Deep Learning-Based Satellite Image Classification Using CNN and MobileNetV2 Transfer Learning

Divya.B [1] , Subashini T.S [2], Bharathidasan.B[3]

[1]Research Scholar, Department of Computer Science and Engineering, Annamalai University,

[2]Professor of Computer Science and Engineering, Annamalai University,

[3]Associate professor, Department of ECE, Sree rama engineering college, Tirupati.

ABSTRACT

Classification of satellite images is a crucial task for remote sensing applications including land cover mapping, urban planning, and environmental monitoring. This work introduces a robust deep learning architecture for classifying high-resolution satellite images into four classes: cultivated area, forest area, desert region, and water bodies. The method uses both a native Convolutional Neural Network (CNN) and transfer fine-tuned MobileNetV2 model learning. Models were tested with accuracy, precision, recall and F1-score. The results demonstrate that MobileNetV2 achieved a test accuracy of 99.14%, which outperformed the custom CNN, which had attained 93.12%, thus confirming the efficacy of transfer learning in satellite image classification.

KEYWORD: Satellite Image Classification, Convolutional Neural Network (CNN), MobileNetV2, Transfer Learning, Remote Sensing, Deep Learning, Land Cover Classification, Image Recognition, Accuracy Assessment, Performance Measures.

1. Introduction

Remote sensing has witnessed significant growth in the past decade, driven by significant technological advancements in satellite imagery and image processing algorithms. Conventional ways to classify Satellite images typically use traditional hand-crafted feature extraction techniques and heuristic laws, which are slow and insensitive to subtle spatial patterns. Contrarily, deep learning-based the techniques, particularly the convolutional neural network (CNN)-based techniques, have seen visibility due to their ability to automatically record complicated spatial relationships within large-scale satellite data sets. The present study examines the efficiency of deep learning approaches by comparing a typical CNN architecture with the use of a transfer learning strategy with MobileNetV2 with a focus on cultivated land, green area, desert, and water to determine how they fare in actual world conditions.

Convolutional Neural Networks (CNN) has demonstrated remarkable performance in picture classification tasks, mostly due to its capacity for hierarchical feature learning [26][28]. CNNs can identify local patterns like edges, textures, and forms in early levels by processing raw pixel input through a series of convolutional layers. Deeper layers then synthesize these features into more abstract representations for precise class separation. In satellite images, where spatial context such as urban layouts, vegetation distribution, and water bodies is crucial to categorization accuracy, this hierarchical structure is very helpful. CNNs automatically train discriminative features, which makes them ideal for high-dimensional remote sensing data, in contrast to conventional techniques that call for manually created features.

This work uses MobileNetV2, a lightweight deep learning model built for resource-constrained contexts, to further improve speed and efficiency. MobileNetV2 significantly reduces processing overhead while maintaining competitive accuracy with the use of depthwise separable convolutions and inverted residual blocks. The model uses pretrained ImageNet weights within a transfer learning framework to effectively adapt to satellite imagery. This approach not only minimizes overfitting, a common issue with short datasets, but it also accelerates convergence, making it a feasible substitute for classifying urban satellite images in situations where data availability may be limited.

This work compares a custom CNN with MobileNetV2-based transfer learning to investigate the trade-offs between model complexity, computational efficiency, and classification accuracy in the context of urban satellite data. Based on their unique constraints be they computational resources, dataset size, or required precision the findings are intended to help academics and practitioners choose the best deep learning approaches. In the end, our work supports the larger endeavor to optimize automated satellite image processing for applications related to environmental monitoring, urban planning, and disaster management.

2. Related Work

Deep learning has revolutionized satellite image classification by making it possible to automatically extract high-level spatial and semantic characteristics. This contrasts with conventional machine learning methods that depend on feature engineering by hand. Initial work by Tumpa and Islam [1] showed how lightweight CNN architectures in conjunction with SVM classifiers could increase classification accuracy for satellite data. By combining predictions from several machine learning models, ensemble approaches like the confidence-based model fusion put out by K et al. [2] have improved resilience. These developments highlight the move toward data-driven methods that can manage the complexity and unpredictability present in data from remote sensing.

A major advancement in effective deep learning for remote sensing was made possible by MobileNetV2 [3], a low-power architecture that reduces computation without sacrificing efficiency by using depthwise separable convolutions and inverted residual blocks. The approach has proven very useful in situations requiring few resources, such as the processing of satellite images. Transfer learning, which adapts models pretrained on large datasets like ImageNet for remote sensing applications, has further improved efficiency. Studies by Hu et al. [4] and Marmanis et al. [5] demonstrated that fine-tuning pretrained CNNs significantly enhances high-resolution remote sensing scene classification, even with minimal labeled data.

Deep learning approaches have been very beneficial for urban land use classification, with researchers investigating different architectures and fusion methodologies. Transfer learning and deep CNNs were used by Xu et al. [8] to extract buildings from extremely high-resolution images. However, Audebert et al. [9] used multimodal deep networks to combine RGB and multispectral inputs, improving classification accuracy. The AID dataset [14] and other benchmark datasets have been essential in standardizing assessments for aerial scene classification. Deep learning has improved object-level classification in addition to scene-level analysis. In order to demonstrate how deep learning may be applied to remote sensing applications, Li et al. [10] created models for the detection and counting of oil palm trees, and Zhang et al. [16] presented a CNN-based technique for autonomous road extraction.

In order to further improve classification performance, recent research has emphasized the need of merging multimodal data with sophisticated learning approaches. In order to take advantage of both spatial and temporal information, Roy et al. [22] and Qi and Zhang [15] combined CNNs with LSTMs and transfer learning. In the meantime, Cheng et al. [12],[27] improved scene classification accuracy by using metric learning to improve discriminative feature representations. Traditional deep learning models like ResNet [18], VGGNet [19], and AlexNet [20] remain fundamental, influencing more effective designs like MobileNetV2 and specific remote sensing adaptations. Comprehensive assessments by Zhu et al. [7], Liu et al. [23], and Zhang et al. [24] show the ground-breaking impacts of deep learning in Earth observation, encompassing tasks from land cover categorization to object recognition. Additionally, Zhao and Du [25] established deep spectral–spatial feature extraction approaches for hyperspectral data, enabling finer-grained classification.

This work conducts a direct comparison between a custom CNN and a MobileNetV2-based transfer learning approach on a multi-class satellite dataset. This work not only validates the advantages of transfer learning in scenarios with limited annotated data but also provides practical insights into model selection for urban satellite image classification.

3. Methodology

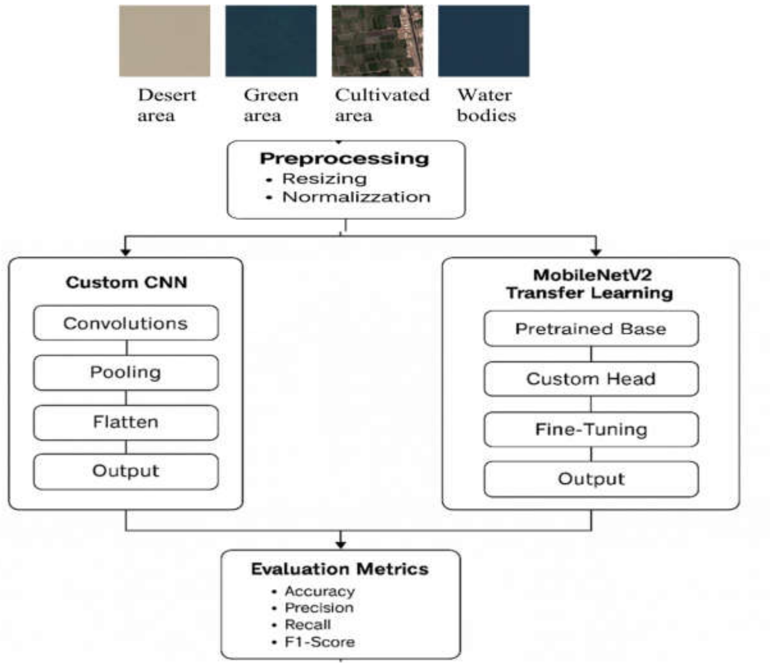

Fig 1 Workflow of the Proposed Satellite Image Classification Approach

3.1 Dataset and Preprocessing

Workflow diagram illustrating the preprocessing steps, model architectures (Custom CNN and MobileNetV2 Transfer Learning), and evaluation metrics used in this work is depicted in Fig 1.

There are 4785 images in the dataset used in this work (3,801 for training, 956 for validation, and 28 for testing), satellite images categorized into four distinct classes: cultivated area, green area, desert region and water bodies. These categories were selected to represent diverse urban and residential structures, thereby enabling the development of a robust classification model.

To ensure consistency and to enhance training quality, all images were resized to a fixed resolution of 224 x 224 x 3. The values of the pixels were adjusted to fall between 0 and 1 by using:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \qquad (eq\ 3.1.1)$$

Where x_{max}, x_{min} stand for the maximum and minimum pixel values respectively, where x is the original pixel intensity and x' is the normalized value. Normalization ensures faster convergence during training and prevents issues caused by varying intensity scales.

3.2 Custom CNN Architecture

A Custom Convolutional Neural Network (CNN) created especially to categorize satellite photos into four: cultivated_area, green_area, desert_region, and water_bodies is one of the methods used in the proposed work. In order to extract textual and geographical information from the input images, CNN starts with convolutional layers. The mathematical expression for the convolution operation is as follows.

$$f(i,j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I(i+m.j+n).K(m,n) \qquad \text{(eq 3.2.1)}$$

Where *I* denote the input image, *K* represents the convolution kernel and $f(i,j)$ is the resulting feature map. To reduce dimensionality while retaining important information, max pooling is applied, defined as

$$p(i,j) = \max_{(m,n)\in R} f(i+m,j+n) \qquad \text{(eq 3.2.2)}$$

Where the pooling region is denoted by *R*. After being extracted, the features are flattened and run through layers that are fully connected. The dense layer's outputs are calculated as

$$z_j = \sigma\left(\sum_{i=1}^{n} w_{i,j}\, x_i + b_j\right) \qquad (eq\ 3.2.3)$$

Where σ is the activation function, x_i are the input features, $w_{i,j}$ and b_j stand for the learnable weights and biases and. Finally, the class probabilities for the four categories are produced via a softmax layer:

$$P(y=j/x) = \frac{e^{zj}}{\sum_{k=1}^{C} e^{zk}}, C=4. \qquad \text{(eq 3.2.4)}$$

Algorithm 1: Custom CNN for Satellite Image Classification

Input: Satellite image dataset D with 4 classes (cultivated_area, green_area, desert_region, water_bodies)

Output: Trained CNN model and predicted labels

1: Load image dataset paths and labels

2: Resize all images to 224 × 224

3: Normalize pixel values to range [0, 1]

4: One-hot encode class labels

5: Split dataset into Train, Validation, and Test sets

6: Define CNN architecture:

7: Conv2D → ReLU → MaxPooling (×3)

```
8:      Flatten → Dense → Dropout → Dense (Softmax with 4 outputs)
9: Compile model with:
10:     Loss = categorical cross-entropy
11:     Optimizer = Adam(learning_rate = 0.001)
12:     Metric = accuracy
13: For each epoch in num_epochs do
14:     For each batch in Train set do
15:         Forward pass → compute predictions
16:         Compute loss
17:         Backpropagate and update weights
18:     end for
19:     Evaluate on Validation set
20:     If validation loss does not improve for patience rounds then
21:         Stop training early
22:     end if
23: end for
24: Evaluate final model on Test set → compute Accuracy, Precision, Recall, F1-score
25: Save trained CNN model
```

3.3 MobileNetV2 Transfer Learning

The second approach utilizes MobileNetV2 through transfer learning to improve classification performance. The foundation for feature extraction is MobileNetV2, which was first trained on the extensive ImageNet dataset. Its pretrained convolutional base efficiently captures high-level spatial representations.

The model is modified for the four satellite image categories by adding a specific classification head. To further enhance performance, selective fine-tuning of deeper convolution layers is carried out, enabling the model to better capture domain specific features. Gradient descent is used to update the model parameters during training.

$$\theta^{(t+1)} = \theta^{(t)} - \eta.\nabla_{\theta}L(\theta) \qquad (eq\ 3.3.1)$$

Where $L(\vartheta)$ is the loss function, η is the learning rate and ϑ stand for the model parameters. The probability distribution for the four classes is produced by the final classification layer using the softmax activation function.

Algorithm 2: MobileNetV2 with Transfer Learning

Input: Satellite image dataset D with 4 classes (cultivated_area, green_area, desert_region, water_bodies)

Output: Fine-tuned MobileNetV2 model and predicted labels

1: Load image dataset paths and labels

2: Resize all images to 224 × 224

3: Normalize pixel values to range [-1, 1]

4: One-hot encode class labels

5: Split dataset into Train, Validation, and Test sets

6: Load pretrained MobileNetV2 (exclude top classification layers)

7: Freeze all base layers

8: Add custom classification head:

9: GlobalAveragePooling2D → Dense → Dropout → Dense (Softmax with 4 outputs)

10: Compile model with:

11: Loss = categorical cross-entropy

12: Optimizer = Adam(learning_rate = 0.0001)

13: Metric = accuracy

14: Train model on Train set for initial epochs with frozen layers

15: Unfreeze last N layers of MobileNetV2

16: Fine-tune model with lower learning rate

17: For each epoch in num_epochs do

18: For each batch in Train set do

19: Forward pass → compute predictions

```
            Compute loss
            Backpropagate and update weights
        end for
        Evaluate on Validation set
        If early stopping condition met then
            Break loop
        end if
end for
Evaluate final model on Test set → compute Accuracy, Precision, Recall, F1-score
Save trained MobileNetV2 model
```

3.4 Model Training Configuration

Table 1. Summary of Model Training Configuration

Feature	**Custom CNN**	**MobileNetV2 Transfer Learning**
Epochs	5 (early stopping used, best model restored)	5 (early stopping used, best model restored)
Total Parameters	11,169,476	2,422,468
Trainable Parameters	11,169,476	164,484
Non-trainable Parameters	0	2,257,984
Optimizer	Adam (0.001)	Adam (0.001)
Loss Function	Categorical Crossentropy	Categorical Crossentropy
Noise / Regularization	Dropout (0.5) to reduce overfitting	Dropout (0.5) on dense layer
Augmentation	Rotation, Zoom, Width/Height shift, Shear, Flip	Same as CNN
Input Image Size	224 × 224 × 3	224 × 224 × 3
Batch Size	32	32

4. Results and Evaluation

This work uses standard classification performance criteria, such as accuracy, loss, precision, recall, and F1-score, to assess the performance of both the custom CNN and MobileNetV2 transfer learning models. These measurements were calculated using a test dataset that was not used for either training or validation.

Confusion Matrix for Custom CNN Model

The confusion matrix was created in order to evaluate the Custom CNN model's performance, as seen in Fig 2. The model's classification accuracy for each class is shown in detail in the matrix.

As seen in the confusion matrix, the model correctly classified the majority of test samples across all four classes: cultivated_area, desert_region, green_area, and water_bodies. The desert_region class achieved perfect classification (1131/1131 correct), while cultivated_area recorded 634 correct predictions out of 644. The green_area class achieved 1400 correct predictions with 100 misclassified as water_bodies, and the water_bodies class had 1291 correct predictions with 217 misclassified as green_area.

Out of the 4,785 test images, the model correctly classified 4,455 and misclassified 330, resulting in a test accuracy of 93.12%. These results indicate strong overall performance for the custom CNN, though classes with similar visual patterns, particularly green_area and water_bodies, remain more challenging to separate accurately.

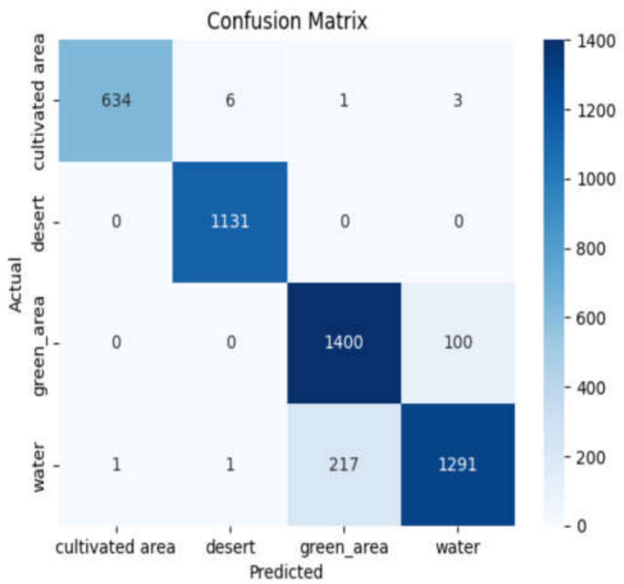

Fig 2: Custom CNN Model Confusion Matrix

Custom CNN Model Results

The self-designed Convolutional Neural Network (CNN) developed for this research achieved a test accuracy of 93.12% with a categorical cross-entropy loss of 0.2071 for the satellite image classification task. The categorical cross-entropy (CCE) loss used to evaluate multi-class classification is given by:

$$L_{CCE} = \sum_{i=1}^{c} y_i log(\hat{y_i}) \qquad (eq\ 4.1)$$

Where y_i is the true class label (one-hot encoded), $\hat{y_i}$ is the predicted probability for class *i* and *C* is the number of classes (in this case, C= 4). The difference between the actual and anticipated class distributions is measured by this loss.

These results demonstrate that the model CNN effectively learned spatial and texture features, showing strong generalization capability across the four classes: cultivated_area, green_area, desert_region, and water_bodies.

Class-wise metrics is presented in Table 2 indicating strong performance for green_area and water_bodies. The desert_region recorded slightly lower precision, which may be attributed to its visual similarity with cultivated_area.

Table 2. Custom CNN model performance

Class	Precision	Recall	F1-Score
cultivated_area	0.89	0.96	0.92
green_area	0.96	1.00	0.95
desert_region	0.87	0.93	0.90
water_bodies	0.93	0.86	0.89

With an average accuracy of 93.12% overall, the CNN demonstrated dependable classification in the majority of categories.

Confusion Matrix for MobileNetV2 Model

Fig 3 shows the confusion matrix for the MobileNetV2 model. A more comprehensive overview of the model's classification results for the four categories of cultivated_area, green_area, desert_region, and water_bodies is provided by this image.

MobileNetV2 exhibits outstanding classification performance in each of the four classes, according to the confusion matrix. While the green_area class obtains 99.2% accuracy with only 12 photos incorrectly labeled as water, the cultivated_area (644/644) and desert_region (1131/1131) classes are categorized with perfect accuracy. Likewise, 98.1% accuracy is recorded by the water_bodies class, with 29 cases incorrectly classified as green_area.

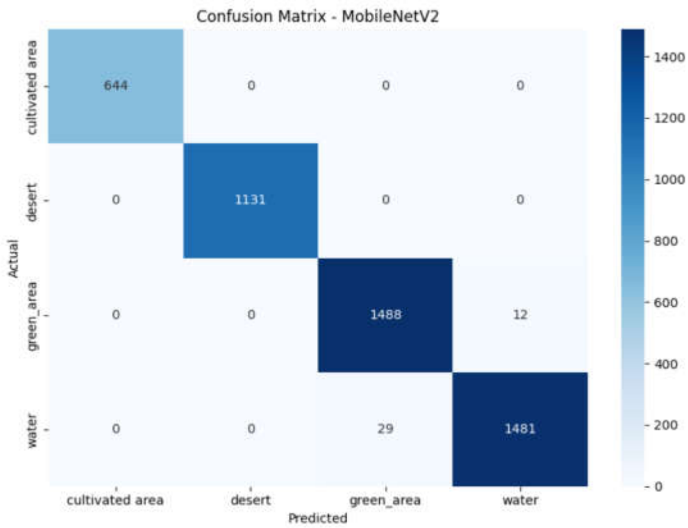

Fig 3: MobileNetV2 Model Confusion Matrix

With 4,744 out of 4,785 test images properly classified and only 41 mistakenly classified, the model's overall accuracy was 99.14%. The usefulness of transfer learning with MobileNetV2 for satellite image categorization is demonstrated by this performance, which significantly surpasses that of the custom CNN with consistent accuracy and high generalization even across visually related classes.

MobileNetV2 model Results

The MobileNetV2 model, fine-tuned through learning on the same dataset, significantly outperformed the custom CNN. It achieved a test accuracy of 99.14% with a categorical cross-entropy loss of 0.0305.

By leveraging pretrained ImageNet weights**,** MobileNetV2's convolutional base provided robust feature representations, while selective fine-tuning of deeper layers allowed effective adaptation to the satellite image domain.

The loss function was identical to that of the CNN which was computed using categorical cross-entropy, while parameter updates followed the gradient descent rule:

$$\theta^{(t+1)} = \theta^{(t)} - \eta . \nabla_{\theta} L(\theta) \qquad \text{(eq4.2)}$$

Where $L(\vartheta)$ is the loss function, η is the learning rate and ϑ stands for the model parameters

The class-wise performance, presented in Table 3, demonstrates near-perfect classification, with water_bodies achieving precision and recall values of 1.00, while other classes maintained performance between 0.98–0.99.

Table 3. MobileNetV2 model performance

Class	Precision	Recall	F1-Score
cultivated_area	0.98	0.99	0.98
green_area	1.00	0.99	0.99
desert_region	0.98	0.98	0.98
water_bodies	0.99	0.96	0.99

The average F1-Score of 0.96 for MobileNetV2 showed that it was better than the custom CNN. Table 4 displays the comparison of total performance.

Table 4. CNN and MobileNetV2's overall performance comparison

Model	**Accuracy (%)**	**Loss**	**Precision (Avg)**	**Recall (Avg)**	**F1-Score (Avg)**
Custom CNN	93.12	0.2071	0.92	0.93	0.92
MobileNetV2	99.14	0.0305	0.96	0.97	0.96

The results confirm that MobileNetV2 provided a 4.25% accuracy improvement over the custom CNN and reduced the categorical cross-entropy loss by a large margin. The higher precision, recall, and F1-scores validate that transfer learning with MobileNetV2 yielded more robust and accurate classification outcomes than the baseline CNN.

4.3 Comparison with Existing Works

A comparative analysis with recent studies is provided in Table 5. The proposed MobileNetV2 model surpassed most of the existing works in terms of classification accuracy, highlighting its efficiency in satellite image classification tasks.

Table 5. Comparison of existing work with the proposed work

Study / Model	**Dataset / Classes**	**Methodology**	**Accuracy (%)**
Tumpa & Islam (2024)	Satellite Imagery (Binary/Multiclass)	Lightweight Parallel CNN + SVM	94.20
K et al. (2023)	Remote Sensing Dataset	Ensemble ML Classifiers (confidence-based)	95.00

Study / Model	**Dataset / Classes**	**Methodology**	**Accuracy (%)**
Hu et al. (2015)	High-Resolution Remote Sensing Imagery	Transfer Learning with Pretrained CNNs	96.00
Audebert et al. (2017)	VHR Urban Images (RGB + Multispectral)	Multimodal Deep CNNs	97.10
Zhang et al. (2019)	Aerial Images	Patch-based CNN for Road Extraction	95.50
Proposed Custom CNN (This Work)	4785 Images (4 Classes)	CNN with 3 Conv-Pool Blocks, Dense, Dropout	**93.12**
Proposed MobileNetV2 Transfer Learning	Same Dataset (4 Classes)	Pretrained MobileNetV2 + Fine-tuned Dense layers	**99.14**

The proposed MobileNetV2 model's accuracy of 99.14% is higher than those reported in related studies, indicating its capability to generalize well across diverse urban satellite scenes.

5. Conclusion and Future Work

This study demonstrates the effectiveness of pretrained convolutional neural network architectures particularly MobileNetV2, for satellite image categorization tasks. Even with a tiny dataset and few training epochs, the model was able to achieve great generalization and superior accuracy by utilizing transfer learning and fine-tuning techniques. The findings demonstrate how lightweight deep learning models may preserve computational efficiency while extracting significant spatial and semantic characteristics from high-resolution satellite data.

Other cutting-edge designs like EfficientNet, ResNet variations (such ResNet-50 and ResNet-152), and attention-based models like Vision Transformers (ViTs) will be included into future work to expand this strategy. Particularly for bigger and more varied datasets, these models may be able to significantly increase classification accuracy and scalability. To improve performance in situations with little data, integrating strategies like multi-task learning frameworks, semi-supervised learning, and robust data augmentation pipelines will also be investigated. These developments may open the door to the creation of scalable and extremely precise satellite image classification systems that have practical uses in environmental monitoring, urban planning, and disaster relief.

6. References

Tumpa, P.P., & Islam, M.S. Lightweight Parallel CNN with SVM for Satellite Imagery Classification. IEEE Transactions on Artificial Intelligence, (2024). https://doi.org/10.1109/TAI.2024.3423813

K, A., B, R., & Neha, P.S. Remote Sensing Image Classification Based on Confidence Score of Ensemble ML Classifiers. Proc. EASCT 2023.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.C. MobileNetV2: Inverted Residuals and Linear Bottlenecks, CVPR, 4510–4520. (2018).

Hu, F., Xia, G.S., Hu, J., & Zhang, L. Transferring Deep Convolutional Neural Networks for the Scene Classification of High-Resolution Remote Sensing Imagery, Remote Sensing, 7(11), 14680–14707, (2015).

Marmanis, D., et al. Deep Learning Earth Observation Classification Using ImageNet Pretrained Networks. IEEE GRSL, 13(1), 105–109. (2016).

Chen, Y., Jiang, H., Li, C., Jia, X., & Ghamisi, P. Deep Feature Extraction and Classification of Hyperspectral Images Based on CNNs. IEEE TGRS, 54(10), 6232–6251. (2016).

Zhu, X.X., et al. Deep Learning in Remote Sensing: A Review. IEEE GRSM, 5(4), 8–36. (2017).

Xu, Y., et al. Building Extraction in Very High Resolution Remote Sensing Imagery Using Deep Learning and Transfer Learning, Remote Sensing, 10(9), 1443. (2018).

Audebert, N., Saux, B.L., & Lefèvre, S. Beyond RGB: Very High Resolution Urban Remote Sensing With Multimodal Deep Networks. ISPRS Journal of Photogrammetry and Remote Sensing, 140, 20–32. (2017).

Li, W., Fu, H., Yu, L., & Cracknell, A Deep Learning Based Oil Palm Tree Detection and Counting for High-Resolution Remote Sensing Images, Remote Sensing, 9(1), 22. . (2017).

Chen, T., Xu, Y., & Ma, Y. Remote Sensing Image Scene Classification Based on a Two-Stage Deep Feature Fusion Framework, Remote Sensing, 12(7), 1160. (2020).

Cheng, G., et al. When Deep Learning Meets Metric Learning: Remote Sensing Image Scene Classification via Learning Discriminative CNNs, IEEE TGRS, 56(5), 2811–2821. (2018).

Xie, Z., Lu, B., & Sha, Z. Learning High-Level Feature for Satellite Image Scene Classification with Limited Labeled Samples, Remote Sensing, 8(8), 682. (2016).

Xia, G.S., et al. AID: A Benchmark Data Set for Performance Evaluation of Aerial Scene Classification, IEEE TGRS, 55(7), 3965–3981. (2017).

Qi, Q., & Zhang, Y. Land Use Classification Based on Transfer Learning Using High-Resolution Satellite Images, Sustainability, 12(13), 5223. (2020).

Zhang, C., et al. Automatic Urban Road Extraction from Remote Sensing Images Using a Patch-Based CNN, Remote Sensing, 11(10), 1165. (2019).

Geng, X., et al. Deep Learning Approaches for Hyperspectral Image Classification: A Review, Remote Sensing, 13(3), 358. (2021).

He, K., Zhang, X., Ren, S., & Sun, J. Deep Residual Learning for Image Recognition, CVPR, 770–778. (2016).

Simonyan, K., & Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv preprint arXiv:1409.1556. (2014).

Krizhevsky, A., Sutskever, I., & Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. NIPS, 1097–1105. (2012).

Li, H., & Zhang, L. Transfer Learning-Based Road Extraction from Satellite Images, Sensors, 21(5), 1610. (2021).

Roy, S., et al. Multimodal Fusion for Satellite Image Classification Using CNN and LSTM, IEEE JSTARS, 13, 5567–5578. (2020).

Liu, L., et al. Deep Learning for Generic Object Detection: A Survey. IJCV, 128, 261–318. (2018).

Zhang, L., Zhang, L., & Du, B. Deep Learning for Remote Sensing Data: A Technical Tutorial on the State of the Art. IEEE GRSM, 4(2), 22–40. (2016).

Zhao, W., & Du, S. Spectral–Spatial Feature Extraction for Hyperspectral Image Classification: A Dimension Reduction and Deep Learning Approach. *IEEE TGRS*, 54(8), 4544–4554. (2016).

Showkat A. Dar and S. Palanivel, Performance Evaluation of Convolutional Neural Networks (CNNs) And VGG on Real Time Face Recognition System, Adv. Sci. Technol. Eng. Syst. J., vol. 6, no. 2, pp. 956–964, 2021, doi: 10.25046/aj0602109.

Showkat A. Dar, S. Palanivel, M. K. Geetha, and M. Balasubramanian, Mouth Image Based Person Authentication Using DWLSTM and GRU, Inf. Sci. Lett., vol. 11, no. 3, pp. 853–862, 2022, doi: 10.18576/isl/110317.

Walid Ayadi , Showkat A. Dar, AI-Powered CNN Model for Automated Lung Cancer Diagnosis in Medical Imaging, Int. J. Stat. Med.Res., vol. 14, pp. 616–625, 2025, doi: 10.6000/1929-6029.2025.14.39.