

A Novel Network-Level Fusion Architecture of Proposed Self-Attention and Vision Transformer Models for Land Use and Land Cover Classification From Remote Sensing Images

Saddaf Rubab ¹, Muhammad Attique Khan ², *Member, IEEE*, Ameer Hamza ³, Hussain Mobarak Albarakati ⁴, Oumaima Saidani ⁵, Amal Alshardan ⁶, Areej Alasiry ⁷, Mehrez Marzougui ⁸, and Yunyoung Nam ⁹

Abstract—Convolutional neural networks (CNNs), in particular, demonstrate the remarkable power of feature learning in remote sensing for land use and cover classification, as demonstrated by recent deep learning techniques driven by vast amounts of data. In this work, we proposed a new network-level fusion deep architecture based on 16-tiny Vision Transformer and SIBNet. In the initial phase, data augmentation has been performed to resolve the problem of data imbalances. In the next step, we proposed a self-attention bottleneck-based inception CNN network named SIBNet. In this network, two architectures are followed. The blocks are designed using inception architecture, and each inception module is created with bottleneck blocks. The 16-tiny vision transformer architecture has been implemented for RS images and fused using a network-level fusion with SIBNet for the first time. Hyperparameters of the proposed model have been initialized using Bayesian Optimization for better training on the RS images. After the fusion, the model was on RS image datasets and extracted deep features from the self-attention layer. The extracted features are classified using a neural network classifier with multiple hidden layers. The

experimental process of the proposed architecture has been performed on two publically available datasets, such as EuroSAT and NWPU, and obtained an accuracy of 97.8 and 98.9%, respectively. A detailed ablation study has been performed to test the proposed models and shows that the fusion model achieved improved accuracy. In addition, a comparison is conducted with recent techniques and proposed methods, showing improved precision, recall, and accuracy.

Index Terms—Augmentation, explainable AI, fusion, land cover, remote sensing (RS), self-attention, vision transformer (ViT).

I. INTRODUCTION

CLIMATE change has led to a drastic growth in the popularity of Earth observation via remote sensing (RS) [1]. These tasks contain land use/land cover classification, mineral mapping, target detection, monitoring of the environment, planning of urban areas, and disaster management [2]. These areas were explored over the past decade using the EO sensors and hyperspectral images; however, such single-source information is insufficient to recognize objects of interest accurately [3]. The recent advancement in RS is based on the new technology gathered data by satellite drones and airplanes for research and development purposes [4]. Data gathering using these advanced technologies, especially for RS, attracted computer vision researchers to develop automated systems that accurately recognize the land cover or land use from the RS images [5]. The success of deep learning in medical imaging, image classification, and image retrieval motivates the researchers of RS to develop new applications at scale [6].

The first challenge for the RS data is high resolution due to a lot of variability [7]. The machine learning methods learned the RS data, but due to high variability, they did not extract enough information and degraded the classification performance [8]. The main challenge behind such performance degradation is traditional approaches that extract the information from the handcrafted features. The conventional handcrafted features do not sufficiently provide better results due to unseen data [9]. The neural networks have shown some improvement in the land cover and land use classification results using RS data; however, the desired accuracy is not achieved due to the change in hidden layers [10]. Several neural networks introduced in the literature

Manuscript received 15 March 2024; revised 13 June 2024; accepted 2 July 2024. Date of publication 11 July 2024; date of current version 5 August 2024. This work was supported in part by the Korea Institute for Advancement of Technology (KIAT), funded by the Korea Government (MOTIE) under Grant P0012724, through the HRD Program for Industrial Innovation, in part by the National Research Foundation of Korea (NRF), funded by the Korea Government (MSIT) under Grant RS-2023-00218176, in part by the Soonchunhyang University Research Fund, and in part by the Deanship of Research and Graduate Studies at King Khalid University, through Large Research Project under Grant GP2/283/45. This research was funded by: Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2024R507), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. (*Corresponding authors: Yunyoung Nam; Muhammad Attique Khan.*)

Saddaf Rubab is with the Department of Computer Engineering, College of Computing and Informatics, University of Sharjah, Sharjah 00000, UAE.

Muhammad Attique Khan and Ameer Hamza are with the Department of Computer Science, HITEC University, Taxila 47080, Pakistan (e-mail: attique.khan@ieee.org).

Hussain Mobarak Albarakati is with the Department of Computer and Network Engineering, College of Computing, Umm Al-Qura University, Makkah 24382, Saudi Arabia (e-mail: hbarakati@uqu.edu.sa).

Oumaima Saidani and Amal Alshardan are with the Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh 11671, Saudi Arabia.

Areej Alasiry and Mehrez Marzougui are with the College of Computer Science, King Khalid University, Abha 61413, Saudi Arabia (e-mail: areej.alasiry@kku.edu.sa; mhrez@kku.edu.sa).

Yunyoung Nam is with the Department of ICT Convergence, Soonchunhyang University, Asan 31538, South Korea (e-mail: ynam@sch.ac.kr).

Digital Object Identifier 10.1109/JSTARS.2024.3426950

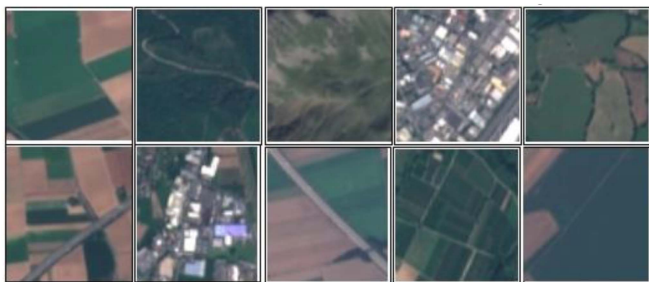


Fig. 1. Sample RS images for land use and land cover classification.

include three hidden layers, four hidden layers, five hidden layers, and ten hidden layers. However, the desired accuracy has not been achieved due to changes in the data variability and low resolution of image pixels [11].

The recent advancement in deep learning for several computer vision applications, such as medical image classification [12], agriculture diseases recognition [13], video surveillance [14], and object classification, has shown significant performance for both detection and classification [15], [16]. The neural networks with 1–10 hidden layers are fixed when many datasets have been used for the training. Therefore, the deep-learning models performed well on large datasets due to hundreds or thousands of hidden layers [17]. The deep-learning models required a large amount of data for training purposes. The one successful area in computer vision is convolutional neural network (CNN) [18]. A CNN model consists of several hidden layers, such as a convolutional layer [19], Relu activation layer [20], pooling layer for the downsampling [21], batch normalization layer [22], fully connected layer [23], and SoftMax layer [24]. The convolutional layer was added for local-level information extractions; however, the global average pooling and fully connected layers extracted the global information of an image [25]. Hence, the CNN model extracted an image's most salient features, which later obtained improved recognition accuracy [26].

Training a CNN model from scratch is difficult due to the availability of good resources, such as GPU, and requires a large amount of data. Therefore, the researchers of CV used the pretrained CNN architecture for the training on transfer learning concepts, especially for RS images [27]. Several pretrained models used for RS and object classification exist in the literature, such as VGG16, AlexNet, ResNet, and EfficientNet. Due to common characteristics and patterns, the RS data for land use and land cover classification is complex and difficult to recognize. As shown in Fig. 1, a few sample RS images are shown. This figure demonstrates that the images of the different class patterns (highway, industrial area, permanent crops, and forest) fall in the other class; hence, the inter and class challenges exist [28]. Moreover, a more complex deep-learning model is required to extract the local information for these kinds of images.

Several techniques are introduced in the literature to classify land use and land cover from RS images [3], [5]. Ma et al. [29] presented a feature enhancement network called FENet for land cover classification using RS images. In the presented method, they focused on edge contour enhancement and added

a self-attention module that extracted local information of an image. Ultimately, they fused the information at different scales and obtained an improved land cover classification accuracy of 82.85%. Hu et al. [30] presented an encoder–decoder multi-guided CNN architecture for land cover classification using RS images. They focused on fusing simple CNN and self-attention modules for abstract-level information extraction. Zhang et al. [31] presented an improved encoder–decoder U-NET architecture for land cover classification using RS images. They initially added an encoder–decoder part and connected it with an attention module for more accurate local information extraction. Regan et al. [32] presented a fusion framework of conventional features and optimized the performance using a whale optimization algorithm. They classified the final features using the LSTM model and performed an experimental process on the EuroSAT dataset. In addition to that, they also highlight the issue of overfitting due to conventional features. Papoutsis et al. [25] presented a deep-learning model based on the vision transformer (ViT) and efficient net architecture. They also consider the wide residual network for the comparison purposes of the presented vision model. Temenos et al. [33] introduced an interpretable CNN architecture based on SHAP for land use and cover classification. The authors used a customized CNN architecture in the presented method and analyzed the performance using an explainable AI method named SHAP. They used the EuroSAT dataset and obtained an improved accuracy of 94.72%. Vinaykumar et al. [34] presented an optimal guidance WOA optimization algorithm to reduce irrelevant features. Using transfer learning techniques, they used pretrained models, such as AlexNet and ResNet50, for feature extraction. The extracted features are optimized using the presented optimization algorithm. The presented method obtained 95.21% accuracy on the NWPU RS imaging dataset. Xie et al. [35] presented an RS image classification task based on the label augmentation and Kullback–Leibler divergence method. They focused on the importance of image augmentation for deep-learning models and created diversity in generated images. The NWPU challenging dataset was used and obtained an accuracy of 93.6%. Xu et al. [36] presented a graph CNN architecture named DFAGCN for land scene classification using RS images. They used the power of graph neural network that extracts the most important information, especially for the RS images. They also introduced a weighted concatenation method for convolutional layer features and fully connected layer features, respectively. The semantic classifier was employed for the classification and obtained the improved accuracy of 93.05% on the NWPU dataset.

Advancements in deep learning, such as self-attention modules and ViTs, can efficiently work for billions of parameters. In the ViTs, the images are broken into 2-D patches, and then an image's most important (local) features are extracted. However, the ViTs faced a large dataset's generalizability, complexity, model size, and scalability. On the other hand, the self-attention-based bottleneck architecture was trained on the large-scale dataset, and local information was extracted. Therefore, in this work, we consider the ViT and self-attention bottleneck module and fuse them at a network level that combines the information of local and global RS imaging data.

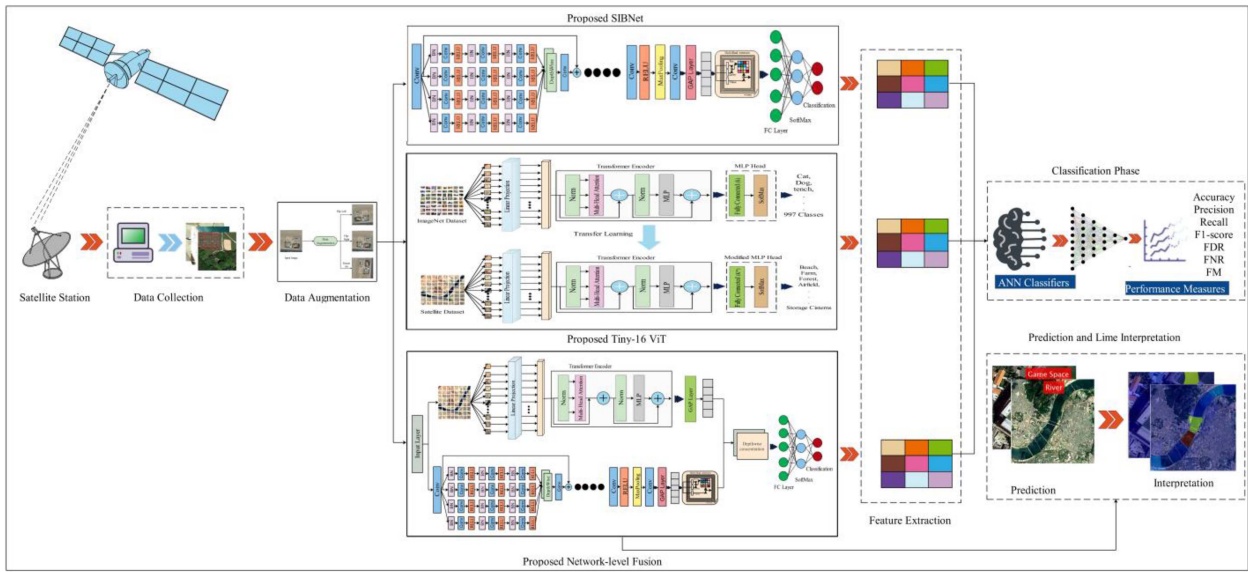


Fig. 2. Proposed framework for the classification of land use and land cover based on network-level fusion using RS images.

The contribution is given as follows.

- 1) The selected datasets have an imbalance problem that creates biases during training models. To overcome this problem, we proposed an augmentation step based on the traditional functions for a few sample classes.
- 2) We designed a self-attention bottleneck-based inception CNN network named SIBNet that handles the challenge of dataset complexity. In this network, two architectures are followed. The blocks are designed using inception architecture, and each inception module is created with bottleneck blocks.
- 3) Proposed a tiny-16 ViT to resolve the challenge of generalizability and important feature extraction. The proposed ViT is fined-tuned using the transfer learning method.
- 4) We proposed a new network based on an internal fusion of SIBNet and tiny-16 ViT using a depthwise concatenation layer and compared it with recently proposed models. The proposed fusion network improves the accuracy and reduces the learning parameters.

The rest of this article is organized as follows. The proposed network-level fusion model is described in Section II that includes augmentation, ViT model, and proposed SIBNet. The results are presented in Section III that includes individual results and proposed fusion results. In addition, a detailed discussion has been added under this section. Finally, Section IV concludes this article.

II. PROPOSED METHODOLOGY

This section presents a deep-learning-based framework for the classification and identification of land use and land cover classification, as shown in Fig. 2. This figure illustrates that the satellite land cover dataset is used for experiments. Initially, the selected datasets are augmented to balance the sample images in each class. After that, a self-attention inception bottleneck-based CNN, SIBNet, is designed. In addition, a ViT tiny-16 was also

implemented, and training was performed. Both models are trained using selected datasets. Following that, a network-level fusion is performed by utilizing the SIBNet and tiny-16 network to improve the framework's performance. Deep features are extracted from the fused model and passed to a neural network classifier for the final classification. Furthermore, the trained fused model is employed for the prediction, and to understand the predictions of models, we employed the locally interpretable model-agnostic explanation (LIME) explainable AI algorithm to generate the feature importance map and highlight the image regions most influential. A detailed description of each step is given as follows.

A. Data Collection and Augmentation

In this work, we used two publicly available datasets for experimental purposes. The utilized datasets are Eurostat¹ and NWPU_RESISC45.² Eurostat contains ten classes: Annual crop, forest, herbaceous vegetation, highway, industrial, pasture, permanent crop, residential, river, and sea lake. These were collected from a sentential satellite with a dimension of $62 \times 64 \times 3$ using a ground sampling distance of 10 m. The other dataset, NWPU_RESISC45, consists of 12 classes: airfield, harbor, beach, dense residential, farm, overpass, forest, game space, parking space, river, sparse residential, and storage tanks.

Each image has a dimension of $256 \times 256 \times 3$ with 96 × 96 dpi. A few samples of the selected dataset are shown in Fig. 3.

The Eurostat dataset has 27 000 samples, and the NWPU_RESISC45 dataset has 10 500 samples. A few of classes from both datasets are imbalanced. The misbalancing problem may create bias in the network. To prevent this problem, the selected datasets are augmented using three basic operations:

¹[Online]. Available: <https://www.kaggle.com/datasets/apollo2506/eurostat-dataset/>

²[Online]. Available: https://figshare.com/articles/dataset/NWPU_RESISC45_Dataset_with_12_classes/16674166



Fig. 3. Few samples of satellite images for the classification of land use and land cover.

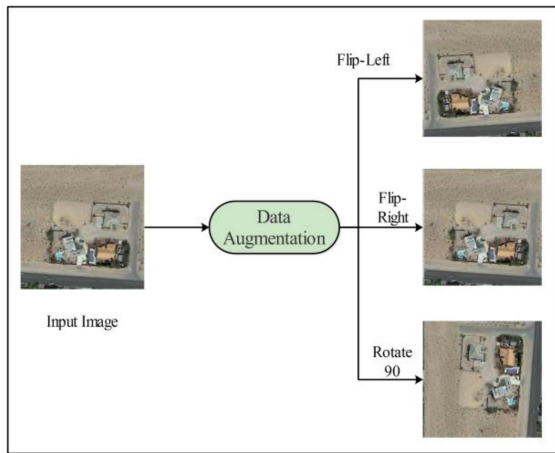


Fig. 4. Data augmentation samples using three operations.

flip left, flip right, and rotate 90. Suppose that the input image is denoted by $\mathcal{S}_{(a,b)}$ with the size of $x \times y \times d$, where x and y are presented as the row and column of the image. The symbol d denoted the channels of the input image. The three operations are performed: flip left, flip right, and rotate 90°. These operations are mathematically defined by (1)–(3)

$$\mathcal{S}_{(a,b)}^{\rightarrow} = \mathcal{S}_a (y + 1 - b) \quad (1)$$

$$\mathcal{S}_{(a,b)}^{\leftarrow} = \mathcal{S}_b (x + 1 - a) \quad (2)$$

$$\mathcal{S}_{(a,b)}^{90^\circ} = \begin{bmatrix} \cos 90^\circ & -\sin 90^\circ \\ \sin 90^\circ & \cos 90^\circ \end{bmatrix} \begin{bmatrix} \mathcal{S}_a \\ \mathcal{S}_b \end{bmatrix} \quad (3)$$

where $\mathcal{S}_{(a,b)}^{90^\circ}$ denoted the rotation 90° operation, $\mathcal{S}_{(a,b)}^{\leftarrow}$ presented the flip left, and $\mathcal{S}_{(a,b)}^{\rightarrow}$ denoted the flip-right operation. This operation is performed on classes with fewer images than the other classes of the datasets. The augmentation operations are visually shown in Fig. 4.

B. Proposed SIBNet

The Inception module, also known as GoogleNet [37], is an extensively implemented component in CNNs, and its initial implementation appeared in the Inception architecture. The objective of the concept was to capture features at diverse spatial resolutions by integrating filters of different dimensions

within a single layer [38]. This process enables the network to efficiently acquire hierarchical visuals of the input data. The fundamental principle underlying the Inception module is the concurrent application of convolutions with filters of different dimensions, followed by the fusion of the resultant outputs [39]. By employing multiple filter sizes, the network can collect features in a manner that is computationally more efficient than utilizing a single-filter size. By promoting the equilibrium of the receptive field, the design enables the network to gather elements of varying sizes and complexity [40]. In this work, we create a customized self-attention bottleneck-based inception CNN named SIBNet for RS image classification. The proposed network consists of three inception modules. Each inception module is designed with bottleneck blocks. The network starts with an input size of 256×256 with a depth of 3. The first inception module is initiated with a convolutional layer with a 3×3 kernel size, 16 depth, and 1×1 stride. In the first inception module, the first bottleneck block is started from the batch normalization layer, and the convolutional layer that has a 2×2 filter size, 1×1 stride, ReLU, and batch normalization is attached. After that, another convolutional layer, ReLU, and batch normalization layer are attached. The configuration of these layers is that the kernel size is 3×3 , the stride is 1×1 , and the padding is the same. The other parallel bottleneck blocks follow the same phenomena as the first one.

Moreover, the information on four bottleneck blocks is concatenated depthwise. An immediate convolutional layer has been attached using 3×3 kernel size, 1×1 stride, and 16 depths. Following that, the information depthwise concatenated is fused with the first convolutional layer. The other two inception modules follow the same procedure as the first one. A global average pooling layer is attached, and a flattened layer is utilized to convert the dimensionality of the feature map into 1-D. In addition, a self-attention layer is attached to extract the prominent information from the network. At the end of SIBNet, a fully connected SoftMax and classification layer are attached to complete the network. Cross entropy is adapted as a loss function for the proposed network. After designing the SIBNet, we trained the model on selected datasets and extracted the prominent features from the self-attention activation. Self-attention [41] involves the dynamic computation of attention weights for each input pair, which is dependent on their respective content. This enables the model to concentrate on the most pertinent aspects of the input for each particular position, resulting in a more adaptable and contextually aware representation. Additionally, it can capture extensive connections and dependencies between all elements of the input sequence, regardless of their spatial separation.

The SIBNet has 80 weighted layers from 138 layers with 7.7M parameters. The architecture of SIBNet is visually illustrated in Fig. 5.

C. Vision Transformer

ViT is a neural network model that converts image inputs into feature vectors using the transformer architecture. The backbone and the head are the two fundamental parts of the ViT [42]. The encoding procedure of the network imposes constraints on the

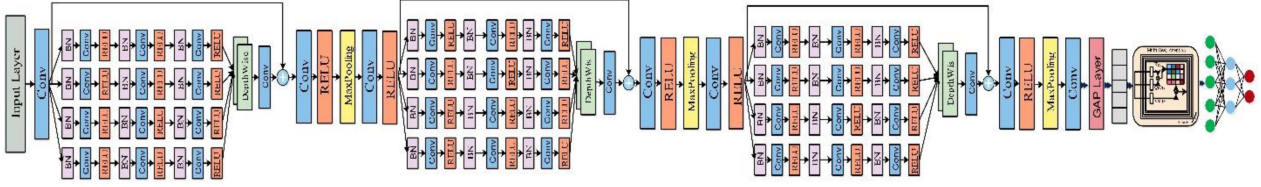


Fig. 5. Proposed SIB architecture for the classification of land scene.

backbone. The backbone receives the input images and produces an output feature vector. The head uses the encoded feature vectors to generate the prediction scores [42], [43]. Suppose $Q = \{\alpha, \beta\}_{i=1}^S$ be a collection of S land scene images, where α_i is an image and β_i is the corresponding label and β_i belongs to $\{1, 2, 3, \dots, c\}$, where c is the number of specified classes for that particular set. The ViT model aims to obtain the ability to map a series of visual patches to their corresponding label. Traditionally, the ViT comprises components of an embedding layer, encoder, and MLP head classifier. Initially, an image I from the training samples was divided into nonoverlapping patches. Every patch is fed to the transformer as a linear vector. So, image I is with a dimension of $r \times c \times d$, where r , c , and d are the row, column, and depth of the input image. The patch is extracted for each dimension using $d \times P_s \times P_s$. The sequence of patches $(p_1, p_2, p_3, \dots, p_k)$ is of length k . The patches are calculated using $k = rw/p^2$, where p is the patch size. Normally, the chosen patch size is 16×16 or 32×32 ; the small patch size generates the longer sequence and vice-versa.

Before passing the sequence of patches into the encoder, it is linearly projected onto an array of the model dimension ∂_d using a learned embedding matrix M_e . The embedded instances are combined by joining them with a learnable classification token $\mathcal{L}_{\text{class}}$, which is essential for the classification task. The transformer perceives the embedded image patches as an unordered set, not considering their arrangement. To maintain the positioning of the patches as seen in the original image, the positional data $\mathcal{E}_{\text{position}}$ is encoded and associated with the patch visualizations. The embedding sequence of patches is mathematically defined as follows:

$$\begin{aligned} \tau_0 = & [\mathcal{L}_{\text{class}}; p_1\epsilon; p_2\epsilon; \dots; p_k\epsilon] \\ & + \mathcal{E}_{\text{position}}, \epsilon \in \mathbb{R}^{(p^2d) \times \partial_d}, \mathcal{E}_{\text{position}} \in \mathbb{R}^{(k+1) \times \partial_d}. \end{aligned} \quad (4)$$

The output of embedded patches τ_0 is passed to the transformer encoder. The encoder part consists of \mathcal{f} identical layers. Each component has two primary subdivisions: a multihead self-attention block and a fully connected feedforward dense block (MLP). The last block includes two dense layers with a GeLU activation function in between. Both parts of the encoder utilize residual skip connections and are followed by a normalization layer (\mathcal{f}_{BN}). Equations mathematically formulate as follows:

$$\tau'_f = \text{MSA} \left(\mathcal{f}_{\text{BN}}(\tau_{f-1}) \right) + \tau_{f-1}, f = 1, \dots, \mathcal{f} \quad (5)$$

$$\tau_f = \text{MLP} \left(\mathcal{f}_{\text{BN}}(\tau'_f) \right) + \tau'_f, f = 1, \dots, \mathcal{f}. \quad (6)$$

In the final stage of the encoder, we extract the initial element in the sequence $\tau_{\mathcal{f}}^0$ and feed it to an outer head classifier for predictions using the following equation:

$$y_{\text{output}} = \mathcal{f}_{\text{BN}}(\tau_{\mathcal{f}}^0). \quad (7)$$

In this work, we utilized a pretrained ViT named tiny-16 for classification purposes. Original tiny-16 was trained on the ImageNet dataset, which has 1000 object classes and 143 layers with 5.7 M parameters. This model accepts the $384 \times 384 \times 3$ size of the input image. We utilized the tiny-16 model for feature extraction. First, we trained the tiny-16 model using the deep transfer learning concept. We replaced the last three layers, namely indexing, fully connected, and softmax, with a new global average, new fully connected, and new softmax layer. After training, the global average activation is utilized for feature extraction from the trained ViT. The dimensions of extracted features are $N \times 192$. The architecture of the tiny-16 ViT is presented in Fig. 6.

D. Proposed Network-Level Fusion

In this work, we proposed a network-level fusion technique. A hybrid model combines a CNN [44] and a ViT [45] model. The purpose of the fusion process is to solve computer vision problems synergistically by leveraging the capabilities of each architecture. The CNN is exceptionally adept at discerning localized details and patterns within images [46], thereby establishing a solid foundation for identifying specific visual objects. On the other hand, the ViT demonstrates exceptional proficiency in discerning the interrelationships and contextual significance of disparate images, thereby facilitating a holistic understanding of the scene [47]. By integrating these two architectural designs, the hybrid model generates an intricate depiction of the input data by seamlessly integrating local and global features [48]. Incorporating these components leads to enhanced representation learning, providing the model with a more holistic understanding of complicated visual scenarios. A significant advantage is the hybrid model's adaptability to different task specifications; it can utilize either local or global features based on the particular demands of the task. Furthermore, by efficiently allocating resources, the hybrid model strikes a balance between computational efficiency and model efficacy, rendering it a feasible optimal for a multitude of computer vision applications.

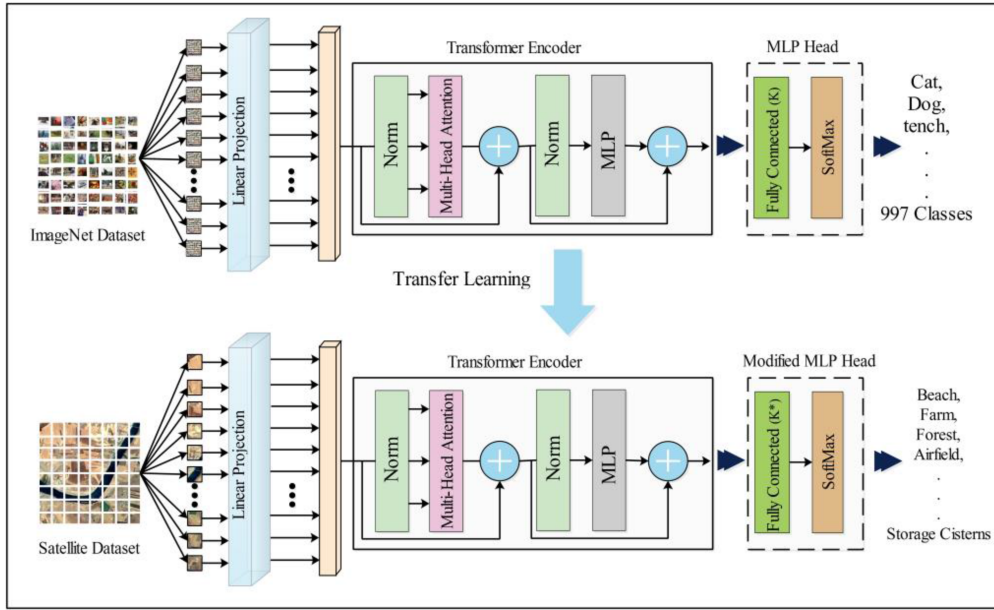


Fig. 6. Tiny ViT with transfer learning for RS image classification.

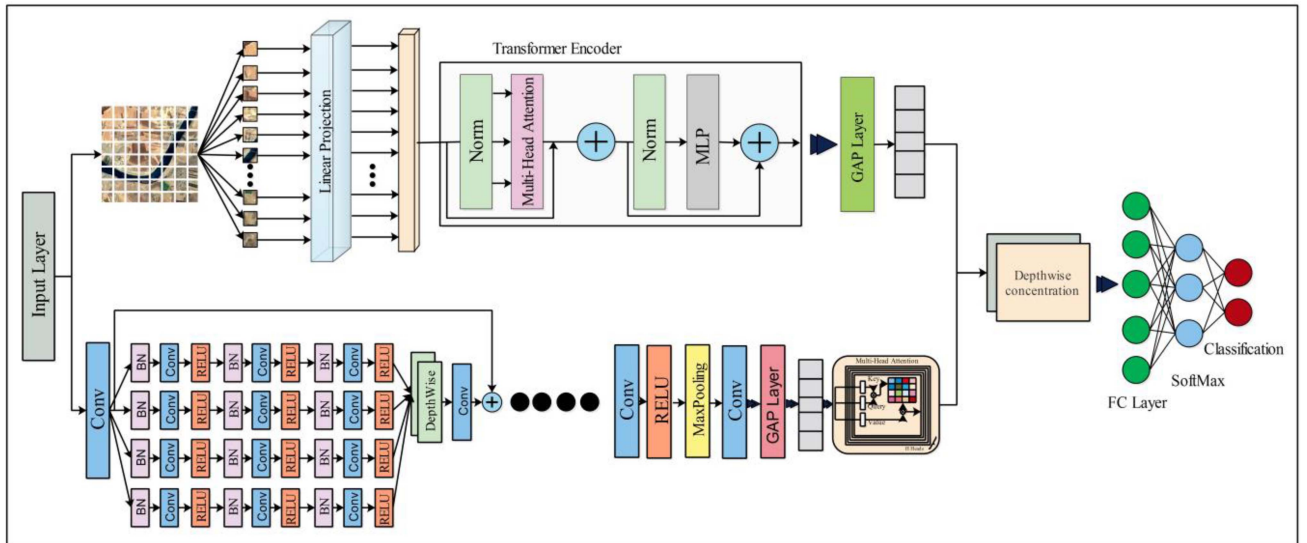


Fig. 7. Visual illustration of the proposed internally fused architecture for RS image classification.

In this work, the proposed SIBNet and tiny-16 ViT are internally fused into a single architecture. Through this process, the local and global information enhances the model's efficiency and improves generalizations in complex scenarios. The last three layers of both models are removed and fused to both architectures using a depthwise concatenation layer. The proposed hybrid model accepts an image size of $384 \times 384 \times 3$. The total parameters of the proposed hybrid architecture are 11.5 M. After designing, the proposed model is trained on the selected datasets, and the features are extracted from the depthwise concatenation layer. The dimensions of extracted features are $N \times 1216$. The architecture of the proposed hybrid model is visually presented in Fig. 7.

III. EXPERIMENTAL RESULTS AND DISCUSSION

A. Experimental Setup

The experimental results of the proposed framework are presented in this section. The two datasets were selected and are described in Section III-A. Initially, the chosen datasets were separated by a 50:50 ratio, indicating that 50% of the data are used for training, and the other 50% are utilized for the testing phase. To better generalize the proposed model, we selected the k -fold validation 10, effective from all the performed trials. The K -fold cross validation is used to balance processing costs and variance, which impact the accuracy of the efficiency estimate. We decided on a value of $k = 10$. The experiment used $N \times 1216$

TABLE I
CLASSIFICATION RESULTS OF THE PROPOSED TINY-16 ViT ON THE EUROSAT DATASET

Classifier	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	FDR (%)	FNR (%)	FM (%)	Time (s)
NNN	95.56	95.69	95.62	95.7	4.44	4.34	95.62	49.066
MNN	96.31	96.28	96.29	96.4	3.69	3.72	96.29	34.559
WNN	96.44	96.28	96.35	96.6	3.56	3.72	96.35	32.417
BNN	95.37	95.71	95.53	95.3	4.63	4.29	95.53	59.422
TNN	95.11	95.17	95.13	95.3	4.89	4.83	95.13	77.377

feature dimensions. The value of k was selected based on the observations and features, and it was observed that the lower values were not effective. The hyperparameters for training the proposed model were learning rate, minibatch size, epochs, and optimizer having values of 0.0025, 64, 30, and SGDM. The proposed model is evaluated using precision, recall, $F1$ -score, accuracy, false discovery rate (FDR), false negative rate (FNR), Fowlkes–Mallows (FM) index, and computation time in seconds. The FDR, FNR, and FM were calculated using (8)–(10). All the experiments were conducted on MATLAB R2023b using a core i7-12Gen Desktop configured with 128 GB RAM, 512 SSD, and 12 GB NVIDIA 3060ti graphic card

$$\text{FDR} = 1 - \text{PPV} \quad (8)$$

$$\text{FNR} = 1 - \text{TPR} \quad (9)$$

$$\text{FM} = \sqrt{\text{TPR} \times \text{PPV}} \quad (10)$$

where PPV presented the positive predictive values, and TPR presented the true positive rate.

B. Results Phase

The results of the proposed framework are described in three phases of each dataset. The first phase describes the tiny-16 ViT, the second phase presents the proposed SIBNet results, and the last phase presents the proposed internally fused model classification results. Below are the results of each dataset.

1) *EuroSAT Dataset Results*: This experiment extracts the features from the modified tiny-16 ViT model, and classification is performed. Initially, the tiny-16 was fine-tuned and trained on the EuroSAT RS dataset. Table I describes the classification results of this dataset, which achieved the highest accuracy of 96.6% for wide neural network (WNN). For this classifier, a few more performance measures are computed, such as precision, recall, $F1$ -score, FDR, FNR, FM, and time. The values of these measures are 96.44%, 96.28%, 96.35%, 3.56, 3.72, 96.35%, and 32.417 (s). The medium neural network (MNN) was done with the second-best accuracy at 96.4%. The remaining classifiers had a 95.7% and 95.3% accuracy rate, respectively. Fig. 8 shows the number of observations obtained by the WNN classifier. The diagonal values show the correct classified value of each class. Furthermore, the computational time of each classifier is noted; the best time for WNN is 32.417 (s), while the worst time for TNN is 77.377 (s).

The SIBNet is created and trained on the EuroSAT dataset in the second phase. The classification results of SIBNet have been

AnnualCrop	1439	3	2	4		9	32		6	5
Forest	1	1478	7	1		6	3			4
HerbaceousVegetation	2	5	1435	2		13	33	4	3	3
Highway	5	2	1	945	6	1	4	5	31	
Industrial			1	6	1224		5	13	1	
Pasture	9	3	16	1	1	963	4		1	2
PermanentCrop	35		43	7	3	4	1151	3	4	
Residential			2	4	13		3	1478		
River	3	3	1	29	1	4	1	1	1201	6
SeaLake	4	3	3	1		2			7	1480

Fig. 8. Confusion matrix of tiny-16 ViT model for WNN classifier using EuroSAT dataset.

presented in Table II. From this table, the WNN achieved the highest accuracy of 88.9%. Additional performance measures are precision, recall, $F1$ -score, FDR, FNR, FM, and time complexity, having values of 87.9%, 88.0%, 87.9%, 12.1, 12.0, 13.2, and 20.96 (s). With an accuracy of 87.4%, the MNN achieved the second-best result. Concerning accuracy, the remaining classifiers scored 86.4%, 86.1%, and 85.9%, respectively. The number of observations of a WNN classifier for an SIBNet is shown in Fig. 9. The diagonal values in this figure indicate which prediction is accurate to classify. Each classifier's computational time is also given; for BNN, the best time is 12.18, while for NNN, the highest taken time is 60.1. Compare the results of this phase with phase 1; it is observed that the accuracy is not improved; however, the time is significantly reduced.

In the last phase, the SIBNet and tiny-16 architecture were internally fused to boost the performance of the proposed framework. The internally fused architecture results are presented in Table III. This table illustrates that the WNN classifier obtained the highest accuracy of 97.8%. The other parameters, including precision rate, recall rate, $F1$ -score, FDR, FNR, FM, and time, are also calculated for the classifier. The values of these performance measures are 97.0%, 96.97%, 96.98%, 3.0, 3.03, 96.98, and 16.14 (s), respectively. With 96.7% accuracy, the NNN came in second best. The remaining classifiers have accuracy rates of 96.6% and 96.5%, respectively, including BNN and TNN. A confusion matrix (seen in Fig. 10) is also provided to verify the results obtained further. Additionally, the computation time is

TABLE II
CLASSIFICATION RESULTS OF THE PROPOSED SIBNET MODEL FOR WNN CLASSIFIER USING THE EUROSAT DATASET

Classifier	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	FDR (%)	FNR (%)	FM (%)	Time (s)
NNN	84.9	84.9	84.9	85.9	15.1	15.1	13.0	60.1
MNN	86.4	86.4	86.4	87.4	13.6	13.6	5.21	36.65
WNN	87.9	88.0	87.9	88.9	12.1	12	13.2	20.96
BNN	85.0	85.0	85.0	86.1	15.0	15.0	13.0	12.18
TNN	85.4	85.3	85.3	86.4	14.6	14.7	13.06	28.51

Bold denotes the best values.

TABLE III
CLASSIFICATION RESULTS OF INTERNALLY FUSED ARCHITECTURE ON THE EUROSAT DATASET

Classifier	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	FDR (%)	FNR (%)	FM (%)	Time (s)
NNN	96.52	96.51	96.51	96.7	3.48	3.49	96.51	20.62
MNN	96.92	96.92	96.92	97.1	3.08	3.08	96.92	17.03
WNN	97.00	96.97	96.98	97.8	3.00	3.03	96.98	16.14
BNN	96.43	96.37	96.39	96.6	3.57	3.63	96.40	19.02
TNN	96.78	96.51	96.64	96.5	3.22	3.49	96.64	22.83

Bold denotes the best values.

True Class \ Predicted Class	AnnualCrop	Forest	HerbaceousVegetation	Highway	Industrial	Pasture	PermanentCrop	Residential	River	SeaLake
AnnualCrop	1337	5	14	33		20	40		36	15
Forest		1465	11	3		12			5	4
HerbaceousVegetation	16	7	1308	25	14	21	72	21	15	1
Highway	26	7	34	674	44	23	73	21	98	
Industrial			5	45	1166		10	22	2	
Pasture	19	9	21	14		892	17		26	2
PermanentCrop	37		81	76	5	12	1017	1	21	
Residential			22	12	29		1	1435	1	
River	37	8	16	102	4	32	25	2	1012	12
SeaLake	6	2		1		6			10	1475

Fig. 9. Confusion matrix of SIBNet model for WNN classifier on EuroSAT dataset.

also recorded for all the listed classifiers. The WNN classifier has a minimum time of 16.14 (s). In contrast, the TNN classifier has the worst time, which is 22.83 (s). The results show that the internal fusion technique significantly improved the accuracy and maintained the computational time. Moreover, the results of the proposed model are compared with a few pretrained models, as shown in Fig. 11. This figure shows that the proposed fused model accuracy is higher than the other pretrained networks. In the pretrained models, InceptionV2 and DenseNet201 obtained a precision rate of 90.62% and 89.56%, respectively. The proposed

True Class \ Predicted Class	AnnualCrop	Forest	HerbaceousVegetation	Highway	Industrial	Pasture	PermanentCrop	Residential	River	SeaLake
AnnualCrop	1444	2	1	3		9	33		6	2
Forest	1	1489	6			4				
HerbaceousVegetation	5	6	1445	4		10	24	2	2	2
Highway	4	1	2	951	6	2	4	5	25	
Industrial				6	1230		9	5		
Pasture	7	2	11	1		967	5		4	3
PermanentCrop	29	1	34	6	3	5	1170	1	1	
Residential			2	2	11		1	1484		
River	7	2	1	30	1	3	2		1200	4
SeaLake	3	1	2			3			4	1487

Fig. 10. Confusion matrix of the network-level fused-WNN classifier on the Eurostat dataset.

fused model improved accuracy by 97%, which is a strength of this model.

2) *NWPU Dataset Results:* The proposed tiny-16 model is fine-tuned and trained on the NWPU_RESISC45 dataset. After training, the features are extracted and passed to neural network classifiers. The results of the NWPU_RESISC45 dataset using the proposed modified tiny-16 ViT have been presented in Table IV. This table presents that the WNN classifier obtained the highest accuracy of 98.2%, and the other performance measure values are 98.65%, 98.78%, 98.71%, 1.35, 1.22, and 98.71%, respectively. The accuracy values of the remaining classifiers, such as NNN, MNN, BNN, and TNN, are 97.3%, 98.3, 96.7%, and 96.9%, respectively. Fig. 12 depicts a confusion

TABLE IV
PROPOSED CLASSIFICATION RESULTS OF TINY-16 ViT MODEL ON NWPU DATASET

CLASSIFIER	PRECISION	RECALL	F1-SCORE	ACCURACY	FDR	FNR	FM	TIME
NNN	97.19	97.19	97.19	97.3	2.81	2.81	97.18	15.991
MNN	97.19	98.28	97.73	98.3	2.81	1.72	97.72	25.044
WNN	98.65	98.78	98.71	98.2	1.35	1.22	98.71	15.956
BNN	98.55	96.52	97.52	96.7	1.45	3.48	97.52	16.547
TNN	96.56	96.71	96.63	96.9	3.29	3.29	96.63	10.687

Bold denotes the best values.

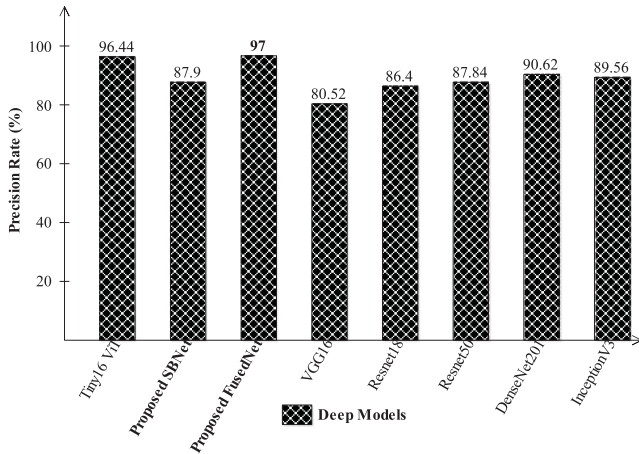


Fig. 11. Comparison of the proposed models' precision rate with pretrained deep-learning models on the EuroSAT dataset.

Airfield	692	1	3	1	3				
Anchorage	348				1	1			
Beach	1	1	346					1	
Dense Residential	2		339	2	3	1	1	1	
Farm	3			691				3	3
Flyover	2		1	346			1		
Forest				346			2	2	
Game Space				1	1	689	3	1	4
Parking Space	1	1			2	344			1
River	2		1	4		2	1	340	
Sparse Residential				1		4			344
Storage Cisterns	1	1		1	1				345

Fig. 12. Confusion matrix of WNN classifier using NWPU dataset for ViT model.

matrix used to validate the WNN classifier performance values. In addition, the computing time of each classifier is noted; the best time for TNN is 10.687 (s), while the worst time for MNN is 25.044 (s).

In the second step, the NWPU data are trained on the proposed SIBNet model, and features are extracted. The extracted features

Airfield	613	1	7	2	26	4		2	4	33		8
Anchorage	5	333	2	1	2		1	1	2	2		1
Beach	8	3	328		4		1			5		
Dense Residential	1			320				21	2		3	2
Farm	19	1	4		636	4	2	2		27	5	
Flyover	5			1	2	329		6	2		2	3
Forest				1	1		332	1		7	8	
Game Space	3	1		13	7	6	3	641	6	2	15	2
Parking Space	4	1			2			5	332		4	1
River	25	3	1		28		2	3		284	3	1
Sparse Residential	3			1	6	2	5	16		2	313	1
Storage Cisterns	5	1		1	2	5		3	1	1		330

Fig. 13. Confusion matrix of SIBNet model for WNN classifier on NWPU dataset.

were fed to neural network classifiers. Table V presents the classification results of the proposed SIBNet on the NWPU dataset. This table demonstrates that the WNN gained a higher accuracy of 91.4% from all the listed classifiers. The other parameters are also computed for this classifier, such as precision, recall, $F1$ -score, FDR, FNR, and FM, having 91.85%, 91.7%, 91.43%, 8.15, 8.3, and 91.63%, respectively. A confusion matrix of the WNN classifier can be used to verify the numerical statistics, as presented in Fig. 13. The accuracies of other classifiers are 88.2%, 90.7%, 88.0%, and 87.2%. Additionally, the computational times for each classifier are calculated, and it is observed that the WNN classifier has the shortest time of 10.85 (s). In contrast, the NNN classifier requires the highest time of 39.45 (s). Compared with Table IV, the proposed SIBNet model accuracy and precision rate are decreased; however, there is improvement noted in the computational time.

In the last step, the SIBNet and tiny-16 models are internally fused, and the proposed network-level fusion results are shown in Table VI. The tabular statistics clearly show that the WNN classifier achieved 98.9% accuracy, and the other parameter values are 98.26%, 98.13%, 98.19%, 1.74, 1.87, 98.19, and 11.242 (s), respectively. A confusion matrix is visually presented in Fig. 14 that can be used to verify numerical statistics further. The MNN obtained second-best accuracy of 98.2%. The remaining classifiers, NNN, BNN, and TNN, achieved 97.7%, 97.6%, and 97.3% accuracy. Computation time is also noted for all the listed

TABLE V
PROPOSED CLASSIFICATION RESULTS OF SIBNET MODEL ON NWPU DATASET

CLASSIFIER	PRECISION	RECALL	F1-SCORE	ACCURACY	FDR	FNR	FM	TIME
NNN	88.65	88.48	88.56	88.2	11.35	11.52	88.56	39.45
MNN	91.17	91.23	91.69	90.7	8.83	8.77	91.31	17.145
WNN	91.85	91.7	91.43	91.4	8.15	8.3	91.63	10.85
BNN	88.51	88.56	88.53	88.0	11.49	11.44	88.53	15.16
TNN	87.68	87.64	87.65	87.2	12.32	12.36	87.65	12.86

Bold denotes the best values.

TABLE VI
PROPOSED CLASSIFICATION RESULTS OF NETWORK-LEVEL FUSION ON NWPU DATASET

CLASSIFIER	PRECISION	RECALL	F1-SCORE	ACCURACY	FDR	FNR	FM	TIME
NNN	97.62	97.24	97.42	97.7	2.38	2.76	97.00	26.119
MNN	98.01	98.00	98.00	98.2	1.99	2.00	98.00	14.073
WNN	98.26	98.13	98.19	98.9	1.74	1.87	98.19	11.242
BNN	97.51	97.34	97.42	97.6	2.49	2.66	97.42	10.871
TNN	97.16	97.38	97.26	97.3	2.84	2.62	97.26	10.572

True Class	Airfield	Anchorage	Beach	Dense Residential	Farm	Flyover	Forest	Game Space	Parking Space	River	Sparse Residential	Storage Cisterns
Airfield	685	1	2		4					2	3	3
Anchorage	1	346			1				1	1		
Beach	4	1	342		1						1	
Dense Residential		1		344				3			1	
Farm					695		1		1	3		
Flyover	2			1		346				1		
Forest	1						346	1				2
Game Space				3			1	689	2		4	
Parking Space	2	2				1		1	342			1
River	3	1			3	1	1			341		
Sparse Residential				2	1	2	2	4			338	
Storage Cisterns	2	1										344

Fig. 14. Confusion matrix of the proposed network-level fused model on NWPU dataset.

classifiers. The minimum time is noted for the TNN classifier, which is 10.572 (s), and the longest time is recorded for the NNN classifier, which is 26.168 (s). Compared with the proposed fusion results with phases 1 and 2, it is clearly shown that the accuracy is improved and time is reduced. Moreover, we compared the results of the proposed models with a few pretrained networks on the selected datasets, as shown in Fig. 16. This figure shows that the proposed fusion network model obtained the highest precision rate of 98.9%, whereas the InceptionV3 model obtained a precision rate of 91.84%.

C. Discussion and Comparison With SOTA

A detailed discussion of the proposed deep architecture has been conducted in this section. The proposed architecture is

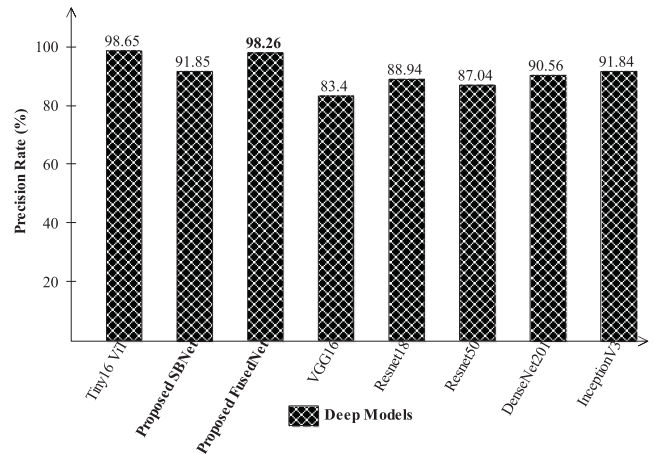


Fig. 15. Comparison of the proposed models' precision rate with pretrained deep-learning models on the NWPU datasets.

based on the network-level fusion of two inside models. The network-level fusion aim is to improve the performance of land use and land cover for RS images. As shown in Fig. 1, it is not easy to recognize the images in a correct class. Therefore, the proposed fusion architecture is designed, as illustrated in Fig. 2. Results are computed for each internal CNN model and checked for their strength in the form of performance. As presented in Tables I, II, IV, and V, the internal model results are presented for both datasets. In these tables, it is noted that the accuracy and precision rates of the ViT model have been improved; however, the computational time of the SIBNet model is better than the ViT. The proposed fusion results are presented in Tables III and VI. These tables have improved accuracy compared with the individual deep-learning models.

In addition to that, the computational time of the fusion process has been decreased. Confusion matrices are also illustrated

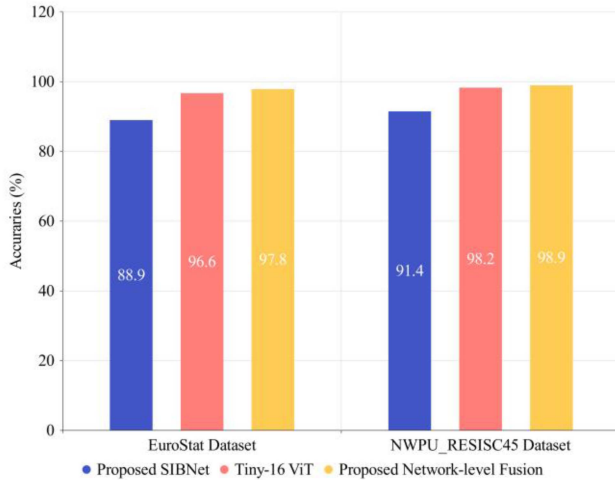


Fig. 16. Stepwise experimental accuracies using the proposed framework on selected datasets.

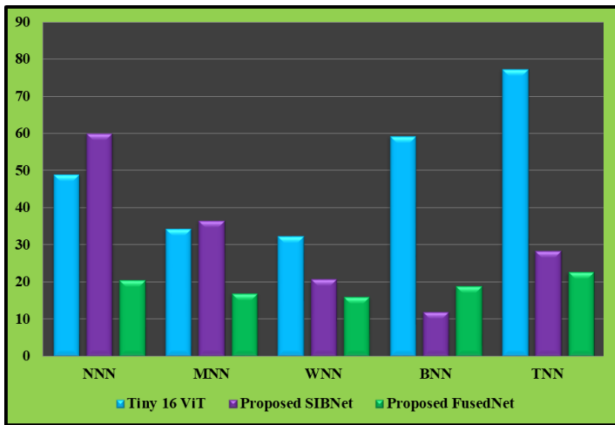


Fig. 17. Time comparison for EuroSAT dataset using three phases of classification results.

in Figs. 10 and 12–14, which can be utilized to confirm the best results. Figs. 11 and 15 show the comparison among the proposed fused and pretrained models' precision rates, and it is analyzed that the proposed precision rate has been improved. Overall, it seems that the primary strength of the proposed framework is network-level fusion. Accuracy-based comparison is also conducted in Fig. 16. On the EuroSAT dataset, the SIBNet achieved an accuracy of 88.9%, tiny-16 achieved 96.6%, and the fused model achieved 97.1% accuracy. On the NWPU_RESISC45 dataset, the classification results for SIBNet achieved 91.4%, tiny-16 ViT achieved 98.2%, and the fused model obtained 98.9% accuracy.

Figs. 17 and 18 show the time-based comparison among both datasets' proposed fused model and middle steps. In Fig. 17, the EuroSAT dataset computation time has been illustrated, and it is observed that the time of the proposed SIBNet has been increased compared with the ViT model; however, after the fusion network process, the time is significantly reduced. Similarly, for the NWPU dataset (see Fig. 18), the computation time for the proposed fused network has been less than the other methods (except NNN and WNN). Overall, the network-level

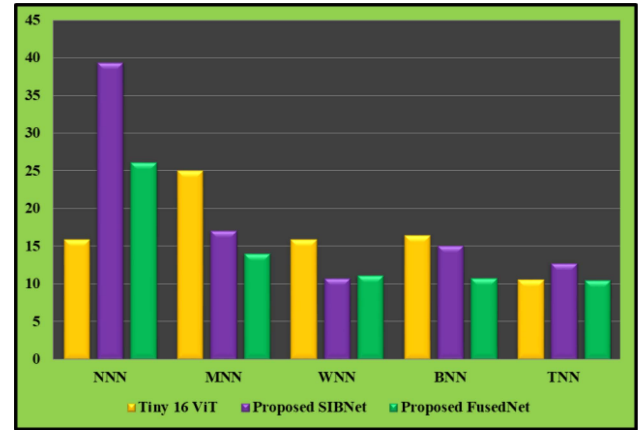


Fig. 18. Time comparison for NWPU dataset using three phases of classification results.

fusion process improves the accuracy precision rate and reduces the computational time.

The interpretability of the proposed fused model is also computed using an explainable AI method named LIME. The main objective of LIME is to provide localized explanations. It approximates the model around the particular occurrence that is being predicted in order to shed light on specific estimates. This feature is very helpful in understanding the individual decisions the model makes. However, Grad-CAM provides a general overview of the important factors that contribute to the prediction, but it may not provide detailed justifications for the classification of a particular instance. After training the proposed fused architecture, the trained model is employed for the predictions and to understand the features determining how our proposed network classifies a particular image. We utilized the LIME method. This approach generates a feature importance map, highlighting the image regions most influential in the network's evaluation of the class score for the designated label. This clarifies the decision-making process, confirming that the network focuses on relevant image features when classifying. Additionally, the LIME technique approximates the complex classification behavior of the proposed network via a simpler regression model. This surrogate model facilitates the identification of individual feature importance within the input image, elucidating their respective contributions to the network's classification score for the designated label. The results of LIME are visually presented in Fig. 19.

Finally, we compare the proposed model accuracy with several state-of-the-art (SOTA) techniques, as given in Table VII. In this table, Temenos et al. [33] obtained 94.72% accuracy on the EuroSAT dataset, which was later improved by Bhatt and V. T. Bhatt [49] to 95.16%. The proposed model obtained 97.8% accuracy on EuroSAT dataset, which is improved than the recent SOTA techniques. Xu et al. [36] used NWPU dataset for the evaluation process and obtained an accuracy of 93.05%, which was later improved by Xie et al. [35] of 93.60. Recently, Vinaykumar et al. [34] used NWPU dataset and obtained an accuracy of 95.21%. The proposed model obtained an improved accuracy of 98.9% and showed the strength of the fusion process.



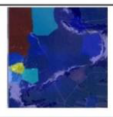


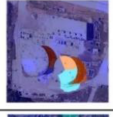


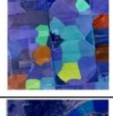


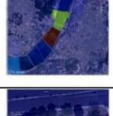



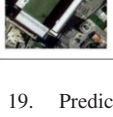

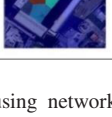
Input Image	Actual Label	Proposed Model Prediction	Correct/Incorrect	Lime Interpretation
	Beach		✓	
	Storage Cisterns		✓	
	Farm		✓	
	River		✓	
	Sparse Residential		✗	
	Game Space		✓	

Fig. 19. Predictions and LIME interpretation results using network-level fusion.

TABLE VII
COMPARISON OF THE PROPOSED MODEL ACCURACY WITH SOTA TECHNIQUES FOR SELECTED DATASETS

Method	Year	Dataset	Accuracy (%)
Temenos et al. [33]	2023	EuroSAT	94.72
Yamashkin et al. [50]	2020	EuroSAT	94.65
Aksoy et al. [51]	2023	EuroSAT	91.05
Vinaykumar et al. [34]	2023	NWPU	95.21
Xie et al. [35]	2021	NWPU	93.60
Xu et al. [36]	2021	NWPU	93.05
Bhatt and V. T. Bhatt [49]	2023	EuroSAT	95.16
Proposed	2024	EuroSAT	97.8
		NWPU	98.9

Bold denotes the best values.

IV. CONCLUSION

RS image classification is required for urban planning and agriculture applications. This article proposed a new network-level fusion deep architecture based on a self-attention mechanism. In the proposed architecture, data augmentation was performed initially for the imbalance issue. The imbalance problem is a challenge for accurately training a custom model. After that, we proposed a network-level fusion-based deep architecture that includes two CNN architectures, ViT and SIBNet. Both models are fused using a depth concatenation layer and append a self-attention layer. In the next step, the proposed model was trained on the selected datasets and deep features were extracted from the self-attention layer. Extracted features are classified

using neural network classifiers and obtained improved accuracy of 97.8% and 98.9% for EuroSAT and NWPU datasets. Overall, we conclude with the following points.

- 1) The augmentation process improved the performance of the proposed model more than the training accuracy on imbalanced datasets.
- 2) The proposed SIBNet model computational time is less than the 16-tiny ViT model. However, the individual performance of the ViT model has been improved.
- 3) Hyperparameter initialization using Bayesian optimization improved the training performance compared with manual initialization.
- 4) The proposed network-level fusion process significantly improved the classification accuracy and reduced the computational time, a strength of this work.

The limitation of this work is the assignment of stride value and filter size. These values can be interpreted using explainable AI in the future study. In this work, we just solved the proposed model after the network-level fusion. In addition, the proposed model will be further explored in the future on Earth hazard datasets.

DATA AVAILABILITY

The datasets of this work are publicly available for research purposes. The utilized datasets are Eurostat (<https://www.kaggle.com/datasets/apollo2506/eurosat-dataset/>) and NWPU_RESISC45 (https://figshare.com/articles/dataset/NWPU-RESISC45_Dataset_with_12_classes/16674166).

CONFLICT OF INTEREST

All authors declared no conflict of interest in this work.

REFERENCES

- [1] M. A. Moharram and D. M. Sundaram, "Land Use and Land Cover Classification with hyperspectral data: A comprehensive review of methods, challenges and future directions," *Neurocomputing*, vol. 536, pp. 90–113, Jun. 2023.
- [2] M. A. Moharram and D. M. Sundaram, "Dimensionality reduction strategies for land use land cover classification based on airborne hyperspectral imagery: A survey," *Environ. Sci. Pollut. Res.*, vol. 30, no. 3, pp. 5580–5602, 2023.
- [3] S. Dutta and M. Das, "Remote sensing scene classification under scarcity of labelled samples—A survey of the state-of-the-arts," *Comput. Geosci.*, vol. 71, 2023, Art. no. 105295.
- [4] A. Vali, S. Comai, and M. Matteucci, "Deep learning for land use and land cover classification based on hyperspectral and multispectral Earth observation data: A review," *Remote Sens.*, vol. 12, no. 15, 2020, Art. no. 2495.
- [5] Y. Y. Ghadi, A. A. Rafique, T. Al Shloul, S. A. Alsubhany, A. Jalal, and J. Park, "Robust object categorization and scene classification over remote sensing images via features fusion and fully convolutional network," *Remote Sens.*, vol. 14, no. 7, 2022, Art. no. 1550.
- [6] W. Huang, Z. Yuan, A. Yang, C. Tang, and X. Luo, "TAE-net: Task-adaptive embedding network for few-shot remote sensing scene classification," *Remote Sens.*, vol. 14, no. 1, 2021, Art. no. 111.
- [7] F. Ullah et al., "Deep hyperspectral shots: Deep snap smooth wavelet convolutional neural network shots ensemble for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, no. 12, pp. 14–34, 2024.
- [8] Q. Zeng, J. Geng, K. Huang, W. Jiang, and J. Guo, "Prototype calibration with feature generation for few-shot remote sensing image scene classification," *Remote Sens.*, vol. 13, no. 14, 2021, Art. no. 2728.

- [9] F. Ullah, I. Ullah, R. U. Khan, S. Khan, K. Khan, and G. Pau, "Conventional to deep ensemble methods for hyperspectral image classification: A comprehensive survey," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 3878–3916, Jan. 2024.
- [10] S. Liu, H. Li, C. Jiang, and J. Feng, "Spectral–spatial graph convolutional network with dynamic-synchronized multiscale features for few-shot hyperspectral image classification," *Remote Sens.*, vol. 16, no. 5, 2024, Art. no. 895.
- [11] R. Alharbi, H. Alhichri, R. Ouni, Y. Bazi, and M. Alsabaan, "Improving remote sensing scene classification using quality-based data augmentation," *Int. J. Remote Sens.*, vol. 44, no. 6, pp. 1749–1765, 2023.
- [12] H. E. Kim, A. Cosa-Linan, N. Santhanam, M. Jannesari, M. E. Maros, and T. Ganslandt, "Transfer learning for medical image classification: A literature review," *BMC Med. Imag.*, vol. 22, no. 1, 2022, Art. no. 69.
- [13] Y. Yuan, L. Chen, H. Wu, and L. Li, "Advanced agricultural disease image recognition technologies: A review," *Inf. Process. Agriculture*, vol. 9, no. 1, pp. 48–59, 2022.
- [14] G. Sreenu and S. Durai, "Intelligent video surveillance: A review through deep learning techniques for crowd analysis," *J. Big Data*, vol. 6, no. 1, 2019, Art. no. 48.
- [15] M. Aljebreen, H. A. Mengash, M. Alamgeer, S. S. Alotaibi, A. S. Salama, and M. A. Hamza, "Land use and land cover classification using river formation dynamics algorithm with deep learning on remote sensing images," *IEEE Access*, vol. 12, pp. 11147–11156, 2024.
- [16] J. Li, U. A. Bhatti, S. A. Nawaz, M. Huang, R. M. Ahmad, and Y. Y. Ghadi, "Remote-sensing image classification: A comprehensive review and applications," in *Deep Learn. Multimedia Process. Appl.*. Boca Raton, FL, USA: CRC Press, 2024, pp. 18–47.
- [17] G. Dang et al., "Joint superpixel and Transformer for high resolution remote sensing image classification," *Sci. Rep.*, vol. 14, no. 1, 2024, Art. no. 5054.
- [18] G. Aziz, N. Minallah, A. Saeed, J. Frnda, and W. Khan, "Remote sensing based forest cover classification using machine learning," *Sci. Rep.*, vol. 14, no. 1, 2024, Art. no. 69.
- [19] K. O'shea and R. Nash, "An introduction to convolutional neural networks," 2015, *arXiv:1511.08458*.
- [20] A. F. Agarap, "Deep learning using rectified linear units (Relu)," 2018, *arXiv:1803.08375*.
- [21] V. Christlein, L. Spranger, M. Seuret, A. Nicolaou, P. Král, and A. Maier, "Deep generalized max pooling," in *Proc. Int. Conf. Document Anal. Recognit.*, 2019, pp. 1090–1096.
- [22] J. Bjorck, C. Gomes, B. Selman, and K. Q. Weinberger, "Understanding batch normalization," in *Proc. 32nd Conf. Neural Inf. Process. Syst.*, 2018, vol. 31, pp. 1–12.
- [23] W. Ma and J. Lu, "An equivalence of fully connected layer and convolutional layer," 2017, *arXiv:1712.01252*.
- [24] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks," 2016, *arXiv:1612.02295*.
- [25] I. Papoutsis, N. I. Bountos, A. Zavras, D. Michail, and C. Tryfonopoulos, "Benchmarking and scaling of deep learning models for land cover image classification," *ISPRS J. Photogramm. Remote Sens.*, vol. 195, pp. 250–268, 2023.
- [26] A. A. Adegun, S. Viriri, and J.-R. Tapamo, "Review of deep learning methods for remote sensing satellite images classification: Experimental survey and comparative analysis," *J. Big Data*, vol. 10, no. 1, 2023, Art. no. 93.
- [27] T. Carneiro, R. V. M. Da Nóbrega, T. Nepomuceno, G.-B. Bian, V. H. C. De Albuquerque, and P. P. Rebouças Filho, "Performance analysis of Google Colaboratory as a tool for accelerating deep learning applications," *IEEE Access*, vol. 6, pp. 61677–61685, 2018.
- [28] S.-. Ishikawa et al., "Example-based explainable AI and its application for remote sensing image classification," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 118, 2023, Art. no. 103215.
- [29] Z. Ma, M. Xia, H. Lin, M. Qian, and Y. Zhang, "FENet: Feature enhancement network for land cover classification," *Int. J. Remote Sens.*, vol. 44, no. 5, pp. 1702–1725, 2023.
- [30] K. Hu et al., "MCSGNet: A encoder–decoder architecture network for land cover classification," *Remote Sens.*, vol. 15, no. 11, 2023, Art. no. 2810.
- [31] G. Zhang, S. N. A. binti Roslan, C. Wang, and L. Quan, "Research on land cover classification of multi-source remote sensing data based on improved U-net network," *Sci. Rep.*, vol. 13, no. 1, 2023, Art. no. 16275.
- [32] D. Regan, G. Ravindran, P. Senthil, M. J. Rani, and S. Chakraborty, "Lightweight GWO-LSTM-based land cover classification using remote sensing images," in *Proc. Int. Conf. Evol. Algorithms Soft Comput. Techn.*, 2023, pp. 1–5.
- [33] A. Temenos, N. Temenos, M. Kaselimi, A. Doulamis, and N. Doulamis, "Interpretable deep learning framework for land use and land cover classification in remote sensing using SHAP," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, Mar. 2023, Art. no. 8500105.
- [34] V. N. Vinaykumar, J. A. Babu, and J. Frnda, "Optimal guidance whale optimization algorithm and hybrid deep learning networks for land use land cover classification," *EURASIP J. Adv. Signal Process.*, vol. 2023, no. 1, 2023, Art. no. 13.
- [35] H. Xie, Y. Chen, and P. Ghamisi, "Remote sensing image scene classification via label augmentation and intra-class constraint," *Remote Sens.*, vol. 13, no. 13, 2021, Art. no. 2566.
- [36] K. Xu, H. Huang, P. Deng, and Y. Li, "Deep feature aggregation framework driven by graph convolutional network for scene classification in remote sensing," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 10, pp. 5751–5765, Oct. 2022.
- [37] X. Jin, J. Chi, S. Peng, Y. Tian, C. Ye, and X. Li, "Deep image aesthetics classification using inception modules and fine-tuning connected layer," in *Proc. 8th Int. Conf. Wireless Commun. Signal Process.*, 2016, pp. 1–6.
- [38] K. Ngo, "FPGA hardware acceleration of inception style parameter reduced convolution neural networks," 2016.
- [39] P. H. Sánchez, "Can a CNN recognize Mediterranean diet?," *AIP Conf. Proc.*, vol. 1773, 2016, Art. no. 020002.
- [40] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2818–2826.
- [41] J.-B. Cordonnier, A. Loukas, and M. Jaggi, "On the relationship between self-attention and convolutional layers, 2019," *arXiv:1911.03584*.
- [42] K. Han et al., "A survey on vision transformer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 87–110, Jan. 2023.
- [43] M. Lin, M. Chen, Y. Zhang, C. Shen, R. Ji, and L. Cao, "Super vision transformer," *Int. J. Comput. Vis.*, vol. 131, pp. 3136–3151, 2023.
- [44] D. Bhatt et al., "CNN variants for computer vision: History, architecture, application, challenges and future scope," *Electronics*, vol. 10, no. 20, 2021, Art. no. 2470.
- [45] G. I. Okolo, S. Katsigiannis, and N. Ramzan, "IEViT: An enhanced vision transformer architecture for chest X-ray image classification," *Comput. Methods Programs Biomed.*, vol. 226, 2022, Art. no. 107141.
- [46] L. Lu, Y. Yi, F. Huang, K. Wang, and Q. Wang, "Integrating local CNN and global CNN for script identification in natural scene images," *IEEE Access*, vol. 7, pp. 52669–52679, 2019.
- [47] A. Bassel, A. B. Abdulkareem, Z. A. A. Alyasseri, N. S. Sani, and H. J. Mohammed, "Automatic malignant and benign skin cancer classification using a hybrid deep learning approach," *Diagnostics*, vol. 12, no. 10, 2022, Art. no. 2472.
- [48] B. Jena, S. Saxena, G. K. Nayak, L. Saba, N. Sharma, and J. S. Suri, "Artificial intelligence-based hybrid deep learning models for image classification: The first narrative review," *Comput. Biol. Med.*, vol. 137, 2021, Art. no. 104803.
- [49] A. Bhatt and V. T. Bhatt, "Dcrff-Lhrf: An improvised methodology for efficient land-cover classification on Eurosat dataset," *Multimedia Tools Appl.*, vol. 83, pp. 54001–54025, 2024.
- [50] S. A. Yamashkin, A. A. Yamashkin, V. V. Zanozin, M. M. Radovanovic, and A. N. Barmin, "Improving the efficiency of deep learning methods in remote sensing data analysis: Geosystem approach," *IEEE Access*, vol. 8, pp. 179516–179529, 2020.
- [51] M. Ç. Aksoy, B. Sirmacek, and C. Ünsalan, "Land classification in satellite images by injecting traditional features to CNN models," *Remote Sens. Lett.*, vol. 14, no. 2, pp. 157–167, 2023.

Saddaf Rubab received the M.Sc. degree in computer software and computer engineering from the NUST College of Electrical and Mechanical Engineering, Rawalpindi, Pakistan, in 2012, and the Doctoral degree in cyber security from the Universiti Teknologi PETRONAS, Seri Iskandar, Malaysia, in 2018.

She is an Assistant Professor with the Department of Computer Engineering, College of Computing and Informatics, University of Sharjah, Sharjah, UAE. She is also affiliated with the National University of Sciences and Technology, Islamabad, Pakistan. She worked on different academic positions from 2009 to 2013. She has worked on forecasts and implementing AI techniques in various interdisciplinary areas. Her research interests include distributed computing, security, and prediction systems.

Muhammad Attique Khan (Member, IEEE) received the master's and Ph.D. degrees in human activity recognition for application of video surveillance and skin lesion classification using deep learning from COMSATS University Islamabad, Islamabad, Pakistan, in 2018 and 2022, respectively.

He is currently an Assistant Professor with Computer Science Department, HITEC University, Taxila, Pakistan. His primary research focus in recent years is medical imaging, COVID-19, MRI analysis, video surveillance, human gait recognition, and agriculture plants using deep learning. He has above 280 publications that have more than 10 000+ citations and an impact factor of 850+ with h-index 61 and i-index 165. He is the Reviewer of several reputed journals, such as the IEEE TRANSACTION ON INDUSTRIAL INFORMATICS, IEEE TRANSACTION OF NEURAL NETWORKS, *Pattern Recognition Letters*, *Multimedia Tools and Application*, *Computers and Electronics in Agriculture*, *IET Image Processing*, *Biomedical Signal Processing Control*, *IET Computer Vision*, *EURASIP Journal of Image and Video Processing*, *IEEE Access*, *MDPI Sensors*, *MDPI Electronics*, *MDPI Applied Sciences*, *MDPI Diagnostics*, and *MDPI Cancers*.

Ameer Hamza is currently working toward the Ph.D. degree in computer science with HITEC University, Taxila, Pakistan.

His major interests include object detection and recognition, video surveillance, medical, and agriculture using deep learning and machine learning. He has published four impact factor papers to date.

Hussain Mobarak Albarakati received the Master's and Ph.D. degrees from the University of Connecticut, USA, in 2016 and 2020, respectively.

He is with the Department of Computer and Network Engineering, College of Computing, Umm Al-Qura University, Makkah, Saudi Arabia. He is currently an Assistant Professor with the university where teaching courses related to AI, networks, and embedded systems. In addition, he is a Senior AI Researcher related to remote sensing and medical. His research interests include artificial intelligence, machine learning, embedded systems, the Internet of Things, cloud computing, software engineering, and networks. He published several research journals in those areas.

Oumaima Saidani received the M.Sc. degree from Paris Dauphine University, Paris, France, in 2010, and the Ph.D. degree from Paris 1-Panthéon Sorbonne University, Paris, France, in 2018, both in computer sciences.

She is currently an Assistant Professor with Information Systems Department, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. Her research interests include information systems engineering, business process engineering, process mining, context-aware computing, deep learning, and artificial intelligence.

Amal Alshardan is with the Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. Her research interests include information systems engineering, business process engineering, process mining, context-aware computing, deep learning, and artificial intelligence.

Areej Alasiry received the B.Sc. degree in information systems from King Khalid University, Abha, Saudi Arabia, and the M.Sc.(Hons.) degree in advanced information systems and the Ph.D. degree in computer science and information systems from Birkbeck College, University of London, London, U.K., in 2010 and 2015, respectively.

She is currently an Assistant Professor with the College of Computer Science, King Khalid University. She also holds the position of the College Vice Dean for Graduate Studies and Scientific Research. Her research interests include machine learning and data science.

Mehrez Marzougui was born in Kasserine, Tunisia, in 1972. He received the B.Sc. degree from the University of Tunis, Tunis, Tunisia, in 1996, and the M.Sc. and Ph.D. degrees from the University of Monastir, Monastir, Tunisia, in 1998 and 2005, respectively, all in electronics.

From 2001 to 2005, he was a Research Assistant with Electronics and Micro-Electronics Laboratory. From 2006 to 2012, he was an Assistant Professor with Electronics Department, University of Monastir. Since 2013, he has been an Assistant Professor with Engineering Department, College of Computer Science, King Khalid University, Abha, Saudi Arabia. He is the author of more than 30 articles. His research interests include hardware/software cosimulation, image processing, and multiprocessor systems on chips.

Yunyoung Nam received the B.S., M.S., and Ph.D. degrees in computer engineering from Ajou University, Suwon, South Korea, in 2001, 2003, and 2007, respectively.

From 2007 to 2010, he was a Senior Researcher with the Center of Excellence in Ubiquitous System, Stony Brook University, Stony Brook, NY, USA, where he was a Postdoctoral Researcher from 2009 to 2013. He was a Research Professor with Ajou University from 2010 to 2011. He was a Postdoctoral Fellow with Worcester Polytechnic Institute, Worcester, MA, USA, from 2013 to 2014. He was the Director of ICT Convergence Rehabilitation Engineering Research Center, Soonchunhyang University, Asan, South Korea, from 2017 to 2020. Since 2020, he has been the Director of ICT Convergence Research Center, Soonchunhyang University, where he is currently an Assistant Professor with the Department of Computer Science and Engineering. His research interests include multimedia database, ubiquitous computing, image processing, pattern recognition, context awareness, conflict resolution, wearable computing, intelligent video surveillance, cloud computing, biomedical signal processing, rehabilitation, and healthcare systems.