remote sensing

MDPI

Article

Joint Classification of Hyperspectral and LiDAR Data Based on Adaptive Gating Mechanism and Learnable Transformer

Minhui Wang [1], Yaxiu Sun [1,*], Jianhong Xiang [1], Rui Sun [1] and Yu Zhong [2]

[1] Key Laboratory of Advanced Ship Communication and Information Technology, Harbin Engineering University, Harbin 150001, China; minhuiwang@hrbeu.edu.cn (M.W.); xiangjianhong@hrbeu.edu.cn (J.X.); sunrui996633@163.com (R.S.)
[2] Agile and Intelligent Computing Key Laboratory, Chengdu 610000, China; jade.zhong@hotmail.com
* Correspondence: sunyaxiu@hrbeu.edu.cn

Abstract: Utilizing multi-modal data, as opposed to only hyperspectral image (HSI), enhances target identification accuracy in remote sensing. Transformers are applied to multi-modal data classification for their long-range dependency but often overlook intrinsic image structure by directly flattening image blocks into vectors. Moreover, as the encoder deepens, unprofitable information negatively impacts classification performance. Therefore, this paper proposes a learnable transformer with an adaptive gating mechanism (AGMLT). Firstly, a spectral–spatial adaptive gating mechanism (SSAGM) is designed to comprehensively extract the local information from images. It mainly contains point depthwise attention (PDWA) and asymmetric depthwise attention (ADWA). The former is for extracting spectral information of HSI, and the latter is for extracting spatial information of HSI and elevation information of LiDAR-derived rasterized digital surface models (LiDAR-DSM). By omitting linear layers, local continuity is maintained. Then, the layer Scale and learnable transition matrix are introduced to the original transformer encoder and self-attention to form the learnable transformer (L-Former). It improves data dynamics and prevents performance degradation as the encoder deepens. Subsequently, learnable cross-attention (LC-Attention) with the learnable transfer matrix is designed to augment the fusion of multi-modal data by enriching feature information. Finally, poly loss, known for its adaptability with multi-modal data, is employed in training the model. Experiments in the paper are conducted on four famous multi-modal datasets: Trento (TR), MUUFL (MU), Augsburg (AU), and Houston2013 (HU). The results show that AGMLT achieves optimal performance over some existing models.

Keywords: hyperspectral image (HSI); light detection and ranging (LiDAR) data; convolutional neural network (CNN); vision transformer; cross-attention

Citation: Wang, M.; Sun, Y.; Xiang, J.; Sun, R.; Zhong, Y. Joint Classification of Hyperspectral and LiDAR Data Based on Adaptive Gating Mechanism and Learnable Transformer. *Remote Sens.* **2024**, *16*, 1080. https://doi.org/10.3390/rs16061080

Academic Editor: Joaquín Martínez-Sánchez

Received: 1 February 2024
Revised: 7 March 2024
Accepted: 13 March 2024
Published: 19 March 2024

1. Introduction

Data from multiple remote sensing imaging devices in the same geographic area are available, which makes it possible to analyze land cover material using multi-modal data. Various remote sensing imaging sensor technologies can effectively capture different features of land cover materials. For example, hyperspectral imagers can acquire reflected spectral information while acquiring ground spatial information [1], and light detection and ranging (LiDAR) can measure the elevation information of ground objects [2]. Integrating multi-modal data allows for the construction of a more detailed and comprehensive feature representation of ground objects.

Since the late 20th century, hyperspectral imaging has emerged as a pivotal detection technique in remote sensing, which employs an imaging spectrometer to precisely segment the spectrum across visible near-infrared, short-wave infrared, and even long-wave infrared ranges. This process generates tens to hundreds of spectral bands for imaging ground objects simultaneously. It captures the spectral details of various ground objects

alongside their spatial distribution, thereby merging image and spectral information effectively. Therefore, hyperspectral image (HSI) is widely used in land-cover classification [3], ecosystem measurement [4], military reconnaissance [5], target detection [6], and many other fields [7–9]. Among them, land cover classification, also known as hyperspectral image classification (HSIC), is particularly important in HSI processing tasks.

HSIC uses spectral dimension information and spatial dimension information to assign a category identifier to each pixel [10]. Early HSIC tasks mainly relied on data from a single mode. Roy et al. [11] integrated the three-dimensional convolutional neural network (3DCNN) and the two-dimensional convolutional neural network (2DCNN) to design a hybrid convolutional neural network (CNN) for spectral–spatial feature representation. Sun et al. [12] designed a classification model with heterogeneous spectral–spatial attention convolutional neural blocks, which simultaneously extracted the three-dimensional (3D) features from HSI. Although CNN has excellent performance, it has some limitations when dealing with long sequence properties of spectral features due to its inherent network backbone structure. Due to the power of the vision transformer, Hong et al. [13] developed a spectral transformer for extracting spectral discriminative features from bands of HSI. While transformer networks excel at simulating global interactions between token embeddings via self-attention (SA) mechanisms, they fall short in effectively disseminating local information among tokens [14]. Therefore, Sun et al. [15] combined the CNN module with a transformer encoder to form a new spectral–spatial feature tokenization transformer for representing sequential relations and high-level semantic features. Wang et al. [16] proposed a new spectral–spatial kernel combined with an improved visual transformation method to extract spectral–spatial features of HSI together.

For HSI, the spectral features of identical ground objects may vary, and, conversely, similar spectral features can correspond to different ground objects [17]. Therefore, it is necessary to supplement the ground object information with the multi-modal remote sensing data in the same area. LiDAR-DSM data, which primarily contain terrain variations and the feature heights of surface objects [18], are often employed in conjunction with HSI for joint classification, thereby enhancing classification accuracy. Compared with HSI alone, the advantage of multi-modal image collaborative classification is that it can fully describe the features of the target and make a more accurate judgment of the target. Consequently, numerous research initiatives have been undertaken to harness the complementary information between HSI and LiDAR-DSM. Pedergnana et al. [19] combined morphological extended attribute profiles on HSI and LiDAR-DSM data with raw spectral data from HSI for classification. However, directly stacking high-dimensional features can trigger the Hughes phenomenon, especially when training samples are scarce. Rasti et al. [20] utilized extinction profiles to derive spatial and elevation information from HSI and LiDAR-DSM data and integrate them with spectral information through a feature fusion method based on Orthogonal Total Variation Component Analysis (OTVCA), which facilitates the processing of fusion features in the lower-dimensional space.

However, traditional methods rely heavily on prior information, so it is difficult to improve the classification accuracy while maintaining robustness. Deep learning can learn high-level semantic information from data using the end-to-end pattern [21]. Roy et al. [22] proposed a joint feature learning fusion mechanism based on CNN and spatial morphological blocks to generate high-precision land cover maps. Song et al. [23] proposed a new hash-based deep metric learning approach that focuses on sample correlations between single-source and cross-source data. Xu et al. [24] used two-branch CNN to extract spatial and spectral information of HSI and a cascaded network to extract elevation information of LiDAR-DSM and carried out block-level fusion and classification. Although CNN has excellent performance, due to its inherent network backbone structure, it has certain limitations in processing long sequence attributes of features. Therefore, inspired by the classification of HSI, researchers have applied the fusion model of CNN and transformer to the joint classification task of HSI and LiDAR-DSM. Ding et al. [25] introduced the Global–Local Transformer Network (GLT-Net), designed to capture the global–local cor-

relation features from inputs, effectively enhancing classification outcomes. This method only concatenated features from HSI and LiDAR-DSM without deep information fusion learning. Zhang et al. [26] developed the Local Information Interaction Transformer (LIIT), addressing the challenge of redundant or deficient complementary information between HSI and LiDAR-DSM data by dynamically integrating multi-modal features via the transformer, also achieving promising results. However, it has some shortcomings in extracting fine-grained information from images. Xu et al. [27] proposed a transformer with multi-branch interaction to extract spectral, spatial, and elevation information simultaneously. Its spectral and spatial information is learned independently before being concatenated, rather than interactive learning on multi-modal data. Roy et al. [28] proposed a transformer backbone to extract feature representations from multiple sources of data and use class tokens for final classification. Zhao et al. [29] proposed a novel dual-branch approach, combining a hierarchical CNN with a transformer network, designed to fuse multi-modal heterogeneous information and enhance joint classification performance. While these two methods enable the interactive learning of multi-modal data, feature extraction using a shallow CNN is relatively simplistic, lacking local fine-grained detail, and the feature dynamics within the fusion structure are inadequate.

To fully extract the local fine-grained features in HSI and LiDAR-DSM data and improve classification performance, a novel adaptive joint classification method based on the adaptive gating mechanism and learnable transformer (AGMLT) is designed. The dual-branch spectral–spatial adaptive gating mechanism (SSAGM) is engineered to concurrently extract spectral–spatial features from HSI and elevation features from LiDAR-DSM. Additionally, the layer scale and learnable transition matrices are incorporated into the original transformer encoder to enhance training dynamics. Learnable transition matrices are further applied to cross-attention, augmenting the attention graphs across various levels. The model training utilized poly loss, ultimately leading to improved classification performance. The key contributions of AGMLT are summarized as follows.

1. The Gated Spatial Attention Unit (GSAU) [30] is introduced into the joint classification of HSI and LiDAR-DSM, which is improved to design a dual-branch SSAGM feature extraction module. SSAGM encompasses the point depthwise attention module (PDWA) and the asymmetric depthwise attention module (ADWA). The PDWA primarily aims at extracting the spectral features from HSI, while the ADWA focuses on extracting spatial information from HSI and elevation information from LiDAR-DSM. This approach allows for the omission of the linear layer to emphasize local continuity without compromising complexity.
2. The learnable transformer (L-Former) is designed to enhance data dynamics and mitigate performance decline as the depth of the transformer increases. The layer scale is incorporated into the output of each residual block, with different output channels being multiplied by distinct values to further refine the features. Concurrently, a learnable transition matrix is integrated into the self-attention (SA) to develop learnable self-attention (LS-Attention, LSA), which addresses the issue of centralized decomposition and facilitates the training of deeper transformers.
3. The learnable transition matrix is integrated into cross-attention, forming learnable cross-attention (LC-Attention). This integration diminishes the similarity among attention maps, thereby augmenting the diversity of the features.
4. Poly loss is implemented for classifying to improve the model training. Remote sensing datasets frequently exhibit uneven distributions and potential overlaps among samples of the same type. Furthermore, the features of data differ across various modalities. Poly loss is a versatile loss function suited for multi-modal data fusion classification.

The rest of this paper is arranged as follows. Section 2 expounds on the relevant theory of the proposed method AGMLT. Section 3 presents the four well-known multi-modal datasets, experimental settings, and various experiments on the datasets. In Section 4, the ablation analysis and performance of different percentages of training samples are discussed. Finally, Section 5 concludes the paper.

2. Methodology

The AGMLT proposed in this paper is shown in Figure 1. Firstly, SSAGM is designed to enhance feature extraction. Then, the L-Former with two learnable matrixes is proposed to increase the data dynamics and prevent performance degradation as the transformer deepens. At the same time, LC-Attention with a learnable matrix enriches the feature information of multi-modal fusion. Finally, poly loss is the loss function for AGMLT, which is more suitable for data with different modes.

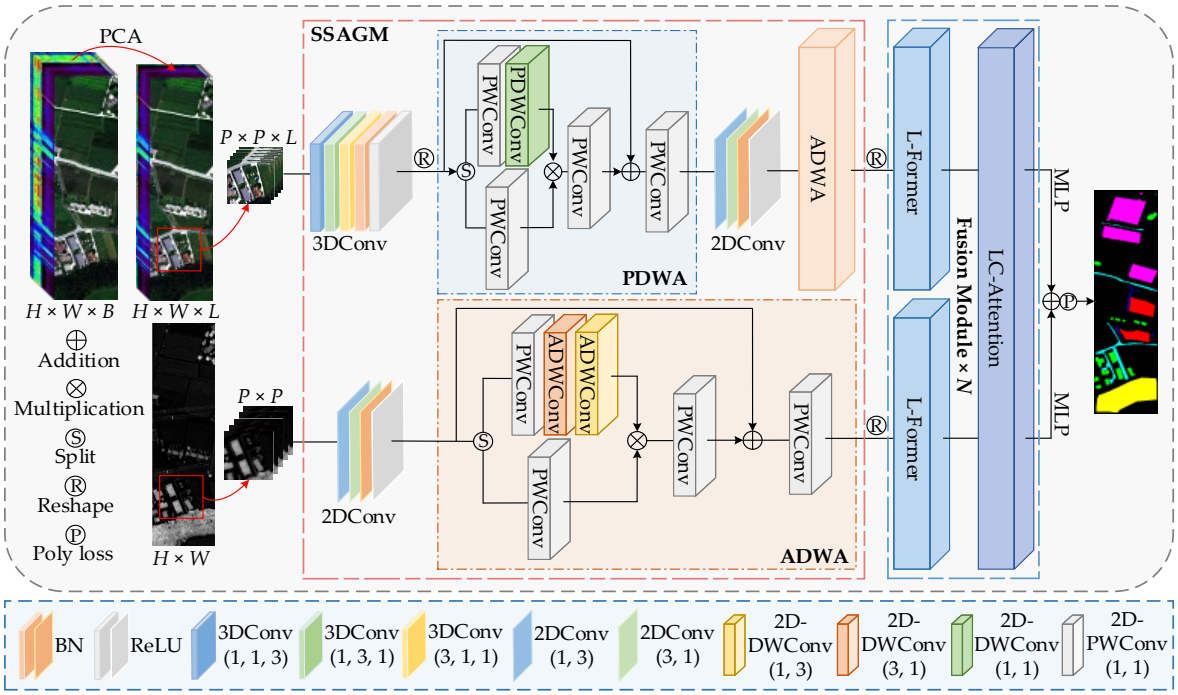


Figure 1. Structure for proposed AGMLT model. The SSAGM is proposed to exclude the linear layer and capture local continuity while considering complexity. L-Former is designed to increase the data dynamics and prevent performance degradation as the transformer deepens. LC-Attention is designed for enriching the feature information. Poly loss is a flexible loss function suitable for multi-modal data fusion classification.

From Algorithm 1, the original input data of AGMLT could be represented as $X_{IN}^{HSI} \in R_{H\times W\times B}$ and $X_{IN}^{LiDAR} \in R_{H\times W}$, where height is H, width is W, and spectra are B. HSI has a large number of spectral bands, which can provide rewarding information but also significantly increases the cost of computing. Principal component analysis (PCA) is used to reduce the spectral number of hyperspectral images. The data after PCA would be reshaped to $X_{PCA}^{HSI} \in R_{H\times W\times L}$, of which L is the number of bands after PCA. Since HSI is the 3D data, X_{PCA}^{HSI} is sent to 3DCNN to extract 3D features X_{3DCNN}^{HSI}. X_{3DCNN}^{HSI} is reshaped to X_{2D}^{HSI} to make the data dimension match the subsequent attention module. Put X_{2D}^{HSI} into the PDWA to focus on extracting spectral features. The output X_{PDWA}^{HSI} is sent to 2DCNN for simple extracting the features. Then, the outputs of 2DCNN are sent to ADWA to extract the spatial information and get the output features X_{ADWA}^{HSI}. LiDAR-DSM is the two-dimensional (2D) data, so it could calculate directly with 2DCNN. The outputs are sent to ADWA for the extraction of the elevation information and to gain the output. X_{ADWA}^{LiDAR}. Next, the reshaped X_{ADWA}^{HSI} and X_{ADWA}^{LiDAR} are integrated into the Fusion Module. The Fusion Module may loop N times. Then, the outputs X_{LLF}^{HSI} and X_{LLF}^{LiDAR} of the L-Former in the Fusion Module are put into LC-Attention for information fusion of multi-modal data. Finally, the outputs X_{LCA}^{HSI} and X_{LCA}^{LiDAR} are fed into the multi-layer perceptron (MLP) separately for the final classification. Poly loss is used to measure the degree of inconsistency between the predicted labels Y_P and the true labels Y_L.

Algorithm 1 The algorithm flow of AGMLT

Input **HSI:** $X_{IN}^{HSI} \in R_{H\times W\times B}$, **LiDAR-DSM:** $X_{IN}^{LiDAR} \in R_{H\times W}$, **Labels:** $Y_L \in R_{H\times W}$, Patches = 11 × 11, PCA = 30.

Output **Prediction:** Y_P.

Initialize: batch size = 64, epochs = 100, learning rate depends on datasets.

PCA: $X_{PCA}^{HSI} \in R_{H\times W\times L}$.

Create all sample patches from X_{PCA}^{HSI}, X_{IN}^{LiDAR}, and divide them into the training sets D_{train} and the test sets D_{test}. (D_{train} contains the labels, and D_{test} does not contain the labels).

Training AGMLT (begin)

for epoch in range(epochs):

  for i, $(D_{train}^{HSI}, D_{train}^{LiDAR}, Y_L)$ in enumerate (D_{train}):

$X_{PCA}^{HSI} \xrightarrow{\text{3DCNN}} X_{3DCNN}^{HSI} \xrightarrow{\text{reshape}} X_{2D}^{HSI} \xrightarrow{\text{PDWA}} X_{PDWA}^{HSI} \xrightarrow{\text{2DCNN}} X_{2DCNN}^{HSI} \xrightarrow{\text{ADWA}} X_{ADWA}^{HSI} \xrightarrow{\text{reshape}} X_{1D}^{HSI}$

$X_{IN}^{LiDAR} \xrightarrow{\text{2DCNN}} X_{2DCNN}^{LiDAR} \xrightarrow{\text{ADWA}} X_{ADWA}^{LiDAR} \xrightarrow{\text{reshape}} X_{1D}^{LiDAR}$

$X_{1D}^{HSI} \xrightarrow{\text{LL-Former}} X_{LLF}^{HSI}, X_{1D}^{LiDAR} \xrightarrow{\text{LL-Former}} X_{LLF}^{LiDAR}$

$X_{LLF}^{HSI}, X_{LLF}^{LiDAR} \xrightarrow{\text{LC-Attention}} X_{LCA}^{HSI}, X_{LCA}^{LiDAR}$

$X_{OUT} = \text{MLP}(X_{LCA}^{HSI}) + \text{MLP}\left(X_{LCA}^{LiDAR}\right)$

Poly loss(X_{OUT}, Y_L)

Training AGMLT (end) and test AGMLT

$Y_P = \text{AGMTL}_{trained}(D_{test})$

2.1. SSAGM

Although the transformer networks can simulate global interactions between token embeddings through the SA, they are less capable of extracting fine grained local feature patterns [31]. Based on the superior ability of CNNs to model spatial context features, it performs exceptionally well in HSI classification tasks. Simultaneously, many applications have proved that CNNs can extract the deep features of LiDAR-DSM [32]. Therefore, we introduce a CNN to extract features from input data. To further enhance feature representation, we are inspired by GSAU [30] to design the SSAGM. The key components of SSAGM are PDWA and ADWA, which enable the linear layer to be excluded and local continuity to be captured while considering complexity.

PDWA is used to extract spectral features from HSI, which is shown on the left side of Figure 2. PDWA includes pointwise convolution (PWConv), point depthwise convolution (PDWConv), multiplication operation, and residual connection. ADWA is mainly used to extract spatial feature information of HSI and elevation information of LiDAR-DSM. Its structure is shown on the right of Figure 2.

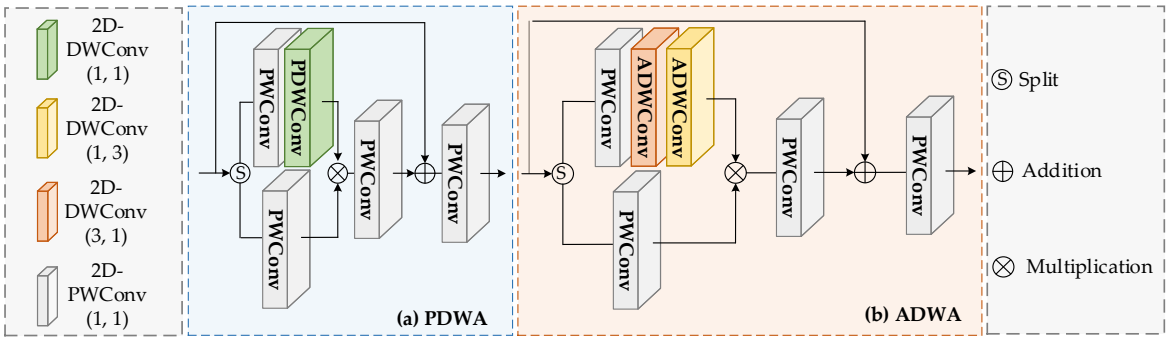


Figure 2. Structure for proposed SSAGM. PDWA is used to extract spectral features from HSI. ADWA is used to extract spatial features from HSI and elevation information from LiDAR-DSM.

The input data of the PDWA are divided into X_{P1} and X_{P2} evenly. X_{P1} is sent to the PWConv layer to obtain X_{PP1}. Feed X_{PP1} into the PDWConv with 1×1 convolution kernel

to yield the output X_{PD}. Groups in the PDWConv layer are equal to the channels of X_{PP1}. Since the convolution kernel size is 1×1 and the number of groups is the same as the input, it achieves the role of focusing on the channel information. X_{P2} also obtains X_{PP2} using a PWConv. To preserve partial original information, there are no operations performed on X_{PP2}. The data obtained by multiplying X_{PD} and X_{PP2} are connected with X_{Pin} via a residual connection. Then, it is sent to the PWConv layer to obtain the output X_{Pout}. The PWConv contains a 1×1 convolution kernel whose purpose is to adjust the data dimension for element-by-element multiplication and residual connection. The main process of PDWA is as follows:

$$\mathrm{PDWA}(X_{P1}, X_{P2}) = F_{\mathrm{PDW}}(X_{P1}) \otimes X_{P2}, \tag{1}$$

where X_{P1} and X_{P2} represent the feature data of the two branches in PDWA, respectively. $F_{\mathrm{PDW}}(\cdot)$ and $\otimes$ represent the PDWConv and multiplication.

ADWA mainly includes the PWConv, two asymmetric depthwise convolution (ADWConv) layers, multiplication operation, and residual connection. This module changes the PDWConv in PDWA to two ADWConv with 3×1 and 1×3 convolution kernels, and other operations are unchanged. The main processes of ADWA are calculated as follows:

$$\mathrm{ADWA}(X_{A1}, X_{A2}) = F_{\mathrm{ADW2}}(F_{\mathrm{ADW1}}(X_{A1})) \otimes X_{A2}, \tag{2}$$

where X_{A1} and X_{A2} represent the features of the two branches in ADWA. $F_{\mathrm{ADW1}}(\cdot)$ and $F_{\mathrm{ADW2}}(\cdot)$ represent two ADWConv.

2.2. L-Former

Figure 3 shows the structural details of the proposed L-Former. Transformer encoders are used to model the deep semantic relationships between tokens of features, which could map the input of L-Former to a sequence of vectors. A class token is embedded in the head of the vector sequence, which obtains the overall sequence. Then, we embed n position encodings into the sequence to obtain multiple tokens. The more proximate the information, the more similarly is encoded. Then, we enter multiple tokens into the transformer encoder. The output of learnable attention (L-Attention) is classified using MLP, which consists of one layer norm (LN) and two fully connected layers. The Gaussian Error Linear Unit (GELU) [33] activation function is used for classification to obtain the final classification result. The above operations are stacked repeatedly N times. As the model goes deep, the attention graphs of the deeper blocks become more similar, which means that adding more blocks to a deep transformer may not improve model performance [34].

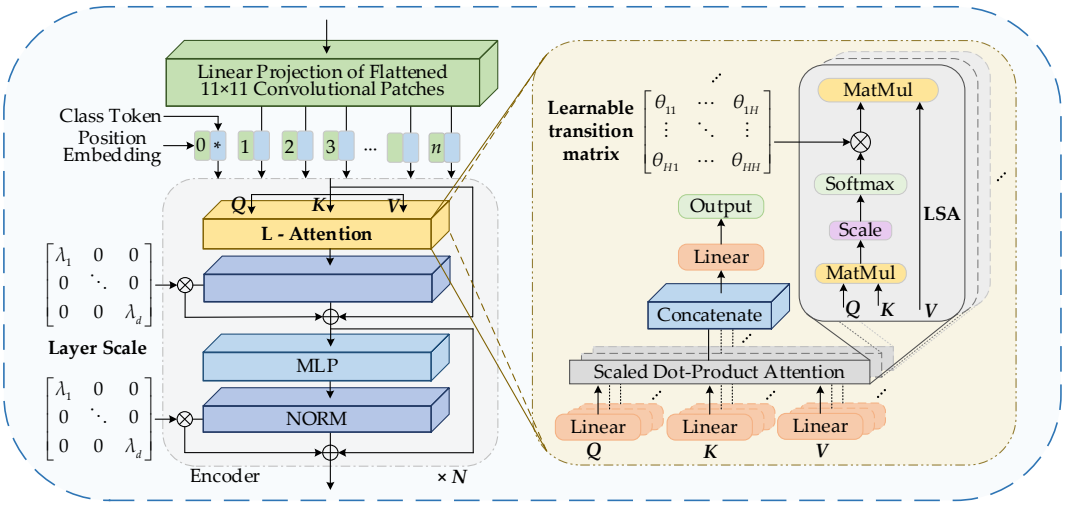


Figure 3. Structure for proposed L-Former. The layer scale makes features more detailed, while the learnable transfer matrix overcomes the problem of centralized decomposition and can train deeper transformers.

Therefore, we introduce the layer scale from cait attention [35] into the transformer encoder. The layer scale adds a learnable diagonal matrix to the output of each residual block, which initialize to near 0. Applying distinct multiplication factors to different channels of the output from SA or MLP refines the features, enhancing their expression quality in the model. It could train deeper volumes. The formulas are as follows:

$$x'_l = x_l + \mathrm{diag}(\lambda_{l,1}, \cdots, \lambda_{l,d}) \times \mathrm{SA}(\eta(x_l)), \tag{3}$$

$$x_{l+1} = x'_l + \mathrm{diag}(\lambda'_{l,1}, \cdots, \lambda'_{l,d}) \times \mathrm{MLP}(\eta(x'_l)), \tag{4}$$

where η is the layer norm and MLP is the feedforward network used in L-Former. $\lambda_{l,1}$ and $\lambda'_{l,1}$ are learnable weights for SA and MLP. The diagonal values are all initialized to the fixed small value σ. When the depth is within 18, σ is set as 0.1, $\sigma = 5 \times 10^{-3}$ is used to the depth within 24, and $\sigma = 5 \times 10^{-6}$ is adopted in the deeper networks.

In order to learn the relationship between feature tokens, $\boldsymbol{W}_q$, $\boldsymbol{W}_k$, and $\boldsymbol{W}_v$ learnable weights are pre-defined for SA. Multiply the feature tokens with the three learnable weights and linearly package them into three different matrices (queries $\boldsymbol{Q}$, keys $\boldsymbol{K}$, and values $\boldsymbol{V}$). The softmax function converts the scores into weight probabilities. And SA is written as follows:

$$\mathrm{SA}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = Softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_K}}\right)\boldsymbol{V}, \tag{5}$$

where d_K represents the dimension of $\boldsymbol{K}$.

At the same time, the learnable transition matrix $\boldsymbol{M} \in R^{N \times N}$ from re-attention is introduced into SA to obtain LSA, which overcomes the problem of concentration breakdown and allows for training a deeper transformer [34].

$$\mathrm{LS-Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \boldsymbol{M}^{\mathrm{T}}\left(Softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_K}}\right)\right)\boldsymbol{V}, \tag{6}$$

where the transformation matrix M is multiplied by the self-attentional mapping of the head dimension. The softmax function is applied to the rows of comparable matrices. Relationships between tokens are modeled by projecting similarities between pairs of $\boldsymbol{Q}$ and $\boldsymbol{K}$, and an attention score is acquired.

We adopt multiple groups of weights to form L-Attention, which is like multi-head attention (MHSA). L-Attention has multiple learnable SA (LSA), and all of these LSA scores are tied together. The expression is as follows:

$$\mathrm{L-Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \mathrm{Concat}(\mathrm{LSA}_1, \mathrm{LSA}_2, \ldots, \mathrm{LSA}_h)\boldsymbol{W}, \tag{7}$$

Here, h is the number of attention heads and $\boldsymbol{W}$ is the parameter matrix.

2.3. LC-Attention

Figure 4 shows the schematic diagram of the fusion encoding module for HSI feature representations and LiDAR-DSM feature representations, respectively.

Taking the fusion encoding module of HSI feature representations as an example, the class token X_{cls}^{HSI} of HSI is spliced with the pixel tokens of LiDAR-DSM data first, and the formulas are

$$X'^{HSI}_{cls} = F^{HSI}\left(X_{cls}^{HSI}\right), \tag{8}$$

$$X_L^{HSI} = \left[\left(X'^{HSI}_{cls}\right) \cup \left(X^{LiDAR} - X_{cls}^{LiDAR}\right)\right], \tag{9}$$

where X_{cls}^{HSI} is the class token of HSI feature representations, and X_{cls}^{LiDAR} is the class token of LiDAR-DSM feature representations. $F^{HSI}(\cdot)$ is a linear mapping function for dimensional alignment. X'^{HSI}_{cls} represents the transformed class token that is consistent with

the X_{cls}^{LiDAR} dimension. X_L^{HSI} is represented the new LiDAR-DSM feature representations, where the original X_{cls}^{LiDAR} is replaced by X'^{HSI}_{cls}.

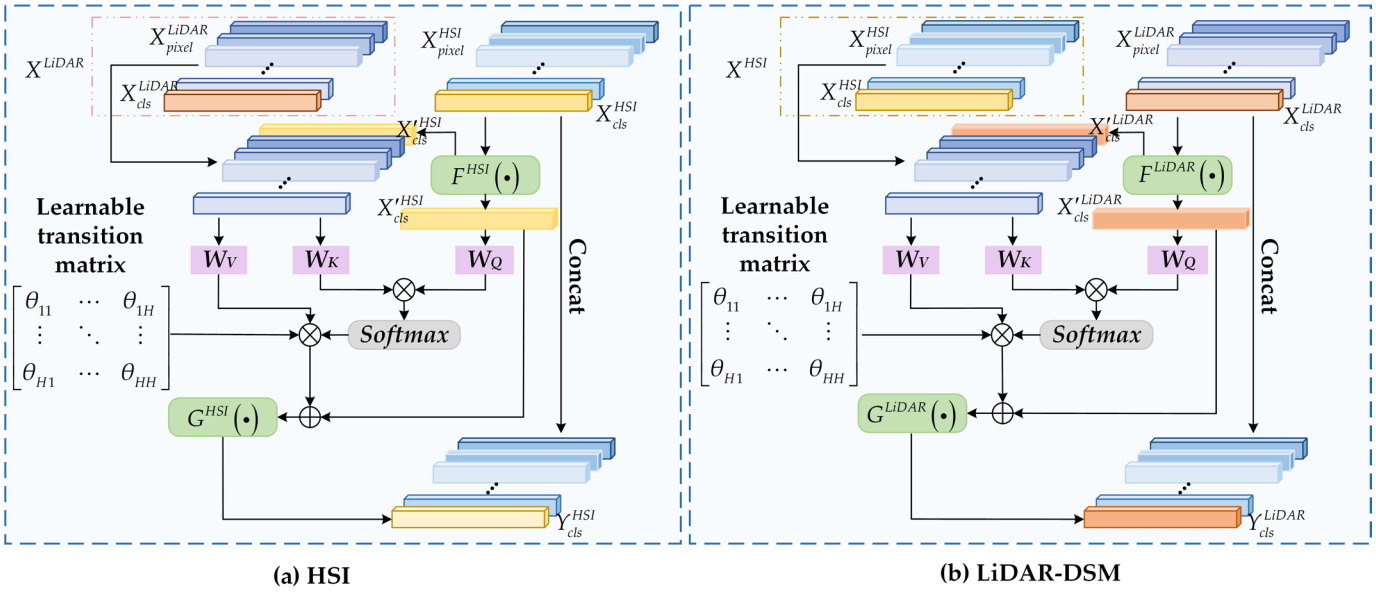


Figure 4. Structure for proposed LC-Attention. (**a**) Fusion encoding module of HSI feature representations; (**b**) fusion encoding module of LiDAR-DSM feature representations.

Then, LC-Attention with the learnable transition matrix $M \in R^{N\times N}$ is used to encode between X'^{HSI}_{cls} and X_L^{HSI}. X'^{HSI}_{cls} is the only query vector for attention operations. Feature fusion representations based on LC-Attention are expressed as follows:

$$\begin{cases} \boldsymbol{Q} = X'^{HSI}_{cls}\boldsymbol{W}_q \\ \boldsymbol{K} = X_L^{HSI}\,\boldsymbol{W}_k \\ \boldsymbol{V} = X_L^{HSI}\,\boldsymbol{W}_v \end{cases}, \tag{10}$$

$$\text{LC}-\text{Attention}\left(X_L^{HSI}\right) = \boldsymbol{M}^{\text{T}}\left(Softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\text{T}}}{\sqrt{C/H}}\right)\right)\boldsymbol{V}, \tag{11}$$

where $\boldsymbol{W}_q$, $\boldsymbol{W}_k$, and $\boldsymbol{W}_v$ are the weight matrices of learning updates, C is the embedded dimension, and H is the number of attention heads.

The time and space complexity of creating the attention diagram is linear because it is only used in the query vector, which makes the entire computation more efficient. Similar to the MHSA, LC-Attention also uses multiple heads, namely MHLCA. After layer norm and residual connection, the formula of LC-Attention is expressed as follows:

$$Y'^{HSI}_{cls} = X'^{HSI}_{cls} + \text{MHLCA}\left(\text{LN}\left(X_L^{HSI}\right)\right), \tag{12}$$

$$Y_{cls}^{HSI} = G^{HSI}\left(Y'^{HSI}_{cls}\right), \tag{13}$$

$$X'^{HSI} = \left[Y_{cls}^{HSI} \cup \left(X^{HSI} - X_{cls}^{HSI}\right)\right], \tag{14}$$

where Y'^{HSI}_{cls} is the class token obtained by learning fusion features. Y'^{HSI}_{cls} is consistent with the class token dimensions of LiDAR-DSM. Y_{cls}^{HSI} indicates a class token with the same dimension as the class token of HSI, which is obtained by linear mapping $G^{HSI}(\cdot)$. At the same time, $G^{HSI}(\cdot)$ is used for dimensional alignment.

The same processing is used for fusion processing in LiDAR-DSM feature representations. The output after fusion is X'^{LiDAR}, and the class token obtained by feature learning is Y_{cls}^{LiDAR}. The new class tokens obtained using feature fusion learning are fed into the classifier for classification.

2.4. Poly Loss

Cross-entropy loss (CE) and focal loss (FC) are the most common choices for training classification networks. However, a good loss function should take a more flexible form for tailoring to different tasks and datasets [36]. For remote sensing datasets, the sample distribution of the same class may be uneven, and some different samples will even overlap. This makes the classification effort more difficult.

Leng et al. [36] proposed poly loss, which decomposed the commonly used classification loss function into a series of weighted polynomial bases through Taylor expansion. CE and FC are decomposed into a series of weighted polynomial bases with polynomial coefficients as the predicted probabilities labeled with class labels. Each polynomial base is weighted by the corresponding polynomial coefficient. Poly loss adjusts the polynomial coefficients for different tasks and datasets, and its formulas are as follows:

$$L_{\mathrm{PC}} = -\log(P_t) + \sum_{j=1}^{N} \varepsilon_j (1 - P_t)^j, \tag{15}$$

$$L_{\mathrm{PF}} = -(1 - P_t)^{\gamma} \log(P_t) + \sum_{j=1}^{N} \varepsilon_j (1 - P_t)^{j+\gamma}, \tag{16}$$

where j represents the power of the polynomial basis and γ represents the power shift of the polynomial term. $\varepsilon_j \in [-1/j, \infty]$ is the perturbation term. It allows us to pinpoint the first N polynomial without worrying about infinitely many higher-order ($j > N + 1$) coefficients. The predicted probability of the model for the target class is shown as P_t. Adjusting the first polynomial term gives the most significant gain, so the poly loss formulas can be reduced to the following:

$$L_{\mathrm{PC}} = -\log(P_t) + \varepsilon_1 (1 - P_t), \tag{17}$$

$$L_{\mathrm{PF}} = -(1 - P_t)^{\gamma} \log(P_t) + \varepsilon_1 (1 - P_t)^{1+\gamma} \tag{18}$$

3. Experimental Results

3.1. Data Description

The performance of the proposed AGMLT method in this paper is evaluated on four public multi-modal datasets: Trento (TR), MUUFL (MU) [37,38], Augsburg (AU), and Houston2013 (HU). Details of all datasets are described as follows.

1. TR

The TR dataset covers a rural area surrounding the city of Trento, Italy. It includes HSI and LiDAR-DSM data with 600 × 166 pixels, and six categories. The HSI has 63 bands in the wavelength range from 420.89 to 989.09 nm. The spectral resolution is 9.2 nm, and the spatial resolution is 1 m. The LiDAR-DSM data consist of a single-channel image containing the altitude of the corresponding ground position, and its image size is the same as that of HSI. The pseudo-color image of HSI, the grayscale image of LiDAR-DSM, and the ground-truth image are shown in Figure 5. The color, class name, training samples, and test samples for the TR dataset are presented in Table 1.

Figure 5. TR dataset. (**a**) Pseudo-color image; (**b**) grayscale image; (**c**) ground-truth image.

Table 1. Details on TR dataset.

No.	Color	Class Name	Training Samples	Test Samples
1		Apple Trees	129	3905
2		Buildings	125	2778
3		Ground	105	374
4		Woods	154	9896
5		Vineyard	184	10,317
6		Roads	122	3052
	Total		819	29,395

2. MU

Both HSI and LiDAR data from the MU dataset were collected in one flight using a flight platform equipped with the CASI-1500 hyperspectral imager and Gemini LiDAR. The MU dataset covers the University of Southern Mississippi Gulf Park Campus, Long Beach, Mississippi, USA. The dataset was acquired in November 2010 with a spatial resolution of 1 m per pixel. The original dataset is 325 × 337 pixels with 72 bands, and the imaging spectral range is between 380 nm and 1050 nm. Due to the influence of imaging noise, the first four and last four bands were removed, and 64 bands were ultimately used. The invalid area on the right of the original image was removed, and the 325 × 220 pixels were retained. A DSM image was generated using LiDAR data, and its spatial resolution was 1 m per pixel. Objects in the imaging scene were labeled into eleven categories. The pseudo-color image of HSI, the grayscale image of LiDAR-DSM, and the ground-truth image are shown in Figure 6. The details of MU dataset are presented in Table 2.

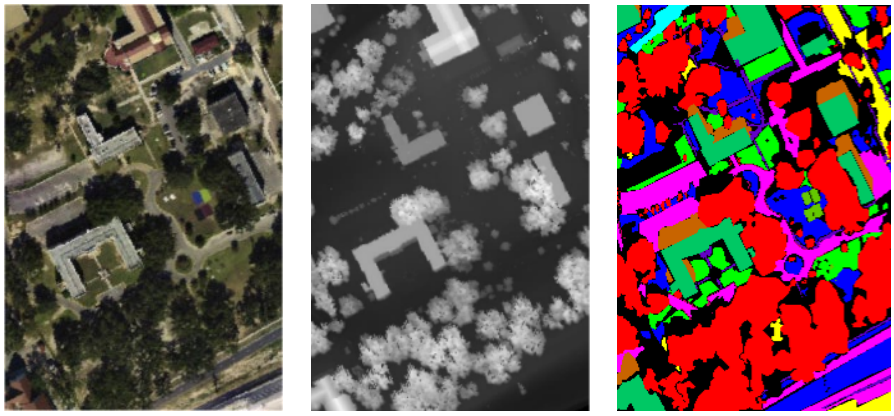


Figure 6. MU dataset. (**a**) Pseudo-color image; (**b**) grayscale image; (**c**) ground-truth image.

Table 2. Details on MU dataset.

No.	Color	Class Name	Training Samples	Test Samples
1		Trees	150	23,096
2		Mostly Grass	150	4120
3		Mixed Ground Surface	150	6732
4		Dirt and Sand	150	1676
5		Road	150	6537
6		Water	150	316
7		Buildings Shadow	150	2083
8		Buildings	150	6090
9		Sidewalk	150	1235
10		Yellow Curb	150	33
11		Cloth Panels	150	119
Total			1650	52,037

3. AU

The AU dataset was captured over the city of Augsburg, Germany. The HSI was obtained using a DAS-EOC HySpex sensor [39], and the LiDAR-DSM data were collected using the DLR-3 K system [40]. The spatial resolutions were down sampled to a unified resolution of 30 m for managing the multi-modal data adequately. The HSI has 180 bands from 0.4 to 2.5 µm, while LiDAR-DSM data have a single raster. The pixel size of AU is 332 × 485, with seven different land cover classes being depicted. The pseudo-color image of HSI, the grayscale image of LiDAR-DSM, and the ground-truth image are shown in Figure 7. Details on the AU dataset are presented in Table 3.

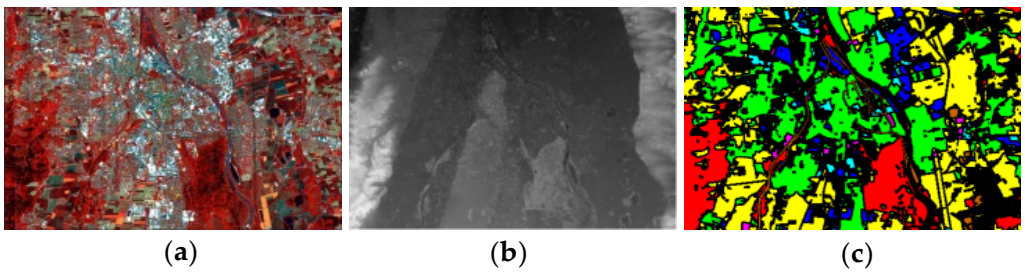


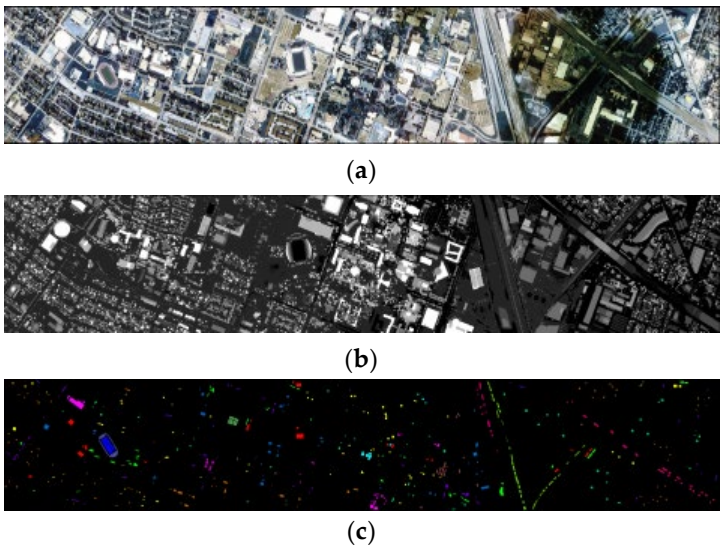
Figure 7. AU dataset. (**a**) Pseudo-color image; (**b**) grayscale image; (**c**) ground-truth image.

Table 3. Details on AU dataset.

No.	Color	Class Name	Training Samples	Test Samples
1		Forest	675	12,832
2		Residential Area	1516	28,813
3		Industrial Area	192	3659
4		Low Plants	1342	25,515
5		Allotment	28	547
6		Commercial Area	82	1563
7		Water	16	1454
	Total		3911	74,383

4. HU

The HU dataset was provided by IEEE GRSS for the 2013 Data Fusion Competition. The scene covers the University of Houston and its surrounding area in Texas, USA. It includes HSI and LiDAR-DSM data with 340 × 1905 pixels, and fifteen categories. The HIS has 144 bands in the wavelength range of 0.38 to 1.05 µm and with a spatial resolution of 2.5 m per pixel. The spatial resolution of LiDAR-DSM data is also 2.5 m per pixel. The pseudo-color image of HIS, the grayscale image of LiDAR-DSM, and the ground-truth image are shown in Figure 8. The color, class name, training samples, and test samples for the HU dataset are shown in Table 4.



(**a**)

(**b**)

(**c**)

Figure 8. HU dataset. (**a**) Pseudo-color image; (**b**) grayscale image; (**c**) ground-truth image.

Table 4. Details on HU dataset.

No.	Color	Class Name	Training Samples	Test Samples
1		Healthy Grass	198	1053
2		Stressed Grass	190	1064
3		Synthetic Grass	192	505
4		Trees	188	1056
5		Soil	186	1056
6		Water	182	143
7		Residential	196	1072
8		Commercial	191	1053

Table 4. *Cont.*

No.	Color	Class Name	Training Samples	Test Samples
9		Road	193	1059
10		Highway	191	1036
11		Railway	181	1054
12		Parking Lot 1	192	1041
13		Parking Lot 2	184	285
14		Tennis Court	181	247
15		Running Track	187	473
	Total		2832	12,197

3.2. Experimental Setting

The experiments related to this paper were conducted on a computer with Windows 11, Intel Core i9 CPU with 32 GB memory, and NVIDIA RTX 3090Ti graphics with 24 GB GPU memory, which were coded with Python 3.8 under pytorch 1.12.0. The sizes of the input images, batch size, and epochs were set to 11 × 11, 64, and 100, respectively. The number of principal components chosen by PCA was set as 30. In order to improve the reliability of the experimental results, training samples and test samples were randomly selected for TR, MU, AU and HU datasets. Since the baseline algorithm in this paper is HCT, the choices of training samples and test samples are consistent with the HCT [29]. Tables 1–4 list the number of training samples and test samples of the four datasets. All experiments were conducted five consecutive times, and the final classification results are average values of the five times. The evaluation index overall accuracy (OA), average accuracy (AA), and statistical kappa coefficient (K), which are commonly used in classification experiments, are chosen as the key evaluation indexes of this paper.

To obtain the best accuracy, it is necessary to compare the experimental results of different experimental parameters. The initial learning rate of Adam, the heads for attention, the depth of encoders and the depth of the Fusion Module are tested on all datasets. The control variable method is used in the experiments, that is, the input size, epochs, experiment times, the number of training samples and test samples are consistent.

3.2.1. Initial Learning Rate

Table 5 shows the influence of initial learning rates for Adam on the experimental results. Initial learning rates of 0.001, 0.0005, and 0.0001 are selected in the experiments. The results show that the best accuracy could be obtained by setting the initial learning rate as 0.0005 on TR and AU datasets, 0.001 on MU dataset, and 0.0001 on HU dataset.

Table 5. OA of different learning rate on each dataset (the bold represents the optimum accuracy).

Datasets	Initial Learning Rate		
	0.001	**0.0005**	**0.0001**
TR	99.66 ± 0.04	**99.72 ± 0.04**	99.58 ± 0.09
MU	**90.16 ± 1.49**	87.44 ± 1.89	87.82 ± 1.03
AU	97.60 ± 0.16	**97.80 ± 0.06**	97.50 ± 0.11
HU	99.65 ± 0.06	99.70 ± 0.05	**99.93 ± 0.02**

3.2.2. Depth and Heads

Figure 9 depicts the synergistic effect of the number of attention heads, the depth of encoders, and the depth of the Fusion Module. The number of heads for L-Attention and LC-Attention are identical, and the depth of each encoder and the Fusion Module are the same. The experiments selected four combinations of 4 + 2, 4 + 1, 8 + 2, and 8 + 1, which

concluded that the best accuracy could be obtained by setting the heads for the attention, and the depth for encoders and the Fusion Module as 4 and 2 on all datasets.

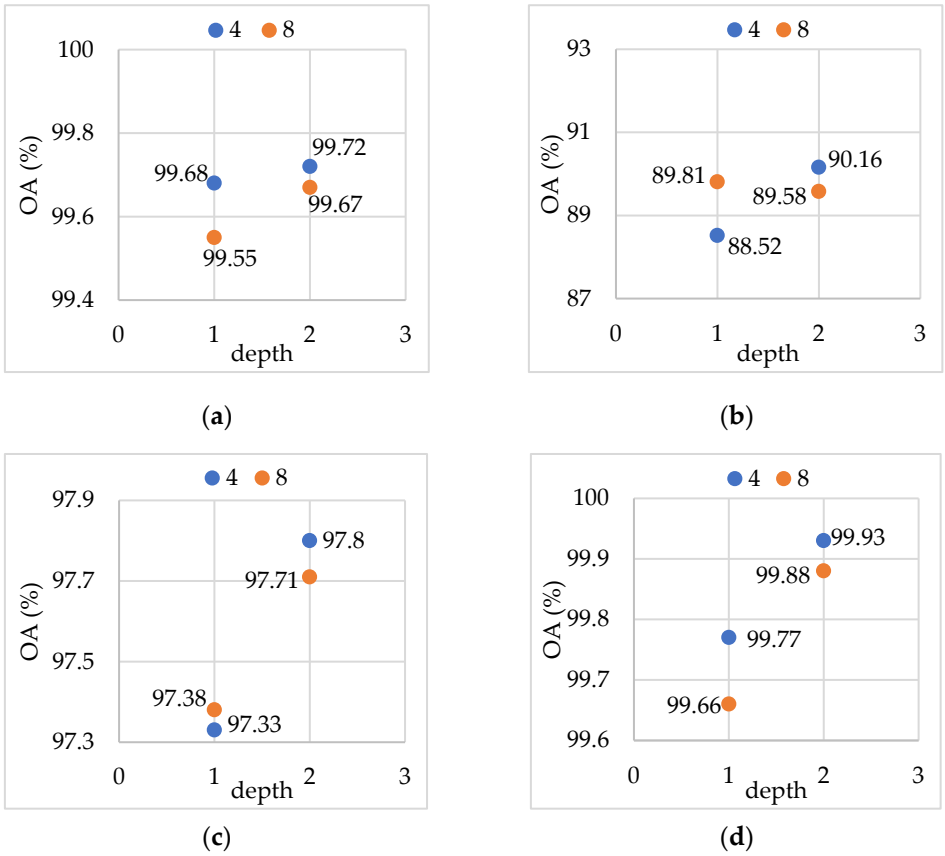


Figure 9. Combined effect of the heads for attention, and the depth for encoders on four datasets. (**a**) TR dataset (4 + 2); (**b**) MU dataset (4 + 2); (**c**) AU dataset (4 + 2); (**d**) HU dataset (4 + 2). The horizontal coordinate is the depth for encoders, and the vertical coordinate is the OA (%) value. The blue circle represents four attention heads, and the orange circle represents eight attention heads.

3.3. Performance Comparison

In this section, the proposed AGMLT is compared with DMCN [41], SpectralFormer [13], SSFTT [15], morpFormer [42], CoupledCNN [32], MFT_PT [28], MFT_CT [28], and HCT [29] for validating the classification performance. The initial learning rates for the baseline HCT are consistent with the original paper, which for the TR and HU are 0.001, for the MU is 0.0001, and for the AU is 0.0005. The depth for the Fusion Encoder of HCT on all datasets is 2, the depth in the transformer encoder and cross-attention is 1, and the attention heads for TR, MU, AU, and HU are 4, 8, 8, and 8 based on the source code. The initial learning rates for DMCN, SpectralFormer, SSFTT, and CoupledCNN are consistent with AGMLT for obtaining optimal performance, and for morpFormer, MFT_PT, and MFT_CT are 0.0005 as in the original papers. The classification results and classification maps on all datasets of the methods are outlined in Section 3.3.1, and Section 3.3.2 shows the comparison of consumption and computational complexity for all methods.

3.3.1. Experimental Results

The classification results of the proposed AGMLT and all the comparison methods are shown in Tables 6–9. It could be seen that the proposed AGMLT achieves the best results on evaluation indicators, with the OA reaching 99.72%, 90.16%, 97.80%, and 99.93%, AA reaching 99.57%, 92.47%, 89.35%, and 99.95%, and K × 100 reaching 99.62%, 87.14%, 96.85%, and 99.93% on the TR, MU, AU and HU datasets, respectively. For evaluating the classification performance.

1. TR dataset

As shown in Table 6, SpectralFormer has the worst classification results, because it directly flattens the image block into the vector, which destroys the internal structure information of the image. Coupled CNN is the second worst because its structure is relatively simple and the ability to extract features is relatively weak. The proposed AGMLT improves 0.37%, 1.73%, 0.54%, 0.70%, 1.33%, 0.61%, 0.27%, and 0.10% on OA compared to DMCN, SpectralFormer, SSFTT, morpFormer, Coupled CNN, MFT_PT, MFT_CT, and HCT. At the same time, the proposed AGMLT improves 0.70%, 3.03%, 0.87%, 1.08%, 2.19%, 0.90%, 0.54%, and 0.26% on AA, and improves 0.49%, 2.31%, 0.72%, 0.93%, 1.77%, 0.81%, 0.36%, and 0.13% on K × 100, respectively. In addition, it could be found that the accuracy of the categories SSFTT, morpFormer, HCT, and the proposed AGMLT reached 100%. The accuracy of the categories SSFTT, morpFormer, and the proposed AGMLT also reached 100%. This is because the distribution of these two samples is simple, which is means that it is easy to learn the feature information. From Figure 10, the salt-and-pepper noise of AGMLT is the least compared to the comparison methods.

Table 6. Classification results of all methods on TR dataset (the bold represents the optimum accuracy).

No.		HSI Input					HSI and LiDAR-DSM Input			
		DMCN	SpectralFormer	SSFTT	morp-Former	Coupled CNN	MFT_PT	MFT_CT	HCT	AGMLT
1	Mean	**99.65**	99.1	98.84	97.89	99.18	97.65	98.2	99.57	99.47
	Std	**0.35**	0.72	0.61	0.75	0.61	0.45	0.44	0.37	0.14
2	Mean	**99.74**	94.49	98.01	96.49	92.92	97.93	98.74	98.85	98.81
	Std	**0.49**	0.39	0.5	2.57	6.24	0.48	0.64	0.28	0.37
3	Mean	99.44	97.54	**100**	**100**	99.68	99.73	98.88	99.41	**100**
	Std	0.56	0.58	**0**	**0**	0.32	0.27	1.12	0.59	**0**
4	Mean	99.99	99.92	**100**	**100**	99.96	99.91	99.99	**100**	**100**
	Std	0.01	0.08	**0**	**0**	0.04	0.09	0.01	**0**	**0**
5	Mean	99.97	99.65	**99.99**	99.97	99.84	99.92	99.96	**99.99**	99.97
	Std	0.03	0.23	**0.01**	0.02	0.16	0.08	0.04	**0.01**	0.02
6	Mean	96.42	88.51	95.38	96.58	92.71	96.87	98.38	98.01	**99.14**
	Std	1.12	5.55	2.23	2.84	5.06	1.46	0.9	0.98	**0.2**
OA (%)	Mean	99.35	97.99	99.18	99.02	98.39	99.11	99.45	99.62	**99.72**
	Std	0.17	0.64	0.12	0.28	1.28	0.19	0.1	0.14	**0.04**
AA (%)	Mean	98.87	96.54	98.7	98.49	97.38	98.67	99.03	99.31	**99.57**
	Std	0.35	0.51	0.22	0.42	1.94	0.3	0.32	0.32	**0.07**
K × 100	Mean	99.13	97.31	98.9	98.69	97.85	98.81	99.26	99.49	**99.62**
	Std	0.58	0.49	0.17	0.38	1.72	0.12	0.14	0.18	**0.05**

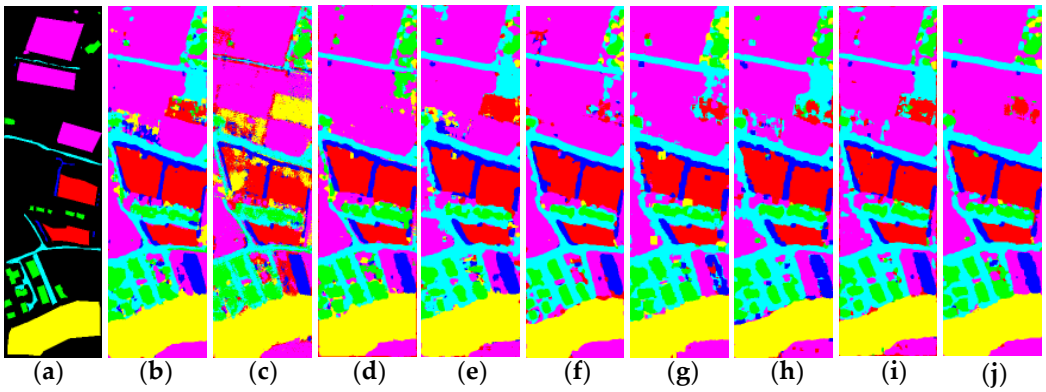


Figure 10. Classification images of different methods on TR. (**a**) Ground-truth image; (**b**) DMCN (99.35%); (**c**) SpectralFormer (97.99%); (**d**) SSFTT (98.18%); (**e**) morpFormer (99.02%); (**f**) Coupled CNN (98.39%); (**g**) MFT_PT (99.11%); (**h**) MFT_CT (99.45%); (**i**) HCT (99.62%); (**j**) AGMLT (99.72%).

2. MU dataset

As shown in Table 7, Coupled CNN has the worst classification results, and MFT_PT is the second worst. This is because MFT_PT only carries out convolutional feature extraction on HSI. The OA of the proposed AGMLT increased by 2.77%, 3.08%, 3.10%, 5.20%, 6.49%, 5.83%, 5.35%, and 2.22% compared to DMCN, SpectralFormer, SSFTT, morpFormer, Coupled CNN, MFT_PT, MFT_CT, and HCT. Meanwhile, the AA increased by 2.38%, 3.17%, 3.29%, 4.90%, 5.21%, 5.70%,5.28%, and 3.11%, and K × 100 increased by 3.54%, 4.00%, 3.95%, 6.53%, 8.28%, 7.38%, 6.77%, and 2.90%, respectively. The uneven and complex sample distribution of the MU dataset presents a significant challenge for classification accuracy across various methods. The AGMLT stands out due to its ability to harness rich dynamic feature information, resulting in a superior classification effect compared to other algorithms. This advantage is likely attributed to the sophisticated design of AGMLT, enabling it to effectively harness the complexities of sample distribution for the MU dataset. From Figure 11, the classification image of AGMLT is closest to the ground-truth image.

Table 7. Classification results of all methods on MU dataset (the bold represents the optimum accuracy).

No.		HSI Input				HSI and LiDAR-DSM Input				
		DMCN	SpectralFormer	SSFTT	morp-Former	Coupled CNN	MFT_PT	MFT_CT	HCT	AGMLT
1	Mean	87.76	88.62	88.16	85.14	86.29	86.42	86.26	90.04	**90.52**
	Std	2.37	0.36	0.57	2.26	0.78	1.22	2.91	3.34	**2.43**
2	Mean	84.85	78.01	84.27	79.49	87.09	81.96	77.81	82.84	**90.56**
	Std	6.81	9.75	9.82	6.40	2.12	3.20	13.68	1.45	**1.81**
3	Mean	78.90	81.75	79.53	81.83	76.96	77.24	79.82	77.69	**82.46**
	Std	3.35	8.58	3.86	2.22	1.58	0.99	2.12	3.65	**1.23**
4	Mean	96.42	94.88	93.89	96.30	94.93	92.79	92.96	94.44	**96.76**
	Std	1.54	2.49	7.73	0.65	2.56	1.91	1.96	2.74	**0.73**
5	Mean	88.05	88.62	84.34	79.83	77.89	79.12	78.89	86.28	**89.69**
	Std	3.91	0.36	3.17	5.17	3.72	1.07	2.45	2.47	**1.63**
6	Mean	99.84	99.43	99.68	99.56	99.84	99.24	99.24	99.40	**99.87**
	Std	0.16	0.57	0.32	0.38	0.19	0.76	0.76	0.60	**0.15**
7	Mean	92.44	91.38	94.30	90.16	92.06	91.22	91.54	92.99	**95.10**
	Std	3.04	2.04	2.57	2.72	0.96	2.59	3.61	2.98	**1.74**
8	Mean	94.56	92.28	93.03	92.82	77.03	90.24	93.18	94.27	**94.62**
	Std	2.32	0.85	1.47	2.00	8.71	1.98	2.58	1.28	**2.76**
9	Mean	75.45	76.79	78.93	76.16	75.30	67.32	70.99	75.67	**83.61**
	Std	3.57	0.93	1.39	6.12	6.98	6.02	5.25	3.28	**0.66**
10	Mean	**94.24**	93.94	86.68	83.03	92.12	87.88	90.30	95.0	93.93
	Std	**5.76**	6.06	10.92	10.43	1.21	12.12	0.61	1.31	9.38
11	Mean	99.24	99.50	98.32	98.99	97.82	98.99	98.15	90.00	**100**
	Std	0.76	0.52	1.68	0.34	2.18	1.01	1.85	9.00	**0.00**
OA (%)	Mean	87.39	87.08	87.06	84.96	83.67	84.33	84.81	87.94	**90.16**
	Std	1.12	1.24	0.85	1.10	1.46	0.76	1.34	0.48	**1.49**
AA (%)	Mean	90.09	89.30	89.18	87.57	87.26	86.77	87.19	89.36	**92.47**
	Std	0.99	1.12	1.64	0.80	1.64	1.52	0.60	1.26	**1.33**
K × 100	Mean	83.60	83.14	83.19	80.61	78.86	79.76	80.37	84.24	**87.14**
	Std	0.21	1.64	0.43	1.31	0.33	0.93	1.59	1.55	**1.86**

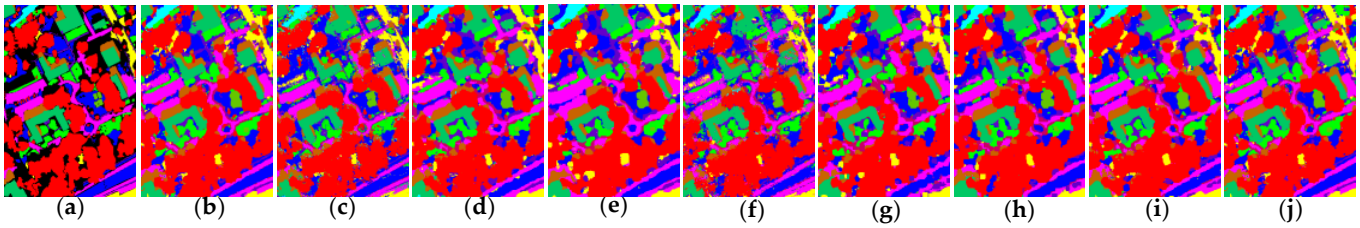


Figure 11. Classification images of different methods on MU. (**a**) Ground-truth Image; (**b**) DMCN (87.39%); (**c**) SpectralFormer (87.08%); (**d**) SSFTT (87.06%); (**e**) morpFormer (84.96%); (**f**) Coupled CNN (83.67%); (**g**) MFT_PT (84.33%); (**h**) MFT_CT (84.81%); (**i**) HCT (87.94%); (**j**) AGMLT (90.16%).

3. AU dataset

As seen in Table 8, similar to the TR dataset, SpectralFormer has the worst classification results, and Coupled CNN is the second worst. The OA of the proposed AGMLT increased by 1.56%, 3.91%, 0.72%, 0.95%, 2.79%, 1.45%, 1.28%, and 0.86% compared to DMCN, SpectralFormer, SSFTT, morpFormer, Coupled CNN, MFT_PT, MFT_CT, and HCT. Simultaneously, the AA of the proposed method increased by 8.32%, 17.69%, 3.23%, 1.32%, 7.91%, 3.74%, 2.46%, and 3.01%, and K × 100 increased by 2.25%, 5.63%, 1.04%, 1.37%, 4.06%, 2.07%, 1.83%, and 1.24%, respectively. From Figure 12, for the proposed AGMLT, the salt-and-pepper noise is the least compared to the comparison methods.

Table 8. Classification results of all methods on AU dataset (the bold represents the optimum accuracy).

No.		HSI Input				HSI and LiDAR-DSM Input				
		DMCN	SpectralFormer	SSFTT	morp-Former	Coupled CNN	MFT_PT	MFT_CT	HCT	AGMLT
1	Mean	98.59	86.10	98.82	97.71	89.59	98.38	98.29	98.75	**99.31**
	Std	0.56	0.44	0.08	0.21	6.02	0.53	1.31	0.49	**0.20**
2	Mean	98.52	96.10	99.02	98.54	98.55	98.20	98.14	98.66	**99.10**
	Std	0.44	1.44	0.33	0.25	0.61	0.26	2.86	0.41	**0.18**
3	Mean	87.64	75.99	90.13	89.69	87.65	89.24	88.60	88.45	**93.10**
	Std	1.51	8.92	1.39	1.46	1.39	2.23	1.20	2.78	**2.23**
4	Mean	99.02	98.66	98.77	98.53	**99.39**	97.88	98.37	98.93	99.29
	Std	0.58	0.34	0.34	0.11	**0.26**	0.28	0.35	0.21	0.12
5	Mean	71.08	48.88	79.09	84.88	75.54	78.43	86.18	81.08	**87.09**
	Std	3.99	7.61	5.60	3.06	7.72	0.36	8.12	7.95	**5.21**
6	Mean	47.82	27.56	70.12	75.45	58.62	70.68	71.17	69.00	**76.69**
	Std	5.15	9.54	3.37	3.58	9.20	3.28	2.02	1.26	**3.59**
7	Mean	64.51	55.50	66.88	**71.36**	60.73	66.41	67.52	69.52	70.85
	Std	1.86	4.95	1.20	**3.58**	1.52	6.30	4.04	4.05	1.95
OA (%)	Mean	96.24	93.89	97.08	96.85	95.01	96.35	96.52	96.94	**97.80**
	Std	1.36	0.27	0.18	0.07	1.31	0.24	0.31	0.33	**0.06**
AA (%)	Mean	81.03	71.66	86.12	88.03	81.44	85.61	86.89	86.34	**89.35**
	Std	2.30	2.58	1.93	1.21	2.86	1.11	1.25	1.51	**0.92**
K × 100	Mean	94.60	91.22	95.81	95.48	92.79	94.78	95.02	95.61	**96.85**
	Std	0.42	0.43	0.25	0.10	1.91	0.34	0.44	0.47	**0.08**

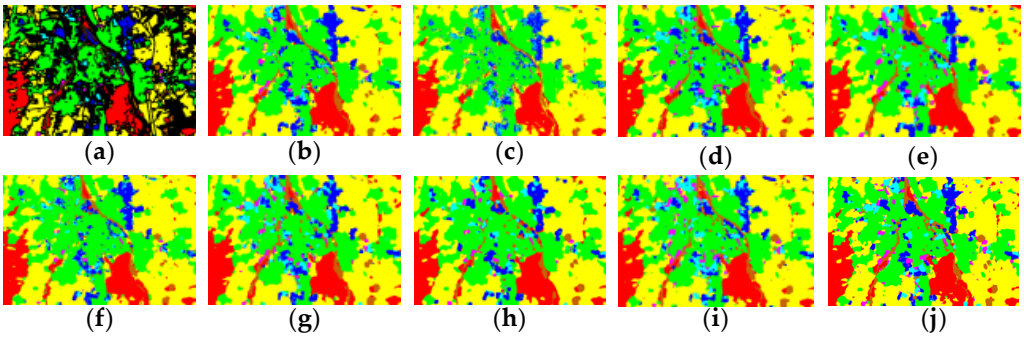


Figure 12. Classification images of different methods on AU. (**a**) Ground-truth Image; (**b**) DMCN (96.24%); (**c**) SpectralFormer (93.89%); (**d**) SSFTT (97.08%); (**e**) morpFormer (96.85%); (**f**) Coupled CNN (95.01%); (**g**) MFT_PT (96.35%); (**h**) MFT_CT (96.52%); (**i**) HCT (96.94%); (**j**) AGMLT (97.80%).

4. HU dataset

As shown in Table 9, Coupled CNN has the worst classification results, and DMCN has the second worst. The proposed AGMLT increased by 1.09%, 1.04%, 0.20%, 0.57%, 1.39%, 0.33%, 0.47%, and 0.20% on OA compared to DMCN, SpectralFormer, SSFTT, morpFormer, Coupled CNN, MFT_PT, MFT_CT, and HCT. The value of AA increased by 0.90%, 1.05%, 0.16%, 0.52%, 1.10%, 0.27%, 0.40%, and 0.17%, and the value of K × 100 increased by 1.19%, 1.13%, 0.22%, 0.63%, 1.52%, 0.36%, 0.52%, and 0.23%, respectively. In addition, DMCN and SpectralFormer have similar classification performance on the HU datasets. Simultaneously, SSFTT and HCT have similar classification performance. From Figure 13,

the higher classification accuracy leads to less salt-and-pepper noise. This indicates that AGMLT effectively enhances the joint classification performance.

Table 9. Classification results of all methods on HU dataset (the bold represents the optimum accuracy).

No.		HSI Input				HSI and LiDAR-DSM Input				
		DMCN	SpectralFormer	SSFTT	morp-Former	Coupled CNN	MFT_PT	MFT_CT	HCT	AGMLT
1	Mean	98.35	99.34	99.74	99.18	**99.91**	99.32	98.94	98.77	99.81
	Std	0.80	0.66	0.26	0.36	**0.09**	0.68	0.97	1.05	0.08
2	Mean	98.54	98.89	**99.91**	99.19	99.94	99.53	99.49	99.70	99.84
	Std	3.24	1.11	**0.09**	0.32	0.06	0.85	0.51	0.30	0.16
3	Mean	98.05	**100**	99.96	99.47	99.92	99.76	99.88	99.92	**100**
	Std	2.88	**0.00**	0.04	0.47	0.08	0.24	0.12	0.08	**0.00**
4	Mean	98.74	99.72	99.66	99.56	94.56	94.43	98.28	99.56	**100**
	Std	0.98	0.28	0.15	0.16	5.15	0.57	1.72	0.35	**0.00**
5	Mean	**100**	99.39	99.92	**100**	**100**	**100**	**100**	**100**	**100**
	Std	**0.00**	0.61	0.08	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
6	Mean	96.89	99.30	**100**	**100**	**100**	**100**	**100**	**100**	**100**
	Std	3.11	0.70	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
7	Mean	96.05	98.06	99.63	98.88	99.06	99.31	99.85	99.74	**100**
	Std	1.37	1.94	0.38	0.65	0.75	0.69	0.15	0.26	**0.00**
8	Mean	94.60	98.33	99.44	98.35	96.81	99.55	99.47	99.87	**100**
	Std	4.45	1.42	0.10	0.25	1.28	0.45	0.53	0.13	**0.00**
9	Mean	94.34	96.20	99.68	98.65	96.94	99.09	99.13	99.23	**100**
	Std	5.52	2.19	0.32	1.45	2.02	0.91	0.87	0.37	**0.00**
10	Mean	99.83	99.83	99.77	99.94	99.83	99.81	99.98	99.98	**100**
	Std	0.17	0.17	0.23	0.09	0.17	0.19	0.02	0.02	**0.00**
11	Mean	99.31	99.48	99.79	100	99.28	99.72	99.49	99.98	**100**
	Std	0.69	0.32	0.21	0.00	0.57	0.28	0.51	0.02	**0.00**
12	Mean	97.14	99.27	99.63	99.46	99.06	99.81	99.29	**99.67**	99.57
	Std	2.51	0.23	0.37	0.20	0.56	0.19	0.23	**0.33**	0.04
13	Mean	94.03	95.99	99.93	98.71	99.65	99.86	99.58	99.72	**100**
	Std	5.96	3.64	0.07	1.82	0.35	0.14	0.42	0.28	**0.00**
14	Mean	99.90	99.92	**100**	**100**	**100**	**100**	**100**	**100**	**100**
	Std	0.10	0.08	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
15	Mean	**100**	99.83	**100**	**100**	**100**	**100**	99.92	**100**	**100**
	Std	**0.00**	0.17	**0.00**	**0.00**	**0.00**	**0.00**	0.08	**0.00**	**0.00**
OA (%)	Mean	98.84	98.89	99.73	99.36	98.54	99.60	99.46	99.73	**99.93**
	Std	0.29	0.70	0.14	0.24	0.49	0.15	0.29	0.16	**0.02**
AA (%)	Mean	99.05	98.90	99.79	99.43	98.85	99.68	99.55	99.78	**99.95**
	Std	0.38	0.45	0.11	0.29	0.31	0.13	0.24	0.22	**0.01**
K × 100	Mean	98.74	98.80	99.71	99.30	98.41	99.57	99.41	99.70	**99.93**
	Std	0.31	0.32	0.16	0.26	0.53	0.16	0.32	0.16	**0.02**

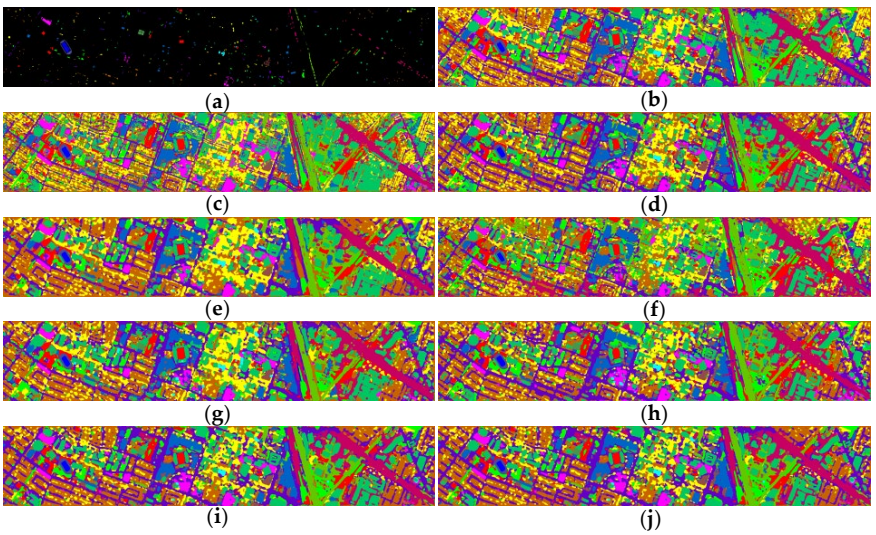


Figure 13. Classification images of different methods on HU. (**a**) Ground-truth image; (**b**) DMCN (98.84%); (**c**) SpectralFormer (98.89%); (**d**) SSFTT (99.73%); (**e**) morpFormer (99.36%); (**f**) Coupled CNN (98.54%); (**g**) MFT_PT (99.60%); (**h**) MFT_CT (99.46%); (**i**) HCT (99.73%); (**j**) AGMLT (99.93%).

3.3.2. Consumption and Computational Complexity

To comprehensively compare the AGMLT with the comparison methods, the total parameters (TPs), training time (Tr), test time (Te) and Flops of all methods are tested in this section. The results are presented in Table 10. Since the data are filled in the convolution part to align the feature sizes, the number of parameters and the complexity of the model are increased, while the learnable features are added to improve the classification accuracy.

Table 10. Consumption and computational complexity of each dataset (the bold represents the optimum accuracy).

Methods	TPs	Tr (s)	Te (s)	Flops	OA (%)	TPs	Tr (s)	Te (s)	Flops	OA (%)
			TR					**MU**		
DMCN	2.77 M	20.22	1.69	3.21 G	99.35 ± 0.17	2.77 M	34.40	3.04	3.21 G	87.39 ± 1.12
SpectralFormer	97.33 K	46.80	3.55	192.68 M	97.99 ± 0.64	97.65 K	93.22	6.22	192.70 M	87.08 ± 1.24
SSFTT	147.84 K	22.08	1.51	447.18 M	99.18 ± 0.12	148.16 K	38.06	2.78	447.20 M	87.06 ± 0.85
morpFormer	62.56 K	38.36	4.38	334.43 M	99.02 ± 0.28	62.56 K	77.67	7.11	334.43 M	84.96 ± 1.10
CoupledCNN	104.18 K	7.68	0.78	169.08 M	98.39 ± 1.28	106.11 K	18.47	1.38	169.20 M	83.67 ± 1.46
MFT_PT	221.29 K	58.50	7.98	312.91 M	99.11 ± 0.19	221.61 K	115.80	14.10	312.93 M	84.33 ± 0.76
MFT_CT	221.29 K	82.33	11.60	312.91 M	99.45 ± 0.10	221.61 K	163.87	20.39	312.93 M	84.81 ± 1.34
HCT	465.62 K	14.53	1.28	519.16 M	99.62 ± 0.14	728.09 K	26.84	2.27	569.55 M	87.94 ± 0.48
AGMLT	837.08 K	50.44	3.97	4.91 G	**99.72 ± 0.04**	837.40 K	120.48	9.55	4.91 G	**90.16 ± 1.49**
Methods			**AU**					**HU**		
DMCN	2.77 M	76.96	3.82	3.21 G	96.24 ± 1.36	2.78 M	23.49	0.93	3.21 G	98.84 ± 0.29
SpectralFormer	97.39 K	202.32	8.03	192.68 M	93.89 ± 0.27	97.91 K	153.84	1.43	192.71 M	98.89 ± 0.70
SSFTT	147.90 K	93.01	3.97	447.18 M	97.08 ± 0.18	148.42 K	28.37	0.37	447.22 M	99.73 ± 0.14
morpFormer	62.56 K	185.38	10.22	334.43 M	96.85 ± 0.07	62.56 K	134.35	1.85	334.43 M	99.36 ± 0.24
CoupledCNN	104.57 K	37.86	2.03	169.11 M	95.01 ± 1.31	107.66 K	27.98	0.37	169.30 M	98.54 ± 0.49
MFT_PT	221.35 K	272.02	20.03	312.91 M	96.35 ± 0.24	221.87 K	195.11	3.32	312.95 M	99.60 ± 0.15
MFT_CT	221.35 K	397.32	29.77	312.91 M	96.52 ± 0.31	221.87 K	332.68	5.50	312.95 M	99.46 ± 0.29
HCT	727.83 K	60.74	3.42	569.52 M	96.94 ± 0.33	728.35 K	58.33	0.87	569.58 M	99.73 ± 0.16
AGMLT	837.14 K	258.56	12.43	4.91 G	**97.80 ± 0.06**	837.66 K	170.65	1.67	4.91 G	**99.93 ± 0.02**

The settings of experiments are the same as previously mentioned. The AGMLT has fewer total parameters than DMCN but a longer running time and larger Flops, and the AGMLT has a shorter running time than MFT_PT and MFT_CT but more total parameters and larger Flops. However, compared with SpectralFormer and SSFTT, the AGMLT has more total parameters, larger Flops, and a longer running time. Taking the TR and HU datasets as examples, the AGMLT has a shorter test time than morpFormer, but more total parameters and Flops, and a longer training time. Taking the MU and AU datasets as examples, the AGMLT has more total parameters, larger Flops, and a longer running time than morpFormer. Furthermore, compared with HCT, the AGMLT has more Flops and a longer running time, and it has more total parameters on the TR dataset and fewer total parameters than other datasets. Finally, the classification performance of AGMLT is optimal.

4. Discussion

4.1. Ablation Analysis

This section takes the TR dataset as an example to conduct ablation experiments to verify the effectiveness of different components. The first column in Table 11 is the convolutional feature extraction module SSAGM shown in Figure 1, and its specific ablation experiments are shown in Table 12. The second column of Table 11 is the L-Former shown in Figure 1, where LS and LTM represent the layer scale and learnable transition matrix in Figure 3, respectively. The third column in Table 11 is the cross-attention with the learnable transition matrix. As outlined in the table, the classification accuracy of the AGMLT proposed in this paper is the best. Each component plays a positive role in improving classification accuracy.

Table 11. Ablation experiments of each component (The √ represents that use current component, and the bold represents the optimum accuracy).

SSAGM	L-Former		LC-Attention	OA (%)	AA (%)	K × 100
	LS	LTM				
	√			99.67 ± 0.03	99.49 ± 0.04	99.56 ± 0.04
		√		99.63 ± 0.01	99.38 ± 0.02	99.50 ± 0.01
√	√		√	99.34 ± 0.08	98.87 ± 0.16	99.11 ± 0.11
√		√	√	99.55 ± 0.09	99.31 ± 0.14	99.40 ± 0.11
	√			99.62 ± 0.13	99.37 ± 0.22	99.49 ± 0.18
√				99.41 ± 0.04	98.95 ± 0.17	99.21 ± 0.06
			√	99.68 ± 0.02	99.46± 0.02	99.57 ± 0.02
√	√			99.43 ± 0.03	98.98 ± 0.13	99.24 ± 0.05
√			√	99.50 ± 0.09	99.12 ± 0.14	99.32 ± 0.12
	√		√	99.46 ± 0.08	99.14 ± 0.13	99.28 ± 0.11
√	√		√	**99.72 ± 0.04**	**99.57 ± 0.07**	**99.62 ± 0.05**

Table 12. Different combinations of PDWA and ADWA (The √ represents that use current component, and the bold represents the optimum accuracy).

PDWA	ADWA(H)	ADWA(L)	OA (%)	AA (%)	K×100
√			99.57 ± 0.03	99.34 ± 0.05	99.43 ± 0.04
	√		99.38 ± 0.06	99.04 ± 0.07	99.17 ± 0.07
		√	99.63 ± 0.15	99.42 ± 0.24	99.50 ± 0.20
√	√		99.36 ± 0.14	98.61 ± 0.20	99.14 ± 0.18
√		√	99.61 ± 0.05	99.40 ± 0.07	99.48 ± 0.07
	√	√	99.51 ± 0.03	99.24 ± 0.05	99.34 ± 0.03
√	√	√	**99.72 ± 0.04**	**99.57 ± 0.07**	**99.62 ± 0.05**

Detailed ablation experiments on the convolutional feature extraction are presented in Table 12. PDWA mainly extracts spectral features of HSI. ADWA(H) is the spatial feature extraction of HSI, while ADWA(L) is the spatial feature extraction of LiDAR-DSM data. In this paper, different combinations of the three attention modules were verified with experiments. Finally, it obtained the best combination and use order, which achieved the best classification effect.

Table 13 shows the effect of asymmetric convolution kernels on the AGMLT. The asymmetric convolution kernel can improve classification accuracy while reducing the number of parameters and Flops of the model. Because the 3D convolution kernel can be divided into many two-dimensional convolution kernels, when the rank of a 2D kernel is 1, it can be equivalent to a series of one-dimensional convolutions, which can strengthen the nuclear skeleton of the CNN while reducing the parameters.

Table 13. The effect of asymmetric convolution for AGMLT (the bold represents the optimum accuracy).

	OA (%)	AA (%)	K × 100	Total Params	Flops
No Asymmetric Convolution	99.62 ± 0.08	99.01 ± 0.14	99.50 ± 0.10	904.71 K	5.39 G
With Asymmetric Convolution	**99.72 ± 0.04**	**99.57 ± 0.07**	**99.62 ± 0.05**	**837.08 K**	**4.91 G**

In this paper, we compare the classification accuracies of HSI or LiDAR-DSM alone and the combination of the two data. As indicated in Table 14, by fusing the HSI and LiDAR-DSM, it is possible to achieve a more accurate and robust classification outcome than would be possible using either source of data alone. HSI can provide rich spectral information, and LiDAR-DSM can supplement accurate orientation and distance information.

Table 14. Ablation analysis of different inputs (the bold represents the optimum accuracy).

Inputs	OA (%)	AA (%)	K×100	OA (%)	AA (%)	K × 100
		TR			MU	
HSI	99.32 ± 0.03	98.95 ± 0.05	99.09 ± 0.04	89.33 ± 0.92	91.83 ± 1.20	86.09 ± 1.19
LiDAR-DSM	97.81 ± 0.64	96.55 ± 1.22	97.06 ± 0.87	68.11 ± 1.61	67.26 ± 5.39	59.55 ± 1.87
HSI + LiDAR-DSM	**99.72 ± 0.04**	**99.57 ± 0.07**	**99.62 ± 0.05**	**90.16 ± 1.49**	**92.47 ± 1.33**	**87.14 ± 1.86**
Inputs		**AU**			**HU**	
HSI	97.45 ± 0.19	89.17 ± 1.21	96.35 ± 0.27	99.76 ± 0.05	99.80 ± 0.05	99.73 ± 0.06
LiDAR-DSM	95.62 ± 1.07	95.62 ± 1.07	95.62 ± 1.07	95.62 ± 1.07	95.62 ± 1.07	95.62 ± 1.07
HSI + LiDAR-DSM	**97.80 ± 0.06**	**89.35 ± 0.92**	**96.85 ± 0.08**	**99.93 ± 0.02**	**99.95 ± 0.01**	**99.93 ± 0.02**

4.2. Loss Functions

AGMLT with different loss functions compared on four multi-modal datasets in this section. L_{CE} stands for cross entropy loss, L_{FC} stands for focal loss, L_{PC} stands for poly loss—CE, and L_{PF} stands for poly loss—focal. As shown in Table 15, L_{PF}, with the best effect, was selected as the loss function of AGMLT.

Table 15. Experimental results using different loss functions (the bold represents the optimum accuracy).

Loss Functions	OA (%)	AA (%)	K × 100	OA (%)	AA (%)	K × 100
		TR			MU	
L_{CE}	99.69 ± 0.05	99.49 ± 0.11	99.58 ± 0.06	89.92 ± 0.77	92.84 ± 0.45	86.84 ± 0.97
L_{FC}	99.69 ± 0.09	99.54 ± 0.13	99.59 ± 0.11	90.09 ± 0.29	92.09 ± 0.39	87.07 ± 0.37
L_{PC}	99.61 ± 0.05	98.99 ± 0.08	99.48 ± 0.06	89.92 ± 0.40	92.47 ± 0.70	86.81 ± 0.51
$\boldsymbol{L_{PF}}$	**99.72 ± 0.04**	**99.57 ± 0.07**	**99.62 ± 0.05**	**90.16 ± 1.49**	**92.47 ± 1.33**	**87.14 ± 1.86**
Loss Functions		**AU**			**HU**	
L_{CE}	97.49 ± 0.27	88.34 ± 0.36	96.41 ± 0.39	99.86 ± 0.05	99.89 ± 0.04	99.85 ± 0.05
L_{FC}	97.63 ± 0.28	88.26 ± 1.37	96.61 ± 0.40	99.75 ± 0.05	99.79 ± 0.04	99.73 ± 0.04
L_{PC}	97.38 ± 0.25	88.42 ± 1.14	96.25 ± 0.36	99.79 ± 0.05	99.75 ± 0.03	99.78 ± 0.05
$\boldsymbol{L_{PF}}$	**97.80 ± 0.06**	**89.35 ± 0.92**	**96.85 ± 0.08**	**99.93 ± 0.02**	**99.95 ± 0.01**	**99.93 ± 0.02**

4.3. Training Percentage

In this section, experiments were conducted to analyze the performance of the proposed AGMLT under different training percentages. The experimental settings are the same as above. The results are shown in Figure 14.

For TR, AU, and HU datasets, 2%, 4%, 6%, and 8% of the total samples are selected as training samples. However, the sample distribution of the MU dataset is particularly uneven, so 5%, 10%, 15%, and 20% of the total samples are selected for training. Experiments have shown that the accuracies of all methods have been significantly improved when the training samples increased. Notably, the AGMLT model exhibited superior performance compared to other methods in all cases, with a particularly notable improvement in accuracy for the MU dataset. It is attributed to the rich learnable features of AGMLT, which adapt more effectively to uneven distributions and improve accuracy. Moreover, the effectiveness of AGMLT across diverse datasets suggests its potential for wide applicability in tasks involving multi-modal data fusion and classification.

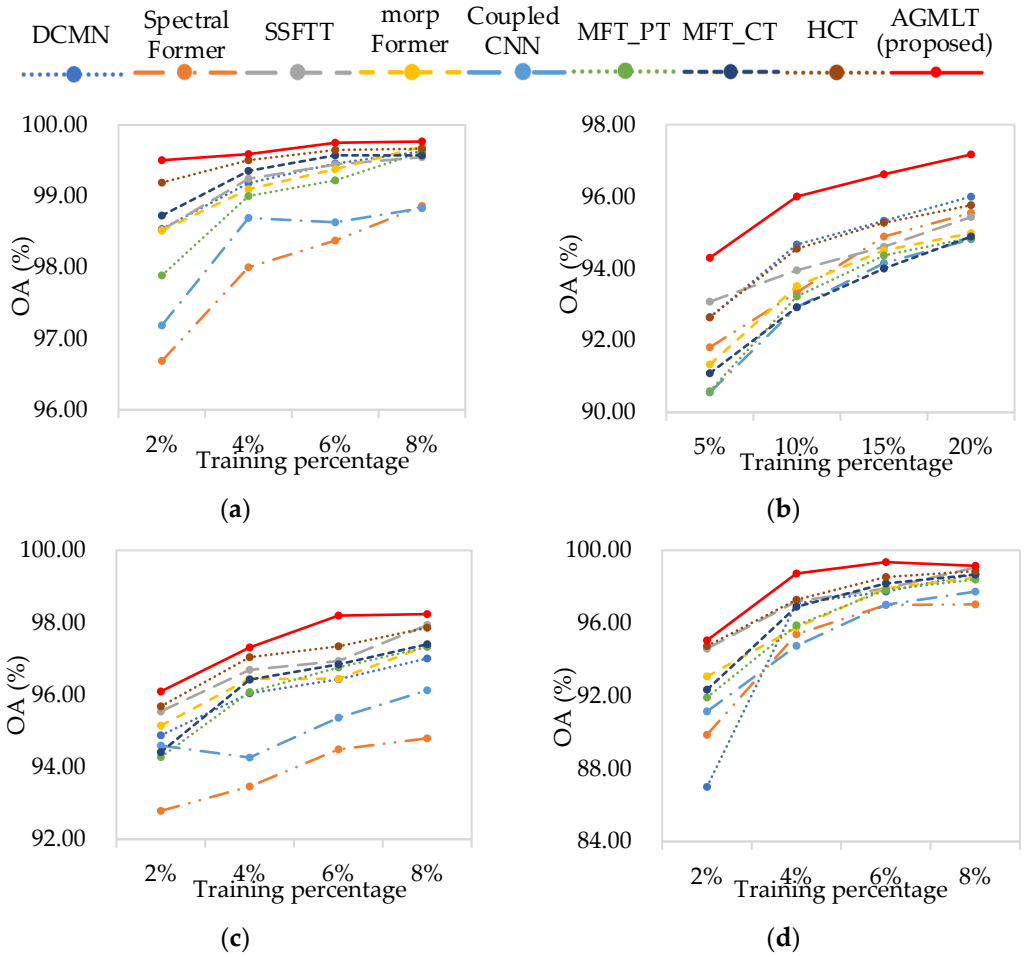


Figure 14. Classification results of different training percentages. (**a**) TR dataset; (**b**) MU dataset; (**c**) AU dataset; (**d**) HU dataset. The accuracies of all methods have been significantly improved when the training samples increased. The AGMLT model exhibited superior performance in all cases.

5. Conclusions

In the study, an adaptive learning model named AGMLT is proposed. Firstly, SSAGM was used to extract local information, which mainly included PDWA and ADWA. The PDWA could extract the spectral information of HSI. The ADWA could extract the spatial information of HSI and the elevation information of LIDAR-DSM. Then, by adding a layer scale and learnable transition matrix to the primary transformer encoder and SA, the data dynamics were improved, and the influence of transformer depth on model classification performance was alleviated. Next, the learnable transfer matrix in LC-Attention enriched the feature information of multi-modal data fusion. Finally, the poly loss training model could adapt to different data. A large number of experiments of AGMLT were carried out to verify the effectiveness and its components.

The data padding in SSAGM increases model complexity and parameters. Therefore, the future scientific research task is designing a precise yet lightweight model. We plan to remove the data padding in SSAGM to shorten the semantic sequence for the subsequent transformer encoder. We propose a multi-scale dynamic gating mechanism combining asymmetric and depthwise separable convolutions to maintain classification performance. The effectiveness of the idea needs to be assessed in future research.

Author Contributions: Conceptualization, M.W., Y.S. and J.X.; methodology, software, validation, writing—original draft, M.W., Y.S. and R.S.; writing—review and editing, M.W., Y.S. and Y.Z.; supervision, Y.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Fundamental Research Funds for the Central Universities, grant number 3072022CF0801, the National Key R&D Program of China, grant number 2018YFE0206500, and the National Key Laboratory of Communication Anti Jamming Technology, grant number 614210202030217.

Data Availability Statement: The MUUFL dataset is at https://github.com/GatorSense/MUUFLGulfport/, the Trento and Augsburg datasets are available at https://github.com/AnkurDeria/MFT?tab=readme-ov-file, and the University of Pavia dataset is at https://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes. All the websites can be accessed on 16 March 2024.

Acknowledgments: The authors are grateful to the peer researchers for their source codes as well as the public HSI datasets.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Czaja, W.; Kavalerov, I.; Li, W. Exploring the high dimensional geometry of HSI features. In Proceedings of the 2021 11th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS), Amsterdam, The Netherlands, 24–26 March 2021; pp. 1–5.
2. Wang, Z.; Menenti, M. Challenges and opportunities in lidar remote sensing. *Front. Remote Sens.* **2021**, *2*, 641723. [CrossRef]
3. Roy, S.K.; Kar, P.; Hong, D.; Wu, X.; Plaza, A.; Chanussot, J. Revisiting deep hyperspectral feature extraction networks via gradient centralized convolution. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5516619. [CrossRef]
4. Hestir, E.; Brando, V.; Bresciani, M.; Giardino, C.; Matta, E.; Villa, P.; Dekker, A. Measuring freshwater aquatic ecosystems: The need for a hyperspectral global mapping satellite mission. *Remote Sens. Environ.* **2015**, *167*, 181–195. [CrossRef]
5. Shimoni, M.; Haelterman, R.; Perneel, C. Hyperspectral imaging for military and security applications: Combining myriad processing and sensing techniques. *IEEE Geosci. Remote Sens. Mag.* **2019**, *7*, 101–117. [CrossRef]
6. Wu, X.; Hong, D.; Chanussot, J. UIU-Net: U-Net in U-Net for infrared small object detection. *IEEE Trans. Image Process.* **2023**, *32*, 364–376. [CrossRef]
7. Carrino, T.A.; Crósta, A.P.; Toledo, C.L.B.; Silva, A.M. Hyper-spectral remote sensing applied to mineral exploration in southern peru:A multiple data integration approach in the chapi chiara gold prospect. *Int. J. Appl. Earth Obs. Geoinf.* **2018**, *64*, 287–300.
8. Schimleck, L.; Ma, T.; Inagaki, T.; Tsuchikawa, S. Review of Near Infrared Hyperspectral Imaging Applications Related to Wood and Wood Products. *Appl. Spectrosc. Rev.* **2022**, *57*, 2098759. [CrossRef]
9. Liao, X.; Liao, G.; Xiao, L. Rapeseed Storage Quality Detection Using Hyperspectral Image Technology–An Application for Future Smart Cities. *J. Test. Eval.* **2022**, *51*, JTE20220073. [CrossRef]
10. Du, P.; Xia, J.S.; Xue, Z.H. Review of hyperspectral remote sensing image classification. *J. Remote Sens.* **2016**, *20*, 236–256.
11. Roy, S.K.; Krishna, G.; Dubey, S.R.; Chaudhuri, B.B. HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 277–281. [CrossRef]
12. Sun, Y.; Wang, M.; Wei, C.; Zhong, Y.; Xiang, J. Heterogeneous spectral-spatial network with 3D attention and MLP for hyperspectral image classification using limited training samples. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2023**, *16*, 8702–8720. [CrossRef]
13. Hong, D.; Han, Z.; Yao, J. SpectralFormer: Rethinking hyperspectral image classification with transformers. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5518615. [CrossRef]
14. Sang, M.; Zhao, Y.; Liu, G. Improving Transformer-Based Networks with Locality for Automatic Speaker Verification. In Proceedings of the 2023 48th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 4–10 June 2023; pp. 1–5.
15. Sun, L.; Zhao, G.; Zheng, Y.; Wu, Z. Spectral–spatial feature tokenization transformer for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–14. [CrossRef]
16. Wang, A.; Xing, S.; Zhao, Y.; Wu, H.; Iwahori, Y. A hyperspectral image classification method based on adaptive spectral spatial kernel combined with improved vision transformer. *Remote Sens.* **2022**, *14*, 3705. [CrossRef]
17. Li, J.; Bioucas-Dias, J.M.; Plaza, A. Spectral–spatial hyperspectral image segmentation using subspace multinomial logistic regression and Markov random fields. *IEEE Trans. Geosci. Remote Sens.* **2011**, *50*, 809–823. [CrossRef]
18. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. PointNet++: Deep hierarchical feature learning on points a metric space. *arXiv* **2017**, arXiv:1706.02413.
19. Pedergnana, M.; Marpu, P.R.; Dalla Mura, M.; Benediktsson, J.A.; Bruzzone, L. Classification of remote sensing optical and LiDAR data using extended attribute profiles. *IEEE J. Sel. Top. Signal Process.* **2012**, *6*, 856–865. [CrossRef]
20. Rasti, B.; Ghamisi, P.; Gloaguen, R. Hyperspectral and LiDAR fusion using extinction profiles and total variation component analysis. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3997–4007. [CrossRef]
21. Zhu, X.X.; Tuia, D.; Mou, L.; Xia, G.S.; Zhang, L.; Xu, F.; Fraundorfer, F. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geosci. Remote Sens. Mag.* **2017**, *5*, 8–36. [CrossRef]
22. Roy, S.K.; Deria, A.; Hong, D. Hyperspectral and LiDAR data classification using joint CNNs and morphological feature learning. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5530416. [CrossRef]

23. Song, W.; Dai, Y.; Gao, Z. Hashing-based deep metric learning for the classification of hyperspectral and LiDAR data. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5704513. [CrossRef]
24. Xu, X.; Li, W.; Ran, Q. Multisource remote sensing data classification based on convolutional neural network. *IEEE Trans. Geosci. Remote Sens.* **2017**, *56*, 937–949. [CrossRef]
25. Ding, K.; Lu, T.; Fu, W.; Li, S.; Ma, F. Global–local transformer network for HSI and LiDAR data joint classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5541213. [CrossRef]
26. Zhang, Y.; Peng, Y.; Tu, B.; Liu, Y. Local Information interaction transformer for hyperspectral and LiDAR data classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *16*, 1130–1143. [CrossRef]
27. Xu, H.; Zheng, T.; Liu, Y.; Zhang, Z.; Xue, C.; Li, J. A joint convolutional cross ViT network for hyperspectral and light detection and ranging fusion classification. *Remote Sens.* **2024**, *16*, 489. [CrossRef]
28. Roy, S.K.; Deria, A.; Hong, D. Multimodal fusion transformer for remote sensing image classification. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5515620. [CrossRef]
29. Zhao, G.; Ye, Q.; Sun, L. Joint classification of hyperspectral and LiDAR data using a hierarchical CNN and transformer. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5500716. [CrossRef]
30. Wang, Y.; Li, Y.; Wang, G.; Liu, X. Multi-scale attention network for single image super-resolution. *arXiv* **2022**, arXiv:2209.14145.
31. Gulati, A.; Qin, J.; Chiu, C.C. Conformer: Convolution-augmented transformer for speech recognition. *arXiv* **2020**, arXiv:2005.08100.
32. Hang, R.; Li, Z.; Ghamisi, P.; Hong, D.; Xia, G.; Liu, Q. Classification of hyperspectral and LiDAR data using coupled CNNs. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 4939–4950. [CrossRef]
33. Hendrycks, D.; Gimpel, K. Gaussian Error Linear Units (gelus). *arXiv* **2016**, arXiv:1606.08415.
34. Zhou, D.; Kang, B.; Jin, X.; Yang, L. DeepViT: Towards deeper vision transformer. *arXiv* **2021**, arXiv:2103.11886v4.
35. Touvron, H.; Cord, M.; Sablayrolles, A. Going deeper with image transformers. *arXiv* **2021**, arXiv:2103.17239v2.
36. Leng, Z.Q.; Tan, M.X.; Liu, C.X. PolyLoss: A polynomial expansion perspective of classification loss functions. In Proceedings of the 2022 10th IEEE Conference on International Conference on Learning Representations (ICLR), Virtual, 25–29 April 2022.
37. Gader, P.; Zare, A.; Close, R.; Aitken, J.; Tuell, G. *Muufl Gulfport Hyperspectral and LiDAR Airborne Data Set*; Technical Report REP-2013–570; University of Florida: Gainesville, FL, USA, 2013.
38. Du, X.; Zare, A. *Scene Label Ground Truth Map for Muufl Gulfport Data Set*; Technical Report 20170417; University of Florida: Gainesville, FL, USA, 2017.
39. Baumgartner, A.; Gege, P.; Köhler, C.; Lenhard, K.; Schwarzmaier, T. Characterisation methods for the hyperspectral sensor HySpex at DLR's calibration home base. *Proc. SPIE* **2012**, *8533*, 371–378.
40. Kurz, F.; Rosenbaum, D.; Leitloff, J.; Meynberg, O.; Reinartz, P. Real time camera system for disaster and traffic monitoring. Proceedings of International Conference on SMPR, Tehran, Iran, 18–19 May 2011; pp. 1–6.
41. Xiang, J.H.; Wei, C.; Wang, M.H.; Teng, L. End-to-End Multilevel Hybrid Attention Framework for Hyperspectral Image Classification. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 5511305. [CrossRef]
42. Swalpa, K.R.; Ankur, D.; Shah, C. Spectral–spatial morphological attention transformer for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2022**, *61*, 5503615.