# Lightweight Distilled Transformer-Based Vision Framework for Detection of Forest Fire and Smoke in Real-World Scenes

**Hassan Akbar[1,3], Tahir Nawaz[1,2,3,*], Md Asaduzzaman[4], Mohammad S. Hasan[5], Waqar Shahid Qureshi[6], Faisal Shafait[2]**

[1]Department of Mechatronics Engineering, College of Electrical & Mechanical Engineering, National University of Sciences and Technology, H-12, Islamabad, Pakistan

[2]Deep Learning Lab, School of Interdisciplinary Engineering & Sciences, National University of Sciences and Technology, H-12, Islamabad, Pakistan

[3]National Centre of Robotics and Automation (NCRA), National University of Sciences and Technology, H-12, Islamabad, Pakistan

[4]Department of Engineering, Staffordshire University, Stoke-on-Trent, UK

[5]Department of Computing, Staffordshire University, Stoke-on-Trent, UK

[6]School of Computer Science, University of Galway, Galway, Ireland

*Corresponding author: Tahir Nawaz (e-mail: tahir.nawaz@ceme.nust.edu.pk)

**Abstract.** Forest fires have become a ravaging threat with incidents growing rapidly across the globe. Several approaches for forest fire detection have been presented over the years, however, the need remains for an effective, computationally efficient, and unified vision-based solution, which can easily be deployable on edge devices for real-world applications. To this end, we present a lightweight model based on a distilled vision transformer (D-ViT) to classify forest imagery into fire, smoke and normal scenarios. We used ResNet50 as a teacher model trained on the target dataset and a compressed D-ViT as a student model trained using the knowledge distillation (KD) approach. Unlike existing approaches, the proposed D-ViT framework is computationally efficient with fewer trainable parameters and is unified in terms of detecting both fire and smoke (whichever is dominant) at longer ranges with visible imagery in the scene. For experimental validation, we deployed the model on Jetson Nano board, and performed an extensive evaluation and analysis of the proposed framework on data collected from public online sources, which we have made

available on request for use by the research community. The proposed D-ViT model achieves an encouraging performance with a processing speed of 18.84 frames per second (FPS) and accuracy of 94% using soft distillation, thus demonstrating a performance improvement over the 90% accuracy obtained with the ViT (without distillation). A comparison with several other standard deep classification models also shows encouraging results, with a better trade-off between accuracy and computational efficiency.

**Keywords**: Forest fire detection, vision transformers, knowledge distillation, lightweight model, visual imaging, early warning system.

## 1. Introduction

Forests not only provide animal habitat and human livelihoods but also provide natural protection from watersheds, prevent soil erosion, and mitigate climate change. Forest fires are unavoidable natural phenomena that have detrimental environmental impacts, adversely affecting the property, human life, and ecosystem [1], [2], [3]. According to the global forest watch [4], a total of 119 Mha tree cover was lost from fires globally from 2001 to 2021. The most tree cover loss (equal to 9.61 Mha) occurred in the year 2016. Indeed, several such incidents have been routinely reported worldwide, causing significant damage to life and ecology [5]. Specifically, in Pakistan, 5.46 Kha of tree cover has been lost between 2001 and 2021 due to fires [4]. There is indeed a dire need to develop a capability, particularly in a developing country like Pakistan, to enable early forest fire detection in real-world conditions to preserve the natural resources of the forests [6].

Previously, several approaches [7] have been proposed to address the challenge under consideration. Approaches exist that utilize aerial drones to patrol and monitor forest areas for fire detection [8], [9], but have limitations in terms of simultaneous area coverage, slower response time, costs and maintenance associated with keeping a fleet of drones, and the need for trained manpower to pilot them. Other approaches have been presented that rely on a network of sensor nodes [10], [11] deployed locally across forest areas at tree heights, measuring parameters such as temperature, smoke, barometric pressure, and humidity for early detection of fire. However, the performance of such systems might suffer due to high false alarms caused by varying environmental conditions, plus the deployment, maintenance, and communication costs of the sensor nodes might be quite challenging. The vision-based approaches used a network of fixed cameras to perform forest fire detection. It is here relevant to mention that relying on traditional appearance-based image features may not be desirable [12], as such approaches could fail under

varying illumination settings. Therefore, the use of discriminative features, as employed in deep learning-based approaches [13], [14], [15], [16], [17], [18] would be preferable. For instance, YOLOv5 [17] has been demonstrated to provide high-performance accuracy to detect forest fires. However, this model may not be the best choice to deploy in a real-time application on a smart edge device due to enhanced network complexity and computational cost. It would therefore be important to explore and test other lightweight network architectures to ensure effective deployment as well as real-time performance that is inevitably important for the application at hand [16], [19], [20]. There exist lightweight solutions that are based on different network architectures; however, they demonstrated the effectiveness of methods at shorter ranges [13], [14], [18] or were generally trained to detect either smoke or fire [1], [2], [3], [14], [16], [18], [21]. It would instead be desirable to have a unified trained network capable of detecting both smoke and fire at longer ranges (of the order of a few kilometers), deployable in real-world forest scenes, to allow early detection and a timely response by government authorities.

Vision Transformers (ViT) have reformed image classification by leveraging attention mechanism to capture long-range dependencies and complex patterns in visual data [22]. Unlike convolutional neural networks, ViTs treat images as sequences of patches, enabling them to model the global context effectively. ViTs have proven their strength by showing high performance on benchmark datasets. However, the high computational cost and large model size pose challenges to their deployment for real-world applications on smart embedded systems, which require lightweight and efficient models [23].

Therefore, the appropriate network choice needs to be made considering factors such as reduced network complexity, a better trade-off between accuracy and computational cost, and the ability to learn discriminative features at longer ranges. To address this requirement, the knowledge distillation framework [24], [25] presents a solution by distilling the knowledge from a large deep pre-trained teacher model into a smaller student model, through this we can significantly reduce the student model's complexity while maintaining its performance [26]. This distillation process involves training the student model to mimic the teacher's outputs while preserving its performance advantages in a compact form, suitable for embedded applications.

In this paper, we present a framework based on a unified, lightweight D-ViT model designed to accurately differentiate between fire, smoke, and normal scenes in video streams captured by camera nodes deployed in actual forest sites. Unlike complex architectures, our model is compact and trained via knowledge distillation (KD) for improved performance and specifically targeted for low-power edge devices. The system incorporates a camera, embedded hardware platform, communication setup, and GUI-based application for the entire process. We demonstrate an extensive performance evaluation and comparison

of the trained model on real public data collected from online sources with other related models in terms of several performance measures as well as the computational cost by means of deployment on an embedded platform (Jetson Nano – Maxwell GPU, Quad-Core ARM Cortex-A57 Processor, 4GB Memory) to assess the suitability for real-world applications. Overall, the results are very encouraging in terms of both performance accuracy and computational cost. We also make the collected data available on request **here** to facilitate its use by the community for reproducibility and comparisons.

This paper is organized as follows. Section 2 presents the proposed framework in detail. Section 3 provides the experimental validation by describing datasets (Sec. 3.1) and a detailed analysis of the results (Sec. 3.2). This is followed by conclusions in Sec. 4.

## 2. Proposed Forest Fire And Smoke Detection Framework

### *2.1 Overview*

The proposed system aims to address the limitations discussed in the previous section and offer a lightweight end-to-end framework for the early detection of forest fire and smoke based on the analysis of visual imagery coming from camera node(s) mounted on accessible high structures and covering longer ranges and areas. Here, we present a proof of concept for a single camera node, but the concept is scalable to a network of camera nodes deployed at multiple locations in a forest. A camera node outputs 2D images that are pre-processed before feeding them into the trained D-ViT network at the sensor edge for the detection of fire and smoke. We performed response-based KD with ResNet50 [27] a deep CNN as a teacher model trained on a target dataset (Fire, Smoke and Normal Images), and compressed distillable ViT as a student model for the main classification task (details in Sec. 2.2).

We used the Jetson Nano developer kit as the embedded platform, which is reliable and widely used in applications involving image processing and deep learning tasks [28] (Sec. 2.3). A graphical user interface-based application is designed to enable an operator in the control room to monitor and access live video stream(s) from multiple node(s) simultaneously (Sec. 2.4). On detection of fire and/or smoke, an early warning is generated, and the information is relayed back to relevant authorities through IoT infrastructure to act in an effective and timely manner.

### *2.2 Proposed Framework*

Knowledge distillation is the transfer of knowledge to a smaller model known as the student typically with less complexity and fewer trainable parameters from a large teacher model. Researchers have utilized different types of KD to leverage the information from the teacher model to guide the learning of the student

model, enabling the student to achieve the desired performance with reduced complexity. These include response-based KD [29], [30] in which the student model is trained to mimic the soft labels (probability distribution over target classes) for each input from the teacher model output. Feature-based KD [31], [32], [33] in which the student model is trained to match the activations of particular hidden layers inside the teacher network. Relation-based KD [34], [35], [36] in which student model can be trained using information that captures the relationship between feature maps in addition to the knowledge contained in the output and intermediate layers of a teacher network. In the proposed framework, training of the D-ViT model through response-based KD aims to replicate the performance of the ResNet50 model while keeping the computational complexity of the D-ViT model to the minimum.

### *2.2.1 ResNet50 as Teacher Model*

ResNet, short for "Residual Network," is a type of convolutional neural network (CNN) developed by Microsoft Research in 2015. It uses residual connections between layers to learn residual errors not captured by previous layers, improving learning and performance in deep networks. This helps to reduce the vanishing gradient problem [37], which is a common issue in deep learning where the error gradients become very small as they are back-propagated through the network, making it difficult for the network to learn.

The ResNet50 architecture consists of three main parts: the stem block, the sequential block, and the classification head (Figure 1). The stem block is the initial part of the network that is responsible for extracting features from the input images. It typically includes several layers, such as convolutional layers, pooling layers, and activation layers, which are designed to process the input data and extract useful features. The sequential block is the middle part of the network and is composed of multiple sequential layers. These layers take the output of the stem block as their input and further process the extracted features to produce more abstract and higher-level representations of input data. The classification head is the final part of the network and is responsible for making predictions based on the features extracted by the stem and sequential blocks. It typically includes one or more fully connected layers and a final output layer that produces the predicted class probabilities.

ResNet50 has been widely used in various computer vision tasks and has achieved state-of-the-art performance in many cases. For example, it has been demonstrated to achieve better accuracy, when fine-tuned with the pre-trained weights of the ImageNet dataset and modified its fully connected layer accordingly [38], [39].

The architecture of ResNet50 [27] is similar to the traditional CNN's with additional skip connections in its residual blocks (Figure 1). The skip connection in the residual network connects the activation output of the previous layer of the network to the next layers by just skipping the layer between two blocks. The residual connection directly adds the input value to the end of the block [40]. This residual connection does not go through the activation functions, thus avoiding the squashing of derivatives and resulting in a higher overall derivative of the block. The identity block is used when the input and the output activations have the same dimension for addition, whereas the convolutional block is utilized when the input and output do not have the same dimensions.

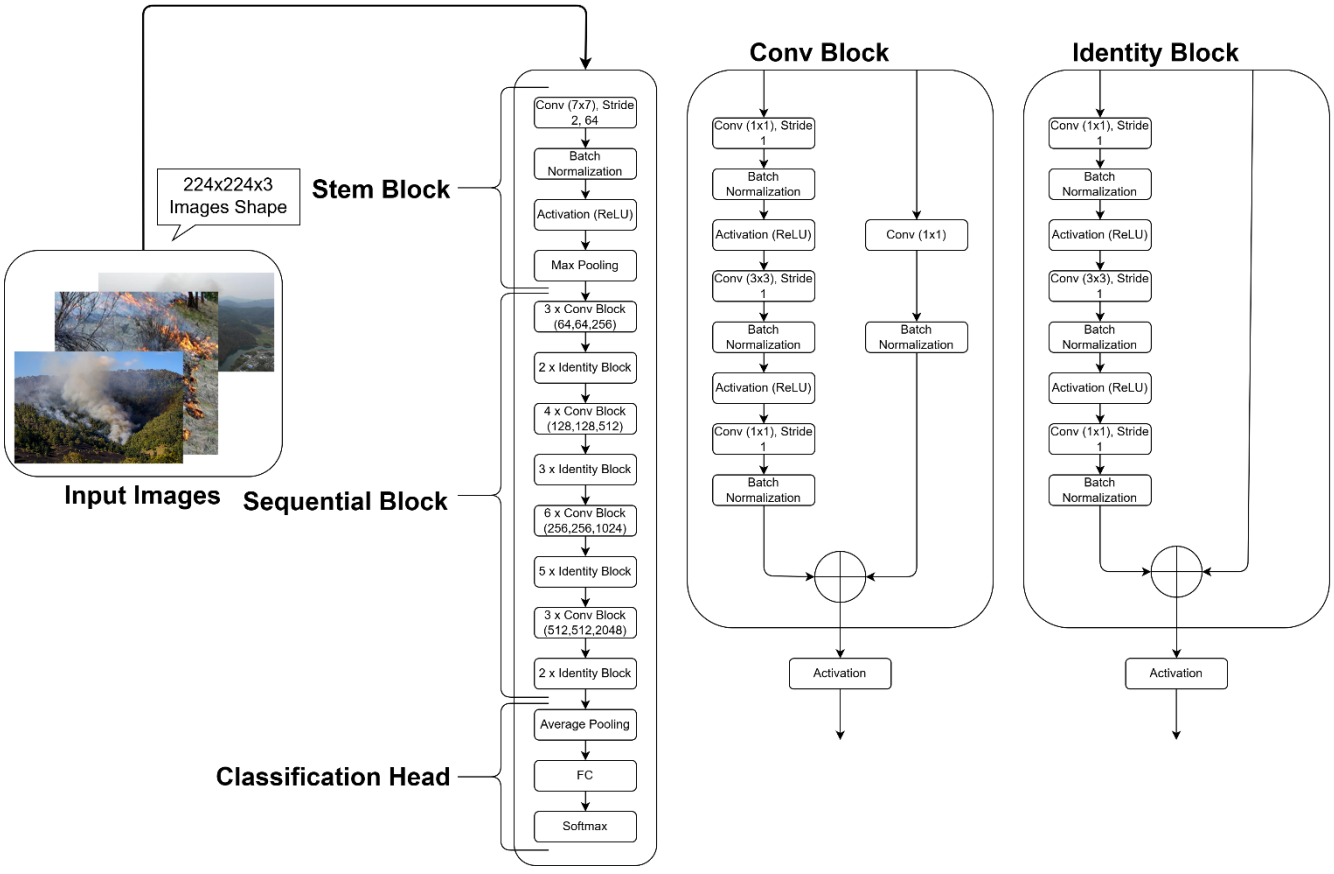


**Figure 1** Architecture of the ResNet50 (Teacher) model showing stem, sequential, and classification head blocks (left), a single residual block with skip connection (middle), and the identity block with skip connection (right).

To pre-train the teacher model on the target dataset for a downstream task the fully connected layer of our ResNet50 has been modified and it contains a layer with 2,048 input features and 128 output features. Then, the ReLU rectified linear activation function is applied and the output of this layer is passed to the layer

with 128 input features and 3 output features, i.e. the number of classes under consideration. This is followed by applying Softmax activation for a prediction that gives the probability of each class label. We used the cross entropy loss function to measure the error between predicted and target values while training the network. We empirically set the learning rate to 0.0005 and epochs to 30. The choices of hyper-parameters are discussed in Sec. 3.

#### *2.2.2 D-ViT as Student Model*

A distillable vision transformer is a variant of the standard vision transformer (ViT) designed to replicate performance through KD from a particular teacher model known for its deep architecture and high performance capabilities [41], [42], [33].

In the proposed D-ViT framework, the input image is first divided into fixed-size patches, which are then linearly embedded into a sequence of vectors. Alongside these patch embeddings, two special tokens are added, the classification token and the distillation token. The classification token is used for the main classification task, while the distillation token is specifically used to learn from the ResNet50 during the distillation process. The distillation token enables the D-ViT to learn additional information from the ResNet50 outputs, helping it to better mimic the teacher's response and to improve its performance. Similar to the classification token, the distillation token interacts with the patch embeddings and the classification token through the self-attention mechanism in each transformer layer [43]. This allows the distillation token to capture information from all parts of the input image and the classification token. The sequence of patch embeddings, along with the classification and distillation token, is passed through multiple layers of the transformer encoder of D-ViT. Each encoder layer comprises multi-head self-attention mechanisms and feed-forward neural networks (MLPs), allowing the model to capture complex patterns and dependencies in the data (Figure 2). During training, the D-ViT is supervised not only by the actual labels but also by the targets provided by the ResNet50 model. In the soft distillation process, soft targets are the output probabilities of the teacher model, which holds information about the data distribution and class relationships. Whereas in hard distillation, the student model is trained using the hard labels from the teacher model, which are the class predictions (the final output class) rather than the probabilities of each class. The training objective combines the standard classification loss with the distillation loss, which measures the divergence between the D-ViT and ResNet50 output distributions. This dual supervision ensures that the D-ViT model learns to replicate the ResNet50 performance while optimizing the primary classification task. During training, the final representation of the distillation token is used to align the student model's outputs

with the teacher model's outputs. This is typically done by minimizing a distillation loss between the student's distillation token output and the teacher's output.

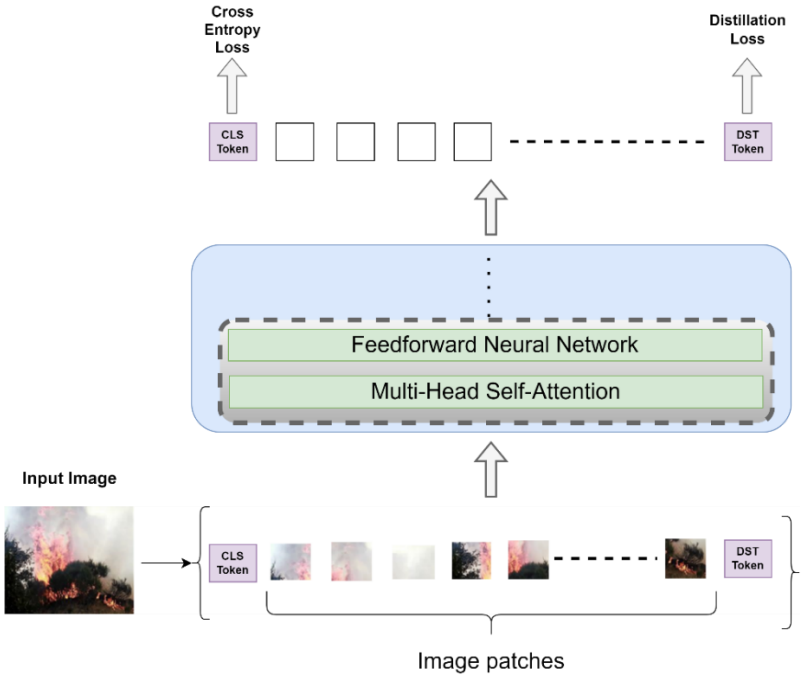


**Figure 2** Architecture of the D-ViT model showing image patches along with classification (CLS) and distillation (DST) token, and Transformer block with Multi-Head Self-Attention and Feedforward neural network.

Overall, a D-ViT utilizes the power of transformer architectures and the efficiency of response-based KD [44] to create a model that is both high-performing and computationally efficient, making it suitable for deployment on resource-constrained edge devices while maintaining high accuracy.Mathematically, the process of distillation is explained as follows. The teacher model logits $T_l$ are computed using equation 1, where the $x$ denotes the input image batch to the teacher model.

$$T_l = teacher(x) \qquad \textit{Equation 1}$$

$T_l$ teacher logits represent the raw predictions of the teacher model for each class (Fire, Smoke, Normal) before the softmax function is applied.

$$(S_l, d_s) = student(x, distill_{token}) \qquad \textit{Equation 2}$$

In equation 2, $S_l$ are the student model logits, $\text{distill}_{\text{token}}$ is the randomly generated distillation token and $d_s$ are the distillation tokens from a student model. The distillation token is a learnable parameter initialized

as a random tensor initially and is adjusted during the training process through backpropagation, just like any other learnable parameter in a network.

$$L_{mlp} = distill\ mlp(d_s) \qquad \textit{Equation 3}$$

In equation 3, $L_{mlp}$ represents the distillation logits produced by the distillation MLP (multi-layer perceptron) unit applied to the distillation tokens from the student model. Specifically MLP unit is a simple fully connected neural network with one or more hidden layers.

$$Loss = L_{CE}(S_l, y) \qquad \textit{Equation 4}$$

Equation 4 is the standard cross-entropy loss between the student logits $S_l$ and the actual labels $y$.

$$L_{distill}(soft) = T^2 \cdot L_{KL}(log\_softmax(\frac{L_{mlp}}{T}), log\_softmax(\frac{T_l}{T})) \qquad \textit{Equation 5}$$

Equation 5 computes the Kullback-Leibler divergence between the softened logits of the student and the teacher, scaled by $T$, where $T$ is the temperature parameter for soft distillation. When the logits are divided by the temperature $T$ and passed through the softmax function, the resulting probability distribution becomes more uniform for higher values of $T$. Temperature scaling enables the student model to not only focus on higher probabilities but also on a relative class relationship.

For hard distillation:

$$L_{distill}(hard) = L_{CE}(L_{mlp}, argmax(T_l)) \qquad \textit{Equation 6}$$

Equation 6 represents the cross-entropy loss calculated between the student model's predictions and the hard labels obtained from the teacher model's highest probability predictions.

Total loss $L_{total}$ is computed using equations either 7 or 8 for the training process based on hard and soft distillation choice (Figure 3).

$$L_{total} = (1 - \alpha) \cdot Loss + \alpha \cdot L_{distill}(soft) \qquad \textit{Equation 7}$$

$$L_{total} = (1 - \alpha) \cdot Loss + \alpha \cdot L_{distill}(hard) \qquad \textit{Equation 8}$$

In the proposed scheme, the student model D-ViT is configured with the following parameters, a patch size of 16x16 pixels. The model's dimensionality is set to 256, which corresponds to the size of the embedding vectors for each patch. The model includes six transformer layers, each with five attention heads. The feedforward network within each transformer layer has an inner dimensionality of 1,024, calculated as four times the embedding dimension. Dropout rates are set to 0.2 for the dropout within the transformer layers and 0.1 for the dropout applied to the patch embeddings.

During the distillation process, distillation logits from distillation MLPs are compared against the teacher's output (Figure 3). The total loss is a weighted combination of the student prediction loss (cross-entropy between the student's predictions and true labels) and the distillation loss. In the forward pass, the student

model makes predictions, and the distillation loss compares these predictions with the teacher model's soft targets or hard labels. In the backward pass, the student model weights are updated based on this total loss to improve its performance during training. The soft distillation loss is calculated using Kullback-Leibler (KL) divergence between the softened (temperature-scaled) logits of the teacher and the student, scaled by a temperature of 15 and combined with an alpha of 0.6, indicating a trade-off between the main task loss and the distillation loss. In contrast, for hard distillation cross-entropy is used instead, comparing the student's distillation logits with the hard labels from the teacher's predictions.

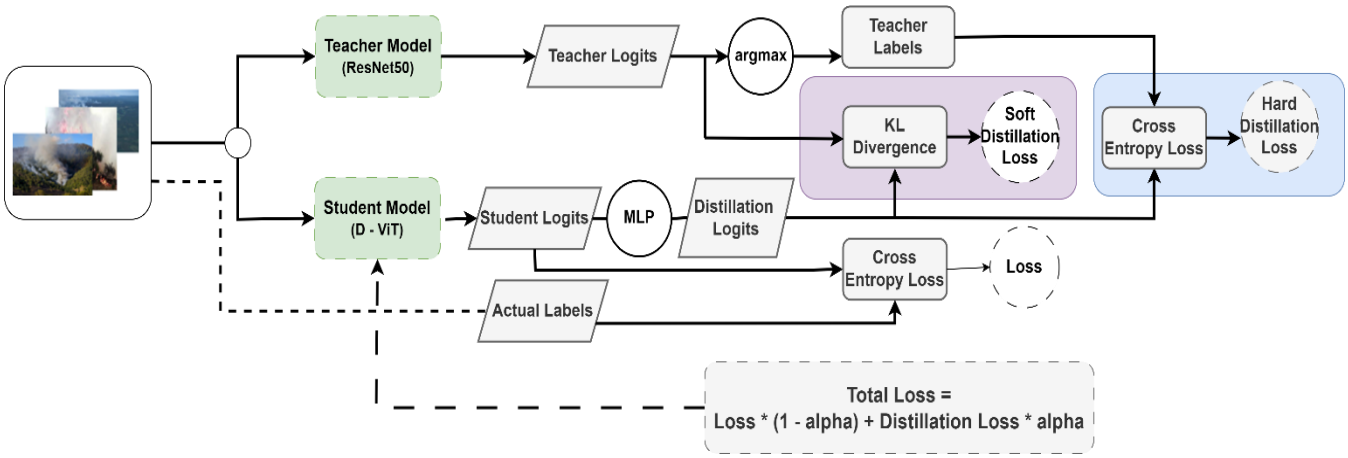


**Figure 3** Block diagram showing the process of knowledge distillation where a student model (D-ViT) learns from a teacher model (ResNet50) using either soft or hard distillation loss.

### *2.3 Components of Edge Node*

We have deployed and evaluated the system on an embedded platform, Nvidia Jetson Nano Maxwell GPU, Quad-Core ARM Cortex-A57 Processor, 4GB Memory. It has a micro SD slot for a micro SD Storage card with OS and data storage, 40-pin extension headers, two power input ports, 5V micro USB and DC barrel jack for 5V power input, four 3.0 USB ports, HDMI output port, Display port connector, and MIPI-CSI two camera connectors, Ethernet port, and UART headers. A network camera can be attached to the Jetson Nano board via a USB port and the system can be powered with a solar plate for battery charging and a supply mechanism for a node to work day and night. A network camera and Wi-Fi communication module are tested with the board for a proof of concept to assess the stability of communication between the node and the control room, supposedly to be managed by authorities. Each of the node that is deployed on the forest site can be accessed remotely. This remote accessing capability is useful to deal with any fault occurring at the node edge while performing scene monitoring.

### *2.4 GUI-based Application*

We have developed a desktop application with a graphical user interface (GUI) that allows users to access and visualize live streams from camera nodes deployed across a forest site. The user interface has address bars where the user can provide the address (e.g. IP or any other address) to access the corresponding data. The purpose of the application is an instantaneous visualization of imaging data plus its storage on a daily basis with the specific date and time stamp information saved for any later use by authorities. Figure 4 shows the application's front end and its interface containing several features. The video streams coming from different camera nodes are simultaneously accessible in different windows. If and when an incident (fire or smoke) is detected, an alert is accordingly generated in the corresponding node window.

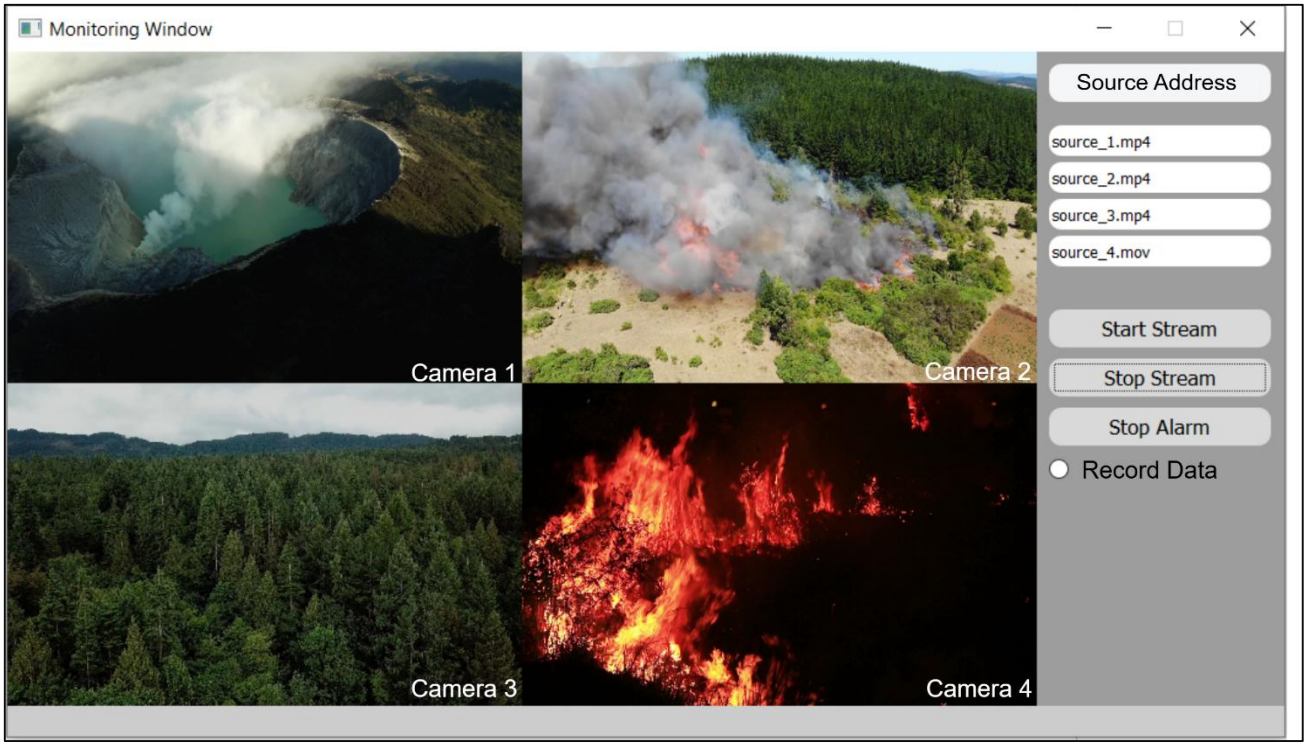


**Figure 4** GUI of the desktop application.

## 3. Experimental Validation

In this section, we first describe the dataset followed by an analysis of the results. As part of the results, we describe the model training as well as performance evaluation and comparison with other approaches along with a discussion on the computational complexity of the network.

*3.1 Dataset*

As for the choice of the dataset, there is an absence of a comprehensive real-world dataset with enough challenges and complexity for the problem under consideration. We made the best effort to create a customized dataset by collecting the available data from various public sources, focusing on fire, smoke, and normal scenes in forests. We carefully excluded any irrelevant samples, like indoor scenes, to keep it specific to forest environments and to ensure there are enough samples for the three classes under consideration: fire, smoke, and normal scenarios. The sources include Forest Fire dataset [45], [46] and Wild fire dataset [47], Dfire dataset [48], and the FESB MLID dataset [49]. Figure 5 shows representative images for fire, smoke, and normal scenarios from the dataset.

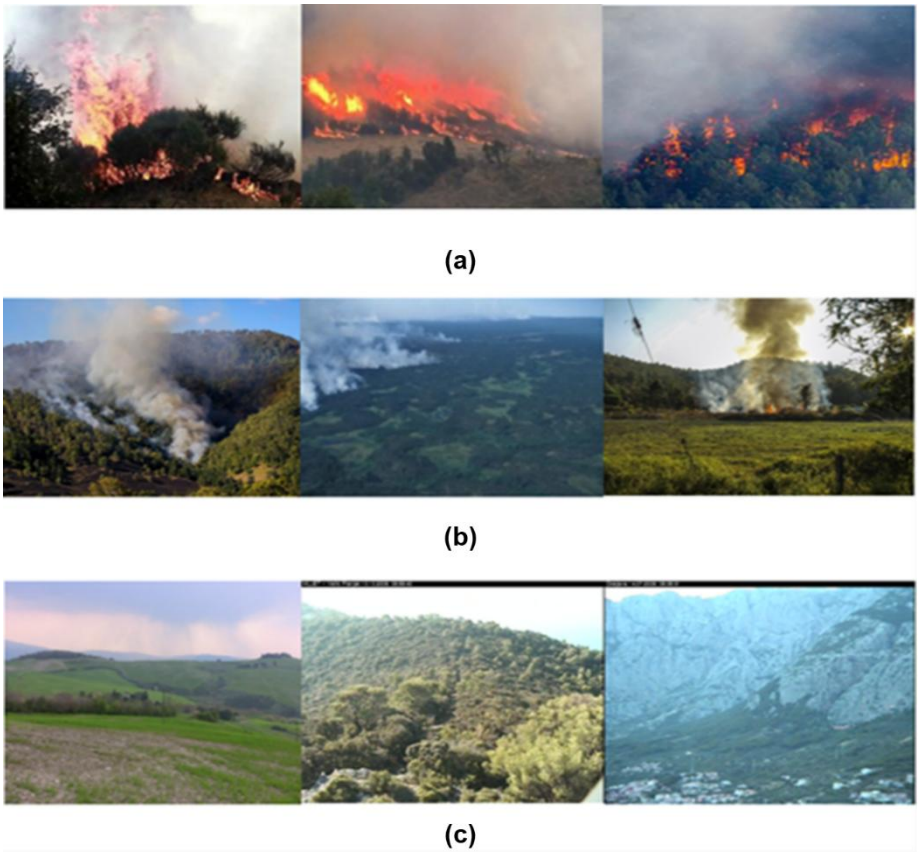


**Figure 5** Representative image samples for (a) Fire, (b) Smoke, and (c) Normal Scenarios.

We therefore created an initial dataset using 3,690 images (70% for training, 20% for validation, 10% for testing): this specifically includes 2,577 training images (859 for each of the three classes), 736 validation images fire, normal, and smoke in the ratio of (246+245+245), and 368 test images (124+122+122). For a more thorough evaluation, more unseen test data was added with 1,132 additional images from the

above-mentioned sources, making the tally for three classes to 1,500 test images: Fire – 500, Normal – 500, Smoke – 500. The dataset has been created keeping in view a diversity of challenges including fire and smoke, specifically for forest scenarios captured from different ranges, varying illumination settings, clutter, and different backgrounds.

The dataset for training was pre-processed by transforming and augmenting, including resizing to 224x224, random resize cropping, random horizontal flipping, random rotation with 20 degrees, and finally normalizing it after converting it to a tensor.

To also test the effectiveness of the framework on temporal data, we collected 30 real-world video sequences from different public online sources (pexels; istockphoto; pixabay; shutterstock), containing fire, smoke, and normal scenarios. The sequences are selected keeping in view various challenges such as varying sequence length, frame resolution, and scene capture from varying ranges.

### *3.2 Results and Analysis*

We used the well-known open-source machine learning framework, PyTorch, to train ResNet50 teacher and D-ViT student models on varying combinations of learning rates and optimizers while keeping epochs fixed to 30 (Table 1, 2).

For training the ResNet50 model, we chose a learning rate of 0.0005 and Adam optimizer as it maximizes accuracy. The training loss changed from 0.000309 to 0.000232, and validation accuracy changed from 0.91 to 0.93, the teacher model achieved an overall accuracy of 95% on test data.

Similarly, for training a D-ViT with ResNet50 as a teacher, we chose a learning rate of 0.000025 and Adam optimizer. The training loss changed from 0.37 to 0.12 and validation accuracy changed from 0.69 to 0.90, and the model achieved an overall accuracy of 94% on test data when trained using soft distillation. Figure 6 shows the training, validation loss and accuracy curves corresponding to the results of the D-ViT for 30 epochs with a learning rate of 0.000025 and Adam optimizer.

For an objective of detailed performance evaluation of the proposed D-ViT, we used the Precision, Recall, F1-score, and Accuracy measures. Precision evaluates the performance in terms of true positives and false positives. Recall assesses the performance based on true positives and false negatives. F1-score computes the performance as a harmonic mean of Precision and Recall scores. Accuracy calculates the performance using true positives, false positives, false negatives, and true negatives.

**Table 1** Choice of hyper-parameters for training the teacher model ResNet50. SGD: Stochastic gradient descent; Adam: Adaptive moment estimation

| Epoch | Learning Rate | Optimizer | Training Loss | Validation Accuracy | Test Accuracy |
|---|---|---|---|---|---|
| 30 | 0.0001 | SGD | 0.000427 - 0.000422 | 0.38-0.49 | 0.46 |
| 30 | 0.0005 | SGD | 0.000437 - 0.000397 | 0.39-0.78 | 0.85 |
| 30 | 0.0001 | Adam | 0.000370 - 0.000230 | 0.88-0.93 | 0.95 |
| 30 | 0.0005 | Adam | 0.000309 - 0.000232 | 0.91-0.93 | 0.95 |

**Table 2** Choice of hyper-parameters for training the student model D-ViT.

| Epoch | Learning Rate | Optimizer | Training Loss | Validation Accuracy | Test Accuracy |
|---|---|---|---|---|---|
| 30 | 0.000025 | SGD | 0.57-0.51 | 0.33-0.30 | 0.50 |
| 30 | 0.000025 | Adam | 0.37-0.12 | 0.69-0.90 | 0.94 |

Table 3 shows the performance of the proposed D-ViT model with hard and soft distillation in terms of the Precision, Recall, F1-score, and Accuracy measures on the test data. For comparison, the table also lists performance scores of four other state-of-the-art deep networks – including InceptionV3, ResNet18, ResNet101, VGG16 – all trained and tested on the same dataset.

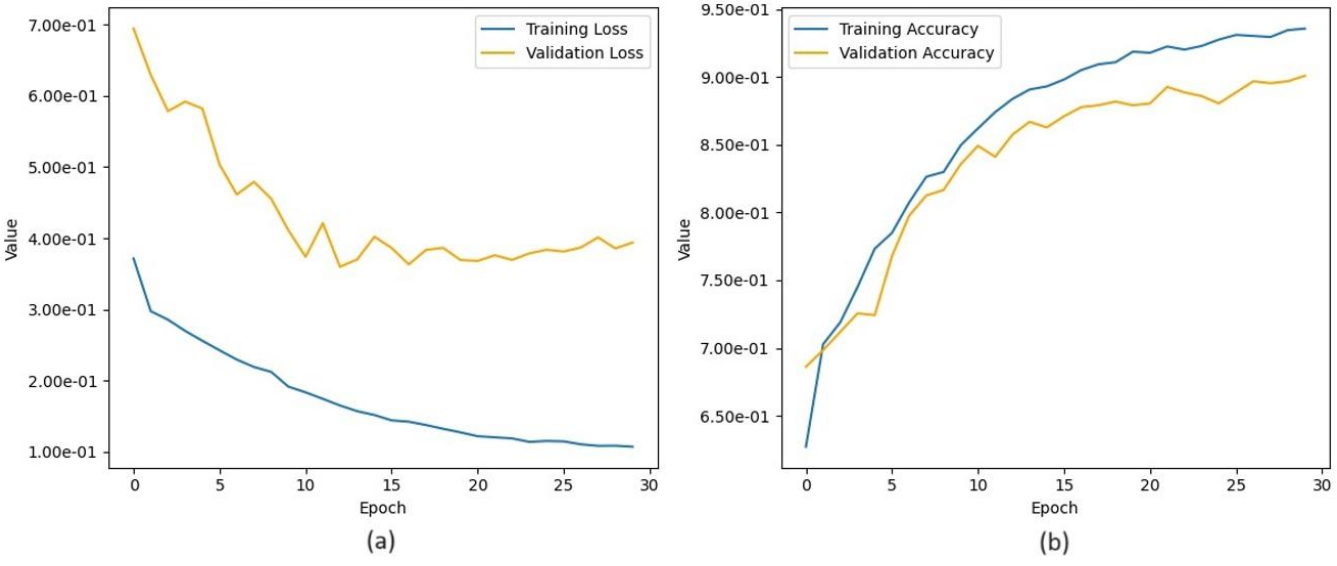


**Figure 6** (a) Training and validation loss, (b) Training and validation accuracy, plots corresponding to a learning rate of 0.000025 and Adam optimizer for a D-ViT model.

Additionally, (Table 3) also includes the performance of the same ViT (Only) when trained and tested without the distillation process. It is evident that the proposed D-ViT achieves the overall best performance in terms of classification accuracy compared to other models and improved performance when trained using soft distillation from ResNet50. The confusion matrix plot (Figure 7) shows how well the proposed D-ViT model predicts the correct classes by comparing actual and predicted values and identifies specific areas of misclassification.

We also computed the cumulative performance in terms of Average Precision, Average Recall, and Average F1 score (Figure 8). D-ViT consistently shows the best performance based on all measures while being the least complex model. Figure 9 shows the qualitative results of the proposed framework on representative test images for the three classes (Fire, Smoke, Normal).

Another important aspect to evaluate for the problem at hand is to assess the computational cost, which is inevitably important for deployment in a real-time application. Table 4 provides the computational performance analysis in terms of the resource utilization (CPU RAM usage, GPU usage) and computational time taken to classify all of the 1,500 test samples by deploying models on the Jetson Nano board.

**Table 3** Performance evaluation and comparison of the proposed D-ViT with other deep learning models in terms of Precision, Recall, F1-score, and Accuracy measures.

| **Model** | **Precision** | | | **Recall** | | | **F1- score** | | | **Test Accuracy** |
|---|---|---|---|---|---|---|---|---|---|---|
| | **Fire** | **Smoke** | **Normal** | **Fire** | **Smoke** | **Normal** | **Fire** | **Smoke** | **Normal** | |
| InceptionV3 | 0.97 | 0.76 | 0.97 | 0.92 | 0.98 | 0.73 | 0.94 | 0.86 | 0.84 | 0.88 |
| ResNet18 | 0.92 | 0.92 | 0.91 | 0.98 | 0.86 | 0.92 | 0.95 | 0.89 | 0.92 | 0.92 |
| ResNet101 | 0.98 | 0.87 | 0.96 | 0.96 | 0.96 | 0.89 | 0.97 | 0.92 | 0.92 | 0.94 |
| VGG16 | 0.97 | 0.87 | 0.97 | 0.98 | 0.96 | 0.86 | 0.98 | 0.91 | 0.91 | 0.93 |
| ViT (Only) | 0.87 | 0.91 | 0.93 | 0.96 | 0.86 | 0.88 | 0.91 | 0.88 | 0.90 | 0.90 |
| D-ViT (Hard) | 0.95 | 0.92 | 0.92 | 0.94 | 0.89 | 0.96 | 0.95 | 0.91 | 0.94 | 0.93 |
| D-ViT (Soft) | 0.95 | 0.92 | 0.94 | 0.97 | 0.91 | 0.93 | 0.96 | 0.92 | 0.93 | 0.94 |

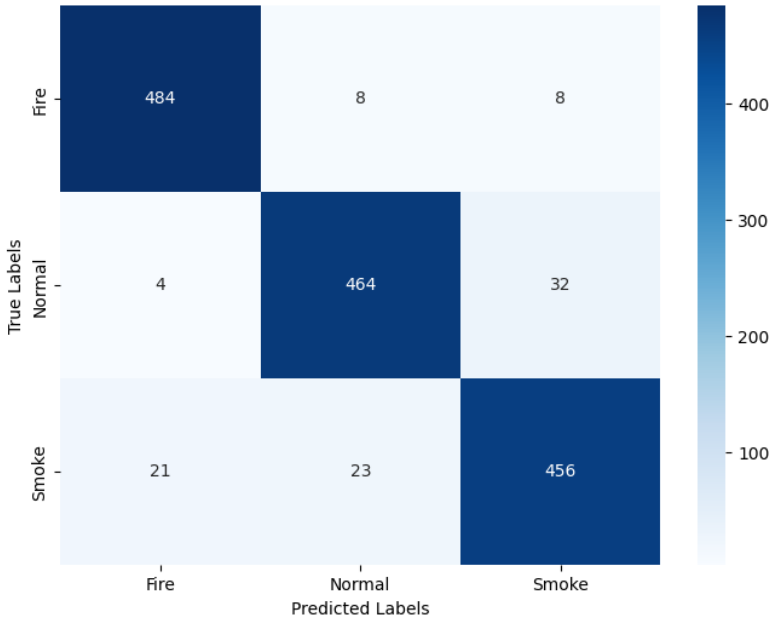


**Figure 7** Confusion matrix plot for test data classification using D-ViT model.

In terms of complexity analysis, D-ViT is ranked first by obtaining the least computational time to classify the entire test data and trainable parameters of (79.62 sec) and (5 million) respectively, with the lowest resource utilization of 1.8GB and 0.3GB for CPU-RAM and GPU respectively, while maintaining the better performance over other deep CNNs. Therefore, the proposed D-ViT with the application of response-based KD using soft distillation presents a preferred choice based on the acceptable trade-off between performance and computational cost (Table 3,4).

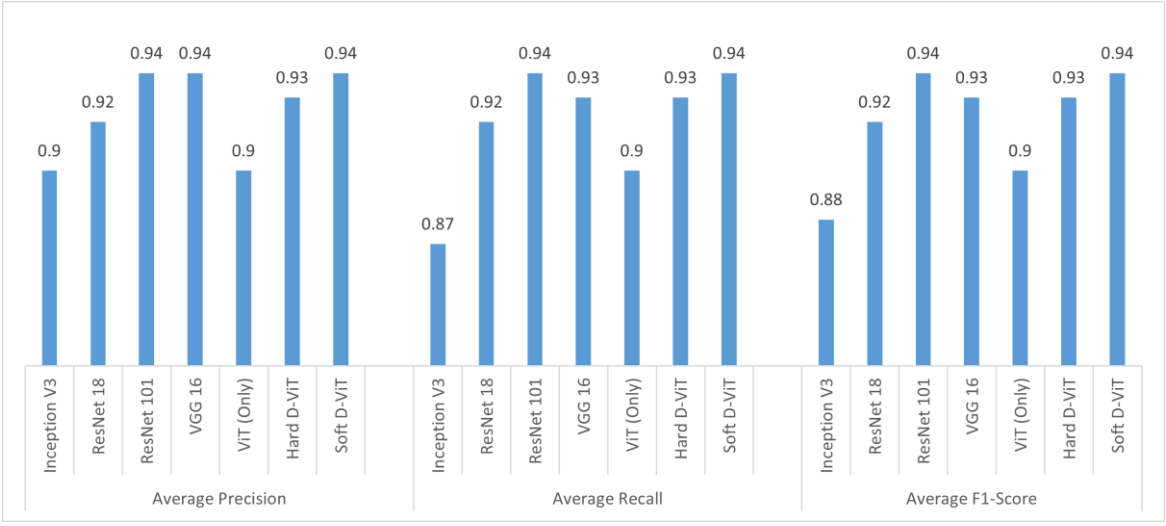


**Figure 8** Cumulative performance in terms of Average Precision, Average Recall, and Average F1-score.


**Figure 9** Qualitative results of the proposed framework on representative test samples for each of the three classes (Fire, Smoke, Normal).

For a more holistic evaluation of D-ViT, we further analyzed its performance on 30 real video sequences containing the 12 Fire, 8 Smoke, and 10 Normal scenarios as already described in Sec. 3.1. The resolution of video sequences ranges from 640x360 to 3840x2160 and the sequence length vary from 57 to 3,564 frames. For each test video, the entire sequence is labeled as either fire, smoke, or normal as its actual label, and the predicted labels for each frame are noted. The overall confusion matrix is then calculated by comparing the predicted and actual labels across all videos, showing the true positives, false positives, true negatives, and false negatives (Figure 10). The cumulative performance in terms of average Precision, Recall and F1-score is calculated to be 0.95, 0.95, and 0.95, respectively, with overall accuracy of 0.95 for all video sequences. The experimentation is performed again on the Jetson Nano board to analyze the real-time performance. This separate analysis highlights the robustness and efficiency of the D-ViT model in real-world, diverse video scenarios. While the proposed model achieved an average FPS of 13.25 on the video sequences, lower than the FPS of 18.84 as obtained on image-based test data (described above). This difference suggests that while D-ViT is effective in terms of classification accuracy across different scenarios, its temporal processing performance may slightly decrease in more complex or varied real-

world conditions. In such cases, the use of more powerful edge devices with better processing power could be preferable to further improve the processing speed.

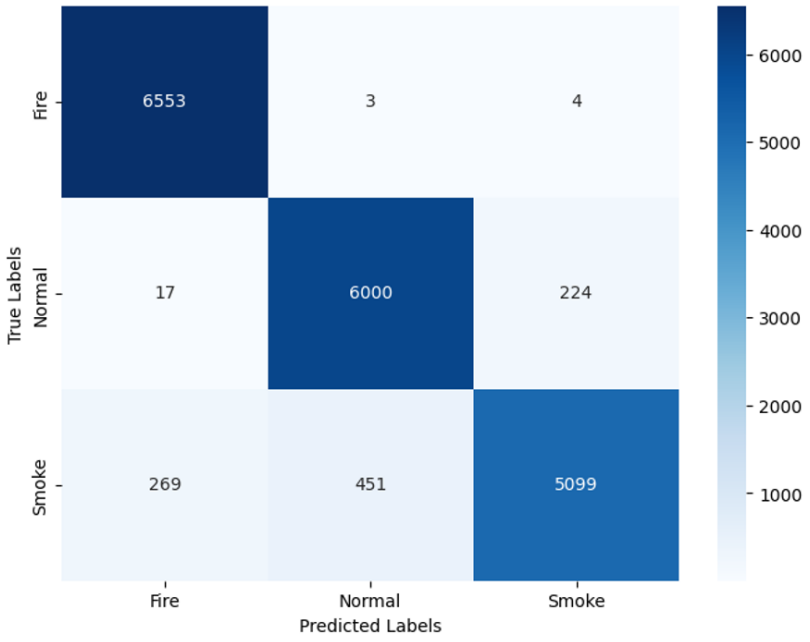


**Figure 10** Confusion matrix plot for the temporal data evaluation using D-ViT model.

**Table 4** Computational performance evaluation in terms of resource utilization and computational time taken to classify all of the test samples, by deploying models on Jetson Nano board.

| **Model** | **Parameters (Million)** | **CPU RAM Usage (GB)** | **GPU Usage (GB)** | **Computational Time (sec)** | **FPS** |
|---|---|---|---|---|---|
| InceptionV3 | 24 | 2.0 | 1.6 | 303.38 | 4.94 |
| ResNet18 | 12 | 2.6 | 0.9 | 214.02 | 7.01 |
| ResNet101 | 45 | 2.2 | 1.5 | 223.37 | 6.72 |
| VGG 16 | 138 | 2.8 | 1.2 | 485.91 | 3.09 |
| D-ViT | 5 | 1.8 | 0.3 | 79.62 | 18.84 |

Indeed, the proposed D-ViT is a lightweight student model with 5 million trainable parameters and 3.10 GFLOPs ($10^9$, floating point operations), achieving an improved classification accuracy of 94% (with distillation) as compared to 90% (without distillation). In comparison, the teacher model (ResNet50) has a much more increased architectural complexity with 24 million parameters and 4.13 GFLOPs, achieving an accuracy of 95%. Therefore, through effective knowledge distillation, the proposed D-ViT model

reports a comparable performance to the teacher model, while offering a significant (4.8 times) reduction in parameters and (~1.33 times) reduction in FLOPs.

## 4. Conclusion

This paper presented an efficient and unified framework that is aimed at the early detection of forest fire as well as smoke, deployable in real-world applications for longer ranges. Specifically, the proposed framework relied on D-ViT for instantaneous classification of the Fire, Smoke, and Normal scenarios using camera node(s) (equipped with edge computing) for forest sites, and generating an alert if and when an incident (fire or smoke) is detected. The framework also consists of a GUI-based desktop application that enables accessibility and visualization of the stream(s) coming from camera node(s), as deployed across a forest site.

We performed an extensive evaluation of the proposed framework on real datasets collected from various public online sources based on several measures. The results show that the proposed D-ViT reports the best trade-off in terms of accuracy and computational cost (when deployed on Jetson Nano) as compared to four other state-of-the-art networks, making it a more suitable choice for real-world applications.

*Availability of Data and Material:* The data used in this study is collected from public online sources (acknowledged/cited in the manuscript) and is made available in an organized form in an online repository to facilitate its use by the research community.

*Competing Interests:* The authors have no competing interests to declare that are relevant to the content of this article.

*References*

[1] Y. Gao and P. Cheng, 'Full-Scale Video-Based Detection of Smoke from Forest Fires Combining ViBe and MSER Algorithms', *Fire Technol*, vol. 57, no. 4, pp. 1637–1666, Jul. 2021, doi: 10.1007/s10694-020-01052-3.

[2] J. Xie, F. Yu, H. Wang, and H. Zheng, 'Class Activation Map-Based Data Augmentation for Satellite Smoke Scene Detection', *IEEE Geosci. Remote Sensing Lett.*, vol. 19, pp. 1–5, 2022, doi: 10.1109/LGRS.2022.3179013.

[3] D. Q. Tran, M. Park, Y. Jeon, J. Bak, and S. Park, 'Forest-Fire Response System Using Deep-Learning-Based Approaches With CCTV Images and Weather Data', *IEEE Access*, vol. 10, pp. 66061–66071, 2022, doi: 10.1109/ACCESS.2022.3184707.

[4] Vizzuality, 'Forest Monitoring, Land Use & Deforestation Trends | Global Forest Watch'. Accessed: Dec. 16, 2022. [Online]. Available: https://www.globalforestwatch.org/

[5] N. Abid *et al.*, 'Burnt Forest Estimation from Sentinel-2 Imagery of Australia using Unsupervised Deep Learning', in *2021 Digital Image Computing: Techniques and Applications (DICTA)*, Gold Coast, Australia: IEEE, Nov. 2021, pp. 01–08. doi: 10.1109/DICTA52665.2021.9647174.

[6] A. Zulfiqar *et al.*, 'AI-ForestWatch: semantic segmentation based end-to-end framework for forest estimation and change detection using multi-spectral remote sensing imagery', *J. Appl. Rem. Sens.*, vol. 15, no. 02, May 2021, doi: 10.1117/1.JRS.15.024518.

[7] F. Bu and M. S. Gharajeh, 'Intelligent and vision-based fire detection systems: A survey', *Image and Vision Computing*, vol. 91, p. 103803, Nov. 2019, doi: 10.1016/j.imavis.2019.08.007.

[8] A. Majeed and S. O. Hwang, 'A Multi-Objective Coverage Path Planning Algorithm for UAVs to Cover Spatially Distributed Regions in Urban Environments', *Aerospace*, vol. 8, no. 11, p. 343, Nov. 2021, doi: 10.3390/aerospace8110343.

[9] Z. Jiao *et al.*, 'A Deep Learning Based Forest Fire Detection Approach Using UAV and YOLOv3', in *2019 1st International Conference on Industrial Artificial Intelligence (IAI)*, Shenyang, China: IEEE, Jul. 2019, pp. 1–5. doi: 10.1109/ICIAI.2019.8850815.

[10] A. Bayo, D. Antolín, N. Medrano, B. Calvo, and S. Celma, 'Early detection and monitoring of forest fire with a wireless sensor network system', *Procedia Engineering*, vol. 5, pp. 248–251, 2010, doi: 10.1016/j.proeng.2010.09.094.

[11] P. K. Singh and A. Sharma, 'An insight to forest fire detection techniques using wireless sensor networks', in *2017 4th International Conference on Signal Processing, Computing and Control (ISPCC)*, solan, India: IEEE, Sep. 2017, pp. 647–653. doi: 10.1109/ISPCC.2017.8269757.

[12] D. Dzigal, A. Akagic, E. Buza, A. Brdjanin, and N. Dardagan, 'Forest Fire Detection based on Color Spaces Combination', in *2019 11th International Conference on Electrical and Electronics Engineering (ELECO)*, Bursa, Turkey: IEEE, Nov. 2019, pp. 595–599. doi: 10.23919/ELECO47770.2019.8990608.

[13] K. Muhammad, J. Ahmad, Z. Lv, P. Bellavista, P. Yang, and S. W. Baik, 'Efficient Deep CNN-Based Fire Detection and Localization in Video Surveillance Applications', *IEEE Trans. Syst. Man Cybern, Syst.*, vol. 49, no. 7, pp. 1419–1434, Jul. 2019, doi: 10.1109/TSMC.2018.2830099.

[14] K. Muhammad, J. Ahmad, I. Mehmood, S. Rho, and S. W. Baik, 'Convolutional Neural Networks Based Fire Detection in Surveillance Videos', *IEEE Access*, vol. 6, pp. 18174–18183, 2018, doi: 10.1109/ACCESS.2018.2812835.

[15] R. Xu, H. Lin, K. Lu, L. Cao, and Y. Liu, 'A Forest Fire Detection System Based on Ensemble Learning', *Forests*, vol. 12, no. 2, p. 217, Feb. 2021, doi: 10.3390/f12020217.

[16] J. Zhang, H. Zhu, P. Wang, and X. Ling, 'ATT Squeeze U-Net: A Lightweight Network for Forest Fire Detection and Recognition', *IEEE Access*, vol. 9, pp. 10858–10870, 2021, doi: 10.1109/ACCESS.2021.3050628.

[17] Z. Xue, H. Lin, and F. Wang, 'A Small Target Forest Fire Detection Model Based on YOLOv5 Improvement', *Forests*, vol. 13, no. 8, p. 1332, Aug. 2022, doi: 10.3390/f13081332.

[18] L. Huang, G. Liu, Y. Wang, H. Yuan, and T. Chen, 'Fire detection in video surveillances using convolutional neural networks and wavelet transform', *Engineering Applications of Artificial Intelligence*, vol. 110, p. 104737, Apr. 2022, doi: 10.1016/j.engappai.2022.104737.

[19] M. Jeong, M. Park, J. Nam, and B. C. Ko, 'Light-Weight Student LSTM for Real-Time Wildfire Smoke Detection', *Sensors*, vol. 20, no. 19, p. 5508, Sep. 2020, doi: 10.3390/s20195508.

[20] A. Jadon, M. Omama, A. Varshney, M. S. Ansari, and R. Sharma, 'FireNet: A Specialized Lightweight Fire & Smoke Detection Model for Real-Time IoT Applications', Sep. 04, 2019, *arXiv*: arXiv:1905.11922. Accessed: Oct. 03, 2022. [Online]. Available: http://arxiv.org/abs/1905.11922

[21] V. L. Kasyap, D. Sumathi, K. Alluri, P. Reddy CH, N. Thilakarathne, and R. M. Shafi, 'Early Detection of Forest Fire Using Mixed Learning Techniques and UAV', *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–12, Jul. 2022, doi: 10.1155/2022/3170244.

[22] A. Dosovitskiy *et al.*, 'An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale', Jun. 03, 2021, *arXiv*: arXiv:2010.11929. Accessed: May 12, 2023. [Online]. Available: http://arxiv.org/abs/2010.11929

[23] L. Papa, P. Russo, I. Amerini, and L. Zhou, 'A survey on efficient vision transformers: algorithms, techniques, and performance benchmarking', *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–20, 2024, doi: 10.1109/TPAMI.2024.3392941.

[24] X. Chen, Q. Cao, Y. Zhong, J. Zhang, S. Gao, and D. Tao, 'DearKD: Data-Efficient Early Knowledge Distillation for Vision Transformers', in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA: IEEE, Jun. 2022, pp. 12042–12052. doi: 10.1109/CVPR52688.2022.01174.

[25] B. Zhao, R. Song, and J. Liang, 'Cumulative Spatial Knowledge Distillation for Vision Transformers', in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, Paris, France: IEEE, Oct. 2023, pp. 6123–6132. doi: 10.1109/ICCV51070.2023.00565.

[26] A. H. Vo and B. T. Nguyen, 'A framework-based transformer and knowledge distillation for interior style classification', *Neurocomput.*, vol. 565, no. C, Feb. 2024, doi: 10.1016/j.neucom.2023.126972.

[27] K. He, X. Zhang, S. Ren, and J. Sun, 'Deep Residual Learning for Image Recognition', Dec. 10, 2015, *arXiv*: arXiv:1512.03385. Accessed: Nov. 19, 2022. [Online]. Available: http://arxiv.org/abs/1512.03385

[28] P. Gonzalez-Gil, A. Robles-Enciso, J. A. Martinez, and A. F. Skarmeta, 'Architecture for Orchestrating Dynamic DNN-Powered Image Processing Tasks in Edge and Cloud Devices', *IEEE Access*, vol. 9, pp. 107137–107148, 2021, doi: 10.1109/ACCESS.2021.3101306.

[29] S. W. Kim and H.-E. Kim, 'TRANSFERRING KNOWLEDGE TO SMALLER NET- WORK WITH CLASS-DISTANCE LOSS', 2017.

[30] G. Hinton, O. Vinyals, and J. Dean, 'Distilling the Knowledge in a Neural Network', Mar. 09, 2015, *arXiv*: arXiv:1503.02531. Accessed: Jun. 30, 2024. [Online]. Available: http://arxiv.org/abs/1503.02531

[31] D. Chen, J.-P. Mei, Y. Zhang, C. Wang, Y. Feng, and C. Chen, 'Cross-Layer Distillation with Semantic Calibration', Aug. 29, 2021, *arXiv*: arXiv:2012.03236. Accessed: Jun. 30, 2024. [Online]. Available: http://arxiv.org/abs/2012.03236

[32] G. Zhou, Y. Fan, R. Cui, W. Bian, X. Zhu, and K. Gai, 'Rocket Launching: A Universal and Efficient Framework for Training Well-Performing Light Net', *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, Art. no. 1, Apr. 2018, doi: 10.1609/aaai.v32i1.11601.

[33] X. Jin *et al.*, 'Knowledge Distillation via Route Constrained Optimization', in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South): IEEE, Oct. 2019, pp. 1345–1354. doi: 10.1109/ICCV.2019.00143.

[34] B. Peng *et al.*, 'Correlation Congruence for Knowledge Distillation', in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South): IEEE, Oct. 2019, pp. 5006–5015. doi: 10.1109/ICCV.2019.00511.

[35] N. Passalis, M. Tzelepi, and A. Tefas, 'Probabilistic Knowledge Transfer for Lightweight Deep Representation Learning', *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 5, pp. 2030–2039, May 2021, doi: 10.1109/TNNLS.2020.2995884.

[36] W. Park, D. Kim, Y. Lu, and M. Cho, 'Relational Knowledge Distillation', in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: IEEE, Jun. 2019, pp. 3962–3971. doi: 10.1109/CVPR.2019.00409.

[37] H. H. Tan and K. H. Lim, 'Vanishing Gradient Mitigation with Deep Learning Neural Network Optimization', p. 4, 2019.

[38] S. Mascarenhas and M. Agarwal, 'A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for Image Classification', in *2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)*, Bengaluru, India: IEEE, Nov. 2021, pp. 96–99. doi: 10.1109/CENTCON52345.2021.9687944.

[39] L. Zhang, M. Wang, Y. Fu, and Y. Ding, 'A Forest Fire Recognition Method Using UAV Images Based on Transfer Learning', *Forests*, vol. 13, no. 7, p. 975, Jun. 2022, doi: 10.3390/f13070975.

[40] A. S. B. Reddy and D. S. Juliet, ‘Transfer Learning with ResNet-50 for Malaria Cell-Image Classification’, in *2019 International Conference on Communication and Signal Processing (ICCSP)*, Chennai, India: IEEE, Apr. 2019, pp. 0945–0949. doi: 10.1109/ICCSP.2019.8697909.

[41] ‘A framework-based transformer and knowledge distillation for interior style classification - ScienceDirect’. Accessed: Jun. 27, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231223010950

[42] Y.-J. Heo, Y. Choi, Y.-W. Lee, and B.-G. Kim, ‘Deepfake Detection Scheme Based on Vision Transformer and Distillation’, *ArXiv*, Apr. 2021, Accessed: Jul. 08, 2024. [Online]. Available: https://www.semanticscholar.org/paper/269c0799be005848362c69faddad5fc8555da618

[43] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, ‘Training data-efficient image transformers & distillation through attention’, Jan. 15, 2021, *arXiv*: arXiv:2012.12877. Accessed: Jul. 08, 2024. [Online]. Available: http://arxiv.org/abs/2012.12877

[44] J. Gou, B. Yu, S. Maybank, and D. Tao, *Knowledge Distillation: A Survey*. 2020. doi: 10.48550/arXiv.2006.05525.

[45] ‘https://www.kaggle.com/datasets/kutaykutlu/forest-fire’.

[46] ‘https://www.kaggle.com/datasets/alik05/forest-fire-dataset’.

[47] ‘wildfire-dataset/wildfire-smoke at master · aiformankind/wildfire-dataset · GitHub’. Accessed: Dec. 16, 2022. [Online]. Available: https://github.com/aiformankind/wildfire-dataset/tree/master/wildfire-smoke

[48] ‘GitHub - gaiasd/DFireDataset: D-Fire: an image data set for fire and smoke detection.’ Accessed: Dec. 16, 2022. [Online]. Available: https://github.com/gaiasd/DFireDataset

[49] ‘Welcome to the Wildfire Observers and Smoke Recognition Homepage’. Accessed: Dec. 16, 2022. [Online]. Available: http://wildfire.fesb.hr/

## Figure Captions:

**Figure 1** Architecture of the ResNet50 (Teacher) model showing stem, sequential, and classification head blocks (left), a single residual block with skip connection (middle), and the identity block with skip connection (right).

**Figure 2** Architecture of the D-ViT model showing image patches along with classification (CLS) and distillation (DST) token, and Transformer block with Multi-Head Self-Attention and Feedforward neural network.

**Figure 3** Block diagram showing the process of knowledge distillation where a student model (D-ViT) learns from a teacher model (ResNet50) using either soft or hard distillation loss.

**Figure 4** GUI of the desktop application.

**Figure 5** Representative image samples for (a) Fire, (b) Smoke, and (c) Normal Scenarios.

**Figure 6** (a) Training and validation loss, (b) Training and validation accuracy, plots corresponding to a learning rate of 0.000025 and Adam optimizer for a D-ViT model.

**Figure 7** Confusion matrix plot for test data classification using D-ViT model.

**Figure 8** Cumulative performance in terms of Average Precision, Average Recall, and Average F1-score.

**Figure 9** Qualitative results of the proposed framework on representative test samples for each of the three classes (Fire, Smoke, Normal).

**Figure 10** Confusion matrix plot for the temporal data evaluation using D-ViT model.

**Biographies:**

**Hassan Akbar** received a bachelor's degree in electrical engineering, specializing in computer engineering. He performed this research while working as a Research Associate at the National Centre of Robotics and Automation (NCRA), National University of Sciences and Technology (NUST), Pakistan. His work focuses on creating solutions for real-world problems using AI. Specifically, he has made contributions by developing AI-based solutions tailored for edge devices. His research interests include computer vision, deep learning, and embedded systems.

**Tahir Nawaz** is Associate Professor and Head of Research at NUST CEME, and Founding Co-Director of the NUST-Staffordshire Creative AI Lab. With over 15 years of academic and industrial experience across the UK, Europe, and Asia, he specializes in computer vision and AI. He previously led R&D projects at Jaguar Land Rover and the University of Reading, and currently consults for Kodifly Limited. His research focuses on multi-modal sensing, surveillance, and autonomous systems.

**Md Asaduzzaman** is an Associate Professor at Staffordshire University. He earned BSc and MSc degrees in applied statistics from the University of Dhaka, an MSc in bioinformatics from Chalmers University, and a PhD in operational research from the University of Westminster. His research focuses on queueing theory, optimisation, and statistical modelling. He is Associate Editor for IEEE Access, Senior Member of IEEE, Associate Fellow of OR Society UK, and Fellow of the Royal Statistical Society.

**Mohammad S. Hasan** is a Senior Lecturer at the University of Staffordshire, UK. He received his BSc and MSc in Computer Science. He obtained his second MSc in Computer and Network Engineering from Sheffield Hallam University, UK and PhD from Staffordshire University, UK in Networked Control Systems over Mobile Ad-hoc Network (MANET). His research interests include Cybersecurity, Cloud Computing, AI, data analytics. He has been program/technical committee member for several conferences e.g. ICEIS, ICCDBC, ICCIP.

**Waqar Shahid Qureshi** is a lecturer at the University of Galway, Ireland, specializing in computer vision, AI, and embedded systems. His research focuses on intelligent transport, precision agriculture, and climate-resilient technologies. He serves on international program committees and collaborates across academia and industry to develop AI-driven solutions for real-world challenges. Dr. Qureshi has extensive experience in teaching and supervising postgraduate students in areas related to programming, image processing, and computer networks.

**Faisal Shafait** is a Professor at the National University of Sciences and Technology (NUST), Islamabad. His research interest is in machine learning, primarily focusing on computer vision and document image analysis. He received the prestigious IAPR/ICDAR Young Investigator Award in 2019. He is serving on the editorial board of leading international journal including IJCV, PR, and IJDAR as well as Area Chair of conferences including ICPR, ICDAR, DAS, and ICFHR.