



Article Feedforward Control of Piezoelectric Ceramic Actuators Based on PEA-RNN

Yongcheng Xiong ^{1,2,3}, Wenhong Jia ^{1,2}, Limin Zhang ^{1,2}, Ying Zhao ² and Lifang Zheng ^{1,2,*}

- ¹ Shanghai Institute of Applied Physics, Chinese Academy of Sciences, Shanghai 201800, China; xiongyc@mail.ustc.edu.cn (Y.X.); jiawenhong@zjlab.org.cn (W.J.); zhanglimin@zjlab.org.cn (L.Z.)
- ² Shanghai Advanced Research Institute, Chinese Academy of Sciences, Shanghai 201204, China; zhaoying@zjlab.org.cn
- ³ University of Chinese Academy of Sciences, Beijing 100049, China
- * Correspondence: zhenglifang@zjlab.org.cn; Tel.: +86-021-20302638

Abstract: Multilayer perceptron (MLP) has been demonstrated to implement feedforward control of the piezoelectric actuator (PEA). To further improve the control accuracy of the neural network, reduce the training time, and explore the possibility of online model updating, a novel recurrent neural network named PEA-RNN is established in this paper. PEA-RNN is a three-input, one-output neural network, including one gated recurrent unit (GRU) layer, seven linear layers, and one residual connection in the linear layers. The experimental results show that the displacement linearity error of piezoelectric ceramics reaches 8.96 μ m in the open-loop condition. After using PEA-RNN compensation, the maximum displacement error of piezoelectric ceramics is reduced to 0.465 μ m at the operating frequency of 10 Hz, which proves that PEA-RNN can accurately compensate piezoelectric ceramics' dynamic hysteresis nonlinearity. At the same time, the training epochs of PEA-RNN are only 5% of the MLP, and fewer training epochs provide the possibility to realize online updates of the model in the future.

Keywords: piezoelectric ceramic actuator; hysteretic nonlinearity; recurrent neural network; feedforward control

1. Introduction

The Shanghai Synchrotron Radiation Facility (SSRF), which operates nearly 5000 h a year, has massive data resources, and the high availability of data brings enormous opportunities for the application of artificial intelligence technology in the field of synchrotron radiation. In recent years, SSRF has explored intelligent beamlines based on the differential evolution algorithm and achieved initial success [1]. However, in the online test, the researchers found that, after the introduction of the second crystal pitch axis of the monochromator [2], the excellent individuals were inherited to the next generation due to its poor motion repeatability, resulting in the intelligent algorithm not converging. SSRF uses many PEAs to achieve micro displacements. However, due to the inherent hysteresis nonlinearity and creep properties, and the defects of the control algorithm, the motion accuracy is not high enough, and the motion repeatability is poor. Studies have shown that, without considering the error caused by vibration, the uncertainty introduced by piezoelectric ceramic hysteresis is generally 15–20%, and the creep error is 1–5% [3].

For a long time, the hysteresis nonlinearity and compensation technology of piezoelectric ceramics have been research hotspots for scholars. The early classical feedforward control technology used mathematical methods to establish a fitting model of the hysteresis loop and to obtain the relationship between the excitation voltage and the actual output displacement by solving its inverse model. Physics-based models and phenomenological models [4] were the main classifications of the hysteresis model in the past. The micro-electromechanical models [5] are typical phenomenological models.



Citation: Xiong, Y.; Jia, W.; Zhang, L.; Zhao, Y.; Zheng, L. Feedforward Control of Piezoelectric Ceramic Actuators Based on PEA-RNN. *Sensors* 2022, 22, 5387. https:// doi.org/10.3390/s22145387

Academic Editor: Senentxu Lanceros-Mendez

Received: 4 July 2022 Accepted: 18 July 2022 Published: 19 July 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). Because this classification method is limited, the classifications of rate-independent hysteresis models and rate-dependent hysteresis models according to whether the input rate is considered [6] are more reasonable. Rate-independent hysteresis models include the classic Preisach model [7], the classic Maxwell model [8], and the polynomial model [9]. Furthermore, rate-dependent hysteresis models include the Bouc–Wen mode [10,11] and the micromechanical model [12,13]. Due to the complex nonlinear characteristics, although scholars have proposed many mathematical models, there are certain limitations in the description of the hysteresis loop by various models. Therefore, there has been no widely accepted general model. More and more researchers have introduced other new piezoelectric ceramic feedforward control methods, such as using the Radial basis function(RBF) network [14] and neural network self-turning control [15].

MLP has been demonstrated to implement feedforward control of PEA. However, MLP needs too many training epochs, and the control displacement error has jumps and other shortcomings, as shown in Figure 1. Based on the control strategy realized by the MLP, this paper further enriches the piezoelectric ceramic operation dataset and adopts a new network structure(PEA-RNN) that conforms to data characteristics. Accurate displacement output of piezoelectric ceramics is achieved with fewer training epochs. The test results show that, under the operating frequency of 10 Hz, the precision of the controlled precision is improved from 8.96 μ m to 0.465 μ m after using PEA-RNN compensation. The influence of each network structure (mainly including MLP and residual connection) is verified further to prove the rationality of the network structure.



Figure 1. Control flaws of MLP. (**a**) Sinusoidal error distribution. (**b**) Dampened sinusoidal error distribution. (**c**) Amplified sinusoidal error distribution.

The rest of this paper is organized as follows: In Section 2, the performance of the experimental platform is evaluated, and the control strategy is preliminarily determined, then the related neural network structures are introduced. Next, the training and the application of PEA-RNN are illustrated, and the effects of MLP and residual connections are verified in Section 3. Finally, this paper is concluded in Section 4.

2. Performance Test and Related Structure

2.1. Piezoelectric Ceramic Hysteresis Loop

2.1.1. Experimental Platform

In this experiment, the digital piezoelectric actuator(E-709.CRG) [16] and the piezo linear precision positioner (P-621.1CD) [17] of Physik Instrumente (PI) are used, as shown in Figure 2. The signal amplifier is integrated into the actuator, and the capacitive sensor is integrated into the positioner. After the actuator receives the command, it sends the voltage to the positioner through the signal amplifier, reads back the current position of the piezoelectric ceramic from the capacitive sensor, and communicates with the computer through USB, as shown in Figure 3.



Figure 2. Experimental platform diagram.



Figure 3. Experimental platform communication diagram.

2.1.2. Primary and Secondary Hysteresis Loops

The hysteresis nonlinearity of piezoelectric ceramics is mainly manifested in input voltage and output displacement curves. The voltage-rising curve does not coincide with the voltage-dropping curve. Many factors affect this characteristic, not only related to the current input voltage, but also related to the input history and the input frequency [18].

The test data were obtained by taking the step signal, the dampened sinusoidal signal, and the amplified sinusoidal signal with a working frequency of 10 Hz as the input. The primary hysteresis loop refers to the largest hysteresis loop formed by the positioner during its entire travel. Furthermore, the secondary hysteresis loop refers to the smaller hysteresis loop.

As shown in Figure 4, the primary hysteresis loop was linearly fitted, and the linear relationship between the output displacement x and the input voltage V was obtained as



$$x = 0.978V + 0.294 \tag{1}$$

Figure 4. Primary hysteresis loop linear analysis result graph.

The maximum deviation of the voltage-rising curve (forward error) is 5.12 μ m; the maximum deviation of the voltage-dropping curve (backward error) is 8.96 μ m.

The test results of the secondary hysteresis loop are shown in Figures 5 and 6, and the maximum errors in the open loop are all about 6 μ m.



Figure 5. Secondary hysteresis loop linear analysis result graph (dampened sinusoidal input signal). (a) Dampened sinusoidal output. (b) Dampened sinusoidal curve comparision. (c) Dampened sinusoidal error distribution.



Figure 6. Secondary hysteresis loop linear analysis result graph (amplified sinusoidal input signal). (a) Amplified sinusoidal output. (b) Amplified sinusoidal curve comparision. (c) Amplified sinusoidal error distribution.

2.1.3. Hysteresis Loop Model

It can be found from the primary and the secondary hysteresis loops that the current position of the positioner has a great relationship with the current output voltage, the last position, and the last input voltage.

The state Φ_t of the piezoelectric ceramic at time t is uniquely represented by the input voltage and output displacement at the current time.

$$\Phi_{\rm t} = \varphi({\rm V}_{\rm t}, x_{\rm t}) \tag{2}$$

where V_t is the input voltage at time t, and x_t is the displacement at time t.

d

From the diagrams of the primary and the secondary hysteresis loops, it is not difficult to see the relationship between the current state and the historical state, and there is an obvious time series relationship between Φ_1, \ldots, Φ_t , which can be represented by a conditional probability model.

$$P(\Phi_1, \cdots, \Phi_t) = \prod_{T=1}^t P(\Phi_T \mid \Phi_1, \cdots, \Phi_{T-1}) \text{ where } P(\Phi_1 \mid \Phi_0) = P(\Phi_1)$$
(3)

As the operating time of PEA increases, more and more piezoelectric ceramic states need to be recorded and calculated, which will directly lead to an increasing control time and a continuous decrease in operating frequency. Therefore, consider introducing the Markov assumption, suppose that in the real situation, the rather long sequence $\Phi_{t-1}, \ldots, \Phi_1$, may not be necessary for the current state Φ_t . Therefore, it is only necessary to satisfy a certain length τ ; using the observation sequence $\Phi_{t-1}, \ldots, \Phi_{t-\tau}$, the current state can be predicted. According to the Markov assumption, (3) can be modified as

$$P(\Phi_1, \cdots, \Phi_t) = \prod_{T=1}^t P(\Phi_T \mid \Phi_{T-\tau}, \cdots, \Phi_{T-1})$$
(4)

To meet the requirements of high response frequency, a simple first-order Markov model is used, that is, $\tau = 1$. (4) can be simplified as

$$P(\Phi_1, \cdots, \Phi_t) = \prod_{T=1}^t P(\Phi_T \mid \Phi_{T-1}) \text{ where } P(\Phi_1 \mid \Phi_0) = P(\Phi_1)$$
 (5)

Bringing (2) into the equation

$$x_{t} = f(V_{t}, x_{t-1}, V_{t-1})$$
(6)

 V_{t-1} is the input voltage at time t - 1, and x_{t-1} is the displacement at time t - 1. After transformation, (7) is obtained:

$$V_{t} = f'(x_{t}, x_{t-1}, V_{t-1})$$
(7)

Finally, modifying the parameters:

$$V_{\rm R} = f'(x_{\rm T}, x_{\rm t-1}, V_{\rm t-1})$$
(8)

For the piezoelectric ceramic positioner, from the given target position x_T , the last time input voltage V_{t-1} and the last time displacement x_{t-1} , the voltage applied to the positioner V_R is calculated by the function f'.

Figure 7 is the principle structure diagram of the whole system. The task of this paper is to design a feedforward control model of a recurrent neural network, perform compensation calculation on the preset target position, and output the result through the controller to apply it to piezoelectric ceramics to realize micro and precise displacement.



Figure 7. Schematic diagram of feedforward control of PEA-RNN.

2.2. Network Structure

2.2.1. Recurrent Neural Network (RNN) and Gated Recurrent Unit (GRU)

In the model, considering the state of the last moment as a hidden variable for processing, it is necessary to add a neural network structure that can describe the hidden variable, which is the RNN [19].

As shown in Figure 8, for a single recurrent neuron, the input x_t , at time t, needs to be calculated in three steps to obtain the output o_t (to simplify the description, the input layer bias and the output layer bias are ignored). Firstly, the input layer weight w_{xh} is multiplied; secondly, the hidden layer output h_{t-1} , at time t – 1, is multiplied by the weight w_{hh} , and

then added to the first step result to obtain h_t ; thirdly, h_t is multiplied by the output layer weight w_{hg} , activated by the activation function, and the final output is o_t .



Figure 8. Recurrent Neural Unit Computational Flowchart.

The first step is usually expressed together with the second step:

$$\mathbf{h}_{t} = \mathbf{w}_{xh}\mathbf{x}_{t} + \mathbf{w}_{hh}\mathbf{h}_{t-1} \tag{9}$$

The third step can be expressed as:

$$\mathbf{o}_{t} = \sigma \Big(\mathbf{w}_{hq} \mathbf{h}_{t} \Big) \tag{10}$$

The hidden layer state h is continuously transmitted and used "recurrently", which coincides with the time sequence characteristic of the piezoelectric ceramic operating data.

The structure of the basic recurrent neural unit is simple, but for long sequence problems, the training likely fails due to gradient disappearance or explosion. GRU is a recurrent neural unit with better performance optimized for the shortcomings of the basic one [19].

The calculation flow of the GRU is shown in Figure 9. GRU mainly includes the reset gate and the update gate, which are vectors in the interval (0,1). The opening of the reset gate controls how well the current input combines with the hidden state, while the opening of the update gate controls the retention of the hidden state [20].

For a given time step t, assuming the input is a mini-batch $X_t \in \mathbb{R}^{n \times d}$ (number of samples: n, number of inputs: d), the hidden state of the previous time step is $H_{t-1} \in \mathbb{R}^{n \times h}$ (number of hidden units: h). Then the reset gate $R_t \in \mathbb{R}^{n \times h}$ and the update gate $Z_t \in \mathbb{R}^{n \times h}$ are calculated as follows:

$$\mathbf{R}_{\mathbf{t}} = \sigma(\mathbf{X}_{\mathbf{t}}\mathbf{W}_{\mathbf{x}\mathbf{r}} + \mathbf{H}_{\mathbf{t}-1}\mathbf{W}_{\mathbf{h}\mathbf{r}} + \mathbf{b}_{\mathbf{r}})$$
(11)

$$\mathbf{Z}_{\mathbf{t}} = \sigma(\mathbf{X}_{\mathbf{t}}\mathbf{W}_{\mathbf{x}\mathbf{z}} + \mathbf{H}_{\mathbf{t}-1}\mathbf{W}_{\mathbf{h}\mathbf{z}} + \mathbf{b}_{\mathbf{z}})$$
(12)

where σ represents the sigmoid function, which is used to convert the input value to the interval (0,1), the expression is

$$\operatorname{sigmoid}(x) = \frac{1}{1 + e^{-x}} \tag{13}$$



Figure 9. GRU computational flow chart.

Next, the reset gate R_t is integrated with the regular hidden state update mechanism to obtain candidate hidden states $\widetilde{H}_t \in \mathbb{R}^{n \times h}$ at time step t

$$\mathbf{H}_{t} = \tanh(\mathbf{X}_{t}\mathbf{W}_{xh} + (\mathbf{R}_{t} \odot \mathbf{H}_{t-1})\mathbf{W}_{hh} + \mathbf{b}_{h})$$
(14)

where the symbol \odot is the Hadamard product operator. The role of the tanh function is to convert the input value to the interval (-1,1); the specific expression is

$$\tanh(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}} \tag{15}$$

The result of the calculation is the candidate because the operation of the update gate still needs to be combined. The element multiplication of \mathbf{R}_t and \mathbf{H}_{t-1} in (14) can reduce the influence of past states. Whenever the terms in the reset gate \mathbf{R}_t approach 1, restore a normal recurrent neural network. For all terms close to 0 in \mathbf{R}_t , the candidate hidden state is the result of an MLP with \mathbf{X}_t as input. Therefore, any pre-existing hidden state is reset to default. Finally, the effect of the update gate \mathbf{Z}_t needs to be combined, which determines to what extent the new hidden state \mathbf{H}_t is the old state \mathbf{H}_{t-1} , and the use of the new candidate state $\widetilde{\mathbf{H}}_t$ quantity. The update gate \mathbf{Z}_t only needs an element-wise convex combination between \mathbf{H}_{t-1} and $\widetilde{\mathbf{H}}_t$ to achieve this. This provides the final update formula for the gated recurrent unit:

$$\mathbf{H}_{t} = \mathbf{Z}_{t} \odot \mathbf{H}_{t-1} + (1 - \mathbf{Z}_{t}) \tag{16}$$

Whenever the update gate Z_t approaches 1, only the old state is kept. At this point, the information from X_t is essentially ignored, effectively skipping time step t in the dependency chain. Conversely, as Z_t approaches 0, the new hidden state H_t approaches the candidate hidden state \widetilde{H}_t . These designs can help deal with the vanishing gradient problem in recurrent neural networks and better capture the dependencies of sequences with long time-step distances. For example, if the update gate is close to 1 for all time steps of the entire subsequence, then regardless of the length of the sequence, the old hidden state at the beginning time step will be easily preserved and passed on to the end of the sequence.

In summary, GRU has the following two salient features: [19]

- 1. Reset gate helps capture short-term dependencies in the sequence;
- 2. Update gate helps capture long-term dependencies in the sequence.

2.2.2. Multilayer Perceptron (MLP)

The structure of MLP is shown in Figure 10. This structure can improve the performance of the network and realize dimensional transformation.



Figure 10. MLP structure diagram.

2.2.3. Residual Connection

Generally, increasing the number of layers can improve the neural network performance, but it will correspondingly increase the computing time and the resources occupied by the calculation. At the same time, the increase in the number of layers will also lead to network degradation—that is, the weight matrix in the network will no longer be sensitive to changes in the input in some dimensions, and the same output is obtained regardless of the input. To optimize the MLP, the number of layers should be compressed as much as possible without loss of performance or very little performance, so the residual connection is introduced into MLP in the experiment.

Focus on the network portion: As shown in Figure 11, suppose the original input is x, and the ideal map you want to train is f(x) (as input to the activation function above). The part in the dashed box on the left needs to fit the mapping f(x) directly, while the part in the dashed box on the right needs to fit the residual mapping f(x)-x. Residual maps tend to be easier to optimize in reality. The right picture is the basic structure of ResNet [21]—the residual block. Using the residual connection can break the symmetry of the neural network, making the input propagate forward faster between layers, and more conducive to constructing deep neural networks.

3. Training and Application of PEA-RNN

3.1. Network Training

The training process is shown in Figure 12. Firstly, initialize the weight value parameters, scramble the dataset randomly, and divide it into a training dataset and a verification dataset according to the ratio of 8:2. Then, take out the training data in groups, take the current actual displacement, the last time voltage and the last time displacement as the input of the neural network, set the initial state to all zeros, and calculate the voltage prediction layer by layer through the forward propagation algorithm. After getting the prediction, compare it with the current actual voltage and calculate the loss value (mean square error, MSE). The neural network weight parameters are updated according to the optimizer in the backpropagation algorithm. Whenever the loss value on the training dataset decreases, the network is verified using the verification dataset. If the loss value is



the minimum loss value on the verification dataset at this time, the current neural network parameters are saved.

Figure 11. Residual connection structure diagram.

In the experiment, the loss function used is the MSE function, and the specific expression is:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$
(17)

where *n* represents the number of samples.

The optimizer uses the most commonly Adam [22], which can adaptively adjust the learning rate α so that the training can converge faster. The parameter update formula is:

$$w_t \leftarrow w_{t-1} - \alpha_t \cdot m_t / (\sqrt{v_t} + \hat{\epsilon}) \tag{18}$$

In the formula, w_t is the weight parameter to be updated; w_{t-1} is the weight parameter at the last moment; α_t is the learning rate, and m_t is the first-order moment estimation value of the gradient, which can adaptively adjust the speed of the learning rate change; v_t is the second-order moment estimate of the gradient, which can prevent the parameter from falling into a local minimum; $\hat{\epsilon} = 10^{-8}$, which is used to prevent the divisor from being 0.



Figure 12. Neural network training process flow chart.

The total dataset comprises the piezoelectric ceramic operating data, with 16,020 sets. During the experiment, set the Batch Size to 32 and the learning rate to 0.01. After 5000 epochs of training, a model that meets the accuracy requirements is obtained. The error decrease on the validation dataset is shown in Figure 13.



Figure 13. RNN Prediction Error Curve.

3.2. Network Application

3.2.1. The Overall Structure of PEA-RNN

As shown in Figure 14 and Table 1, the designed deep neural network includes one recurrent layer (hidden state dimension is 128) and seven linear layers. The ReLU function is used as the activation function between the linear layers, and the second layer of the linear layer connects with the fifth layer by Residual connection. The input layer contains

three dimensions (current actual displacement, the last time voltage, and the last time displacement), and the output layer is one-dimensional (input voltage prediction).



Figure 14. PEA-RNN Structure Diagram.

Table 1. RNN Specific Structure Parameters.

Layer Name	Input Dimension	Output Dimension	Bias	Hidden State Dimension
GRU layer	3	3	True	128
Linear 1	3	16	True	/
Linear 2	16	32	True	/
Linear 3	32	64	True	/
Linear 4	64	128	True	/
Linear 5	128	32	True	/
Linear 6	32	4	True	/
Linear 7	4	1	True	/

According to the design, for the ideal input voltage V_k required to move to the target displacement x_k at time k, the operation process is as follows:

Step 1: The target displacement x_k , the output displacement x_{k-1} and the input voltage V_{k-1} at time k - 1 directly form the input matrix $\mathbf{X}^{(k)}(\mathbf{X}^{(k)} \in \mathbb{R}^{1 \times 3})$ of the neural network at time k, and set the initial hidden state matrix $\mathbf{H}_0^{(k)}(\mathbf{H}_0^{(k)} \in \mathbb{R}^{1 \times 128})$ to zero:

$$\mathbf{X}^{(k)} = \begin{bmatrix} x_{k-1} & V_{k-1} & x_k \end{bmatrix}$$
(19)

$$\mathbf{H}_0{}^{\mathbf{k}} = \mathbf{0} \tag{20}$$

Step 2: GRU processes the input matrix sequentially and then outputs the corresponding result and the hidden state matrix containing trend information.

$$\mathbf{H}_{1}^{(k)} = \mathbf{Z}_{0}^{(k)} \odot \mathbf{H}_{0}^{(k)} + \left(\mathbf{1} - \mathbf{Z}_{0}^{(k)}\right) \odot \widetilde{\mathbf{H}}_{0}^{(k)}$$
(21)

$$\mathbf{H}_{2}^{(k)} = \mathbf{Z}_{1}^{(k)} \odot \mathbf{H}_{1}^{(k)} + \left(1 - \mathbf{Z}_{1}^{(k)}\right) \odot \widetilde{\mathbf{H}}_{1}^{(k)}$$
(22)

$$\mathbf{H}_{3}^{(k)} = \mathbf{Z}_{2}^{(k)} \odot \mathbf{H}_{2}^{(k)} + \left(\mathbf{1} - \mathbf{Z}_{2}^{(k)}\right) \odot \widetilde{\mathbf{H}}_{2}^{(k)}$$
(23)

Step 3: The output result $\mathbf{H}_{3}^{(k)}$ of GRU reduces the dimension from 128 to 3 through a linear layer, and obtains the operation result of the recurrent layer $\mathbf{O}_{\mathbf{RNN}}^{(k)}(\mathbf{O}_{\mathbf{RNN}}^{(k)} \in \mathbb{R}^{1 \times 3})$.

Step 4: The operation result of the recurrent layer $O_{RNN}^{(k)}$ continues to be used as the input of the MLP $X_{MLP}^{(k)}$. After the calculation of the MLP, the dimension is reduced from 3 to 1, This result is the ideal input voltage V_k at time k.

$$\mathbf{X}_{\mathbf{MLP}}^{(k)} = \mathbf{O}_{\mathbf{RNN}}^{(k)} \tag{24}$$

$$V_{k} = \mathbf{F} \left(\mathbf{X}_{\mathbf{MLP}}^{(k)} \right) \tag{25}$$

where $\mathbf{F}(\cdot)$ represents the complex nonlinear computation constructed by the MLP.

After the calculation of the recurrent layer and the MLP, the ideal input voltage V_k of the piezoelectric driver is finally obtained.

$$V_{k} = f'(x_{k}, x_{k-1}, V_{k-1})$$
(26)

3.2.2. Experimental Test

According to Figure 15, the validity of the model was verified on the experimental platform (Figure 2). Set the target position on the PC to change with the linear motion trajectory, sinusoidal motion trajectory, amplified sinusoidal motion trajectory, and damped sinusoidal motion trajectory with a working frequency of 10 Hz. After PEA-RNN feed-forward compensation, the prediction is sent to the controller to generate input voltage. Finally, test the actual position of the positioner.



Figure 15. Linear motion control renderings. (a) Output. (b) Error.

The experimental results are shown in Figures 16–18. When set as the step signal, the target position has a good linear relationship with the actual position, and the maximum error value is 0.210 μ m. When the target is set to a sinusoidal signal, the maximum

error is 0.396 μ m. When the target is set to a dampened sinusoidal signal, the maximum error is 0.230 μ m. The maximum error is 0.465 μ m when the target is set to an amplified sinusoidal signal.



Figure 16. Sinusoidal motion control renderings. (a) Curve comparision. (b) Error distribution.



Figure 17. Dampened Sinusoidal motion control renderings. (a) Curve comparision. (b) Error distribution.



Figure 18. Amplified Sinusoidal motion control renderings. (a) Curve comparision. (b) Error distribution.

3.3. Ablation Experiment

To more clearly demonstrate the contribution of each structure in the PEA-RNN to the model, two ablation experiments were performed to evaluate the effect of the MLP and the residual connection, respectively.

3.3.1. The Impact of MLP

This experiment aims to verify the effect of MLP on the control accuracy of PEA-RNN. The experimental method adopted is to directly delete the MLP in PEA-RNN, so the new network has only one GRU layer. After the network is fully trained with the same training strategy as the original, the output error of the piezoelectric ceramic is tested under the condition that the input is a sinusoidal signal. The results are shown in Figure 19. The maximum output error is 0.599 μ m, an increase of 51.3% compared to the original network.

After removing the MLP, the new network is a normal RNN, including a GRU layer. The linear fitting ability of the network is reduced, the error distribution is also relatively discrete, and the error at the starting point is significant. The MLP at the tail can enhance the nonlinear expression ability of PEA-RNN so that the model can describe the hysteresis nonlinearity of piezoelectric ceramics in appropriate detail.

3.3.2. The Impact of Residual Connection

This section evaluates its impact on model performance by removing the residual connection in PEA-RNN alone, and the experimental test results are shown in Figure 20. At the beginning of the piezoelectric ceramic displacement output, the error is too large, indicating that the neural network has not completely fitted the original curve after the same epochs of training, and the training error has not dropped to a relative minimum. A part of the error value is around $\pm 1 \mu m$, which is far worse than the original network control accuracy.



Figure 19. PEA-RNN without MLP control errors. (a) Curve comparision. (b) Error distribution.



Figure 20. PEA-RNN without Residual Connection control errors. (a) Curve comparision. (b) Error distribution.

The role of the residual connection is to speed up the neural network training process. At the same time, it is beneficial to build a deeper network, which can also increase the nonlinear expression ability of the network.

4. Conclusions

Based on the PyTorch framework, this paper builds a deep neural network model named PEA-RNN, with one recurrent layer and seven linear layers. The ReLU function is used as the activation function between the linear layers. The residual connection is used between the second and fifth layers of the linear layers. The model's design principle and training process is given, and the model obtained by training is applied to the feedforward compensation of the positioner.

The test results show that the maximum displacement error is reduced from 8.96 μ m to 0.465 μ m under the control of PEA-RNN with the input of the 10 Hz operating frequency. Through ablation experiments to verify the role of each structure in the PEA-RNN, MLP can effectively enhance the nonlinear expression ability of the model, and the residual connection can not only accelerate the training process but also enhance the nonlinear expression of the model. These results show that PEA-RNN constructed in this paper can accurately describe the dynamic hysteresis nonlinearity of piezoelectric ceramics, realize the real-time intelligent beam line modulation system, and create the possibility for online updating of the model.

Author Contributions: Conceptualization, Y.X. and L.Z. (Lifang Zheng); methodology, Y.X. and W.J.; software, Y.X.; validation, Y.X. and L.Z. (Lifang Zheng); formal analysis, Y.X.; investigation, Y.X.; resources, L.Z. (Limin Zhang); data curation, Y.X. and L.Z. (Lifang Zheng); writing—original draft preparation, Y.X. and L.Z. (Lifang Zheng); writing—review and editing, Y.X.; supervision, L.Z. (Lifang Zheng) and Y.Z.; project administration, L.Z. (Lifang Zheng), L.Z. (Limin Zhang) and Y.Z.; funding acquisition, Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China Youth Fund grant 12105351.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

- MLP Multilayer perceptron
- PEA Piezoelectric actuator
- RBF Radial basis function
- GRU Gated recurrent unit
- SSRF Shanghai Synchrotron Radiation Facility
- PI Physik Instrumente
- MSE Mean square error
- ReLU Linear rectification function

References

- 1. Shi, Y.; Gao, M.; Jia, W.; Yin, G.; Gao, X.; Du, Y.; Zheng, L. Intelligent commissioning system based on EPICS and differential evolution algorithm for synchrotron radiation beamline. *Nucl. Tech.* **2020**, *43*, 3–9.
- 2. Jia, W.; Ma, S.; Zheng, L. The control system for water-cooled DCMS in SSRF. Nucl. Sci. Tech. 2015, 26, 26–31.
- Zhou, X. Research of Control in the Micro-Positioning System Actuated by Piezo-Ceramic. Ph.D. Thesis, Zhejiang University, Zhejiang, China, 2004.
- 4. Gu, G.Y.; Zhu, L.M.; Su, C.Y.; Ding, H.; Fatikow, S. Modeling and control of piezo-actuated nanopositioning stages: A survey. *IEEE Trans. Autom. Sci. Eng.* **2014**, *13*, 313–332. [CrossRef]
- 5. Landis, C.M. Non-linear constitutive modeling of ferroelectrics. Curr. Opin. Solid State Mater. Sci. 2004, 8, 59–69. [CrossRef]
- Gan, J.; Zhang, X. A review of nonlinear hysteresis modeling and control of piezoelectric actuators. *AIP Adv.* 2019, 9, 040702. [CrossRef]
- 7. Preisach, F. Über die magnetische Nachwirkung. Z. für Phys. 1935, 94, 277–302. [CrossRef]
- 8. Maas, J.; Schulte, T.; Frohleke, N. Model-based control for ultrasonic motors. *IEEE/ASME Trans. Mechatron.* **2000**, *5*, 165–180. [CrossRef]
- 9. Lining, S.; Changhai, R.; Weibin, R.; Liguo, C.; Minxiu, K. Tracking control of piezoelectric actuator based on a new mathematical model. *J. Micromechan. Microeng.* **2004**, *14*, 1439. [CrossRef]
- 10. Bouc, R. A mathematical model for hysteresis. Acta Acust. United Acust. 1971, 24, 16–25.

- 11. Wen, Y.K. Method for random vibration of hysteretic systems. J. Eng. Mech. Div. 1976, 102, 249–263. [CrossRef]
- Delibas, B.; Arockiarajan, A.; Seemann, W. Rate dependent properties of perovskite type tetragonal piezoelectric materials using micromechanical model. Int. J. Solids Struct. 2006, 43, 697–712. [CrossRef]
- 13. Kamlah, M.; Tsakmakis, C. Phenomenological modeling of the non-linear electro-mechanical coupling in ferroelectrics. *Int. J. Solids Struct.* **1999**, *36*, 669–695. [CrossRef]
- Liu, D.; Fang, Y.; Wang, H.; Dong, X. Adaptive novel MSGA-RBF neurocontrol for piezo-ceramic actuator suffering rate-dependent hysteresis. Sens. Actuators A Phys. 2019, 297, 111553. [CrossRef]
- 15. Li, W.; Zhang, C.; Gao, W.; Zhou, M. Neural network self-tuning control for a piezoelectric actuator. *Sensors* **2020**, *20*, 3342. [CrossRef] [PubMed]
- 16. Instrumente, P. E-709.CRG. Available online: https://www.physikinstrumente.store/eu/E-709.CRG/ (accessed on 14 July 2022).
- 17. Instrumente, P. P-621.1CD. Available online: https://www.physikinstrumente.store/eu/p-621.1cd/ (accessed on 14 July 2022).
- Xiong, Y.; Jia, W.; Zhang, L.; Zheng, L. Research on Feedforward Compensation of Piezoelectric Ceramics Based on Deep Neural Network(DNN). *Piezoelectrics Acoustooptics* 2022, 44, 35.
- 19. Zhang, A.; Li, M.; Lipton, Z.C.; Simola, A.J. Dive Into Deep Learning; Posts & Telecom Press Co., Ltd.: Beijing, China, 2019.
- Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* 2014, arXiv:1406.1078.
- He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
- 22. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. arXiv 2014, arXiv:1412.6980.