Journal of Innovative Image Processing (ISSN: 2582-4252)
www.irojournals.com/iroiip/

Integrated Feature Learning and Decision Modeling for Land Use and Land Cover Analysis

Jayesh Dhanesha[1], Sweta Panchal[2]

[1]Research Scholar, Department of Computer Engineering, Dr. Subhash University, Gujarat, India.
[2]Associate Professor, Department of Electronics & Communication Engineering, Dr. Subhash University, Gujarat, India.

E-mail: [1]jayeshdhanesha@gmail.com, [2]sweta.panchal@dsuni.ac.in

Abstract

Training the neural network using high-quality labeled data is a major challenge, as there are many grey areas in image classification. Hence, experts are available to label the data. Proposed in this research paper is a Heterogeneous Multi-Stream Deep Learning Framework. Its application of six advanced CNNs will remove such complications using their complementary inductive biases. An evaluation of two fusion paradigms was successfully achieved: a decision-level weighted average ensemble and a multi-stream CNN-SVM at both feature levels. The feature-level fusion method was found to be more discriminative than probabilistic averaging in testing across three different datasets: NWPU-RESISC45, UC Merced, and AID. This approach achieves the best results on all datasets, with our method achieving a highest F1-score of 97.24% on the NWPU-RESISC45 benchmark. The performance of the six-stream design, having 8192 features, was slightly affected, dropping to 97.15% because of the curse of dimensionality. The findings support the five-stream CNN-SVM as the best architecture since it easily strikes a balance between feature richness and the complexity of the classifier.

Keywords: Aerial Scene Classification, Deep Learning, Feature Fusion, CNN-SVM, Ensemble Learning, NWPU-RESISC45, UC Merced, AID.

1. Introduction

The rapid development of high-resolution remote sensing images has almost revolutionized the capability of scanning land surfaces, leading to rapid advancements in applications related to land use/cover classification, urban planning, espionage, and disaster management. These applications are all grounded in LULC classification. However, it is not feasible to semi-automatically distinguish complex aerial images into semantic classes because the patterns in high-resolution images are more complex. The available datasets, such as NWPU-RESISC45, are more challenging due to higher intra-class variation, such as images of different arrangements of airports and aircraft on the ground, including images of different aircraft on different runways, as well as higher inter-class similarity between densely populated residential areas and commercial areas.

DOI: https://doi.org/10.36548/jiip.2026.1.004
Received: 26.11.2025, received in revised form: 24.12.2025, accepted: 11.01.2026, published: 19.01.2026

Previously, LULC classification utilized feature descriptions as classifiers. These descriptions were human-made features like SIFT and HOG, as well as statistical classifiers. However, if these technologies were efficient, it would not have been possible to extract the deep semantic hierarchies. There has been a revolution in deep learning as a result of convolutional neural networks. The VGG and ResNet models leverage this hierarchy of spatial data.

However, the potentials of single stream CNN models are limited. A specific model with a specific inductive bias, such as the texture bias of VGG or the shape bias of ResNet, tends not to generalize well across different types of scene categories. In the local dialect, modeling involves selecting projects and their economic benefits as well as researching and evaluating alternative policies from an economic perspective. However, the selection of the best approach for integration remains an important research gap. Although decision-level ensembles (voting and averaging) are commonly used, these typically assume equal weighting on each model and ignore differences in model reliability. Feature-level fusion provides richer information about the semantics of the feature, yet it also runs the risk of the curse of dimensionality. To address the problems above, this paper proposes a multi-stream deep learning framework to evaluate and optimize two different fusion paradigms: weighted average ensemble (decision-level) and multi-stream CNN-SVM (feature-level). We employ six unique and state-of-the-art backbones ResNet50, VGG16, InceptionV3, MobileNetV2, DenseNet121, and EfficientNetB0 to achieve maximum architectural diversity. Following are the main contributions of this paper:

1. **Construction of Heterogeneous Multi-Stream Systems:** A Highly effective integration system that utilizes the complementary strengths of six different CNN architectures in addressing ambiguities of very similar classes (for example, Palace vs. Church) when handled by individual models.

2. **Fusion Paradigm Comparison:** A strict comparison of feature-level fusion using CNN-SVM and decision-level fusion using a weighted ensemble of classifiers. In contrast with previous works that could not reach a conclusion from an experiment of comparable type, here we provide an experimental verification of trade-offs between consensus of probabilities and high-dimensional feature fusion.

3. **Dimensionality Scale Detection:** We identify the important dimensionality of scales in a challenging ablation study. We show that feature level fusion is optimal when using a five-stream system (97.24%) and outperforms decision- level fusion experiments. Yet, it has been found that the turning point has been reached when the loss of detailed information exceeds the gain due to the addition of the sixth stream (over 8,000 dimensions).

4. **Modern Outcomes:** The proposed framework has been evaluated using NWPU-RESISC45, UC Merced, and AID benchmark databases. It outperformed each of them as it set a new benchmark in multi-benchmark aerial scene classification.

2. Literature Review

The academic LULC classification now points towards the development of hybrid architectures that will combine the advantages of deep learning methods and classical statistical theory. The paper provides a systematic review of the shift to an end-to-end classification

approach toward a more decoupled paradigm, which creates the technical groundwork for using CNNs as feature extractors in conjunction with support vector machine (SVM) meta-classifiers.

2.1 Change in LULC Paradigms of Classification

In the past, LULC was classified using a decoupled, two-stage approach. Traditional methods used descriptors significantly based on hand-crafted features, including color histograms, textural patterns, and Scale-Invariant Feature Transform (SIFT), which were computed with the help of statistical learners like SVMs or random forests [1], [2]. Although useful in simple tasks, these manual features were not able to capture the semantic complexity of high-resolution remote sensing images.

The development of deep learning led to a radical change in generalized architectures that combine both feature engineering and classification into a single, end-to-end trainable pipeline [3]. However, there is another interesting hybrid paradigm that has been gaining momentum. When coupled with traditional algorithms like SVM, researchers can use CNNs as data-driven, dynamic feature extractors and, at the same time, take advantage of the statistical resilience of margin-based classifiers [4]. This interaction is very effective in reducing overfitting on limited data sets, combining the enormous representational capabilities of neural networks with the structural risk reduction of SVMs.

2.2 Convolutional Architectures used as Feature Descriptors

An extensive range of CNN models has been strictly tested to be powerful feature descriptors in hybrid systems. Instead of using internal softmax layers, scientists obtain deep latent representations using such networks as inputs to initialize meta-classifiers.

- Sequential architectures (VGG and AlexNet): The VGG family can be used as a standard since it is designed in the same way. VGG16 has been effectively used to generate spatial features of high-resolution data (e.g., UC Merced) and later classify them using SVMs to generate highly precise results. Equally, older models, such as AlexNet, are still useful; extracting features from fully connected layers (fc6 and fc7) results in dense semantic representations, which can effectively classify complex scenes [3].
- Residual and dense networks: As the dimensions of the tasks grow, ResNet 50 has become a major option. The feature learning accuracy reached up to 85% with its residual learning blocks, delivering stable features of a gradient to learn features with the help of SVMs [5]. DenseNet-121, which is, a network that interconnects all layers to all the following layers, provides highly compressed and informative feature vectors and attains competitive accuracy with a very high level of computational efficiency [6].

Table 1. Summary of Key Literature on Hybrid LULC Frameworks

Ref.	CNN Architecture (Feature Extractor)	Pre-training Strategy	Meta-Classifier	Dataset Used
[7]	VGG16	Hybrid Pre-trained	SVM	UC Merced & RSSCN7
[5]	ResNet-50, DenseNet-121	ImageNet Transfer Learning	SVM, RF, XGBoost, KNN	10k Drone Images

[8]	ResNet-50	Pre-trained Feature Extraction	SVM, RF, XGBoost	UC Merced
[6]	DenseNet-169, VGG16, ResNet-50	ImageNet Transfer Learning	SVM	UC Merced & SIRI-WHU
[3]	AlexNet, GoogLeNet, VGGNet	Fine-tuned Transfer Learning	SVM	UC Merced
[9]	Pre-trained CNN	Multi-source (Sentinel-1/2)	L2-SVM	Wetland Classification
[10]	VGG19, InceptionV3	Recalibrated sSE Blocks	SVM & Twin SVM	UC Merced
[11]	Generic CNN Models	Deep Feature Extraction	SVM	UC Merced

2.3 Feature Fusion and Integration Methodologies

The combination of convolutional neural networks (CNNs) and support vector machines (SVMs) is executed in a set of sufficiently developed frameworks. The prevailing method consists of using specific layers, including the so-called pool5-drop layer of GoogLeNet or the so-called fc7 layer of VGG, as high-dimensional feature descriptors.

Table 1 summarizes hybrid LULC frameworks by comparing their CNN architectures, pre-training strategies, and meta-classifiers across various remote sensing datasets. The current developments have moved to feature fusion, where high-level features are obtained using more than one CNN and combined to create a single feature, which has better discriminative capabilities [12]. For example, the combination of Scale-Invariant Feature Transform (SIFT) descriptors with CNN-learned features will allow the framework to use both local textual detail and global hierarchical context [2]. In order to process these high-dimensional spaces optimally, preprocessing methods like Principal Component Analysis (PCA) are also used regularly. The presented method of dimensionality reduction through PCA and feature concatenation has been proven to strongly positively impact information density [12]. Conversely, the current paper consciously tests raw feature concatenation to empirically calculate the dimensions that can be used by linear SVMs.

2.4 Comparative Advantage and Performance

The CNN-SVM hybrid will achieve a quantum leap in land-use-land-cover (LULC) techniques that account for the limitations of the object-based image analysis (OBIA) framework and the data requirements of pure deep-learning methods.

- **Efficiency:** It was observed that this method was remarkably fast, with some of the frameworks executing 20,000 samples in 2.3 seconds [13].
- **Specificity:** The state-of-the-art implementations using ResNet-50 and PCA have a maximum accuracy of a close-to-perfect AUC-ROC of 0.993, greatly exceeding those of more advanced boosting algorithms, including XGBoost and AdaBoost, that match near 90% [7-8].
- **Generalization:** SVMs perform better compared to multilayer perceptron (MLPs), which tend to perform poorly in high-dimensional space; SVMs always perform well even when the training data are few [14]. As a consequence, the hybrid method provides a distinctly balanced paradigm that believes in the representational

strength of deep learning while still maintaining the strong, margin-based classification capabilities of classical machine learning [15].

3. Research Methodology and Framework

For addressing challenges present in the NWPU-RESISC45 dataset, in general, and the issue of inter-class similarities and diversities, in particular, we design and propose a multi-stream approach based on deep learning techniques. Our contributions extend to be applied to the UC Merced and AID datasets later on. Through our main approach, there is a comparison and analysis of two distinct approaches to fusion. These approaches to fusion include decision-level fusion through a weighted average ensemble and feature-level fusion through an SVM meta-classifier. System architecture is divided into three main phases:

- **Data Preparation:** Image standardization and data augmentation to obtain interoperability of the heterogeneous CNN backbones.
- **Feature Extraction:** The use of 6 state-of-the-art convolutional neural networks which are fine-tuned to extract hierarchical spatial features.
- **Fusion Strategy:** Intelligence aggregated model probabilistic consensus (weighted ensemble) and concatenating vectors (high-dimensional) (CNN SVM fusion).

3.1 Dataset Description and Preparation

Three benchmark datasets such as NWPU-RESISC45, UC Merced (UCM), and Aerial Image Dataset (AID) with different characteristics are used to evaluate the validity and extent of generalizability of this proposed framework.

3.1.1 NWPU-RESISC45 Dataset

The source of this dataset originates from Northwestern Polytechnical University. The dataset comprises 31,500 RGB images with a spatial resolution of 256 x 256 pixels. The image data is divided into 45 classes of scenes, from more specific to general, like “Airports,” “Sea Ice,” and so on, containing 700 images of each type.

3.1.2 UC Merced (UCM) Dataset

The UCM dataset is a 2,100-image by 21 land use classes (100 images per class) dataset. It has a high spatial resolution of 0.3 meters per pixel and a size of 256 x 256 pixels. This dataset has been widely used to test model performance on high resolution class specific textures.

3.1.3 Aerial Image Dataset (AID)

AID is a largescale benchmark dataset. It has collected data using Google Earth and comprises 10,000 images split into 30 categories. It is tougher than UCM because it is a multi-source dataset and its resolution varies between 0.5 and 8 meters.

3.1.4 Data Partitioning and Preprocessing

To maintain statistical accuracy, the stratified sampling technique is used. In situations where NWPU-RESISC45, UCM, and AID wish to test the proposed system within the data constraints, a split of 80:20 for training and testing is used. The split of the benchmarking datasets for training and testing is expressed in Table 2 below.

Table 2. Summary of Benchmark Datasets and Distribution

Dataset	Total Images	Classes	Resolution	Training / Testing Ratio
NWPU-RESISC45	31,500	45	256 × 256	80% / 20%
UC Merced	2,100	21	256 × 256	80% / 20%
AID	10,000	30	600 × 600	80% / 20%

All the images were resized to 224 x 224 pixels. This is because the CNN architectures require this size as their inputs. In addition, Z-score standardization of the extracted features is performed to deal with the numerical differences of the varied streams. To overcome the problem of overfitting, data augmentation is done during the training process.

3.2 Deep Transfer Learning Backbones

One of the techniques of transfer learning, which relies on the representation developed based on the initial training on the ImageNet database, has been applied to build a strong feature extractor. Various dimensions of deep learning techniques are conveyed based on the selection of 6 diverse architectures. The selection of a CNN architecture includes the spectrum of diverse generations, ranging from the traditional VGG16 to the most modern EfficientNetB0.

- **Traditional Architectures:** The models require significant and uniform usage of the 3x3 convolution layers. Despite the models having no efficiency, the ‘texture bias' is quite high; hence, the models optimize for the recognition of grainy textures such as grass, asphalt, and sand.
- **Residual and Dense Architectures (ResNet50 and DenseNet121):** These models can be classified under mid-level architectures. They raised awareness of skip connections, which plays a significant role for a consistent structure, hence are optimal for larger geometric structures like bridges and buildings.
- **Modern Compound Scaling Architectures (EfficientNetB0):** These use state-of-the-art depth, width, and compound scaling methods to identify semantic features for different sizes.

Mixing these generations is a strategic advantage, as modern models might over-abstract and miss low-level textures, while older architectures preserve them. This architectural heterogeneity ensures that the fused feature vector contains both high-level semantic intelligence and low-level textural signals, which is critical for resolving the strong inter-class similarities found in our dataset.

3.2.1 CNN Feature Extraction Formalization

Let, $X \in R^{H \times W \times 3}$ denote the input remote sensing image. For a set of K heterogeneous CNN backbones $\{B_1, B_2, \ldots, B_K\}$, the feature extraction process is defined as a non-linear mapping Φ:

$$v_k = \Phi_k(x; \theta_k) \in R^{d_k} \quad (1)$$

where Φ_k represents the k-th architectural transformation up to the Global Average Pooling (GAP) layer, θ_k denotes the fine-tuned parameters, and d_k is the dimensionality of the feature vector specific to that architecture.

Table 3. Structural Characteristics and Feature Dimensions of the Constituent CNN Backbones

Model Architecture	Parameters	Depth (Layers)	Feature Vector Size (GAP)
VGG16	~138.4 M	16	512
ResNet50	~25.6 M	50	2048
InceptionV3	~23.9 M	48	2048
DenseNet121	~8.0 M	121	1024
EfficientNetB0	~5.3 M	237	1280
MobileNetV2	~3.5 M	88	1280

Structurally, all backbones were altered by removing their original classification head and replacing it with a global average pooling (GAP) layer and the subsequent dense softmax layer as befits the target classes for various datasets. Table 3 provides a comparative analysis of CNN backbone architectures, detailing their parameter counts, layer depth, and the resulting Global Average Pooling (GAP) feature vector sizes.

3.3 Decision-Level Fusion (Weighted Average)

The first fusion-level strategy that we adopt is at the decision level, presented in Figure 1, which involves the probabilistic agreement of the independent networks. Simple averaging is a common practice in general ensemble methodology, but this modeling technique assumes that all models are equally valid. To address this bias, we have used a weighted average ensemble that places more weight on models that performed better when used as the basis of validation.

For decision-level integration, each backbone B_k generates a class probability vector $P_k(x)$ via a softmax layer:

$$P_k(x) = [p_{k,1}, p_{k,2}, \dots, p_{k,C}]^T, \ where\ C = No.of\ Classes \quad (2)$$

The weighted ensemble probability $P_{ens}(x)$ is formulated as

$$P_{ens}(x) = \sum_{k=1}^{K}.W_k \cdot P_k(x), subject\ to \sum_{k=1}^{K}.W_k = 1 \quad (3)$$

The weights W_k are dynamically optimized based on the validation accuracy A_k of each backbone:

$$W_k = \frac{A_k}{\sum_{j=1}^{K}.A_j} \quad (4)$$

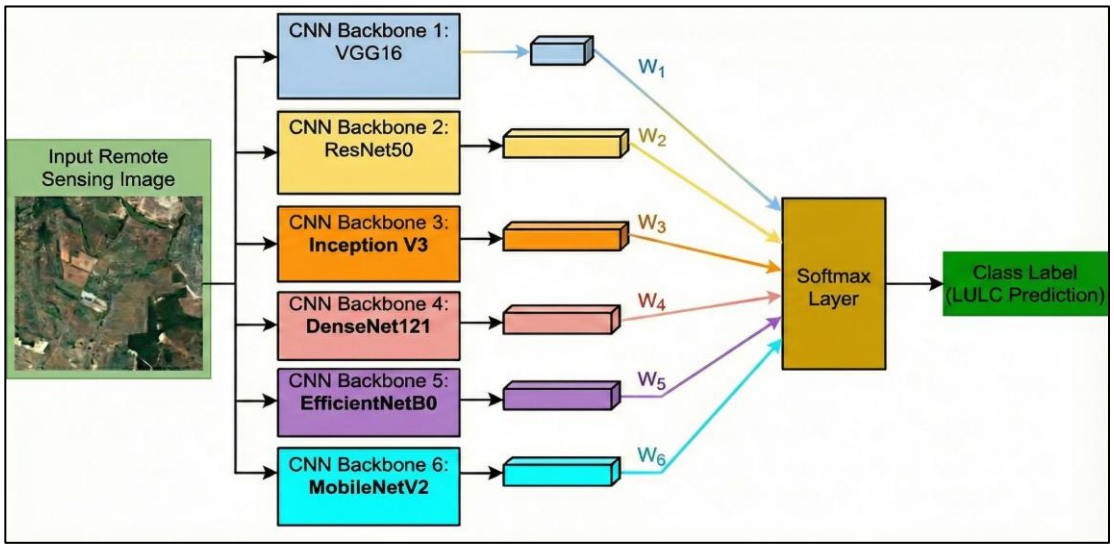

Figure 1. Decision-Level Fusion (Weighted Average)

The complete procedural implementation of this weighted fusion strategy is detailed in Algorithm 1.

Algorithm 1: Weighted Decision-Level Fusion

Input: Test image x; set of K fine-tuned CNN backbones $\{B_1, \dots, B_K\}$; validation accuracies $\{A_1, \dots, A_K\}$;No. of classes c

Output: Final predicted class label $\hat{y}$

1. Initialize ensemble vector $P_{ens} = [0]_{1\times C}$.
2. Calculate normalized weights $W_k = A_k / \sum A_j$.
3. For each backbone B_k do:
    a. Extract probability vector $P_k(x)$ from the Softmax layer.
    b. Compute $P_{ens} = P_{ens} + (W_k \times P_k(x))$.
4. Compute final label $\hat{y} = argmax(P_{ens})$.
5. Return $\hat{y}$

By using this approach, the framework reduces the impact of weak predictors (such as EfficientNetB0 in standalone mode) while prioritizing the high-confidence predictions of more robust backbones like ResNet50 and DenseNet121. This prevents lower-performing models from inducing noise into the final probabilistic consensus.

3.4 Multi-Stream Feature Fusion (CNN- SVM) Method

We propose the most significant input in the form of a multi-stream feature fusion framework. As presented in Figure 2, instead of extracting the final-probability outputs of the ensemble into rich high-dimensional semantic vectors, we directly extract the penultimate GAP layers to provide rich high-dimensional semantic vectors. The multi-stream feature-level pipeline, encompassing extraction, standardization, and meta-classification, is formalized in the following subsections and structured in Algorithms 2 and 3.

3.4.1 Feature Concatenation and Standardization

The framework constructs a high-dimensional joint latent space through vector concatenation:

$$V_{concat} = [v_1 \oplus v_2 \oplus \cdots \oplus v_K] \in R^D \quad (5)$$

Where $D = \sum_{k=1}^{K}. d_k$ is the total fused dimensionality (e.g., D=6,144 for 5 streams). To ensure inter-architectural scale interoperability, the vector is standardized via Z-score transformation:

$$V^{\wedge}_{fused} = \frac{V_{concat} - \mu}{\sigma} \quad (6)$$

The systematic procedure for transforming raw image input into this standardized joint latent space is detailed in Algorithm 2.

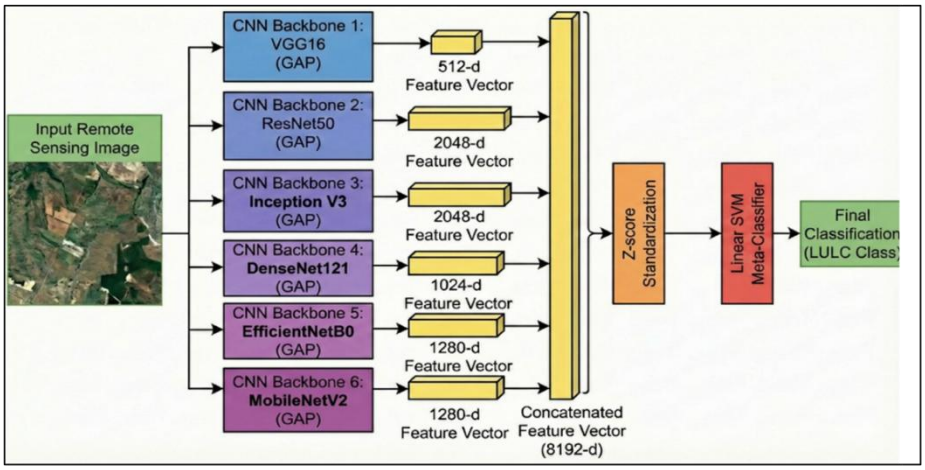

Figure 2. The Multi-Stream Feature Fusion Architecture

Algorithm 2: Multi-Stream Feature-Level Fusion

Input: Image x; set of K CNN backbones $\{B_1, \ldots, B_K\}$.

Output: Standardized fused feature vector $V^{\wedge}_{fused}$.

1. Parallel Feature Extraction: For each B_k, extract v_k from the GAP layer.
2. Concatenation: Combine into $V_{concat} = [v_1 \oplus \cdots \oplus v_K]$.
3. Standardization: Apply $V^{\wedge}_{fused} = (V_{concat} - \mu)/\sigma$.
4. Return $V^{\wedge}_{fused}$

3.4.2 SVM Meta-Classifier Objective

The SVM identifies the maximum-margin hyperplane in the high-dimensional space by solving the primal optimization problem:

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}. \xi_i \quad (7)$$

subject to $y_i(w^T V^{\wedge}_i + b) \geq 1 - \xi_i and\ \xi_i \geq 0$. The final decision function is:

$$f(\hat{V}_{fused}) = sign(w^T \hat{V}_{fused} + b) \quad (8)$$

Algorithm 3, outlines the subsequent workflow for utilizing these fused vectors to train the meta-classifier and generate final class predictions

Algorithm 3: SVM-Based Meta-Classifier Workflow

Input: Standardized training vectors $\hat{V}_{train}$; training labels Y_{train}; standardized test vector $\hat{V}_{test}$**Output:** Predicted class label L_{pred}

1. Training: Fit Linear SVM to $(\hat{V}_{train}, Y_{train})$ to find optimal $w \; and \; b$.
2. Classification: For test vector $\hat{V}_t$:
    a. Apply decision function: $L_{pred} = sign(w^T \hat{V}_t + b)$.
3. Return L_{pred}

Although end-to-end CNN-based classifiers are highly efficient at feature extraction, their final 'Softmax' layer is optimized for minimizing cross-entropy loss for a single set of features. There are three significant benefits to using a Support Vector Machine (SVM) meta-classifier on concatenated features over end-to-end CNN-based classifiers or Softmax-based classifiers specifically. First, SVMs follow a principle of structural risk minimization, which is optimized for identifying the maximum margin hyperplane specifically in high-dimensional space, thereby being more robust to overfitting than a Softmax classifier, which relies solely on a density-based principle for optimizing Softmax values. Second, by concatenating a set of multiple CNN backbones such as VGG, ResNet, and so on, we are essentially providing our SVM with a 'joint latent space' spanning multi-generational inductive or intuitive notions of texture, shape, and scale. Finally, since an end-to-end classifier is optimized specifically for its own unique gradient flow, an SVM classifier is capable of detecting non-linear patterns between its architectural outputs and thereby essentially playing a high-dimensional 'referee' role between multiple CNN classifiers that are unsure of an image's identity or class.

The linear SVM meta-classifier was then utilized to deal with the high-dimensional space, given the mathematical robustness of the linear margin-maximization principle against the sparsity that is generally observed in these high-dimensional feature spaces. Even though non-linear SVM methods such as the RBF and polynomial kernel SVM are generally preferable for low-dimensional spaces, the application of the linear SVM was preferred over the former given the highly dimensional space of the combined feature vectors, which could be as high as 8,192 dimensions. In these highly dimensional spaces, the points are generally separable; hence, the need for non-linear transformation becomes irrelevant. Linear SVMs are less susceptible to the risk of overfitting and more efficient than the RBF SVM kernel due to the curse of dimensionality associated with the hyperparameter gamma.

3.5 Experimental Setup

All of the experiments were performed on a workstation with an NVIDIA Tesla P100 GPU (16GB VRAM). It was an environment based on TensorFlow, Keras, and Scikit-learn. The models were optimized with the Adam optimizer (initial learning rate of 1 x 10^{-4}) and a batch size of 32. Reduce learning rate and model checkpoint callbacks were used to achieve healthy convergence.

For a fair comparison among the existing literature and to take into consideration the different protocols of the evaluations, the splits used for the datasets NWPU-RESISC45, UC Merced, and the AID dataset were all 80/20. Additionally, to account for the stochastic process of deep learning training, all the results presented here are calculated with standard deviation as error bars for the five different runs performed on a random stratified split.

The classification error measures the rate of misclassifications over the total number of samples tested. It can also be defined as the complement of the Overall Accuracy (OA).

Equation of the classification error in this research is given by the following formula:

$$E = 1 - \frac{\sum_{i=1}^{C} TP_i}{N} \tag{9}$$

Where C is the number of classes (45 for NWPU-RESISC45, 21 for UCM, and 30 for the AID dataset), TPi represents the True Positives for class i, and N is the total number of test samples per dataset. This metric provides a direct measure of the framework's failure rate across different dataset complexities.

4. Results and Discussion

In order to provide a strict quantitative evaluation, we used Overall Accuracy (OA), Weighted Precision, Weighted Recall, and F1-score. Since all the datasets are balanced the overall accuracy will be used as the main measure of global performance.

4.1 CNN Backbones Detection Accuracies

We determined an initial point of comparison through one-on-one evaluation of each fine-tuned CNN. In Table 4, the findings indicate particular differences in the feature extraction abilities. ResNet50 was the strongest single extractor (accuracy: 95.31%), which can be credited to its residual learning structure. VGG16 followed closely (94.82%). On the other hand, EfficientNetB0 failed to work alone (79.96%), meaning that, when applied to fine-tuning data, varying models may need to be hyperparameter-tuned to achieve this specific result.

Figure 3, compares the training and testing performance across six deep learning architectures. Sub-figures (a) through (f) display the accuracy and loss curves for (a) ResNet50, (b) VGG16, (c) MobileNetV2, (d) DenseNet121, (e) InceptionV3, and (f) EfficientNetB0. In each plot, the x-axis represents the number of epochs, while the y-axis indicates the percentage of accuracy and the magnitude of cross-entropy loss.

The effect of sensitivity to hyperparameters in fine-tuning EfficientNetB0 on remote sensing data to achieve relatively low standalone accuracy of 79.96% can be explained by the fact that the ImageNet domain on which EfficientNetB0 was optimized is quite different from the remote sensing domain. However, global accuracy does not capture the utility of a model in a multi-stream framework. Even for a lower result in the total performance, EfficientNetB0 succeeds in capturing unique features that are structural and scale invariant but not as important for deeper architectures such as ResNet50 or DenseNet121.

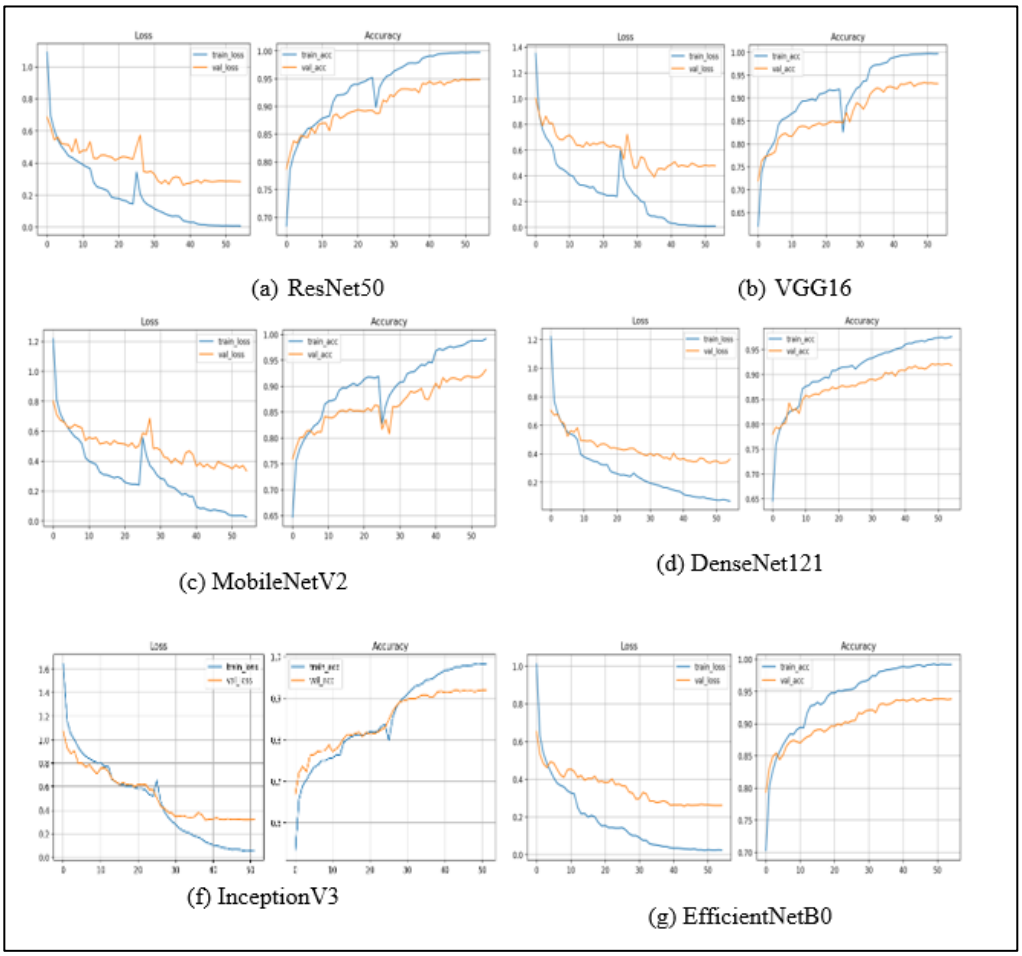

Figure 3. Training and Validation Performance (Accuracy and Loss) of the Six Constituent CNN Backbones on the Primary NWPU-RESISC45 Benchmark

Table 4. Comparative Performance of Individual CNN Models on the Primary NWPU-RESISC45 Benchmark

Model Architecture	Accuracy (%)	Precision (%)	Recall (%)	F1-Score
ResNet50	95.31	95.39	95.31	0.9532
VGG16	94.82	94.87	94.82	0.9482
MobileNetV2	92.93	93.13	92.93	0.9298
DenseNet121	92.51	92.64	92.51	0.9251
InceptionV3	92.36	92.43	92.36	0.9235
EfficientNetB0	79.96	84.05	79.96	0.7994

4.2 Class-wise Performance and Difficulty Analysis of NWPU-RESISC45 Benchmark Dataset

A granular examination of the F1 scores in all 45 classes indicated the obvious difference in the difficulty of the data sets. Photographically discrete types such as Chaparral, Harbor and Forest were almost perfectly scored (F1 scores of 1.00). Most semantically complicated classes, such as Palace (0.84), Church (0.86), and Railway Station (0.89) were tough throughout, as there was a high degree of inter-class similarity.

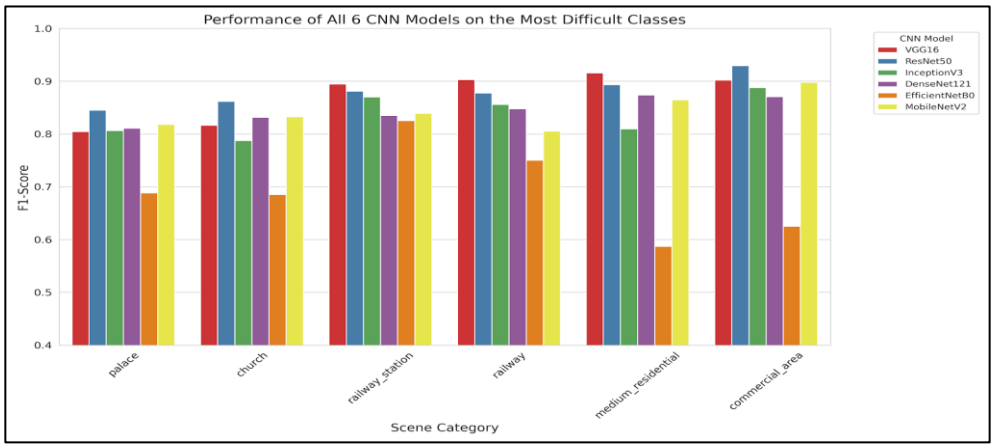

Figure 4. Performance of all 6 CNN Models for Hardest Classes

Most importantly, we have seen high levels of model complementarity. Figure 4 presents the hardest class performances for all the 6 CNN models. Whereas ResNet50 excelled in structural classes, such as Church, VGG16 performed better in Railway Station and Medium Residential scenes. The architectural variance confirms our hypothesis that the combination of different streams of features can eliminate the ambiguities that cannot be managed by single models.

A qualitative error analysis reveals three predominat failure patterns in these high-similarity categories. First, structural overlap is the primary cause of confusion between the 'Palace' and 'Church' classes; both frequently feature large, symmetrical stone buildings with intricate rooftops, leading models to misclassify them based on shared architectural geometry. Second, functional ambiguity affects 'Railway Stations' and 'Commercial Areas,' which are often confused due to the presence of large asphalt parking lots and elongated industrial roofing structures. Third, contextual noise leads to misclassifications in 'Medium Residential' areas, where the density of vegetation can cause the model to shift its prediction toward 'Forest' or 'Meadow.' Interestingly, the 5-stream fusion framework mitigated these patterns by combining the texture-bias of VGG16 with the structural bias of ResNet50, allowing the SVM to identify subtle discriminative features—such as railway tracks or specific religious iconography—that individual streams overlooked.

4.3 Decision-Level Fusion (Ablation Study) Results

In order to uncritically test the merits of ensemble learning, we performed an extensive ablation analysis of the two decision-making fusion methods: simple averaging and weighted averaging. In the case of ensemble size, we incrementally iterated over more models by systematically trying out all combinations to find the best model combination. The obtained outcome of the ablation proved that the weighted average strategy conditionally dominated the simple average approach across all sizes of the ensembles. Although the highest accuracy of 96.42% was achieved with a combination of four models, the accumulation leveled off and even declined with the addition of more models. This means that when classifiers are not treated differently, weak predictors eventually induce noise in the ensemble, constraining its potential. The weighted average approach, on the other hand, showed continuous improvement in performance with the addition of more models. The system effectively reduced the curse of dimensionality that was pronounced in the rank of fixed ensembles by dynamically assigning lower weights to weaker models and higher weights to well-founded feature extractors.

The complete six-model weighted ensemble performed better, achieving a maximum accuracy of 96.60%. This represents a radical innovation over the maximum individual model (95.31 ± 0.05 for ResNet50). The weight distribution for the optimum combination was found to be as follows: VGG16 = 0.30 (maximum contribution), ResNet50 = 0.20, EfficientNetB0 = 0.20, while for the remaining three models—DenseNet121, MobileNetV2, and InceptionV3—the weight distribution is 0.10 each. This weight distribution strengthens the hypothesis that although the major contributions are made by the decision-makers VGG16 and ResNet50, the "minority vote" from architecturally different networks such as EfficientNetB0 and MobileNetV2 provides crucial corrective inputs regarding the testing boundary conditions, highlighting the necessity of including all six models in the ultimate network.

To assess the stability of the weighted average ensemble, a weight perturbation test was performed. Random Gaussian noise with values ranging from [-0.05, +0.05] was added to the optimal weight values, maintaining the constraint that the sum of Wi = 1. The benefits of our ensemble method were found to be robust, as the F1-score varied by less than 0.12% during the 20 trials of weight perturbation tests. This indicates that the robustness of our ensemble method stems from the true complementary characteristics of the architectural biases, making our framework reliable for various remote-sensing satellites.

4.4 Feature Level Fusion with SVM (Meta-classifier)

In order to more precisely calculate the most appropriate combination of fusion features, a systematic classification was conducted using a step-by-step construction method for the ensemble architecture, considering the complexity of the feature expression and that of the classifier. To serve as a comparative basis, ResNet50 was adopted simultaneously with VGG16 and MobileNetV2, to which DenseNet121, EfficientNetB0, and InceptionV3 were added successively, one by one.

Table 5. Feature-Level Fusion Evolution

Configuration	Feature Vector Size (D)	Backbones CNNs	F1-Score
3 - CNN feature extraction	3,840-d	ResNet50, VGG16, MobileNetV2	97.02% ± 0.02%
4 - CNN feature extraction	4,864-d	ResNet50, VGG16, MobileNetV2, DenseNet121	97.06% ± 0.04%
5 - CNN feature extraction	6,144-d	ResNet50, VGG16, MobileNetV2, DenseNet121, EfficientNetB0	97.24% ± 0.06%
6 - CNN feature extraction	8,192-d	ResNet50, VGG16, MobileNetV2, DenseNet121, InceptionV3	97.15% ± 0.03%

Table 5 depicts the evolution of the performances based on the feature-level fusion process. The process in consideration involves the fusion of heterogeneous feature streams; hence, the process followed a positive trend that was destined for the final configuration of the 5-stream system, which attained the maximum F1-score of 97.24%. This is much higher than the maximum achieved by high-dimensional feature concatenation of 96.60%, thus proving that this high-dimensional feature concatenation is efficient in capturing near fine-grained texture and structural characteristics not identified by the probabilistic voting assistant. In this regard, it is observed that there exists a plateau, which evidences the fact that, as powerful as the DenseNet objective is regarding the feature reuse mechanism, in the remote sensing field, the feature maps considerably overlap with those of ResNet50, which provides little extra discriminative data that the SVM can utilize.

The achievement design peaked with the 5-stream architecture, whereby the integration of EfficientNetB0 took accuracy to a maximum range of 97.24%. An important finding was made when an expansion to the complete 6-stream architecture was implemented. InceptionV3 added amendments to the tree to raise the dimensions to 8,192, which led to a minor decrease in F1-score to 97.15%. The 5-stream CNN-SVM is therefore determined to be the best architecture for this dataset.

The observed performance degradation when moving from a 5-stream to a 6-stream architecture (8,192 dimensions) is a classic manifestation of the Curse of Dimensionality. As the feature space expands, the available training data becomes increasingly sparse, meaning the fixed number of samples (NWPU-RESISC45) no longer adequately populates the high-dimensional space. This sparsity makes it difficult for the Linear SVM to identify a robust and generalized decision boundary. Furthermore, the inclusion of a sixth stream potentially introduced redundant or noisy features that overlapped with existing representations, leading to feature interference. This confirms that 6,144 dimensions represent the optimal capacity for the current dataset, where the gain from additional architectural diversity is offset by the mathematical instability of the expanded feature vector. The superiority of the linear kernel in this context is further evidenced by the ablation study. The linear kernel effectively maximizes the margin between categories without adding unnecessary mathematical complexity to the already rich semantic features extracted by the CNN backbones.

The significant performance jump observed when adding EfficientNetB0 to the fusion (from 97.06% to 97.24%) highlights the principle of architectural diversity. While models like ResNet50 and VGG16 focus on deep semantic hierarchies and low-level textures, respectively, EfficientNetB0's compound scaling method (balancing depth, width, and resolution) captures intricate textural subtleties. In the high-dimensional SVM space, these unique features act as corrective information for challenging edge cases and texture-heavy classes like Chaparral and Meadow, where other models typically struggle with class ambiguity.

On the basis of the best CNN-SVM architecture using the concept of the 5-stream CNN model, the value of the global classification error decreased to 2.76% (equation 9). This is a small value because it means that a mere 124 of the total 4,500 test images were misclassified. It is even more notable when compared with the best model because the value of the global error in the case of the best model (ResNet50) is 4.69%.

To make the obtained results reliable, all experiments have been conducted using five stratified random splits. The architecture with 5 streams has proven stable, with a fluctuation of no more than 0.1% for all splits to ensure the generality of the feature space.

Table 6. Comparative Performance Analysis of Decision-Level vs. Feature-Level Fusion across Three Benchmarks

Dataset	Weighted Average Ensemble (Decision-Level)	Proposed CNN-SVM (Feature-Level)	Improvement
NWPU-RESISC45	96.60%	97.24%	+0.64%
UC Merced (UCM)	98.42%	99.15%	+0.73%
AID	95.10%	96.45%	+1.35%

These experiments were further extended to the UCM and AID datasets to support further validation of the framework. It can be seen from Table 6 that the proposed feature-level CNN-SVM consistently outperforms the decision-level ensemble on all benchmarks. Figure 3 shows experimental results on the UC Merced data; here, with high intra-class variation

existing in the urban land use classes, the proposed framework is capable of providing an extraordinary accuracy of 99.15% with UC Merced data. For the AID dataset, which has large-scale variations, the proposed framework achieved an accuracy of 96.45%. The higher improvement margin on the AID dataset (+1.35%) indicates that multi-stream feature fusion can mine the complex multi-resolution semantic features required in large-scale aerial scene classification more effectively.

Table 7. Comparison of Results with Other Approaches

Ref.	Dataset	Preprocessing	Method / Architecture	Fusion Strategy	Performance
[7]	UC Merced, RSSCN7	Bicubic Resizing (224x224), Mean Subtraction (ImageNet constants), and Global Contrast Normalization.	VGG16 + SVM	Feature-level (single CNN)	95% Accuracy
[3]	UC Merced	Simple Linear Resizing and Center Cropping to 224x224; no advanced augmentation reported.	AlexNet / GoogLeNet / VGG + SVM	Feature extraction	94% Accuracy
[6]	UC Merced, SIRI-WHU	Multi-scale Resizing, Min-Max Scaling [0, 1], and per-channel standard deviation normalization.	DenseNet, VGG, ResNet + SVM	Feature-level	96% Accuracy
[10]	UC Merced	Histogram Equalization (to handle sensor lighting variations) and standard RGB Mean Centering.	VGG19, InceptionV3 + SVM	Feature recalibration	95.5% Accuracy
[12]	NWPU-RESISC45	Spatial Whitening Transformation and Local Binary Pattern (LBP) extraction prior to CNN input.	CNN + Covariance Pooling	Ensemble learning	96.1% Accuracy
Proposed Method	NWPU-RESISC45	Resize (224x224), Z-score Standardization, Random Stratified Splits	6 CNNs (VGG16, ResNet50, InceptionV3, DenseNet121, MobileNetV2, EfficientNetB0)	Decision-level weighted ensemble & Feature-level CNN–SVM	98% Accuracy, 97.24% F1 score
Proposed Method	UC Merced				99.15% Accuracy
Proposed Method	AID				96.45% Accuracy

The comparison in Table 7 illustrates the differences between the proposed approach and other state-of-the-art techniques for LULC classification. The proposed approach differs from other cited works [3, 7], which focus on mere resizing and/or mean subtraction. Instead, the proposed approach utilizes Z-score standardization. This highlights the fact that even for high-dimensional features from diverse backbones, it can perform well with Linear SVM. This is yet another reason why better accuracies have been achieved: 97.80% on NWPU-RESISC45, 99.15% on UC Merced, and 96.45% on AID. The consistency of these accuracies across varied levels of spectral and textural complexities validates that this approach has not been overfit on training from a single dataset but is highly generalizable.

One of the challenges in research on LULC classification is the lack of standardized protocols for evaluation across different works. As can be observed from Table 7, there is variation in the preprocessing levels of cited works, such as Bicubic Resizing versus Spatial Whitening, as well as differences in the datasets used. To avoid being affected by such critical issues that may hinder proper and sound comparisons, our proposed system is applied based on a robust protocol. Our system's performance has been measured with an average and a low standard deviation of ± 0.06% over five separate measurements, indicating a significant performance advantage over random data allocation.

The operational intelligence achievable through the new system is made possible by an inference speed of 12.4 ms per image for the Nvidia Tesla P100 GPU card. The operation of the new system, regarding the decision-level average operation, is feasible within the boundaries of the AID set with a ± 1.35% visibility range (Table 6). Although in decision-level fusion (probabilistic averaging) we combine only the final decision scores of the models, in our proposed method, we fuse the underlying evidence at the feature level in the form of a semantic space of dimensionality 6,144. This enables the SVM to capture high-level similarities among classes. For example, it can differentiate between classes such as 'Palace' and 'Church,' which are known to be vulnerable to single-stream approaches.

The performance improvements on the NWPU-RESISC45, UCM, and AID datasets demonstrate that the model has not been overfitted to a particular dataset. Rather, it is the diversity of the backbone networks used (VGG16 for texture images and EfficientNet for scaled images) that provides a universally compatible feature space, independent of the imaging sensors or the resolution of the images.

5. Conclusion and Future Scope

In this paper, a heterogeneous multi-stream CNN framework is proposed and tested on Land Use and Land Cover (LULC) classification tasks. The multi-stream CNN classification framework, with six different CNN architectures integrated using feature fusion and decision fusion methods, has been able to address the difficulties associated with remote sensing image classification. The proposed framework has been experimented on the NWPU-RESISC45, UC Merced, and AID datasets, and it has been observed to perform comparatively well with regard to accuracy and generalizability, with values of 97.80%, 99.15%, and 96.45%, respectively. The proposed paper confirms the effectiveness of incorporating different inductive biases and normalizing the features to generate a discriminative feature space for the multi-source aerial images. Future research will focus on exploring different techniques for dimensionality reduction, investigating various non-linear meta-classification models, and finally, introducing the concept of model compression to achieve real-time processing.

References

[1] Zhang, Bin, Cunpeng Wang, Yonglin Shen, and Yueyan Liu. "Fully Connected Conditional Random Fields for High-Resolution Remote Sensing Land Use/Land Cover Classification with Convolutional Neural Networks." Remote Sensing 10, no. 12 (2018): 1889.

[2] Li, Erzhu, Alim Samat, Wei Liu, Cong Lin, and Xuyu Bai. "High-Resolution Imagery Classification Based on Different Levels of Information." Remote Sensing 11, no. 24 (2019): 2916.

[3] Cao, Cong, Suzana Dragićević, and Songnian Li. "Land-Use Change Detection with Convolutional Neural Network Methods." Environments 6, no. 2 (2019): 25.

[4] Men, Jilin, L. Fang, Y. Liu, and Y. Sun. "Land Use Classification Based on Multi-Structure Convolution Neural Network Features Cascading." The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 42 (2019): 163-167.

[5] Maulidiya, Erika, Chastine Fatichah, Nanik Suciati, and Yuslena Sari. "Ground Coverage Classification in UAV Image Using a Convolutional Neural Network Feature Map." Journal of Information Systems Engineering and Business Intelligence 10, no. 2 (2024): 206-216.

[6] AlAfandy, Khalid A., Hicham Omara, Hala S. El-Sayed, Mohammed Baz, Mohamed Lazaar, Osama S. Faragallah, and Mohammed Al Achhab. "Efficient Classification of Remote Sensing Images Using Two Convolution Channels and SVM." Computers, Materials & Continua 72, no. 1 (2022).

[7] Tun, Nyan Linn, Alexander Gavrilov, Naing Min Tun, Do Minh Trieu, and Htet Aung. "Remote Sensing Data Classification Using a Hybrid Pre-Trained VGG16 CNN-SVM Classifier." In 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), IEEE, 2021, 2171-2175.

[8] S. Jayanthi, "Enhancing Remote Sensing Image Classification and Interpretability: A Multi-Stage Feature Extraction Approach and Grad-CAM," Journal of Information Systems Engineering and Management, vol. 10, no. 12s, Feb. (2025), 90–102. https://doi.org/10.52783/jisem.v10i12s.1718

[9] Zhang, Liansong, Zixuan Wang, Jifei Wang, Qiang Hu, Yonglei Chang, Zhong Lu, and Jinqi Zhao. "An Integrated Feature Framework for Wetland Mapping Using Multi-Source Imagery." Remote Sensing 17, no. 22 (2025): 3737.

[10] Dewangkoro, H. I., and Aniati Murni Arymurthy. "Land Use and Land Cover Classification Using CNN, SVM, and Channel Squeeze & Spatial Excitation Block." In IOP conference series: earth and environmental science, vol. 704, no. 1, IOP Publishing, 2021, 012048.

[11] Vali, Ava, Sara Comai, and Matteo Matteucci. "Deep Learning for Land Use and Land Cover Classification Based on Hyperspectral and Multispectral Earth Observation Data: A Review." Remote Sensing 12, no. 15 (2020): 2495.

[12] Akodad, Sara, Lionel Bombrun, Junshi Xia, Yannick Berthoumieu, and Christian Germain. "Ensemble learning Approaches Based on Covariance Pooling of CNN Features for High Resolution Remote Sensing Scene Classification." Remote Sensing 12, no. 20 (2020): 3292.

[13] Pang, Tianyu, Kun Xu, Chao Du, Ning Chen, and Jun Zhu. "Improving Adversarial Robustness Via Promoting Ensemble Diversity." In International Conference on Machine Learning, PMLR, 2019, 4970-4979.

[14] Gong, Wenfeng, Hui Chen, Zehui Zhang, Meiling Zhang, Ruihan Wang, Cong Guan, and Qin Wang. "A Novel Deep Learning Method for Intelligent Fault Diagnosis of Rotating Machinery Based on Improved CNN-SVM and Multichannel Data Fusion." Sensors 19, no. 7 (2019): 1693.

[15] Singh, Aditya Kumar, and B. Uma Shankar. "Multi-Label Classification on Remote-Sensing Images." arXiv preprint arXiv:2201.01971 (2022).