Original Article

COLOUR IMAGE PROCESSING USING MODIFIED QUATERNION NEURAL NETWORK

Surendra K Waghmare[1], Rupali B. Patil[2], Manisha G. Waje[3], Kavita V. Joshi[4], Vaishali P. Raut[5], Prabhat Chandra Shrivastava[6]

[1,2,3,4,5,6] G.H. Raisoni College of Engineering and Management Wagholi Pune Maharashtra.
{surendra.waghmare358@gmail.com, prabhatphd@gmail.com}

DOI: 10.47750/pnr.2022.13.S08.647

Abstract

Convolutional neural networks are becoming more popular as a means of resolving issues related to the extraction of colour picture features. The inherent advantage of quaternion neural network had led to a recent upsurge in research related to its implementation in image, voice and signal processing. Incorporating quaternion algebra to a neural network can decrease the neural parameters and in spite of decreased parameters, it still achieves state-of-the-art performance. On the other hand, the interconnection of the colour image channels is not taken into consideration in the general network. Because of this, the authors of this research suggest a newly designed quaternion convolutional neural network (QCNN), which always handles colour triples as a whole in order to prevent the loss of information. In order to completely combine the data from the different colour channels, the first quaternion convolution process has been created and shown. In order to provide an even higher level of protection for the accuracy of colour information, the quaternion batch normalisation and pooling processes are developed and designed in the quaternion domain. During this time, information on the attention mechanism is being included into the proposed QCNN in order to improve its overall performance. Experiments show that the proposed model is superior in terms of efficiency to both the conventional convolutional neural network and another QCNN with the same structure. Furthermore, the proposed model demonstrates superior performance in terms of colour image classification and colour image forensics.

Keywords: Convolutional Neural Networks, Quaternion Neural Network, Quaternion Domain

INTRODUCTION

ML's profound influence is apparent across a wide range of practical applications, including NLP and pattern recognition. Machine learning has a significant impact on the development of numerous systems used extensively to address a wide range of practical issues in recent years. ANN have achieved better results in a variety of practical applications as both the quantity and quality of accessible data and computational power have increased in conjunction [1, 2, 3]. With the advent of a quick and effective model training technique, DNN began to surpass other conventional machine learning algorithms [4].

Multidimensionality is common in real-world data, necessitating a methodologically distinct approach to analysing the interrelationships present within. To modify images, for instance, the (Red, Green and Blue) RGB channels, which describe the colours in a pixel, are used. Pixels/points are defined by their coordinates and they are utilised as inputs for both machine-based calculations and human perception. The three coordinates in the colour space that make up a pixel's RGB components depict a more complex shade like pink or brown. Neural networks need to capture these connections if they are to successfully generalise and reflect the pixel's multidimensional perspective [5]. Architectures that work with real numbers treat the components of an input vector as separate entities. By illustration presented in [6] showed that a real-valued (Convolutional Neural

Network) CNN could not be used in heterogeneous settings since it cannot learn to recognise colour from a grayscale picture. Then, as reference [7] showed, a real-valued (Neural Network) NN cannot remember the relations between the coordinates throughout the transformation into the 3D space; therefore, it loses the 3D shape of the input object. Multidimensional number-based neural networks have been offered as a solution for this problem.

The ease of representation of a multidimensional value by a quaternion attracts its diverse usage in real world. Recently, research into quaternion neural networks has blossomed. Because of the Hamilton product, QNNs may learn the local relations between its parts, as illustrated in equation 1. Machine learning's core difficulty is in the computation of reliable representations of massive data sets. To do this, an effective model must effectively capture both local connections. In this paragraph, we have explained why it is preferable to use a QNN rather than a real valued one while trying to encode relationships between features. Finding an appropriate multidimensional data representation for capturing the relations between the attributes of observable entity is a necessary first step. In order to more effectively represent the information that makes up an image, As opposed to a collection of one-dimensional pieces that may be connected to each other, it is beneficial to conceive of each pixel as a full entity comprised of three highly related elements. It is also possible that the parameter estimation and the latent relations between a pixel's RGB components cannot be encoded accurately using NN with real-valued parameters. Using quaternions in place of real numbers provides a very efficient solution to this issue. A quaternion is a four-dimensional object that can be used to construct and manipulate objects with up to four constituents. These latent relations inside a quaternion's characteristics can be captured by a quaternion neural network, which is made possible by the quaternion algebra.

BACKGROUND AND RELATED WORK

Arena et al. (1994) were the first to present QNNs, and they developed a specialised backpropagation algorithm to enable QNNs to be learned efficiently in the same way as real-valued neural networks are. Many subsequent publications dove further into the fundamentals of QNNs, exploring aspects like activation functions [8, 9], loss functions [10], initialising parameters [11], and building novel architectures [12] to better handle real-world multidimensional data. It is now possible to represent higher dimensional spaces using newer neural designs and techniques. In [13], the authors suggest a backpropagation method for neural networks that makes use of the product of three spatial dimensions. Though they have certain similarities with quaternions, 3D vectors are not handled using the same procedures.

Real-valued convolutional neural networks (CNN) are a kind of neural network that has been suggested [14] in order to more accurately capture relationships that take place between nearby input features. In the case of a CNN that works with actual values, the learning process for the model must take into account both the internal and external dependencies simultaneously. To be more specific, the relations that exist between each element that composes a multidimensional input feature are evaluated equally with the relations that exist between this input feature and the others. This applies to all of the relationships that exist between the input features. Quaternion convolutional neural networks (QCNN) is a solution that has been suggested by [15] to help relieve this issue. In point of fact, the quaternion algebra is responsible for encoding all of the internal relations, while the convolution process is responsible for learning all of the outward relations. Regarding the QMLP, each and every parameter of the QCNN is represented by a quaternion number.

In addition to the fundamental framework of a QCNN, authors [16] have suggested three image-processing tasks as a means of evaluating the effectiveness of the model. As a result, the deep QCNN that was suggested was tested on the CIFAR-10 and CIFAR-100 image classification tasks [17] and on the KITTI image segmentation challenge [18]. In both sets of trials, a total of 115 layers were layered in a residual manner [19] in order to facilitate rapid training convergences. The values of the pixel's red, green, and blue pixels are used to represent the x, y, and z components of quaternions, while the value of the pixel's grayscale component represents the real portion. In terms of accuracy, QCNNs perform much better than both real and complex valued CNNs, regardless of the task. The Quaternion-Valued Neural Network (QCNN) that was developed proves that quaternion-valued models are capable of competing with and even outperforming state-of-the-art methods by achieving the best possible results in more recent benchmarks.

More recently, in [20], researchers offered to explore the influence of the Hamilton product on a QCNN using a grayscale to colour picture job. In their proposal, the researchers used the image task. In specifically, the authors compressed a singular grayscale picture by using a quaternion and a real-valued convolutional auto encoder (QCAE, CAE) during the training phase of the research. Both models, much like any other auto encoder, have had the reconstruction error of the input picture at the output minimised in order to provide the best possible results. When it comes time to put the models through their paces, a colour picture is shown to them, despite the fact that they were only trained on a grayscale picture. It is interesting to note that the QCAE was capable of producing a colour picture that was faithfully reproduced at the output. On the other hand, the real-valued CAE generated a correct reconstruction of the test picture, but it did so in grayscale rather than in colour, and it did not learn the colour space. This phenomenon demonstrates the power of the Hamilton product to coerce QNNs into learning an internal connection inside the input characteristics that they are given.

PROPOSED APPROACH

Our proposed approach can be summarized in different parts of the implementation process of Quaternion in Convolutional Neural Network.

Convolution of a Quaternion

If we consider Q1 to be a quaternion weight and Q2 be an input of the same domain i.e. quaternion input, then the convolution process is nothing but a Hamilton product of the two quaternions as shown in the equation 1. In the convolution process, quaternion multiplication takes the role of point-to-point plain multiplication with a weight of W. The simplified convolution formula is illustrated in equation 1, which is also a quaternion matrix, and its real and imaginary parts may be derived by adding and subtracting four conventional convolutions, respectively. The quaternion convolution procedure is substantially simplified using this approach, as is the implementation of neural network design. Using the breakdown shown in Figure 1, the convolution operation may be broken down into its component parts. Conv5's cross product may be expressed as equation 2.

$$\begin{aligned} Q1 \otimes Q1 = &(a_1 a_2 - b_1 b_2 - c_1 c_2 - d_1 d_2) \\ &+(a_1 b_2 + b_1 a_2 + c_1 d_2 - d_1 c_2)\boldsymbol{i} \\ &+(a_1 c_2 - b_1 d_2 + c_1 a_2 + d_1 b_2)\boldsymbol{j} \\ &+(a_1 d_2 + b_1 c_2 - c_1 b_2 + d_1 a_2)\boldsymbol{k} \end{aligned} \tag{1}$$

where, Q1 and Q2 are two quaternions represented as:

$Q1 = (a_1 + b_1\boldsymbol{i} + c_1\boldsymbol{j} + d_1\boldsymbol{k})\ , and$

$Q2 = (a_2 + b_2\boldsymbol{i} + c_2\boldsymbol{j} + d_2\boldsymbol{k})$

Imaginary axes are ***i, j*** *and* ***k***

$$\begin{bmatrix} b_1 \\ c_1 \\ d_1 \end{bmatrix} \oplus \begin{bmatrix} b_2 \\ c_2 \\ d_2 \end{bmatrix} = \begin{bmatrix} c_1 d_2 - d_1 c_2 \\ d_1 b_2 - b_1 d_2 \\ b_1 c_2 - c_1 b_2 \end{bmatrix} \tag{2}$$

Learning data distribution is the heart of training a neural network. There is a dramatic drop in the network's generalisation capacity after the training data distribution no longer matches that of the test data. In addition, the training pace of the network will be drastically slowed down if the distribution of each batch of training data is different, since the network will learn to adapt to the varied distribution in each iteration. In order to deal with these limitations, batch normalisation was developed. Quantum batch normalisation deviates from the standard method of batch normalising by using a different formula to get the mean and variance. Quaternion mean and variance are defined as follows in equation 3 and 4.

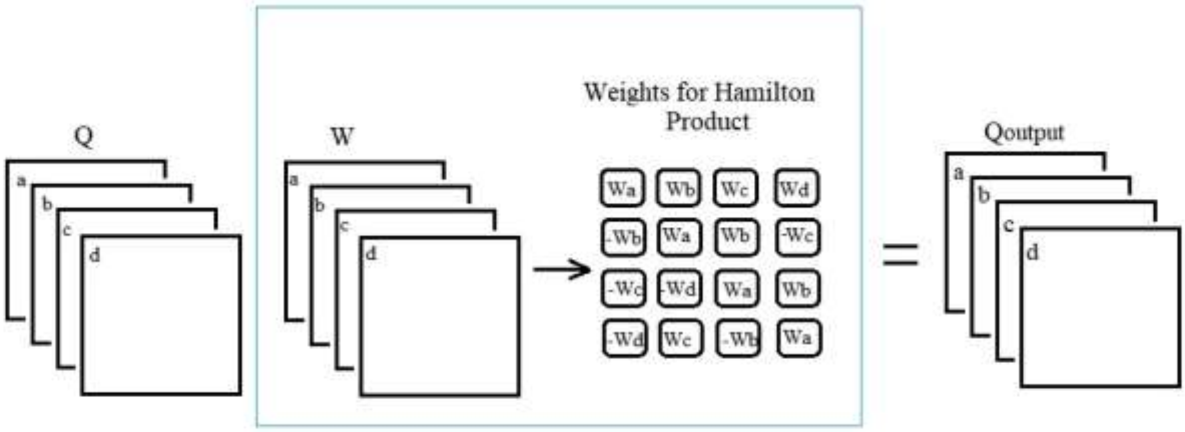
$$QE(x) = \frac{1}{T}\sum_{i=1}^{T} a_1 + b_1\boldsymbol{i} + c_1\boldsymbol{j} + d_1\boldsymbol{k}$$
$$= \overline{a_1} + \overline{b_1}\boldsymbol{i} + \overline{c_1}\boldsymbol{j} + \overline{d_1}\boldsymbol{k} \quad (3)$$

$$QE(x) = \frac{1}{T}\sum_{i=1}^{T}(x - QE(x))(x - QE(x))^*$$
$$= \frac{1}{T}\sum_{i=1}^{T} \Delta{a_1}^2 + \Delta{b_1}^2 + \Delta{c_1}^2 + \Delta{d_1}^2 \quad (4)$$

The quaternion mean and variance are denoted here by QE(x) and QV (x), where x is the quaternion entity denoted as $(a_1 + b_1\boldsymbol{i} + c_1\boldsymbol{j} + d_1\boldsymbol{k})$

Quaternion mean remains a quaternion, whereas quaternion variance is analogous to the conventional variance as a real number, when extended to the quaternion domain from typical statistical characteristics.

Figure 1: A quaternion Neural Network representation

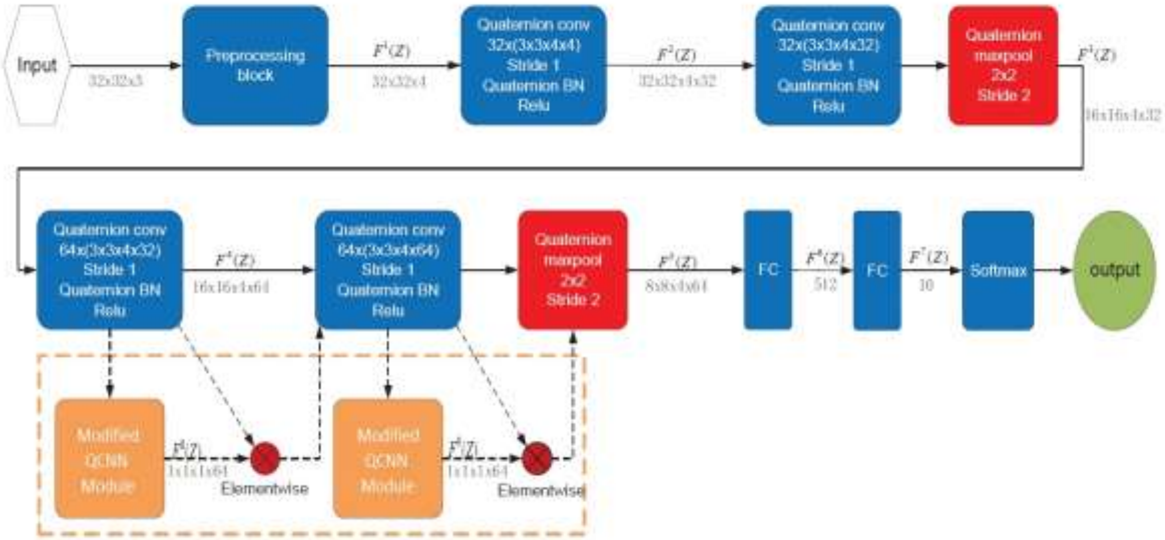


RESULT AND DISCUSSION

The foundation of the proposed QCNN model is laid out in this section. Then, we put the QCNN model's reconstructed parts to the test to see how well they function. Then, the suggested model is contrasted with Pure

QCNN and its advantage is carefully examined in terms of colour picture categorization. Good performance in colour picture processing is shown by using the suggested model to the detection of double JPEG compression.

Figure 2: Diagrammatic representation of the workflow process of the modification in basic Quaternion Convolutional Neural Network.



We constructed a simple QCNN to evaluate the functionality of the module it is a part of. A total of four layers are included: two convolutional blocks, two max-pooling layers, and two fully connected layers to round things up. There are two convolution layers in each convolution block, and these layers perform the convolution, batch normalisation, and activation operations using quaternions. Commonly, we utilise an activation function called ReLU.

In Figure 2, we see the overall network architecture and some of the individual parameters. Each convolution kernel and feature map in the proposed model is a width-by-height-by-quaternion-size i.e. (width x height x 4) quaternion matrix, which is different from the standard convolutional neural network format. Keep in mind that the updated quaternion module shown here is the additional module and was not included in the original QCNN. This is just a technique for boosting the efficiency of the network.

Table 1: Table showing the accuracy of the proposed modified QCNN with the traditional QCNN

Methodology	**Dataset**	**Test Accuracy**
Basic QCNN	MS-COCO	0.8128
Modified QCNN	MS-COCO	0.8621

Four components of a quaternion may be normalised throughout the process of training a network. In order to evaluate the effectiveness of the modified QCNN in comparison to the original QCNN, the input data is flipped and shifted. Cross entropy loss is used to train the network as a whole with a learning rate of 0.00011. At epoch 75, the training is complete. Tables 1 and 2 display the data obtained throughout the experiments. In comparison to previous models, the one suggested here performs better. Although batch normalisation has a little effect on the feature map's data distribution, the conventional approach nonetheless leads to a breakdown in colour coherence.

Since the quaternion method always considers the whole colour channel, it avoids this loss. The significance of color-channel correlation is further shown in the comparison's outcome.

Table 2: Table showing the classification accuracy of the proposed modified QCNN with the traditional QCNN

Methodology	**Dataset**	**Test Accuracy**
Basic NN	MS-COCO	0.7422
Basic QCNN	MS-COCO	0.8255
Modified QCNN	MS-COCO	0.8701

CONCLUSION

This research presents a new design for convolutional neural networks called QCNN, as well as an enhanced version called attention based QCNN. A set of quaternion-based network layers are suggested to accommodate the new quaternion data flow format. CNN has tried its hand at a novel model in the realm of mathematics, and the results have been promising in the areas of colour picture categorization and forensics. As time goes on, we want to improve upon this framework even more. The presence of the guiding matrix introduces the possibility of several maximum values inside a given pooling frame. Due to the diversity of the quaternions represented by maximum, finding a single maximum may be challenging. It is important to note that the aggregate pooling result will change depending on the maximum positions used. To solve this issue, we use the angle cosine theorem to determine the degree of similarity between the maximum vector and other non-maximum vectors.

REFERENCES

[1] Isokawa, Teijiro, Tomoaki Kusakabe, Nobuyuki Matsui, and Ferdinand Peper. "Quaternion neural network and its application." In International conference on knowledge-based and intelligent information and engineering systems, pp. 318-324. Springer, Berlin, Heidelberg, 2003.

[2] Parcollet, Titouan, Mohamed Morchid, and Georges Linarès. "A survey of quaternion neural networks." Artificial Intelligence Review 53, no. 4 (2020): 2957-2982.

[3] Zhu, Xuanyu, Yi Xu, Hongteng Xu, and Changjian Chen. "Quaternion convolutional neural networks." In Proceedings of the European Conference on Computer Vision (ECCV), pp. 631-647. 2018.

[4] Narayan, Vipul, and A. K. Daniel. "Design consideration and issues in wireless sensor network deployment." (2020): 101-109.

[5] Kusamichi, Hiromi, Teijiro Isokawa, Nobuyuki Matsui, Yuzo Ogawa, and Kazuaki Maeda. "A new scheme for color night vision by quaternion neural network." In Proceedings of the 2nd international conference on autonomous robots and agents, vol. 1315. 2004.

[6] Parcollet, Titouan, Mirco Ravanelli, Mohamed Morchid, Georges Linarès, Chiheb Trabelsi, Renato De Mori, and Yoshua Bengio. "Quaternion recurrent neural networks." arXiv preprint arXiv:1806.04418 (2018).

[7] Shang, Fang, and Akira Hirose. "Quaternion neural-network-based PolSAR land classification in Poincare-sphere-parameter space." IEEE Transactions on Geoscience and Remote Sensing 52, no. 9 (2013): 5693-5703.

[8] Shen, Wen, Binbin Zhang, Shikun Huang, Zhihua Wei, and Quanshi Zhang. "3d-rotation-equivariant quaternion neural networks." In European Conference on Computer Vision, pp. 531-547. Springer, Cham, 2020.

[9] Narayan, Vipul, and A. K. Daniel. "CHOP: Maximum coverage optimization and resolve hole healing problem using sleep and wake-up technique for WSN." ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal 11.2 (2022): 159-178.

[10] Liu, Yang, Yanling Zheng, Jianquan Lu, Jinde Cao, and Leszek Rutkowski. "Constrained quaternion-variable convex optimization: A quaternion-valued recurrent neural network approach." IEEE transactions on neural networks and learning systems 31, no. 3 (2019): 1022-1035.

[11] Luo, Lincong, Hao Feng, and Lijun Ding. "Color image compression based on quaternion neural network principal component analysis." In 2010 International Conference on Multimedia Technology, pp. 1-4. IEEE, 2010.

[12] Tan, Guoqiang, Zhanshan Wang, and Zhan Shi. "Proportional-integral state estimator for quaternion-valued neural networks with time-varying delays." IEEE Transactions on Neural Networks and Learning Systems (2021).

[13] Zhang, Weiwei, Chunlin Sha, Jinde Cao, Guanglan Wang, and Yuan Wang. "Adaptive quaternion projective synchronization of fractional order delayed neural networks in quaternion field." Applied Mathematics and Computation 400 (2021): 126045.

[14] Chen, Yonghui, Xian Zhang, and Yu Xue. "Global exponential synchronization of high-order quaternion Hopfield neural networks with unbounded distributed delays and time-varying discrete delays." Mathematics and Computers in Simulation 193 (2022): 173-189.

[15] Vipul, Narayan, and A. K. Daniel. "A novel protocol for detection and optimization of overlapping coverage in wireless sensor network." International Journal of Engineering and Advanced Technology 8.6 (2019): 422-462.
[16] Granero, Marco Aurélio, Cristhian Xavier Hernández, and Marcos Eduardo Valle. "Quaternion-Valued Convolutional Neural Network Applied for Acute Lymphoblastic Leukemia Diagnosis." In Brazilian Conference on Intelligent Systems, pp. 280-293. Springer, Cham, 2021.
[17] Matsumoto, Yuya, Ryo Natsuaki, and Akira Hirose. "Full-Learning Rotational Quaternion Convolutional Neural Networks and Confluence of Differently Represented Data for PolSAR Land Classification." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 15 (2022): 2914-2928.
[18] Onyekpe, Uche, Vasile Palade, Stratis Kanarachos, and Stavros-Richard G. Christopoulos. "A quaternion gated recurrent unit neural network for sensor fusion." Information 12, no. 3 (2021): 117.
[19] Narayan, Vipul, et al. "E-Commerce recommendation method based on collaborative filtering technology." International Journal of Current Engineering and Technology 7.3 (2017): 974-982.
[20] Li, Qi, Xingyuan Wang, Bin Ma, Xiaoyu Wang, Chunpeng Wang, Zhiqiu Xia, and Yunqing Shi. "Image steganography based on style transfer and quaternion exponent moments." Applied Soft Computing 110 (2021): 107618.
[21] Bayro-Corrochano, Eduardo. "A survey on quaternion algebra and geometric algebra applications in engineering and computer science 1995–2020." IEEE Access (2021).
[22] Narayan, Vipul, and A. K. Daniel. "FBCHS: Fuzzy Based Cluster Head Selection Protocol to Enhance Network Lifetime of WSN." ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal 11.3 (2022): 285-307.
[23] Narayan, Vipul, and A. K. Daniel. "Energy Efficient Protocol for Lifetime Prediction of Wireless Sensor Network using Multivariate Polynomial Regression Model." Journal of Scientific & Industrial Research 81.12 (2022): 1297-1309.