remote sensing

MDPI

Article

How to Learn More? Exploring Kolmogorov–Arnold Networks for Hyperspectral Image Classification

Ali Jamali [1,*], Swalpa Kumar Roy [2], Danfeng Hong [3,4], Bing Lu [1] and Pedram Ghamisi [5,6]

[1] Department of Geography, Simon Fraser University, 8888 University Dr, Burnaby, BC V5A 1S6, Canada; b_lu@sfu.ca
[2] Department of Computer Science and Engineering, Alipurduar Government Engineering and Management College, Bakla 736206, India; swalpa@agemc.ac.in
[3] Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China; hongdf@aircas.ac.cn
[4] School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100049, China
[5] Machine Learning Group, Helmholtz-Zentrum Dresden-Rossendorf (HZDR), Helmholtz Institute Freiberg for Resource Technology, 09599 Freiberg, Germany; p.ghamisi@gmail.com
[6] Lancaster University, Lancaster LA1 4YR, UK
* Correspondence: alij@sfu.ca

Abstract: Convolutional neural networks (CNNs) and vision transformers (ViTs) have shown excellent capability in complex hyperspectral image (HSI) classification. However, these models require a significant number of training data and are computational resources. On the other hand, modern Multi-Layer Perceptrons (MLPs) have demonstrated a great classification capability. These modern MLP-based models require significantly less training data compared with CNNs and ViTs, achieving state-of-the-art classification accuracy. Recently, Kolmogorov–Arnold networks (KANs) were proposed as viable alternatives for MLPs. Because of their internal similarity to splines and their external similarity to MLPs, KANs are able to optimize learned features with remarkable accuracy, in addition to being able to learn new features. Thus, in this study, we assessed the effectiveness of KANs for complex HSI data classification. Moreover, to enhance the HSI classification accuracy obtained by the KANs, we developed and proposed a hybrid architecture utilizing 1D, 2D, and 3D KANs. To demonstrate the effectiveness of the proposed KAN architecture, we conducted extensive experiments on three newly created HSI benchmark datasets: QUH-Pingan, QUH-Tangdaowan, and QUH-Qingyun. The results underscored the competitive or better capability of the developed hybrid KAN-based model across these benchmark datasets over several other CNN- and ViT-based algorithms, including 1D-CNN, 2DCNN, 3D CNN, VGG-16, ResNet-50, EfficientNet, RNN, and ViT.

Keywords: hyperspectral data; vision transformer; KAN; Kolmogorov–Arnold networks; MLP; deep learning

check for updates

Citation: Jamali, A.; Roy, S.K.; Hong, D.; Lu, B.; Ghamisi, P. How to Learn More? Exploring Kolmogorov–Arnold Networks for Hyperspectral Image Classification. *Remote Sens.* **2024**, *16*, 4015. https://doi.org/10.3390/rs16214015

Academic Editor: Salah Bourennane

Received: 28 August 2024
Revised: 23 October 2024
Accepted: 24 October 2024
Published: 29 October 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Hyperspectral remote sensing has drawn a lot of interest lately for a variety of Earth observation uses [1–4]. Mapping the physical, biological, or geographical dimensions of ecosystems is necessary to monitor the temporal and spatial patterns of Earth surface activities and comprehend how they work. Because each pixel contains a wealth of spectral information, hyperspectral imaging (HSI) has been applied extensively in a variety of real-world applications, including precision agriculture [5], military object detection [6], and land use land cover mapping [7,8]. Because it offers precise and detailed information about the physical and chemical properties of objects that are imaged, HSI has grown to be an essential tool in the industry. Notably, the detailed features produce effective classification results that are too intricate for conventional methods, i.e., a nonlinear correlation among the obtained spectrum data and the corresponding object, such as buildings [3].

As opposed to standard panchromatic and multi-spectral imagery captured by satellites, HSI supplies hundreds of contiguous narrow spectral bands, offering an improved detailed and accurate technique for discerning Earth objects [9]. HSI is especially useful for more refined classification because of its capacity to identify subtle spectral characteristics that standard imagery is unable to detect [10]. The majority of techniques used in the early stages of HSI classification research concentrated on handcrafted extraction of features, such as extended morphological profiles (EMPs) [11] and extended extinction profiles (EEPs). However, these conventional classification techniques are limited in their ability to retrieve high-level characteristics of images, and they are associated with "shallow" models. As a result, these techniques typically fall short of achieving greater accuracy. Recently, it has been established that deep learning (DL) is a powerful feature extractor that effectively recognizes the nonlinear problems that have emerged in a variety of computer vision tasks. This promotes the encouraging outcomes of using DL for HSI data classification [12–14].

Convolutional neural networks (CNNs), because of their superior local contextual modeling capabilities, are widely used in spectral–spatial HSI data classification. While CNN-based methods are advantageous for spatial–contextual identification, they suffer greatly from handling spectral sequential data because long-range dependencies are often difficult for CNNs to capture correctly [15]. While the existing CNN-based techniques have shown promising results [16], they continue to encounter several difficulties. For instance, the receptive field is constrained, data are lost during the downsampling phase, and deep networks require a large amount of processing power [17]. On the other hand, in the field of computer vision, vision transformers (ViTs) have demonstrated significant promise recently [18–22]. By means of the incorporation of a multi-layer perceptron (MLP) and a multi-headed self-attention (MHSA) module, ViTs are capable of acquiring global long-range data interactions in the input sequential data. Because of this capability, the application of transformers to the classification of HSI data is expanding rapidly [23–26].

However, due to their quadratic computational complexity, transformers need a substantially higher amount of training data than CNNs and have a relatively high computational cost [27]. ViTs and CNNs have been surpassed in image classification tasks by modern MLP algorithms, such as MLP-Mixer [27] and ResMLP [28], which have demonstrated an excellent classification capability. These modern MLP models require significantly less training data compared to CNNs and ViTs, achieving state-of-the-art classification accuracy [29]. In addition, SpectralMamba [30] has been proposed for hyperspectral image classification to further reduce computational complexity while effectively improving the classification performance. This work is notable as the first to introduce the Mamba framework into the hyperspectral remote sensing field.

In the past few months, Kolmogorov–Arnold Networks (KANs), which are inspired by the Kolmogorov–Arnold representation theorem, were proposed as viable alternatives for MLPs [31]. KANs employ feature learnable activation functions on edges, or "weights", in comparison with MLPs, which have fixed activation functions on nodes, or "neurons". KANs do not use any linear weights at all; instead, a uni-variate function with spline parameterization serves as a substitute for each weight parameter. Thus, in this research, we assess and evaluate the capability and effectiveness of KAN models for complex HSI data classification over several other CNN- and vision-based models. The contributions of this paper can be summarized as follows:

- We introduce a hybrid architecture based on KANs, a technique that achieves competitive or better HSI classification accuracy over several well-known CNN- and ViT-based algorithms.
- We incorporate 1D, 2D, and 3D KAN modules to enhance the ability of linear KANs in image classification tasks. This hybrid architecture increases the discriminative capability of the KAN architecture.
- We conduct extensive experiments on a brand new, complex HSI dataset called Qingdao UAV-borne HSI (QUH), including QUH-Tangdaowan, QUH-Qingyun,

and QUH-Pingan [32]. These experiments prove the effectiveness of the proposed KAN architecture.

The remainder of the paper is structured as follows. In Section 2, we examine the structure and various modules developed in the proposed KAN-model-based architecture. Subsequently, we conduct comprehensive experiments, including a thorough discussion of the obtained HSI data classification results, as detailed in Section 4. The paper concludes with a summary provided in Section 5.

2. Proposed Methodology

Multilayer perceptrons (MLPs) are the foundation of many modern deep learning models. KANs were recently presented as an alternative to MLPs [31]. KANs are motivated by the Kolmogorov–Arnold representation theorem [33], whereas MLPs are inspired by the universal approximation theorem. Similar to MLPs, KANs have fully-connected structures. But MLPs employ fixed activation functions on nodes (referred to as "neurons"), while KANs place learnable activation functions on edges (referred to as "weights"). Instead of using linear weight matrices, KANs use a learnable 1D function parametrized as a spline for each weight parameter. Nodes in KANs do nothing more than add up incoming signals without using any non-linearities. The straightforward modification of KANs to use an activation function on the edges allows them to surpass MLPs in terms of accuracy as well as interpretability on small-scale machine learning challenges. In function-fitting tasks, smaller KANs can attain accuracy levels that are comparable to or higher than larger MLPs. KANs are known to have faster neural scaling laws than MLPs, both in theory and in practice [31]. Splines can be easily adjusted locally, are precise for low-dimensional functions, and can transition between different resolutions. However, due to their limited ability to take advantage of compositional structures, splines suffer greatly from the curse of dimensionality (COD). In contrast, MLPs are less prone to COD because of their feature learning capabilities. However, in low dimensions, their accuracy is inferior to splines due to their incapacity to optimize univariate functions. It should be noted that KANs are just combinations of splines and MLPs, utilizing their respective advantages and avoiding their respective disadvantages, despite their sophisticated mathematical interpretation. In order to correctly learn a function, the model must be able to approximate the univariate functions (internal degrees of freedom) as well as learn the compositional structure (external degrees of freedom). Because of their internal similarity to splines and their external similarity to MLPs, KANs are able to optimize learned features with remarkable accuracy in addition to being able to learn new features.

Are KANs similar to MLP? An MLP can be expressed as stacking N layers and each layer may be expressed as a linear combination of the weight matrix (W) followed by non-linear operations (δ) for the input $X \in \mathcal{R}^{p_{in}}$:

$$MLP(x) = (W_{N-1} \circ \delta \circ W_{N-2} \circ \cdots W_1 \circ \delta \circ W_0)\, x \tag{1}$$

On the other hand, a general KAN model consists of nesting N layers and the output map can be defined as follows:

$$KAN(x) = (\Phi_{N-1} \circ \Phi_{N-2} \circ \cdots \Phi_1 \circ \Phi_0)\, x \tag{2}$$

where Φ_i represents the i-th layer of the entire KAN models. Let p_{in} and p_{out} be the dimension of the input and output for each KAN layer, then Φ consists of $p_{in} \times p_{out}$ 1D learnable activation function ϕ:

$$\Phi = \{\phi_{i,j}\} \quad i = 1, 2, \ldots p_{in}, \quad j = 1, 2, \ldots p_{out} \tag{3}$$

The outcome of KAN models while computing from layer n to layer $n + 1$ may be shown in matrix form as follows:

$$X_{n+1} = \underbrace{\begin{bmatrix} \phi_{n,1,1}(\cdot) & \phi_{n,1,2}(\cdot) & \cdots & \phi_{n,1,p_n}(\cdot) \\ \phi_{n,2,1}(\cdot) & \phi_{n,2,2}(\cdot) & \cdots & \phi_{n,2,p_n}(\cdot) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{n,p_{n+1},1}(\cdot) & \phi_{n,p_{n+1},2}(\cdot) & \cdots & \phi_{n,p_{n+1},p_n}(\cdot) \end{bmatrix}}_{\Phi_n} X_n \tag{4}$$

It is evident that KANs treat non-linearities and linear transformations collectively in Φ, whereas MLPs treat them separately as W and δ. To ensure the representation power of $\phi_{i,j}$ and Φ_i, as shown in Figure 1, in the KAN models a basis function $b(x)$ (similar to that of residual connections) is included, such that the activation function $\phi(x)$ is the sum of the many spline function and the basis function $b(x)$, as defined by:

$$\phi(x) = w_b\, b(x) + w_s\, spline(x) \tag{5}$$

where $b(x) = silu(x) = x/(1 + e^{-x})$, spline(x)= $\sum_i$ c_i $B_i(x)$, and c_i are trainable. For more details, refer to Liu et al. [31].

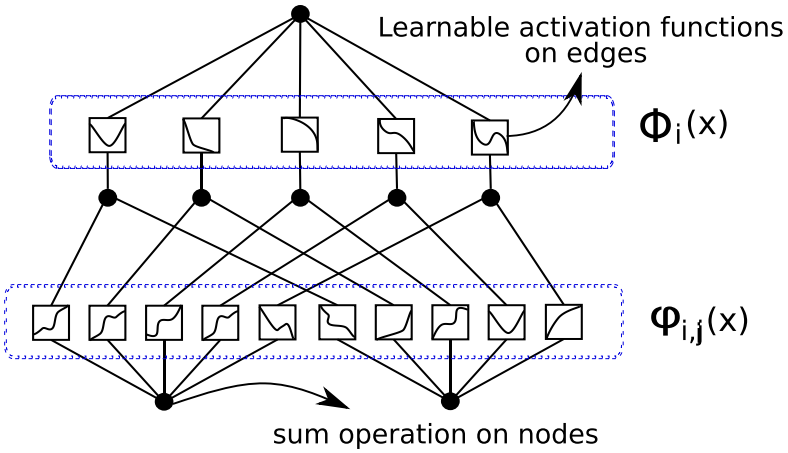


Figure 1. The overall architecture of the Kolmogorov–Arnold networks.

Classical vs. KAN Convolution: KAN convolutions are perhaps similar to traditional convolutions operation, except that each element is given a learnable non-linear activation function, which is then added to the kernel and the associated pixels in the image patch, rather than the dot product between the two. The kernel of the KAN convolution is equivalent to a KAN linear Layer of 9 inputs and 1 output neuron (shown in Figure 2). The output pixel of that convolution step is the sum of $\phi_i(x_i)$ for each input i, to which we have applied a ϕ_i learnable function. To visualize the difference between classical vs. KAN convolution, consider the input image patch $X \in \mathbf{R}^{W \times H}$, the output $O \in \mathbf{R}^{H' \times W'}$, the kernel K, and Φ for the convolutional

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1w} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2w} \\ x_{31} & x_{32} & x_{33} & \cdots & x_{3w} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{h1} & x_{h2} & x_{h3} & \cdots & x_{hw} \end{bmatrix}_{H \times W} \tag{6}$$

and KAN kernel are defined in Equation (7), respectively.

$$K = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix} \; and \; \Phi = \begin{bmatrix} \phi_{11}(\cdot) & \phi_{12}(\cdot) & \phi_{13}(\cdot) \\ \phi_{21}(\cdot) & \phi_{22}(\cdot) & \phi_{23}(\cdot) \\ \phi_{31}(\cdot) & \phi_{32}(\cdot) & \phi_{33}(\cdot) \end{bmatrix} \tag{7}$$

The output of the classical convolutional operation ($*$) can be obtained as follows:

$$o_{i,j} = \sum_{m,n=0}^{K-1} x_{i+m,j+n} K_{m,n} \tag{8}$$

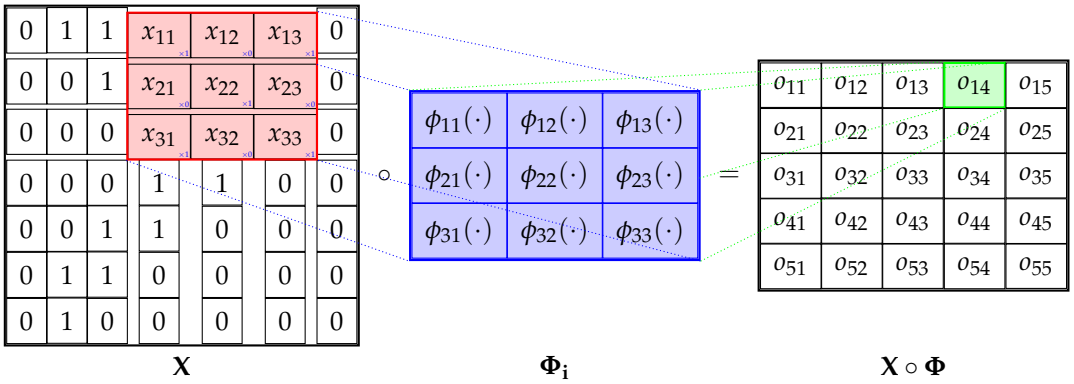


Figure 2. Pictorial representation of the KAN convolution operation where x, and Φ represent the input sub-patch and B-splines, respectively. The output of $o_{14} = x \circ \Phi$ can be calculated as $\phi_{11}(x_{11}) + \phi_{12}(x_{12}) + \phi_{13}(x_{13}) + \phi_{21}(x_{21}) + \phi_{22}(x_{22}) + \phi_{23}(x_{23}) + \phi_{31}(x_{31}) + \phi_{32}(x_{32}) + \phi_{33}(x_{33})$.

In the case of KAN convolution, the inner function $\phi(\cdot)$ may be represented as a matrix containing several activation functions as shown in Equation (7). We also have an input matrix (X) that will cycle through each activation function and has $n \times n$ characteristics. It should be noted that here, $\phi(\cdot)$ denotes the activation function rather than the weights. These activation functions are called B-splines. Let's add all of the functions, which are just basic polynomial curves and these curves are dependent upon the X input. The output of the KAN convolutional operation ($\circ$) can be obtained as follows:

$$o_{i,j} = \sum_{m,n} \phi_{m,n}(x_{i+m,j+n}) \tag{9}$$

Similarly, the above Equation (9) can easily be extended for the input image $X \in \mathbf{R}^{H \times W \times C_{in}}$ with C_{in} channels by applying a set of KAN kernels Φ, which produces the output $O \in \mathbf{R}^{H' \times W' \times C_{out}}$ as follows:

$$o_{i,j,c} = \sum_{m,n,c} \phi_{m,n,c}(x_{i+m,j+n}) \tag{10}$$

HybridSN an Embedding by KAN Layer: We experimentally selected a KAN architecture similar to the hybrid spectral network [34], as seen in Figure 3. The hybrid spectral network was proposed in 2020 and is considered to be a successful architecture in hyperspectral feature extraction and classification. Considering an input hyperspectral image of $X_{in} \in \mathbf{R}^{H \times W \times B}$, where H, W, and B indicate the height,, width, and number of spectral bands, respectively. We first utilized a principle component analysis (PCA) algorithm to reduce the number of input channels/bands in all HSI datasets to D, expressed as follows:

$$X^{'} = f^{PCA}(X_{in}) \tag{11}$$

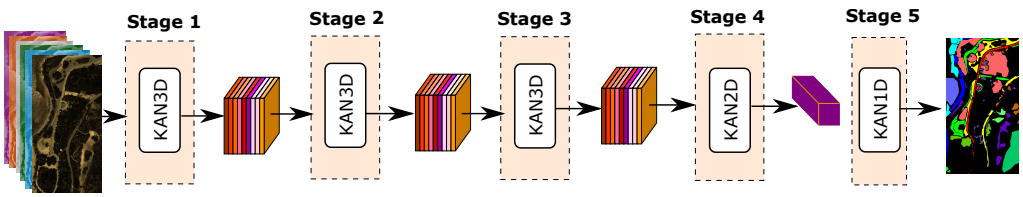


Figure 3. The overall architecture of the proposed hybrid KAN.

To enhance the HSI classification accuracy obtained by the KAN models, we developed and proposed a hybrid KAN-network-based architecture consisting of three consecutive 3D KANs with 8, 16, and 32 output channels (feature maps), expressed as follows:

$$X'' = KAN_{3D}(KAN_{3D}(KAN_{3D}(X))) \quad (12)$$

Then, one 2D KAN layer with an output channel (output map) of 64 is employed immediately after the third 3D KAN. The resulting feature maps are then flattened and sent to a 1D KAN layer with a hidden layer of 32 and output map/channel equivalent to the number classes in the HSI data, expressed as:

$$class = KAN_{1D}(KAN_{2D}(X)) \quad (13)$$

The architecture of the proposed KAN-based model layer-wised is presented in Table 1.

Table 1. The layer-wise summary of the proposed hybrid KAN architecture with a window size of 9×9. The last layer is based on the Tangdaowan dataset.)

Layer (Type)	Kernel Size	Stride	Number of Kernels/Filters	Output Shape
KAN3D-1	1	1	8	(8, 9, 9, 1)
KAN3D-2	1	1	16	(16, 9, 9, 1)
KAN3D-3	1	1	32	(32, 9, 9, 1)
Reshape	-	-	-	(32, 9, 9)
KAN2D-1	3	2	64	(64, 5 , 5)
Max pooling	3	3	-	(64, 1 , 1)
Flatten	-	-	-	(64, 1)
KAN1D-1	-	-	32	(64, 32, 18)

3. Datasets

The HSI data benchmarks that we used were located in Qingdao City, Shandong Province, China's West Coast New Area. This city is close to China's Yellow Sea coast and features a wealth of both natural and artificial surroundings, as well as rapid urbanization. Because the morphology and distribution of each region's land cover are so complex, it is not easy to classify them precisely. An unmanned aerial vehicle (UAV) equipped with hyperspectral sensors was used to collect these datasets. More specifically, the UAV platform was the DJI M600 Pro. A hyperspectral sensor called the Gaiasky mini2-VN imaging spectrometer was used. Image mosaicking, radiometric calibration, and atmospheric and geometric corrections were all carried out using the instrument manufacturer's SpecView software [32]. It is important to note that the HSI data benchmarks used in this study were more challenging than existing major HSI datasets, such as Indian Pines and Pavia University, due to their high inter-class and intra-class similarity.

3.1. QUH-Tangdaowan

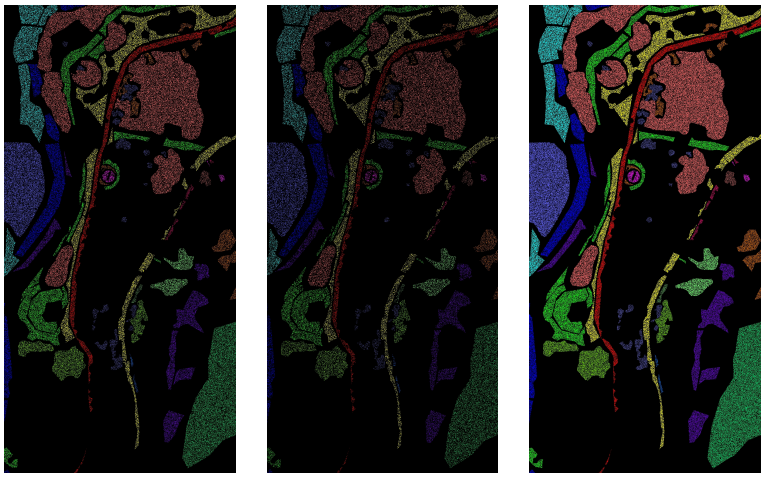
The QUH-Tangdaowan dataset was surveyed on 18 May 2021, in Tangdao Bay National Wetland Park, Qingdao, China. The UAV operated at a height of 300 m with a spatial resolution of approximately 0.15 m. This dataset comprises 176 bands with a wavelength range of 400–1000 nm and an image pixel size of 1740 × 860. Table 2 and Figure 4 illustrate the number of training, validation, and test data in this dataset.

Table 2. Number of training, validation, and test ground truth data in the QUH-Tangdaowan dataset.

Class No.	Color	Class	Train	Validation	Test	Total
1		Rubber track	7755	5170	12,924	25,849
2		Flaggingv	16,666	11,111	27,776	55,553
3		Sandy	10,211	6807	17,019	34,037

Table 2. *Cont.*

Class No.	Color	Class	Train	Validation	Test	Total
4		Asphalt	18,207	12,138	30,345	60,690
5		Boardwalk	559	372	931	1862
6		Rocky shallows	11,137	7425	18,563	37,125
7		Grassland	4238	2825	7064	14,127
8		Bulrush	19,226	12,817	32,044	64,087
9		Gravel road	9208	6139	15,348	30,695
10		*Ligustrum vicaryi*	535	357	891	1783
11		Coniferous pine	6371	4247	10,618	21,236
12		Spiraea	225	150	374	749
13		Bare soil	506	337	843	1686
14		*Buxus sinica*	266	177	443	886
15		*Photinia serrulata*	4206	2804	7010	14,020
16		Populus	42,271	28,181	70,452	140,904
17		*Ulmus pumila* L.	2940	1961	4901	9802
18		Seawater	12,682	8455	21,138	42,275
-	-	Total	167,209	111,473	278,684	557,366



(**a**) (**b**) (**c**)

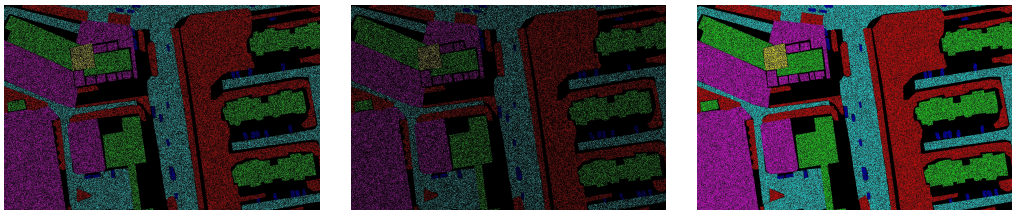
Figure 4. Pictorial view of the QUH-Tangdaowan data benchmark: (**a**) the annotation of the training samples, (**b**) the annotation of the validation samples, and (**c**) the test samples.

3.2. QUH-Qingyun

The QUH-Qingyun dataset was surveyed on 18 May 2021, in the vicinity of the Qingyun Road primary school and residential area in Qingdao, China. The UAV captured images with an image pixel size of 880 × 1360, 270 bands ranging from 400 to 1000 nm at a height of 300 m with a spatial resolution of approximately 0.15 m. Table 3 and Figure 5 illustrate the number of training, validation, and test data in this dataset.

Table 3. Number of training, validation, and test ground truth data in the QUH-Qingyun dataset.

Class No.	Color	Class	Train	Validation	Test	Total
1		Trees	83,445	55,630	139,075	278,150
2		Concrete building	53,853	35,902	89,757	179,512
3		Car	4135	2757	6891	13,783
4		Ironhide building	2930	1953	4884	9767
5		Plastic playground	65,320	43,547	108,868	217,735
6		Asphalt road	76,784	51,189	127,973	255,946
-	-	Total	286,467	190,978	477,448	954,893

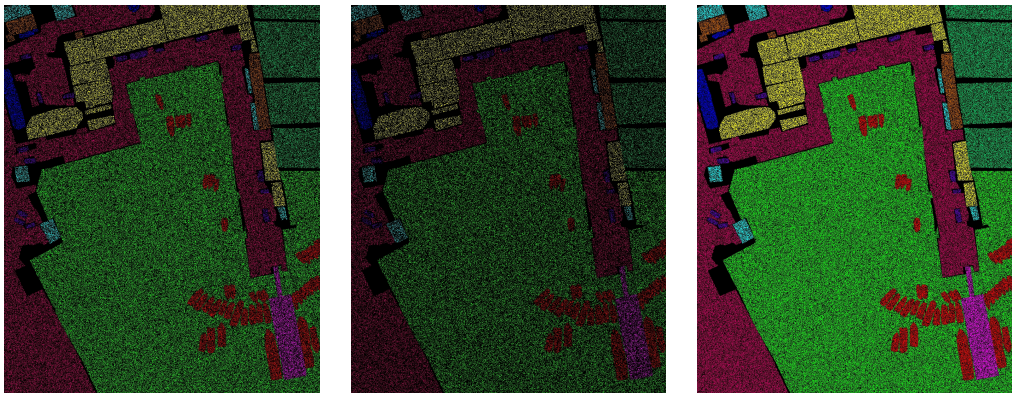


(**a**) (**b**) (**c**)

Figure 5. Pictorial view of the QUH-Qingyun data benchmark: (**a**) the annotation of the training samples, (**b**) the annotation of the validation samples, and (**c**) the test samples.

3.3. QUH-Pingan

On 19 May 2021, at Huangdao Pingan Passenger Ship Terminal in Qingdao, China, the QUH-Pingan dataset was collected. The UAV operated at a height of 200 m above the ground with a spatial resolution of approximately 0.10 m. This dataset comprises 176 bands with a wavelength range of 400–1000 nm and an image pixel size of 1230 × 1000. Table 4 and Figure 6 present the number of training, validation, and test data in this HSI dataset.



(**a**) (**b**) (**c**)

Figure 6. Pictorial view of the QUH-Pingan data benchmark: (**a**) the annotation of the training samples, (**b**) the annotation of the validation samples, and (**c**) the test samples.

Table 4. Number of training, validation, and test ground truth data in the QUH-Pingan dataset.

Class No.	Color	Class	Train	Validation	Test	Total
1		Ship	14,680	9787	24,468	48,935
2		Seawater	173,434	115,622	289,057	578,113
3		Trees	2504	1669	4172	8345
4		Concrete structure building	26,692	17,794	44,487	88,973
5		Floating pier	6228	4152	10,379	20,759
6		Brick houses	4226	2817	7043	14,086
7		Steel houses	4197	2798	6996	13,991
8		Wharf construction land	24,934	16,623	41,556	83,113
9		Car	2432	1622	4054	8108
10		Road	82,954	55,303	138,257	276,514
-	-	Total	342,281	228,187	570,469	1,140,937

3.4. Experimental Setting

This section describes the comparative approaches and experimental settings used to evaluate the proposed KAN-based model. The overall accuracy (OA), average accuracy (AA), Kappa accuracy (κ), and per-class accuracies were calculated across all HSI datasets. The percentage of accurately mapped samples was the main focus of the overall and average

accuracy. On the other hand, the Kappa (κ) accuracy resulted from statistical testing and offered information about how well classification models functioned in comparison to random selection. Essentially, the accuracy of Kappa (κ) depends on the number of classes in the dataset and the probability that sample points will be assigned a random label. As such, it functions as a more reliable accuracy metric than OA and AA, which could be deceptive in instances of unbalanced datasets. Comparative analysis against state-of-the-art methods was conducted to assess the effectiveness of the KAN models. In more detail, the HSI classification results obtained by the KAN models were evaluated agianst several other models, including 1D-CNN, 2DCNN, 3D CNN, VGG-16 [35], ResNet-50 [36], EfficientNet [37], RNN [38], and ViT [39].

4. Results

4.1. Statistical Results

Table 5 and Figure 7 illustrate the HSI classification results and maps produced by the developed CNN- and transformer-based architectures in the Tangdaowan HSI dataset. The results revealed that the KAN models, specifically the developed HybridKAN architecture, obtained a competitive HSI classification accuracy compared with other well-known CNNs and ViTs. The developed HybridKAN achieved the highest average accuracy (97.12%), while the ResNet-50 model achieved the best overall accuracy (98.09%) and Kappa value (97.82%). The HSI data classification results underscored the effectiveness of the KAN models compared with the other classification models. The 3D KAN models similar to its counterpart of 3D CNN demonstrated the least HSI classification accuracy with an average accuracy of 76.14% compared with that of 1D KAN (96.15%), 2D KAN (96.47%), and the HybridKAN (97.12%). As seen in Figure 7, HybridKAN illustrated the most homogeneous classification map with much less noise compared with the other classification architectures, showcasing its high capability in accurate HSI data classification.

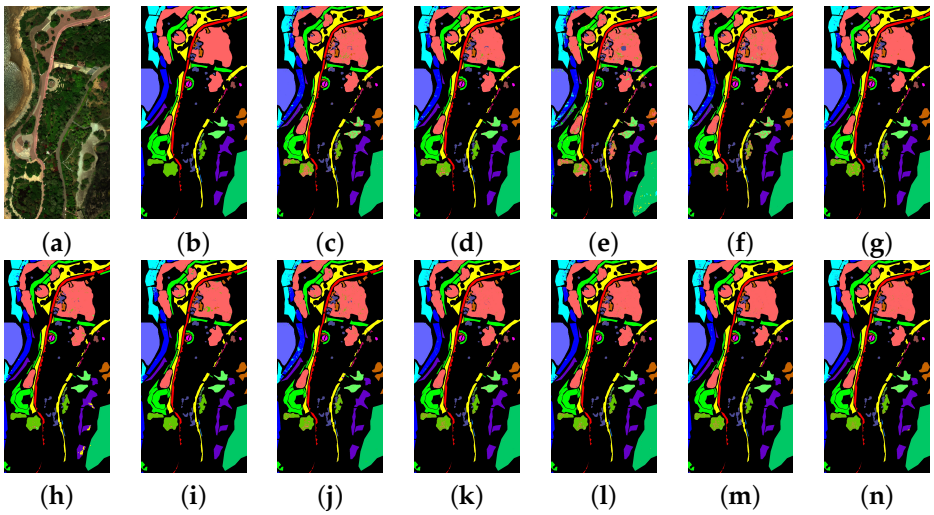


Figure 7. The predicted land cover maps obtained for the Tangdaowan HSI dataset: (**a**) RGB image (**b**) GT (**c**) 1D-CNN (**d**) 2D-CNN (**e**) 3D-CNN (**f**) DRNN (**g**) ResNet50 (**h**) VGG-16 (**i**) EfficientNet (**j**) ViT (**k**) 1D-KAN (**l**) 2D-KAN (**m**) 3D-KAN (**n**) HybridKAN, respectively.

Table 5. Classification results in terms of OA, AA, and Kappa (in %) obtained on the Tangdaowan dataset.

Class No.	1DCNN	2DCNN	3DCNN	VGG16 [35]	ResNet50 [36]	EfficientNet [37]	RNN [38]	ViT [39]	1DKAN [31]	2DCKAN	3DKAN	HybridKAN
1	99.80	99.82	99.45	99.96	99.93	99.93	99.93	**100.0**	99.95	99.91	98.81	99.85
2	98.42	99.49	87.00	**99.75**	97.49	99.48	97.83	99.56	99.52	99.52	95.91	99.48
3	94.98	96.62	86.46	97.63	**98.54**	97.02	92.37	92.96	97.31	96.74	92.33	97.24
4	99.17	99.89	92.02	**99.96**	99.89	99.92	98.26	99.20	99.78	99.60	98.52	99.71
5	95.38	97.42	87.20	99.78	**99.89**	99.24	97.63	97.85	97.88	97.63	38.86	99.35
6	88.54	94.66	86.84	94.38	93.82	92.71	90.92	94.81	**97.21**	96.87	90.61	97.34
7	81.14	90.74	40.00	81.92	94.70	96.06	78.70	91.25	93.84	90.72	72.72	**95.78**
8	99.81	99.92	98.34	99.83	**99.99**	99.93	99.84	99.80	99.97	99.94	98.85	99.89
9	96.89	99.35	90.84	92.96	99.79	**99.95**	97.55	99.70	99.45	99.44	94.33	99.51
10	92.81	**98.54**	82.57	98.42	96.18	85.18	97.75	93.93	95.22	95.72	70.38	97.75
11	64.80	84.77	46.78	94.76	84.95	**96.83**	64.72	86.68	87.24	91.83	45.16	90.34
12	77.54	86.89	30.87	93.04	92.78	87.43	89.83	78.87	94.18	91.25	56.99	**95.85**
13	98.22	**100.0**	91.91	**100.0**	99.88	99.52	99.76	98.81	99.70	99.53	96.07	99.06
14	83.74	98.41	37.09	89.16	95.25	**100.0**	86.00	98.64	97.69	95.67	45.74	97.25
15	79.71	96.40	79.31	**97.08**	96.81	95.83	83.05	96.16	94.07	93.68	77.42	95.04
16	93.56	96.35	86.28	96.85	**98.99**	96.38	92.78	95.66	96.52	97.16	88.58	97.20
17	65.53	92.51	74.53	**97.44**	97.12	96.89	67.41	91.83	90.79	94.20	71.58	93.66
18	99.84	99.93	98.54	99.90	99.91	**99.99**	99.83	99.69	99.86	99.94	98.76	99.91
OA	93.81	97.32	87.63	97.43	**98.09**	97.89	93.59	96.90	97.68	97.90	91.11	98.08
AA	89.44	96.21	75.61	96.27	96.99	96.79	90.79	95.30	96.15	96.47	76.14	**97.12**
$\kappa(\times 100)$	92.93	96.95	85.82	97.08	**97.82**	97.61	92.69	96.48	97.36	97.61	89.80	97.81

On the other hand, in the Pingan HSI dataset, as seen in Table 6 and Figure 8, the highest HSI data classification result was obtained by the VGG-16 CNN model with an overall accuracy, Kappa value, and average accuracy of 99.06%, 98.61%, and 98.09%, respectively. In this HSI dataset, the developed HybridKAN architecture demonstrated a competitive HSI classification accuracy compared with other models, with an average accuracy, Kappa value, and overall accuracy of 95.95%, 97.74%, and 98.48%, respectively. Similar to the Tangdaowan dataset, the 3D KAN model with an average accuracy of 81.53% illustrated the least classification accuracy over the 2DKAN (94.14%), 1DKAN (95.42%), and the HybridKAN (95.95%). While the statistical results showed a slightly better classification accuracy with VGG-16 over HybridKAN, as seen in Figure 8, the HybridKAN architecture produced much less noise and a more homogeneous classification map.

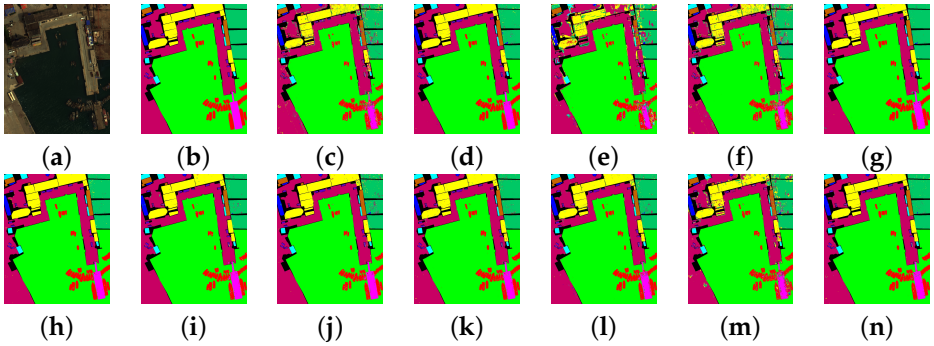


Figure 8. The predicted land cover maps were created for the Pingan data set: (**a**) RGB image (**b**) GT (**c**) 1D-CNN (**d**) 2D-CNN (**e**) 3D-CNN (**f**) DRNN (**g**) ResNet50 (**h**) VGG-16 (**i**) EfficientNet (**j**) ViT (**k**) 1D-KAN (**l**) 2D-KAN (**m**) 3D-KAN (**n**) HybridKAN, respectively.

Moreover, as seen in Table 7 and Figure 9, the best HSI data classification accuracy was obtained by the developed HybridKAN architecture in terms of overall accuracy (97.06%) and Kappa value (96.11%) in the Qingyun HSI dataset. The highest average accuracy was achieved by the 2DCNN model (95.60%) over the other developed classification models. The proposed HybridKAN architecture obtained the highest average accuracy (94.91%) as compared to 3DKAN (84.69%), 1DKAN (92.11%), and 2DKAN network (92.93%). Overall, the obtained results showed significant capability of KAN models for complex land cover land use mapping using HSI data. We used a simple and straightforward architecture similar to traditional CNN-based models (e.g., Hybrid SN [34]), yet the developed model based on the KAN architectures illustrated competitive or better HSI data classification capability compared to other developed CNN- and ViT-based classification models.

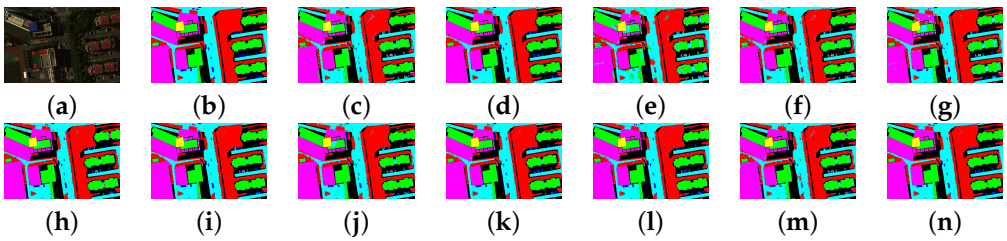


Figure 9. The predicted land cover maps were created for the Qingyun dataset: (**a**) RGB image (**b**) GT (**c**) 1D-CNN (**d**) 2D-CNN (**e**) 3D-CNN (**f**) DRNN (**g**) ResNet50 (**h**) VGG-16 (**i**) EfficientNet (**j**) ViT (**k**) 1D-KAN (**l**) 2D-KAN (**m**) 3D-KAN (**n**) HybridKAN, respectively.

Table 6. Classification results in terms of OA, AA, and Kappa (in %) obtained on the Pingan dataset.

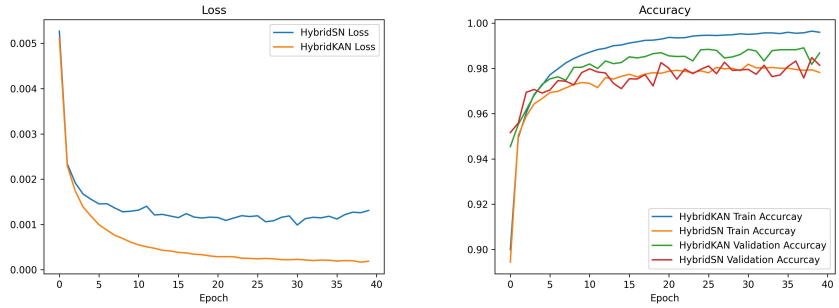
Class No.	1DCNN	2DCNN	3DCNN	VGG16 [35]	ResNet50 [36]	EfficientNet [37]	RNN [38]	ViT [39]	1DKAN [31]	2DKAN	3DKAN	HybridKAN
1	78.73	93.17	73.12	**97.04**	95.23	89.84	73.72	85.26	91.44	91.14	75.78	92.91
2	98.84	99.59	98.58	99.36	99.59	99.56	98.91	**99.61**	99.47	99.36	99.18	99.50
3	95.18	99.08	80.16	99.25	**99.83**	97.69	95.97	95.39	97.96	93.64	89.33	96.98
4	83.38	98.11	60.58	**99.69**	98.27	98.59	80.03	97.88	96.84	96.45	83.21	97.54
5	68.01	95.00	58.77	**96.03**	94.44	93.94	67.41	95.36	91.39	88.50	72.72	93.42
6	88.20	**98.35**	53.40	98.75	98.42	97.58	89.49	97.48	97.54	95.81	86.52	97.30
7	87.36	98.77	81.34	**98.88**	99.02	97.29	85.37	98.69	98.63	96.68	79.45	98.15
8	86.74	97.22	76.71	98.39	**98.97**	98.27	84.12	94.06	96.53	95.91	83.43	96.87
9	54.66	90.94	30.67	**94.35**	91.39	89.93	43.88	91.36	86.52	82.81	50.59	89.11
10	95.12	**99.36**	88.00	99.18	98.67	99.05	95.48	98.89	98.74	98.30	94.45	98.91
OA	93.82	98.79	88.67	**99.06**	98.86	98.61	93.18	98.08	98.24	97.84	93.36	98.48
AA	83.62	96.96	67.37	**98.09**	97.38	96.18	81.44	95.40	95.42	94.14	81.53	95.95
$\kappa(\times 100)$	90.77	98.20	83.01	**98.61**	98.31	97.93	89.80	97.13	97.38	96.78	90.09	97.74

Table 7. Classification results in terms of OA, AA, and Kappa (in %) obtained from the Qingyun dataset.

Class No.	1DCNN	2DCNN	3DCNN	VGG16 [35]	ResNet50 [36]	EfficientNet [37]	RNN [38]	ViT [39]	1DKAN [31]	2DKAN	3DKAN	HybridKAN
1	94.62	97.53	90.96	96.82	97.28	**97.64**	95.36	97.42	96.45	96.50	93.72	97.17
2	92.28	**97.89**	84.23	96.01	87.62	96.04	92.88	95.02	96.73	**97.89**	92.94	97.66
3	31.82	**85.48**	10.11	71.31	55.82	69.04	38.55	53.76	73.01	73.07	47.39	82.62
4	97.03	99.24	95.66	99.52	97.62	**99.83**	98.54	98.15	99.10	98.62	96.89	99.07
5	92.71	98.20	91.21	**98.54**	93.09	96.95	92.35	95.82	97.38	97.91	95.85	98.05
6	90.02	95.23	83.80	95.05	93.13	95.69	90.63	93.13	95.32	95.68	90.80	**96.35**
OA	91.63	96.98	87.19	96.24	92.80	96.27	92.15	94.83	96.13	96.55	92.74	**97.06**
AA	83.08	**95.60**	75.44	92.87	87.43	92.53	84.72	88.88	92.11	92.93	84.69	94.91
$\kappa(\times 100)$	88.89	96.00	82.94	95.03	90.43	95.05	89.59	93.15	94.87	95.43	90.37	**96.11**

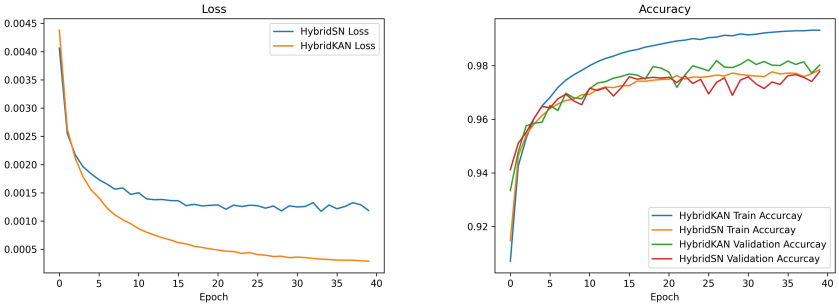
4.2. Convergence Graph Between HybridSN and Its KAN Version

Because the training process in deep learning may prove time-consuming and it is not always evident when the network has acquired sufficient information, convergence is an important tool. Nevertheless, when the validation and training error cease to decrease, a deep learning model is considered to have been converged. An ideal solution is not always guaranteed by convergence; this relies on several variables, including the network's architecture, the hyperparameters, and the quality of the HSI data. As seen in Figures 10–12, the HybridKAN architecture, which utilizes KAN layers, is superior over the HybridSN using convolutional layers in terms of lower loss, higher train accuracy, and higher validation accuracy. The HyperKAN model requires a smaller number of epochs for its convergence, which is vital in the remote sensing filed with the existence of high-dimensional and complex data. This proves the better capability of the developed HyperKAN model over the HybridSN classification algorithm.



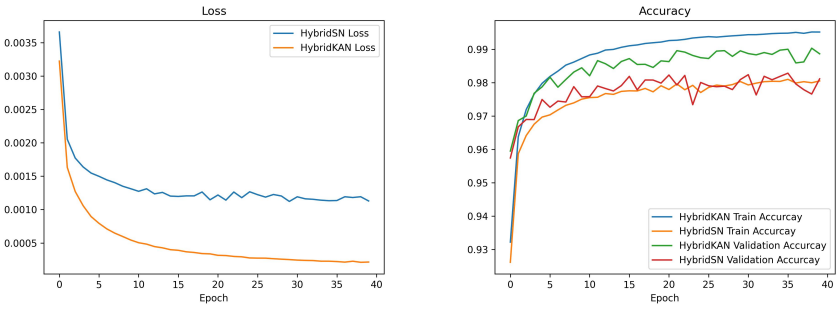
(**a**) (**b**)

Figure 10. The convergence graph of (**a**) loss and (**b**) train accuracy between HybridSN and its KAN version over the Tangdaowan HSI benchmark dataset for 40 epochs.



(**a**) (**b**)

Figure 11. The convergence graph of (**a**) loss and (**b**) train accuracy between HybridSN and its KAN version over the Qingyun HSI benchmark dataset for 40 epochs.

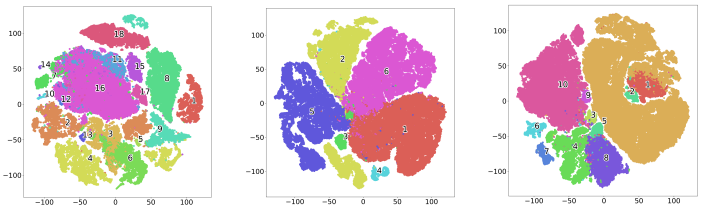


(**a**) (**b**)

Figure 12. The convergence graph of (**a**) loss and (**b**) train accuracy between HybridSN and its KAN version over the Pingan HSI benchmark dataset for 40 epochs.

4.3. Feature Visualization of KAN Using t-SNE

The many spectral ranges that make up HS data allow for the comprehensive capture of details over a large range of electromagnetic wavelengths. As such, it can be difficult to visualize these high-dimensional characteristics. Nevertheless, t-Distributed Stochastic Neighbour Embedding (t-SNE) [40] may make it easier to observe the complex spectral–spatial features that the developed HybridKAN extracts in a two-dimensional space. To analyze the representational abilities of our model, this visualization is essential because it provides insights that may not be immediately clear from a direct examination of the raw data. The feature distributions for HybridKAN in 2D feature space by t-SNE are shown in Figure 13. As can be seen in Figure 13, the developed KAN-based architecture demonstrated excellent feature separation capability for recognizing complex land covers in all three HSI data benchmarks, according to the results obtained by the t-SNE algorithm. Furthermore, due to its weighted non-linear function rather than traditional MLPs with fixed non-linear activation functions, HybridKAN's classification map, as shown in Figures 7–9, showed the least amount of noise and the most homogeneous classification map when compared with the other implemented algorithms. Furthermore, compared with the conventional ViT architecture, it is clear that HybridKAN's classification maps are far less noisy.



(**a**) (**b**) (**c**)

Figure 13. The t-SNE visual presentation of the HybridKAN in data benchmark of (**a**) Tangdaowan, (**b**) Qingyun, and (**c**) Pingan.

4.4. Hyperparameter Sensitivity Analysis

The complexity and quantity of parameters in a classification algorithm are important factors to consider in the remote sensing field. In comparison, in 2D and 3D KAN models, HybridKAN model has more parameters, but this increase is justified. The higher OA, AA, and k in Tables 5–7 demonstrate the considerable boost to the classification performance of the hybrid model, which justifies the trade-off. Furthermore, as shown in Figures 7–8, the visual classification maps produced by the HybridKAN architecture yield less noise and more homogeneous classification maps. This perspective emphasizes the notion that the trade-off of greater model complexity (as indicated by the greater number of parameters in Table 8) is matched with a demonstrable and supported improvement in the model's capacity to categorize high-spectrum imagery correctly.

Table 8. Number of parameters in the developed classification algorithms. (* To reduce the number of trainable parameters, VGG-16, EfficientNet, and ResNet-50 have been modified).

Model	1DCNN	2DCNN	3DCNN	VGG16 * [35]	ResNet50 * [36]	EfficientNet * [37]	RNN [38]	ViT [39]	1DKAN [31]	2DCKAN	3DKAN	HybridKAN
Total number of parameters	29,170	60,902	4,282	1,174,162	211,826	177,406	7,686	152,586	565,458	14,743	50,826	135,090
Number of trainable parameters	29,170	60,902	4,282	1,174,162	211,826	177,406	7,686	152,586	565,458	14,743	50,826	135,090
Number of non-trainable parameters	0	0	0	0	0	0	0	0	0	0	0	0
Forward/backward pass size (MB)	0.01	0.04	0.01	0.64	0.07	0.29	0.11	18.63	0.07	0.12	0.03	0.12
Params size (MB)	0.11	0.23	0.02	4.48	0.81	0.68	0.03	0.58	2.16	0.06	0.19	0.52
Estimated Total Size (MB)	0.12	0.27	0.03	5.12	0.88	0.97	0.14	19.22	2.23	0.18	0.23	0.63

5. Conclusions

This research proposed and discussed a KAN-model-based architecture for complex land use land cover mapping using HSI data, which employs 1D, 2D, and 3D KAN models. The classification results on three highly complex HSI datasets demonstrate that the developed classification model, HybridKAN, was competitive or better statistically and visually over several other CNN- and ViT-based algorithms, including 1D-CNN, 2D-CNN, 3D-CNN, VGG-16, ResNet-50, EfficientNet, RNN, and ViT. The obtained results underscored the significant potential use of KAN models in complex remote sensing tasks. The HSI data classification ability of the proposed hybrid KAN architecture compared with other CNN-and ViT-based classification models is shown over three HSI benchmark datasets: QUH-Pingan, QUH-Tangdaowan, and QUH-Qingyun. The results underscored the competitive or better capability of the developed hybrid model across these benchmark datasets compared with state-of-the-art classification architectures.

Author Contributions: Conceptualization, A.J. and S.K.R.; methodology, A.J. and S.K.R.; software, A.J.; validation, D.H. and P.G.; formal analysis, A.J.; investigation, D.H. and P.G.; resources, A.J.; data curation, A.J.; writing—original draft preparation, A.J. and S.K.R.; writing—review and editing, D.H., B.L. and P.G.; visualization, A.J. and S.K.R.; supervision, B.L. and P.G.; project administration, A.J.; funding acquisition, A.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data and code are available at: https://github.com/aj1365/HSIConvKAN.

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

References

1. Hong, D.; Zhang, B.; Li, X.; Li, Y.; Li, C.; Yao, J.; Yokoya, N.; Li, H.; Ghamisi, P.; Jia, X.; et al. Spectralgpt: Spectral remote sensing foundation model. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 5227–5244. [CrossRef] [PubMed]
2. Ghamisi, P.; Yokoya, N.; Li, J.; Liao, W.; Liu, S.; Plaza, J.; Rasti, B.; Plaza, A. Advances in Hyperspectral Image and Signal Processing: A Comprehensive Overview of the State of the Art. *IEEE Geosci. Remote. Sens. Mag.* **2017**, *5*, 37–78. [CrossRef]
3. Ullah, F.; Ullah, I.; Khan, R.U.; Khan, S.; Khan, K.; Pau, G. Conventional to Deep Ensemble Methods for Hyperspectral Image Classification: A Comprehensive Survey. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.* **2024**, *17*, 3878–3916. [CrossRef]
4. Li, C.; Zhang, B.; Hong, D.; Jia, X.; Plaza, A.; Chanussot, J. Learning Disentangled Priors for Hyperspectral Anomaly Detection: A Coupling Model-driven and Data-driven Paradigm. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, 1–14. [CrossRef]
5. Khan, A.; Vibhute, A.D.; Mali, S.; Patil, C. A systematic review on hyperspectral imaging technology with a machine and deep learning methodology for agricultural applications. *Ecol. Inform.* **2022**, *69*, 101678. [CrossRef]
6. Ke, C. Military object detection using multiple information extracted from hyperspectral imagery. In Proceedings of the 2017 International Conference on Progress in Informatics and Computing (PIC), Nanjing, China, 15–17 December 2017; pp. 124–128. [CrossRef]
7. Hong, D.; Zhang, B.; Li, H.; Li, Y.; Yao, J.; Li, C.; Werner, M.; Chanussot, J.; Zipf, A.; Zhu, X.X. Cross-city matters: A multimodal remote sensing benchmark dataset for cross-city semantic segmentation using high-resolution domain adaptation networks. *Remote Sens. Environ.* **2023**, *299*, 113856. [CrossRef]
8. Roy, S.K.; Sukul, A.; Jamali, A.; Haut, J.M.; Ghamisi, P. Cross Hyperspectral and LiDAR Attention Transformer: An Extended Self-Attention for Land Use and Land Cover Classification. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5512815. [CrossRef]
9. Landgrebe, D. Hyperspectral image data analysis. *IEEE Signal Process. Mag.* **2002**, *19*, 17–28. [CrossRef]
10. Li, J.; Marpu, P.R.; Plaza, A.; Bioucas-Dias, J.M.; Benediktsson, J.A. Generalized Composite Kernel Framework for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2013**, *51*, 4816–4829. [CrossRef]
11. Fauvel, M.; Chanussot, J.; Benediktsson, J.A.; Sveinsson, J.R. Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles. *IEEE Int. Geosci. Remote Sens. Symp.* **2007**, *46*, 4834–4837. [CrossRef]
12. He, X.; Chen, Y.; Huang, L.; Hong, D.; Du, Q. Foundation Model-based Multimodal Remote Sensing Data Classification. *IEEE Trans. Geosci. Remote Sens.* **2023**, *62*, 5502117. [CrossRef]
13. He, X.; Chen, Y.; Huang, L. Bayesian Deep Learning for Hyperspectral Image Classification With Low Uncertainty. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5506916. [CrossRef]
14. Li, C.; Zhang, B.; Hong, D.; Yao, J.; Chanussot, J. LRR-Net: An interpretable deep unfolding network for hyperspectral anomaly detection. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5513412. [CrossRef]
15. Xue, Z.; Tan, X.; Yu, X.; Liu, B.; Yu, A.; Zhang, P. Deep Hierarchical Vision Transformer for Hyperspectral and LiDAR Data Classification. *IEEE Trans. Image Process.* **2022**, *31*, 3095–3110. [CrossRef] [PubMed]

16. Chen, Y.; Jiang, H.; Li, C.; Jia, X.; Ghamisi, P. Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 6232–6251. [CrossRef]
17. Yu, Q.; Wei, W.; Li, D.; Pan, Z.; Li, C.; Hong, D. HyperSINet: A Synergetic Interaction Network Combined with Convolution and Transformer for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 5508118. [CrossRef]
18. Li, S.; Song, W.; Fang, L.; Chen, Y.; Ghamisi, P.; Benediktsson, J.A. Deep Learning for Hyperspectral Image Classification: An Overview. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 6690–6709. [CrossRef]
19. Lee, H.; Kwon, H. Going Deeper with Contextual CNN for Hyperspectral Image Classification. *IEEE Trans. Image Process.* **2017**, *26*, 4843–4855. [CrossRef]
20. Ran, Q.; Zhou, Y.; Hong, D.; Bi, M.; Ni, L.; Li, X.; Ahmad, M. Deep transformer and few-shot learning for hyperspectral image classification. *CAAI Trans. Intell. Technol.* **2023**, *8*, 1323–1336. [CrossRef]
21. Yao, J.; Zhang, B.; Li, C.; Hong, D.; Chanussot, J. Extended vision transformer (ExViT) for land use and land cover classification: A multimodal deep learning framework. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–15. [CrossRef]
22. Yang, X.; Ye, Y.; Li, X.; Lau, R.Y.K.; Zhang, X.; Huang, X. Hyperspectral Image Classification With Deep Learning Models. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 5408–5423. [CrossRef]
23. Liang, L.; Zhang, Y.; Zhang, S.; Li, J.; Plaza, A.; Kang, X. Fast Hyperspectral Image Classification Combining Transformers and SimAM-Based CNNs. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5522219. [CrossRef]
24. Hong, D.; Han, Z.; Yao, J.; Gao, L.; Zhang, B.; Plaza, A.; Chanussot, J. SpectralFormer: Rethinking Hyperspectral Image Classification with Transformers. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5518615. [CrossRef]
25. Mei, S.; Song, C.; Ma, M.; Xu, F. Hyperspectral Image Classification Using Group-Aware Hierarchical Transformer. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5539014. [CrossRef]
26. Li, C.; Zhang, B.; Hong, D.; Zhou, J.; Vivone, G.; Li, S.; Chanussot, J. CasFormer: Cascaded transformers for fusion-aware computational hyperspectral imaging. *Inf. Fusion* **2024**, *108*, 102408. [CrossRef]
27. Tolstikhin, I.O.; Houlsby, N.; Kolesnikov, A.; Beyer, L.; Zhai, X.; Unterthiner, T.; Yung, J.; Steiner, A.; Keysers, D.; Uszkoreit, J.; et al. MLP-Mixer: An all-MLP Architecture for Vision. In *Advances in Neural Information Processing Systems*; Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W., Eds.; Curran Associates, Inc.: New York, NY, USA, 2021; Volume 34, pp. 24261–24272.
28. Touvron, H.; Bojanowski, P.; Caron, M.; Cord, M.; El-Nouby, A.; Grave, E.; Izacard, G.; Joulin, A.; Synnaeve, G.; Verbeek, J.; et al. ResMLP: Feedforward Networks for Image Classification With Data-Efficient Training. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 5314–5321. [CrossRef]
29. Jamali, A.; Roy, S.K.; Hong, D.; Atkinson, P.M.; Ghamisi, P. Spatial-Gated Multilayer Perceptron for Land Use and Land Cover Mapping. *IEEE Geosci. Remote Sens. Lett.* **2024**, *21*, 5502105. [CrossRef]
30. Yao, J.; Hong, D.; Li, C.; Chanussot, J. Spectralmamba: Efficient mamba for hyperspectral image classification. *arXiv* **2024**, arXiv:2404.08489.
31. Liu, Z.; Wang, Y.; Vaidya, S.; Ruehle, F.; Halverson, J.; Soljačić, M.; Hou, T.Y.; Tegmark, M. KAN: Kolmogorov-Arnold Networks. *arXiv* **2024**, arXiv:abs/2404.19756.
32. Fu, H.; Sun, G.; Zhang, L.; Zhang, A.; Ren, J.; Jia, X.; Li, F. Three-dimensional singular spectrum analysis for precise land cover classification from UAV-borne hyperspectral benchmark datasets. *ISPRS J. Photogramm. Remote Sens.* **2023**, *203*, 115–134. [CrossRef]
33. Kolmogorov, A.N. On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition. In *Doklady Akademii Nauk*; Russian Academy of Sciences: Moscow, Russia, 1957; Volume 114, pp. 953–956.
34. 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]
35. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556.
36. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 30 June 2016.
37. Koonce, B., EfficientNet. In *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*; Apress: Berkeley, CA, USA, 2021; pp. 109–123. [CrossRef]
38. Mou, L.; Ghamisi, P.; Zhu, X.X. Deep Recurrent Neural Networks for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3639–3655. [CrossRef]
39. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16 $\times$ 16 Words: Transformers for Image Recognition at Scale. *arXiv* **2010**, arXiv:2010.11929.
40. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *Mach. Learn. Res.* **2008**, *9*, 2579–2605.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.