

PROCEEDINGS OF SPIE

SPIDigitalLibrary.org/conference-proceedings-of-spie

Quaternion based neural network for hyperspectral image classification

Rao, Shishir Paramathma, Panetta, Karen, Agaian, Sos

Shishir Paramathma Rao, Karen Panetta, Sos Agaian, "Quaternion based neural network for hyperspectral image classification," Proc. SPIE 11399, Mobile Multimedia/Image Processing, Security, and Applications 2020, 113990S (26 May 2020); doi: 10.1117/12.2558808

SPIE.

Event: SPIE Defense + Commercial Sensing, 2020, Online Only

Quaternion Based Neural Network for Hyperspectral Image Classification

Shishir Paramathma Rao^{*a}, Karen Panetta^a, Sos Agaian^b

^aDepartment of Electrical and Computer Engineering, Tufts University, Medford, MA, USA 02155;

^bDistinguish professor of Computer Science, The City University of New York, New York City, NY, USA 10017

ABSTRACT

Neural networks have emerged to be the most appropriate method for tackling the classification problem for hyperspectral images (HSI). Convolutional neural networks (CNNs), being the current state-of-art for various classification tasks, have some limitations in the context of HSI. These CNN models are very susceptible to overfitting because of 1) lack of availability of training samples, 2) large number of parameters to fine-tune. Furthermore, the learning rates used by CNN must be small to avoid vanishing gradients, and thus the gradient descent takes small steps to converge and slows down the model runtime. To overcome these drawbacks, a novel quaternion based hyperspectral image classification network (QHIC Net) is proposed in this paper. The QHIC Net can model both the local dependencies between the spectral channels of a single-pixel and the global structural relationship describing the edges or shapes formed by a group of pixels, making it suitable for HSI datasets that are small and diverse. Experimental results on three HSI datasets demonstrate that the QHIC Net performs on par with the traditional CNN based methods for HSI Classification with a far fewer number of parameters.

Keywords: Classification, deep learning, hyperspectral imaging, spectral-spatial feature learning

1. INTRODUCTION

Hyperspectral imaging (HSI) is one of the import techniques in remote sensing and plays an essential role in delivering a rich information source for numerous applications. It collects the electromagnetic spectrum data of wavelengths ranging from visible to near-infrared and is characterized into hundreds of continuous narrow observation bands. In HSIs, each pixel can be denoted as a high-dimensional vector consisting of spectral reflectance values of a specific wavelength. Analysis of these images is of very high importance in many applications such as urban development [1, 2], land change monitoring[3, 4], resource management [5], and precision agriculture [6, 7]. HSI classification, the method of assigning each pixel to a class based on the spectral characteristics, is one of the most sought after research in the remote sensing ad hyperspectral community [8].

To analyze the hyperspectral images, various classification techniques have been researched in the past. The early studies explored the role of spectral signatures for classification purposes and proposed many one-dimensional spectral-wise classification techniques. Some of them include decision trees [9], random forest [10], and support vector machines (SVM) [11]. In random forest method, several decision trees are generated in the training phase and the output class of each hyperspectral pixel is obtained by the integrated predictions of each decision tree. SVM based approaches, in contrast, have achieved significant results due to their ability to work with high dimensional data with limited training samples. SVM maps the data to high-dimensional feature space and seeks for a hyperplane that can separate the data samples. Additionally, there are some extensions of SVM which have better discriminatory abilities [12, 13]. However, both SVM and random forest-based approaches are considered shallow learning. They are severely limited in their capabilities to handle high dimensional and densely nonlinear data such as the hyperspectral images [14].

Furthermore, logistics regression [15], dynamic subspace [16] methods and techniques such as principal component analysis (PCA) [17, 18], independent component analysis (ICA) [19], and linear discriminant analysis (LDA) [20] which are based on designing effective feature extraction and dimensionality reduction have also been proposed. However, the classification accuracy of these pixel-wise classifiers is unsatisfactory as they are not considering the spatial context. To alleviate these issues, many spectral-spatial feature-based classification methods [21, 22] that incorporate spatial contextual information to the pixel-based classifiers have been proposed and have been reported to help improve the classification

accuracies. Additionally, sparse representation methods such as [23, 24], which are based on the observation that hyperspectral pixels can be represented by a linear combination of a few common pixels from the same class, have also been proposed. Although these spectral-spatial methods have had some success in obtaining good performance, they are limited in their performance due to their inability to generalize and scale, lack of adaptation to varying context, and the need for human expertise to tailor the parameters. Moreover, the handcrafted features and the shallow-based features that are detected by these methods are task specific and may not be enough to discriminate subtle inter- and intra-class variations.

Recently, deep learning has been used increasingly for big data analysis and has achieved enormous success in a variety of computer vision tasks such as object detection [25], face recognition [26], image classification [27], image segmentation [28]. Based on these promising results, there have been increasing efforts to use them for HSI classification. When compared to the handcrafted feature learning methods, deep learning techniques can automatically extract informative features through a set of hierarchical layers, with each layer extracting different features. Specifically, the earlier layers extract primitive features while the deeper layer extract more complex and abstract features that are robust to the nonlinear input data. There have been many one-dimensional deep learning architectures such as stacked auto-encoders [29, 30], restricted Boltzmann machine (RBM), and its extension deep belief network (DBN) [31], recurrent neural network [32] proposed in the literature. These networks equipped with fully connected layers have an undesirable consequence of having to train a large number of parameters with a limited number of available labeled training samples for remote sensing HSI image classification [14, 33]. Further, these one-dimensional vector-based feature alignment networks also lead to loss of structural information since they do not consider the spatial correlation existing between the neighboring pixels.

Very recently, deep learning-based methods for spectral-spatial classification has been proposed. A 3D convolutional neural network (CNN) with L2 regularization for deep feature extraction and classification was proposed in [34]. To overcome the sub-optimal performance due to limited number of training samples, residual learning with deeper and wider networks for HSI was introduced in [35]. A self-improving CNN model, which combines the fractional-order Darwinian particle swarm optimization algorithm with CNN, for optimally selecting the best set of bands to train the CNN has been proposed in [36]. Further, a deep learning method that combines the traditional 3D convolution to join spectral-spatial information processing and 1D convolution for spectral signature processing have been introduced in [37]. While these networks achieve good results, they still face some challenges such as 1) heavy models: the remarkable results achieved by deep learning methods is due to the use of deeper and wider models which require a large number of parameters to learn the complicated representations of the data [38] and require a large amount of labeled data; 2) high-dimensional data: it becomes difficult to effectively learn the low level to the highest level semantic interpretations/features. Even though CNNs have traditionally been used for this purpose, the problem still remains very challenging [37].

Focusing of the challenges mentioned above, a novel quaternion hyperspectral image classification network (QHIC Net), which represents the hyperspectral data in the quaternion domain, is proposed. This establishes a new paradigm for hyperspectral data processing by offering a convenient way to capture both the local and the global spectral-spatial dependencies. Specifically, this neural network architecture enables learning of a) the internal correlations between the multiple components of the multidimensional spectral signatures utilizing the quaternion algebra, and b) the global dependencies that describe the contextual connections between the entities using the network architecture. The proposed method also reduces the number of parameters required to learn the high dimensional data representations. The main contributions of the paper are summarized below.

- 1) A new paradigm for processing hyperspectral image data utilizing quaternions has been provided.
- 2) The network, utilizing quaternion algebra, can exploit the interrelationships between the spectral bands of the hyperspectral data along with spatio-spectral processing.
- 3) The proposed work is one of the first attempts to successfully utilize quaternion algebra specifically for deep learning-based hyperspectral image classification.

The rest of the paper is organized as follows. Section 2 presents the background of the quaternion theory and quaternion neural networks. Section 3 describes the application of the quaternion theory for hyperspectral data and the proposed QHIC network. In section 4, the network setup, experimental results, and comparison with other approaches are provided. Finally, Section 5 summarizes the contributions of this work and discusses future directions.

2. BACKGROUND

A quaternion is a hypercomplex number, which is an extension of the real and complex domain and was first described by Hamilton in 1843. Mathematically, a quaternion Q in the four-dimensional domain of $\mathbb{H}$ can be represented as:

$$Q = r1 + xi + yj + zk, \quad r, x, y, z \in \mathbb{R}, \quad (1)$$

$\mathbb{R}$ stands for real value domain, and thus r, x, y , and z are all real numbers. $1, \mathbf{i}, \mathbf{j}$, and $\mathbf{k}$ are the quaternion unit vectors. In a quaternion Q , r is the real part ($\text{Re}(Q)$) while $xi + yj + zk$ is the imaginary part with $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$. If $r = 0$, then Q is called a pure quaternion. Similar to the real and complex numbers, a series of operations can be defined for quaternions:

- Addition or subtraction: $Q_1 \pm Q_2 = (r_1 \pm r_2) + (x_1 \pm x_2)\mathbf{i} + (y_1 \pm y_2)\mathbf{j} + (z_1 \pm z_2)\mathbf{k}$. (2)

- Scalar multiplication: $\lambda Q = \lambda r + \lambda x\mathbf{i} + \lambda y\mathbf{j} + \lambda z\mathbf{k}$. (3)

- Conjugation: $Q^* = r1 - xi - yj - zk$. (4)

- Norm: $|Q| = \sqrt{QQ^*} = \sqrt{r^2 + i^2 + j^2 + k^2}$. (5)

- Quaternion multiplication:
$$Q_1 \otimes Q_2 = (r_1 r_2 - x_1 x_2 - y_1 y_2 - z_1 z_2) + (r_1 x_2 + x_1 r_2 + y_1 z_2 - z_1 y_2)\mathbf{i} + (r_1 y_2 - x_1 z_2 + y_1 r_2 + z_1 x_2)\mathbf{j} + (r_1 z_2 + x_1 y_2 - y_1 x_2 + z_1 r_2)\mathbf{k}$$
 (6)

The proposed QHIC network will utilize these operations. Quaternion convolution can be obtained by extending the quaternion multiplication or Hamilton product criteria. Unlike the real-valued multiplication, it is to be noted that, quaternion space multiplication is not commutative, implying $Q_1 \otimes Q_2 \neq Q_2 \otimes Q_1$.

2.1 Quaternion convolution

The quaternion convolution $O = W * I$, as described in [39], can be obtained by convolving the quaternion weight matrix with the input vector. This is equivalent to the Hamilton product of the quaternion weight $W = W_r + W_x\mathbf{i} + W_y\mathbf{j} + W_z\mathbf{k}$, with the input quaternion $I = I_r + I_x\mathbf{i} + I_y\mathbf{j} + I_z\mathbf{k}$ and is defined as follows:

$$W * I = (W_r * I_r - W_x * I_x - W_y * I_y - W_z * I_z) + (W_x * I_r + W_r * I_x - W_z * I_y + W_y * I_z)\mathbf{i} + (W_y * I_r + W_z * I_x + W_r * I_y - W_x * I_z)\mathbf{j} + (W_z * I_r - W_y * I_x + W_x * I_y + W_r * I_z)\mathbf{k}, \quad (7)$$

and can be represented in the matrix form as follows:

$$\begin{bmatrix} W_r & -W_x & -W_y & -W_z \\ W_x & W_r & -W_z & W_y \\ W_y & W_z & W_r & -W_x \\ W_z & -W_y & W_x & W_r \end{bmatrix} * \begin{bmatrix} I_r \\ I_x \\ I_y \\ I_z \end{bmatrix} = \begin{bmatrix} O_r \\ O_x \\ O_y \\ O_z \end{bmatrix} \quad (8)$$

As seen in eq (7) & (8), Hamilton product allows a quaternion neural network to exploit the interrelationship between the features of the quaternion. Notice that the result of the quaternion convolution, as stated in eq (8), produces a unique linear combination of all the axes for the result of a single axis. This is the resultant of the Hamilton product, which forces each axis of the kernel (weight matrix) to interact with each axis of the quaternion hyperspectral image. This strategy helps in learning complicated features while suppressing the degrees of freedom of the model's training parameter. It can be seen that the degree of freedom of the weight matrix W is 4, while the real-valued neural network would have a parametrization of $4 \times 4 = 16$, which leads to a 4-fold saving. Furthermore, the quaternion convolution can be viewed as the result of adding and subtracting four ordinary convolutions. This significantly simplifies the quaternion convolution operation and can be easily implemented as neural network architecture. The weight initialization recommendations given in [39] are followed for the proposed network in this paper.

3. QUATERNION HYPERSPECTRAL IMAGE CLASSIFICATION NETWORK

This section defines the ways in which hyperspectral data can be represented as a quaternion and its interpretations, the internal hyperspectral quaternion representations, and the neural network architecture.

3.1 Hyperspectral quaternion representation

The proposed QHIC network utilizes the quaternion extension of the real-valued hyperspectral data along with the real-valued matrices to perform quaternion operations. To achieve this, the first step is to convert the hyperspectral input data to the quaternion domain. In the traditional color image case, each of the three channels is associated with the three imaginary axes of the quaternion space and having either zeros or the gray levels related to the real axis. Similarly, in the hyperspectral case, the total number of bands or the depth of the hyperspectral data cube can be divided into four groups, and each group can be associated with each of the axes of the quaternion domain. In case the total number of bands is not divisible by 4, it is padded with additional zero-bands. Conceptually, the hyperspectral data split is very similar to the color image quaternion extension, with an exception in the number of channels per group. This is illustrated in Figure 1, where, in the color image instance, each group (Red, Green, Blue) has a single response (channel), ignoring the cross-talk from the neighboring group. In contrast, in the hyperspectral instance, each group has multiple responses (channel). Another approach for representing the hyperspectral data in the quaternion space is visualized in Figure 1 (c). This approach combines unrelated wavelengths into a single group. In other words, all the blue color bands will be combined into a single group associated with the real axis. In this particular illustration, this would combine 300 nm, 480 nm, and 730 nm in a single group. In contrast, the approach modeled in Figure 1 (b), related wavelengths are grouped, and it follows the physical representations of a hyperspectral image and helps in extracting the interrelationships between these physically related groups. The approach illustrated in Figure 1 (b) is used in this paper.

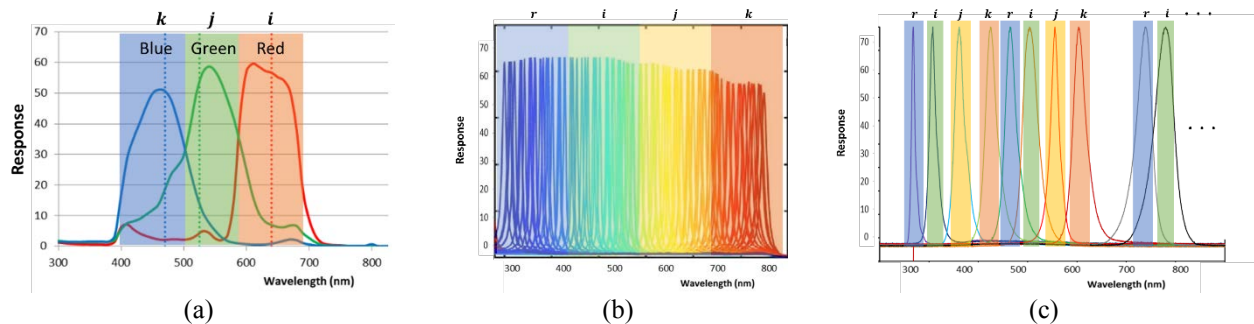


Figure 1: Visualization of color camera (a) and hyperspectral camera (b & c) wavelength split for quaternion representation.

The proposed QHIC network is the quaternion extension of the real-valued hyperspectral image classification network defined in [37]. In the quaternion convolution layer (for both 3D and 1D), all the parameters are quaternions, including inputs, outputs, and weights. The quaternion convolution algebra as defined in eq (7) and eq (8) will be utilized and consequently, the input vector of size N will have to be divided into four parts following the method illustrated in Figure 1 (b). The first part will correspond to r , the second part will correspond to xi , the third to yj , and the fourth to zk , and the quaternion will be defined as $Q = r1 + xi + yj + zk$.

3.2 A general overview of the architecture

A joint spatio-spectral model that can not only merge the spatial information with the spectral signature but can also exploit the inter-band correlation of the hyperspectral data is presented in this paper. The proposed architecture uses a 3-D quaternion-based convolution that simultaneously processes the spatial and spectral data to make maximum utilization of the rich information present in the hyperspectral data with low parameter cost. To perform 3D convolutions, each pixel of the quaternion domain hyperspectral image will be associated with an $n \times n$ spatial neighborhood and f spectral bands such that $f \leq \text{total number of bands}/4$, to account for the quaternion split. Thus, each pixel can be considered as a volume defined by $n \times n \times f$. An illustrative example of the 3D quaternion convolution is provided in Figure 2. As seen in the illustration, a $3 \times 3 \times 3$ quaternion 3D kernel is used for convolution. Notice that each axis of the volumetric kernel (r, i, j, k -axes) interact with each axis of the input volumetric quaternion image data. Such an approach of reusing the filters in a unique combination for each axis is the core idea of learning the interrelationships between these axes.

An overview of the QHIC network architecture is presented in Figure 3. To have an efficient and deep representation of the input hyperspectral image, various blocks of convolutional layers are stacked on top of each other. To start with, a set of 3D quaternion convolutional layers are used to handle the volumetric input data. These layers encompass 3D quaternion kernels that perform quaternion convolution on the height, width, and the depth axis of the input. This stack of 3D layers is followed by a set of 1D quaternion convolutional layers that focus on the spectral content discarding the spatial neighborhood. This is followed by an average pooling and fully connected layer and ends with a softmax classifier. In effect, the architecture produces 3D features that are gradually transitioned to the 1D feature. To reduce from 3D volume to 1D vectors, careful considerations of kernel strides and paddings are utilized. Stride represents the consecutive positions of the convolutional kernel and padding is used to ensure that the boundary conditions are handled to make sure that the

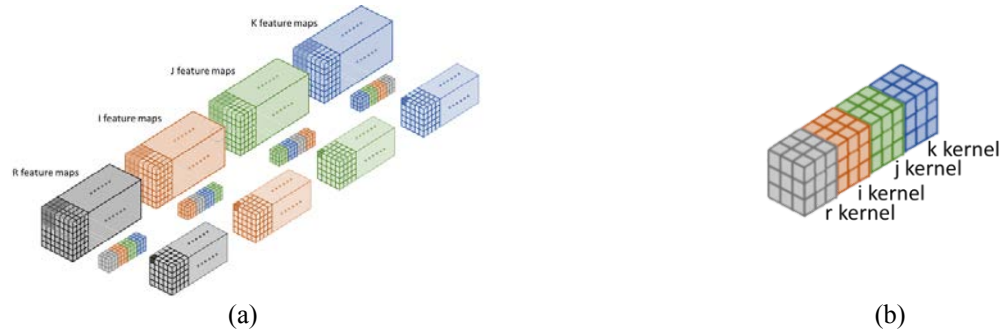


Figure 2: Visualization of (a) 3D quaternion convolution for hyperspectral data and (b) a quaternion 3D kernel of size $3 \times 3 \times 3$.

convolution output retains the size of the input. Each layer l of the 3D quaternion convolution is characterized by k_l number of quaternion kernels of size $(m_l \times m_l \times f_l)$, where $m_l \leq n$ and $f_l \leq f$. These convolutional layers serve the dual purpose of acting as the spatio-spectral 3D quaternion convolution with stride one and serving as the pooling layer with stride ≥ 2 . Further, a combination of removal of padding along the spatial dimension, dilation, and stride leads the way to progressively reduce the data dimension and transitions to the 1D vector. This is fed to the 1D quaternion convolutional layers, each with p_l kernels. Finally, the output 1D vectors are fed to a fully connected layer, which ends with a softmax classifier. The output of the classifier is equal to the number of target classes. An example of the proposed QHIC network architecture is illustrated in Figure 4, where $f = 104$, $n = 5$, $f_l = 3$ or 2 , and $m_l = 3$ or 1 . Strides are alternated between 1 and 2 to create the pooling effect after every convolution.

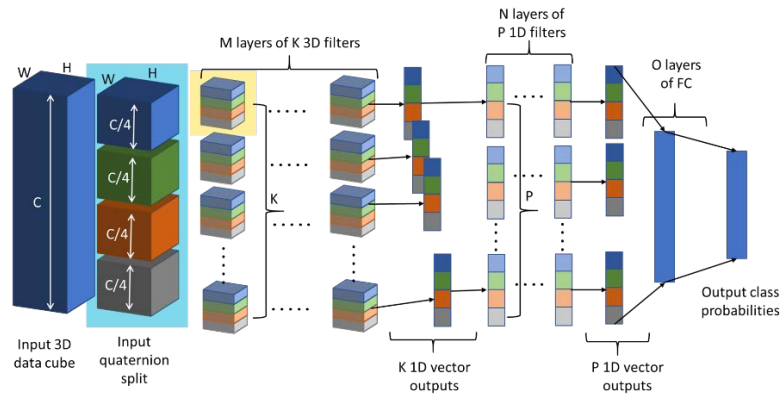


Figure 3: Overview of the Quaternion Hyperspectral Image Classification architecture.

The architecture utilizes the Adam optimizer with weight decay was selected for this use case after testing various other optimizers. ReLU activations, along with a 0.5 probability dropout on the output of the 1D quaternion convolution layer with quaternion weights initialization as specified in [39] was used. The learning rate was initially set to 0.001 to rapidly search for the local minimum and used the reduce learning rate on plateau scheduler to adjust the learning rate during training.

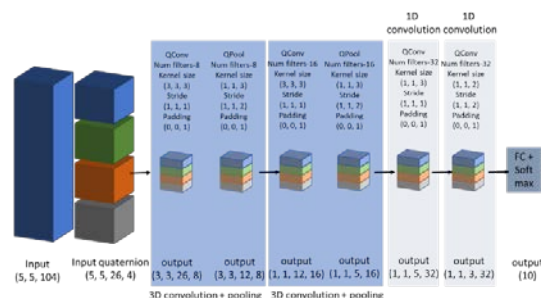


Figure 4: Example illustration of the evolution of feature shapes of each layer.

4. EXPERIMENTS AND ANALYSIS

In this section, the experiments conducted on various hyperspectral datasets using the proposed QHIC network are presented and compared with other state-of-the-art approaches.

4.1 Datasets

Experiments are conducted on the Pavia University, Pavia center datasets, which were acquired by the reflective optics system imaging spectrometer (ROSIS) sensor and the Indian Pines dataset which are collected by the airborne visible/infrared imaging spectrometer (AVIRIS) sensor and the images are as shown in Figure 5. The Pavia University dataset comprises of the data captured over the Engineering School at the University of Pavia and presents nine classes, including water, trees, asphalt, self-blocking bricks, bitumen, tiles, shadows, meadows, and bare soil. The image resolution is 610 x 340 pixels with a spatial resolution of 1.3 meters per pixel. The image consists of 103 spectral bands ranging from 430 to 860 nm. The Pavia Center data comprises of 102 spectral band datasets with 1096 x 1096 pixels and a spatial resolution of 1.3 meters per pixel. This dataset also consists of nine classes, including asphalt, meadows, gravel, trees, painted metal sheets, bare soil, bitumen, self-blocking bricks, and shadows. The Indian pines dataset was acquired over the Indian Pines agricultural site in northwestern Indiana. It comprises of 220 spectral bands ranging from 400 to 2500 nm, of which 20 spectral bands cover the regions of water absorption and have not been considered for the experiments. The image consists of 145 x 145 pixels with a spatial resolution of 20 meters per pixel. It has 16 different classes, including alfalfa, corn-no-till, corn-min, corn, grass-pasture, grass-trees, grass-pasture-mowed, hay-windrowed, oats, soybean-no-till, soybean-min-till, soybean-clean, wheat, woods, building-grass-trees, stone-steel-towers.

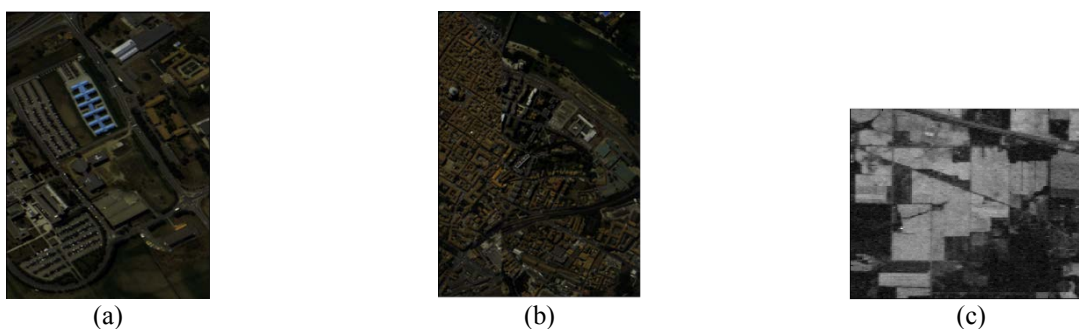


Figure 5: Hyperspectral dataset. (a) Pavia University, (b) Pavia Center, (c) Indian Pines

4.2 Different architectures

After conducting extensive sets of experiments, it was found that doubling the number of filters after every 3D pooling layer and doubling the filters for the first 1D convolution layer gave the best results. Figure 6 shows the overall architecture with the variations in the depth of the network, where a convolution layer with stride as one is denoted by QConv and convolution layer with stride as two is denoted by QPool and the number of filters per layer is shown in [num filters]. The schemes a-d represent variations in network depth with 4, 6, and two 8 layers deep network and are as detailed below.

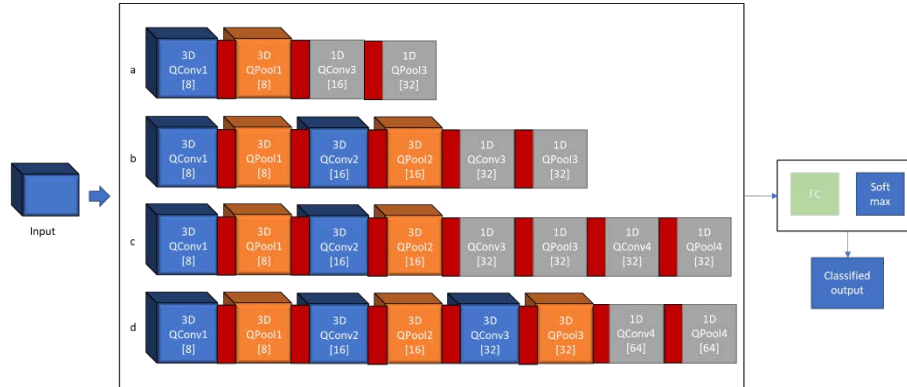


Figure 6: 3D quaternion convolution architecture overview.

- 1) Four-layer network (a): This network is inspired by the real-valued network described in [37] and is created by two 3D layers and two 1D layers along with FC layer and softmax. The two layers in both the 3D and 1D instance are alternate between convolution layers with stride as 1 and 2, respectively, to play the role of conv layer and pooling layer.
- 2) Six-layer network (b): To test the performance of deeper architectures, a six-layer network was created with four 3D layers and two 1D layers. The duality of conv/pool layers is applied sequentially to the network, and the number of filters is doubled after every 3D pooling layer and at the beginning of the 1D layer.
- 3) Eight-layer network (c and d): To estimate whether the addition of more spatospectral 3D layers or the addition of more spatially localized and spectrally dominant 1D layers will improve the performance of the system, two eight-layer networks were developed. In the first network, four 3D layers and four 1D layers were used. Similarly, in the second network, six 3D layers and two 1D layers were used. In both the networks, the filter size is doubled after every 3D pooling layer, and at the beginning of the 1D layer and the dual conv/pool layer scheme is maintained.

Additionally, to estimate the tradeoff between deeper vs. wider networks, each of the above-mentioned network schemes (a-d) is tested with 8, 12, and 16 as the initial number of filters. This study will help in understanding the selection of the number of filters or the width with respect to the network depth to maximize accuracy and minimize cost.

4.3 Experiments and results

Table 1: Accuracy level of the four-layer architecture (network a)

Spatial neighborhood	Initial number of filters	Number of parameters [Pavia, Indian Pines]	Pavia University	Pavia Center	Indian Pines
3 x 3	8	1794, 1913	64.750 %	93.664 %	52.202%
5 x 5	8	1794, 1913	69.227 %	95.007 %	47.603%
7 x 7	8	1794, 1913	59.799 %	94.450 %	45.002%
3 x 3	12	3682, 3857	76.142 %	96.980 %	50.138%
5 x 5	12	3682, 3857	73.950%	96.569 %	57.777%
7 x 7	12	3682, 3857	74.292%	96.435 %	50.642%
3 x 3	16	6234, 6465	74.545%	97.250 %	54.266%
5 x 5	16	6234, 6465	76.529%	97.660 %	51.406%
7 x 7	16	6234, 6465	76.329%	96.336 %	52.869%

Experiments were conducted on the above mentioned 4-, 6-, 8-layer architectures. All tests were executed on an octa-core Intel i9-9900k CPU laptop with Nvidia RTX 2080 GPU. The results presented from the proposed architectures were obtained using the PyTorch library [40]. For fair comparisons, Pavia University and Pavia Center have been trained with 5% of data and 30% of data for Indian pines. Also, since the number of channels in the input datasets varies for the Pavia dataset and Indian Pines dataset, the number of parameters varies slightly and is indicated the tables given below.

Table 2: Accuracy level of the six-layer architecture (network b)

Spatial neighborhood	Initial number of filters	Number of parameters [Pavia, Indian Pines]	Pavia University	Pavia Center	Indian Pines
3 x 3	8	2658, 2889	76.307%	97.489%	64.733%
5 x 5	8	2658, 2889	78.541%	97.309%	66.699%
3 x 3	12	5482, 5825	78.487%	97.707%	62.084%
5 x 5	12	5482, 5825	77.939%	97.396%	62.051%
3 x 3	16	9306, 9761	76.575%	97.893%	81.212%
5 x 5	16	9306, 9761	76.888%	97.833%	80.546%

Table 3: Accuracy level of the eight-layer architecture (network c)

Spatial neighborhood	Initial number of filters	Number of parameters [Pavia, Indian Pines]	Pavia University	Pavia Center	Indian Pines
3 x 3	8	5346, 5577	75.709%	97.567%	68.747%
5 x 5	8	5346, 5577	80.121%	97.753%	87.096%
3 x 3	12	11434, 11777	81.420%	97.780%	87.177%
5 x 5	12	11434, 11777	83.568%	97.680%	90.834%
3 x 3	16	19802, 20257	69.216%	97.959%	84.690%
5 x 5	16	19802, 20257	85.694%	98.313%	87.551%

Table 4: Accuracy level of the eight-layer architecture (network d)

Spatial neighborhood	Initial number of filters	Number of parameters [Pavia, Indian Pines]	Pavia University	Pavia Center	Indian Pines
3 x 3	8	10018, 10473	79.582%	98.448%	84.105%
5 x 5	8	10018, 10473	87.486%	97.130%	73.980%
3 x 3	12	21706, 22385	77.867%	97.773%	76.418%
5 x 5	12	21706, 22385	89.464%	98.270%	86.413%
3 x 3	16	37850, 38753	89.267%	98.426%	86.348%
5 x 5	16	37850, 38753	89.594%	98.469%	91.971%

The four-layer architecture results have been presented in *Table 1*. The initial number of filters refer to the number of output filters for the first quaternion convolution. The number of filters for the following stages can be easily inferred by following "filter size is doubled after every 3D pooling layer and at the beginning 1D layer". The spatial neighborhood size is an essential factor to be determined and is highly dependent on the data. As seen in this table for some dataset, a spatial neighborhood of 3 x 3 works well, while for others, 5 x 5 works well. In general, 7 x 7 was found to be an expanded area and did not yield very impressive results. Similarly, tables 2-4 gives the accuracy levels for the various network

architectures (b-d). As seen in these tables, the deeper network gives better results than the shallower ones; however, they have higher parametric cost. Also, comparing network c and network d, the addition of the 3D quaternion convolutional layer adds more value than adding a 1D layer. To decrease the parametric cost of deeper network architectures, 1 x 1 x 3 pool layers along with a fewer number of filters were employed, and the results have been tabulated in *Table 5*.

Table 5: Accuracy levels of the squeezed architectures (network a-d)

network	Spatial neighborhood	Initial number of filters	Number of parameters [Pavia, Indian Pines]	Pavia University	Pavia Center	Indian Pines
a	5 x 5	12	870, 961	72.346%	96.486%	55.713%
	5 x 5	16	1218, 1337	74.336%	96.884%	51.081%
b	3 x 3	12	2654, 2745	75.404%	97.091%	65.887%
	3 x 3	16	3658, 3777	76.324%	97.843%	61.498%
c	5 x 5	12	4202, 4293	79.073%	98.256%	83.732%
	5 x 5	16	6362, 6481	79.902%	98.161%	78.807%
d	5 x 5	16	7570, 7689	82.697%	98.321%	83.927%

Table 6: Comparison with state-of-the-art methods

network	Spatial neighborhood	Initial number of filters	Number of parameters	Pavia University	Pavia Center	Indian Pines
d	5 x 5	16	37850	89.594%	-	-
d	5 x 5	16	7570	-	98.321%	-
d	5 x 5	16	38753	-	-	91.971%
b*	5 x 5	12	5482	95.111%	-	-
b*	5 x 5	8	2658	-	98.111%	-
[35]	-	-	610600	96.73%	98.88%	93.61
[41]	-	-	81408	92.56%	-	-
[42]	-	-	-	92.99 $\pm$ 2.02%	-	86.62 $\pm$ 2.36%
[43]	-	-	29890	94.57%	98.52%	-
[14]	-	-	-	-	-	85.76%
[37]	5 x 5	-	6862	97.2%	-	-
[37]	3 x 3	-	3681	-	98.9%	-
*indicates that 10% of the total available samples were used for training.						

Finally, comparisons with other state-of-the-art networks have been tabulated in *Table 6*. As seen in the table, the proposed QHIC network is producing results on par with other methods, and it utilizes far fewer parameters. The initial hypotheses that the network is robust to the amount of data available for training have been proved incorrect. As seen in the table, when training with 5% of data, the results for Pavia University and Indian Pines will not give very high accuracy. However, when the training data is increased to 10%, the accuracy increases sharply. This is indicative that the models need to be more robust to produce highly accurate results.

5. CONCLUSION

Hyperspectral image classification requires careful consideration of both spatial and spectral components. The proposed QHIC network, with its ability to process volumetric data and the ability to extract the correlation between the spectral bands helps in accurately classifying with a very low parametric cost. One of the primary concerns now is to further the research into the use of quaternions for hyperspectral data and enhance the created models to be more robust to the shortcomings of the training data. As future work, the preliminary quaternion network introduced in this work can be extended to other network architectures like Residual and Dense networks and explore the possibility of extending the quaternion hyperspectral space to tackle other issues such as hyperspectral band selection, unmixing, target and anomaly detection, and super-resolution.

REFERENCES

- [1] P. Ghamisi, M. Dalla Mura, and J. A. Benediktsson, "A survey on spectral-spatial classification techniques based on attribute profiles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 5, pp. 2335-2353, 2014.
- [2] J. Li, M. Khodadadzadeh, A. Plaza, X. Jia, and J. M. Bioucas-Dias, "A discontinuity preserving relaxation scheme for spectral-spatial hyperspectral image classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 2, pp. 625-639, 2015.
- [3] C. Wu, B. Du, and L. Zhang, "Slow feature analysis for change detection in multispectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 5, pp. 2858-2874, 2013.
- [4] B. Demir, F. Bovolo, and L. Bruzzone, "Updating land-cover maps by classification of image time series: A novel change-detection-driven transfer learning approach," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 1, pp. 300-312, 2012.
- [5] L. G. Olmanson, P. L. Brezonik, and M. E. Bauer, "Airborne hyperspectral remote sensing to assess spatial distribution of water quality characteristics in large rivers: The Mississippi River and its tributaries in Minnesota," *Remote Sensing of Environment*, vol. 130, pp. 254-265, 2013.
- [6] D. Haboudane, J. R. Miller, E. Pattey, P. J. Zarco-Tejada, and I. B. Strachan, "Hyperspectral vegetation indices and novel algorithms for predicting green LAI of crop canopies: Modeling and validation in the context of precision agriculture," *Remote sensing of environment*, vol. 90, no. 3, pp. 337-352, 2004.
- [7] Y. Ge, J. A. Thomasson, and R. Sui, "Remote sensing of soil properties in precision agriculture: A review," *Frontiers of Earth Science*, vol. 5, no. 3, pp. 229-238, 2011.
- [8] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Deep learning for hyperspectral image classification: An overview," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6690-6709, 2019.
- [9] S. Delalieux, B. Somers, B. Haest, T. Spanhove, J. V. Borre, and C. Mùcher, "Heathland conservation status mapping through integration of hyperspectral mixture analysis and decision tree classifiers," *Remote sensing of environment*, vol. 126, pp. 222-231, 2012.
- [10] J. Ham, Y. Chen, M. M. Crawford, and J. Ghosh, "Investigation of the random forest framework for classification of hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, pp. 492-501, 2005.
- [11] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Transactions on geoscience and remote sensing*, vol. 42, no. 8, pp. 1778-1790, 2004.
- [12] M. Fauvel, J. A. Benediktsson, J. Chanussot, and J. R. Sveinsson, "Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, no. 11, pp. 3804-3814, 2008.
- [13] L.-Z. Huo and P. Tang, "Spectral and spatial classification of hyperspectral data using SVMs and Gabor textures," in *2011 IEEE International Geoscience and Remote Sensing Symposium*, 2011: IEEE, pp. 1708-1711.
- [14] L. Mou, P. Ghamisi, and X. X. Zhu, "Unsupervised spectral-spatial feature learning via deep residual Conv-Deconv network for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 1, pp. 391-406, 2017.
- [15] J. Li, J. M. Bioucas-Dias, and A. Plaza, "Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 11, pp. 4085-4098, 2010.

- [16] B. Du and L. Zhang, "Target detection based on a dynamic subspace," *Pattern Recognition*, vol. 47, no. 1, pp. 344-358, 2014.
- [17] G. Licciardi, P. R. Marpu, J. Chanussot, and J. A. Benediktsson, "Linear versus nonlinear PCA for the classification of hyperspectral data based on the extended morphological profiles," *IEEE Geoscience and Remote Sensing Letters*, vol. 9, no. 3, pp. 447-451, 2011.
- [18] S. Prasad and L. M. Bruce, "Limitations of principal components analysis for hyperspectral target recognition," *IEEE Geoscience and Remote Sensing Letters*, vol. 5, no. 4, pp. 625-629, 2008.
- [19] A. Villa, J. A. Benediktsson, J. Chanussot, and C. Jutten, "Hyperspectral image classification with independent component discriminant analysis," *IEEE transactions on Geoscience and remote sensing*, vol. 49, no. 12, pp. 4865-4876, 2011.
- [20] T. V. Bandos, L. Bruzzone, and G. Camps-Valls, "Classification of hyperspectral images with regularized linear discriminant analysis," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 3, pp. 862-873, 2009.
- [21] L. He, J. Li, C. Liu, and S. Li, "Recent advances on spectral-spatial hyperspectral image classification: An overview and new guidelines," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 3, pp. 1579-1597, 2017.
- [22] P. Ghamisi *et al.*, "New frontiers in spectral-spatial hyperspectral image classification: The latest advances based on mathematical morphology, Markov random fields, segmentation, sparse representation, and deep learning," *IEEE geoscience and remote sensing magazine*, vol. 6, no. 3, pp. 10-43, 2018.
- [23] Y. Chen, N. M. Nasrabadi, and T. D. Tran, "Hyperspectral image classification using dictionary-based sparse representation," *IEEE transactions on geoscience and remote sensing*, vol. 49, no. 10, pp. 3973-3985, 2011.
- [24] L. Fang, S. Li, X. Kang, and J. A. Benediktsson, "Spectral-spatial hyperspectral image classification via multiscale adaptive sparse representation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 12, pp. 7738-7749, 2014.
- [25] Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 11, pp. 3212-3232, 2019.
- [26] Y. Sun, D. Liang, X. Wang, and X. Tang, "Deepid3: Face recognition with very deep neural networks," *arXiv preprint arXiv:1502.00873*, 2015.
- [27] T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "PCANet: A simple deep learning baseline for image classification?," *IEEE transactions on image processing*, vol. 24, no. 12, pp. 5017-5032, 2015.
- [28] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481-2495, 2017.
- [29] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE Journal of Selected topics in applied earth observations and remote sensing*, vol. 7, no. 6, pp. 2094-2107, 2014.
- [30] C. Tao, H. Pan, Y. Li, and Z. Zou, "Unsupervised spectral-spatial feature learning with stacked sparse autoencoder for hyperspectral imagery classification," *IEEE Geoscience and remote sensing letters*, vol. 12, no. 12, pp. 2438-2442, 2015.
- [31] Y. Chen, X. Zhao, and X. Jia, "Spectral-spatial classification of hyperspectral data based on deep belief network," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 6, pp. 2381-2392, 2015.
- [32] L. Mou, P. Ghamisi, and X. X. Zhu, "Deep recurrent neural networks for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 3639-3655, 2017.
- [33] P. Ghamisi and J. A. Benediktsson, "Feature selection based on hybridization of genetic algorithm and particle swarm optimization," *IEEE Geoscience and remote sensing letters*, vol. 12, no. 2, pp. 309-313, 2014.
- [34] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 6232-6251, 2016.
- [35] H. Lee and H. Kwon, "Going deeper with contextual CNN for hyperspectral image classification," *IEEE Transactions on Image Processing*, vol. 26, no. 10, pp. 4843-4855, 2017.
- [36] P. Ghamisi, Y. Chen, and X. X. Zhu, "A self-improving convolution neural network for the classification of hyperspectral data," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 10, pp. 1537-1541, 2016.
- [37] A. B. Hamida, A. Benoit, P. Lambert, and C. B. Amar, "3-D deep learning approach for remote sensing image classification," *IEEE Transactions on geoscience and remote sensing*, vol. 56, no. 8, pp. 4420-4434, 2018.
- [38] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016.

- [39] T. Parcollet *et al.*, "Quaternion convolutional neural networks for end-to-end automatic speech recognition," *arXiv preprint arXiv:1806.07789*, 2018.
- [40] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, 2019, pp. 8024-8035.
- [41] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *Journal of Sensors*, vol. 2015, 2015.
- [42] B. Fang, Y. Li, H. Zhang, and J. C.-W. Chan, "Hyperspectral images classification based on dense convolutional networks with spectral-wise attention mechanism," *Remote Sensing*, vol. 11, no. 2, p. 159, 2019.
- [43] S. Dwivedi, M. Mandal, S. Yadav, and S. K. Vipparthi, "3D CNN with Localized Residual Connections for Hyperspectral Image Classification," *arXiv preprint arXiv:1912.03000*, 2019.