Quaternion Convolutional Neural Network for Color Image Classification and Forensics

Article *in* IEEE Access · February 2019

DOI: 10.1109/ACCESS.2019.2897000

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Quaternion Convolutional Neural Network for Color Image Classification and Forensics

QILIN YIN[2], JINWEI WANG[1,2,3], (MEMBER, IEEE), XIANGYANG LUO[3], (MEMBER, IEEE), JIANGTAO ZHAI[2], SUNIL KR. JHA[2] AND YUN-QING SHI[4], (FELLOW, IEEE)

[1]Jiangsu Collaborative Innovation Center of Atmospheric Environment and Equipment Technology (CICAEET), Nanjing University of Information Science & Technology, Nanjing, 210044
[2]Department of Computer and Software, Nanjing University of Information Science and Technology, Jiangsu, 210044 China
[3]State Key Laboratory of Mathematical Engineering and Advanced Computing, Henan 450001, China
[4]Department of Electrical Computer Engineering, New Jersey Institute of Technology, New Jersey NJ07102, USA

Corresponding author: Jinwei Wang (e-mail: wjwei_2004@163.com and jwang@njit.edu).

ABSTRACT The convolutional neural network is widely popular for solving the problems of color image feature extraction. However, in the general network, the interrelationship of the color image channels is neglected. Therefore, a novel quaternion convolutional neural network (QCNN) is proposed in this paper, which always treats color triples as a whole to avoid information loss. The original quaternion convolution operation is presented and constructed to fully mix the information of color channels. The quaternion batch normalization and pooling operations are derived and designed in quaternion domain to further ensure the integrity of color information. Meanwhile, the knowledge of the attention mechanism is incorporated to boost the performance of the proposed QCNN. Experiments demonstrate that the proposed model is more efficient than the traditional convolutional neural network and another QCNN with the same structure, and has better performance in color image classification and color image forensics.

INDEX TERMS Quaternion convolutional neural network, quaternion based layers, color image classification, color image forensics, attention mechanism

I. INTRODUCTION

NOWADAYS, as a common and effective information carrier, a color image has penetrated into every corner of social life, which leads to our increasing demand for image processing, e.g., color image classification or color image forensics. Classification techniques enable computers to replace human beings to complete classification tasks. Forensics technology ensures the security of image information. In real life, most color images are stored in JPEG format. If an image is tampered with, it will undergo decompression, and then compressed to form a double JPEG compressed image. Therefore, double JPEG compression is an inevitable process in image tampering. Detecting double JPEG compression provides strong support for image forensics. No matter the color image classification or forensics, the essence of them is to extract the features of the color images, and to classify the images by effective features. Hence, a model that can extract effective features is crucial for various image processing tasks.

The main purpose of the initial convolutional neural networks (CNNs) were to extract features. In the field of computer vision, CNNs have received great success, e.g., AlexNet [1], VGG [2], NIN [3], ResNet [4] and DenseNet [5], etc., which all achieved state-of-the-art performance in many vision tasks [30]–[33]. A traditional convolutional neural network consists of one or several convolutional layers, followed by some fully-connected layers of neurons. Each convolution block usually produces feature maps by four steps, e.g., convolution, batch normalization, non-linear activation, and pooling. Due to the existence of different convolution kernels (filters), the first step can extract hierarchical features that help the network to accomplish image processing tasks better from inputs. The second step is to normalize the inputs of each layer, accelerating the network training and making the network independent of network initialization. The non-linear activation is applied to each element of the inputs, e.g., Relu or TanH function, in the third step. The fourth step is to reduce the size of feature

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2897000, IEEE Access

maps, including mean-pooling or max-pooling. Generally speaking, a complete CNN usually contains several cascaded convolutional layers and ends with a few fully-connected layers followed by a softmax classifier for a classification task.

In the last two years, attention model has been widely used in various types of deep learning tasks such as natural language processing [17]–[20], computer vision [21]–[25] and speech recognition [26], [27]. From the naming of attention model, it is obvious that it borrows from the human vision attention mechanism. Human vision rapidly focuses on the target area by scanning the global image, and then invests more attention to obtain more detailed information of the target and suppress useless information. Just like this, through the attention model, the neural network can autonomously and purposefully select key features and discard redundant features.

In the computer vision field, attention mechanisms can be roughly divided into two types: soft attention [21]–[23] and hard attention [24], [25]. They all are used to learn the distribution of weights, but the difference is that the former does the weighting of all features, while the latter is to discard some features directly and retain the most useful features. In [21], a bottom-up top-down structure is adopted as the main frame of the attention module to expand the receptive field of high-level features and making neural network pay more attention to highly abstract local features. This method is also called as spatial attention. In [22], unlike spatial attention is implemented which selects local significant features from each feature map, but the total number of feature maps remains unchanged and channel attention retains the complete information of a feature map but reduces the number of useful feature maps. The performance of most classical CNNs can be improved when combined with the above methods.

As mentioned above, the traditional CNN framework is becoming more and more mature, but it still has some drawbacks, like when dealing with the color images, general CNNs just treat the RGB three channels as three unrelated feature maps. Although during the convolution process, per convolution kernel sum up the convolution result of different channels as a single output, this still neglects the interrelationship of the RGB three color channels, resulting in the loss of feature information.

Focusing on the question above, we propose a quaternion convolutional neural network (QCNN) model. According to Sangwine [6]–[8], a color image can be represented as a quaternion matrix, each element of which is a pure quaternion consisting of 3-tuple (RGB or LMS). The advantage of this method is that a color can be processed as a hypercomplex number, rather than three separate channels, reserving the interrelationship information between the color channels mostly. So, a color image is circulated and processed in the form of a quaternion matrix in the QCNN. Compared with the traditional CNNs, QCNN not only preserves the vertical relationship between the features, but also maintains the horizontal relationship of the color channels.

Considering that the data format in the network is a quaternion matrix, we reconstructed the convolution block, including the four steps above. Quaternion convolution layer maps the color image directly to the four-dimensional space through quaternion rotation. In the higher dimensional space, the neural network can achieve richer feature information. In general, since the four parts of quaternion contain a real part and three imaginary parts, it can be also treated as a four-dimensional vector. Hence, during the quaternion batch normalization operation, the traditional scalar mean and variance could be replaced by the vector mean and variance. Similarly, we cannot pool four parts separately on account of four parts that should be taken as a unit. The quaternion matrix can be simplified on the basis of the guidance matrix. Accordingly, non-linear activation can be applied to each element of a four-dimensional vector to form the quaternion activation layer.

The essence of attention mechanism is to reuse feature maps, give new weight to feature maps and mark out key features. Regardless of the real domain or hypercomplex, this idea is universal. We try to incorporate an attention mechanism into the QCNN model to improve the proposed model performance further. The following experimental results show that our model can outperform the traditional CNN model and the other QCNN model [9], which has only recently been proposed and is called Pure QCNN by us below.

The rest of the paper is as follows: Section II introduces the quaternion algebra. Section III mainly analyses the key components of the QCNN model, e.g., quaternion convolution, quaternion pooling, quaternion batch normalization, and quaternion attention module. Section IV compares and demonstrates the consequence of the experiment. Section V summarizes this paper.

II. QUATERNION ALGEBRA

The quaternion is an extension of the real and complex domains, including one real part $Re(.)$ and one imaginary part $Im(.)$. Usually, imaginary part has three elements and a quaternion can be defined as:

$$q = \underbrace{a}_{Re(q)} + \underbrace{b\boldsymbol{i} + c\boldsymbol{j} + d\boldsymbol{k}}_{Im(q)} a, b, c, d \in \mathbb{R} \tag{1}$$

$\mathbb{R}$ stands for real value field, therefore, a, b, c and d are all real numbers. $\boldsymbol{i}$, $\boldsymbol{j}$, $\boldsymbol{k}$ are three different imaginary unit vectors and must obey the quaternion rules i.e. $\boldsymbol{i}^2 = \boldsymbol{j}^2 = \boldsymbol{k}^2 = \boldsymbol{ijk} = -1$. In a quaternion, a is the real part ($Re(q)$) while $b\boldsymbol{i}+c\boldsymbol{j}+d\boldsymbol{k}$ is the imaginary part ($Im(q)$). If a equals 0, we can call q is a pure quaternion. Similar to real numbers or complex numbers, quaternion also has some algorithm rules:

- Addition or subtraction: $q_1 \pm q_2 = (a_1 \pm a_2) + (b_1 \pm b_2)\boldsymbol{i} + (c_1 \pm c_2)\boldsymbol{j} + (d_1 \pm d_2)\boldsymbol{k}$
- Conjugation: $q^* = a - b\boldsymbol{i} - c\boldsymbol{j} - d\boldsymbol{k}$
- Norm: $|q| = \sqrt{qq^*} = \sqrt{a^2 + b^2 + c^2 + d^2}$
- Quaternion multiplication:

$$
\begin{aligned}
q_1 q_2 = \quad & (a_1a_2 - b_1b_2 - c_1c_2 - d_1d_2) + \\
& (a_1b_2 + b_1a_2 + c_1d_2 - d_1c_2)\boldsymbol{i} + \\
& (a_1c_2 - b_1d_2 + c_1a_2 + d_1b_2)\boldsymbol{j} + \\
& (a_1d_2 + b_1c_2 - c_1b_2 + d_1a_2)\boldsymbol{k}
\end{aligned} \tag{2}
$$

The proposed model QCNN will use these quaternion operations. In particular, the criterion of quaternion multiplication can be extended to the quaternion convolution that we mention next.

III. COMPONENTS OF QUATERNION CNN

Unlike traditional real-valued neural networks use the real number matrix as the measurement unit of feature map, each feature map is a quaternion matrix in the QCNN. Hence, general neural network components are not incompatible with the quaternion matrix. They should be reconstructed to be suitable for feature maps in the form of a quaternion matrix.

A. QUATERNION CONVOLUTION

Quaternion representation has many applications in signal processing. As shown in [10], quaternion filter has been used in color texture segmentation and can successfully divide an image into regions on the basis of texture. Among the experimentation, the idea of the operation of the quaternion filter can be extended to quaternion convolution. Let $W = A + B\boldsymbol{i} + C\boldsymbol{j} + D\boldsymbol{k}$ be a quaternion convolution kernel matrix, and $X = a + b\boldsymbol{i} + c\boldsymbol{j} + d\boldsymbol{k}$ the quaternion input matrix. When a color image is involved, the real part a is set to 0, and the input matrix becomes the pure quaternion matrix. Similar to the traditional convolution, the quaternion-valued convolution $W \otimes X$ can be defined as follows:

$$
\begin{aligned}
W \otimes X = \quad & (Aa - Bb - Cc - Dd) + \\
& (Ab + Ba + Cd - Dc)\boldsymbol{i} + \\
& (Ac - Bd + Ca + Db)\boldsymbol{j} + \\
& (Ad + Bc - Cb + Da)\boldsymbol{k}
\end{aligned} \tag{3}
$$

When the convolution operation is performed, point-to-point plain multiplication under the cover of mask W is replaced by quaternion multiplication. Where after the convolution formula can be simplified by combining the similar terms, the result is shown in Equation (3), which is also a quaternion matrix and that the real part and the imaginary part of the result matrix can be obtained by adding and subtracting four ordinary convolutions, respectively. This method greatly simplifies the quaternion convolution operation and makes it easier to implement neural network architecture. The entire convolution process can be decomposed as illustrated in Fig. 1. The cross product of $Conv_5$ can be formulated as Equation (4).

$$
\begin{bmatrix} B \\ C \\ D \end{bmatrix} \oplus \begin{bmatrix} b \\ c \\ d \end{bmatrix} = \begin{bmatrix} Cd - Dc \\ Db - Bd \\ Bc - Cb \end{bmatrix} \tag{4}
$$

B. QUATERNION BATCH NORMALIZATION

The essence of the neural network learning process is to learn data distribution. Once the distribution of training data is different from that of test data, the generalization ability of the network is greatly reduced. Furthermore, if the distribution of each batch of training data is different, the network will learn to adapt to the different distribution in each iteration, which will greatly reduce the training speed of the network. Batch normalization was created to solve these constraints. The difference between quaternion batch normalization and traditional batch normalization is that the way to calculate the mean and variance is different. In [28], the quaternion mean and variance have been defined as follows.

$$
\begin{aligned}
QE(x) \quad & = \frac{1}{T}\sum_{i=1}^{T} q_0 + q_1\boldsymbol{i} + q_2\boldsymbol{j} + q_3\boldsymbol{k} \\
& = \bar{q}_0 + \bar{q}_1\boldsymbol{i} + \bar{q}_2\boldsymbol{j} + \bar{q}_3\boldsymbol{k}
\end{aligned} \tag{5}
$$

$$
\begin{aligned}
QV(x) \quad & = \frac{1}{T}\sum_{i=1}^{T} (x - QE(x))(x - QE(x))^* \\
& = \frac{1}{T}\sum_{i=1}^{T} (\Delta q_0{}^2 + \Delta q_1{}^2 + \Delta q_2{}^2 + \Delta q_3{}^2)
\end{aligned} \tag{6}
$$

Here, $QE(x)$ and $QV(x)$ denote the quaternion mean and quaternion variance of x respectively, where $x = q_0 + q_1\boldsymbol{i} + q_2\boldsymbol{j} + q_3\boldsymbol{k}$, $\Delta q_i = q_i - \bar{q}_i$, $i = 0, 1, 2, 3$, $\bar{q}_i$ denotes the mean of q_i. Compared with traditional statistical features, when they are extended to quaternion domain, quaternion mean is still a quaternion, but quaternion variance is similar to the traditional variance as a real number. This character is beneficial to define quaternion batch normalization $QBN(x_i)$.

$$
\begin{aligned}
QBN(x_i) \quad & = \gamma\left(\frac{x_i - QE(x)}{\sqrt{QV(x) + \varepsilon}}\right) + \beta \\
& = \gamma\left(\frac{q_0^i - \bar{q}_0}{\sqrt{QV(x)+\varepsilon}} + \frac{(q_1^i - \bar{q}_1)\boldsymbol{i}}{\sqrt{QV(x)+\varepsilon}} \right. \\
& \left. + \frac{(q_2^i - \bar{q}_2)\boldsymbol{j}}{\sqrt{QV(x)+\varepsilon}} + \frac{(q_3^i - \bar{q}_3)\boldsymbol{k}}{\sqrt{QV(x)+\varepsilon}}\right) \\
& + (\beta_0 + \beta_1\boldsymbol{i} + \beta_2\boldsymbol{j} + \beta_3\boldsymbol{k}) \\
& = \frac{\gamma(q_0^i - \bar{q}_0)}{\sqrt{QV(x)+\varepsilon}} + \beta_0 \\
& + \left(\frac{\gamma(q_1^i - \bar{q}_1)}{\sqrt{QV(x)+\varepsilon}} + \beta_1\right)\boldsymbol{i} \\
& + \left(\frac{\gamma(q_2^i - \bar{q}_2)}{\sqrt{QV(x)+\varepsilon}} + \beta_2\right)\boldsymbol{j} \\
& + \left(\frac{\gamma(q_3^i - \bar{q}_3)}{\sqrt{QV(x)+\varepsilon}} + \beta_3\right)\boldsymbol{k}
\end{aligned} \tag{7}
$$

Here, γ is a scalar that initializes to 1, representing stretch scale, β is a quaternion that initializes to 0, representing shift scale, ε is a non-zero minimum. γ and β are trainable parameters that participate in network weight updates. Finally, quaternion batch normalization can be decomposed into four traditional batch normalization corresponding to the four parts of the quaternion, but they share the same quaternion variance.

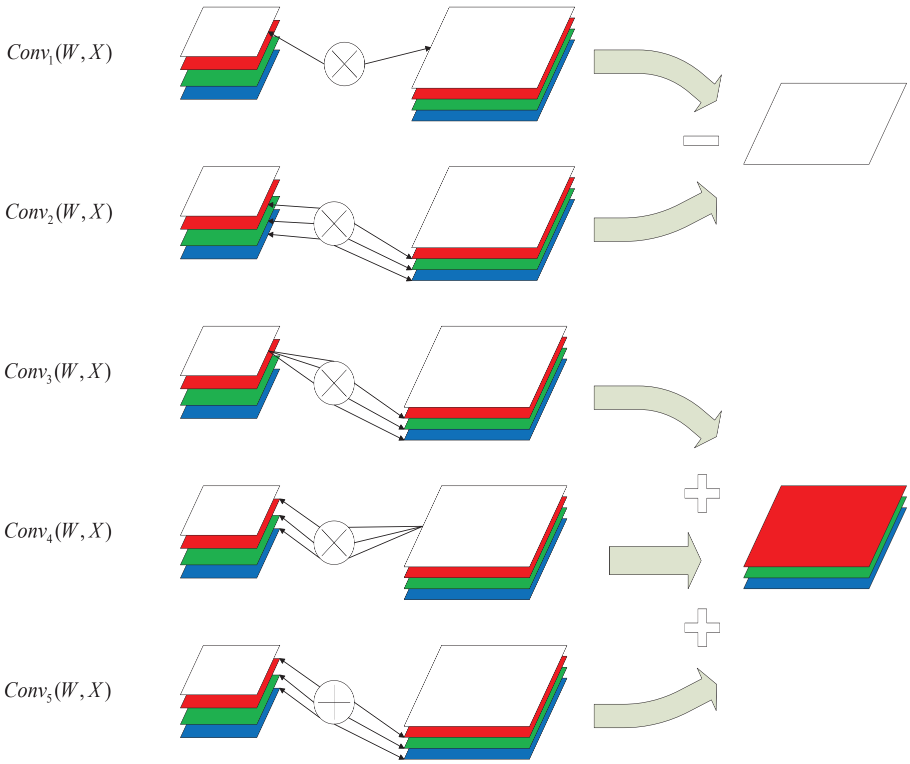


FIGURE 1: Diagrammatic overview of a quaternion kernel matrix convolve a quaternion input matrices. The operator $\otimes$ denotes the normal convolution of two real-valued matrix. The operator $\oplus$ denotes a special convolution that the cross product is used to make the data correspond and convolve. In $Conv_1$ and $Conv_2$, each part of kernel convolve with each part of input correspondingly and separately. The result of real part subtracts the result of three imaginary parts, becoming the real component of the final quaternion. In $Conv_3$, real part of kernel convolve with three imaginary parts of input separately. In $Conv_4$, real part of input convolve with three imaginary parts of kernel separately. In $Conv_5$, three imaginary parts of kernel and input are convolved in a special way that six matrices are paired by the cross product of vectors to produce six conventional convolution. The sum of the result of $Conv_3$, $Conv_4$ and $Conv_5$ yields the imaginary part of final quaternion.

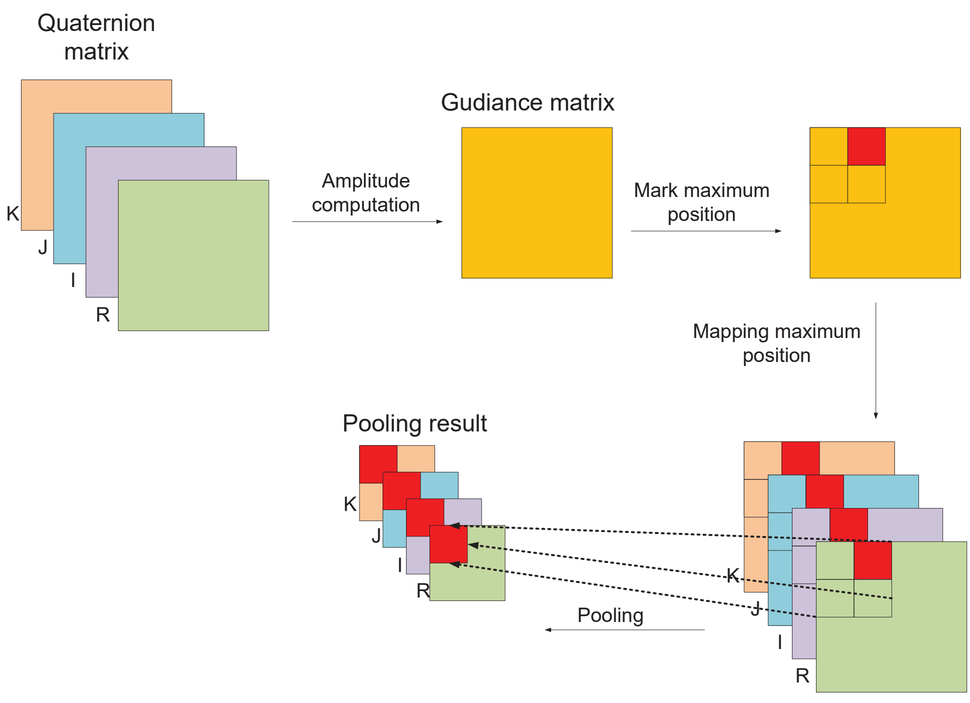


FIGURE 2: Diagrammatic overview of the max pooling of the entire quaternion matrix.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2897000, IEEE Access

C. QUATERNION POOLING

There are many types of pooling layers, e.g., Max-pooling and Mean-pooling, in the real-valued neural network. They all can be extended to hypercomplex domain. For the Mean-pooling operation, pooling the real part and three imaginary parts of the quaternion matrix separately will not affect final pooling result. However, in terms of Max-pooling, if we pool each part individually, this will create a data mess. Because we cannot make sure the position of the maximum of each part is corresponding, we can use some algorithms to get the guidance matrix for the quaternion matrix. Then, according to the max pooling result of the guidance matrix, the four parts of the quaternion matrix can be simplified. Combining the four parts, the max pooling of the entire quaternion matrix is completed finally. The whole process can be shown in Fig. 2.

The amplitude and argument of quaternion are special transformation features of a quaternion. These features decompose the image from the system of polar coordinates, and the amplitude shows the energy distribution of the image, three arguments show the spatial position and texture details of color images. These features provide many options for constructing guidance matrix for different tasks. In this paper, the amplitude is selected as the basis for constructing the guidance matrix.

D. QUATERNION ATTENTION MODULE

Attention mechanism has been proven to work well in general CNNs, it enables the neural network to learn the way humans observe objects, enhancing the interpretability of the neural network. Furthermore, there are many methods to realize different attention mechanisms for various tasks. In the proposed quaternion model, attention module is a small branch that follows the convolution layer to accept the output from the previous layer and calculate the weight of each channel. Finally, according to the weights, good features are stimulated and useless ones are suppressed by element-wise operation on the channel dimension. The whole quaternion attention module can be illustrated in Fig. 3.

Let $T \in H \times W \times 4C$ be the input of the quaternion attention module, the number of channel $4C$ shows the uniqueness of the proposed quaternion network too. Because of the characteristic, we need to modify the attention module in the general network accordingly to accommodate the data format of a quaternion. Firstly, T is transformed by an operation F_{tr} to generate guidance data $S \in H \times W \times C$. When it comes to compressing quaternion data, the transformation features of quaternion provide us different choices just like what we do with the quaternion pooling operation. Secondly, S is pooled by a global average pooling $F_{ga}(.)$ to estimate statistical characteristics of each channel distribution. This process can be formulated as Equation (8).

$$z = \frac{1}{H \times W} \sum_{j=1}^{H} \sum_{i=1}^{W} S(i,j) \quad z \in 1 \times 1 \times C \tag{8}$$

Thirdly, z only represents the channel priority distribution for a mini-batch sample, which is not applicable to the entire training set, let alone the test set. Therefore, multiple full connection operations $F_{de}(.)$ should be added to the quaternion module to train a generalized channel weight mask using a correlation between channels. Equation (9) can be used to describe this process.

$$\begin{aligned} \tilde{z} = \quad & F_{de}(W, z) = \partial(W_2 \sigma(W_1 z)) \\ & W_1 \in \tfrac{C}{r} \times C, W_2 \in C \times \tfrac{C}{r} \end{aligned} \tag{9}$$

$W_1 z$ is a full connection operation, σ is a activation operation that is adopted in Relu activation function, $W_2 \sigma(W_1 z)$ is another a full connection operation, ∂ is also a activation function, but Sigmoid activation function is chosen. r is a hyper-parameter, it determines the proportion of data compression to reduce the amount of computation. Finally, channel mask $\tilde{z}$ is combined with T by operation $F_{mask}(.)$ to highlight beneficial features and inhibit unbeneficial ones. The mathematical expression for this process is Equation (10).

$$T* = F_{mask}(\tilde{z}, T) = \tilde{z} \cdot T \tag{10}$$

E. TYPICAL NONLINEAR LAYER

Activation is essential in traditional CNNs, which provides nonlinear operation to the network and improves the expressive ability of the network model. Similarly, QCNN cannot be without an activation. Many activations have been proposed for quaternion [11], whereas the split activation is applied in the proposed model (the method is mentioned in [12], [13]) is defined as follows:

$$\mathbb{F}(q) = \mathbb{F}(a) + \mathbb{F}(b)\,\boldsymbol{i} + \mathbb{F}(c)\,\boldsymbol{j} + \mathbb{F}(d)\,\boldsymbol{k} \tag{11}$$

where $\mathbb{F}$ corresponding to any standard activation function.

F. FULLY-CONNECTED LAYER

Fully-connected layers act as a classifier like a SVM [14] in the common CNNs, real-valued fully-connected layer can also be used in QCNN. However, when full-connected layers receive quaternion inputs before, quaternion matrices should flatten to be compatible with a real-valued fully-connected layer. Four parts of quaternion input are flattened out firstly and connected later.

IV. EXPERIMENT

In this section, first of all, the basic framework of the proposed QCNN model is introduced. Next, the performance of the components which we reconstruct for the QCNN model is tested. And then, in terms of color image classification, the proposed model is compared with Pure QCNN and its advantage is analyzed in details. Finally, the proposed model is used to detect double JPEG compression to show good performance in color image forensics.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2897000, IEEE Access

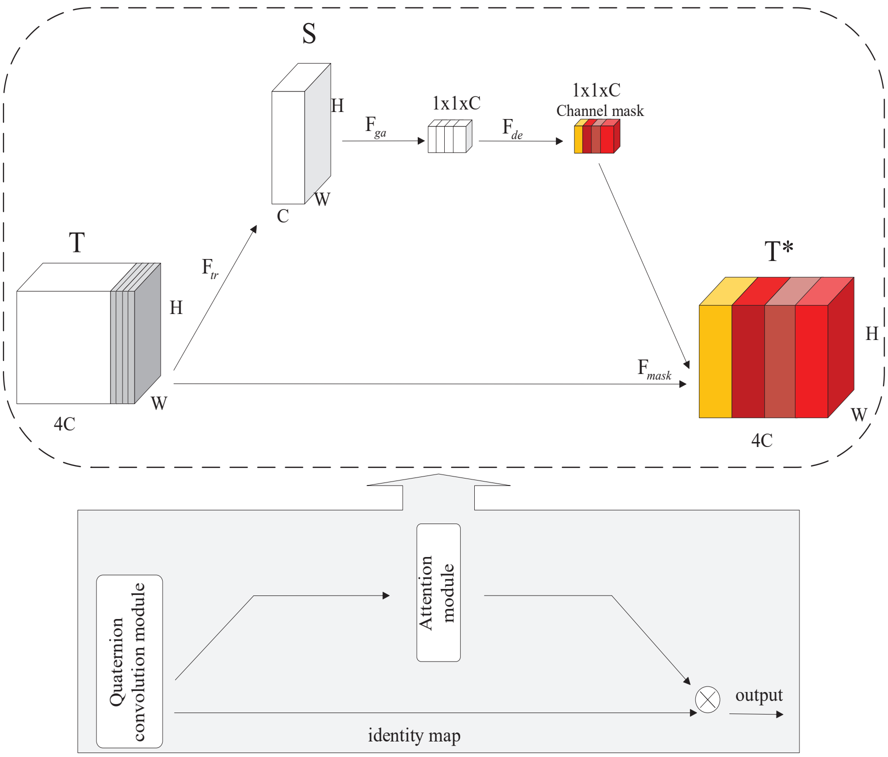


FIGURE 3: Diagrammatic overview of quaternion attention module

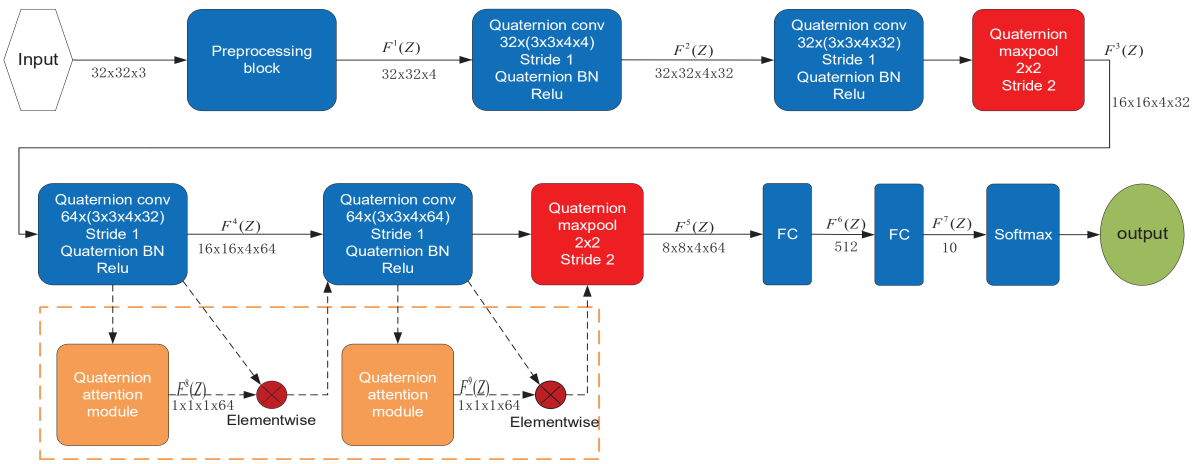


FIGURE 4: Diagrammatic overview of the architecture of the basic QCNN

A. ARCHITECTURE

In order to test the performance of the QCNN component module, we built a basic QCNN. It contains two convolution blocks, two max-pooling layers, and end with two fully-connected layers. Each convolution block is composed of two quaternion-valued convolution layers, each of which contains convolution, batch normalization, and activation operations. ReLU is used as a standard activation function. The whole network structure and specific parameters are shown in Fig. 4. Different from traditional convolutional neural networks, each convolution kernel and feature map in the proposed model is a quaternion matrix with a size of $width \times height \times 4$. Note that the quaternion attention module shown in the dotted orange box is not an essential component of the basic QCNN. It is just a method to improve network performance.

B. RATIONALITY OF COMPONENTS

Some components which we reconstruct are not necessary for the proposed model, but the performance of the proposed model is enhanced by them. Experimental results and analysis are shown below.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2897000, IEEE Access

TABLE 1: The results of the QCNN with different batch normalization

Model	Data set	Test accuracy
QCNN (with traditional batch normalization)	Cifar-10	0.8309
QCNN (with quaternion batch normalization)	Cifar-10	**0.8415**

1) Impact of the quaternion batch normalization

In the actual network training process, four parts of quaternion can also be normalized, respectively. To be compared, the basic QCNNs with traditional batch normalization and quaternion batch normalization are tested on the typical vision task: color image classification. Input data is augmented by shifting and flipping. The whole network is trained with cross entropy loss, optimized by Adam [15] with learning rate set at 0.0001. The training ends at epoch 80. The experimental results are shown in Table 1.

The proposed model with quaternion batch normalization achieves higher accuracy. Although the impact of batch normalization on the distribution of data on the feature map is small, the traditional method still results in the loss of interrelationship between color channels. The quaternion approach does not cause this loss because it always treats the color channel as a whole. The comparison result also reflects the importance of the correlation between color channels.

2) Impact of quaternion attention module

As Fig. 5(a) shows the convergence of the basic QCNN model training on cifar-10, the trend of loss curve in training and testing proves that the network model shows slightly overfitting phenomenon in the second half of the training. Although some training tricks for solving overfitting problems have been applied to networks, e.g., dropout, data augment, overfitting phenomenon still exists. Compared with the traditional CNNs, the proposed model can extract 4 times more features under the same network structure, which is bound to cause the redundancy of features. This is why an attention mechanism is imported into the model. Attention mechanism can help network model sparse rich features to reduce the bad influence of redundant information.

The construction of the attention based QCNN model is to add the quaternion attention module after each convolution layer of the second convolution block on the basis of the original QCNN. Since the shallow convolutional layer of the network only extracts the detailed features of the image, while the deep convolutional layer can extract more abstract high-level features that can be mapped to specific local areas of the image, it is more meaningful to add the attention module in the deep convolutional layer than in the shallow layer. The dotted orange box in Fig. 4 represents the addition of the quaternion attention module. The convergence of the attention based QCNN model is displayed in Fig. 5(b). By contrast, the attention based QCNN model still showed a good training trend in the second half of training. The final test accuracy can be seen in Table 2.

TABLE 2: The test accuracy of the QCNN and the the attention based QCNN

Model	Data set	Test accuracy
QCNN	Cifar-10	0.8415
attention based QCNN	Cifar-10	**0.8537**

TABLE 3: Experiment results in classification tasks

Model	Data set	Test accuracy
Real network	Cifar-10	0.7546
Pure QCNN	Cifar-10	0.7778
QCNN(proposed)	Cifar-10	**0.8415**
attention based QCNN(proposed)	Cifar-10	**0.8537**

C. COLOR IMAGE CLASSIFICATION

For comparison, the structure of the proposed model is shown in Fig. 4 is the same as the Pure QCNN model [9], and the training conditions are the same. In addition, real-valued CNN [16] with the same structure also participates in the comparison.

The experimental accuracy of the those models is given in Table 3, we can find that the proposed QCNN model not only performs better than the traditional CNN in terms of the classification task, but also better than Pure QCNN. Pure QCNN just uses the quaternion tool to rotate and stretch the color image, ignoring the real part of the quaternion, but the image transformation process still exists in three-dimensional space. More restrictions have been added to the entire network, and higher requirements have been put forward for the initialization of network weights, too. By contrast, the proposed model makes full use of the quaternion characteristic, mapping the color image into four- dimensional space to achieve high-level abstract features. It also has higher degrees of freedom without special weight initialization and parameter adjustment rules.

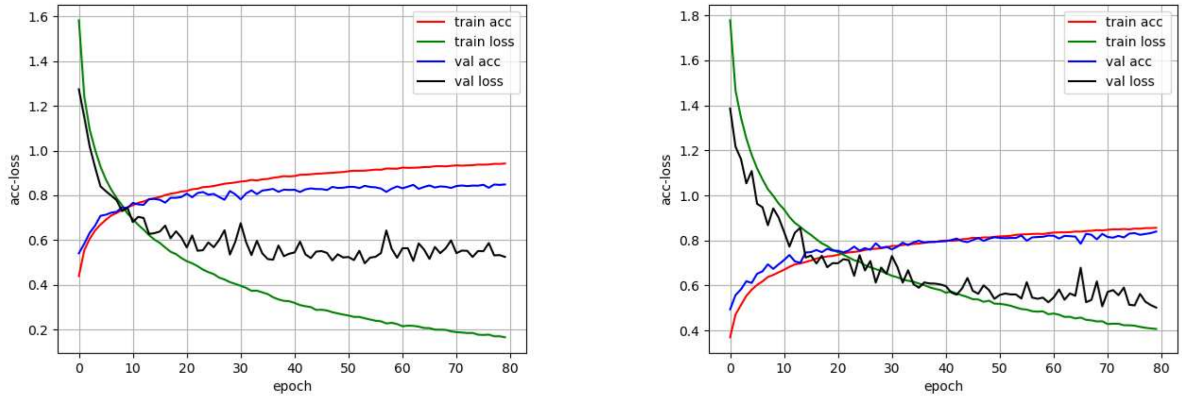
D. COLOR IMAGE FORENSICS

The double JPEG compression detection, which is one of the color image forensics is selected as a test experiment. Different from the classification task, the basic QCNN model should be changed to adapt the forensics task. The new QCNN model contains seven convolution layers, two max-pooling layers, a global-mean-pooling layer and end with two fully-connected layers. The Uncompressed Color Image Database (UCID) [29] which has 1338 images with a size of 512 x 384 or 384 x 512 is used as a data set. Since the input of the QCNN model requires to be the same size, images of the UCID are scaled to 512 x 512. The images are also preprocessed to extract error, which consists of rounding error and truncation error. Then, error images are treated as the input of the proposed model. Notice that the value

TABLE 4: Experiment results of two methods on UCID

QF	Huang's	Proposed
90	0.8956	**0.9917**
80	0.8380	**0.9846**
70	0.7246	**0.7783**

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2897000, IEEE Access



(a) QCNN

(b) Attention_based QCNN

FIGURE 5: The loss and accuracy of the QCNN and the attention based QCNN during the training and testing

range of truncation error and rounding error is too wide, so mixing truncation error and rounding error will produce some negative effects

For double JPEG compression detection, Huang et al. [32] first proposed a new perturbation threshold method. Compared to it, the proposed model has better detection capability. The comparison results of the two methods are shown in Table 4. Experimental results show that the detection accuracy of the proposed model has exceeded 95% at QF> 80, because when QF>80, the probability of rounding error is far greater than the truncation error, so the extracted error image is mainly composed of rounding error. When QF<80, the truncation error will increase, and the truncation error will be mixed into the error image, thus leading to a decrease of experimental results.

V. CONCLUSIONS AND FUTURE WORK

The present study introduces a novel convolutional neural network architecture QCNN and its improved version attention based QCNN. In order to adapt the new quaternion data flow format, a series of corresponding quaternion-based network layers are proposed. The new model is an attempt by CNN, in other number field and has got good results in the color image classification and forensics. In the future, we will continue to refine this model. Due to the existence of the guidance matrix, there may be multiple maximum values in a pooling window. It is a difficult task to find out one maximum because the quaternions represented by maximum are different. Different maximum positions will affect the final pooling result. Therefore, the angle cosine theorem is used to calculate the correlation between the maximum vector and other non-maximum vectors to deal with this problem.

ACKNOWLEDGMENT

This work was supported in part by the Natural Science Foundation of China under Grants (Nos. 61772281, U1636219, 61702235, 61502241, 61272421, 61232016, 61402235 and 61572258), in part by the National Key R&D Program of China(Grant No. 2016YFB0801303 and 2016QY 01W0105), in part by the plan for Scientific Talent of Henan Province (Grant No.2018JR0018), in part by the Natural Science Foundation of Jiangsu Province, China under Grant BK20141006, and in part by the Natural Science Foundation of the Universities in Jiangsu Province under Grant 14KJB520024, the PAPD fund and the CICAEET fund.

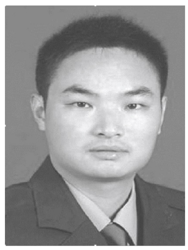
REFERENCES

[1] A. Krizhevsky, I. Sutskever and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” Advances in neural information processing systems., pp. 1097-1105, 2012.
[2] S. Karen, and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv: 1409.1556, 2014.
[3] M. Lin, Q. Chen and S. Yan, “Network in network,” arXiv preprint arXiv: 1312.4400, 2013.
[4] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” IEEE conference on computer vision and pattern recognition., Las Vegas, America, pp. 770-778, June. 2016.
[5] G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger, “Densely connected convolutional networks,” IEEE conference on computer vision and pattern recognition., Hawaii, America, vol. 1, No. 2, pp. 3, July. 2017.
[6] C. J. Evans, S. J. Sangwine and T. A. Ell, “Hypercomplex color-sensitive smoothing filters,” Image Processing., vol. 1, pp. 541-544, 2000.
[7] T. A. Ell and S. J. Sangwine, “Hypercomplex Fourier transforms of color images,” IEEE Transactions on image processing., vol. 16, no. 1, pp. 22-35, 2007.
[8] N. Le Bihan and S. J. Sangwine, “Quaternion principal component analysis of color images,” Image Processing., vol. 1, pp. I-809, Sep. 2003.
[9] X. Zhu, Y. Xu, H. Xu and C. Chen, “Quaternion convolutional neural networks,” the European Conference on Computer Vision., Keller, pp. 631-647, Sep. 2018.
[10] L. Shi and B Funt, “Quaternion color texture segmentation,” Computer Vision and image understanding., vol. 107, no. 1, pp. 88-96, 2007.
[11] D. Xu, L. Zhang and H. Zhang, “Learning algorithms in quaternion neural networks using ghr calculus,” Neural Network World., vol. 27, no. 3, pp. 271, 2017.
[12] P. Arena, L. Fortuna, L. Occhipinti and M. G. Xibilia, “Neural networks for quaternion-valued function approximation,” Circuits and Systems., vol. 6, pp. 307-310, May. 1994.
[13] T. Parcollet, M. Morchid, P. M. Bousquet, R. Dufour, G. LinarÍÍs and R De Mori, “Quaternion neural networks for spoken language understanding,” Spoken Language Technology Workshop., pp. 362-368, Dec. 2016.

[14] C. C. Chang and C. J. Lin, “LIBSVM: a library for support vector machines,” ACM transactions on intelligent systems and technology., vol. 2, no. 3, pp. 27, 2011.
[15] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv: 1412.6980, 2014.
[16] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel, “Backpropagation applied to handwritten zip code recognition,” Neural computation., vol. 1, no. 4, pp. 541-551, 1989.
[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez and I. Polosukhin, “Attention is all you need,” Advances in Neural Information Processing Systems., pp. 5998-6008, 2017.
[18] T. Shen, T. Zhou, G. Long, J. Jiang, S. Pan and C. Zhang, “Disan: Directional self-attention network for rnn/cnn-free language understanding,” arXiv preprint arXiv: 1709.04696, 2017.
[19] P. Verga, E. Strubell and A. McCallum, “Simultaneously Self-Attending to All Mentions for Full-Abstract Biological Relation Extraction,” arXiv preprint arXiv: 1802.10569, 2018.
[20] Z. Wang, D. Yan, R. Wang, L. Xiang and T. Wu, “Speech Resampling Detection Based on Inconsistency of Band Energy,” CMC: Computers, Materials & Continua., vol.56, no.2, pp. 247-259, 2018.
[21] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang and X Tang, “Residual attention network for image classification,” arXiv preprint arXiv: 1704.06904, 2017.
[22] J. Hu, L. Shen and G. Sun, “Squeeze-and-excitation networks,” arXiv preprint arXiv: 1709.01507, 2017.
[23] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng and Z. Zhang, “The application of two-level attention models in deep convolutional neural network for fine-grained image classification,” the IEEE Conference on Computer Vision and Pattern Recognition., pp. 842-850, 2015.
[24] B. Zhao, X. Wu, J. Feng, Q. Peng and S. Yan, “Diversified visual attention networks for fine-grained object classification,” IEEE Transactions on Multimedia., vol. 19, no. 6, pp. 1245-1256, 2017.
[25] M. F. Stollenga, J. Masci, F. Gomez and J. Schmidhuber, “Deep networks with internal selective attention through feedback connections,” Advances in neural information processing systems., pp. 3545-3553, 2014.
[26] W. Chan, N. Jaitly and Q. V. Le, “Speech recognition with attention-based recurrent neural networks,” U.S. Patent, no. 9, pp. 799-327, Oct. 2017.
[27] C. Weng, J. Cui, G. Wang, J. Wang, C. Yu, D. Su and D. Yu, “Improving Attention Based Sequence-to-Sequence Models for End-to-End English Conversational Speech Recognition,” Interspeech., pp. 761-765, 2018.
[28] J. Wang, T. Li, X. Luo, Y. Q. Shi, R. Liu and S. K. Jha, “Identifying Computer Generated Images Based on Quaternion Central Moments in Color Quaternion Wavelet Domain,” IEEE Transactions on Circuits and Systems for Video Technology., 2018. DOI: 10.1109/TCSVT.2018.2867786.
[29] F. Huang, J. Huang and Y. Q. Shi, “Detecting double jpeg compression with the same quantization matrix,” IEEE Transactions on Information Forensics and Security., vol. 5, no. 4, pp. 848-856, 2010.
[30] Z. Pan, J. Lei, Y. Zhang and F. Wang, “Adaptive fractional-pixel motion estimation skipped algorithm for efficient HEVC motion estimation,” ACM Transactions on Multimedia Computing, Communications, and Applications., vol. 14, no. 1, pp. 12-19, Jan. 2018.
[31] Z. Pan, X. Yi and L. Chen, “Motion and disparity vectors early determination for texture video in 3D-HEVC,” Multimedia Tools and Applications., pp. 1-18, 2018. DOI: https://doi.org/10.1007/s11042-018-6830-7
[32] S. Zhou, W. Liang, J. Li and J. U. Kim, “Improved VGG Model for Road Traffic Sign Recognition,” CMC: Computers, Materials & Continua., vol.57, no.1, pp. 11-24, 2018.
[33] Q. Cui, S. McIntosh and H. Sun, “Identifying Materials of Photographic Images and Photorealistic Computer Generated Graphics Based on Deep CNNs,” CMC: Computers, Materials & Continua., vol. 055, No. 2, pp. 229-241, 2018.

QILIN YIN received his B.S. in network engineering from Nanjing University of Information Science & Technology in 2018. He is pursuing M.S. degree in computer science and technology from Nanjing University of Information Science & Technology. His research interests include multimedia forensics.

JINWEI WANG is a professor from Nanjing University of Information Science & Technology. He received B.S. degree in automatic control from Inner Mongolia University of Technology in 2000. He received Ph.D. in information security at Nanjing University of Science & Technology in 2007 and was a visiting scholar in Service Anticipation Multimedia Innovation (SAMI) Lab of France Telecom R&D Center (Beijing) in 2006. He worked as a senior engineer at the 28th research institute, CETC from 2007 to 2010. He worked as a visiting scholar at New Jersey Institute of Technology, NJ, USA from 2014 to 2015. His research interests include multimedia copyright protection, multimedia forensics, multimedia encryption and data authentication. He has published more than 50 papers, hosted and participated in more than 10 projects.



XIANGYANG LUO received his B.S., M.S., and Ph.D degrees from the State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou, China, in 2001, 2004, and 2010, respectively. He is the author or co-author of more than 100 refereed international journal and conference papers. He is currently an associate professor of the State Key Laboratory of Mathematical Engineering and Advanced Computing. His research interests are image steganography and steganalysis technique.

JIANGTAO ZHAI received the B.S. degree in electrical and information engineering and the M.S. and Ph.D. degrees in control science and engineering from the Nanjing University of Science and Technology in 2006, 2008, and 2013, respectively. He is currently an Associate Professor with the Nanjing University of Information Science and Technology. His research interests include multimedia communication and network security.

SUNIL KR. JHA received the B.Sc. and M.Sc. degrees in Physics from the VBS Purvanchal University, India, in 2003 and 2005 respectively and the PhD degree in Physics from Banaras Hindu University, India, in 2012. From 2012 to 2014, he was a Postdoctoral Researcher at the Kyushu University, Japan and Hanyang University, Korea. From February 2015 to September 2017, he worked as Assistant Professor at University of Information Science and Technology, Macedonia and University of Information Technology and Management, Poland. Currently, he is working as Associate Professor at Nanjing University of Information Science and Technology. His research interests include chemical sensors, data mining and pattern recognition applications, renewable energy, visualization, and animation.

YUN-QING SHI has joined the Department of Electrical and Computer Engineering at the New Jersey Institute of Technology (NJIT), Newark, NJ since 1987, and is currently a professor there. He obtained his B.S. degree and M.S. degree from the Shanghai Jiao Tong University, Shanghai, China; his Ph.D. degree from the University of Pittsburgh, PA. His research interests include multimedia data hiding, forensics and security (robust watermarking, reversible data hiding, authentication, steganography and steganalysis, tampering detection, computer graphics classification from photographic images, camera identification, detection of double JPEG/MPEG compression), visual signal processing and communications (motion analysis, video compression and transmission), applications of image processing, computer vision and pattern recognition to industrial automation and biomedical engineering, theory of multidimensional systems and signal processing (robust stability of linear systems, 2-D spectral factorization, 2-D/3-D interleaving).

• • •