Journal of Computers Vol. 33 No. 3, June 2022, pp. 73-84
doi: 10.53106/199115992022063303006

Coordinate Transformation of Buoy Based on Fractional Complex Valued Neural Network

Zhifu Guo[1], Han Xue[2*]

[1] Xiamen AIDS To Navigation Department of Donghai Navigation Safety Administration, Xiamen, Fujian, China
510948550@qq.com
[2] College of Navigation, Jimei University, Xiamen, Fujian, China.
imlmd@163.com

Received 14 July 2021; Revised 14 November 2021; Accepted 14 December 2021

Abstract. To reduce the error of coordinate transformation, a fractional order convex valued neural network (FCVNN) is explored. The convergence is proved. We take the longitude as the real part of the input, and take the latitude as the imaginary part of the input. Thus, we construct the complex valued input of the CVNN. All longitudes and latitudes on the earth are perpendicular to each other. This satisfies that the real part and imaginary part of the complex form an orthogonal unit basis. Input Xiamen 92 space coordinates and WGS-84 space coordinates obtained from the transformation as the training samples. The weights of neural network are updated. Input the geodetic coordinates of test data and output the geodetic coordinates of WGS84 corresponding to the results. FCVNN is applied to Xiamen 92 coordinate transformation, and the transformation accuracy is improved. Using the orthogonality of longitude and latitude, CVNN are constructively used to solve the coordinate transformation problem. Many insights can be transferred from real domain to complex domain. If the data exists naturally in the complex domain, or can be meaningfully moved to the complex plane, the complex neural network should be used for the task.

Keywords: seven parameters method, coordinate transformation, neural network, Xiamen 92 coordinate, error

1 Introduction

The seven parameter model is usually used for the transformation between two different three-dimensional Cartesian coordinate systems. In general, the least square method is used to calculate seven parameters [1-2]. In order to reduce the error of coordinate transformation, Wang et al. [3] proposed an improved artificial bee colony algorithm for seven parameter calculation of coordinate transformation. Chen et al. [4] designed coordinate transformation based on BP neural network. Cui et al. [5] proposed the convolution neural network GPS coordinate transformation method.

Neural network has been used for coordinate transformation. Complex valued neural network (CVNN) has attracted extensive attention in academic circles because of its strong mapping ability and good adaptability. CVNN has been applied to areas, especially in signal processing, in which the input data has natural interpretation in complex domain. The use of complex numbers allows neural networks to deal with noise on complex planes. Different strategies can be used to transfer many activation functions from real domain to complex domain. Chen et al. [6] studied the master-slave synchronization of CVNN based on event triggering. Song et al. [7] discussed the input state exponential stability of stochastic CVNN with neutral and discrete delays. Huang et al. [8] studied the finite time passivity and finite time synchronization of two classes of coupled memristor CVNN with and without time-varying delays. Li et al. [9] used the deep CVNN trained by alternating frequency phase amplitude coupling features to separate different regions. Song et al. [10] discussed the synchronization problem of fractional CVNN with reaction-diffusion term in finite time interval. Aouiti et al. [11] considered the finite time and fixed time synchronization problem of time-delay recurrent CVNN controlled by sliding mode with discontinuous activation function and different parameters.

However, the neural network itself has some shortcomings, gradient descent method is an effective method for training neural network [12-14]. Fractional calculus was born in 1695, it can deal with disturbance, fast response, small overshoot, small chattering effect, has good control performance [15-20]. Wang et al [21] proposed a fractional gradient descent method for BP neural network training, and estimated the error fractional gradient defined by traditional quadratic energy function with Caputo derivative. Two simulation results show the performance of the proposed fractional order BP algorithm on three small data sets and one large data set. Khan et al. [22] proposed a new radial basis function neural network fractional gradient descent learning algorithm, which is a con-

* Corresponding Author

vex combination of the traditional fractional gradient descent method and the improved Riemannian derivative. The optimal solution of the system identification problem is analyzed, and the closed form Wiener solution of the least square problem is obtained. The weight updating rules of fractional order RBF neural network are derived. It is suitable for four main estimation problems: nonlinear system identification, pattern classification, time series and function approximation.

However, complex valued neural networks face some essential problems, such as the selection standard of complex activation function is not clear, the traditional cost function has no complex derivative, the learning algorithm is too complex, and the algorithm lacks convergence analysis, which limits the effective application of CVNN. In this paper, we take the longitude as the real part of the input, and take the latitude as the imaginary part of the input. Thus, we construct the complex valued input of the CVNN. All longitudes and latitudes on the earth are perpendicular to each other. This satisfies that the real part and imaginary part of the complex form an orthogonal unit basis.

In this paper, the fractional order complex valued neural network is designed, its convergence is proved, and the influence of error covariance is analyzed. FCVNN is applied to Xiamen 92 coordinate transformation, and the transformation accuracy is improved. The contributions are summarized blow.

(1) A novel fractional convex valued neural network (FCVNN) is explored.

(2) Its convergence is proved.

(3) We take the longitude as the real part of the input, and take the latitude as the imaginary part of the input. Thus, we construct the complex valued input of the CVNN. All longitudes and latitudes on the earth are perpendicular to each other. This satisfies that the real part and imaginary part of the complex form an orthogonal unit basis.

2. Related Work

2.1. CVNN

In CVNN, the input is z:

$$z = z_R + iz_I, \tag{1}$$

$$\varphi = [\varphi_1, \varphi_2, \cdots, \varphi_P], \tag{2}$$

Where z_R is the real part of the input vector. z_I is the imaginary part of z. $\| x \|$ is the Euclidean norm. φ denotes the neurons excitation function. H is the hidden nodes number. w is the weight connecting the hidden layer and output layer. N denotes the input nodes number. w_R is real part of the w. w_I is imaginary part of w. o denotes the output of neural network:

$$o = W(n)\varphi(z(n)), \tag{3}$$

$$W(n) = [w_1(n)\ w_2(n) \cdots, w_H(n)], \quad w(n) = w_R(n) + w_I(n), \tag{4}$$

The CVNN frame is in Fig. 1:

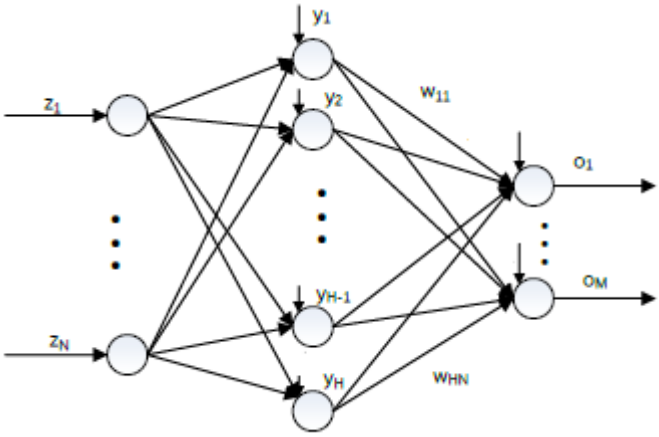

Fig. 1. CVNN frame

2.2. Riemann-Liouville Fractional Order Calculus

Denote t as the integral end point. Denote t_0 as the integral starting point. For a function x which is defined in $[t_0,t]$, the fractional integrator is:

$$ {}_{t_0}D_t^{\alpha}x(t)=\frac{1}{\Gamma(\alpha)}\int_{t_0}^{t}(t-\tau)^{\alpha-1}x(\tau)d\tau, \tag{5}$$

where α is the fractional order. τ is the integral variable. Γ is the gamma function:

$$ \Gamma(x)=\int_0^{\infty}e^{-\tau}\tau^{x-1}d\tau, \tag{6}$$

The fractional derivative is:

$$ {}_{t_0}D_t^{\alpha}x(t)=\frac{d^m}{dt^n}\left[\frac{1}{\Gamma(m-\alpha)}\int_{t_0}^{t}(t-\tau)^{m-\alpha-1}x(\tau)d\tau\right], \tag{7}$$

where $\alpha\in[m-1,m)$. m is a positive integer near α.

Lemma 1. For the function $f(x)=x^m$ when $0\le m\le\alpha<m+1$, the following formula holds:

$$ {}_{x_0}D_x^{\alpha}f(x)=\frac{\Gamma(m+1)}{\Gamma(m-\alpha+1)}x^{m-\alpha}, \tag{8}$$

2.3 Seven Parameter Space Coordinate Transformation

The seven parameters are divided into three categories: rotation, scaling and translation. Let K be the scale change parameter, (△ X, △ Y, △ Z) be the translation change parameter, and (r_x, r_y, r_z) be the rotation parameter. The seven parameter conversion formula is as follows:

$$ \begin{bmatrix}X_2\\Y_2\\Z_2\end{bmatrix}=\begin{bmatrix}\Delta X\\\Delta Y\\\Delta Z\end{bmatrix}+(1+k)\begin{bmatrix}1&r_z&r_y\\-r_z&1&r_x\\r_y&-r_x&1\end{bmatrix}\begin{bmatrix}X_1\\Y_1\\Z_1\end{bmatrix} \tag{9}$$

At the end of 1995, the Xiamen new coordinate system, which was established by Xiamen second and third class GPS network, was put out of use at the same time as the old Xiamen coordinate system. The main parameters of 92 coordinate system are shown in Table 1.

Table 1. Main parameters of Xiamen 92 coordinate system

Ellipsoid	Beijing 54 ellipsoid (krasov ellipsoid)
Projection mode	Gauss projection
Central longitude	118.5°
Scale factor	1.0
East coordinates	-400000
North coordinates	-2700000

3 Main Results

3.1 FCVNN Algorithm

Define d as the expected output. d_R is the real part of d. d_I is the imaginary part of d. The square error function is defined as:

$$E = \frac{1}{2}\sum_{n=1}^{H}[(w\varphi_R(n) - d_R)^2 + (w\varphi_I(n) - d_I)^2], \tag{10}$$

$$d = d_R + d_I, \tag{11}$$

Denote:

$$g_R = \frac{1}{2}(w\varphi_R - d_R)^2, \tag{12}$$

$$g_I = \frac{1}{2}(w\varphi_I - d_I)^2, \tag{13}$$

Differentiating (12) can obtain:

$$g'_R = y'(w\varphi_R - d_R), \tag{14}$$

Differentiating (13) can obtain:

$$g'_I = y'(w\varphi_I - d_I), \tag{15}$$

Taking the second derivative of (12) can obtain:

$$g''_R = y''(w\varphi_R - d_R) + (y')^2, \tag{16}$$

Taking the second derivative of (13) can obtain:

$$g''_I = y''(w\varphi_I - d_I) + (y')^2, \tag{17}$$

The loss function can be minimized by adjusting the weight. Denote Δw_R^n as the adjustment increment of w_R.

Denote Δw_I^n as the adjustment increment of w_I.

$$\Delta w_R^n = w_R^{n+1} - w_R^n, \tag{18}$$

$$\Delta w_I^n = w_I^{n+1} - w_I^n, \tag{19}$$

$$\Delta w^n = \Delta w_R^n + i\Delta w_I^n, \tag{20}$$

The updating rule is:

$$\Delta w_R^n = -\lambda D_{w_R}^{\alpha} E, \tag{21}$$

$$\Delta w_I^n = -\lambda D_{w_I}^{\alpha} E, \tag{22}$$

where is $\lambda > 0$ the learning rate.

Differentiating E with respect to $\mathbf{w_R}$ can obtain:

$$D_{w_R}^{\alpha} E = g'_R \varphi_R \frac{w_R^{1-\alpha}}{\Gamma(2-\alpha)}, \tag{23}$$

Differentiating E with respect to w_I can obtain:

$$D_{w_I}^{\alpha} E = g'_I \varphi_I \frac{w_I^{1-\alpha}}{\Gamma(2-\alpha)}, \tag{24}$$

Substituting (23) into (21) can obtain

$$\Delta w_R^n = -\lambda g'_R \varphi_R \frac{w_R^{1-\alpha}}{\Gamma(2-\alpha)}, \tag{25}$$

Substituting (24) into (22) can obtain

$$\Delta w_I^n = -\lambda g'_I \varphi_I \frac{w_I^{1-\alpha}}{\Gamma(2-\alpha)}, \tag{26}$$

3.2 Convergence Analysis

Assume 1. $|\varphi|$, $|g'|$, $|g''|$, $|w^{1-\alpha}|$ are uniformly bounded. There exist constant $c_0>0$, $c_1>0$, $c_2>0$ such that:

$$|\varphi_I| \le c_0, \tag{27}$$

$$|\varphi_R| \le c_0, \tag{28}$$

$$|g'_R| \le c_1, \tag{29}$$

$$|g''_R| \le c_1, \tag{30}$$

$$| g'_I | \le c_1, \tag{31}$$

$$| g''_I | \le c_1, \tag{32}$$

$$| w_R^{1-\alpha} | \le c_2, \tag{33}$$

$$| w_I^{1-\alpha} | \le c_2, \tag{34}$$

Theorem 1. Suppose the assumptions (27)-(32) are valid, and the weight of CVNN is updated by (25)-(26). Then one has:

$$E(w^{n+1}) \le E(w^n) \tag{35}$$

$$\lim_{n\to\infty} D^{\alpha}_{w_R} E = 0, \tag{36}$$

$$\lim_{n\to\infty} D^{\alpha}_{w_I} E = 0, \tag{37}$$

Proof of Theorem 1. It can be deduced from (3):

$$\begin{aligned} & g'_R(o_R^{n+1} - o_R^n) + g'_I(o_I^{n+1} - o_I^n) \\ & = g'_R(\varphi_R w_R^{n+1} - \varphi_R w_R^n) + g'_I \varphi_I(\varphi_I w_I^{n+1} - \varphi_I w_I^n) \\ & = g'_R \varphi_R \Delta w_R + g'_I \varphi_I \Delta w_I, \end{aligned} \tag{38}$$

From (25) one can obtain:

$$g'_R \varphi_R = -\frac{\Gamma(2-\alpha)}{\lambda w_R^{1-\alpha}} \Delta w_R^n, \tag{39}$$

From (26) one can obtain:

$$g'_I \varphi_I = -\frac{\Gamma(2-\alpha)}{\lambda w_I^{1-\alpha}} \Delta w_I^n, \tag{40}$$

Substituting (39) and (40) into (38) can obtain

$$\begin{aligned} & g'_R(o_R^{n+1} - o_R^n) + g'_I(o_I^{n+1} - o_I^n) \\ & = -\frac{\Gamma(2-\alpha)}{\lambda w_R^{1-\alpha}} \| \Delta w_R^n \|^2 - \frac{\Gamma(2-\alpha)}{\lambda w_I^{1-\alpha}} \| \Delta w_I^n \|^2 \\ & \le -\frac{\Gamma(2-\alpha)}{\lambda c_2} (\| \Delta w_R \|^2 + \| \Delta w_I \|^2), \end{aligned} \tag{41}$$

From (30) and (32) one can obtain:

$$\begin{aligned}
&\sum^{H} g''_R(o_R^{n+1}-y_R^n)^2+g''_I(o_I^{n+1}-o_I^n)^2\\
&\le \frac{c_1}{2}\sum^{H}(o_R^{n+1}-o_R^n)^2+(o_I^{n+1}-o_I^n)^2\\
&=\frac{c_1}{2}\sum^{H}(\varphi_R w_R^{n+1}-\varphi_R w_R^n)^2+(\varphi_I w_I^{n+1}-\varphi_I w_I^n)^2\\
&=\frac{c_1}{2}\sum^{H}(\varphi_R \Delta w_R^n)^2+(\varphi_I \Delta w_I^n)^2\\
&\le c_1 H(c_0^2\,||\,\Delta w_R^n\,||^2+c_0^2\,||\,\Delta w_I^n\,||^2)\\
&=c_1 H c_0^2(||\,\Delta w_R^n\,||^2+||\,\Delta w_I^n\,||^2),
\end{aligned} \tag{42}$$

It can be deduced from the Taylor mean value theorem with Lagrange remainder:

$$\begin{aligned}
&E(w^{n+1})-E(w^n)\\
&=\sum g(o_R^{n+1})-g(o_R^n)+g(o_I^{n+1})-g(o_I^n)\\
&=\sum g'_R(o_R^{n+1}-o_R^n)+g'_I(o_I^{n+1}-o_I^n)+g''_R(o_R^{n+1}-o_R^n)^2+g''_I(o_I^{n+1}-o_I^n)^2,
\end{aligned} \tag{43}$$

Substituting (41) and (42) into (43) can obtain

$$\begin{aligned}
&E(w^{n+1})-E(w^n)\\
&\le -\frac{\Gamma(2-\alpha)}{\lambda c_2}(||\,\Delta w_R^n\,||^2+||\,\Delta w_I^n\,||^2)+c_1 H c_0^2(||\,\Delta w_R^n\,||^2+||\,\Delta w_I^n\,||^2)\\
&=-(\frac{\Gamma(2-\alpha)}{\lambda c_2}-c_1 H c_0^2)(||\,\Delta w_R^n\,||^2+||\,\Delta w_I^n\,||^2),
\end{aligned} \tag{44}$$

We take:

$$\lambda<\frac{\Gamma(2-\alpha)}{c_1 c_2 c_0^2 H}, \tag{45}$$

Then, the following results hold:

$$E(w^{n+1})\le E(w^n), \tag{46}$$

This indicates the monotonicity error function E. Since $E(w^n)\ge 0$ and each bounded monotonic sequence converges, $E(w^n)$ is convergent. Therefore, there exists $E^*\ge 0$

$$\lim_{n\to+\infty} E(w^n)=E^*, \tag{47}$$

Denote

$$\eta = \frac{\Gamma(2-\alpha)}{\lambda c_2} - c_1 H c_0^2, \tag{48}$$

Substituting (48) into (44) yields:

$$E(w^{n+1}) - E(w^n) \le -\eta \sum_{i=1}^{n} || \Delta w_R^n ||^2 + || \Delta w_I^n ||^2, \tag{49}$$

Thus, we have:

$$E(w^{n+1}) \le E(w^1) - \eta \sum_{i=1}^{n} || \Delta w_R^n ||^2 + || \Delta w_I^n ||^2 < \infty, \tag{50}$$

Since $E(w^{n+1}) \ge 0$, we can get:

$$\eta \sum_{i=1}^{n} || \Delta w_R^n ||^2 + || \Delta w_I^n ||^2 \le E(w^1) < \infty, \tag{51}$$

When $n \to +\infty$, it holds that

$$\lim_{n \to +\infty} || \Delta w_R^n ||^2 + || \Delta w_I^n ||^2 = 0, \tag{52}$$

Thus, we have

$$\lim_{n \to +\infty} || \Delta w_R^n ||^2 = 0, \tag{53}$$

$$\lim_{n \to +\infty} || \Delta w_I^n ||^2 = 0, \tag{54}$$

When $n \to +\infty$, it holds that:

$$\lim_{n \to +\infty} || w^{n+1} - w^n || = 0, \tag{55}$$

Thus there exists a w^*:

$$\lim_{n \to +\infty} w^n = w^*. \tag{56}$$

3.3 Coordinate Transformation of Light Buoy Based on Neural Network

The model of FCVNN algorithm is as follows:

Step 1: initialize the neural network model;

Step 2: set the Gaussian projection parameters: central meridian, x-coordinate constant and Y coordinate constant, and scale factor;

Step 3: the longitude and latitude of geodetic coordinate under WGS84 ellipsoid are transformed into the space rectangular coordinates of WGS84 by using WGS84 ellipsoid parameters;

Step 4: the plane rectangular coordinates of Xiamen 92 are projected, and then converted to the spatial rectan-

gular coordinates of Xiamen 92 after the krasovsky ellipsoid parameters are used.

Step 5: input training samples, input the Xiamen 92 spatial coordinates and WGS-84 spatial coordinates obtained from the transformation of several known points;

Step 6: update the weight of neural network;

Step 7: calculate the hidden layer;

Step 8: calculate the output layer;

Step 9: calculate the output error;

Step 10: if the error does not meet the learning requirements, turn step 4 until the training results meet the requirements;

Step 11: initialize the neural network with the obtained weight;

Step 12: input the geodetic coordinates of Xiamen 92, and output the geodetic coordinates of WGS84 corresponding to the results.

4.Experiment

4.1 Coordinate Transformation of Points

WGS84 and Xiamen 92 coordinate systems are selected as test data. The conversion results of this example are shown in Table 2 to Table 4.

Table 2. Coordinates of control points in safe operation area of drilling zone 1

Points	X	Y	Latitude	Longitude
A1	2718385.652	463445.905	24°34’7.32”	118°8’23.56”
A2	2718298.641	463336.796	24°34’4.49”	118°8’19.69”
A3	2717466.634	463165.453	24°34’37.43”	118°8’13.68”
A4	2717347.303	463583.961	24°34’33.57”	118°8’28.57”
A5	2717976.512	463760.505	24°34’54.06”	118°8’34.78”

Table 3. Coordinates of control points in safe operation area of drilling zone 2

	X	Y	Latitude	Longitude
B1	2720887.744	463226.980	24°35’28.62”	118°8’15.55”
B2	2718625.880	463351.561	24°34’15.12”	118°8’20.19”
B3	2718630.374	463671.012	24°34’15.27”	118°8’31.54”
B4	2720898.186	463538.795	24°34’28.97”	118°8’26.63”

Table 4. Coordinates of control points in safe operation area of drilling zone 2

	X	Y	Latitude	Longitude
C1	2723086.338	463063.276	24°36’40.04”	118°8’9.52”
C2	2722709.575	463118.320	24°36’27.82”	118°8’11.51”
C3	2720737.377	463235.378	24°35’23.71”	118°8’15.86”
C4	2720749.385	463548.997	24°35’24.13”	118°8’27.01”
C5	2722757.111	463432.271	24°36’29.38”	118°8’22.67”
C6	2723413.449	463325.049	24°36’50.69”	118°8’18.80”

The drilling is plotted in AutoCAD as shown in Fig. 2.

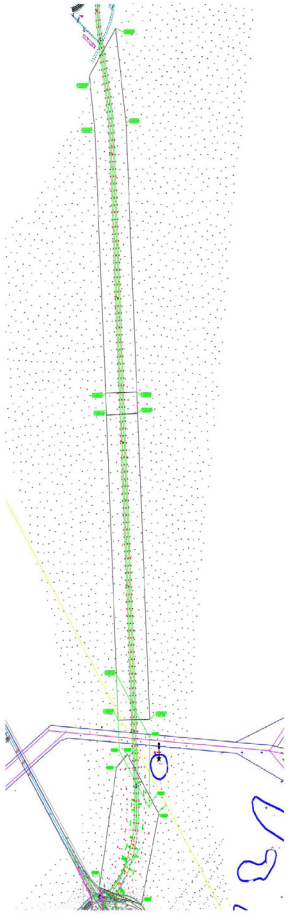

Fig. 2. Points in the AutoCAD

4.2 Coordinate Transformation of Buoys

The conversion results of buoys are shown in Table 5.

Table 5. Conversion examples of buoys

Buoy	X	Y	Latitude	Longitude
209#	2703011.8117	467803.7018	024:25:48.349120	118:10:57.080455
210#	2703018.1032	467480.7968	024:25:48.529424	118:10:45.617442
DX25	2702279.9504	465859.8746	024:25:24.414279	118:09:48.141578
DX27	2700816.8706	463467.9174	024:24:36.668374	118:08:23.371452

4.3 Comparison with Other Algorithms

Compared with the actual results, the checking accuracy is calculated. Comparing the results of this example with those of the least square method, the accuracy error is shown in Table 6.

Table 6. Comparison of conversion accuracy error

Coordinate	Least square method	RBF	FCVNN
X	1.29cm	1.25cm	1.24cm
Y	1.36cm	1.30cm	1.28cm

It can be seen from the above calculation results that compared with the least square method and RBF, FCVNN

neural network can effectively restrain the influence of transformation error and improve the accuracy of coordinate transformation.

4.4 Discussion

Using the orthogonality of longitude and latitude, CVNN are constructively used to solve the coordinate transformation problem. Many insights can be transferred from real domain to complex domain. If the data exists naturally in the complex domain, or can be meaningfully moved to the complex plane, the complex neural network should be used for the task. The study of information and gradient flow can help identify tasks that benefit from complex valued neural networks. Complex value models allow greater degrees of freedom. If the input data has the function of natural mapping to complex numbers, and the noise in the input data is distributed on the complex plane, complex value embedding can be learned from real value data. The introduction of complex numbers as parameters also determines the trade-off between task specific performance and computational cost.

5 Conclusion

In order to reduce the error of coordinate transformation, a fractional complex valued neural network is designed, its convergence is proved, and the influence of error covariance is analyzed. Input training samples, input Xiamen 92 space coordinates and WGS-84 space coordinates WGS84 obtained from the transformation of multiple known points; The weights of neural network are updated; The hidden layer of neural network is calculated; The output layer of neural network is calculated; The output error of neural network is calculated; Until the training results meet the requirements; The weights are used to initialize the neural network; Input the geodetic coordinates of test data and output the geodetic coordinates of WGS84 corresponding to the results. FCVNN is applied to Xiamen 92 coordinate transformation, and the transformation accuracy is improved. Compared with the least square method and RBF neural network, FCVNN can effectively suppress the influence of system error and abnormal error of transformed data, and improve the accuracy of coordinate transformation. Further research will improve the conversion accuracy.

Future work of this research encompasses more experimental validation and improve the conversion accuracy.

6 Acknowledgement

The work is supported by the Science Foundation of the Fujian Province (Grant No. 2021J01819).

References

[1] J. Wang, F. Zhang, Effects of random errors on Bursa seven-parameter transformation model, Science of Surveying and Mapping 41(9)(2016) 20-24.
[2] W. Duan, Y. Xu, L. Wang, D. Zheng, A senven-parameter model based on single ellipsoid of CGCS2000 frame, Science of Surveying and Mapping 42(6)(2017) 50-54.
[3] J. Wang, D. Du, J. Qiu, The improved artificial bee colony algorithm and its application in calculating seven parameters of coordinate transformation, Journal of Anhui University (Natural Science Edition) 39(6)(2015) 23-28.
[4] G. Chen, Coordinate transformation based on BP neural network, Standardization of Surveying and Mapping 34(3)(2018) 61-62.
[5] F. Cui, S. Zhao, GPS coordinates transformation based on convolutional neural network, Bulletin of Surveying and Mapping 3(2019) 1-5.
[6] Q. Chen, H. Bin, Z. Huang, Event-based master-slave synchronization of complex-valued neural networks via pinning impulsive control, Neural Networks 145(2022) 374-385.
[7] Q. Song, Z. Zhao, Y. Liu, F.E. Alsaadi, Mean-square input-to-state stability for stochastic complex-valued neural networks with neutral delay, Neurocomputing 470(2022) 269-277.
[8] Y. Huang, F. Wu, Finite-time passivity and synchronization of coupled complex-valued memristive neural networks, Information Sciences 580(2021) 775-800.
[9] C. Li, S. Liu, Z. Wang, Classifying interictal epileptiform activities in intracranial EEG using complex-valued convolutional neural network, International Journal of Psychophysiology 168(2021) S104-S105.
[10]X. Song, X. Sun, J. Man, S. Song, Q. Wu, Synchronization of fractional-order spatiotemporal complex-valued neural networks in finite-time interval and its application, Journal of the Franklin Institute 358(16)(2021) 8207-8225.
[11]C. Aouiti, M. Bessifi, Sliding mode control for finite-time and fixed-time synchronization of delayed complex-valued recurrent neural networks with discontinuous activation functions and nonidentical parameters, European Journal of Control

59(2021) 109-122.
[12]P. Yin, S. Zhang, J. Lyu, S. Osher, Y. Qi, J. Xin, Blended coarse gradient descent for full quantization of deep neural networks, Research in the Mathematical Sciences 6(2019) 14.
[13]M. Kobayashi, Gradient descent learning for quaternionic Hopfield neural networks, Neurocomputing 260(2017) 174-179.
[14]L. Wang, Y. Yang, R. Min, S. Chakradhar, Accelerating deep neural network training with inconsistent stochastic gradient descent, Neural Networks 93(2017) 219-229.
[15]K. Orman, K. Can, A. Basci, A. Derdiyok, An adaptive-fuzzy fractional-order sliding mode controller design for an unmanned vehicle, Elektronika Ir Elektrotechnika 24(2)(2018) 12-17.
[16]J. Wang, C Shao, Y Chen, Fractional order sliding mode control via disturbance observer for a class of fractional order systems with mismatched disturbance, Mechatronics 53(2018) 8-19.
[17]D. Deepika, S. Kaur, S. Narayan, Uncertainty and disturbance estimator based robust synchronization for a class of uncertain fractional chaotic system via fractional order sliding mode control, Chaos, Solitons & Fractals 115(2018) 196-203.
[18]K.E. Govinda, J. Arunshankar, Control of nonlinear two-tank hybrid system using sliding mode controller with fractional-order PI-D sliding surface, Computers & Electrical Engineering 71(2018) 953-965.
[19]G. Sun, L. Wu, Z. Kuang, Z. Ma, J. Liu, Practical tracking control of linear motor via fractional-order sliding mode, Automatica 94(2018) 221-235.
[20]Y. Wang, S. Jiang, B. Chen, H. Wu, A new continuous fractional-order nonsingular terminal sliding mode control for cable-driven manipulators, Advances in Engineering Software 119(2018) 21-29.
[21]J. Wang, Y. Wen, Y. Gou, Z. Ye, H. Chen, Fractional-order gradient descent learning of BP neural networks with Caputo derivative, Neural Networks 89(2017) 19-30.
[22]S. Khan, I. Naseem, M. A. Malik, R. Togneri, M. Bennamoun, A fractional gradient descent-based RBF neural network, Circuits, Systems, and Signal Processing 37(2018) 5311-5332.