National Research University

Higher School of Economics

As a manuscript

Pavlov Stanislav Vladimirovich

GENERALIZATION OF NEURAL NETWORKS ON THE DUAL NUMBERS ALGEBRA

PhD Dissertation Summary

for the purpose of obtaining academic degree

Doctor of Philosophy in Computer Science

Academic Supervisor:

Doctor of Science, Professor Valeriy Kalyagin

Nizhny Novgorod — 2024

GENERAL CHARACTERISTICS OF WORK

Neural networks are currently used in classification, detection, semantic segmentation, and natural language processing. However, problems of limited accuracy and the balance between inference time and accuracy of algorithms significantly limit the scope of their application. Modern approaches of generalization of neural networks to hypercomplex algebras are aimed at solving these problems [1], [2].

The relevance of the topic is due to the increasing popularity of the direction of generalization of neural networks on hypercomplex algebras. On the one hand, this is due to the fact that raw data are often presented in a complex form, on the other hand, a number of new studies demonstrate the superiority of the use of complex neural networks, compared to real ones, including greater accuracy [3] and better convergence [4].

Complex neural networks are increasingly being used in various applications and research tasks [5]. Such tasks as radio signal turnover, image processing and computer vision, processing and analysis of audio signals, Signal processing from radars and sanaras, cryptography, time series prediction, associative memory, wind prediction, robotics, traffic control, spam detection, predictions in agroculture and others.

The author B. Widrow was one of the first to propose a complex LMS algorithm and showed its efficiency in comparison with real ones [6]. In the work of A. Hirose on radio signal processing [7] compares the generalization characteristics of complex-valued and real-valued neural networks in terms of coherence of processed signals. The problem of function approximation (temporal signal interpolation) is studied. Simulations and real experiments show that complex-valued neural networks with the amplitude-phase activation function show a smaller generalization error, than really significant networks, such as bivariate real-valued neural networks. Also, the following authors have also made significant contributions to the application of the ideas of complex neural networks in the processing of radio signals: B. Widrow, T. Kim, S. Scardapane, Y. Quan, M. Catelani, A. Marseet, I. Cha, S. Chen, D. Jianping, W. Gong, A. Uncini, M. Scarpiniti, R. Huang, M. Solazzi, N. Benvenuto, A. B. Suksmono, A. Hirose, Y. Chistyakov, A. Minin, J. Zhang, S. Liu, M. Peker, S. Hu, Y. Suzuki, T. Ding [6]-[34].

A significant contribution to the development of complex networks for computer vision was made by the authors: M. Arjovsky, C.-A. Popa, M. Matlacz, J. N. Eisenberg, P. Virtue, E. Eisenberg, R. S. Zemel, C. Trabelsi, S. Amilia, M. Miyauchi, A. Hirose, Y. Liu, R. F. Olanrewaju, R. Hata, Y. Kominami, C.-A. Popa, L. Li, [1], [35]-[56]. Complex deep neural networks had been limited for some time in application to computer vision problems due to the lack of necessary building blocks. A landmark work by C. Trabelsi [45] provides key components for complex deep neural networks and demonstrates their application to convolutional neural networks and LSTM. C. Trabelsi proposed complex convolutions and several variants of algorithms for complex packet normalization, a strategy for initializing weights for complex neural networks, and also showed advantages over real analogs in computer vision problems. In the work of the author S. Gu [57] proposes a complex analogue of VGG, a complex analogue of a fully-connected layer, and shows the advantage of such an architecture for recognition problems, achieving better quality in a similar class of architectures for the time.

Many researchers expanded the use of complex neural networks to other tasks: processing and analysis of audio signals - C. Trabelsi, D. Hayakawa, M. Kataoka, M. Kinouchi, A. Y. H. Al-Nuaimi, Y.-S. Lee, C. S. Tay [45], [58-63], processing signals from radars and sanaras - J. Gao, M. Wilmanski, I. N. Aisenberg, K. Oyama, X. Yao [64]-[69], cryptography - T. Dong [70], time series prediction – I. N. Aisenberg [66], associative memory - S. Jankowski, T. Miyajima [71-72], wind prediction - H. H. Cevik, T. Kitajima, D. P. Mandic [73-75], robotics - Y. Maeda [76], traffic flow control - I. Nishikawa [77-78], spam detection - J. Hu [79], predictions in agriculture – I. N. Aisenberg [37].

These stunning results inspired a further generalization of neural networks to other hypercomplex numbers, in particular to dual numbers. Dual numbers are already have application in screw theory (F. M. Dimentberg) [80], also dual numbers allow automatic derivatives (A. Güneş Baydin, R. Kiran) [81-82]. At the same time, there is only one basic attempt to apply dual numbers in neural networks (Y. Okawa) [83], which used the properties of dual numbers for input data. This area deserves further study.

The aim of the dissertation is to generalize neural networks to the algebra of dual numbers in order to achieve a better ratio of quality-speed calculations.

To achieve this goal, the following tasks were solved:

1) To develop a mathematical basis for neural networks on hypercomplex algebras.
2) To develop a methodology for constructing neural networks on hypercomplex algebras.
3) Conducting computational experiments to demonstrate the benefits of the new approach.

The object of the study is neural networks.

The subject of research is mathematical and algorithmic support of the generalization of neural networks to hypercomplex algebra.

The scientific novelty of the work consists in the following:

1) The formula of hypercomplex norm and the algorithm of batch normalization based on this norm are defined. A derivative formula for functions of second-order hypercomplex variables is also defined.
2) Fifteen neural network operators on hypercomplex algebra are defined, including basic (convolution, linear, group batch norm, pooling, linear rectification block), dual holomorphic operators are defined. Computational experiments on construction of hypercomplex neural networks were carried out. A procedure for transferring knowledge from real to hypercomplex networks has been developed.
3) The advantages (performance and accuracy) of the developed approach to solving a number of problems (computer vision, detection of gravitational waves and music transcription) are shown, due to the identification of dual-type features that have not been considered previously.

Practical value. The possibility of applying the developed approach to solving practical problems is shown (basic realizations of hypercomplex networks, classical problems of computer vision, detection of gravitational waves, music transcription task, as well as improvements with application of dual holomorphic neural networks). The software implementation in the MindSpore open access product was completed.

Implementation of the results of work. The results of the study were introduced into the educational process at the Department of Applied Mathematics and Informatics. Hypercomplex operators and networks added to Open Access Product MindSpore.

Methods of research. Modern methods of machine learning, deep neural network theory, hypercomplex number algebra are used.

Key results to be presented:

1) A hypercomplex norm formula and a batch normalization algorithm based on this norm. Derivative formula for functions of second-order hypercomplex algebras.
2) Development of neural network architecture for hypercomplex algebras. Development and software implementation of neural network operators on hypercomplex algebras.
3) Results for assessing the effectiveness of various strategies for transferring knowledge from real neural networks to hypercomplex ones.
4) Efficiency (in terms of speed and accuracy) of hypercomplex neural networks application in tasks of classical computer vision, detection of gravitational waves and transcription of music and others.

Results reliability. The reliability of the results is ensured by the correct development of the mathematical apparatus and the conduct of experimental researches.

Work approbation. The main provisions and results of the dissertation were reported and discussed at the following scientific and technical conferences and seminars:

1) IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS, Madrid, Spain, November' 29 – December' 2 2022), talk «Dual-valued Neural Networks».
2) IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, June' 4 – June' 10 2023), talk "Learning Properties of Holomorphic Neural Networks of Dual Variables".
3) Scientific seminar at the Applied AI Center, Skolkovo Institute of Science and Technology (November 12, 2023), talk "Generalization of neural networks to the algebra of dual numbers."

4) Extended scientific seminar of the laboratory of Algorithms and Technologies for Networks Analysis of the National Research University Higher School of Economics (December 20, 2023), talk “Generalization of neural networks on the dual numbers algebra.”

Publications. On the topic of the dissertation were published 3 papers.

Personal contribution: the author developed the key idea of generalizing neural networks to the algebra of dual numbers, mathematical apparatus and algorithmic part, carried out experiments and made conclusions.

Publications of higher level:

1） Pavlov, S.; Kozlov, D.; Bakulin, M.; Zuev, A.; Latyshev, A.; Beliaev, A. Generalization of Neural Networks on Second-Order Hypercomplex Numbers. Mathematics 2023, 11, 3973. https://doi.org/10.3390/math11183973. Scopus Q1 journal, Q2 in base scimagojr.com. Personal contribution: the author developed the key idea of generalizing neural networks to the algebra of dual numbers, developed basic concepts and approaches, the mathematical apparatus (including the dual derivative and dual gradient) and the algorithmic part (including dual architectures and neural network operators), conducted training experiments, made optimizations and conclusions.

Standard level publications:

2） Dmitry Kozlov; Stanislav Pavlov; Alexander Zuev; Mikhail Bakulin; Mariya Krylova, Igor Kharchikov. Dual-valued Neural Networks. 2022 18th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS, Madrid, Spain). DOI: 10.1109/AVSS56176.2022.9959227. Core B, IEEE

Xplore, WoS. Personal contribution: the author developed the key idea of generalizing neural networks to the algebra of dual numbers, the mathematical apparatus (including the dual norm) and the algorithmic part (including the dual operators of neural networks), conducted training experiments and made conclusions.

3） Dmitry Kozlov; Mikhail Bakulin; Stanislav Pavlov; Aleksandr Zuev; Mariya Krylova, Igor Kharchikov. Learning Properties of Holomorphic Neural Networks of Dual Variables. ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP, Rhodes Island, Greece). DOI: 10.1109/ICASSP49357.2023.10095457. Core B, IEEE Xplore, WoS, and 13th conference by Impact Factor https://research.com/conference-rankings/computer-science. Personal contribution: the author developed the key idea of generalizing neural networks to the algebra of dual numbers, the concept of holomorphic dual networks, the mathematical apparatus (including the concept of dual holomorphic functions) and the algorithmic part (including dual holomorphic operators of neural networks), conducted training experiments, made optimizations and conclusions.

Structure and scope of work. The dissertation consists of an introduction, two chapters, a conclusion, a list of references and appendices. Totally 97 pages of text containing 14 figures, 10 tables and 58 formulas. References contain 118 sources.

MAIN CONTENT OF WORK

The introduction substantiates the relevance of the dissertation work, formulates the purpose and objectives of the research, explains the scientific novelty and practical value of the obtained results, presents the submitted for the protection of the position, and gives the general characteristics of the work.

The first chapter describes the theory of dual numbers in neural networks: the problem is set, the generalization of neural networks to all hypercomplex numbers of the second order, the representation of data in hypercomplex algebra, hypercomplex operations, norm of hypercomplex numbers, hypercomplex batch normalization, and backpropagation of the loss function gradient, hypercomplex converter.

In this study of dual operators and neural networks, author founds that many aspects of this problem are very similar (in some cases coincide) to complex and double numbers. All these algebras (dual $\mathbb{D}$, complex $\mathbb{C}$, and double $\mathbb{S}$ or complex-split) are second-order hypercomplex algebras.

By definition, the set of dual numbers is a commutative ring $<\mathbb{D},+,\times>$, which is a superset of the real number field $<\mathbb{R},+,\times>$, and has a special element ε so that each element $\mathbb{D}$ can be represented as $a+\varepsilon b$ for $a,b \in \mathbb{R}$, and $\varepsilon^2 = 0$, $\varepsilon \neq 0$. Similarly, a set of double numbers $<\mathbb{S},+,\times>$ is a superset $<\mathbb{R},+,\times>$ with a special element j so that each element of $\mathbb{S}$ can be represented as $a + jb$ for $a,b \in \mathbb{R}$, and $j^2 = 1$ (but $j \notin \mathbb{R}$). Dual-, double-, and complex rings are isomorphic of a special Clifford algebra including one, e_0, e_1, and e_0e_1, where $e_0^2 = -1$, $e_1^2 = \sigma \in \{0,1,-1\}$ and $-e_0e_1 = e_1e_0 = \tau$, which is equivalent to ε, j or i, respectively. In each case $\tau^2 = \sigma$, according to the definition of the corresponding numbers ring. This mathematical unity allows us to generalize all hypercomplex algebras of the second order and to deal with them in the same way.

The basic mathematical operations for these numbers are:

$$(x_1 + \tau y_1) \pm (x_2 + \tau y_2) = (x_1 \pm x_2) + \tau(y_1 \pm y_2)$$

$$(x_1 + \tau y_1) \cdot (x_2 + \tau y_2) = (x_1 x_2 + \sigma\, y_1 y_2) + \tau(x_1 y_2 + y_1 x_2)$$

$$\frac{x_1 + \tau y_1}{x_2 + \tau y_2} = \frac{x_1 x_2 - \sigma y_1 y_2}{x_2^2 - \sigma y_2^2} + \tau \frac{x_2 y_1 - x_1 y_2}{x_2^2 - \sigma y_2^2}$$

$$(x + \tau y\,)^* \;=\; x - \tau y$$

Author links the imaginary part of the original input to the dual component

$$x \;+\; iy \;\Rightarrow\; x \;+\; \varepsilon y \text{ or } x \;+\; jy$$

To explain convolution in hypercomplex algebra, author uses a matrix representation of second-order hypercomplex numbers that uses real numbers. It is known that the algebras of complex, dual, and double numbers u=x+τy are isomorphic to the algebras of the second-order real matrices of the form $\begin{pmatrix} x & y \\ \sigma y & x \end{pmatrix}$. Thus, the convolution of the hypercomplex filter $W = W_x + \tau W_y$ and the hypercomplex value input u=x+τy can be expressed as follows:

$$W * u = \begin{pmatrix} W_x & W_y \\ \sigma W_y & W_x \end{pmatrix} * \begin{pmatrix} x & y \\ \sigma y & x \end{pmatrix}$$
$$= \begin{pmatrix} W_x * x + \sigma W_y * y & W_x * y + W_y * x \\ \sigma\big(W_x * y + W_y * x\big) & W_x * x + \sigma W_y * y \end{pmatrix}.$$

Linear layer. To generalize the linear layers for two-dimensional algebras, author uses the matrix representation of second-order hypercomplex numbers. A linear layer with hypercomplex inputs and weights is equivalent to the superposition of real linear layers:

$$\mathbb{H}L(W, B, u) = \mathbb{R}L(W_x, B_x, u_x) + \tau^2 \mathbb{R}L\big(W_y, 0, u_y\big)$$
$$+ \tau\left(\mathbb{R}L\big(W_x, 0, u_y\big) + \mathbb{R}L\big(W_y, B_y, u_x\big)\right),$$

where $\mathbb{R}L(w, b, x)$ denotes a linear layer with a real value, with w, b and x for weights, offsets and input data respectively.

The average pool operation involves calculating the arithmetic average for each block of elements. This means collapsing each n×n element block by its average value. This is equivalent to convolution when the pitch is equal to the size of the kernel, where the weight of the kernel is equal to the real numbers $\frac{1}{n^2}$. Since the kernel $W = W_x + \tau W_y$ in this case is completely real, $W_y = 0$, and convolution formula can be simplified:

$$W * u = W(x + \tau y) = W_x * (x + \tau y) = W_x * x + \tau W_x * y.$$

Based on this expression, the average pooling is equivalent to two real average pooling operations: each of them is applied independently to each component of the input data $u = x + \tau y$:

$$\mathbb{H}\mathrm{AvgPool}(u) = \mathbb{R}\mathrm{AvgPool}(x) + \tau\mathbb{R}\mathrm{AvgPool}(y)$$

Activation functions are used to introduce nonlinearity into neural networks. There are many activation functions based on real numbers, and a greater variety of them is based on hypercomplex numbers. Among real-valued activations, there is a family of ReLU-type functions that help solve the problem of damping gradient. Functions of this type are also used in complex algebra. For example, we already know the application of ReLU to real and imaginary parts separately [45]. In this study author extends this definition and apply it to other algebras:

$$\mathbb{H}\mathrm{Relu}(u) = \mathbb{R}\mathrm{Relu}(x) + \tau\mathbb{R}\mathrm{Relu}(y).$$

Norm

Traditionally, in mathematical and physical literature, the modulus of a complex number is defined as

$$\mathbb{C}|z| = \sqrt{zz^*} = \sqrt{(x + iy)(x - iy)} = \sqrt{x^2 + y^2}.$$

However, generalization of this method to other types of hypercomplex numbers does not work very well:

$$\mathbb{H}|z| = \sqrt{uu^*} = \sqrt{(x + \tau y)(x - \tau y)} = \begin{cases} |x|, & u \in \mathbb{D} \\ \sqrt{x^2 - y^2}, & u \in \mathbb{S} \end{cases}.$$

This result is hardly applicable to objective. The dual norm does not depend on the dual part. In the case of double numbers, the function is not defined for half of the elements. It is therefore necessary to develop another formula that extends the standard norm of complex numbers to dual and double algebras.

To define the expression, turn to the matrix representation $A = \begin{pmatrix} x & y \\ \sigma y & x \end{pmatrix}$ of the hypercomplex number u= x+τy. Then author connects the norm of this matrix to the norm of the original dual number. There are several ways to define a matrix norm. First, define $\mathbb{R}^{m\times n}$ as a vector space of matrices with m rows and n columns of records in a real field $\mathbb{R}$. Author uses the norm of the matrix caused by the norm of the vector $\|\cdot\|_2$ on $\mathbb{R}^{2\times 2}$ and vector of the norm $\|\cdot\|_2$ on $\mathbb{R}^{2\times 1}$ and author sets dual norm:

$$\|u\|^2 = \sup\{\|At\|_2^2 : t \in \mathbb{R}^{2\times 1}, \|t\|_2^2 = 1\}.$$

For a matrix that corresponds to the hypercomplex number u = x+τy, the norm expression is as follows:

$$\|At\|_2^2 = \left(\begin{pmatrix} x & y \\ \sigma y & x \end{pmatrix}\begin{pmatrix} t_1 \\ t_2 \end{pmatrix}\right)^T \begin{pmatrix} x & y \\ \sigma y & x \end{pmatrix}\begin{pmatrix} t_1 \\ t_2 \end{pmatrix}$$

$$= (xt_1 + yt_2)^2 + (xt_2 + \sigma y t_1)^2 = \begin{bmatrix} t_1^2 + t_2^2 = 1 \\ t_1 = \sin\varphi \\ t_2 = \cos\varphi \end{bmatrix}$$

$$= x^2 + y^2(\cos^2\varphi + \sigma^2 \sin^2\varphi) + xy \sin 2\varphi\,(1 + \sigma).$$

In order to find the extremum of this function, equates the derivative to zero and solve for φ

$$f'(\varphi) = y^2 \sin 2\varphi\,(\sigma^2 - 1) + 2xy \cos 2\varphi\,(1 + \sigma) = 0$$

$$\tan 2\varphi = -\frac{2x}{y(1-\sigma)} \quad \Rightarrow \quad \begin{aligned} \cos 2\varphi &= \mp \frac{y(1-\sigma)}{\sqrt{4x^2 + y^2(1-\sigma)^2}} \\ \sin 2\varphi &= \pm \frac{2x}{\sqrt{4x^2 + y^2(1-\sigma)^2}} \end{aligned}$$

In the end, author gets the maximum function:

$$\|u\|^2 = x^2 + \frac{y^2(1+\sigma^2)}{2} + |y|(1+\sigma)\sqrt{x^2 + y^2\left(\frac{1-\sigma}{2}\right)^2}.$$

From this it is easy to see that

$$\|z\| = \frac{|y|(1+\sigma)}{2} + \sqrt{x^2 + y^2\left(\frac{1-\sigma}{2}\right)^2}.$$

Special cases of complex, dual, double numbers (σ=-1,0,1) lead to the following formula for the norm of hypercomplex numbers:

$$\mathbb{H}\|u\| = \mathbb{H}\|x + \tau y\| = \begin{cases} \sqrt{x^2 + y^2}, & u \in \mathbb{C} \\ \left|\frac{y}{2}\right| + \sqrt{x^2 + \left(\frac{y}{2}\right)^2}, & u \in \mathbb{D} \\ |x| + |y|, & u \in \mathbb{S} \end{cases}$$

Hypercomplex batch normalization.

Known formula for batch normalization [84]:

$$\hat{x}^{(k)} = \frac{x^{(k)} - E\left[x^{(k)}\right]}{\sqrt{V[x^{(k)}]}},$$

where the covariance matrix $V\left[x^{(k)}\right]$ is defined as:

$$V\left[x^{(k)}\right] = \begin{pmatrix} \mathrm{Cov}\left(x_r^{(k)}, x_r^{(k)}\right) & \mathrm{Cov}\left(x_r^{(k)}, x_i^{(k)}\right) \\ \mathrm{Cov}\left(x_i^{(k)}, x_r^{(k)}\right) & \mathrm{Cov}\left(x_i^{(k)}, x_i^{(k)}\right) \end{pmatrix}.$$

Similar to the real-value approach, complex packet normalization also has an additional linear transformation with two parameters $\hat{\gamma}^{(k)}$, $\hat{\beta}^{(k)}$ with real-valued parameters [45]:

$$\hat{\gamma}^{(k)} = \begin{pmatrix} \hat{\gamma}_{rr}^{(k)} & \hat{\gamma}_{ri}^{(k)} \\ \hat{\gamma}_{ir}^{(k)} & \hat{\gamma}_{ii}^{(k)} \end{pmatrix}, \quad \hat{\beta}^{(k)} = \begin{pmatrix} \hat{\beta}_{r}^{(k)} \\ \hat{\beta}_{i}^{(k)} \end{pmatrix}.$$

That is, complex batch normalization:

$$\mathbb{C}\mathrm{BN}\left[\hat{x}^{(k)}\right] = \hat{\gamma}^{(k)}\hat{x}^{(k)} + \hat{\beta}^{(k)}.$$

Author generalizes the batch normalization process for dual and double tensors. To achieve this, use the rule proposed above. We cannot use the same procedure as for complex batch normalization (from [45]) for double entry for the following reason:

$$\mu^{(k)} = E\left[x^{(k)}\right] = \mu_r^{(k)} + \varepsilon\mu_d^{(k)},$$

$$\Gamma^{(k)} = E\left[\left(x^{(k)} - \mu^{(k)}\right)^2\right]$$
$$= E\left[\left(x_r^{(k)} - \mu_r^{(k)}\right)^2\right]$$
$$+ \varepsilon E\left[\left(x_r^{(k)} - \mu_r^{(k)}\right)\left(x_d^{(k)} - \mu_d^{(k)}\right)\right],$$

$$C^{(k)} = E\left[\left(x^{(k)} - \mu^{(k)}\right)\overline{\left(x^{(k)} - \mu^{(k)}\right)}\right] = E\left[\left(x_r^{(k)} - \mu_r^{(k)}\right)^2\right].$$

Here we see that both the covariance Γ and the pseudo-covariance C do not independent from $E\left[\left(x_d^{(k)} - \mu_d^{(k)}\right)^2\right]$. This problem is analogous to the independence of the dual number norm from its dual part. Therefore, we must find an alternative way to generalize packet normalization for all types of second-order hypercomplex algebras.

In this study, author presents a method based on the concept of norm for hypercomplex numbers derived above.

First, define the mean by the traditional channel method, which is the same for all algebras:

$$\mu^{(k)} = E\left[u^{(k)}\right] = \mu_x^{(k)} + \tau\mu_y^{(k)}.$$

Then determine the dispersion specific to the type of hypercomplex algebra:

$$\mathrm{Var}\left[u^{(k)}\right] = \frac{1}{m-1}\sum_{i=1}^{m}\left\|u_i^{(k)} - \mu^{(k)}\right\|^2,$$

where $\|\cdot\|$ defined above. Convert the input data as:

$$\hat{u}^{(k)} = \frac{u^{(k)} - \mu^{(k)}}{\sqrt{\mathrm{Var}[u^{(k)}] + \delta}},$$

where δ is a small number needed to avoid dividing by zero. The final step is hypercomplex channel scaling and offset:

$$\mathbb{H}\mathrm{BN}\left[\hat{u}^{(k)}\right] = \hat{\gamma}^{(k)}\hat{u}^{(k)} + \hat{\beta}^{(k)},$$

where $\hat{\gamma}^{(k)}$ and $\hat{\beta}^{(k)}$ are hypercomplex weights and bias, respectively. In the special case of complex numbers:

$$\begin{aligned}\mathbb{C}\mathrm{Var}[z] &= E[(z-\mu)(z-\mu)^*] \\ &= E[(x-\mu_x)^2] + E\left[\left(y-\mu_y\right)^2\right] = \Gamma(z).\end{aligned}$$

So divide the centered input by $\sqrt{\mathbb{C}\mathrm{Var}[z]}$, which gives the covariance $\Gamma(\hat{z}) = 1$. Note that in general the pseudocovariance is not equal to zero:

$$\begin{aligned}C(\hat{z}) = E[\hat{z}^2] = E[(\hat{x} + i\hat{y})^2] &= E[\hat{x}^2 - \hat{y}^2] + 2iE[\hat{x}]E[\hat{y}] \\ &\neq 0.\end{aligned}$$

Backpropagation of the loss function gradient. An important part of neural network training is gradient computation. The calculation of the loss function gradient relies on a back-propagation algorithm that uses a chain rule.

In this study, author studies the problem of gradient propagation in the algebra of hypercomplex numbers. Classical definition of the derivative of a function $f(u) = f(x + \tau y) = v(x, y) + \tau w(x, y)$ of a hypercomplex element $u = x + \tau y$, where v and w are real functions:

$$f'(u) = \lim_{\Delta u \to 0}\frac{f(u + \Delta u) - f(u)}{\Delta u}.$$

This limit can only exist if it is defined at Δu approaches to zero along the real axis $\Delta u = x$ or imaginary axis $\Delta u = \tau\Delta y$. In both cases, it should produce the same result.

Leveling these special cases, author obtains the generalized equivalents of the Cauchy-Riemann equations:

$$\frac{\partial v}{\partial x} = \frac{\partial w}{\partial y} \quad \frac{\partial v}{\partial y} = \tau^2 \frac{\partial w}{\partial x} = \sigma \frac{\partial w}{\partial x}.$$

The functions satisfying these equations are called holomorphic. In practice, this is a strong constraint, and most existing operators do not meet the Cauchy-Riemann criteria.

To overcome this, use the approach invented by Wirtinger for complex numbers. It uses variable substitution to rewrite the function of a complex variable $f(z)$ as holomorphic functions of two variables $f(z, z^*)$. Author extends this approach to all second-order algebras:

$$x = \frac{u + u^*}{2} \quad y = \frac{u - u^*}{2\tau}.$$

For complex and double numbers we can easily eliminate $\frac{1}{\tau}$ by multiplying the numerator and the denominator by τ and with $\tau^2 = \sigma \in \mathbb{R}$ in the denominator.

First, author explains what should be calculated. In the case of complex networks, researchers usually use real x_n and the imaginary y_n parts of the weights z_n as separate real channels, and update them using the real derivative of the loss function:

$$\begin{matrix} x_{n+1} = x_n - \alpha \frac{\partial L}{\partial x} \\ y_{n+1} = y_n - \alpha \frac{\partial L}{\partial y} \end{matrix} \rightarrow z_{n+1} = z_n - \alpha \left(\frac{\partial L}{\partial x} + i \frac{\partial L}{\partial y} \right).$$

Here author generalizes these calculations for all second-order hypercomplex numbers, considering:

$$\mathrm{u}_{\mathrm{n}+1} = \mathrm{u}_{\mathrm{n}} - \alpha\left(\frac{\partial \mathrm{L}}{\partial \mathrm{x}} + \tau\frac{\partial \mathrm{L}}{\partial \mathrm{y}}\right).$$

Author defines the expression inside the parentheses as a hypercomplex gradient and calculate it through the gradient of the hypercomplex operator f as:

$$\frac{\partial \mathrm{L}}{\partial \mathrm{x}} + \tau\frac{\partial \mathrm{L}}{\partial \mathrm{y}} = \frac{\partial \mathrm{L}}{\partial \mathrm{f}}\left(\frac{\partial \mathrm{f}}{\partial \mathrm{x}} + \tau\frac{\partial \mathrm{f}}{\partial \mathrm{y}}\right) + \left(\frac{\partial \mathrm{L}}{\partial \mathrm{f}}\right)^{*}\left(\frac{\partial \mathrm{f}}{\partial \mathrm{x}} - \tau\frac{\partial \mathrm{f}}{\partial \mathrm{y}}\right)^{*}.$$

It is worth noting that $\frac{\partial \mathrm{L}}{\partial \mathrm{x}} + \tau\frac{\partial \mathrm{L}}{\partial \mathrm{y}}$ can be expressed in u and u^{*}:

$$\frac{\partial \mathrm{f}}{\partial \mathrm{x}} + \tau\frac{\partial \mathrm{f}}{\partial \mathrm{y}} = \begin{cases} 2\frac{\partial \mathrm{f}}{\partial \mathrm{u}^{*}} & \mathrm{u} \in \mathbb{C} \\ 2\frac{\partial \mathrm{f}}{\partial \mathrm{u}} & \mathrm{u} \in \mathbb{S} \\ \frac{\partial \mathrm{f}}{\partial \mathrm{u}} + \frac{\partial \mathrm{f}}{\partial \mathrm{u}^{*}} + \varepsilon^{2}\left(\frac{\partial \mathrm{f}}{\partial \mathrm{u}} - \frac{\partial \mathrm{f}}{\partial \mathrm{u}^{*}}\right) & \mathrm{u} \in \mathbb{D} \end{cases}$$

Author implemented this formula for dual algebra. This approach shows the same result as the calculation of two real derivatives. Experiments show that the training time is also roughly the same.

Conversion to hypercomplex networks.

The challenge is to transform existing real number-based neural network architectures into networks that use hypercomplex values. This transformation is designed to represent the neural network in a more general way and achieve higher accuracy by using more parameters. The following steps should be taken as a starting point:

1. Load real-valued model with pre-trained weights. The model accuracy is expected to be the best for the chosen architecture on a certain dataset.
2. Convert the model into the hypercomplex one. The real part of the weights is initialized as the original loaded weights and the imaginary part is filled with zeros. This approach yields a hypercomplex model with the same accuracy metric as the original model.

3. Train the resulting model. Starting from the checkpoint that gives relatively high accuracy, the model uses its increased generalization ability (due to the addition of the imaginary channel for data) to receive higher results than the source model.

Central Kernel Alignment (CKA) Metrics

In this section author conducts a comparative analysis of features generated by real and hypercomplex (dual, complex) models. Main hypothesis is that hypercomplex neural networks extract features that are different from features generated by a real model with the same architecture. To test this hypothesis, author introduces several similarity metrics.

Author uses several standard methods, such as:

- Correlations (Kendall coefficients τ and Pearson r) between distributions for the classification problem obtained from different networks;
- Analysis of vector positions for embeddings of the last layer of networks working in different algebras;
- Difference between CAMs generated by neural networks for the same samples.

These methods have been shown to be statistically unstable and difficult to interpret, so the Central Kernel Alignment (CKA) metric is used. It is represented as a matrix, each element of which denotes the Hilbert-Schmidt independence criterion (HSIC) for the layers of two different networks:

$$\mathrm{HSIC}(\mathrm{K},\ \mathrm{L}) = \frac{1}{\mathrm{b}(\mathrm{b}-3)}\left(\mathrm{tr}(\tilde{\mathrm{K}}\tilde{\mathrm{L}}) + \frac{\sum_{\mathrm{i,j}=1}^{\mathrm{b}} \mathrm{K}_{\mathrm{ij}} \sum_{\mathrm{i,j}=1}^{\mathrm{b}} \tilde{\mathrm{L}}_{\mathrm{ij}}}{(\mathrm{b}-1)(\mathrm{b}-2)} - \frac{2}{\mathrm{b}-2}\sum_{\mathrm{i,j}=1}^{\mathrm{b}} (\mathrm{K}\tilde{\mathrm{L}})_{\mathrm{ij}}\right),$$

where $\begin{matrix} \widetilde{K}_{ij} = K_{ij} - \delta_{ij} K_{ij} \\ \widetilde{L}_{ij} = L_{ij} - \delta_{ij} L_{ij} \end{matrix}$, b is the size of the packet, and all tensors are smoothed out $b \times c \times h \times w$ to 2D matrices $b \times c * h * w$..

$$\mathrm{CKA}_{i,j}(X, Y) = \frac{\frac{1}{n}\sum_{k=1}^{n} \mathrm{HSIC}\left(X_i^{(k)} X_i^{(k)^T}, Y_j^{(k)} Y_j^{(k)^T}\right)}{\sqrt{\frac{1}{n}\sum_{k=1}^{n} \mathrm{HSIC}\left(X_i^{(k)} X_i^{(k)^T}, X_i^{(k)} X_i^{(k)^T}\right)} \sqrt{\frac{1}{n}\sum_{k=1}^{n} \mathrm{HSIC}\left(Y_j^{(k)} Y_j^{(k)^T}, Y_j^{(k)} Y_j^{(k)^T}\right)}}.$$

Author tested this metric on real ResNet18 models trained on the CIFAR-100 dataset from scratch for two different initial values. To do this, author calculated a metric for pairs of layers taken from different models. Author expected the hidden representations of these models to be close because they are the same model trained on the same data set.

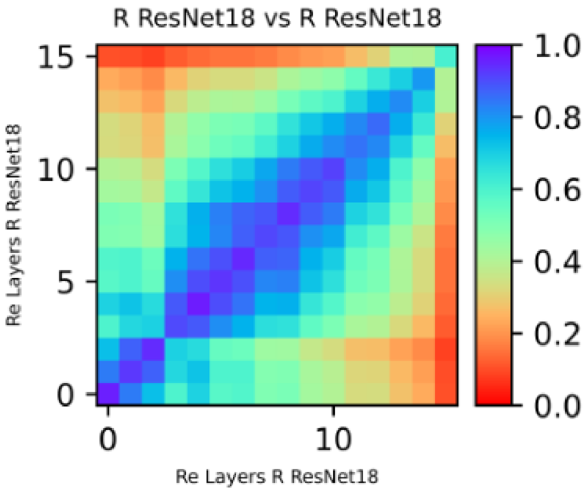

Figure 6. The value of the CKA metric for ResNet18 models trained with different initial values.

As expected, CKA is close to 1 for diagonal features because both models with the same layer number must have the same representation. The original metric CKA is defined for sets of real numbers, so author extends this approach to the cases of hypercomplex algebras. Author tried the following methods:

- Matrix representation of complex and dual numbers;
- The norm of complex and dual numbers;
- Combining double/imaginary components with real ones;
- Compute separately real and hypercomplex parts.

The proposed methods yielded similar results, so the latter approach was chosen as more demonstrative.

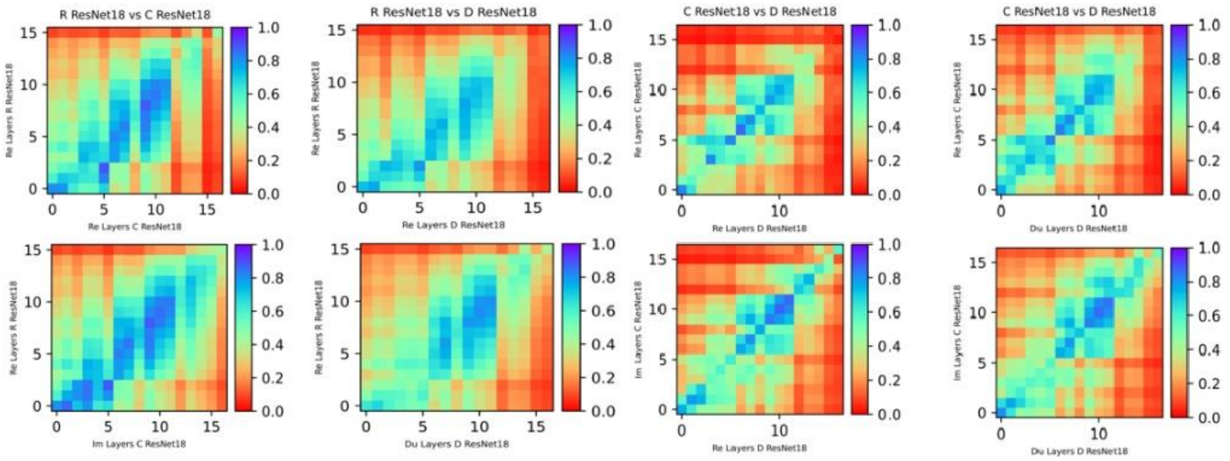

Figure 7. CKA metric values calculated separately for real and hypercomplex parts for a pair of ResNet18 models from different algebras.

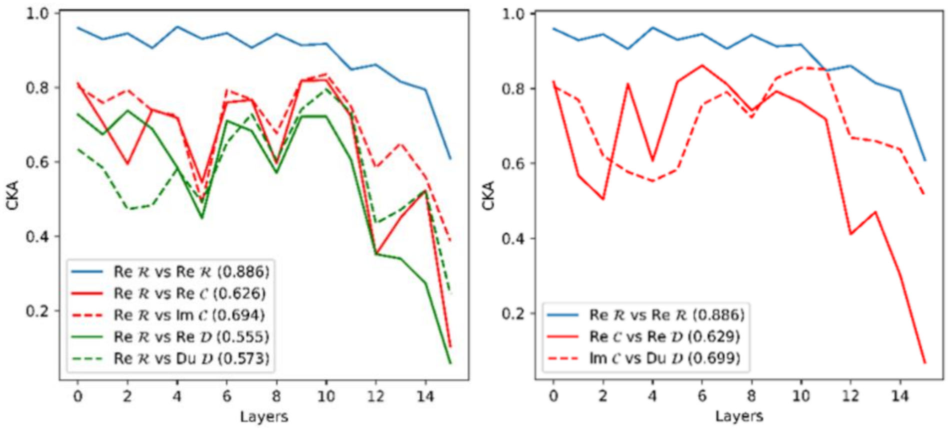

Figure 8. Diagonal values of CKA metrics (left) between output real NS and real/dual/complex part of hypercomplex NS (right) some special cases.

Figure 8 shows that the characteristics of models based on different algebras are quite close. Thus, author makes a conclusion about the possibility of transfer of knowledge for real neural networks into hypercomplex and vice versa. Further knowledge distillation results also support this assumption.

At the same time, the average CCA values for $\mathrm{Re}\,\mathbb{R}$ and $\mathrm{Re}\,\mathbb{D}$ is 0.555 and for $\mathrm{Re}\,\mathbb{R}$ and $\mathrm{Du}\,\mathbb{D}$ is 0.573, so dual-value neural networks have some new features (features) that are not represented in models with valid values. This metric also shows that the dual and complex values are quite close, $\mathrm{Im}\,\mathbb{C}$ and $\mathrm{Du}\,\mathbb{D}$ 0.699.

Knowledge Transfer (KT) from Neural Networks with Real Values

The previous subsection shows that the hidden data representation for real and hypercomplex models is close. This prompted us to test the possibility of transferring knowledge from real-valued models to hypercomplex models.

To do this, author used ResNet model weights with valid values, pre-trained on the ImageNet dataset from PyTorch, and initialized the real part of the hypercomplex model weights with these values. The imaginary part of the scales is zero.

The results of the experiments are presented in Table 2. It is evident that the use of pre-trained scales leads to an increase in the accuracy of the models. A remarkable fact is that even after knowledge transfer, hypercomplex (dual and complex) models show better accuracy than real ones.

Table 2. The result (accuracy, %) of training of hypercomplex ResNet18 models with loaded weights, on Dataset CIFAR-100, classification task

	Without KT, %	With KT, %

Real	75.9	79.9
Dual	**78.3**	**82.1**
Complex	78.2	81.9

Based on the results of the experiments, it is concluded that knowledge transfer contributes to increasing the convergence of dual models (Fig. 9)

The second chapter presents realizations of hypercomplex networks, classical problems of computer vision, detection of gravitational waves, the music transcription, as well as dual holomorphic neural networks.

Hypercomplex Toy Net

Before proceeding to deep learning models, author starts with a demonstrative neural network designed for predicting values of noised functions of a hypercomplex argument. This architecture remains the same for all algebras, but for each algebra author uses own implementation of operators. To show the advantage of models based on hypercomplex operators, author compares their results with the result of a real model of the same architecture. The mean standard deviation is used as a loss function. Author trains these four models over 1000 epochs with the same number of parameters to predict the values of two functions: $\mathrm{Ai}(\mathrm{u})$ - Airy function of the first kind and $\mathrm{J}_3(\mathrm{u})$ - Bessel function of the first kind of the third order. Author also adds noises with normal distribution to the values of the training sample (and test sample) functions.

Table 3. Toy Net - value of loss function.

Function	$\mathrm{Ai}(\mathrm{u})$	$\mathrm{J}_3(\mathrm{u})$

Algebra	$u \in \mathbb{C}$	$u \in \mathbb{D}$	$u \in \mathbb{S}$	$u \in \mathbb{C}$	$u \in \mathbb{D}$	$u \in \mathbb{S}$
Real	0.026	0.045	0.013	0.018	0.017	0.015
Complex	**0.008**	0.015	0.015	**0.009**	0.012	0.011
Dual	0.017	**0.009**	0.017	0.013	**0.009**	0.012
Double	0.023	0.050	**0.010**	0.017	0.018	**0.010**

Table 3 shows that the smallest mean square error (RME) between the hypercomplex value of the function and the two-component (hypercomplex) model prediction is achieved by a neural network, belonging to the same type of hypercomplex numbers as the original function (and its argument). For example, a complex-valued function is best predicted by a complex-valued model, and so on. Thus, author concludes that networks based on hypercomplex numbers can learn the dependencies or regularities underlying the corresponding algebra.

Classical Computer Vision CV Problems

The process of data preparation needs to be clarified before CV classification can be tackled. To convert a real image into a hypercomplex format, let's use a more general method than the one already proposed (not us), where the authors convert a real image into a complex form using $[R, G, B] \Longrightarrow [R + iG, G + iB]$, claiming, that this type of encoding captures channel correlations and tint changes.

In this study, author generalizes this idea and use two types of preprocessing. The first one $[R, G, B] \Longrightarrow [R + \tau G, G + \tau B, B + \tau R]$, which author calls color combination (CC). The second variant is a linear transformation (LT) with learnable parameters that convert

[R, G, B] into six real channels, which are later converted into three hypercomplex channels. This type of preprocessing seems to be preferable because it allows model to determine the best color space for the task. In addition, CC is a special case of LT where the transformation matrix is fixed.

Author takes the ResNet18 architecture as the basis of model, which is generalized to second-order hypercomplex algebras by replacing real operators with their counterparts in hypercomplex algebras. Author uses a stochastic gradient descent with a momentum of 0.9 to optimize the real sign loss functions, treating the real and the hypercomplex parts as separate channels with a real sign. For the transformation of hypercomplex features into real ones, author applies the hypercomplex norm associated with the corresponding algebra. The cross entropy loss between the input and the target is used as a criterion for these problems. The learning rate is planned according to the following rule: 0.1 for the first 60 epochs, 0.02 for the 61-120 epochs, 0.004 for the 121-160 epochs, 0.0008 for the 161-200 epochs. Image classification results for CIFAR-10, CIFAR-100 and SVHN are shown in Table 4.

Table 4. Accuracy (%) of models (real, dual, complex, double) for classification problems on CIFAR-100, CIFAR-10, SVNH, with different pre-processing with Color Combination (CC) and Linear Transformation (LT).

Dataset	CIFAR-100		CIFAR-10		SVHN	
Real	74.37		93.83		95.95	
Algebra	Pre-processing					
	LT	CC	LT	CC	LT	CC

Dual	76.30	76.12	94.27	94.32	96.35	96.23
Complex	77.12	76.65	94.56	94.45	96.35	96.04
Double	75.52	75.76	94.23	94.05	96.30	96.20

Table 4 shows that all models based on second-order numbers achieve higher accuracy than real ones. In addition, the complex-valued neural network shows better metric values than other hypercomplex models.

It can be seen from Table 5 that the transition from a real model to a hypercomplex model significantly increases the computational complexity. In addition, linear transformation takes longer than combining colors. The network with complex values shows the worst performance, but the implementation of the Re-Im representation helps to reduce the gap. Table 5 also shows that the use of group convolutions causes the model to slow down. The diagonal representation of double numbers reduces the output time by more than 1.5 times. However, dual networks are not the optimal model, as they show the worst accuracy of all hypercomplex models.

Table 5. Average inference time (μs) of hypercomplex ResNet-50 models (real, dual, complex, double) for CIFAR-100, batch size = 1.

Algebra	Inference time, μs	
	CPU	GPU
Real	20.07	4.04
Dual	85.74	12.81
Complex	114.82	15.85

Double	108.95	15.73

From Tables 4 and 5, author concludes that neural networks based on dual numbers represent a reasonable balance between increasing computational complexity and achieving greater accuracy.

Detection of gravitational waves

This part deals with the problem of signal detection using hypercomplex networks for the G2Net dataset [84]. The data set consists of simulated noisy signals, similar to gravitational waves, recorded by a system of three ground-based laser interferometers: LIGO Hanford, LIGO Livingston and Virgo [85-87]. As a rule, gravitational waves are emitted during cosmic events, such as black hole fusion [85]. The G2Net dataset contains records of emulated events of the same nature.

To classify the original signal, author preprocess the data for the image and then pass it through neural networks. The purpose of preprocessing is to build a representative frequency map of the original signal. The CQT algorithm is considered effective for the analysis of gravitational waves.

Table 6. Average values of metrics (%) of models for gravitational wave detection.

Algebra	Accuracy, %	AUC ROC
Real	76.45	0.82
Complex	78.73	0.84
Dual	**79.24**	**0.85**

Double	77.41	0.84

This preprocessing translates the time series into a frequency portrait (Figure 13), which is treated as an image in subsequent steps. Thus, changes in the frequency characteristics of the signal at a certain moment are reflected as visual features, such as specific shape and color, in the resulting image. To classify images obtained after the CQT algorithm [84], author again uses the ResNet18 model, whose operators change to corresponding ones in different algebras.

The model is optimized using the Stochastic Gradient Descent algorithm with momentum of 0.9 and weight decay of $5 \cdot 10^{-5}$ for L2 regularization. The regularization is enhanced by adding dropout with value of 0.2. The scheduler for learning rate uses Cosine Annealing policy with $T_{max} = 6$, and initial value of learning rate is set to $5 \cdot 10^{-2}$. Training lasts for 30 epochs.

From Table 6, one can see that all models on the hypercomplex algebras outclass the real-valued model in both accuracy and AUC ROC. The best result is achieved by a dual-valued neural network.

Music transcription task

In this part author shows the results of automatic transcription of music. Experiments are performed with the MusicNet dataset [88]. To improve computational efficiency, author resamples the original signal from 44.1 kHz to 11 kHz, based on the algorithm from [89]. This allows us to reduce computational overhead without any significant loss of information. As in [88], '2303', '2382', '1819' are used as a test subset, and the other 327 files are used as a training set. author is doing all the experiments with the complex representation of the frequency spectrum. For the real model, consider the real and imaginary components of the spectrum as separate channels.

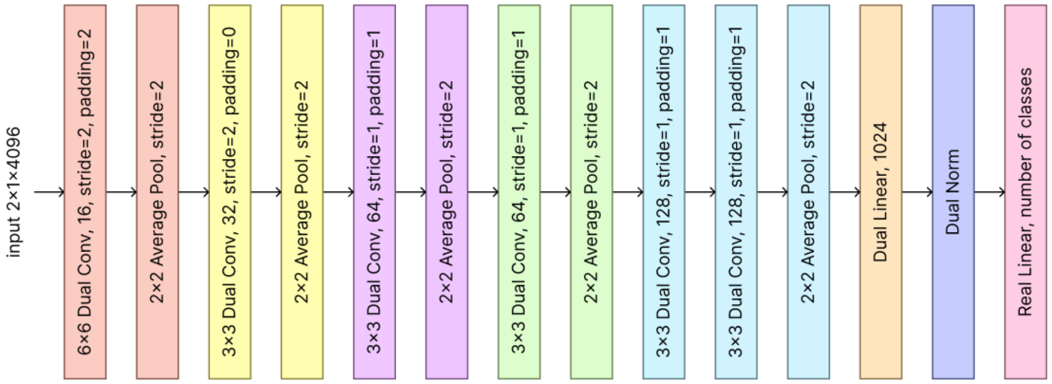
input 2×1×4096
6×6 Dual Conv, 16, stride=2, padding=2
2×2 Average Pool, stride=2
3×3 Dual Conv, 32, stride=2, padding=0
2×2 Average Pool, stride=2
3×3 Dual Conv, 64, stride=1, padding=1
2×2 Average Pool, stride=2
3×3 Dual Conv, 64, stride=1, padding=1
2×2 Average Pool, stride=2
3×3 Dual Conv, 128, stride=1, padding=1
3×3 Dual Conv, 128, stride=1, padding=1
2×2 Average Pool, stride=2
Dual Linear, 1024
Dual Norm
Real Linear, number of classes

Figure 14. Dual DeepConvNet architecture.

The structure is the same in a complex case up to the replacement of complex blocks with dual analogs.

Author also uses the DeepConvNet architecture developed in [45] (Figure 14). This network consists of six one-dimensional convolutional layers. The first has a filter size of six, and the other layers have a kernel size of three. The convolution blocks are followed by a real-valued linear layer with 2048 links for the real model or a complex/dual linear layer with 1024 links for the complex/dual model and a ReLU activation function. Before passing through the last layer, you must change the data representation from complex/dual form to real form. To preserve all the information, author combines real and imaginary/dual components into one shared channel. Finally, author applies a real-valued linear layer with 84 bonds and a sigmoid activation function. The number of units in the last statement corresponds to the number of notes present in the data set. For real, dual, and complex models, author uses the component activation function of ReLU as described earlier. In all experiments, author uses an input window of 4096 samples or its FFT (which corresponds to the 8192 window used in the baseline) and predict notes in the center of the window. All networks are optimized with Adam [90]. Author starts with a learning rate of 10^{-3} for the first 10 epochs, and then reduce it by 10 times for each of the 10, 100, 120 and 150 epochs.

The complex network is initialized using the unitary initialization scheme respecting the He criterion as it was described in [45]. The dual-valued and real-valued models are initialized by the He initialization according to the method proposed in [91], basing on a uniform distribution. The results are summarized in Table 7. Precision-Recall dependency is depicted in Figure 15.

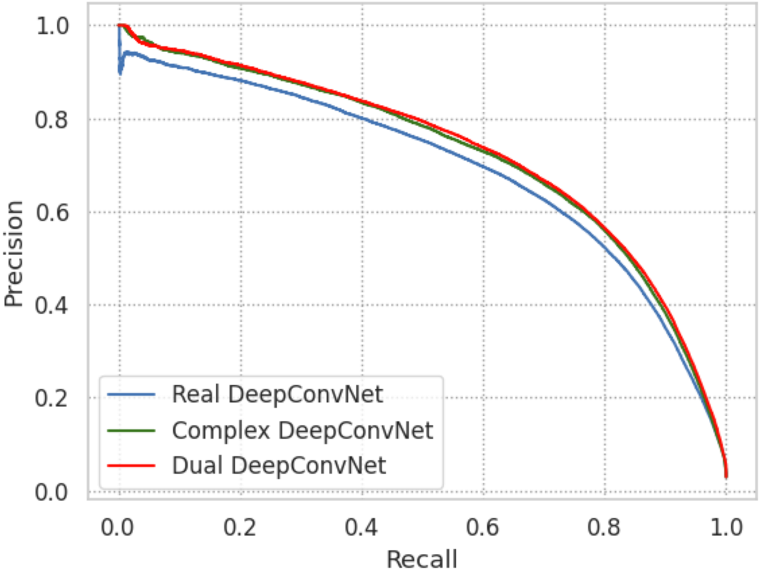

Figure 15. Precision-recall curve for MusicNet dataset.

Table 7. Comparison of main metrics of real and hypercomplex DeepConvNet models (real, dual, complex) on the music transcription task, on dataset: MusicNet.

Model: DeepConvNet	Average Precision, %	Parameters, MB	Inference time, µs
Real	68.9	34.2	**37**
Dual	**73.4**	34.2	50

Complex	73.2	34.2	64

It can be seen from Table 7 that the best average precision is achieved by a dual neural network. In addition, as will be shown, the dual-valued model has a ×0.8 inference time of the complex model.

Dual holomorphic neural networks

In this section author defines dual holomorphic layers and show the results of models based on them. Holomorphic means that these layers satisfy the Cauchy-Riemann conditions for dual values of functions.

1) Cauchy-Riemann conditions for the function of dual variables

For a complex-valued function to be differentiable, special equation must be satisfied, which are called the Cauchy-Riemann conditions. Originally, the Cauchy–Riemann equations are certain criteria needed for a complex function $f(x+iy)=u(x,y)+iv(x,y)$ to be holomorphic (complex differentiable), where u and v are real-valued functions of two variables. These equations impose restrictions for $u(x,y)$ and $v(x,y)$:

$$\frac{\partial v}{\partial x}=\frac{\partial w}{\partial y} \qquad \frac{\partial v}{\partial y}=-\frac{\partial w}{\partial x}$$

There are analogous conditions for a dual-valued function $f(x+\varepsilon y)=u(x,y)+\varepsilon v(x,y)$ to be holomorphic (in sense of dual numbers):

$$\frac{\partial v}{\partial x}=\frac{\partial w}{\partial y}, \qquad \frac{\partial v}{\partial y}=0.$$

Using Taylor series expansion for dual-valued step, one can show that the above restrictions imply that a holomorphic function of the dual variable is expanded to the following form:

$$f(x+\varepsilon y)=f(x)+\varepsilon y f'(x).$$

In this study author considers dual functions and operators as an analytic extension of real functions. So author assumes that f(x) is real for any real x. To be clear, this is a sufficient, but not necessary, condition for a function to be holomorphic.

2) Dual-valued Holomorphic Operators

General formula of dual-valued convolution for the input $Z = X + \varepsilon Y$ and weight $W = W_r + \varepsilon W_d$ with bias $b = b_r + \varepsilon b_d$ is

$$Z * W + b = X * W_r + b_r + \varepsilon(Y * W_r + X * W_d + b_d).$$

One can see, that general formula of dual-valued convolution in general case of weight matrix W does not satisfy the analogous conditions for a dual-valued function to be holomorphic (in sense of dual numbers). To make sure a dual convolution is holomorphic, we must impose a restriction $Du(W) \equiv 0$. This condition is based on the fact that, for a linear function $f(x + \varepsilon y) = a_r x + b_r + \varepsilon(a_r y + a_i x + b_d)$ the limits of its increment, as the argument approaches zero along the real axis or the dual axis, are equal if and only if the condition $a_d = 0$ is true. So, author yields the following equation for holomorphic dual-valued convolution:

$$Z * W + b = X * W_r + b_r + \varepsilon(Y * W_r + b_d).$$

It has about two times less parameters than general dual convolution, or, in other words about the same number of parameters as the real-valued one.

Author also defined several holomorphic dual activation functions:

$$ReLU(d) = \begin{cases} x + \varepsilon y, & x \geq 0 \\ 0 & x < 0 \end{cases} = \begin{cases} d, & x \geq 0 \\ 0 & x < 0 \end{cases},$$

$\sigma(d) = \sigma(x) + \varepsilon y \sigma(x)\big(1 - \sigma(x)\big)$, where $\sigma(x) = \frac{1}{1+e^{-x}}$,

$tanh(d) = tanh(x) + \varepsilon y(1 - \tanh(x)^2)$, where $tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$.

3) Dual-valued input generation

As complex numbers, dual numbers are essentially pairs of real values. Based on this similarity, author proposes to use complex-valued input in both dual and complex neural networks. In this work, author uses two methods to convert real input data into a complex format: the Fourier transform for the MusicNet dataset and the Q constant transform (CQT) for the G2Net dataset.

Author also developed an alternative variant of transformation. Author notices that $Du\big(f(x + \varepsilon y)\big)$ is mainly determined by the derivative of the function at the same point as $Re\big(f(x + \varepsilon y)\big) = f(x)$. Basing on this, author proposes to transform real-valued numbers of input to the dual numbers as follows:

$$Input \to Input + \varepsilon(Input)',$$

where $(Input)'$ is a function of $Input,$ which in a sense author calls a derivative of that input. The specific definition of the derivative depends on the task. For example, if input is a time series then it seems natural to define the derivative with respect to time as the difference of signal strength at adjacent time points.

4) Experimental results.

To test hypothesis, author conducts several experiments with previously used datasets: G2Net, MusicNet. Author also tested this approach on the ImageNet dataset.

Table 8. Average precision on the G2Net dataset.

Model	Input	BN	Average precision, %
Real	$\lvert CQT\rvert$	Real	76.5
Complex	CQT	Complex	78.7
Dual	CQT	Complex	73.5

Dual	CQT	Dual	**79.2**
Dual	$\lvert CQT\rvert + \varepsilon\lvert CQT\rvert'$	Dual	51.7
Holomorphic dual	CQT	Dual	77.0
Holomorphic dual	$\lvert CQT\rvert + \varepsilon\lvert CQT\rvert'$	Dual	77.6
Holomorphic dual	$\lvert CQT\rvert + \varepsilon\lvert CQT\rvert'$	Holomorphic dual	78.4

Table 9. Average precision on the MusicNet dataset.

Model	Average precision, %
Real	68.9
Complex	**73.4**
Dual	73.2
Holomorphic dual	71.2

Table 10. Average precision on the ImageNet dataset.

Model	Top-1	Top-5
Real	69.76	89.08
Dual	70.76	89.58
Holomorphic dual	**70.79**	**89.63**

Tables 8, 9 and 10 show that, like other networks of second-order algebras, holomorphic dual models show better metrics than the corresponding real-valued models, only slightly behind the dual models. In practice, the inference speedup depends on the architecture and is 10-25% compared to dual models, which may be worth the trade-off with some accuracy. In addition, holomorphic models have about half as many parameters as dual models of the same architecture. These advantages make holomorphic dual networks a viable option for hardware constraints.

In conclusion, the main results of the work are formulated.

References

[1] M. Arjovsky, A. Shah, and Y. Bengio, “Unitary Evolution Recurrent Neural Networks,” vol. 48, 2015, [Online]. Available: http://arxiv.org/abs/1511.06464

[2] T. Nitta, “The computational power of complex-valued neuron,” in *Joint International Conference ICANN/ICONIP*, 2003, pp. 993–1000.

[3] R. Chakraborty, Y. Xing, and S. X. Yu, “SurReal: Complex-Valued Learning as Principled Transformations on a Scaling and Rotation Manifold,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 3, pp. 940–951, 2022, doi: 10.1109/TNNLS.2020.3030565.

[4] U. Singhal, Y. Xing, and S. X. Yu, “Co-domain Symmetry for Complex-Valued Deep Learning,” Dec. 2021, Accessed: Mar. 14, 2022. [Online]. Available: http://arxiv.org/abs/2112.01525

[5] J. Bassey, L. Qian, and X. Li, “A Survey of Complex-Valued Neural Networks,” 2021, [Online]. Available: http://arxiv.org/abs/2101.12249

[6] B. Widrow, J. M. McCool, and M. Ball, “The complex LMS algorithm,” *Proc. IEEE*, vol. 63, pp. 719–720, 1975.

[7] A. Hirose and S. Yoshida, “Generalization Characteristics of Complex-Valued Feedforward Neural Networks in Relation to Signal Coherence,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 23, pp. 541–551, 2012.

[8] T. Kim and T. Adalı, “Fully complex backpropagation for constant envelope signal processing,” *Neural Networks Signal Process. X. Proc. 2000 IEEE Signal Process. Soc. Work. (Cat. No.00TH8501)*, vol. 1, pp. 231–240 vol.1, 2000.

[9] T. Kim and T. Adalı, “Fully Complex Multi-Layer Perceptron Network for

Nonlinear Signal Processing," *J. VLSI signal Process. Syst. signal, image video Technol.*, vol. 32, pp. 29–43, 2002.

[10] S. Scardapane, S. Van Vaerenbergh, A. Hussain, and A. Uncini, "Complex-Valued Neural Networks With Nonparametric Activation Functions," *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 4, pp. 140–150, 2018.

[11] Y. Quan, D. Li, W. Zhenyong, C. Liu, and C. He, "Channel Estimation and Pilot Design for Uplink Sparse Code Multiple Access System based on Complex-Valued Sparse Autoencoder," *IEEE Access*, 2019.

[12] M. Catelani *et al.*, "MLMVNN for parameter fault detection in PWM DC-DC converters and its applications for buck DC-DC converter," *2016 IEEE 16th Int. Conf. Environ. Electr. Eng.*, pp. 1–6, 2016.

[13] A. Marseet and F. Sahin, "Application of complex-valued convolutional neural network for next generation wireless networks," *2017 IEEE West. New York Image Signal Process. Work.*, pp. 1–5, 2017.

[14] I. Cha and S. A. Kassam, "Channel Equalization Using Adaptive Complex Radial Basis Function Networks," *IEEE J. Sel. Areas Commun.*, vol. 13, pp. 122–131, 1995.

[15] S. Chen, S. McLaughlin, and B. Mulgrew, "Complex-valued radial basic function network, Part I: Network architecture and learning algorithms," *Signal Process.*, vol. 35, pp. 19–31, 1994.

[16] S. Chen, S. Mclaughlin, and B. Mulgrew, "Complex-valued radial basis function network, Part II: Application to digital communications channel equalisation," *Signal Process.*, vol. 36, pp. 175–188, 1994.

[17] D. Jianping, N. Sundararajan, and P. Saratchandran, "Communication channel

equalization using complex-valued minimal radial basis function neural networks," *IEEE Trans. neural networks*, vol. 13 3, pp. 687–696, 2002.

[18] W. Gong, J. Liang, and D. Li, "Design of high-capacity auto-associative memories based on the analysis of complex-valued neural networks," *2017 Int. Work. Complex Syst. Networks*, pp. 161–168, 2017.

[19] A. Uncini, L. Vecci, P. Campolucci, and F. Piazza, "Complex-valued neural networks with adaptive spline activation function for digital-radio-links nonlinear equalization," *IEEE Trans. Signal Process.*, vol. 47, pp. 505–514, 1999.

[20] M. Scarpiniti, D. Vigliano, R. Parisi, and A. Uncini, "Generalized splitting functions for blind separation of complex signals," *Neurocomputing*, vol. 71, pp. 2245–2270, 2008.

[21] R. Huang and M.-S. Chen, "Adaptive equalization using complex-valued multilayered neural network based on the extended Kalman filter," *WCC 2000 - ICSP 2000. 2000 5th Int. Conf. Signal Process. Proceedings. 16th World Comput. Congr. 2000*, vol. 1, pp. 519–524 vol.1, 2000.

[22] M. Solazzi, A. Uncini, E. D. Di Claudio, and R. Parisi, "Complex discriminative learning Bayesian neural equalizer," *ISCAS'99. Proc. 1999 IEEE Int. Symp. Circuits Syst. VLSI (Cat. No.99CH36349)*, vol. 5, pp. 343–346 vol.5, 1999.

[23] N. Benvenuto, M. Marchesi, F. Piazza, and A. Uncini, "Non linear satellite radio links equalized using blind neural networks," *[Proceedings] ICASSP 91 1991 Int. Conf. Acoust. Speech, Signal Process.*, pp. 1521–1524 vol.3, 1991.

[24] A. B. Suksmono and A. Hirose, "Adaptive Beamforming by Using Complex-Valued Multi Layer Perceptron," 2003.

[25] A. Hirose and M. Kiuchi, “Coherent optical associative memory system that processes complex-amplitude information,” *IEEE Photonics Technol. Lett.*, vol. 12, pp. 564–566, 2000.

[26] Y. Chistyakov, E. Kholodova, A. Minin, H.-G. Zimmermann, and A. Knoll, “Modeling of electric power transformer using complex-valued neural networks,” 2011.

[27] A. Minin, Y. Chistyakov, E. Kholodova, H.-G. Zimmermann, and A. Knoll, “Complex Valued Open Recurrent Neural Network for Power Transformer Modeling,” *J. Appl. Math. \& informatics*, vol. 6, pp. 41–48, 2012.

[28] J. Zhang and Y. Wu, “A New Method for Automatic Sleep Stage Classification,” *IEEE Trans. Biomed. Circuits Syst.*, vol. 11, pp. 1097–1110, 2017.

[29] S. Liu *et al.*, “A Multilevel Artificial Neural Network Nonlinear Equalizer for Millimeter-Wave Mobile Fronthaul Systems,” *J. Light. Technol.*, vol. 35, pp. 4406–4417, 2017.

[30] M. Peker, B. Şen, and D. Delen, “A Novel Method for Automated Diagnosis of Epilepsy Using Complex-Valued Classifiers,” *IEEE J. Biomed. Heal. Informatics*, vol. 20, pp. 108–118, 2016.

[31] S. Hu and A. Hirose, “Proposal of Millimeter-Wave Adaptive Glucose-Concentration Estimation System Using Complex-Valued Neural Networks,” *IGARSS 2018 - 2018 IEEE Int. Geosci. Remote Sens. Symp.*, pp. 4074–4077, 2018.

[32] S. Hu, S. Nagae, and A. Hirose, “Millimeter-Wave Adaptive Glucose Concentration Estimation With Complex-Valued Neural Networks,” *IEEE Trans. Biomed. Eng.*, vol. 66, pp. 2065–2071, 2017.

[33] Y. Suzuki and M. Kobayashi, “Complex-valued bidirectional auto-associative memory,” *2013 Int. Jt. Conf. Neural Networks*, pp. 1–7, 2013.

[34] T. Ding and A. Hirose, “Fading Channel Prediction Based on Combination of Complex-Valued Neural Networks and Chirp Z-Transform,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 25, pp. 1686–1695, 2014.

[35] C.-A. Popa, “Deep Hybrid Real-Complex-Valued Convolutional Neural Networks for Image Classification,” *2018 Int. Jt. Conf. Neural Networks*, pp. 1–6, 2018.

[36] M. Matlacz and G. Sarwas, “Crowd counting using complex convolutional neural network,” *2018 Signal Process. Algorithms, Archit. Arrange. Appl.*, pp. 88–92, 2018.

[37] I. N. Aizenberg and A. Gonzalez, “Image Recognition using MLMVN and Frequency Domain Features,” *2018 Int. Jt. Conf. Neural Networks*, pp. 1–8, 2018.

[38] P. Virtue, S. X. Yu, and M. Lustig, “Better than real: Complex-valued neural nets for MRI fingerprinting,” *2017 IEEE Int. Conf. Image Process.*, pp. 3953–3957, 2017.

[39] E. Aizenberg and I. N. Aizenberg, “Batch linear least squares-based learning algorithm for MLMVN with soft margins,” *2014 IEEE Symp. Comput. Intell. Data Min.*, pp. 48–55, 2014.

[40] I. N. Aizenberg, S. Alexander, and J. Jackson, “Recognition of Blurred Images Using Multilayer Neural Network Based on Multi-valued Neurons,” *2011 41st IEEE Int. Symp. Mult. Log.*, pp. 282–287, 2011.

[41] I. N. Aizenberg, D. Paliy, J. M. Zurada, and J. Astola, “Blur Identification by Multilayer Neural Network Based on Multivalued Neurons,” *IEEE Trans. Neural*

Networks, vol. 19, pp. 883–898, 2008.

[42] I. N. Aizenberg, N. N. Aizenberg, C. Butakoff, and E. Farberov, "Image recognition on the neural network based on multi-valued neurons," *Proc. 15th Int. Conf. Pattern Recognition. ICPR-2000*, vol. 2, pp. 989–992 vol.2, 2000.

[43] I. N. Aizenberg and C. Butakoff, "Image processing using cellular neural networks based on multi-valued and universal binary neurons," *J. VLSI signal Process. Syst. signal, image video Technol.*, vol. 32, pp. 169–188, 2000.

[44] R. S. Zemel, C. K. I. Williams, and M. C. Mozer, "Lending direction to neural networks," *Neural Networks*, vol. 8, pp. 503–512, 1995.

[45] C. Trabelsi, O. Bilaniuk, Dmitriy Serdyuk, Sandeep Subramanian, J. F. Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, C. Pal, "Deep complex networks," International Conference on Learning Representations, Poster, 2018.

[46] S. Amilia, M. D. Sulistiyo, and R. N. Dayawati, "Face image-based gender recognition using complex-valued neural network," *2015 3rd Int. Conf. Inf. Commun. Technol.*, pp. 201–206, 2015.

[47] M. Miyauchi, M. Seki, A. Watanabe, and A. Miyauchi, "Interpretation of Optical Flow Through Complex Neural Network," 1993.

[48] M. Miyauchi, M. Seki, A. Watanabe, and A. Miyauchi, "Interpretation of optical flow through neural network learning," *[Proceedings] Singapore ICCS/ISITA `92*, pp. 1247–1251 vol.3, 1992.

[49] A. Hirose, T. Higo, and K. Tanizawa, "Holographic Three-Dimensional Movie Generation with Frame Interpolation Using Coherent Neural Networks," *2006 IEEE Int. Jt. Conf. Neural Netw. Proc.*, pp. 492–497, 2006.

[50] A. Hirose, T. Higo, and K. Tanizawa, “Efficient generation of holographic movies with frame interpolation using a coherent neural network,” *IEICE Electron. Express*, vol. 3, pp. 417–423, 2006.

[51] Y. Liu, H. Huang, and T. Huang, “Gain parameters based complex-valued backpropagation algorithm for learning and recognizing hand gestures,” *2014 Int. Jt. Conf. Neural Networks*, pp. 2162–2166, 2014.

[52] R. F. Olanrewaju, O. O. Khalifa, A. H. Abdulla, and A. M. Z. M. Khedher, “Detection of alterations in watermarked medical images using Fast Fourier Transform and Complex-Valued Neural Network,” *2011 4th Int. Conf. Mechatronics*, pp. 1–6, 2011.

[53] R. Hata and K. Murase, “Multi-valued autoencoders for multi-valued neural networks,” *2016 Int. Jt. Conf. Neural Networks*, pp. 4412–4417, 2016.

[54] Y. Kominami, H. Ogawa, and K. Murase, “Convolutional neural networks with multi-valued neurons,” *2017 Int. Jt. Conf. Neural Networks*, pp. 2673–2678, 2017.

[55] C.-A. Popa, “Complex-Valued Deep Boltzmann Machines,” *2018 Int. Jt. Conf. Neural Networks*, pp. 1–8, 2018.

[56] L. Li, L. Wang, F. L. Teixeira, C. Liu, A. Nehorai, and T. jun Cui, “DeepNIS: Deep Neural Network for Nonlinear Electromagnetic Inverse Scattering,” *IEEE Trans. Antennas Propag.*, vol. 67, pp. 1819–1825, 2018.

[57] S. Gu and L. Ding, “A Complex-Valued VGG Network Based Deep Learing Algorithm for Image Recognition,” *2018 Ninth Int. Conf. Intell. Control Inf. Process.*, pp. 340–343, 2018.

[58] D. Hayakawa, T. Masuko, and H. Fujimura, “Applying Complex-Valued Neural

Networks to Acoustic Modeling for Speech Recognition," *2018 Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, pp. 1725–1731, 2018.

[59] M. Kataoka, M. Kinouchi, and M. Hagiwara, "Music information retrieval system using complex-valued recurrent neural networks," *SMC'98 Conf. Proceedings. 1998 IEEE Int. Conf. Syst. Man, Cybern. (Cat. No.98CH36218)*, vol. 5, pp. 4290–4295 vol.5, 1998.

[60] M. Kinouchi and M. Hagiwara, "Memorization of melodies by complex-valued recurrent network," *Proc. Int. Conf. Neural Networks*, vol. 2, pp. 1324–1328 vol.2, 1996.

[61] A. Y. H. Al-Nuaimi, M. F. Amin, and K. Murase, "Enhancing MP3 encoding by utilizing a predictive Complex-Valued Neural Network," *2012 Int. Jt. Conf. Neural Networks*, pp. 1–6, 2012.

[62] Y.-S. Lee, C.-Y. Wang, S.-F. Wang, J.-C. Wang, and C.-H. Wu, "Fully complex deep neural network for phase-incorporating monaural source separation," *2017 IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 281–285, 2017.

[63] C. S. Tay, K. Tanizawa, and A. Hirose, "Error Reduction in Holographic Movies Using a Hybrid Learning Method in Coherent Neural Networks,", ICANN, Lecture Notes in Computer Science, vol 4668, Springer, 2007.

[64] J. Gao, B. Deng, Y. Qin, H. Wang, and X. Li, "Enhanced Radar Imaging Using a Complex-Valued Convolutional Neural Network," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, pp. 35–39, 2017.

[65] M. Wilmanski, C. Kreucher, and A. O. Hero, "Complex input convolutional neural networks for wide angle SAR ATR," *2016 IEEE Glob. Conf. Signal Inf. Process.*,

pp. 1037–1041, 2016.

[66] I. N. Aizenberg and C. Moraga, “Multilayer Feedforward Neural Network Based on Multi-valued Neurons (MLMVN) and a Backpropagation Learning Algorithm,” *Soft Comput.*, vol. 11, pp. 169–183, 2006.

[67] K. Oyama and A. Hirose, “Adaptive phase-singular-unit restoration with entire-spectrum-processing complex-valued neural networks in interferometric SAR,” *Electron. Lett.*, vol. 54, pp. 43–45, 2018.

[68] X. Yao, X. Shi, and F. Zhou, “Complex-Value Convolutional Neural Network for Classification of Human Activities,” *2019 6th Asia-Pacific Conf. Synth. Aperture Radar*, pp. 1–6, 2019.

[69] X. Yao, X. Shi, and F. Zhou, “Human Activities Classification Based on Complex-Value Convolutional Neural Network,” *IEEE Sens. J.*, vol. 20, pp. 7169–7180, 2020.

[70] T. Dong and T. Huang, “Neural Cryptography Based on Complex-Valued Neural Network,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 31, pp. 4999–5004, 2019.

[71] S. Jankowski, A. Lozowski, and J. M. Zurada, “Complex-valued multistate neural associative memory,” *IEEE Trans. neural networks*, vol. 7 6, pp. 1491–1496, 1996.

[72] T. Miyajima, F. Baisho, K. Yamanaka, K. Nakamura, and M. Agu, “A Phasor Model with Resting States,” *IEICE Trans. Inf. Syst.*, vol. 83, pp. 299–301, 2000.

[73] H. H. Cevik, Y. E. Acar, and M. Çunkaş, “Day Ahead Wind Power Forecasting Using Complex Valued Neural Network,” *2018 Int. Conf. Smart Energy Syst. Technol.*, pp. 1–6, 2018.

[74] T. Kitajima and T. Yasuno, “Output prediction of wind power generation system using complex-valued neural network,” *Proc. SICE Annu. Conf. 2010*, pp. 3610–

3613, 2010.

[75] D. P. Mandic, S. Javidi, S. L. Goh, A. Kuh, and K. Aihara, “Complex-valued prediction of wind profile using augmented complex statistics,” *Renew. Energy*, vol. 34, pp. 196–201, 2009.

[76] Y. Maeda, T. Fujiwara, and H. Ito, “Robot control using high dimensional neural networks,” *2014 Proc. SICE Annu. Conf.*, pp. 738–743, 2014.

[77] I. Nishikawa, K. Sakakibara, T. Iritani, and Y. Kuroe, “2 types of complex-valued Hopfield networks and the application to a traffic signal control,” *Proceedings. 2005 IEEE Int. Jt. Conf. Neural Networks, 2005.*, vol. 2, pp. 782–787 vol. 2, 2005.

[78] I. Nishikawa, T. Iritani, and K. Sakakibara, “Improvements of the Traffic Signal Control by Complex-Valued Hopfield Networks,” *2006 IEEE Int. Jt. Conf. Neural Netw. Proc.*, pp. 459–464, 2006.

[79] J. Hu, Z. Li, Z. Hu, D. Yao, and J. Yu, “Spam Detection with Complex-Valued Neural Network Using Behavior-Based Characteristics,” *2008 Second Int. Conf. Genet. Evol. Comput.*, pp. 166–169, 2008.

[80] F. M. Dimentberg, *The screw calculus and its applications in mechanics*. WP-AFB, Ohio : Foreign Technology Division, 1968.

[81] A. Güneş Baydin, B. A. Pearlmutter, A. Andreyevich Radul, and J. Mark Siskind, “Automatic differentiation in machine learning: A survey,” *J. Mach. Learn. Res.*, vol. 18, pp. 1–43, 2018.

[82] R. Kiran and K. Khandelwal, “Automatic implementation of finite strain anisotropic hyperelastic models using hyper-dual numbers,” *Comput. Mech.*, vol. 55, no. 1, pp. 229–248, 2015, doi: 10.1007/s00466-014-1094-1.

[83] Y. Okawa and T. Nitta, “Learning Properties of Feedforward Neural Networks Using Dual Numbers,” in *Proceedings, APSIPA Annual Summit and Conference*, 2021, pp. 187–192.

[84] The G2Net Dataset: [site]: URL: https://www.kaggle.com/competitions/g2net-gravitational-wave-detection/data.

[85] B. P. Abbott and others, “Observation of Gravitational Waves from a Binary Black Hole Merger,” Phys. Rev. Lett., vol. 116, no. 6, 2016, doi: 10.1103/PhysRevLett.116.061102.

[86] R. M. Shannon and others, “Gravitational waves from binary supermassive black holes missing in pulsar observations,” Science (80-.)., vol. 349, no. 6255, pp. 1522–1525, 2015, doi: 10.1126/science.aab1910.

[87] K. S. Thorne, “Gravitational Waves from Compact Bodies.” arXiv, 1995. doi: 10.48550/ARXIV.GR-QC/9506084.

[88] J. Thickstun, Z. Harchaoui, and S. M. Kakade, “Learning features of music from scratch,” 5th Int. Conf. Learn. Represent. ICLR 2017 - Conf. Track Proc., pp. 1–14, 2017.

[89] J. O. Smith, “Digital Audio Resampling.” 2002. [Online]. Available: https://ccrma.stanford.edu/~jos/resample/

[90] D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., pp. 1–15, 2015.

[91] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-Decem, pp. 770–778, Dec. 2016, doi: 10.1109/CVPR.2016.90.