Although the use of ensemble methods and more advanced architectures provide more flexibility, they also add complexity and computational overhead. Evidently, current approaches lack proper control over dataset imbalance and distributional bias, which then provokes the use of generative adversarial networks to generate balanced representations and efficiency, more fairness, more robustness, and generalization in forest-fire prediction and surveillance.

Building on our previous work on bias mitigation [2] and efficient in-network data processing [23], [24], we aim to overcome the drawbacks of existing techniques of bias mitigation in forest fire. We present a novel technique that combines the CNN based-DenseNet201 [25] with a new sequential autoencoder architecture. The idea is to utilize the high-level global and low-level textural information to perform an overall analysis of nonfire and fire-related features. This approach attempts to overcome the difficulties of the current models by improving specificity in the feature extraction and classification. Planned to be robust, generalized, and successful in various circumstances, it focuses on complicated relationships between multiple aspects of the environment and makes use of high-level methods of extracting and improving features. The results of the study show that there is an evident advancement in the performance, creating a new standard of accuracy and robustness of forest fire detection system, and indicating the effectiveness of the bias-mitigation strategy. Table I presents the major challenges of the existing methods and the major advantages of the proposed scheme over these methods.

Addressing the inefficiencies of existing methods to address the complicated interaction of environmental drivers and adapt to changing fire situations, the proposed work makes novel contributions to the precision of fire detection in the forest at the initial stage and can be summarized as follows.

1) We critically examine the drawbacks of state-of-the-art methods of bias mitigation in forest fire detection, particularly their inability and steep decline in detection accuracy in rough terrains and real-time dynamic scenarios.
2) We present a new DL model, combining DenseNet201 with the sequential autoencoder model. In this way, the reduction of bias is achieved by and exhaustive analysis of high-level features as well as low-level features, which, in turn, allow significant improvement in specificity and precision in detecting forest fires.
3) We employ entropy to measure the diversity and richness of information in the dataset. By increasing the dataset's entropy through our autoencoder-based feature distinction and superimposed image synthesis, we effectively reduce biases and ensure a more balanced and fair training process for the classification models.
4) Extensive experiments on two publicly available datasets is presented to assess the performance of the proposed framework. The findings reveal that our method performs better than existing state-of-the-art methods as it outperforms the existing methods in terms of accuracy, FAR, precision, recall, and a set of identified parameters.

The rest of this article is organized as follows: Section II presents a comprehensive explanation of the proposed methodology, followed by the Section III, which details the datasets used for the analysis. This is followed by Section IV, where the details

TABLE I
COMPARATIVE ANALYSIS OF AI-BASED FOREST FIRE DETECTION METHODS

Models	Complexity	Acc.	FPR	FNR	Precision	Key Challenges	Advantages of Proposed
DenseNet201 [6]	High	Good	High	Fair	Fair	1- Environmental variability, sensor-induced noise, visual bias, class imbalance	1- Dual class-specific disentanglement
ResNet50 [6]	High	Fair	High	High	Low	2- Environmental variability, residual noise propagation, contextual bias, imbalance untreated	2- Residual-guided superimposed synthesis
InceptionV3 [6]	High	Fair	Fair	Fair	Fair	3- Multiscale noise amplification, illumination bias, dataset-specific overfitting	3- Entropy-regularized feature diversity
Low-level Debiasing [10]	Medium	Fair	Fair	Fair	Low	4- Shallow decorrelation, static bias correction, imbalance unaddressed	4- Explicit overlapping-pattern mitigation
Adversarial Debiasing [26]	High	Low	High	Fair	Low	5- Training instability, gradient amplification under noise, feature suppression	5- Deterministic reconstruction-based synthesis
Proposed Method	**Medium**	**Excellent**	**Low**	**Good**	**Excellent**		6- Mixed feature-space learning

TABLE II
AUTOENCODER ARCHITECTURE SUMMARY

Component	AE_1 (Fire)	AE_2 (Nonfire)
Conv Layers	3	3
Filters	32-64-128	64-128-256
Kernel Size	3×3	3×3
Pooling	2×2 MaxPool	2×2 MaxPool
Bottleneck	$20 \times 20 \times 128$	$20 \times 20 \times 256$
Output Activation	Sigmoid	Sigmoid

of the experimental setup is presented and a thorough analysis of the results and corresponding discussions are presented in Sections V and VI, respectively. Finally, Section VII concludes this article and reports future directions.

II. PROPOSED METHODOLOGY

The proposed methodology for forest fire detection utilizes the DL architecture that addresses bias and improves the accuracy of classification. It is broken down into four major sections: 1) data preprocessing and augmentation; 2) dual autoencoder architecture to learn class-specific features; 3) superimposed image synthesis; and 4) model training and evaluation. The proposed system combines class-specific representation learning to data synthesis with the aim to minimize visual biases between fire and nonfire classes. The resultant enriched dataset is then utilized to train *DenseNet201*, which has been noted to be high-performing in image classification tasks. Dual autoencoders and superimposed image synthesis in the *DenseNet201* structure provide a higher detection rate and a steep mitigation in bias. Extensive experiments prove the effectiveness of the proposed methodology when compared to existing methods. Fig. 1 depicts the general structure of the suggested framework. The scheme illustrated in Fig. 1 shows that the pipeline starts with data preprocessing and augmentation followed by a feature extraction with the autoencoder that takes care of independent learning on the latent representation of fire and nonfire. This step is followed by using the output to generate superimposed images thus increasing the training set. The combined dataset is then employed to train the classifier resulting in bias-free prediction which improves generalization and detection reliability.

A. Data Preprocessing and Augmentation

As can be seen through the first stage of preprocessing provided in Fig. 1, the dataset is carefully systematized and structured to make it fit in the model input. The process starts with the classification of images as fire or nonfire, with the origin of both directories. Individual images are converted to RGB color space and resized to 80×80 pixels, to provide uniformity of the input dimensions ($I_{\text{resized}} = f_{\text{resize}}(I, 80 \times 80)$). To augment the dataset, techniques, such as *rotation* ($I_{\text{rot}} = f_{\text{rotate}}(I, \theta)$), *width* and *height shifts* ($I_{\text{shift}} = f_{\text{shift}}(I, \delta_x, \delta_y)$), *shear transformations* ($I_{\text{shear}} = f_{\text{shear}}(I, \phi)$), *zoom* ($I_{\text{zoom}} = f_{\text{zoom}}(I, z)$), and *horizontal flips* ($I_{\text{flip}} = f_{\text{flip}}(I)$) are employed. These artificial data enrichment enhance the dataset size and variability and alleviate overfitting and introduce simulation of a variety of real-world conditions. After *augmentation*, pixel values are scaled to the [0,1] range ($I_{\text{norm}} = \frac{I}{255}$), which gives all images the same value set to make the neural network training. Such preprocessing steps, such as *augmentation* and *normalization* are essential in developing a balanced and diverse data collection. The preprocessed data is then used for training-set enrichment in the proposed bias-aware framework, and the test set is kept isolated to prevent leakage of data in the evaluation process. After the data preprocessing is complete the augmented and normalized dataset is then fed through the proposed hybrid methodology which comprises three crucial steps: 1) *autoencoder-based feature distinction*; 2) *superimposed image synthesis creating overlay images*; and 3) *integrated feature extraction and bias mitigation*.

Algorithm 1: Proposed Forest Fire Image Classification Pipeline.

The workflow entails the stages of preprocessing, and augmentation, bias-sensitive dataset enrichment through autoencoders and superimposed synthesis, and finally, training and evaluation.

Require: Labeled dataset $\mathcal{D} = \{(X_i, y_i)\}$, $y \in \{\text{fire}, \text{nonfire}\}$

Ensure: Trained classifier and evaluation metrics

1: $\mathcal{D} \leftarrow$ PREPROCESSANDAUGMENT($\mathcal{D}$)
2: $\mathcal{D}_{\text{train}}, \mathcal{D}_{\text{test}} \leftarrow$ TRAINTESTSPLIT($\mathcal{D}, splitscale$)
3: $\mathcal{D}_{\text{train}}^{\text{enriched}} \leftarrow$ DEBIASANDSYNTHESIZE($\mathcal{D}_{\text{train}}$)
4: $model \leftarrow$ TRAINANDEVALUATECLASSIFIER($\mathcal{D}_{\text{train}}^{\text{enriched}}, \mathcal{D}_{\text{test}}$)
5: **return** $model$ and metrics

B. Dual Autoencoder Architecture for Class-Specific Feature Learning

The *autoencoder-based feature distinction* is a method that seeks to increase the differentiation ability of the classification

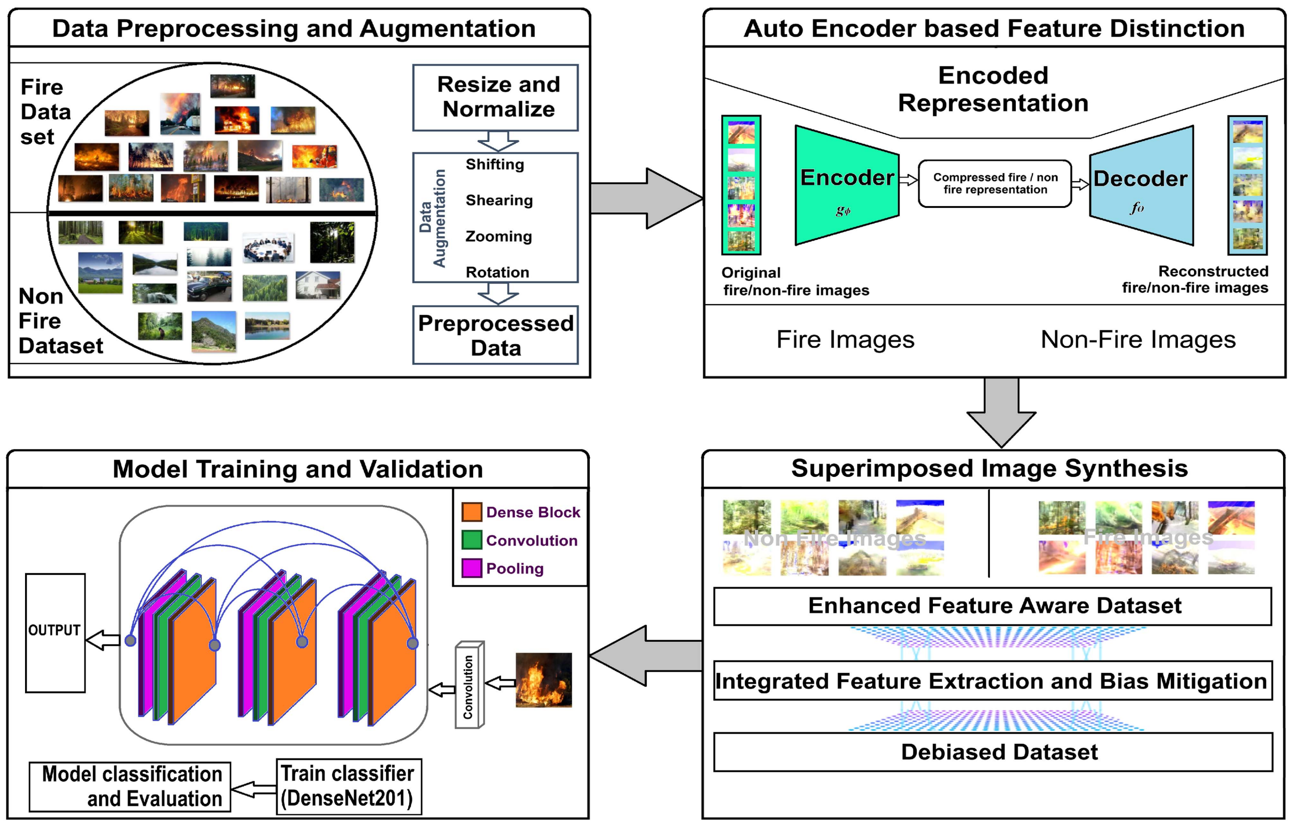


Fig. 1. Proposed model.

model, the features around the forest fire classification that may add bias to the classification process are learned separately and encoded, which includes *fog*, *smoke*, *lighting*, and *background*. This model uses two different autoencoders, namely: 1) *Autoencoder1* (fire related); and 2) *Autoencoder2* (nonfire), where *Autoencoder1* is trained on fire-related images and develops fire related features and environmental factors, whereas, *Autoencoder2* is trained on nonfire related images and learns background features and nonfire related features. It should be pointed out that no further independent autoencoders are added to $\text{Autoencoder}_{\text{fire}}$ and $\text{Autoencoder}_{\text{nonfire}}$. The debiasing phase uses the trained $\text{Autoencoder}_{\text{nonfire}}$ as a shared reconstruction block instead of using a third distinct autoencoder. The encoder consists of several *convolutional layers* with a progressively larger filter size and ReLU activation functions, and is succeeded by *max-pooling* layers to downsize spatial dimensions.

Both $\text{Autoencoder}_{\text{fire}}$ (AE_1) and $\text{Autoencoder}_{\text{nonfire}}$ (AE_2) follow a convolutional encoder–decoder structure. The encoder consists of three convolutional layers with 3×3 kernels and ReLU activation functions, each followed by batch normalization and 2×2 MaxPooling for spatial downsampling. For AE_1, the convolutional filters are 32, 64, and 128, resulting in a bottleneck feature representation of size $20 \times 20 \times 128$. For AE_2, a higher capacity configuration is used with filters 64, 128, and 256, producing a bottleneck of size $20 \times 20 \times 256$ to better capture complex nonfire background variations. The decoder mirrors the encoder structure using upsampling layers followed by convolutional layers with ReLU activation. A final 3×3 convolution with Sigmoid activation reconstructs the normalized image output. The models are trained using the Adam optimizer to minimize mean squared error (mse) loss. The architecture details are presented in Table II.

Mathematically, for an input image X, the encoding process can be expressed as follows:

$$Z = f_{\text{enc}}(X) \tag{1}$$

where Z represents the encoded feature representation. The decoder reconstructs the input from these encoded features using upsampling and convolutional layers, described by

$$\hat{X} = f_{\text{dec}}(Z) \tag{2}$$

where $\hat{X}$ is the reconstructed image. The objective of training the autoencoders is to minimize the mse between the original images and their reconstructions, which is achieved as

$$L_{\text{MSE}} = \frac{1}{n} \sum_{i=1}^{n} \left(X^{(i)} - \hat{X}^{(i)} \right)^2 \tag{3}$$

where n is the number of images, X is the original image, and $\hat{X}$ is the reconstructed image. Specifically, for *Autoencoder1* trained on fire images, the loss function is defined as follows:

$$L_{\text{fire}} = \frac{1}{n} \sum_{i=1}^{n} \left(X_{\text{fire}}^{(i)} - \hat{X}_{\text{fire}}^{(i)} \right)^2 \tag{4}$$

Algorithm 2 DebiasAndSynthesize (Dual Autoencoders + Superimposition)

This trains two class-specific Autoencoders to learn fire and nonfire representations independently. The algorithm is aimed to generate structured superimposed samples to expand feature diversity and mitigate bias.

Require: Training images $X_{\text{fire}}, X_{\text{nonfire}}$
Ensure: Enriched training set $\mathcal{D}_{\text{train}}^{\text{enriched}}$
1: Train AUTOENCODER1 on X_{fire}
2: Train AUTOENCODER2 on X_{nonfire}
3: $R_{\text{fire}} \leftarrow \text{AUTOENCODER1}(X_{\text{fire}})$
4: $R_{\text{nonfire}} \leftarrow \text{AUTOENCODER2}(X_{\text{nonfire}})$
5: $S_{\text{nonfire}} \leftarrow \text{SUPERIMPOSE}(R_{\text{fire}}, X_{\text{nonfire}})$
6: $S_{\text{fire}} \leftarrow \text{SUPERIMPOSE}(R_{\text{nonfire}}, X_{\text{fire}})$
7: $\mathcal{D}_{\text{train}}^{\text{enriched}} \leftarrow X_{\text{fire}} \cup X_{\text{nonfire}} \cup S_{\text{fire}} \cup S_{\text{nonfire}}$
8: **return** $\mathcal{D}_{\text{train}}^{\text{enriched}}$

and for *Autoencoder2* trained on nonfire images, the loss function is as follows:

$$L_{\text{nonfire}} = \frac{1}{n}\sum_{i=1}^{n}\left(X_{\text{nonfire}}^{(i)} - \hat{X}_{\text{nonfire}}^{(i)}\right)^2. \tag{5}$$

The optimization process involves adjusting the parameters of both autoencoders using the *Adam* optimizer to minimize their respective loss functions over several epochs. This process is represented as

$$\theta_{\text{fire}}^{*} = \arg\min_{\theta_{\text{fire}}} L_{\text{fire}} \tag{6}$$

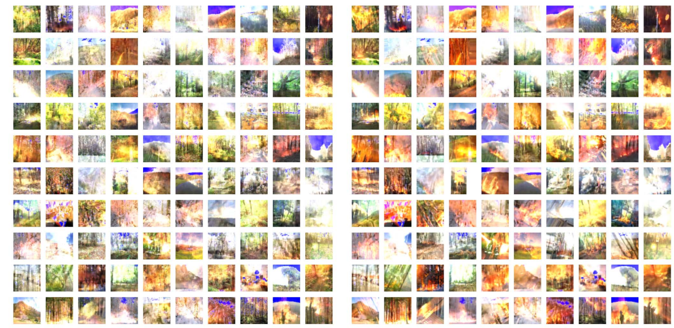
$$\theta_{\text{nonfire}}^{*} = \arg\min_{\theta_{\text{nonfire}}} L_{\text{nonfire}} \tag{7}$$

where θ_{fire} and θ_{nonfire} are the parameters of *Autoencoder1* and *Autoencoder2*, respectively. By independently training the two autoencoders, we ensure that each sutoencoder learns a compact and specialized representation of its respective class. This dual autoencoder methodology is essential for identifying and encoding biases introduced by *fog*, *smoke*, *lighting*, and *background features*, thereby enhancing the model's accuracy and robustness in forest fire detection across varied environmental conditions.

C. Superimposed Image Synthesis

The *superimposed image synthesis* attempts to improve the dataset by overprinting the reconstructed images of the fire and nonfire to expand the range of features. This is used to efficiently add extra feature space, boosting robustness of the classification model and reducing bias. The first step is the use of *Autoencoder1* to extract *fire features* and reconstruct fire-like images, denoted as R_{fire}. These reconstructed images are then overlaid on the *nonfire images* X_{nonfire} to produce synthesized nonfire images that contain fire characteristics as demonstrated in Fig. 2.

The key point to note is that R_{fire} and R_{nonfire} are learnable structured reconstructions of their autoencoders and not arbitrary pixel noise. Thus, the superimposition infuses class-relevant



(a) (b)

Fig. 2. Superimposed Image Synthesis. (a) Superimposed nonfire images. (b) Superimposed fire images.

visual-based information and does not affect the spatial consistency.

1) *Reconstruction of Fire Images:*

$$S_{\text{nonfire},i} = \text{clip}(X_{\text{nonfire},i} + R_{\text{fire},i}, 0, 1). \tag{8}$$

It should be noted that the superimposition operation uses pixelwise addition of residual generalization, and not more complicated methods of blending including alpha blending or attention-based fusion. The reason behind this decision is twofold. To begin with, the reconstructed outputs R_{fire} and R_{nonfire} are reconstructed feature maps, which are trained on class-specific encoding of features instead of entire semantic objects; hence, their addition is controlled feature perturbation instead of random image overlay. The application of the superimposition is controlled residual, in which the reconstructed class-specific features are imposed without changing the global spatial structure. Second, the $\text{clip}(\cdot)$ operation limits pixel values to the valid range between [0,1], so that synthesized samples are both visually realistic and semantically valid. The additive formulation, in contrast to alpha blending, maintains intensity-based fire signals and adds controlled background interaction. Since the generated residues are based on structurally aligned inputs, semantic plausibility is preserved, without the need of explicit attention-based spatial masking.

Similarly, *Autoencoder2* is utilized to extract *nonfire features* and reconstruct nonfire-like images, denoted as R_{nonfire}. These reconstructed images are superimposed onto the *fire images* X_{fire} to create synthesized fire images with nonfire features, as shown in Fig. 2.

1) *Reconstruction of Nonfire Images:*

$$R_{\text{nonfire},i} = f_{\text{AE2}}(X_{\text{nonfire},i}) \tag{9}$$

where f_{AE2} is the function learned by Autoencoder2, and $X_{\text{nonfire},i}$ is the ith nonfire image.

2) *Superimposition of Nonfire Features onto Fire Images:*

$$S_{\text{fire},i} = \text{clip}(X_{\text{fire},i} + R_{\text{nonfire},i}, 0, 1) \tag{10}$$

here, $S_{\text{fire},i}$ is the synthesized fire image with nonfire features.

Algorithm 3: Superimpose Operation.

The algorithm superimposes reconstructed residual data onto the underlying images and censor pixel values so that the pixel values can be within admissible intensity levels. As a result, the proposed approach maintains the spatial integrity and it inlays class-relevant data.

```
function SUPERIMPOSE R, X
Require: Residual/reconstructed images R, base images X
Ensure: Superimposed images S
    for i ← 1 to length(X) do
        S[i] ← clip(X[i] + R[i], 0, 1)
    end for
    return S
end Function
```

This training space offers a richer and more varied feature space to the classification model by incorporating these synthesized images in the training set. This addresses the issue of bias since the model will not be overly dependent on simplified or isolated attributes which are very specific to both fire and nonfire. The overlapping space results in the generation of a mixed feature space in which the binary difference between fire and nonfire is not as evident, and the model is compelled to acquire more subtle differences. The features of images that are required to be enriched allow it to be classified more accurately due to the training of the model on the other features. The chance of having a false alarm on a nonfire is minimized with a more sophisticated set of characteristics. It is also through this that the model is more accurate in identifying fire conditions in the presence of disturbing nonfire factors. To further explain the synthesis process, Fig. 2 presents representative examples of superimposed fire and nonfire images. These synthesized samples are then added to the original fire and nonfire images and create a more enriched training dataset, which is then used to train the DenseNet201 classifier. This integration ensures that the classifier is trained on original and mixed feature conditions.

D. Model Training and Evaluation

To enhance the number of features, the proposed methodology obtains an increased number of features by combining both original and superimposed images covering *fire* and *nonfire* categories. The resulting consolidated piece of data D is represented as

$$D = X_{\text{fire}} \cup S_{\text{fire}} \cup X_{\text{nonfire}} \cup S_{\text{nonfire}}. \tag{11}$$

A feature extraction function f is defined to map the raw integrated data D to a higher dimensional, more informative feature space F is as follows:

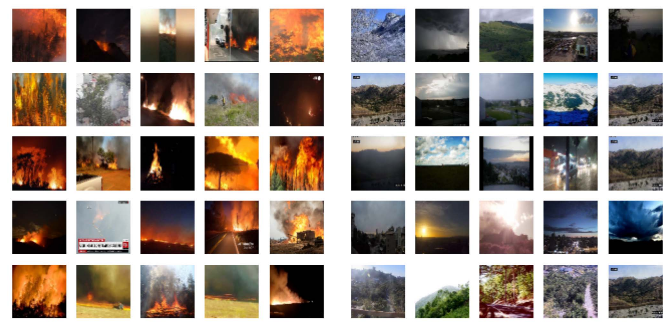
$$F = f(D). \tag{12}$$

The function f transforms the data from its original space $\mathbb{R}^n$ to a more enriched feature space $\mathbb{R}^m$ as follows:

$$f : \mathbb{R}^n \rightarrow \mathbb{R}^m. \tag{13}$$

TABLE III
DATASET PARAMETERS

Data set	Class	Training	Validation	Testing	Total
Wildfire	Fire	738	185	407	1330
	Nonfire	1810	453	488	2751
	Total	2548	638	895	4081
DFire	Fire	3200	800	800	4800
	Nonfire	3200	800	800	4800
	Total	6400	1600	1600	9600



(a) (b)

Fig. 3. Sample images representing different classes. (a) Fire images. (b) Nonfire images.

The approach leverages Shannon entropy $H(F)$ to measure the unpredictability and diversity within the feature space is defined by

$$H(F) = -\sum_i p(f_i) \log p(f_i). \tag{14}$$

Here, $p(f_i)$ denotes the empirical distribution of feature activations across the dataset. Entropy is used to quantify feature diversity and reduce representational bias. In practice, $p(f_i)$ is obtained by applying a normalization function (e.g., softmax-based normalization across feature activations) to convert continuous feature magnitudes into a discrete probability distribution prior to entropy computation. A higher $H(F)$ indicates a more diverse and less predictable feature space, which helps in reducing model bias by ensuring varied data representations. The objective function for training incorporates the mse from the autoencoders and the entropy of the feature space is as follows:

$$L_{\text{total}} = L_{\text{MSE}} + \lambda H(F). \tag{15}$$

Here, λ is a regularization parameter that balances reconstruction fidelity and feature-space diversity. It controls the tradeoff between minimizing reconstruction error and encouraging distributed feature representations. A moderate value of λ ensures that the model maintains accurate class-specific encoding while preventing feature-space collapse toward dominant dimensions.

It is important to clarify that the entropy is not computed directly over raw continuous high-dimensional feature vectors. Instead, the feature activations are first transformed into a normalized probabilistic representation. Specifically, the feature magnitudes are aggregated across samples and normalized using a softmax-based scaling to obtain a valid probability distribution satisfying $\sum_i p(f_i) = 1$ and $p(f_i) \geq 0$. Thus, $p(f_i)$ represents

Algorithm 4: Train and Evaluate the Classifier.

This fine-tunes the proposed model using the enriched dataset.

Require: $X_{\text{train}}, y_{\text{train}}, X_{\text{test}}, y_{\text{test}}$

Ensure: Trained model and evaluation metrics

1: $model \leftarrow$ DENSENET201($\texttt{include\_top} = False$, $\texttt{weights} = \texttt{imagenet}$)

2: Add classification head: GAP $\rightarrow$ DENSE$(2048, \texttt{relu}) \rightarrow$ DROPOUT$(0.5) \rightarrow$ DENSE$(1, \texttt{sigmoid})$

3: Compile model with optimizer=`adam`, loss=`binary_crossentropy`

4: Train for 100 epochs with batch size 32 and validation split 0.2

5: $predictions \leftarrow model.predict(X_{\text{test}})$

6: Compute Accuracy, Precision, Recall, FPR, and FNR

7: **return** $model$ and evaluation metrics

the normalized activation strength of feature dimension i across the dataset rather than a probability density in the continuous feature space. This transformation ensures mathematical validity of the entropy computation while preserving interpretability. The entropy term therefore acts as a regularization mechanism that discourages overconcentration of representation in a limited subset of feature dimensions and promotes a more informative and distributed latent feature space.

III. Dataset Description

The proposed method is tested and validated on two publicly available datasets. The details of data splitting are shown in Table III and sample images of the various classes are shown in Fig. 3.

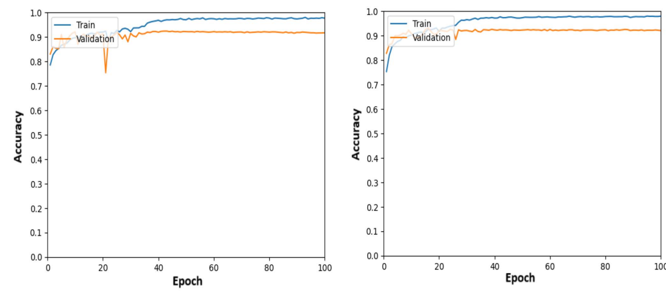
1) *Dataset 1: Wildfire Data* [27], provides a comprehensive collection of images specifically curated for wildfire detection. It has a training set of 738 fire and 1810 nonfire images, a validation set containing 185 fire and 453 nonfire images, and the testing set with 407 fire and 488 nonfire images. These images cover diverse conditions that are relevant to forest fire detection, such as presence of different light conditions as well as more adverse conditions like fog and smoke.
2) *Dataset 2: DFireDataset* [28], combines images of multiple sources that offers a mixed set of fire, no-fire, and smoke images. This dataset consists of 3200 training images in every category (fire, no fire, and smoke), 800 validation images, and 800 testing images in each of the categories. DFire contains more than 21 000 images, which are annotated in the YOLO format, with a constraint on categories that are associated only with fire, smoke, fire and smoke, and none. The source of images is diversified in terms of the environment and condition.

IV. Experimental Setup

The proposed method, was used in combination with the state-of-the-art models that were used to make comparative contributions: namely, the conventional model [6], the low-level bias approach [10], adversarial debiasing approach [26], augmentation + PCA classification [29], and fireXnet [9]. OpenCV was used to do preprocessing of all the images and convert them into RGB and resize them into a standard size of 80×80 pixels. The pixels were brought to the scale of $[0, 1]$ as they were divided by 255. In addition, a red background was superimposed on nonfire images to create visual bias that was intentional in the model, thus increasing its ability to discern features of the fire against the possibility of noise and bias.

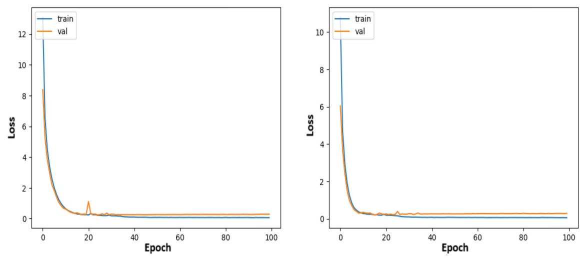
"Keras ImageDataGenerator," was used to augment the data and add variability and robustness to the data by rotating, shifting width and height, shear, zooming, and flipping horizontally. This was to ensure that there was a reduction in overfitting to simulate various real-world circumstances. The proposed feature-distinction scheme, along with the *DenseNet201* model to classify an image, were trained and tested on an NVIDIA $A100$ GPU accessed through Google Colab. The autoencoder structures were convolutional and max-pooling layers in the encoder using preceding upsampling and convolutional layers in the decoder. *DenseNet201* which is a trained model on ImageNet, was altered by adding dense and dropout layers to classify binary (Fire/NonFire). All models used the batch size of 32, and 100 epochs as a training regime. The batch size was chosen to maximize the efficiency of the computation and the stability of the gradients in the two datasets. The initial learning rate was set to 1×10^{-4} as the initial experiments on learning showed stable convergence. The learning rate was reduced by a factor of 0.2 on finding a plateau in the validation loss with the ReduceLROnPlateau callback. The autoencoders were trained to minimize the mse, and the model *DenseNet201* was optimized on binary cross-entropy loss and the Adam optimizer. The regularization parameter in the entropy-based objective, is given by λ and was tuned through preliminary experiments within a controlled range to achieve a balanced tradeoff between reconstruction fidelity and feature diversity without destabilizing training.

The classification model was tested on an independent test dataset run through the same debiasing autoencoder (i.e., the trained Autoencoder $_{\text{nonfire}}$, reused as a common reconstruction component). Model performance was also evaluated in terms of *accuracy* (Fig. 4) and *loss* metrics (Fig. 5), visualized during *training*, and a *confusion matrix* was used to evaluate performance of the classifier, where true positives, false positives, true negatives, and false negatives are identified.



(a) (b)

Fig. 4. Accuracy of proposed model on identified datasets. (a) Accuracy on Wildfire dataset. (b) Accuracy on DFire dataset.



(a) (b)

Fig. 5. Loss curve of proposed model on identified datasets. (a) Loss curve on Wildfire dataset. (b) Loss curve on DFire dataset.

TABLE IV
PERFORMANCE METRICS USED FOR EVALUATION

Metric	Definition	Metric	Definition
Accuracy	$\frac{\mathrm{TP+TN}}{\mathrm{TP+TN+FP+FN}}$	Precision	$\frac{\mathrm{TP}}{\mathrm{TP+FP}}$
Recall	$\frac{\mathrm{TP}}{\mathrm{TP+FN}}$	FPR	$\frac{\mathrm{FP}}{\mathrm{FP+TN}}$
FNR	$\frac{\mathrm{FN}}{\mathrm{TP+FN}}$	Confusion Matrix	$\begin{bmatrix} \mathrm{TP} & \mathrm{FP} \\ \mathrm{FN} & \mathrm{TN} \end{bmatrix}$

V. Results and Analysis

A. Performance Parameters

The performance of the proposed framework is evaluated on a set of evaluation metrics and are presented in Table IV. The identified metrics are chosen with the aim of not only performance validation of the model on various aspects but also to offer an understanding of the accuracy and effectiveness of performance at different circumstances. The evaluation metrics have been selected to address the problem of the imbalance in the classes and the prohibitive price of the safety-insensitive mistakes of detecting forest-fires occurrence scenario. Precision may be misleading when nonfire samples predominate, in this case, false-positive rate (FPR), false-negative rate (FNR), and precision are directly presented. These measures provide a relative evaluation of false-alarm performance and missed-detection risk which are vital to working wildfire surveillance.

B. Model Accuracy

Fig. 4 shows the accuracy plots of the proposed model and gives a holistic idea of the performance of the model using both *Wildfire* and *DFire* dataset. The accuracy on the Wildfire dataset is reported to be significantly high and gradually gets better with time, as shown in Fig. 4. A very small variation in the line of validation accuracy is an indication that the proposed model is able to generalize efficiently and is not overfitting. A shallow trough at the 20th epoch shows that there is a minor adjustment period where the model learns to distinguish between subtle features of fire and nonfire images under diverse conditions. The fact that the accuracy has been very high in the rest of the epochs is a pointer that the model has effectively represented the key features that are necessary to carry out effective classification.

C. Model Loss

The loss curve for the proposed model, as shown in Fig. 5, provides useful information on the learning curves on both the datasets. In the case of Wildfire data, the training loss begins with a very large value and decreases at a very high rate in the first epochs indicating a rapid learning. The validation loss closely follows the training loss, which means that there is not much overfitting. When the two losses converge toward the lower end, it is an indication of continuous error minimization. The spikes observed in the validation loss indicates that the model is unable to cope with particular samples, but this is corrected in subsequent epochs. Similarly, the loss curve for the DFire dataset, shown in Fig. 5, shows an immediate drop in training and validation losses. The similarity of the two curves is indicative that the two models can be generalized across diverse samples, i.e., fire, no-fire, and smoke images.

D. Comparative Analysis

The proposed method is compared to various existing approaches *DenseNet201* [6], *ResNet50* [6], *InceptionV3* [6], *the low-level bias approach* [10], *adversarial debiasing approach* [26], *augmentation + PCA classification* [29], and *fireXnet* [9] wherein the model is tested on two different datasets to have a fair analysis on diverse environmental conditions.

Traditional models, including *DenseNet201*, *VGG16*, *ResNet50*, and *InceptionV3*, are not effective in forest fire detection due to weak generalization and limited ability of feature extraction suited to the harsh domain of forest fires. These models have difficulty with environmental variability and tend to overfit providing little insight into unique features of fires like different intensities and presence of smoke. Their architectures, good for general tasks, do not have domain-specific enhancements needed to differentiate between fires and similar phenomena, such as sunlight reflections or fog. The *augmentation + PCA* detection method has reduced the FPR but high FNR which is a measure of poor performance. Although, data augmentation increases robustness, it ignores the essential subtle fire properties and PCA ignores fire-specific properties leading to a high FNR. Therefore, this method does not adequately cover the multifaceted biases of fire detection. The *low-level debiasing* approach modulates low-level image characteristics, i.e., color, texture, and edge details, to reduce bias but oversimplifies the problem. Forest fire detection relies on complicated features that are not described by low-level features and this algorithm does not reflect complex patterns, such as smoke and flame indicators. When the training data are not diverse enough, it struggles to generalize across a wide range of fire scenarios. The *adversarial debiasing* method provides an adversarial element that is used to identify and minimize biases, but provokes difficulties with balancing the learning process. *FireXnet* has high FNR and the simple architecture and low-resource capability restrict its ability to model dynamic and complex features of fire.

On the other hand, the proposed solution achieves superior performance by integrating a holistic bias mitigation strategy that addresses both low-level and high-level features. It makes use of the sophisticated feature-extraction of the proposed DL framework to identify the important features of fire, including dynamic movements of smoke and variations in the flame. Methods like data augmentation and transfer learning enhance

TABLE V
PERFORMANCE COMPARISON ON IDENTIFIED DATASETS (OVERALL)

Dataset	Method	Accuracy %	FPR	FNR	Precision	Recall
Wildfire	DenseNet201 [6]	0.66	0.59	0.02	0.79	0.66
	Resnet50 [6]	0.62	0.63	0.05	0.74	0.62
	InceptionV3 [6]	0.66	0.56	0.04	0.77	0.66
	Augmentation + PCA [29]	0.71	0.16	0.44	0.71	0.71
	Low-level Debiasing [10]	0.66	0.58	0.02	0.79	0.66
	Adversarial Debiasing [26]	0.46	0.66	0.40	0.48	0.46
	FireXnet [9]	0.59	0.02	0.75	0.76	0.59
	Proposed	**0.80**	**0.21**	**0.19**	**0.80**	**0.80**
DFire	DenseNet201 [6]	0.57	0.85	0.01	0.75	0.57
	Resnet50 [6]	0.60	0.79	0.01	0.75	0.60
	InceptionV3 [6]	0.60	0.78	0.02	0.74	0.60
	Augmentation + PCA [29]	0.63	0.37	0.36	0.63	0.63
	Low-level Debiasing [10]	0.60	0.79	0.01	0.75	0.60
	Adversarial Debiasing [26]	0.49	0.85	0.17	0.48	0.48
	FireXnet [9]	0.55	0.01	0.90	0.76	0.55
	Proposed	**0.76**	**0.25**	**0.22**	**0.76**	**0.76**

TABLE VI
PERFORMANCE COMPARISON WITH VARYING EPOCHS

Dataset	Method	Epochs	Training Accuracy %	Validation Accuracy %	Training Loss	Validation Loss
Wildfire	DenseNet 201 [6]	1	0.94	0.95	0.20	0.25
		50	0.99	0.93	0.01	0.39
		100	0.97	0.91	0.01	0.79
	Resnet [6]	1	0.93	0.01	0.27	0.55
		50	0.99	0.92	0.04	0.77
		100	0.96	0.94	0.02	0.45
	Inception V3 [6]	1	0.89	0.91	0.30	0.84
		50	0.96	0.94	0.01	0.32
		100	0.99	0.97	0.02	0.08
	Low level [10]	1	0.93	0.94	0.18	0.30
		50	0.99	0.96	0.01	0.13
		100	0.99	0.97	0.01	0.11
	FireXnet [9]	1	0.82	0.91	0.41	0.68
		50	0.98	0.89	0.06	0.50
		100	0.99	0.94	0.03	0.22
	Proposed Approach	**1**	**0.99**	**0.83**	**0.01**	**0.83**
		50	**0.97**	**0.92**	**0.08**	**0.26**
		100	**0.98**	**0.92**	**0.06**	**0.29**
DFire	DenseNet 201 [6]	1	0.94	0.90	0.16	0.47
		50	0.99	0.96	0.07	0.40
		100	0.99	0.96	0.04	0.26
	Resnet [6]	1	0.94	0.01	0.16	0.81
		50	0.99	0.95	0.05	0.14
		100	0.99	0.95	0.07	0.30
	Inception V3 [6]	1	0.86	0.76	0.33	0.11
		50	0.99	0.97	0.01	0.07
		100	0.99	0.98	0.01	0.07
	Low level [10]	1	0.95	0.69	0.01	0.12
		50	0.99	0.60	0.06	0.26
		100	0.99	0.92	0.4	0.60
	FireXnet [9]	1	0.77	0.34	0.47	0.70
		50	0.98	0.96	0.04	0.09
		100	0.99	0.98	0.01	0.02
	Proposed Approach	**1**	**0.75**	**0.82**	**0.01**	**0.06**
		50	**0.97**	**0.92**	**0.07**	**0.25**
		100	**0.98**	**0.92**	**0.05**	**0.29**

generalization, adapting the model to various fire scenarios and conditions. Specialized training strategies, including loss functions tailored to balancing between false positives and false negatives and adaptive learning rates, are beneficial toward making models sensitive with respect to false positive and false negative. Incorporating domain-specific insights aligns the model closely with real-world fire behaviors, achieving high accuracy while reducing both false alarms and missed detections through a deep understanding of fire dynamics and environmental contexts. Table V presents an overall comprehensive performance report of all the methods, while Table VI presents the evaluation outcomes of the state-of-the-art methods, against the proposed method, with *varying number of epochs*.

E. Confusion Matrix

The confusion matrix of all the methods are presented in Figs. 6 and 7 for both the datasets. The confusion matrices in Fig. 6 provide a comparative analysis of the proposed model and other state-of-the-art approaches on the Wildfire dataset. The proposed model, depicted in Fig. 6, shows a balanced performance with a relatively low FPR of 70 and a higher TPR of 304. This indicates a good ability to distinguish between fire and nonfire images. Other models, like FireXnet and DenseNet201, also performed well but showed higher FPRs, suggesting a tradeoff between detecting all fire instances and minimizing false alarms.

The confusion matrices in Fig. 7, provide a comparative analysis of the proposed model and the other approaches on the DFire dataset. The proposed model shows a balanced performance with a relatively low FPR of 176 and a higher TPR of 624. This indicates a good ability to distinguish between fire and nonfire images. Other models, like FireXnet and DenseNet201 showed higher FPR, suggesting a tradeoff between detecting all fire instances and minimizing false alarms. For example, FireXnet correctly identified 799 fire images but also had a false positive count of 201, indicating a more conservative detection. DenseNet201, with 796 true positives and only four false positives, showed a cautious yet highly accurate detection. The augmentation + PCA approach shows a higher FPR of 289, despite a solid true positive rate of 511, indicating less balanced performance. Similarly, the adversarial debiasing approach exhibited a higher FPR of 136, which could lead to more false alarms in practical applications. Overall, the proposed method demonstrates superior performance, maintaining a good balance between true positives and minimizing false positives, thus showing better generalization across diverse scenarios in the both the datasets.

F. Entropy Performance Evaluation

Entropy is a measure of randomness of a dataset. For image datasets, it quantifies the diversity of pixel values, reflecting the richness of information contained within the images. To represent it mathematically, entropy H of a dataset is given as

$$H(F) = -\sum_{i} p(f_i) \log p(f_i). \tag{16}$$

A high value of entropy indicates that the dataset has diverse features and is rich in information value. A high entropy is an important factor for evaluating the fairness of a classification dataset as it has a lower risk of biased generalizations toward unseen samples. Therefore, for a fair analysis of the algorithm's ability to mitigate bias, we calculate the entropy of both the original biased dataset and the debiased dataset obtained after autoencoder training. As shown in Table VII and Fig. 8, the proposed method is able to increase the entropy value of the dataset by focusing on essential features and compressing and

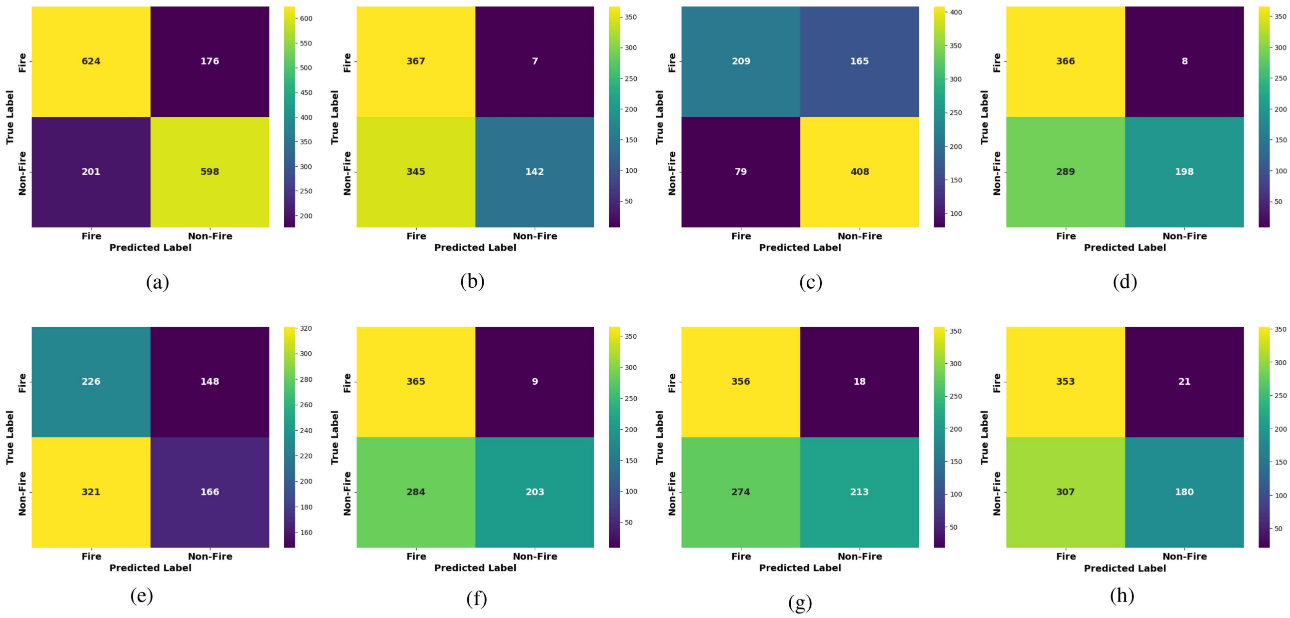


Fig. 6. Confusion matrix for all models on Wildfire dataset. (a) Proposed model. (b) FireXnet model [9]. (c) Augmentation + PCA [29]. (d) DenseNet201 [6]. (e) Adversarial debiasing [26]. (f) Low-level debiasing [10]. (g) InceptionV3 [6]. (h) Resnet50 [6].

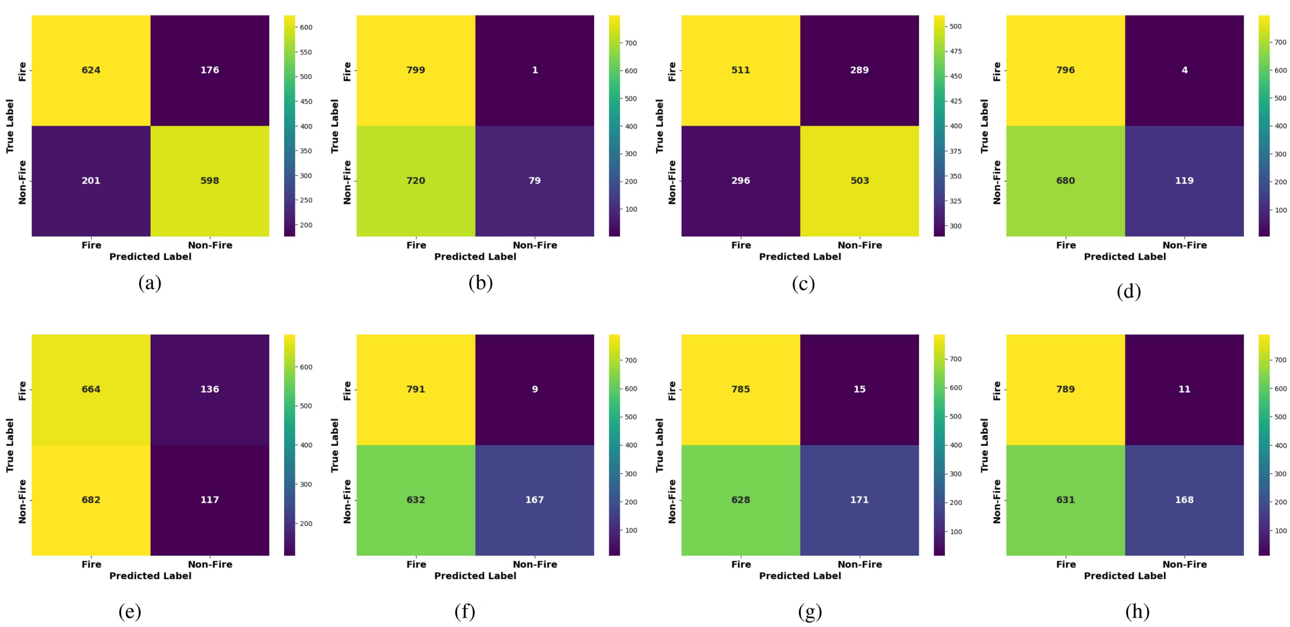


Fig. 7. Confusion matrix for all models on DFire dataset. (a) Proposed model. (b) FireXnet model [9]. (c) Augmentation + PCA [29]. (d) DenseNet201 [6]. (e) Adversarial debiasing [26]. (f) Low-level debiasing [10]. (g) InceptionV3 [6]. (h) Resnet50 [6].

TABLE VII
ENTROPY PERFORMANCE EVALUATION RESULTS

Dataset	Original Dataset Entropy	Debiased Dataset Entropy
Wildfire Data	5.371	15.469
DFire Dataset	9.8906	15.7472

reconstructing the input data through the autoencoders. This allows the model to diversify the the feature space, reducing repetitive patterns, and leading to a more balanced dataset for training.

VI. Discussion

A. Impact of Bias-Aware Representation Learning

The accuracy of the proposed approach on the *Wildfire detection image data* dataset [27], is 0.80, while DenseNet201 [6] is at 0.66, ResNet50 [6] is at 0.62, InceptionV3 [6] is at 0.66, and FireXnet [9] is at 0.59. A similar trend is observed on the *DFireDataset* [28], where the proposed approach attains an accuracy value of 0.76 accuracy compared to 0.57–0.60 for standard deep architectures.

More importantly, improvements are reflected in the form of reduction in false positive and negative rate. In the case of [27],

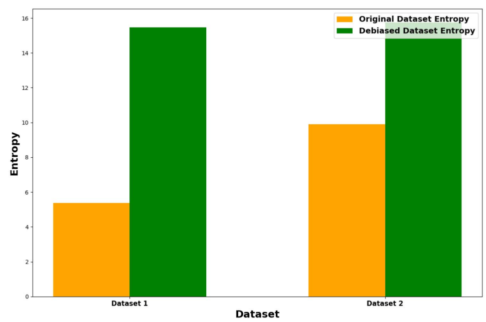


Fig. 8. Entropy performance evaluation.

the error profile of the proposed method with an FPR $= 0.21$ and FNR $= 0.19$, gives a more accurate representation of both the error rates compared to the adversarial debiasing [26] and low-level debiasing methods [10]. In [28], where background variability and illumination changes were more pronounced, the reduction in FNR $= 0.22$ suggests an increase in sensitivity to finer fire patterns.

These improvements can be owed to the dual autoencoder structure, which explicitly disentangles separates fire-related and nonfire-related latent encoders. The model learns specific encoding by the class before the classification processes, reducing dependence on nonspecific environmental characteristics, including haze, smoke-like clouds, or illumination artifacts. The superimposition based on the structured residual also boosts the training distribution and further urges the training distribution to develop fine-tuning correlations.

B. Effect of Superimposed Data Enrichment

The proposed superimposed synthesis has improved performance in generalization as compared to conventional augmentation strategies (accuracy 0.71 on [27] and 0.63 on [28]) and application. The residual-guided overlay produces controlled overlap of features between the fire and nonfire sets unlike the geometric transformations which do not cause features to overlap. This results in a mixed feature space that minimizes binary oversegregation and and improves robustness under ambiguous environmental conditions.

The entropy-regularized training objective

$$L_{\text{total}} = L_{\text{MSE}} + \lambda H(F)$$

further enhances feature diversity and prevents latent representation collapse. As shown in Table VI, validation accuracy remains stable across 50 and 100 epochs without significant divergence, suggesting reduced overfitting compared to baseline models.

C. Generalization Across Training Regimes

Epochwise comparisons show that DenseNet201 [6] and InceptionV3 [6] often achieve near-perfect training accuracy while exhibiting fluctuations in validation performance at higher epochs, indicating overfitting. In contrast, the proposed framework maintains stable validation accuracy across 50 and 100 epochs on both [27] and [28]. This stability supports the hypothesis that representation-level debiasing improves generalization under diverse environmental conditions.

D. Broader Applicability to Environmental Monitoring

Although this work focuses on forest fire detection, the proposed bias-aware data quality enhancement framework is inherently task-agnostic. Many environmental monitoring tasks, such as flood detection, deforestation mapping, oil-spill identification, and drought assessment face the same problems, such as class imbalance, environmental variability, sensor noise, and visual ambiguity between background and foreground classes.

Adapting the dual autoencoder paradigm to such tasks by independently learning class-specific latent representations (e.g., flooded versus nonflooded regions), then treating the remnant superimposition as controlled to augment the training diversity. According to the entropy-based regulation approach, the representations of features are guaranteed to be informative and delicately distributed in diverse sensing conditions.

E. Integration With Existing Monitoring Systems

Systemswise, the proposed framework may serve as a data-quality improvement layer in the already existing remote sensing infrastructures. In satellite-based early warning systems, the dual autoencoder module may serve as a preprocessing phase refining inputs representations prior to classification. There is potential to use the method as a bias-mitigation mechanism in AEV-based pipelines to reduce false alarms caused by shadows, reflections, or a dense haze of smoke. Because the architecture uses DenseNet201 [6] as the classification backbone, the addition of it into existing DL pipelines requires minimal structural modification. The enrichment phase works mostly during the training process, thus making the deployment complexity as similar to of the traditional CNN-based systems.

F. Limitations

Despite the improved performance, certain limitations must be acknowledged. The proposed framework establishes a two-step pipeline that involves two autoencoders with the ensuing DenseNet201 classifier. While the enrichment stage is applied during training, the overall architecture introduces additional computational overhead compared to single-stage CNN models. This may affect scalability in real-time or resource-constrained deployment scenarios. Furthermore, performance may degrade under extremely low-resolution imagery or severe sensor noise conditions, where fine-grained fire patterns become less distinguishable and latent feature disentanglement becomes more challenging. In addition, generalization to entirely new sensor modalities or geographical regions not represented in the training datasets may require domain adaptation mechanisms to ensure robustness under distribution shifts.

VII. Conclusion

This work presents a novel DL-based methodology for forest fire detection and addresses the limitations of traditional forest fire detection systems. Our approach uses two autoencoders applied over DenseNet201 and achieves more accurate, reliable, and efficient performance when compared with state-of-the-art methods. The integration of original and superimposed images for training autoencoders proves to be a significant enabler in making the proposed method more effective in distinguishing complex fire and nonfire scenarios, reducing false positives and improving true negatives. By tuning the DenseNet201 model on an enriched dataset, we achieve a high classification accuracy, which emphasizes the importance of advanced feature extraction in the field of image-based recognition systems. Using overlapping images to enrich the training dataset greatly improves the model's ability to handle ambiguous scenarios, a common challenge in fire detection. Future research will focus on reducing computational complexity through lightweight architectural variants and model compression strategies to facilitate real-time deployment. In addition, domain generalization techniques will be explored to enhance robustness across unseen sensors and geographical regions. Investigating performance under extreme low-resolution and noisy sensing conditions, as well as extending the framework to multispectral and multimodal remote sensing data, represents a promising direction for advancing reliable AI-based environmental monitoring systems.

References

[1] V. K. Singh, N. Anand, A. Pal, A. Chowdhury, and A. Srivastava, "Anomaly detection in dynamic graphs using multiple encoding strategies via transformers," *IEEE Trans. Consum. Electron.*, vol. 71, no. 2, pp. 5173–5188, May 2025.

[2] V. K. Singh, D. Agarwal, V. K. Gediya, R. S. Rathore, and W. Jiang, "Mitigating class imbalance in forest fire prediction with gan-augmented data fusion," *Inf. Fusion*, vol. 129, 2026, Art. no. 104005.

[3] M. Ahmad et al., "Hyperspectral image classification—traditional to deep models: A survey for future prospects," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 968–999, 2021.

[4] S. Bagui and K. Li, "Resampling imbalanced data for network intrusion detection datasets," *J. Big Data*, vol. 8, 2021, Art. no. 6.

[5] F. Kamalov, S. Moussa, and J. Avante Reyes, "KDE-based ensemble learning for imbalanced data," *Electronics*, vol. 11, no. 17, 2022, Art. no. 2703.

[6] M. Nallakaruppan, S. Pillai, G. Bharadwaj, and B. Balusamy, "Early detection of forest fire using deep image neural networks," in *Proc. IEEE IAS Glob. Conf. Emerg. Technol.*, 2023, pp. 1–5.

[7] X. Xie, J. Wang, Y. Zhang, J. Xin, L. Mu, and X. Xue, "A UAV forest fire detection method based on dual-light vision images," in *Proc. CAA Symp. Fault Detection, Supervision Saf. Tech. Processes*, 2023, pp. 1–6.

[8] W. Wang, Q. Huang, H. Liu, Y. Jia, and Q. Chen, "Forest fire detection method based on deep learning," in *Proc. Int. Conf. Cyber-Phys. Social Intell.*, 2022, pp. 23–28.

[9] K. Ahmad et al., "Firexnet: An explainable AI-based tailored deep learning model for wildfire detection on resource-constrained devices," *Fire Ecol.*, vol. 19, no. 1, pp. 1–19, 2023.

[10] V. Sengar, B. Vivek, G. Bhattacharya, J. Gubbi, A. Pal, and P. Balamuralidhar, "Low-level bias discovery and mitigation for image classification," in *Proc. IEEE Int. Conf. Signal Process. Commun.*, 2022, pp. 1–5.

[11] C. Wang, J. Tian, J. Cao, and X. Wang, "Deep learning-based UAV detection in pulse-Doppler radar," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5105612.

[12] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9268–9277.

[13] S. Sagawa, P. W. Koh, T. B. Hashimoto, and P. Liang, "Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization," 2020, *arXiv:1911.08731*.

[14] H. He, S. Zha, and H. Wang, "Unlearn dataset bias in natural language inference by fitting the residual," in *Proc. 2nd Workshop Deep Learn. Approaches Low-Resource NLP*, 2019, pp. 132–142.

[15] B. Kim, H. Kim, K. Kim, S. Kim, and J. Kim, "Learning not to learn: Training deep neural networks with biased data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Los Alamitos, CA, USA, Jun. 2019, pp. 9004–9012.

[16] J. Nam, H. Cha, S. Ahn, J. Lee, and J. Shin, "Learning from failure: Training debiased classifier from biased classifier," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, vol. 33, pp. 20673–20684.

[17] M. Pezeshki, S.-O. Kaba, Y. Bengio, A. Courville, D. Precup, and G. Lajoie, "Gradient starvation: A learning proclivity in neural networks," *Proc. Annu. Conf. Neural Inf. Process. Syst.*, 2021, pp. 1256–1272.

[18] S. Srivastava, D. Agarwal, B. K. Chaurasia, V. K. Singh, R. S. Rathore, and W. Jiang, "Federated blockchain-based resilient identity management for securing consumer vehicular networks," *IEEE Trans. Consum. Electron.*, early access, Dec. 25, 2025, doi: 10.1109/TCE.2025.3647809.

[19] Y. Li et al., "Federated domain generalization: A survey," *Proc. IEEE*, vol. 113, no. 4, pp. 370–410, Apr. 2025.

[20] N. Ruan et al., "Edge AI for Earth observation," *IEEE Internet Comput.*, vol. 29, no. 3, pp. 31–40, May/Jun. 2025.

[21] Q. Zhang et al., "Satcooper: Enhancing cooperative inference analytics for satellite service via multi-exit DNNs," *IEEE Trans. Mobile Comput.*, vol. 24, no. 9, pp. 8314–8328, Sep. 2025.

[22] P. K. Donta et al., "Socio-technical aspects of agentic AI," 2025, *arXiv:2601.06064*.

[23] V. K. Singh, J. Singh, R. S. Rathore, and D. Nema, "Gradient-based compression and approximate computing for deep network optimization in multimedia data processing," *IEEE Trans. Consum. Electron.*, early access, Jan. 21, 2026, doi: 10.1109/TCE.2026.3656716.

[24] V. K. Singh et al., "Low-light image enhancement for edge-based security surveillance in 6G-IoT visual systems," *IEEE Internet Things J.*, vol. 13, no. 5, pp. 8309–8322, Mar. 2026.

[25] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*, 2018, pp. 4700–4708.

[26] B. H. Zhang, B. Lemoine, and M. Mitchell, "Mitigating unwanted biases with adversarial learning," in *Proc. AAAI/ACM Conf. AI, Ethics, Soc.*, New York, NY, USA, 2018, pp. 335–340.

[27] B. Dincer, "Wildfire detection image data," 2023, *Kaggle Dataset*, Accessed: Jul. 2024. [Online]. Available: https://www.kaggle.com/datasets/brsdincer/wildfire-detection-image-data

[28] P. V. A. de Venancio, A. C. Lisboa, and A. V. Barbosa, "An automatic fire detection system based on deep convolutional neural networks for low-power, resource-constrained devices," *Neural Comput. Appl.*, vol. 34, no. 18, pp. 15349–15368, 2022.

[29] L. Jayasinghe, N. Wijerathne, C. Yuen, and M. Zhang, "Feature learning and analysis for cleanliness classification in restrooms," *IEEE Access*, vol. 7, pp. 14871–14882, 2019.

Vishal Krishna Singh received the bachelor's degree in information technology, the master's degree in computer technology and application, and the Ph.D. degree in information technology from the Indian Institute of Information Technology, Allahabad, Uttar Pradesh, India, in 2010, 2013, and 2018, respectively.

He is currently a Lecturer and is associated with the Networks and Communications Research Group, School of Computer Science and Electronics Engineering, University of Essex, Colchester, U.K. His research interests include Internet of Things, wireless sensor networks, in-network inference, machine learning, and data analytics.

Soumendu Chakraborty (Senior Member, IEEE) received the B.E. degree in information technology from the University Institute of Technology, University of Burdwan, Barddhaman, West Bengal, India, in 2005, the M.Tech. degree in computer science and engineering from GLA University, Mathura, Uttar Pradesh, India, in 2013, and the Ph.D. degree in information technology from the Indian Institute of Information Technology, Allahabad, Uttar Pradesh, India, in 2018.

He has 20 years of teaching and research experience. He is currently an Assistant Professor with the Indian Institute of Information Technology, Lucknow, Uttar Pradesh, India. He has authored or coauthored more than 20 research publications in journals and conferences. His research interests include computer vision, machine learning, deep learning, image processing, biometric systems, image stegnography, and pattern recognition.

Raman Singh received the M.Tech. degree from Indian Institute of Information Technology (IIIT), Lucknow, Uttar Pradesh, India, in 2024.

He was a Research Assistant with IIIT Lucknow. His research interests are in machine learning, computer vision, image processing, and pattern recognition.

Rajkumar Singh Rathore (Senior Member IEEE) received the bachelor's degree, dual master's, and doctorate degrees in computer science and engineering.

He currently serves as the Head of Cyber Security of Connected and Autonomous Systems at CINC, Head of Cyber Physical and Networks Systems at CeRISS, and Programme Director for the MSc in Computing and IT in the Department of Computer Science, School of Technologies, Cardiff Metropolitan University, Cardiff, U.K. He has several years of experience in teaching, learning, and research excellence. His research interests include wireless communications, the Internet of Things, cyber-physical systems, cyber security and privacy, connected and autonomous vehicles, EV charging infrastructure management, intelligent drone networking, and applications of AI and machine learning with focus on Transparent, Accountable and Responsible AI (TAR-AI).

Dr. Rathore is currently the Director of Policies at the ACM Emerging Interest Group on Trustworthy and Responsible Systems and a Fellow of the Higher Education Academy (HEA), U.K. His research has been supported by Nottingham Trent University, Nottingham, U.K., and Manchester Metropolitan University, Manchester, U.K.

Weiwei Jiang (Senior Member IEEE) received the B.Sc. degree of electronic engineering and Ph.D. degree of information and communication engineering from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 2013 and 2018, respectively.

He is currently an Assistant Professor with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, China, and Key Laboratory of Universal Wireless Communications, Ministry of Education, Beijing, China. He has authored or coauthored more than 60 academic papers in IEEE Transactions and other journals, with more than 3900 citations in Google Scholar. His research interests include artificial intelligence, machine learning, big data, wireless communication and edge computing.

Dr. Jiang is one of 2023 and 2024 Stanford's List of World's Top 2% Scientists.