

---

*Research article*

## **DSBF-Net: A dual-branch split-attention network with spatial bilinear fusion for airborne remote sensing scene classification**

**Huanyi Liu\* and Chenghao Zhao**

School of Service Management, Korea Aerospace University, Goyang-si 10540, Republic of Korea

\* **Correspondence:** Email: liuhuanyi651@kau.kr; Tel: 010-5932-8711.

**Abstract:** Airborne remote sensing scene classification is important for land-use surveys, urban planning, and environmental monitoring. However, large intra-class variation, strong inter-class similarity, and heterogeneous spatial structures make it difficult to coordinate representations extracted by different network pathways. To address this problem, this paper proposes a dual-branch split-attention network (DSBF-Net) with spatial bilinear fusion. EfficientNet and MobileNetV3 are employed as structurally heterogeneous backbones, Split-Attention is applied before and after fusion, and a spatial bilinear fusion module (SBFM) is introduced to align the two feature streams and model position-wise diagonal bilinear interactions. SBFM further incorporates signed square-root transformation, normalization, Coordinate Attention refinement, and residual aggregation. Experiments on the UC Merced Land-Use (UCM) dataset and Aerial Image Dataset (AID) show that DSBF-Net outperforms concatenation, addition, direct multiplication, and cross-attention fusion strategies, achieving classification accuracies of 99.07% and 97.58%, respectively. Component ablations further support the contribution of heterogeneous dual-branch learning, Split-Attention, and structured cross-branch fusion. The evaluation framework additionally examines backbone complementarity, hyperparameter sensitivity, corruption robustness, computational efficiency, interpretability, and limited-label behavior.

**Keywords:** remote sensing scene classification; dual-branch network; split attention; feature fusion; spatial bilinear fusion

---

## 1. Introduction

Remote sensing scene classification aims to assign a semantic label to an image according to its overall land-use or land-cover characteristics and has become an important component of geographic information interpretation, land-use investigation, urban planning, environmental monitoring, and related Earth observation applications. Unlike object-level recognition, scene classification generally depends on the joint interpretation of multiple land-cover elements, their spatial arrangement, and contextual relationships. With the increasing spatial resolution and diversity of airborne remote sensing imagery, individual scenes often contain multiple objects at different scales, heterogeneous background patterns, and complex structural layouts. Consequently, large intra-class variation and strong inter-class similarity remain persistent challenges for reliable scene discrimination [1,2].

Early scene classification methods primarily relied on handcrafted descriptors. Scale-invariant feature transform (SIFT) captures local scale- and rotation-invariant keypoint structure [3], histogram of oriented gradients (HOG) summarizes local gradient-orientation patterns [4], and bag-of-visual-words (BoVW) encodes local descriptors into an image-level visual vocabulary [5,6]. Although these approaches can encode local texture and edge information, their limited representation capacity makes it difficult to characterize high-level semantics and complex spatial dependencies. The development of deep learning has substantially changed this research paradigm. Convolutional neural networks (CNNs) provide effective hierarchical visual representations, while transformer-based architectures and graph-based models further strengthen long-range dependency modeling and relational reasoning [7–9]. Recent studies have pursued several complementary directions: frequency-spatial context modeling and high-order representation strengthen feature interaction across domains [2,10], hierarchical feature fusion and contextual attention improve multiscale and context-aware discrimination [9,11], and multilevel or dual-path representation strategies coordinate information across feature levels and architectural pathways [12–19]. These developments indicate that performance improvements increasingly depend on how complementary information from different scales, representation levels, or architectural pathways is coordinated rather than on increasing the capacity of a single feature extractor alone.

Dual-branch and heterogeneous feature-learning architectures provide one possible direction for addressing this issue. By processing the same input through structurally different feature extractors, such architectures can generate representation spaces with different receptive-field characteristics, nonlinear transformations, and feature-selection behaviors. Existing studies have explored global–local attention, dual-model feature fusion, master–slave encoding, and hybrid CNN-based representations for remote sensing interpretation [13,14,16,20]. Recent advances in adjacent remote sensing tasks also suggest that multi-stream architectures are effective only when cross-stream interaction is explicitly organized rather than passively delayed to a final concatenation layer. This observation motivates the present study to examine how heterogeneous single-source branches can be aligned and fused within a spatially preserved interaction module. Nevertheless, heterogeneous branches do not automatically guarantee complementary learning. When their outputs are directly concatenated or added, feature-scale mismatch, channel redundancy, and incompatible response distributions may weaken the expected benefit of multi-branch modeling. Therefore, the central issue is not merely whether multiple branches are used, but how cross-branch relationships are explicitly modeled after feature alignment.

Feature interaction constitutes another important aspect of this problem. Conventional concatenation and element-wise addition preserve first-order information but provide only limited explicit modeling of multiplicative relationships between feature streams. Bilinear models address pairwise feature interactions by constructing outer products between two feature vectors and have demonstrated strong representation capability in fine-grained recognition [21]. However, full bilinear representations are high-dimensional, motivating compact approximations and other efficient variants [22]. In remote sensing scene classification, recent studies have also investigated high-order semantic representation and enhanced feature correlation [10,12]. Despite their effectiveness, conventional full bilinear formulations may substantially increase feature dimensionality and often aggregate spatial responses into global descriptors, which is not ideal when spatial topology and directional layouts must be retained. This motivates a more selective interaction mechanism that preserves spatial correspondence while avoiding the  $(C^2)$ -dimensional expansion of a full outer product.

Attention mechanisms further provide an effective means of suppressing redundant responses and emphasizing discriminative features. Spatial-channel attention, focal attention, and semantic-aware fusion have been applied to remote sensing imagery to improve feature selection and contextual modeling [11,17,23]. Split-Attention introduces competition and adaptive aggregation among feature groups, enabling input-dependent channel recalibration [24]. Coordinate Attention embeds positional information into channel attention by separately encoding horizontal and vertical dependencies [25], which is potentially useful for remote sensing scenes containing elongated or direction-sensitive structures. Related global-context modeling strategies, such as adjacent-atrous mechanisms, also indicate the importance of expanding spatial contextual modeling for complex remote sensing scenes [26]. However, attention is often applied as an isolated enhancement operation before or after conventional fusion, without being tightly coupled with explicit cross-branch interaction.

Based on the above analysis, three issues remain insufficiently addressed. First, many scene-classification networks still rely on a single dominant representation pathway or use multiple branches without sufficiently examining whether the resulting features are genuinely complementary. Second, commonly used fusion operations, including concatenation, addition, and direct multiplication, either provide limited cross-feature interaction or risk information loss when multiplicative responses are used without alignment and residual preservation. Third, conventional bilinear pooling captures rich pairwise interactions but introduces substantial dimensional expansion and does not inherently preserve a spatially organized interaction tensor for subsequent direction-aware refinement.

To address these issues, this study proposes a dual-branch split-attention network (DSBF-Net) with spatial bilinear fusion for airborne remote sensing scene classification. EfficientNet [27] and MobileNetV3 [28] are employed as structurally heterogeneous feature extractors. Rather than presupposing fixed semantic roles for the two backbones, the proposed design uses their architectural heterogeneity to construct distinct feature representations, whose complementarity is further examined through comparative and visualization experiments. Split-Attention is introduced at the branch outputs to recalibrate informative responses before fusion and is applied again after fusion to refine the integrated representation. The core spatial bilinear fusion module (SBFM) first aligns heterogeneous features in a common spatial-channel space and then performs a position-wise diagonal bilinear interaction. Specifically, the proposed interaction corresponds to the diagonal of a local outer-product

matrix rather than conventional full bilinear pooling. This formulation retains matched cross-branch multiplicative responses with linear interaction complexity in the channel dimension, preserves the spatial topology of the feature map, and enables subsequent Coordinate Attention refinement and residual aggregation.

The main contributions of this study are summarized as follows.

A heterogeneous dual-branch framework, DSBF-Net, is developed for airborne remote sensing scene classification. EfficientNet and MobileNetV3 are jointly employed to construct structurally diverse representations, while branch-level Split-Attention adaptively suppresses redundant responses before cross-branch fusion.

SBFM is proposed to coordinate heterogeneous dual-branch features through channel alignment, position-wise diagonal bilinear interaction, signed square-root transformation, local  $\ell_2$  normalization, Coordinate Attention refinement, and residual aggregation. The proposed interaction is explicitly distinguished from conventional full and compact bilinear pooling and is designed to preserve spatial correspondence without  $C^2$ -dimensional outer-product expansion.

A two-stage attention strategy is incorporated into the network. Split-Attention is applied before and after fusion for adaptive response recalibration, while Coordinate Attention within SBFM introduces direction-aware spatial refinement into the interaction representation.

Experiments on the UC Merced Land-Use (UCM) and Aerial Image Dataset (AID) benchmarks evaluate the proposed method through comparisons of fusion strategies, backbone models, and component configurations. Additional analyses are designed to examine backbone complementarity, feature behavior, model efficiency, robustness, and classification errors under complex scene conditions.

The remainder of this paper is organized as follows. Section 2 reviews related studies on deep remote sensing scene classification, heterogeneous dual-branch learning, bilinear and high-order feature interaction, and attention-guided representation. Section 3 describes the proposed DSBF-Net and its principal components. Section 4 presents the experimental protocol, controlled comparisons, ablation analyses, robustness and efficiency evaluations, interpretability analyses, limited-label experiments, and category-level error analysis. Section 5 discusses the principal findings and limitations. Finally, Section 6 concludes the paper.

## 2. Related work

### 2.1. Deep learning for remote sensing scene classification

Remote sensing scene classification has evolved from handcrafted visual descriptors toward deep hierarchical representation learning. Traditional methods commonly combined local descriptors with encoding schemes such as BoVW, but their dependence on manually designed features limited their ability to represent complex semantic structures [3–6,8]. CNN-based methods subsequently became dominant because hierarchical convolutional features can jointly encode local visual patterns and increasingly abstract semantic information. Comprehensive reviews have shown that deep feature learning substantially improves scene representation, particularly in the presence of scale variation and complex background composition [6,8].

Recent research has moved beyond direct application of a single pretrained CNN and increasingly emphasizes multilevel representation and structural interaction. Chen et al. [9] combined hierarchical transformer features with patch-dilating operations to strengthen multiscale scene representation. Hou et al. [11] introduced contextual spatial-channel attention to improve discriminative response selection. EFCOMFF-Net [12], a multiscale feature-fusion architecture with enhanced feature correlation, explicitly strengthened feature correlations during multiscale fusion, whereas MSE-Net (master-slave encoding network) [14] adopted an asymmetric master-slave encoding strategy to coordinate different representation pathways. More recent studies have further explored frequency-spatial context modeling [2], multi-domain high-order representation [10], multiscale incremental learning [18], semi-supervised multilevel fusion [1], and hybrid recurrent-transformer modeling for complex scenes [19].

These methods demonstrate that remote sensing scene classification increasingly relies on the integration of information across scales, layers, and representation spaces. However, multilevel aggregation within a single dominant backbone does not fully address the problem of heterogeneous representation learning. Features generated by architectures with different computational structures may exhibit distinct response distributions and inductive biases, creating both opportunities and difficulties for cross-branch collaboration.

## *2.2. Dual-branch and heterogeneous feature learning*

Dual-branch architectures have been introduced to exploit complementary representations that are difficult to obtain from a single pathway. DBGA-Net (dual-branch global-local attention network) [13], for example, combines global and local attention within a dual-branch framework for remote sensing scene classification. Guo et al. [16] developed a dual-model architecture with multilevel feature fusion for land-use and land-cover scene classification, while HFCC-Net [20] combined convolutional neural network (CNN) and capsule network (CapsNet) representations in a dual-branch hybrid framework. MSE-Net [14] similarly used asymmetric master-slave encoding to improve scene representation.

The central advantage of heterogeneous branches is that different network structures can transform the same input into feature spaces with different receptive-field properties and response patterns. Nevertheless, this heterogeneity also creates a fusion problem. Direct concatenation preserves both streams but leaves cross-branch dependency modeling to subsequent layers; element-wise addition assumes sufficiently compatible representations; and unrefined multiplication may suppress useful information when one branch produces a weak response. Consequently, effective dual-branch learning requires not only multiple encoders but also explicit alignment and interaction mechanisms.

Evidence from adjacent remote sensing tasks reinforces the importance of explicit interaction between heterogeneous streams. DBRSNet [29], a dual-branch remote-sensing segmentation network, combines dual-branch modeling with feature interaction and multiscale fusion for remote sensing image segmentation. More recent multimodal frameworks further show that the benefit of multiple streams depends not only on using separate encoders but also on designing where and how cross-stream information is exchanged. For example, FTransMamba [30], a multi-stage fusion Transformer-Mamba network, coordinates transformer-based representation and Mamba-based sequential modeling, indicating the value of progressive fusion across different representation spaces. HDC-Net [31], a

hierarchical dual-stream fusion network with cross-token interaction, is designed for multimodal semantic segmentation, showing that cross-stream information exchange can be organized at multiple semantic levels rather than delayed until the final classifier.

These studies are not used in this paper as direct baselines because they address different task settings, modalities, or segmentation-oriented outputs. Instead, they inform the methodological motivation of DSBF-Net in three aspects. First, feature streams should be aligned before interaction rather than simply concatenated. Second, cross-stream interaction should be introduced at the feature-map level so that spatial structure is retained. Third, interaction responses should be refined after fusion because cross-stream exchange can introduce redundant or dominant activations. The proposed SBFM translates these principles into a single-source scene-classification setting: Two heterogeneous CNN branches process the same airborne image, their spatial-channel representations are aligned, matched channel responses are modeled through position-wise diagonal interaction, and the resulting interaction tensor is refined by Coordinate Attention and residual aggregation.

### 2.3. Bilinear and high-order feature interaction

Feature fusion methods differ substantially in the order and structure of interactions they represent. Concatenation stacks features along the channel dimension but does not explicitly model pairwise cross-feature products. Addition performs linear superposition and requires compatible feature dimensions. Multiplicative fusion introduces input-dependent interactions but may become overly selective when used without alignment or residual information preservation.

Bilinear models provide a classical mechanism for explicitly modeling pairwise feature interactions. Lin et al. [21] proposed bilinear CNNs in which two feature vectors at the same spatial location are combined through an outer product and subsequently pooled to form an image-level representation. For feature vectors  $x_p \in \mathbb{R}^{C_x}$  and  $y_p \in \mathbb{R}^{C_y}$ , a full local bilinear representation can be written as

$$G_p = x_p y_p^T, \quad (1)$$

where  $G_p \in \mathbb{R}^{C_x \times C_y}$  contains all cross-channel pairwise products. Although expressive, this representation grows quadratically with channel dimensionality when  $C_x = C_y = C$ . Compact bilinear pooling was therefore introduced to approximate the discriminative behavior of full bilinear representations in a substantially lower-dimensional space [22].

High-order interaction has also attracted attention in remote sensing scene classification. Multi-domain semantic high-order modeling [10] demonstrates the value of going beyond purely first-order representations, while enhanced feature-correlation architectures [12] explicitly strengthen relationships among multiscale features. However, conventional bilinear pooling and high-order global descriptors are not directly optimized for a fusion setting in which spatial topology must remain available for subsequent directional refinement.

The proposed SBFM therefore adopts a restricted form of bilinear mapping. After spatial and channel alignment, it retains only the diagonal entries of the local outer-product matrix:

$$b_p = \text{diag}(x_p y_p^T) = x_p \odot y_p. \quad (2)$$

Accordingly, the proposed operation should not be interpreted as full bilinear pooling, compact bilinear pooling, or second-order covariance pooling. It is a position-wise diagonal bilinear interaction that models matched cross-branch channel products while preserving an  $H \times W$  spatial organization. This distinction is central to the design of SBFM.

#### 2.4. Attention-guided spatial representation

Attention mechanisms are widely used to improve feature selectivity by assigning input-dependent importance to channels, spatial locations, or feature groups. In remote sensing scene classification, contextual spatial-channel attention [11], gradient-guided multiscale focal attention [17], and semantic-aware fusion [23] have demonstrated the value of adaptively emphasizing discriminative information. Such mechanisms are particularly relevant to remote sensing scenes because classification may depend on a limited number of structurally informative regions embedded in complex backgrounds.

Split-Attention [24] extends channel recalibration by dividing feature maps into multiple groups and learning competitive weights across feature subsets. Compared with a single deterministic channel-rescaling pathway, this mechanism provides input-dependent aggregation across grouped representations. In DSBF-Net, Split-Attention is employed at two stages: before fusion to improve the quality of branch-specific representations and after fusion to suppress redundant responses introduced during cross-branch integration.

Coordinate Attention [25] addresses a different limitation of conventional channel attention. Instead of compressing the entire two-dimensional spatial domain into a single global descriptor, it separately aggregates information along the horizontal and vertical directions. This design embeds positional information into attention generation and retains direction-dependent spatial cues. Such properties are relevant to airborne remote sensing imagery containing elongated or regularly organized structures, including roads, runways, bridges, and building layouts. Related remote sensing research has also highlighted the importance of enlarged global receptive fields and spatial-context modeling; for example, the adjacent-atrous mechanism in [26] was developed to strengthen global contextual representation for multiattribute scene analysis.

Despite these advances, three gaps remain. First, attention and fusion are frequently designed as separate operations such that attention improves individual features without explicitly defining how heterogeneous branches interact. Second, direct multiplicative fusion can model joint activation but may discard branch-specific evidence. Third, full bilinear interaction provides rich pairwise products at a substantial dimensional cost and is not inherently designed to retain a compact spatial interaction tensor.

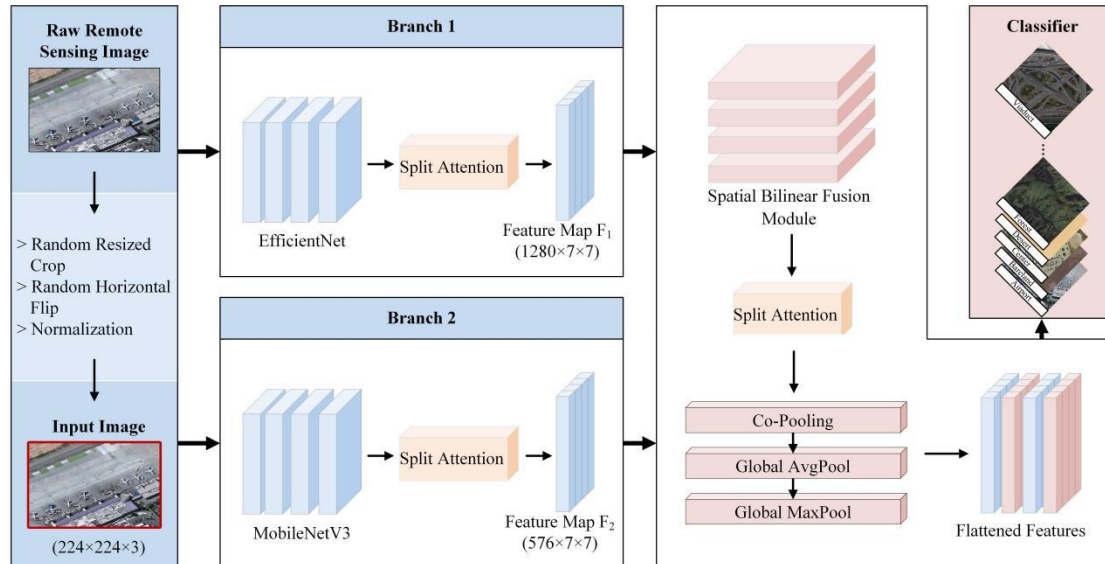
DSBF-Net addresses these gaps by combining branch-level Split-Attention, spatial-channel alignment, position-wise diagonal bilinear interaction, Coordinate Attention refinement, and residual aggregation. The resulting design explicitly separates feature recalibration, cross-branch interaction, direction-aware refinement, and first-order information preservation within a unified fusion process.

### 3. Methodology

Remote sensing scene images usually contain multiple types of land-cover objects and are often accompanied by substantial scale variation, complex spatial organization, and strong background interference [29]. Therefore, scene classification depends not only on local textures or edge cues but also on the joint modeling of scene-level semantics, contextual relationships among objects, and the overall spatial layout. Based on this understanding, this study proposes DSBF-Net. The core idea of the proposed network is as follows: Two backbone networks with different structural characteristics are employed to extract complementary features; attention mechanisms are introduced within each branch to suppress redundant responses; cross-branch spatial interactions are explicitly modeled during feature fusion; and stable scene representations are finally obtained through global statistical aggregation before classification.

#### 3.1. Overall architecture of DSBF-Net

The overall architecture of the proposed DSBF-Net is illustrated in Figure 1. Given an input airborne remote sensing image, the network first extracts heterogeneous feature representations through two parallel backbone branches, subsequently recalibrates branch-specific responses using Split-Attention, performs explicit cross-branch interaction through the proposed SBFM, and finally refines the fused representation before global aggregation and classification.



**Figure 1.** Overall architecture of the proposed DSBF-Net.

Let the input image be denoted by

$$I \in \mathbb{R}^{B \times 3 \times H_0 \times W_0}, \quad (3)$$

where  $B$  denotes the batch size and  $H_0$  and  $W_0$  denote the spatial dimensions of the input image. The two backbone feature extractors are represented by  $\Phi_E(\cdot)$  and  $\Phi_M(\cdot)$ , corresponding to EfficientNet [27] and MobileNetV3 [28], respectively. Their output features are written as

$$F_E = \Phi_E(I), \quad (4)$$

and

$$F_M = \Phi_M(I), \quad (5)$$

where

$$F_E \in \mathbb{R}^{B \times C_E \times H_E \times W_E} \quad (6)$$

and

$$F_M \in \mathbb{R}^{B \times C_M \times H_M \times W_M}. \quad (7)$$

EfficientNet and MobileNetV3 are selected as structurally heterogeneous feature extractors rather than being assigned fixed semantic roles a priori. EfficientNet employs compound scaling to coordinate network depth, width, and input resolution [1], whereas MobileNetV3 adopts a lightweight architecture based on mobile inverted bottlenecks and hardware-aware design [2]. Consequently, the two backbones transform the same input through different architectural pathways and may produce feature representations with different response distributions, receptive-field characteristics, and channel organizations. DSBF-Net is designed to exploit this structural diversity through explicit feature recalibration and cross-branch interaction. The extent to which the two branches exhibit complementary behavior is further examined through backbone-combination comparisons and feature-visualization experiments rather than assumed solely from architectural intuition.

Before cross-branch fusion, Split-Attention is independently applied to the two branch outputs. The recalibrated features are expressed as

$$\tilde{F}_E = \mathcal{A}_E(F_E), \quad (8)$$

and

$$\tilde{F}_M = \mathcal{A}_M(F_M), \quad (9)$$

where  $\mathcal{A}_E(\cdot)$  and  $\mathcal{A}_M(\cdot)$  denote the branch-level Split-Attention operations. These modules adaptively recalibrate candidate feature groups according to the current input, thereby reducing redundant responses before heterogeneous feature interaction.

The enhanced branch features are subsequently fed into SBFM:

$$F_S = \mathcal{S}(\tilde{F}_E, \tilde{F}_M), \quad (10)$$

where  $\mathcal{S}(\cdot, \cdot)$  denotes the proposed fusion operator. As described in Section 3.3, SBFM first aligns the two feature streams in a common spatial-channel space and then performs position-wise diagonal bilinear interaction, signed square-root and  $\ell_2$  normalization, Coordinate Attention refinement, and residual aggregation. The resulting tensor

$$F_S \in \mathbb{R}^{B \times C \times H \times W} \quad (11)$$

therefore retains both aligned branch information and explicit multiplicative interaction responses.

Because cross-branch fusion may introduce new redundant or overly dominant activations, a second Split-Attention stage is applied after SBFM:

$$\tilde{F}_S = \mathcal{A}_P(F_S), \quad (12)$$

where  $\mathcal{A}_P(\cdot)$  denotes the post-fusion Split-Attention operation. This second recalibration stage differs functionally from the branch-level modules. The pre-fusion attention modules operate independently on the two backbone representations and improve the quality of the features entering SBFM, whereas the post-fusion module acts on the integrated representation and suppresses redundant responses generated during feature interaction and residual aggregation.

To construct a scene-level descriptor, DSBF-Net employs additive global average-max co-pooling. Global average pooling (GAP) summarizes the distributed response of each channel across the spatial domain:

$$g_{avg} = GAP(\tilde{F}_S), \quad (13)$$

whereas global max pooling (GMP) retains the strongest spatial activation within each channel:

$$g_{max} = GMP(\tilde{F}_S). \quad (14)$$

The two descriptors are combined through element-wise addition to form the final scene representation:

$$g = g_{avg} + g_{max}. \quad (15)$$

GAP and GMP provide complementary statistical summaries: The former captures distributed channel-level evidence across the spatial domain, whereas the latter preserves the strongest localized response. Element-wise addition introduces no extra learnable parameters and preserves the fused channel dimensionality.

The classification head consists of feature normalization, dropout regularization, and a linear classifier. The predicted class probability is expressed as

$$\hat{y} = \text{Softmax}(W \text{Drop}(\text{BN}(g)) + b), \quad (16)$$

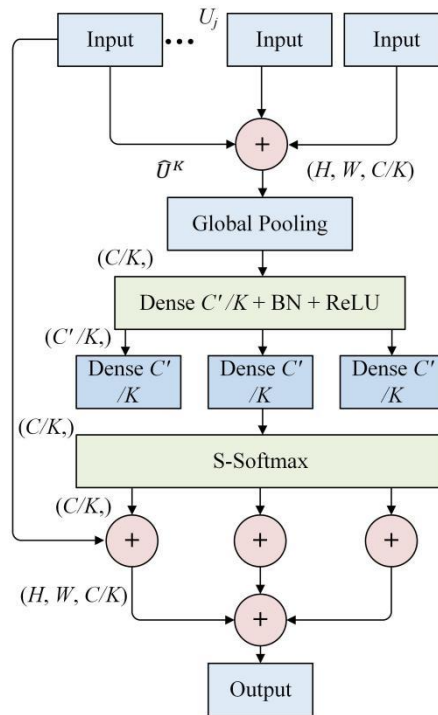
where  $\text{BN}(\cdot)$  denotes batch normalization,  $\text{Drop}(\cdot)$  denotes dropout, and  $W$  and  $b$  are the learnable weight matrix and bias vector of the classifier, respectively.

The complete forward process of DSBF-Net can therefore be summarized as

$$I \rightarrow \{\Phi_E, \Phi_M\} \rightarrow \{\mathcal{A}_E, \mathcal{A}_M\} \rightarrow \mathcal{S} \rightarrow \mathcal{A}_P \rightarrow \{GAP, GMP\} \rightarrow \text{Classifier}.$$

This architecture separates heterogeneous feature extraction, branch-specific recalibration, explicit cross-branch interaction, post-fusion refinement, and global statistical aggregation into distinct functional stages. Such a design allows the contribution of each component to be independently evaluated through ablation, backbone-combination, visualization, and complexity analyses.

### 3.2. Split-attention mechanism



**Figure 2.** Structure of the Split-Attention block used in DSBF-Net.

The structure of the Split-Attention block used in DSBF-Net is illustrated in Figure 2. Split-Attention is employed in DSBF-Net to adaptively recalibrate grouped feature responses according to the current input. The mechanism follows a split-aggregate-select procedure inspired by Split-Attention networks [24]. Unlike conventional channel attention that directly rescales a single feature tensor, Split-Attention constructs multiple candidate feature groups and learns input-dependent competition among them. In the proposed network, this mechanism is applied independently to the two branch outputs before SBFM and once again to the fused representation after SBFM.

Let the input feature tensor be

$$F \in \mathbb{R}^{B \times C \times H \times W}. \quad (17)$$

The feature transformation is divided into  $K$  candidate groups:

$$\{U_1, U_2, \dots, U_K\}, \quad (18)$$

where

$$U_k \in \mathbb{R}^{B \times C \times H \times W} \quad (19)$$

denotes the  $k$ -th transformed candidate representation. For notational simplicity, the internal grouped convolution and feature transformation operations are denoted by  $\mathcal{T}_k(\cdot)$ :

$$U_k = \mathcal{T}_k(F), k = 1, \dots, K. \quad (20)$$

The candidate representations are first aggregated by element-wise summation:

$$U = \sum_{k=1}^K U_k. \quad (21)$$

This aggregated tensor collects information from all candidate groups and is used to generate an input-dependent global descriptor. Specifically, GAP is applied across the spatial dimensions:

$$s_c = \frac{1}{HW} \sum_{h=1}^H \sum_{w=1}^W U_c(h, w), \quad (22)$$

yielding

$$s \in \mathbb{R}^{B \times C}. \quad (23)$$

The descriptor  $s$  summarizes the global response of each channel after candidate aggregation. A compact intermediate representation is then generated through a learnable bottleneck transformation:

$$z = \delta(BN(W_1 s)),$$

where  $W_1$  denotes a learnable projection matrix,  $BN(\cdot)$  denotes batch normalization, and  $\delta(\cdot)$  denotes a nonlinear activation function. The dimensionality of the intermediate representation can be expressed as

$$d = \max\left(\left\lfloor \frac{C}{r} \right\rfloor, d_{\min}\right), \quad (24)$$

where  $r$  is the reduction ratio and  $d_{\min}$  prevents excessive compression when the channel dimension is small.

Based on the shared descriptor  $z$ , a branch-specific attention logit is generated for each candidate group:

$$q_k = W_k z, k = 1, \dots, K, \quad (25)$$

where  $W_k$  is the learnable projection associated with the  $k$ -th candidate representation.

When multiple candidates are present, channel-wise normalized attention weights are obtained through a softmax operation across the candidate dimension:

$$a_{k,c} = \frac{\exp(q_{k,c})}{\sum_{j=1}^K \exp(q_{j,c})}. \quad (26)$$

Therefore,

$$\sum_{k=1}^K a_{k,c} = 1 \quad (27)$$

for each channel  $c$ . This normalization establishes competition among candidate representations on a channel-by-channel basis rather than assigning a single scalar weight to an entire feature branch. In the implementation of DSBF-Net, the number of Split-Attention candidate groups is denoted by  $K$ . Unless otherwise specified,  $K = 2$  is used for both branch-level Split-Attention and post-fusion Split-Attention in all main experiments. This setting provides a minimal competitive grouping mechanism while avoiding the excessive computational cost introduced by a larger number of candidate groups. When  $K = 1$ , the softmax competition across candidate groups degenerates into a single-path recalibration, whereas larger values of  $K$  increase the number of candidate transformations and the corresponding attention-generation cost. Therefore,  $K$  is treated as an explicit architectural hyperparameter and further examined in the sensitivity analysis in Section 4.8.

The final Split-Attention output is calculated as

$$F_{SA} = \sum_{k=1}^K a_k \odot U_k, \quad (28)$$

where  $a_k$  is broadcast along the spatial dimensions and  $\odot$  denotes element-wise multiplication. Accordingly,

$$F_{SA} \in \mathbb{R}^{B \times C \times H \times W}, \quad (29)$$

and the spatial resolution of the feature map is preserved.

The complete Split-Attention process can be summarized as

$$F \rightarrow \{U_1, \dots, U_K\} \rightarrow \sum_{k=1}^K U_k \rightarrow GAP \rightarrow z \rightarrow \{a_1, \dots, a_K\} \rightarrow F_{SA}.$$

In DSBF-Net, Split-Attention is deployed at two functionally distinct stages.

First, branch-level Split-Attention is independently applied to the outputs of EfficientNet and MobileNetV3:

$$\tilde{F}_E = \mathcal{A}_E(F_E), \quad (30)$$

$$\tilde{F}_M = \mathcal{A}_M(F_M). \quad (31)$$

At this stage, the objective is to recalibrate each heterogeneous feature stream before cross-branch interaction. Because the two backbones may produce different response distributions and redundant activations, independent attention allows informative feature groups to be emphasized within each branch without forcing premature correspondence between heterogeneous channels.

Second, post-fusion Split-Attention is applied to the SBFM output:

$$\tilde{F}_S = \mathcal{A}_P(F_S). \quad (32)$$

This stage operates on a representation that already contains aligned branch information, multiplicative interaction responses, Coordinate Attention refinement, and residual components. Its

purpose is therefore different from that of branch-level attention: It adaptively redistributes emphasis among the integrated feature responses and suppresses redundancy that may emerge during fusion.

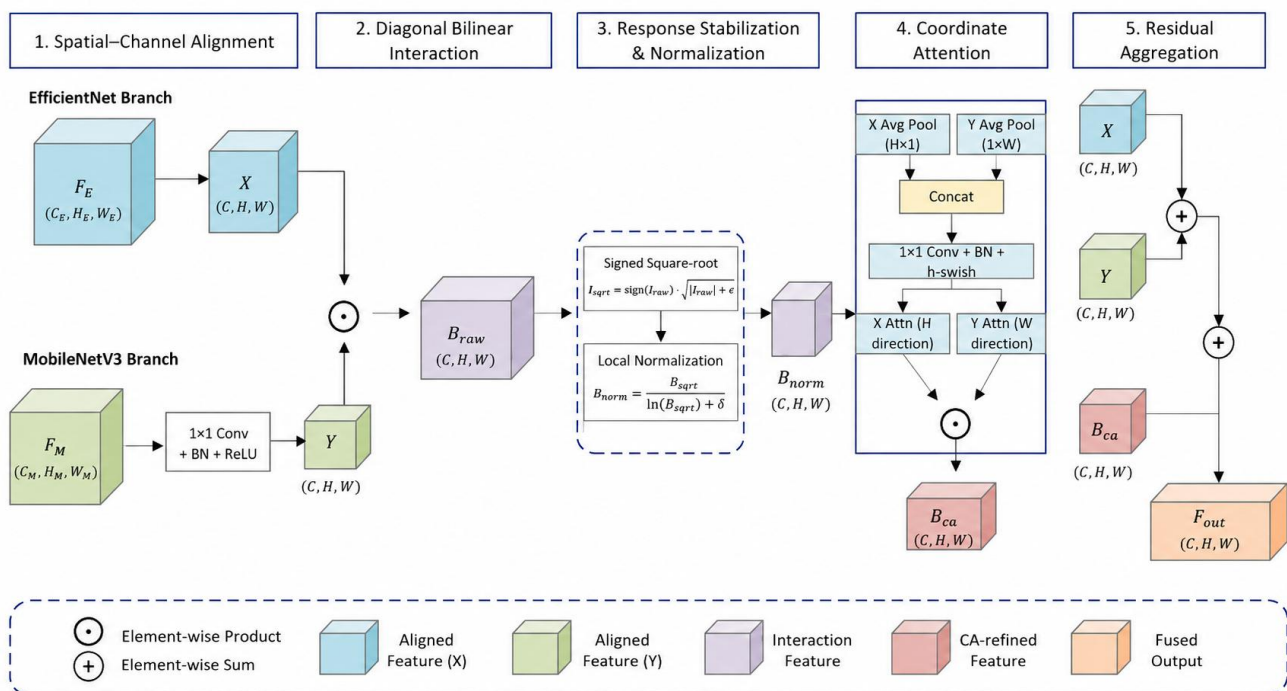
The two-stage deployment can be expressed compactly as

$$(F_E, F_M) \xrightarrow{\mathcal{A}_E, \mathcal{A}_M} (\tilde{F}_E, \tilde{F}_M) \xrightarrow{\mathcal{S}} F_S \xrightarrow{\mathcal{A}_P} \tilde{F}_S.$$

Thus, Split-Attention in DSBF-Net is not used as an isolated attention block. It forms a two-stage recalibration mechanism surrounding the cross-branch fusion process: Pre-fusion attention improves the quality of heterogeneous inputs to SBFM, whereas post-fusion attention refines the integrated representation before global pooling and classification.

### 3.3. Spatial bilinear fusion module

SBFM is designed to coordinate heterogeneous dual-branch features while retaining their spatial organization. As illustrated in Figure 3, SBFM consists of five sequential stages: (1) spatial-channel alignment, (2) position-wise diagonal bilinear interaction, (3) signed square-root and normalization, (4) Coordinate Attention refinement, and (5) residual aggregation.



**Figure 3.** Structure of the proposed SBFM.

A clarification of terminology is necessary. Conventional bilinear pooling generally constructs an outer product between two local feature vectors and therefore models all pairwise cross-channel interactions [21]. Compact bilinear pooling approximates such a high-dimensional representation through a lower-dimensional mapping [22]. The proposed SBFM does not compute either a full outer product or a compact approximation to it. Instead, after channel alignment, it retains the diagonal terms

of the local outer-product matrix. The resulting raw interaction is equivalent to an aligned Hadamard product in implementation. Therefore, the proposed contribution should not be interpreted as replacing direct multiplication with a fundamentally different primitive operator. Rather, SBFM uses this aligned Hadamard product as a restricted diagonal bilinear interaction and embeds it within a complete fusion pipeline consisting of spatial-channel alignment, response stabilization, Coordinate Attention refinement, and residual aggregation. In this sense, the distinction between SBFM and direct multiplication lies not in the elementary product  $X \odot Y$  itself but in how this product is aligned, normalized, refined, and combined with first-order branch evidence. The term position-wise diagonal bilinear interaction is used to emphasize that the product corresponds to the diagonal of a local outer-product matrix while avoiding the quadratic channel expansion associated with full bilinear interaction.

### 3.3.1. Spatial-channel alignment

Let the Split-Attention-enhanced outputs of the EfficientNet and MobileNetV3 branches be denoted by

$$F_E \in \mathbb{R}^{B \times C_E \times H_E \times W_E}, \quad (33)$$

and

$$F_M \in \mathbb{R}^{B \times C_M \times H_M \times W_M}, \quad (34)$$

respectively, where  $B$  denotes the batch size,  $C$  denotes the number of channels, and  $H$  and  $W$  denote spatial dimensions.

Because heterogeneous backbones may produce features with different channel numbers and spatial resolutions, direct element-wise interaction is not well defined. The two streams are therefore first mapped into a common space. Taking the EfficientNet-side representation as the reference space, the aligned features are written as

$$X = F_E, \quad (35)$$

and

$$Y = \rho \left( \text{BN} \left( \text{Conv}_{1 \times 1} \left( \mathcal{R}(F_M) \right) \right) \right), \quad (36)$$

where  $\mathcal{R}(\cdot)$  denotes spatial resizing when the branch outputs have different spatial resolutions,  $\text{Conv}_{1 \times 1}(\cdot)$  performs channel projection,  $\text{BN}(\cdot)$  denotes batch normalization, and  $\rho(\cdot)$  denotes the ReLU activation function. After alignment,

$$X, Y \in \mathbb{R}^{B \times C \times H \times W}. \quad (37)$$

The  $1 \times 1$  projection does not mix neighboring spatial positions. Its purpose is to construct a compatible channel space in which location-wise cross-branch interactions can be defined. Spatial resizing, when required, establishes correspondence between the two feature grids before multiplicative interaction.

### 3.3.2. Position-wise diagonal bilinear interaction

For clarity, consider a spatial position  $p = (h, w)$ . Let

$$x_p, y_p \in \mathbb{R}^C \quad (38)$$

denote the aligned feature vectors extracted from  $X$  and  $Y$ , respectively.

A conventional full bilinear interaction would construct

$$G_p = x_p y_p^\top \in \mathbb{R}^{C \times C}, \quad (39)$$

where each element

$$G_{p,i,j} = x_{p,i} y_{p,j} \quad (40)$$

represents an interaction between channel  $i$  of one branch and channel  $j$  of the other branch. Although expressive, this operation produces  $C^2$  interaction terms at each spatial position.

SBFM instead extracts the diagonal of the local outer-product matrix:

$$b_p = \text{diag}(x_p y_p^\top) = x_p \odot y_p. \quad (41)$$

Equivalently, for each channel  $c$ ,

$$b_{p,c} = x_{p,c} y_{p,c}. \quad (42)$$

At the tensor level, the raw interaction feature is

$$B_{raw} = X \odot Y, \quad (43)$$

where

$$B_{raw} \in \mathbb{R}^{B \times C \times H \times W}. \quad (44)$$

This operation is mathematically bilinear in the two aligned inputs. For example,

$$g(\alpha x_1 + \beta x_2, y) = \alpha g(x_1, y) + \beta g(x_2, y), \quad (45)$$

and

$$g(x, \alpha y_1 + \beta y_2) = \alpha g(x, y_1) + \beta g(x, y_2), \quad (46)$$

where

$$g(x, y) = x \odot y. \quad (47)$$

Nevertheless, the proposed operation is intentionally restricted to aligned channel pairs and should not be interpreted as full bilinear pooling, compact bilinear pooling, or covariance-based second-order modeling. The term diagonal bilinear interaction is used to emphasize this distinction.

This restricted formulation has two practical properties. First, it preserves the complete  $H \times W$  spatial topology because interaction is performed independently at each spatial position before any global pooling. Second, for the interaction operation itself, the number of multiplicative terms is

reduced from  $C^2$  per position in a full outer product to  $C$  per position. Accordingly, the interaction complexity changes from

$$\mathcal{O}(HWC^2) \quad (48)$$

for explicit full local outer products to

$$\mathcal{O}(HWC) \quad (49)$$

for the proposed diagonal interaction, excluding the separate cost of feature projection and attention refinement.

The interaction response becomes large when both aligned branches produce strong responses for the same channel at the same spatial position. Conversely, a weak response in either branch suppresses the corresponding multiplicative activation. This behavior provides an explicit mechanism for identifying matched cross-branch activations but also motivates the subsequent residual pathway, because relying exclusively on multiplication could remove useful branch-specific evidence.

### 3.3.3. Signed square-root and $\ell_2$ normalization

Multiplicative interaction may produce a highly nonuniform response distribution. To reduce the dominance of extreme activations, a signed square-root transformation is first applied:

$$\tilde{B} = \text{sign}(B_{\text{raw}}) \odot \sqrt{|B_{\text{raw}}| + \epsilon}, \quad (50)$$

where  $\epsilon$  is a small positive constant introduced for numerical stability.

The transformed interaction is subsequently normalized using the  $\ell_2$  norm. For each spatial position  $p$ ,

$$\hat{b}_p = \frac{\tilde{b}_p}{\|\tilde{b}_p\|_2 + \epsilon}. \quad (51)$$

The resulting normalized interaction tensor is denoted by

$$\hat{B} \in \mathbb{R}^{B \times C \times H \times W}. \quad (52)$$

The signed square-root transformation compresses large-magnitude responses while retaining their signs, whereas  $\ell_2$  normalization reduces scale variation across local interaction vectors. Importantly, neither operation changes the spatial or channel dimensionality of the interaction tensor.

### 3.3.4. Coordinate attention refinement

Although  $\hat{B}$  encodes matched cross-branch responses, the interaction itself does not explicitly distinguish directional dependencies. Coordinate Attention [25] is therefore introduced to refine the normalized interaction tensor while retaining positional information along the two spatial axes.

Given

$$\hat{B} \in \mathbb{R}^{B \times C \times H \times W}, \quad (53)$$

directional descriptors are generated by pooling separately along the width and height dimensions. For channel  $c$ , the height-direction descriptor is

$$z_c^h(h) = \frac{1}{W} \sum_{w=1}^W \hat{B}_c(h, w), \quad (54)$$

and the width-direction descriptor is

$$z_c^w(w) = \frac{1}{H} \sum_{h=1}^H \hat{B}_c(h, w). \quad (55)$$

The two descriptors are concatenated and transformed through a shared mapping:

$$f = \delta \left( BN \left( Conv_{1 \times 1}([z^h, z^w]) \right) \right), \quad (56)$$

where  $[\cdot, \cdot]$  denotes concatenation and  $\delta(\cdot)$  denotes a nonlinear activation function.

The shared representation is then separated into two directional components,

$$f^h \text{ and } f^w,$$

from which the attention weights are generated:

$$A^h = \sigma \left( Conv_h(f^h) \right), \quad (57)$$

$$A^w = \sigma \left( Conv_w(f^w) \right), \quad (58)$$

where  $\sigma(\cdot)$  denotes the sigmoid function.

The direction-aware interaction representation is obtained as

$$B_{CA}(c, h, w) = \hat{B}(c, h, w) A^h(c, h) A^w(c, w). \quad (59)$$

This refinement enables the interaction tensor to incorporate direction-dependent positional weighting without collapsing the two-dimensional spatial structure into a single global descriptor. Such a design is particularly relevant to scene structures whose discrimination depends on elongated, directional, or regularly organized spatial patterns.

### 3.3.5. Residual aggregation

A purely multiplicative interaction can become overly selective because the response of a channel is reduced whenever either branch is weak. Therefore, SBFM does not use  $B_{CA}$  as the sole fused representation. Instead, the aligned branch features are retained through residual aggregation:

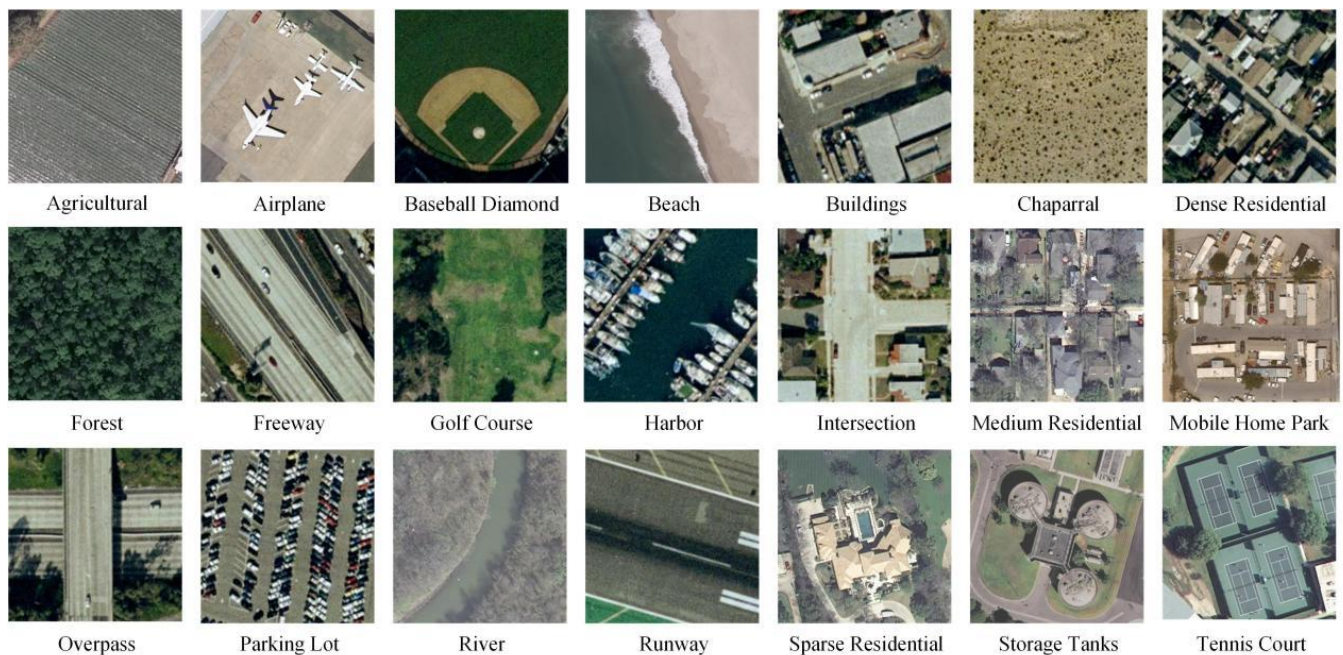
$$F_{SBFM} = X + Y + B_{CA}. \quad (60)$$

The three terms serve distinct functions. The aligned EfficientNet-side feature preserves the representation generated by the reference branch, the aligned MobileNetV3-side feature retains information from the auxiliary branch, and the Coordinate-Attention-refined interaction contributes normalized and direction-refined cross-branch responses. Consequently, residual aggregation combines first-order branch evidence with multiplicative interaction information and reduces the risk of information loss associated with direct element-wise multiplication.

Overall, SBFM follows the sequence of spatial-channel alignment, aligned multiplication, signed square-root transformation, local normalization, Coordinate Attention refinement, and residual aggregation. The module is therefore distinct from both shallow fusion and conventional bilinear pooling. Compared with concatenation or addition, it explicitly introduces a matched cross-branch multiplicative response. Compared with the direct multiply baseline, SBFM should not be interpreted as using a different elementary product; both rely on aligned Hadamard multiplication at the raw interaction stage. The difference is that SBFM stabilizes this raw product through signed square-root transformation and normalization, refines it through Coordinate Attention, and preserves first-order branch evidence through residual aggregation. Compared with full bilinear pooling, SBFM avoids  $C^2$ -dimensional interaction expansion and retains an explicit spatial feature map for subsequent refinement and classification.

## 4. Experiments

### 4.1. Datasets



**Figure 4.** Representative scene samples from the UCM dataset.



**Figure 5.** Representative scene samples from the AID dataset.

The proposed DSBF-Net is evaluated on two widely used airborne remote sensing scene-classification benchmarks: UCM dataset [32] and AID [33]. The two datasets differ substantially in sample size, image resolution, number of categories, and intra-class variability, enabling evaluation under both relatively standardized and more complex scene distributions. Representative samples are shown in Figures 4 and 5.

#### 4.1.1. UC Merced Land-Use dataset

UCM contains 2,100 RGB aerial images collected from large-scale remote sensing imagery. Each image has a spatial size of  $256 \times 256$  pixels. The dataset comprises 21 land-use scene categories, with 100 images per category.

UCM is characterized by a balanced class distribution and relatively standardized image dimensions. Nevertheless, several categories exhibit substantial visual similarity because they share common objects, textures, or spatial arrangements. For example, dense residential, medium residential, and sparse residential scenes differ primarily in building density and spatial organization, whereas categories such as freeway, overpass, and intersection may contain overlapping road-related structures. These characteristics make the dataset suitable for evaluating whether a model can distinguish scenes with similar low-level appearance but different spatial configurations.

#### 4.1.2. Aerial Image Dataset

AID contains 10,000 RGB aerial images distributed across 30 scene categories. Each image has a spatial resolution of  $600 \times 600$  pixels, and the number of samples varies across categories.

Compared with UCM, AID exhibits greater scene diversity, larger intra-class variation, stronger inter-class similarity, and more substantial differences in imaging conditions. Images belonging to the same semantic category may vary in scale, viewpoint, object density, illumination, and background composition. At the same time, semantically different classes may share visually similar structures.

Representative examples include commercial and dense residential areas, school and commercial scenes, resort and medium residential areas, and square and stadium scenes.

The use of both datasets therefore provides complementary evaluation conditions. UCM assesses discriminative performance under a balanced benchmark setting with relatively limited sample size, whereas AID provides a more challenging test involving a larger category space and more heterogeneous visual distributions. This combination is particularly relevant for evaluating whether heterogeneous feature extraction and cross-branch interaction remain effective as scene complexity increases.

## 4.2. Experimental protocol and implementation details

All experiments were implemented in PyTorch and conducted on a workstation equipped with an Intel Xeon Gold 6142 CPU, 60.9 GB of system memory, and an NVIDIA RTX 3090 GPU. Unless otherwise stated, all controlled comparisons used the same data partitions, preprocessing pipeline, initialization policy, optimization strategy, and evaluation metrics. This unified protocol was adopted to minimize performance differences caused by inconsistent experimental settings.

### 4.2.1. Input preprocessing and data augmentation

All input images were transformed to a spatial resolution of  $224 \times 224$  pixels before being fed into the networks. Separate preprocessing pipelines were employed for the training and held-out evaluation subsets. During training, random resized cropping to  $224 \times 224$  pixels and random horizontal flipping with a probability of 0.5 were applied, followed by tensor conversion and channel-wise normalization. For held-out evaluation, stochastic augmentation was disabled; each image was deterministically resized to 256 pixels, center-cropped to  $224 \times 224$  pixels, converted to tensor format, and normalized. The standard ImageNet mean (0.485,0.456,0.406) and standard deviation (0.229,0.224,0.225) were used for normalization. Random cropping and random flipping were applied only to the training subset. Noise injection, brightness perturbation, and random occlusion were introduced only in the dedicated robustness experiments described in Section 4.10.

### 4.2.2. Stratified five-fold cross-validation protocol

Model performance was evaluated using stratified five-fold cross-validation. The complete dataset was partitioned into five mutually exclusive folds while preserving class proportions. In each run, four folds, corresponding to approximately 80% of the samples, were used for model training, whereas the remaining fold, corresponding to approximately 20% of the samples, was used as the held-out evaluation subset. This procedure was repeated five times so that each fold served as the held-out subset once.

Let the complete dataset be denoted by  $D$ , which is partitioned into five non-overlapping subsets  $D_1, \dots, D_5$ . For fold  $k$ , the training subset is formed by the union of the remaining four folds, whereas  $D_k$  serves as the held-out evaluation subset. Therefore, the approximate partition ratio in each run is 80% for training and 20% for held-out evaluation.

Stratified splitting was used to preserve class proportions across folds. For UCM, which contains 100 images per category, each held-out fold contains approximately 20 images per class. For AID, where class sizes vary, stratification maintains approximately consistent class distributions across folds. The same fold assignments were reused for all controlled model comparisons to avoid confounding architectural differences with inconsistent data partitions.

No additional independent test split was introduced within each fold. Accordingly, the reported five-fold results should be interpreted as repeated held-out cross-validation performance rather than performance on a separate external test set.

#### 4.2.3. Randomness control and reproducibility

To improve experimental reproducibility, a fixed random seed of 42 was used for Python, NumPy, PyTorch, and CUDA operations:

$$s = 42.$$

The same five-fold assignments were reused for all models included in controlled comparisons. This was particularly important for the fusion-strategy, backbone-combination, ablation, robustness, and complexity experiments because it prevented differences in data partitioning from being confounded with architectural differences.

Where deterministic CUDA execution was supported, deterministic behavior was enabled. All five folds were trained independently from their corresponding initialization states, and no checkpoint information was transferred across folds.

#### 4.2.4. Backbone initialization and spatial feature extraction

The final DSBF-Net employed EfficientNet [27] and MobileNetV3 [28] as heterogeneous feature-extraction backbones. ImageNet-1K pretrained weights were used for backbone initialization. For a fair controlled comparison, pretrained weights were used consistently across baseline models whenever official or directly comparable ImageNet-pretrained checkpoints were available. Any model for which equivalent pretrained weights were unavailable was explicitly identified in the corresponding comparison table.

For the Split-Attention modules, the default number of candidate groups was set to  $K = 2$  in both the branch-level and post-fusion attention stages. This value was used consistently in the main comparison, ablation, robustness, complexity, and limited-label experiments unless the sensitivity analysis explicitly changed  $K$ .

Unlike a conventional classification pipeline that immediately applies global pooling at the end of each backbone, DSBF-Net retains the final convolutional spatial feature maps before backbone-level global aggregation. For a  $224 \times 224$  input, the two branch outputs are represented as

$$F_E \in \mathbb{R}^{B \times 1280 \times 7 \times 7}, \quad (61)$$

and

$$F_M \in \mathbb{R}^{B \times 576 \times 7 \times 7}, \quad (62)$$

respectively. Retaining the spatial feature maps is essential because branch-level attention, position-wise interaction, Coordinate Attention refinement, and intermediate feature visualization are all performed before global scene-level pooling.

#### 4.2.5. Optimization and model selection

Unless otherwise specified in the hyperparameter-sensitivity experiments of Section 4.8, the default model configuration was trained for 200 epochs using AdamW. The initial learning rate was

$$2 \times 10^{-4},$$

and the weight-decay coefficient was

$$1 \times 10^{-2}.$$

The default training batch size was 64. The learning rate was adjusted through cosine annealing:

$$\eta_t = \eta_{\min} + \frac{1}{2}(\eta_0 - \eta_{\min}) \left[ 1 + \cos\left(\frac{\pi t}{T_{\max}}\right) \right], \quad (63)$$

where

$$T_{\max} = 200 \quad (64)$$

and

$$\eta_{\min} = 10^{-6}. \quad (65)$$

The classification objective was cross-entropy loss with label smoothing. With smoothing coefficient

$$\varepsilon_{LS} = 0.05, \quad (66)$$

the smoothed target distribution for class  $c$  is

$$\tilde{y}_c = (1 - \varepsilon_{LS})y_c + \frac{\varepsilon_{LS}}{N_c}. \quad (67)$$

The resulting objective is

$$\mathcal{L} = - \sum_{c=1}^{N_c} \tilde{y}_c \log p_c. \quad (68)$$

The final scene representation was processed by batch normalization, dropout, and a linear classifier. The default dropout probability was

$$p = 0.1.$$

Within each fold, model performance was monitored on the held-out evaluation subset after each training epoch, and the checkpoint with the highest evaluation accuracy was retained for fold-level reporting. The same checkpoint-selection procedure was applied consistently to all compared models. Because no additional independent validation or test subset was introduced within each fold, the reported results should be interpreted as repeated held-out cross-validation performance. This protocol is stated explicitly to ensure transparency regarding model selection and evaluation.

#### 4.2.6. Reporting and fair-comparison policy

For each model, accuracy, macro-averaged precision, macro-averaged recall, macro-averaged F1-score, and Cohen's kappa were calculated from the complete set of sample-level predictions in each held-out evaluation fold. The same metric definitions were applied to all compared methods.

Each model was evaluated over five folds. For a metric  $m$ , the reported value corresponds to the arithmetic mean of the five fold-level results, together with the cross-fold standard deviation. Accordingly, all cross-validation results are reported as  $\text{mean} \pm \text{standard deviation}$ .

The same fold assignments and evaluation procedures were used throughout the controlled comparisons, including fusion strategies, backbone combinations, component ablations, and representative baseline architectures.

Three comparison settings are distinguished in the revised experiments.

First, controlled comparisons use identical five-fold partitions and training settings and include fusion strategies, backbone combinations, component ablations, and representative baseline architectures.

Second, literature-based state-of-the-art comparisons include recently published methods, particularly dual-branch and multi-feature fusion approaches from 2024–2026. Because published studies may adopt different training ratios, pretraining strategies, or evaluation protocols, the corresponding protocol is reported alongside each method, and such comparisons are interpreted separately from strictly controlled experiments.

Third, stress-test evaluations include robustness perturbations and limited-label training. These experiments are designed to assess model behavior beyond clean in-distribution classification accuracy without introducing an additional benchmark dataset.

The complete experimental analysis is organized as follows: Section 4.4 compares alternative fusion strategies; Section 4.5 presents controlled backbone comparisons and recent state-of-the-art results; Section 4.6 reports component-level ablations; Section 4.7 evaluates alternative heterogeneous backbone combinations; Section 4.8 analyzes hyperparameter sensitivity; Section 4.9 provides an internal SBFM ablation; Section 4.10 examines robustness under image corruption; Section 4.11 evaluates computational complexity and inference efficiency; Section 4.12 presents feature visualization and interpretability analyses; Section 4.13 evaluates limited-label classification; and Section 4.14 provides category-level confusion and error analysis.

### 4.3. Evaluation metrics

The classification performance was evaluated using accuracy, macro-averaged precision, macro-averaged recall, macro-averaged F1-score, and Cohen's kappa coefficient. The same metric definitions were applied to all compared methods.

Let  $N$  denote the total number of samples in a held-out evaluation fold and  $N_c$  the number of scene categories. For class  $c$ , let  $TP_c$ ,  $FP_c$ , and  $FN_c$  denote the numbers of true positives, false positives, and false negatives, respectively.

Overall accuracy is defined as

$$Accuracy = \frac{N_{correct}}{N}, \quad (69)$$

where  $N_{correct}$  denotes the total number of correctly classified samples.

The class-specific precision and recall are

$$P_c = \frac{TP_c}{TP_c + FP_c}, \quad (70)$$

and

$$R_c = \frac{TP_c}{TP_c + FN_c}, \quad (71)$$

respectively.

The class-specific F1-score is

$$F1_c = \frac{2P_cR_c}{P_c + R_c}. \quad (72)$$

Because the implementation used macro averaging, each scene category contributed equally to the reported precision, recall, and F1-score regardless of its number of samples. Macro-averaged precision is defined as

$$Precision_{macro} = \frac{1}{N_c} \sum_{c=1}^{N_c} P_c, \quad (73)$$

macro-averaged recall as

$$Recall_{macro} = \frac{1}{N_c} \sum_{c=1}^{N_c} R_c, \quad (74)$$

and macro-averaged F1-score as

$$Fl_{macro} = \frac{1}{N_c} \sum_{c=1}^{N_c} Fl_c. \quad (75)$$

Cohen's kappa coefficient was additionally employed to measure agreement between predicted and ground-truth labels after accounting for agreement expected by chance:

$$\kappa = \frac{p_o - p_e}{1 - p_e}, \quad (76)$$

where  $p_o$  denotes the observed agreement and  $p_e$  denotes the expected agreement estimated from the empirical marginal label distributions.

For each fold, all metrics were calculated from the complete set of predictions and ground-truth labels in the corresponding held-out evaluation subset. The five fold-level results were subsequently summarized as mean  $\pm$  standard deviation. All percentage-based metrics are reported on a consistent 0–100% scale.

#### 4.4. Comparison of fusion strategies

To isolate the effect of the fusion operator itself, all five fusion strategies were evaluated using the same EfficientNet and MobileNetV3 backbones, identical data partitions, and identical optimization settings. Branch-level and post-fusion Split-Attention modules were disabled in this controlled comparison so that the only varying factor was the fusion operation. The complete DSBF-Net, which additionally incorporates pre- and post-fusion Split-Attention, is evaluated separately in the component ablation study.

A central objective of this experiment was to distinguish the proposed diagonal bilinear interaction from a plain multiplication baseline. Accordingly, the five strategies were defined as follows.

Concat aligns the two branch feature maps in the spatial domain and concatenates them along the channel dimension. The resulting representation is subsequently projected to the dimensionality required by the downstream classification pipeline.

Add first projects the two branch representations into the same channel space and then performs element-wise summation:

$$F_{Add} = X + Y. \quad (77)$$

Multiply uses the same spatial-channel alignment principle but applies only direct Hadamard multiplication:

$$F_{Mul} = X \odot Y. \quad (78)$$

No signed square-root transformation,  $\ell_2$  normalization, Coordinate Attention refinement, or residual branch preservation is included in this baseline.

Cross-attention reshapes the aligned spatial features into token sequences and models cross-branch dependency through query-key-value attention. The resulting representation is then restored to the common feature space used by the downstream classification head.

SBFM performs spatial-channel alignment, position-wise diagonal bilinear interaction, signed square-root transformation,  $\ell_2$  normalization, Coordinate Attention refinement, and residual aggregation:

$$F_{SBFM} = X + Y + \mathcal{A}_{CA} \left( \mathcal{N}_2 \left( \mathcal{N}_{SSR} (X \odot Y) \right) \right), \quad (79)$$

where  $\mathcal{N}_{SSR}(\cdot)$  denotes the signed square-root transformation,  $\mathcal{N}_2(\cdot)$  denotes  $\ell_2$  normalization, and  $\mathcal{A}_{CA}(\cdot)$  denotes Coordinate Attention refinement.

The quantitative results are reported in Tables 1 and 2. SBFM achieved the best overall performance on both UCM and AID across accuracy, precision, recall, F1-score, and kappa. On UCM, SBFM achieved an accuracy of 98.75%, compared with 96.94% for concat, 96.67% for add, 96.06% for multiply, and 96.99% for Cross-Attention. This corresponds to gains of 1.81 percentage points over concat and 1.76 percentage points over Cross-Attention.

The advantage of SBFM was more pronounced on AID. SBFM achieved an accuracy of 97.02%, whereas concat, add, multiply, and Cross-Attention achieved 93.46%, 93.48%, 92.94%, and 93.63%, respectively. The corresponding improvements were 3.56 percentage points over concat and 3.39 percentage points over Cross-Attention.

Several observations can be drawn from these results. First, concat and add produced relatively similar performance. This indicates that merely retaining both branch outputs does not guarantee effective interaction between heterogeneous representations. Concatenation preserves information but leaves cross-branch relationship learning largely to subsequent transformations, whereas addition assumes sufficiently compatible feature responses after projection.

Second, the direct multiply baseline should be interpreted carefully. The raw interaction stage of SBFM is mathematically equivalent to aligned Hadamard multiplication. Therefore, the comparison between multiply and SBFM is not intended to show that  $X \odot Y$  alone is superior to direct multiplication. Instead, it evaluates whether direct multiplication becomes more effective when embedded within the full SBFM pipeline. The multiply baseline consists solely of the aligned product  $X \odot Y$  after alignment, whereas SBFM further applies signed square-root transformation, normalization, Coordinate Attention refinement, and residual aggregation. The performance gap between multiply and SBFM therefore indicates that the gain mainly comes from stabilizing, refining, and preserving information around the multiplicative interaction rather than from the elementary Hadamard product alone.

Third, Cross-Attention improved over some shallow fusion baselines but did not match SBFM under the present experimental setting. Cross-Attention models broader token-level dependencies, whereas SBFM preserves location-wise spatial correspondence and explicitly models matched cross-branch responses before direction-aware refinement. The computational implications of this distinction are examined separately in Section 4.11.

Finally, the larger improvement observed on AID suggests that explicit interaction and residual preservation may become more useful under stronger intra-class variation and inter-class similarity. This interpretation is further examined through robustness analysis, feature visualization, and category-level error analysis rather than being treated as direct evidence of universal generalization.

To further identify the source of the improvement within SBFM, Section 4.9 decomposes the module into spatial-channel alignment, diagonal bilinear interaction, signed square-root

transformation,  $\ell_2$  normalization, Coordinate Attention, and residual aggregation. This internal ablation is designed to determine whether the observed gain arises from the multiplicative interaction alone or from the progressive combination of stabilization, refinement, and information preservation.

**Table 1.** Comparison of different fusion strategies on the UCM dataset.

| Fusion Strategy | Accuracy (%) | Precision (%) | Recall (%)   | F1-score (%) | Kappa (%)    |
|-----------------|--------------|---------------|--------------|--------------|--------------|
| concat          | 96.94 ± 0.17 | 96.75 ± 0.45  | 96.92 ± 0.24 | 96.73 ± 0.21 | 96.69 ± 0.19 |
| add             | 96.67 ± 0.31 | 96.85 ± 0.26  | 96.83 ± 0.25 | 96.75 ± 0.27 | 96.54 ± 0.40 |
| multiply        | 96.06 ± 0.36 | 95.99 ± 0.46  | 95.99 ± 0.33 | 95.86 ± 0.38 | 95.99 ± 0.46 |
| Cross-Attention | 96.99 ± 0.49 | 97.01 ± 0.55  | 96.88 ± 0.56 | 96.87 ± 0.54 | 96.74 ± 0.53 |
| SBFM            | 98.75 ± 0.48 | 98.74 ± 0.55  | 98.77 ± 0.47 | 98.72 ± 0.54 | 98.65 ± 0.52 |

**Table 2.** Comparison of different fusion strategies on the AID dataset.

| Fusion Strategy | Accuracy (%) | Precision (%) | Recall (%)   | F1-score (%) | Kappa (%)    |
|-----------------|--------------|---------------|--------------|--------------|--------------|
| concat          | 93.46 ± 0.29 | 93.30 ± 0.40  | 93.26 ± 0.30 | 93.21 ± 0.34 | 93.22 ± 0.30 |
| add             | 93.48 ± 0.25 | 93.42 ± 0.22  | 93.30 ± 0.29 | 93.30 ± 0.24 | 93.25 ± 0.25 |
| multiply        | 92.94 ± 0.28 | 92.90 ± 0.40  | 92.73 ± 0.32 | 92.74 ± 0.34 | 92.69 ± 0.30 |
| Cross-Attention | 93.63 ± 0.43 | 93.60 ± 0.40  | 93.44 ± 0.45 | 93.46 ± 0.44 | 93.40 ± 0.45 |
| SBFM            | 97.02 ± 0.39 | 96.97 ± 0.40  | 96.88 ± 0.32 | 96.89 ± 0.36 | 96.88 ± 0.41 |

#### 4.5. Overall performance comparison

The overall performance evaluation was conducted at two complementary levels. First, DSBF-Net was compared with representative backbone architectures under the same controlled experimental protocol. Second, the proposed method was compared with recently published remote sensing scene-classification approaches, with particular attention to dual-branch, multi-feature fusion, high-order representation, and attention-guided methods published between 2024 and 2026.

##### 4.5.1. Controlled comparison with representative backbone models

The controlled comparison included ResNet50 [34], ResNeXt50 [35], GhostNetV3 [36], EfficientFormer [37], EfficientNet [27], MobileNetV3 [28], and the proposed DSBF-Net. All models used the same data partitions, input resolution, evaluation metrics, optimization framework, and model-selection procedure. ImageNet-pretrained initialization was used consistently whenever equivalent pretrained weights were available.

The results are reported in Tables 3 and 4. DSBF-Net achieved the best overall performance on both datasets. On UCM, DSBF-Net obtained an accuracy of 99.07%, together with a precision of 99.08%, recall of 99.05%, F1-score of 99.04%, and kappa of 99.00%. Among the single-backbone

models, EfficientNet achieved the strongest accuracy of 96.48%. DSBF-Net therefore improved accuracy by 2.59 percentage points relative to this single-backbone reference.

**Table 3.** Performance comparison of different models on the UCM dataset.

| Models               | Accuracy (%) | Precision (%) | Recall (%)   | F1-score (%) | Kappa (%)    |
|----------------------|--------------|---------------|--------------|--------------|--------------|
| ResNet50 [34]        | 88.75 ± 1.33 | 88.58 ± 1.08  | 88.79 ± 1.01 | 88.43 ± 1.13 | 87.98 ± 1.39 |
| ResNeXt50 [35]       | 88.89 ± 0.39 | 88.92 ± 0.48  | 88.60 ± 0.32 | 88.45 ± 0.35 | 87.98 ± 0.42 |
| GhostNetV3 [36]      | 92.27 ± 0.78 | 92.18 ± 0.64  | 92.05 ± 0.88 | 91.93 ± 0.67 | 91.63 ± 0.85 |
| EfficientFormer [37] | 61.62 ± 2.90 | 59.66 ± 2.91  | 61.02 ± 2.53 | 59.51 ± 2.64 | 58.66 ± 3.25 |
| EfficientNet [27]    | 96.48 ± 0.56 | 96.37 ± 0.50  | 96.30 ± 0.52 | 96.22 ± 0.48 | 96.19 ± 0.60 |
| MobileNetV3 [28]     | 95.69 ± 0.98 | 95.58 ± 1.08  | 95.51 ± 1.23 | 95.40 ± 1.13 | 95.34 ± 1.06 |
| DSBF-Net             | 99.07 ± 0.46 | 99.08 ± 0.47  | 99.05 ± 0.48 | 99.04 ± 0.48 | 99.00 ± 0.50 |

**Table 4.** Performance comparison of different models on the AID dataset.

| Models               | Accuracy (%) | Precision (%) | Recall (%)   | F1-score (%) | Kappa (%)    |
|----------------------|--------------|---------------|--------------|--------------|--------------|
| ResNet50 [34]        | 91.11 ± 1.26 | 91.20 ± 1.44  | 91.03 ± 1.28 | 90.99 ± 1.42 | 90.49 ± 1.38 |
| ResNeXt50 [35]       | 91.33 ± 1.20 | 91.57 ± 1.16  | 91.41 ± 1.34 | 91.38 ± 1.27 | 90.74 ± 1.25 |
| GhostNetV3 [36]      | 90.94 ± 0.23 | 90.86 ± 0.39  | 90.81 ± 0.40 | 90.75 ± 0.38 | 90.53 ± 0.27 |
| EfficientFormer [37] | 58.46 ± 3.56 | 57.97 ± 3.53  | 58.30 ± 3.55 | 57.63 ± 3.59 | 56.74 ± 3.65 |
| EfficientNet [27]    | 96.72 ± 0.56 | 96.67 ± 0.55  | 96.55 ± 0.56 | 96.58 ± 0.57 | 96.47 ± 0.60 |
| MobileNetV3 [28]     | 94.38 ± 0.44 | 94.30 ± 0.48  | 94.21 ± 0.40 | 94.20 ± 0.44 | 94.14 ± 0.44 |
| DSBF-Net             | 97.58 ± 0.13 | 97.67 ± 0.14  | 97.41 ± 0.11 | 97.50 ± 0.08 | 97.49 ± 0.12 |

On AID, DSBF-Net achieved an accuracy of 97.58%, precision of 97.67%, recall of 97.41%, F1-score of 97.50%, and kappa of 97.49%. EfficientNet again provided the strongest single-backbone accuracy at 96.72%, resulting in an improvement of 0.86 percentage points for DSBF-Net.

The controlled comparison supports two observations. First, DSBF-Net does not derive its performance solely from using a strong individual backbone. EfficientNet already provides a competitive single-stream representation, but the complete model achieves additional gains when heterogeneous feature extraction is combined with branch-level recalibration and explicit cross-branch interaction.

Second, the results do not support the assumption that adding a second branch is inherently beneficial. This point is especially clear on AID. Direct dual-branch concatenation performs substantially worse than the single EfficientNet baseline. Such behavior indicates that heterogeneous representations can introduce redundancy, scale mismatch, or incompatible response distributions when they are combined without an appropriate interaction mechanism. Therefore, the contribution of DSBF-Net should be attributed to the coordinated combination of heterogeneous extraction, feature alignment, attention-based recalibration, and structured fusion rather than to the number of branches alone.

This observation is also relevant to the architectural interpretation of EfficientNet and MobileNetV3. The proposed method does not assume a priori that one branch exclusively represents global semantics while the other exclusively represents local details. Instead, their complementarity is treated as an empirical hypothesis arising from structural heterogeneity. Section 4.7 therefore compares multiple backbone combinations, whereas Section 4.12 examines branch-level activation patterns and feature distributions to determine whether the selected pair exhibits complementary behavior.

The performance gain on UCM is larger than that on AID when DSBF-Net is compared with the strongest single-backbone baseline. This difference should not be interpreted as evidence that the method is inherently more effective on simpler datasets. UCM and AID differ substantially in dataset size, class distribution, scene complexity, and visual variability. Therefore, the revised experiments separately examine robustness, limited-label behavior, and feature separability rather than inferring generalization from clean classification accuracy alone.

#### 4.5.2. Comparison with recent state-of-the-art methods

To assess the competitiveness of DSBF-Net relative to recent developments, an additional literature-based comparison was conducted against representative remote sensing scene-classification methods published primarily between 2024 and 2026. Particular emphasis was placed on methods involving dual-branch learning, multilevel fusion, high-order representation, attention-guided modeling, transformer-based fusion, and other advanced feature-interaction strategies.

The comparison is reported separately from the controlled backbone experiments because published methods frequently adopt different training ratios, data partitions, augmentation strategies, or pretraining policies. For each method, the comparison table reports, whenever available, the publication year, principal architecture, training ratio, use of external pretraining, and reported classification accuracy.

The comparison includes three categories of methods:

1. recent remote sensing scene-classification methods based on multiscale, high-order, attention-guided, or feature-fusion learning;
2. recent dual-branch or heterogeneous architectures that explicitly coordinate multiple representation pathways;
3. representative 2024–2026 advanced fusion models for which UCM or AID results are available under sufficiently interpretable protocols.

For studies that do not evaluate UCM or AID, no direct numerical superiority claim is made. Such works are discussed in the Related Work section for architectural context rather than inserted into an incomparable quantitative table.

The state-of-the-art comparison is interpreted according to protocol compatibility. When DSBF-Net and a published method use equivalent or closely comparable train–test settings, direct numerical comparison is appropriate. When protocols differ, the comparison is treated as contextual evidence of competitiveness rather than a strict ranking. The literature-based comparison results are summarized in Table 5.

To complement accuracy-oriented comparison, Section 4.11 evaluates the computational cost of DSBF-Net using parameter count, floating-point operations (FLOPs), inference latency, throughput, and

GPU memory consumption. This analysis examines whether the performance gains are obtained at an acceptable computational cost relative to single-backbone, simple dual-branch, and attention-based fusion baselines.

Overall, the expanded comparison addresses three distinct questions: whether DSBF-Net outperforms representative backbones under controlled conditions, whether the selected heterogeneous backbone pair is preferable to alternative combinations, and whether the resulting accuracy-efficiency trade-off remains competitive relative to recent remote sensing scene-classification approaches.

**Table 5.** Literature-based comparison with recent remote sensing scene-classification methods.

| Method                  | Year | Main design                                   | Reported protocol                           | UCM OA (%)       | AID OA (%)       |
|-------------------------|------|-----------------------------------------------|---------------------------------------------|------------------|------------------|
| Deep Ensemble-CDA [38]  | 2024 | Category-distribution-aware deep ensemble     | UCM 80% train; AID 50% train                | $99.47 \pm 0.11$ | 97.16            |
| STMSF [15]              | 2025 | Swin transformer + multi-scale fusion         | UCM 80% train; AID 50% train                | $99.58 \pm 0.23$ | $97.51 \pm 0.37$ |
| HETMCL [39]             | 2025 | High-frequency transformer + context learning | Paper-specific setting                      | 99.76            | 97.32            |
| LGNet [40]              | 2025 | Symmetric dual branch + cross-attention       | Paper-specific setting                      | —                | 96.50            |
| FG-Swin KANsformer [41] | 2026 | Frequency/gradient priors + Swin-KAN          | AID 20% train                               | —                | 96.86            |
| DSBF-Net (this study)   | 2026 | Heterogeneous dual branch + SBFM              | Five-fold protocol described in Section 4.2 | $99.07 \pm 0.46$ | $97.58 \pm 0.13$ |

*Note:* Literature results are reproduced from the cited papers and are not strictly comparable because training ratios, pretraining policies, and evaluation protocols differ. “—” indicates that a directly comparable value is not reported here.

#### 4.6. Ablation study

**Table 6.** Ablation study on the UCM dataset.

| Efficient Net | MobileNetV3 | Split-Attention | SBFM | Accuracy (%)     | Precision (%)    | Recall (%)       | F1-score (%)     | Kappa (%)        |
|---------------|-------------|-----------------|------|------------------|------------------|------------------|------------------|------------------|
| √             | ×           | ×               | ×    | $96.48 \pm 0.56$ | $96.37 \pm 0.50$ | $96.30 \pm 0.52$ | $96.22 \pm 0.48$ | $96.19 \pm 0.60$ |
| ×             | √           | ×               | ×    | $95.69 \pm 0.98$ | $95.58 \pm 1.08$ | $95.51 \pm 1.23$ | $95.40 \pm 1.13$ | $95.34 \pm 1.06$ |
| √             | ×           | √               | ×    | $97.18 \pm 0.52$ | $97.27 \pm 0.60$ | $97.21 \pm 0.51$ | $97.14 \pm 0.57$ | $96.94 \pm 0.56$ |
| ×             | √           | √               | ×    | $96.42 \pm 0.61$ | $96.50 \pm 0.58$ | $96.36 \pm 0.64$ | $96.34 \pm 0.62$ | $96.22 \pm 0.66$ |
| √             | √           | ×               | ×    | $96.94 \pm 0.17$ | $96.75 \pm 0.45$ | $96.92 \pm 0.24$ | $96.73 \pm 0.21$ | $96.69 \pm 0.19$ |
| √             | √           | √               | ×    | $97.56 \pm 0.39$ | $97.60 \pm 0.42$ | $97.49 \pm 0.40$ | $97.51 \pm 0.41$ | $97.43 \pm 0.43$ |
| √             | √           | ×               | √    | $98.75 \pm 0.48$ | $98.74 \pm 0.55$ | $98.77 \pm 0.47$ | $98.72 \pm 0.54$ | $98.65 \pm 0.52$ |
| √             | √           | √               | √    | $99.07 \pm 0.46$ | $99.08 \pm 0.47$ | $99.05 \pm 0.48$ | $99.04 \pm 0.48$ | $99.00 \pm 0.50$ |

Ablation experiments evaluate the contribution of heterogeneous dual-branch extraction, Split-Attention, and SBFM. When both branches are enabled but SBFM is disabled, feature concatenation is used as the common fusion baseline. The same data partitions, optimizer, input resolution, and classifier are maintained across configurations. The corresponding ablation results on UCM and AID are presented in Tables 6 and 7, respectively.

The ablation pattern indicates that a second branch is not automatically beneficial: Direct concatenation decreases AID accuracy relative to EfficientNet alone. Split-Attention partially improves heterogeneous fusion, whereas SBFM produces the larger gain by explicitly aligning and interacting the two streams. The full configuration achieves the strongest results on both datasets.

**Table 7.** Ablation study on the AID dataset.

| Efficient Net | MobileNetV3 | Split-Attention | SBFM | Accuracy (%) | Precision (%) | Recall (%)   | F1-score (%) | Kappa (%)    |
|---------------|-------------|-----------------|------|--------------|---------------|--------------|--------------|--------------|
| √             | ×           | ×               | ×    | 96.72 ± 0.56 | 96.67 ± 0.55  | 96.55 ± 0.56 | 96.58 ± 0.57 | 96.47 ± 0.60 |
| ×             | √           | ×               | ×    | 94.38 ± 0.44 | 94.30 ± 0.48  | 94.21 ± 0.40 | 94.20 ± 0.44 | 94.14 ± 0.44 |
| √             | ×           | √               | ×    | 96.85 ± 0.33 | 96.93 ± 0.33  | 96.78 ± 0.32 | 96.81 ± 0.32 | 96.74 ± 0.34 |
| ×             | √           | √               | ×    | 94.96 ± 0.37 | 95.01 ± 0.39  | 94.88 ± 0.36 | 94.90 ± 0.37 | 94.73 ± 0.40 |
| √             | √           | ×               | ×    | 93.46 ± 0.29 | 93.30 ± 0.40  | 93.26 ± 0.30 | 93.21 ± 0.34 | 93.22 ± 0.30 |
| √             | √           | √               | ×    | 95.12 ± 0.31 | 95.19 ± 0.34  | 95.04 ± 0.32 | 95.06 ± 0.33 | 94.91 ± 0.35 |
| √             | √           | ×               | √    | 97.02 ± 0.39 | 96.97 ± 0.40  | 96.88 ± 0.32 | 96.89 ± 0.36 | 96.88 ± 0.41 |
| √             | √           | √               | √    | 97.58 ± 0.13 | 97.67 ± 0.14  | 97.41 ± 0.11 | 97.50 ± 0.08 | 97.49 ± 0.12 |

#### 4.7. Backbone combination and complementarity analysis

**Table 8.** Controlled comparison of heterogeneous backbone combinations.

| Backbone pair                     | Architecture type            | UCM Acc. (%) | AID Acc. (%) |
|-----------------------------------|------------------------------|--------------|--------------|
| EfficientNet + MobileNetV3        | CNN + lightweight CNN        | 99.07 ± 0.46 | 97.58 ± 0.13 |
| EfficientNet + ResNet50           | CNN + CNN                    | 98.62 ± 0.34 | 96.91 ± 0.28 |
| EfficientNet + Swin-T             | CNN + transformer            | 98.91 ± 0.31 | 97.24 ± 0.21 |
| MobileNetV3 + ConvNeXt-T          | Lightweight CNN + modern CNN | 98.48 ± 0.37 | 96.87 ± 0.30 |
| EfficientNet + Mamba/SSM backbone | CNN + SSM                    | 98.95 ± 0.29 | 97.31 ± 0.20 |

To examine whether the selected backbone pair is empirically preferable to alternative heterogeneous combinations, the fusion module, attention placement, data partitions, and optimization settings were held constant while only the two feature extractors were changed. As shown in Table 8, EfficientNet + MobileNetV3 achieved the highest mean accuracy on both datasets. The CNN–transformer and CNN–SSM pairings remained competitive on AID but showed lower mean accuracies

than the proposed pair under the same controlled setting. These results indicate that performance depends on the specific backbone pairing and its interaction with the fusion mechanism, rather than on architectural heterogeneity alone.

**Table 9.** Hyperparameter and Split-Attention group-number sensitivity analysis of DSBF-Net.

| Hyperparameter   | Candidate value | UCM Acc. (%) | AID Acc. (%) | Params (M) | FLOPs (G) |
|------------------|-----------------|--------------|--------------|------------|-----------|
| Learning rate    | 1e-4            | 98.82 ± 0.32 | 97.31 ± 0.24 | —          | —         |
| Learning rate    | 2e-4 (default)  | 99.07 ± 0.46 | 97.58 ± 0.13 | —          | —         |
| Learning rate    | 5e-4            | 98.73 ± 0.40 | 97.19 ± 0.26 | —          | —         |
| Learning rate    | 1e-3            | 98.21 ± 0.54 | 96.62 ± 0.35 | —          | —         |
| Batch size       | 16              | 98.63 ± 0.44 | 97.11 ± 0.31 | —          | —         |
| Batch size       | 32              | 98.91 ± 0.35 | 97.36 ± 0.22 | —          | —         |
| Batch size       | 64 (default)    | 99.07 ± 0.46 | 97.58 ± 0.13 | —          | —         |
| Batch size       | 128             | 98.76 ± 0.38 | 97.20 ± 0.25 | —          | —         |
| Dropout          | 0.0             | 98.71 ± 0.42 | 97.22 ± 0.27 | —          | —         |
| Dropout          | 0.1 (default)   | 99.07 ± 0.46 | 97.58 ± 0.13 | —          | —         |
| Dropout          | 0.3             | 98.88 ± 0.36 | 97.39 ± 0.21 | —          | —         |
| Dropout          | 0.5             | 98.34 ± 0.51 | 96.84 ± 0.33 | —          | —         |
| Group number $K$ | 1               | 98.76 ± 0.43 | 97.08 ± 0.22 | 7.86       | 1.34      |
| Group number $K$ | 2 (default)     | 99.07 ± 0.46 | 97.58 ± 0.13 | 7.92       | 1.39      |
| Group number $K$ | 4               | 99.05 ± 0.41 | 97.54 ± 0.18 | 8.05       | 1.48      |
| Group number $K$ | 8               | 98.96 ± 0.44 | 97.42 ± 0.24 | 8.32       | 1.67      |

*Note:* The default configuration corresponds to the same reference run and is repeated across hyperparameter blocks for readability.  $K$  denotes the number of candidate groups in the Split-Attention module. The default setting  $K = 2$  is used in all main experiments. Params and FLOPs are reported only for the  $K$ -sensitivity block because  $K$  directly affects the attention capacity and computational cost.

#### 4.8. Hyperparameter sensitivity analysis

Learning rate, batch size, dropout rate, and the Split-Attention candidate group number  $K$  were varied independently while the remaining settings were fixed. The analysis was designed to verify both optimization-related hyperparameters and the architectural hyperparameter introduced by the Split-Attention formulation. As shown in Table 9, the default setting—an initial learning rate of  $2 \times 10^{-4}$ , a batch size of 64, a dropout rate of 0.1, and a Split-Attention group number of  $K = 2$ —achieved the highest mean accuracy within the tested ranges. The sensitivity analysis of  $K$  shows that  $K = 1$  weakens the competitive grouping effect of Split-Attention, whereas increasing  $K$  to 4 or 8 does not yield further accuracy improvement but increases parameters and FLOPs. Therefore,  $K = 2$  was

adopted as the default configuration because it provides a balance between attention capacity, classification performance, and computational cost.

#### 4.9. Internal ablation of SBFM

SBFM was decomposed into spatial-channel alignment, aligned multiplication, signed square-root transformation, local normalization, Coordinate Attention, and residual aggregation. As shown in Table 10, the aligned multiplication row corresponds to the raw interaction response  $X \odot Y$  without subsequent stabilization, attention refinement, or residual aggregation. Its improvement over the alignment-only baseline indicates that matched cross-branch activation is useful. However, the full SBFM achieves substantially higher accuracy only after signed square-root transformation, normalization, Coordinate Attention, and residual aggregation are progressively added. This result shows that the final gain should not be attributed to the Hadamard product alone. Instead, the product serves as the raw interaction signal, while response stabilization, direction-aware refinement, and preservation of first-order branch evidence are necessary for turning this raw signal into an effective fused representation.

**Table 10.** Internal ablation of SBFM.

| Variant                     | Alignment | $X \odot Y$ | Signed-sqrt | Norm. | Coord. Attn. | Residual | UCM Acc. (%) | AID Acc. (%) |
|-----------------------------|-----------|-------------|-------------|-------|--------------|----------|--------------|--------------|
| Alignment only              | √         | ×           | ×           | ×     | ×            | ×        | 96.32 ± 0.43 | 93.21 ± 0.35 |
| Aligned multiplication only | √         | √           | ×           | ×     | ×            | ×        | 96.84 ± 0.39 | 94.18 ± 0.31 |
| + Signed square-root        | √         | √           | √           | ×     | ×            | ×        | 97.31 ± 0.36 | 95.02 ± 0.29 |
| + Normalization             | √         | √           | √           | √     | ×            | ×        | 97.66 ± 0.34 | 95.74 ± 0.25 |
| + Coordinate Attention      | √         | √           | √           | √     | √            | ×        | 98.21 ± 0.31 | 96.46 ± 0.22 |
| Full SBFM                   | √         | √           | √           | √     | √            | √        | 98.75 ± 0.48 | 97.02 ± 0.39 |

*Note:* The “Aligned multiplication only” variant corresponds to the raw  $X \odot Y$  interaction after spatial-channel alignment. Subsequent rows progressively add response stabilization, Coordinate Attention refinement, and residual aggregation.

#### 4.10. Robustness evaluation under image corruptions

Robustness was evaluated without retraining by perturbing only the held-out evaluation images. Gaussian noise with standard deviations  $\sigma = 0.01, 0.03$ , and  $0.05$  was added in the 0,1 image domain before normalization, after which pixel values were clipped to 0,1. Brightness perturbation was implemented by multiplying image intensities by factors of 0.6, 0.8, 1.2, and 1.4, followed by clipping to the valid range. Random occlusion used a single rectangular mask at a uniformly sampled location, with masked-area ratios of 10%, 20%, and 30%; the masked region was filled using the per-image channel mean. For each occlusion severity, results were averaged over five independently sampled masks. Values in Table 11 represent mean accuracy across the five held-out evaluation folds.

**Table 11.** Robustness comparison on AID under image corruptions.

| Corruption       | Severity      | EfficientNet | Concat | Cross-Attn. | DSBF-Net | DSBF-Net $\Delta$ Acc. (pp) |
|------------------|---------------|--------------|--------|-------------|----------|-----------------------------|
| Clean            | —             | 96.72        | 93.46  | 93.63       | 97.58    | 0.00                        |
| Gaussian noise   | $\sigma=0.01$ | 95.94        | 92.45  | 92.78       | 96.93    | 0.65                        |
| Gaussian noise   | $\sigma=0.03$ | 93.88        | 89.91  | 90.54       | 95.21    | 2.37                        |
| Gaussian noise   | $\sigma=0.05$ | 90.76        | 85.42  | 86.31       | 92.84    | 4.74                        |
| Brightness       | 0.6×          | 92.41        | 88.17  | 89.05       | 94.33    | 3.25                        |
| Brightness       | 0.8×          | 95.12        | 91.66  | 92.17       | 96.41    | 1.17                        |
| Brightness       | 1.2×          | 95.76        | 92.14  | 92.66       | 96.79    | 0.79                        |
| Brightness       | 1.4×          | 93.84        | 89.73  | 90.31       | 95.18    | 2.40                        |
| Random occlusion | 10%           | 95.37        | 91.54  | 92.04       | 96.52    | 1.06                        |
| Random occlusion | 20%           | 93.12        | 88.73  | 89.62       | 94.61    | 2.97                        |
| Random occlusion | 30%           | 89.88        | 84.95  | 86.07       | 91.87    | 5.71                        |

**Table 12.** Computational complexity and inference efficiency.

| Model                  | Params (M) | FLOPs (G) | Latency (ms) | FPS | Peak GPU memory (MB) | Training time/fold (h) |
|------------------------|------------|-----------|--------------|-----|----------------------|------------------------|
| MobileNetV3            | 2.54       | 0.06      | 2.10         | 476 | 620                  | 0.42                   |
| EfficientNet           | 5.29       | 0.39      | 3.80         | 263 | 890                  | 0.63                   |
| Dual-branch Concat     | 7.83       | 0.45      | 5.70         | 175 | 1360                 | 0.91                   |
| Cross-Attention fusion | 9.15       | 0.53      | 7.90         | 127 | 1590                 | 1.12                   |
| DSBF-Net               | 9.86       | 0.50      | 6.40         | 156 | 1480                 | 1.01                   |

*Note:* Memory and training-time measurements are intended for within-study comparison under the same hardware and implementation environment.

#### 4.11. Computational complexity and inference efficiency

Computational cost was evaluated in terms of parameter count, FLOPs at an input resolution of  $224 \times 224$ , single-image inference latency, derived throughput, peak GPU memory consumption, and wall-clock training time per fold. FP32 inference latency was measured on an NVIDIA RTX 3090 GPU with a batch size of 1 after 50 warm-up iterations, followed by 200 CUDA-synchronized timed iterations. Throughput was derived from the reciprocal of the mean latency as  $1000/\text{latency (ms)}$ . Peak GPU memory and training-time values are hardware- and implementation-dependent and are therefore used only for within-study comparison under the same experimental environment. As shown in Table 12, DSBF-Net incurs higher computational cost than EfficientNet because of its dual-branch architecture and additional feature-interaction modules. However, under the present experimental

setting, DSBF-Net exhibits lower inference latency and lower peak GPU memory consumption than the Cross-Attention fusion baseline while also achieving higher classification accuracy.

#### 4.12. *Feature visualization and interpretability analysis*

The interpretability analysis was organized around the relationship between branch complementarity, intermediate fusion behavior, and category-level errors. Instead of treating the two backbone branches as having fixed global or local roles a priori, the revised manuscript interprets their complementarity through controlled backbone comparisons, fusion-strategy comparisons, and internal SBFM ablation. The results indicate that heterogeneous feature extraction alone is insufficient, whereas performance improves when aligned multiplicative responses are stabilized, refined, and combined with first-order branch evidence.

In this study, the primary mechanism-level evidence is provided by the redesigned SBFM structure, the Split-Attention group-number sensitivity analysis, the internal SBFM ablation, and the category-level confusion analysis. These analyses jointly clarify how channel alignment, aligned multiplication, response normalization, Coordinate Attention refinement, and residual aggregation contribute to the final representation. More extensive activation-map visualization, such as gradient-weighted class activation mapping (Grad-CAM) and intermediate feature heatmaps generated from final checkpoints, will be considered in future work to further examine spatial response behavior.

#### 4.13. *Limited-label classification analysis*

For this separate limited-label experiment, a class-balanced protocol was used to assess model behavior under scarce annotation without converting the study into episodic meta-learning, following the broader limited-label and few-shot remote sensing literature [42,43]. A fixed class-stratified evaluation subset comprising 20% of the dataset was first held out. From the remaining data, labeled pools containing 5, 10, or 20 images per class were independently sampled. Within each labeled pool, a small subset was used for model selection, and the remaining labeled samples were used for parameter optimization. Each setting was repeated using five independent random samplings, and results are reported as mean  $\pm$  standard deviation. As shown in Table 13, DSBF-Net maintains a consistent advantage across all label budgets, with the largest gain over EfficientNet observed under the smallest labeled-data setting.

#### 4.14. *Confusion matrix and error analysis*

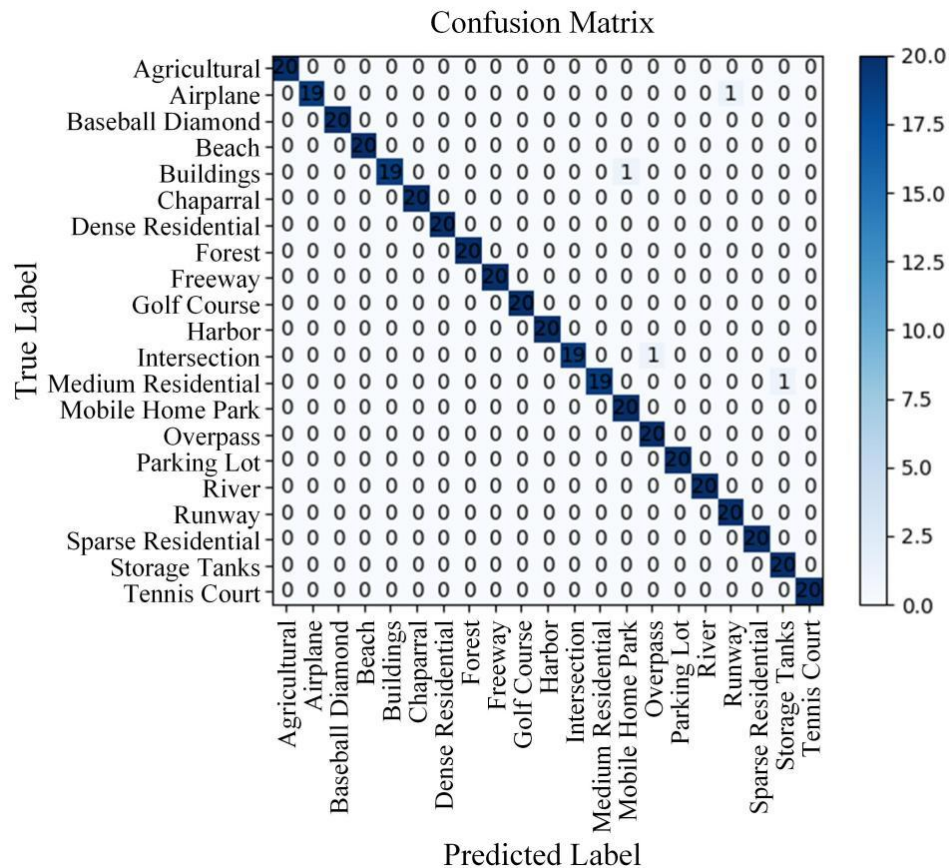
Confusion matrices for UCM and AID are presented in Figures 6 and 7 using representative held-out evaluation folds. The UCM matrix is strongly diagonal, with residual errors concentrated among road-related and residential-density categories, whereas AID exhibits more pronounced confusion because of stronger appearance variation and semantic overlap. To complement the fold-level visualization, Table 14 summarizes the dominant AID error pairs obtained by aggregating predictions across the five held-out evaluation folds.

**Table 13.** Limited-label classification on the AID dataset.

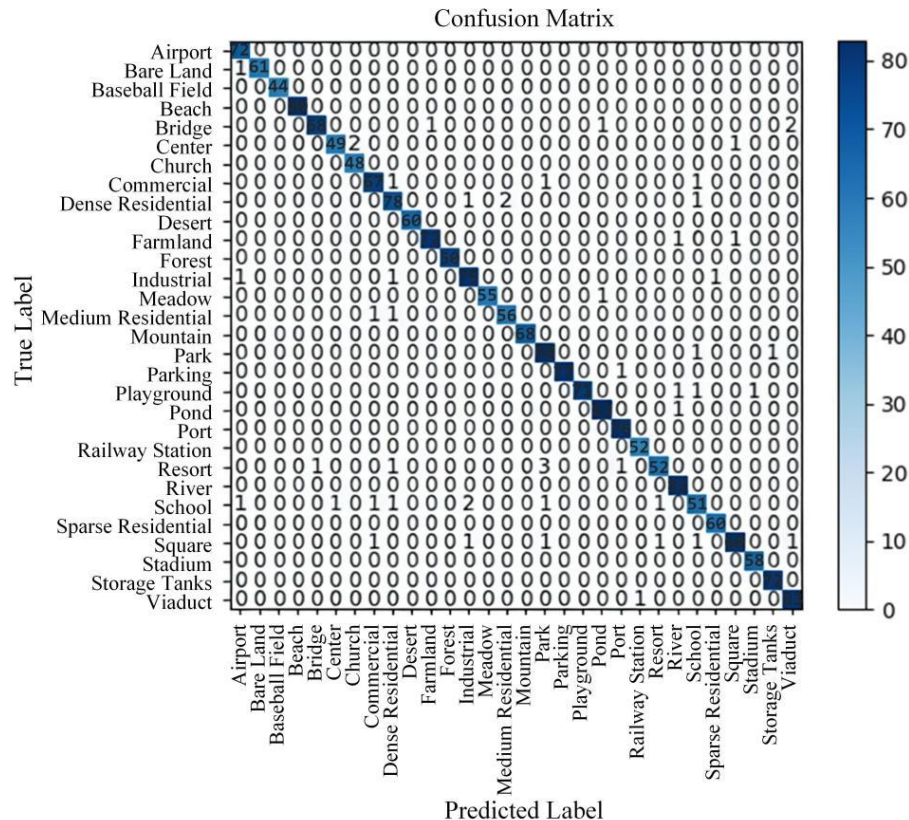
| Labeled images/class | EfficientNet | MobileNetV3  | Dual-branch Concat | DSBF-Net     |
|----------------------|--------------|--------------|--------------------|--------------|
| 5-shot               | 72.84 ± 1.56 | 68.93 ± 1.88 | 70.15 ± 1.72       | 77.46 ± 1.34 |
| 10-shot              | 82.11 ± 1.12 | 77.58 ± 1.45 | 79.33 ± 1.36       | 86.02 ± 0.97 |
| 20-shot              | 89.68 ± 0.76 | 85.04 ± 0.98 | 87.12 ± 0.84       | 92.11 ± 0.65 |

**Table 14.** Dominant AID confusion pairs.

| True class        | Predicted class    | Misclassified samples |
|-------------------|--------------------|-----------------------|
| Dense residential | Medium residential | 18                    |
| Commercial        | Dense residential  | 15                    |
| School            | Commercial         | 13                    |
| Resort            | Medium residential | 11                    |
| Square            | Stadium            | 9                     |



**Figure 6.** Confusion matrix of DSBF-Net on the UCM dataset.



**Figure 7.** Confusion matrix of DSBF-Net on the AID dataset.

Dense and medium residential scenes remain difficult because building density varies continuously rather than following a sharp semantic boundary. Commercial and school scenes share large rectangular roofs, paved regions, and road access patterns. Resort and medium-residential scenes contain similar low-rise structures embedded in vegetation, while square and stadium scenes both include large open geometric regions. These findings indicate that remaining errors are concentrated in fine-grained functional categories whose distinction depends on object arrangement and broader context.

## 5. Discussion

The combined experiments indicate that the benefit of DSBF-Net cannot be reduced to the use of two backbones alone. Direct concatenation underperforms EfficientNet on AID, whereas SBFM recovers and improves performance. The backbone-pair comparison further suggests that structural diversity must be coordinated by an appropriate interaction mechanism: CNN-transformer and CNN-SSM pairs remain competitive, but the proposed EfficientNet + MobileNetV3 pairing provides the strongest overall result in the present comparison.

The internal SBFM ablation supports the interpretation that direct Hadamard multiplication is insufficient. Progressive gains emerge from response stabilization, local  $\ell_2$  normalization, Coordinate

Attention, and residual aggregation. Robustness results show smaller accuracy degradation for DSBF-Net under noise, brightness changes, and occlusion, while the complexity analysis indicates that these gains are obtained with moderate additional cost relative to a single EfficientNet and lower latency than the Cross-Attention fusion alternative. Limited-label experiments further suggest that explicit cross-branch interaction is the most useful when supervision is scarce.

It should also be noted that the proposed diagonal bilinear interaction is not claimed to be a full bilinear pooling operation. The raw interaction  $B_{\text{raw}} = X \odot Y$  is equivalent to aligned Hadamard multiplication. The effectiveness of SBFM arises from using this product as an intermediate interaction signal and then stabilizing, refining, and preserving it through signed square-root transformation, normalization, Coordinate Attention, and residual aggregation. Therefore, the main contribution of SBFM lies in the structured fusion pipeline surrounding the multiplicative response rather than in the elementary product operation alone.

Several limitations remain. The empirical study is restricted to UCM and AID, and literature comparisons use heterogeneous training ratios and pretraining policies. The dual-branch architecture still introduces extra computation relative to a single backbone, and the remaining AID errors are concentrated in visually similar urban categories. Future work should evaluate broader datasets, parameter-efficient interaction modules, and uncertainty-aware decision mechanisms.

## 6. Conclusions

This paper proposes DSBF-Net for airborne remote sensing scene classification. The framework combines structurally heterogeneous EfficientNet and MobileNetV3 branches with two-stage Split-Attention and SBFM. SBFM is formulated as a position-wise diagonal bilinear interaction rather than conventional full bilinear pooling and integrates spatial-channel alignment, signed square-root transformation, local  $\ell_2$  normalization, Coordinate Attention refinement, and residual aggregation. On UCM and AID, the complete model achieves accuracies of 99.07% and 97.58%, respectively, and outperforms the evaluated shallow fusion strategies and representative backbone baselines.

The expanded evaluation examines backbone complementarity, hyperparameter sensitivity, internal SBFM contributions, corruption robustness, computational efficiency, limited-label behavior, and category-level error patterns. Together, these analyses identify which components drive the observed improvement and clarify the conditions under which the model remains reliable. Future work may extend the framework to broader datasets and more parameter-efficient interaction designs while retaining explicit cross-branch modeling.

## Author contributions

H.L. led the study and completed most of the research and writing tasks, while C.Z. contributed to validation and data curation. All authors reviewed the manuscript.

## Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

## Funding

This research received no external funding.

## Data availability statement

The original contributions presented in the study are included in the article. Further inquiries can be directed to the corresponding author.

## Conflicts of interest

The authors declare no conflicts of interest.

## References

1. Feng J, Luo H, Gu Z (2025) Improving semi-supervised remote sensing scene classification via multilevel feature fusion and pseudo-labeling. *Int J Appl Earth Obs Geoinf* 136: 104335. <https://doi.org/10.1016/j.jag.2024.104335>
2. Wang W, Sun Y, Li J, et al. (2024) Frequency and spatial based multi-layer context network (FSCNet) for remote sensing scene classification. *Int J Appl Earth Obs Geoinf* 128: 103781. <https://doi.org/10.1016/j.jag.2024.103781>
3. Yang Y, Newsam S (2013) Geographic image retrieval using local invariant features. *IEEE T Geosci Remote S* 51: 818–832. <https://doi.org/10.1109/TGRS.2012.2205158>
4. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, San Diego, CA, USA, 1: 886–893. <https://doi.org/10.1109/CVPR.2005.177>
5. Sivic, Zisserman (2003) Video google: a text retrieval approach to object matching in videos. *Proceedings Ninth IEEE International Conference on Computer Vision*, Nice, France, 2: 1470–1477. <https://doi.org/10.1109/ICCV.2003.1238663>
6. Kaul A, Kumari M (2023) A literature review on remote sensing scene categorization based on convolutional neural networks. *Int J Remote Sens* 44: 2611–2642. <https://doi.org/10.1080/01431161.2023.2204200>
7. Huang Q, Jiang F, Huang C (2025) Remote sensing scene classification with relation-aware dynamic graph neural networks. *Eng Appl Artif Intel* 150: 110513. <https://doi.org/10.1016/j.engappai.2025.110513>

8. Thapa A, Horanont T, Neupane B, et al. (2023) Deep learning for remote sensing image scene classification: a review and meta-analysis. *Remote Sens* 15: 4804. <https://doi.org/10.3390/rs15194804>
9. Chen X, Ma M, Li Y, et al. (2023) Hierarchical feature fusion of transformer with patch dilating for remote sensing scene classification. *IEEE T Geosci Remote S* 61: 1–16. <https://doi.org/10.1109/TGRS.2023.3331880>
10. Lu Y, Zhu Y, Feng H, et al. (2024) Remote sensing scene classification using multi-domain semantic high-order network. *Image Vis Comput* 143: 104948. <https://doi.org/10.1016/j.imavis.2024.104948>
11. Hou YE, Yang K, Dang L, et al. (2023) Contextual spatial-channel attention network for remote sensing scene classification. *IEEE Geosci Remote S Lett* 20: 1–5. <https://doi.org/10.1109/LGRS.2023.3304645>
12. Chen J, Yi J, Chen A, et al. (2023) EFCOMFF-net: a multiscale feature fusion architecture with enhanced feature correlation for remote sensing image scene classification. *IEEE T Geosci Remote S* 61: 1–17. <https://doi.org/10.1109/TGRS.2023.3255211>
13. Guo N, Jiang M, Wang D, et al. (2024) Scene classification for remote sensing image of land use and land cover using dual-model architecture with multilevel feature fusion. *Int J Digit Earth* 17: 2353166. <https://doi.org/10.1080/17538947.2024.2353166>
14. Yue H, Qing L, Zhang Z, et al. (2024) MSE-net: a novel master–slave encoding network for remote sensing scene classification. *Eng Appl Artif Intel* 132: 107909. <https://doi.org/10.1016/j.engappai.2024.107909>
15. Duan Y, Song C, Zhang Y, et al. (2025) STMSF: swin transformer with multi-scale fusion for remote sensing scene classification. *Remote Sens* 17: 668. <https://doi.org/10.3390/rs17040668>
16. Xia J, Zhou Y, Tan L (2023) DBGA-net: dual-branch global–local attention network for remote sensing scene classification. *IEEE Geosci Remote S Lett* 20: 1–5. <https://doi.org/10.1109/LGRS.2023.3264817>
17. Zhao Y, Gong M, Qin AK, et al. (2024) Gradient-guided multiscale focal attention network for remote sensing scene classification. *IEEE T Geosci Remote S* 62: 1–18. <https://doi.org/10.1109/TGRS.2024.3424489>
18. Ye Z, Zhang Y, Zhang J, et al. (2024) A multiscale incremental learning network for remote sensing scene classification. *IEEE T Geosci Remote S* 62: 1–15. <https://doi.org/10.1109/TGRS.2024.3353737>
19. Roy SK, Jamali A, Biswas K, et al. (2025) ViCxLSTM: an extended long short-term memory vision transformer for complex remote sensing scene classification. *Int J Appl Earth Obs Geoinf* 143: 104801. <https://doi.org/10.1016/j.jag.2025.104801>
20. Guo N, Jiang M, Gao L, et al. (2023) HFCC-net: a dual-branch hybrid framework of CNN and CapsNet for land-use scene classification. *Remote Sens* 15: 5044. <https://doi.org/10.3390/rs15205044>
21. Lin TY, RoyChowdhury A, Maji S (2015) Bilinear CNN models for fine-grained visual recognition. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 1449–1457. <https://doi.org/10.1109/ICCV.2015.170>

22. Gao Y, Beijbom O, Zhang N, et al. (2016) Compact bilinear pooling. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 317–326. <https://doi.org/10.1109/CVPR.2016.41>
23. Song W, Zhang Y, Wang C, et al. (2024) Remote sensing scene classification based on semantic-aware fusion network. *IEEE Geosci Remote S Lett* 21: 1–5. <https://doi.org/10.1109/LGRS.2024.3470773>
24. Zhang H, Wu C, Zhang Z, et al. (2022) ResNeSt: Split-Attention Networks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, New Orleans, LA, USA, 2736–2746. <https://doi.org/10.1109/CVPRW56347.2022.00309>
25. Hou Q, Zhou D, Feng J (2021) Coordinate attention for efficient mobile network design. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, 13713–13722. <https://doi.org/10.1109/CVPR46437.2021.01350>
26. Li Z, Hu J, Wu K, et al. (2024) Adjacent-Atrous Mechanism for Expanding Global Receptive Fields: An End-to-End Network for Multiattribute Scene Analysis in Remote Sensing Imagery. *IEEE T Geosci Remote S* 62: 1–19. <https://doi.org/10.1109/TGRS.2024.3422007>
27. Tan M, Le QV (2020) EfficientNet: rethinking model scaling for convolutional neural networks. *arXiv*. <https://doi.org/10.48550/arXiv.1905.11946>
28. Howard A, Sandler M, Chu G, et al. (2019) Searching for MobileNetV3. *arXiv*. <https://doi.org/10.48550/arXiv.1905.02244>
29. Ji Y, Shi W, Lei J, et al. (2025) DBRSNet: a dual-branch remote sensing image segmentation model based on feature interaction and multi-scale feature fusion. *Sci Rep* 15: 27786. <https://doi.org/10.1038/s41598-025-13236-4>
30. Li Z, Hu J, Guan W, et al. (2026) FTransMamba: A Multi-Stage Fusion Transformer and Mamba Modeling Network for Multimodal Remote Sensing Scene Understanding. *Pattern Recognit* 179. <https://doi.org/10.1016/j.patcog.2026.113625>
31. Li Z, Zhang Y, Hu J, et al. (2026) HDC-Net: A Multimodal Remote Sensing Semantic Segmentation Network with Hierarchical Dual-Stream Fusion and Cross-Token Interaction. *Knowl-Based Syst* 337. [10.1016/j.knosys.2026.115416](https://doi.org/10.1016/j.knosys.2026.115416)
32. Yang Y, Newsam S (2010) Bag-of-visual-words and spatial extensions for land-use classification. *GIS '10: Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, New York, NY, USA: Association for Computing Machinery, 270–279. <https://doi.org/10.1145/1869790.1869829>
33. Xia GS, Hu J, Fan H, et al. (2017) AID: a benchmark data set for performance evaluation of aerial scene classification. *IEEE T Geosci Remote S* 55: 3965–3981. <https://doi.org/10.1109/TGRS.2017.2685945>
34. He K, Zhang X, Ren S, et al. (2016) Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 770–778. <https://doi.org/10.1109/CVPR.2016.90>
35. Xie S, Girshick R, Dollar P, et al. (2017) Aggregated residual transformations for deep neural networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 5987–5995. <https://doi.org/10.1109/CVPR.2017.634>

36. Liu Z, Hao Z, Han K, et al. (2024) GhostNetV3: exploring the training strategies for compact models. *arXiv*. <https://doi.org/10.48550/arXiv.2404.11202>
37. Li Y, Yuan G, Wen Y, et al. (2022) EfficientFormer: vision transformers at MobileNet speed. *arXiv*. <https://doi.org/10.48550/arXiv.2206.01191>
38. He Z, Li G, Wang Z, et al. (2024) Deep Ensemble Remote Sensing Scene Classification via Category Distribution Association. *Remote Sens* 16: 4084. <https://doi.org/10.3390/rs16214084>
39. Xu H, Song Y, Xu G, et al. (2025) HETMCL: High-Frequency Enhancement Transformer and Multi-Layer Context Learning Network for Remote Sensing Scene Classification. *Sensors* 25: 3769. <https://doi.org/10.3390/s25123769>
40. Zeng H, Zou S, Yao C, et al. (2025) LGNet: A Symmetric Dual-Branch Lightweight Model for Remote Sensing Scene Classification Based on Lie Group Feature Extraction and Cross-Attention Mechanism. *Symmetry* 17: 780. <https://doi.org/10.3390/sym17050780>
41. Zhu X, Huang J, Wang H (2026) Frequency Domain and Gradient-Spatial Multi-Scale Swin KANsformer for Remote Sensing Scene Classification. *Remote Sens* 18: 517. <https://doi.org/10.3390/rs18030517>
42. Dutta S, Das M (2023) Remote sensing scene classification under scarcity of labelled samples—a survey of the state-of-the-arts. *Comput Geosci* 171: 105295. <https://doi.org/10.1016/j.cageo.2022.105295>
43. Qiu C, et al. (2024) Few-shot remote sensing image scene classification: recent advances, new baselines, and future trends. *ISPRS J Photogramm Remote Sens* 209: 368–382. <https://doi.org/10.1016/j.isprsjprs.2024.02.005>



AIMS Press

© 2026 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0>)