# Comparison of Deep Learning Architectures for Nonlinear System Identification of a Hysteretic Piezoelectric Precise Positioner ⋆

**Victor Henrique A. Ribeiro** * **Pedro H. L. S. P. Domingues** **
**Gilberto Reynoso-Meza** * **Micky Rakotondrabe** ***
**Leandro dos Santos Coelho** * **Helon V. H. Ayala** **

* *Industrial and Systems Engineering Graduate Program, Pontifícia Universidade Católica do Paraná, Curitiba, Brazil (e-mail: victor.henrique@pucpr.edu.br, g.reynosomeza@pucpr.br, leandro.coelho@pucpr.br).*
** *Department of Mechanical Engineering, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, Brazil (e-mail: phd.engmec@gmail.com, helon@puc-rio.br).*
*** *Laboratoire Génie de Production, National School of Engineering in Tarbes (ENIT-INPT), University of Toulouse, Tarbes, France (e-mail: mrakoton@enit.fr).*

**Abstract:** The characterization of hysteretic components poses a difficult nonlinear system identification problem. Several studies have addressed this by employing artificial neural networks, where deep learning (DL) has recently gained attention in system identification tasks. However, there is a lack of studies comparing different deep neural network (DNN) architectures. Therefore, this work proposes the comparison of three DNN architectures, including feedforward neural networks (FFNN), long short term memory (LSTM), and convolutional neural networks (CNN), for the characterization of a piezoelectric positioning system (positioner) typified by hysteresis. Moreover, Bayesian optimization is employed for hyperparameter tuning in all DNN architectures. Results show that all DL architectures achieved desirable values for the coefficient of determination ($R^2$) and root mean squared error (RMSE). However, LSTM obtains the best overall results, outperforming both the FFNN and CNN, being a more appropriate black-box architecture for identifying frequency-dependent hysteresis loop shapes.

*Keywords:* System identification, Hyperparameter tuning, Bayesian optimization, Deep Learning, Convolutional neural network, Long short term memory

## 1. INTRODUCTION

System identification is the science field interested in data driven methods for the mathematical representation of systems dynamics (Ljung, 1999), being used in a wide range of problems, such as biomechanical prostheses model extraction (Abdelhady et al., 2017), fault diagnosis (Abid et al., 2019), and industrial robots calibration (Zhao et al., 2019).

In essence, system identification may be performed by two approaches when using measured data: i) black-box, which extracts the model from input excitation and output response of the system; and ii) grey-box, which first generates a partial or complete model of the system considering an a priori knowledge of the process and then uses data to adjust its parameters. The black-box modeling stands out when compared to the grey-box approach, for demanding less information from the system and for the ease of implementation, since many processes become difficult to model considering first principles, making the grey-box identification a laborious task. (Billings, 2013; Worden et al., 2007; Tangirala, 2018).

Piezoelectric materials are extensively used to develop actuators for precise positioning systems (precise positioners) thanks to their high frequency rates, high resolution, portability, and ease of integration (Rakotondrabe, 2014). However, when these piezoelectric positioners operate at micro or nano-scale applications, the direct sensing is hindered or made impossible by the measured signals' uncertainties, which promotes the use of soft sensing through artificial feedback loops (Clévy et al., 2011) or alternatively from piezoelectric self-sensing approaches (Rakotondrabe, 2013). Meanwhile, these approaches require appropriate characterization of the piezoelectric positioners. The literature provides some strategies for modeling these systems, including Duhem, Bouc-Wen, Preisach, and Prandtl-Ishlinskii models (Gu et al., 2014; Rakotondrabe, 2017) to represent their hysteresis nonlinearity, linear approxima-

⋆ This study was financed in part by the *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior* (CAPES), the *Conselho Nacional de Desenvolvimento Científico e Tecnológico* (CNPq), and the *Fundação Araucária* (FAPPR) - Brazil - Finance Codes: 159063/2017-0-PROSUC, 310079/2019-5-PQ2, 437105/2018-0-Univ, 405580/2018-5-Univ, and PRONEX-042/2018

tion, and logarithmic function for their creep nonlinearity (Rakotondrabe, 2012). While an ad-hoc tuning of these models might appear difficult, the control design considering the model artificial feedbacks is also a challenging task (Rakotondrabe, 2014). Therefore, by avoiding difficulties in i) obtaining general solutions in differential-equation based methods (i.e. Duhem and Bouc-Wen models); ii) implementing Preisach model by the presence of double integrals; or iii) representing hysteresis effect in asymmetric behavior, in case of Prandtl-Ishlinskii model (Gu et al., 2014), black-box system identification becomes attractive.

Two degrees of freedom (dof) piezoelectric actuators (PAs) were modeled through Duhem hysteresis model with parameters identification made by artificial neural network (ANN) in Wang et al. (Wang and Chen, 2017). Ayala et al. (Ayala et al., 2018) used a shallow ANN at high frequency rates in a nonlinear autoregressive exogenous input (NARX) model for the same application. A nano-scale PA was identified through a recurrent neural network (RNN) in order to implement a model predictive control with real-time trajectory tracking in Xie et al. (Xie and Ren, 2019). Still, a variant of RNN was used to set up a frequency-dependent hysteresis model with memory for a nano-scale PA in Wu et al. (Wu et al., 2020).

The use of ANNs for system identification has been devised for some decades (Kumpati et al., 1990), being initially performed by feedforward neural network (FFNN), such as radial basis functions (RBFs) (Ayala and dos Santos Coelho, 2016) and multilayer perceptrons (Subudhi and Jena, 2011). However, the training signals time-dependency fosters the use of RNNs for this application, since the complexity conferred by recurrences allows the representation of time dimension directly, without the need for delays (Schrauwen et al., 2007). Still, deep learning (DL) has been attracting the scientific community attention due to the results obtained in the image processing area (LeCun et al., 2015) and already suggests strong links to the system identification execution (Ljung et al., 2020). De La Rosa and Li (De la Rosa et al., 2015) proposed a modified random algorithm with multiple hidden layers for nonlinear identification. The output weights were trained by a conventional random algorithm, while the hidden ones are defined by the input data and a modified restricted Boltzmann machine. Three nonlinear identification benchmarks were used to validate the technique. Mattos et al. (Mattos et al., 2017) proposed the application of a novel robust latent autoregressive model treatment, used to account uncertainty in the regression task in a deep and recurrent structure, known as recurrent Gaussian process, in order to identify nonlinear systems. The vibration response estimation of linear and nonlinear single dof and a full-scale 3-story multi dof systems were proposed in Wu et al. (Wu and Jahanshahi, 2019a), through deep convolutional neural network (CNN) and conventional FFNN. Six nonlinear identification benchmarks were used to show the model robustness against noisy data.

However, the literature still presents a lack of differente DNN architectures comparison for system identification, specially for piezoelectric micromanipulators. Therefore, this work presents the following contributions: i) the employment of DL to characterize a precise piezoelectric positioner typified by hysteresis; ii) the comparison of three different DL architectures, i.e. FFNN, long short term memory (LSTM), and CNN; and iii) the Bayesian Optimization application for hyperparameter tuning. Results on multiple different frequencies show the advantages of LSTM over the other methods. Specifically, the LSTM architecture was able to adequately model frequency dependent hysteresis loop shapes, using real-world measured data acquired from an actual piezoelectric positioner.

The remainder of this paper is structured as follows. First, Section 2 details the data collection procedure, the deep neural network (DNN) architectures, and the Bayesian optimization for hyperparameter tuning. Next, Section 3 describes the experimental procedure and discusses the results for optimization and model comparison. Finally, the paper is concluded with final remarks and future work.

## 2. MATERIALS AND METHODS

This section introduces details for reproducing the characterization of hysteretic piezoelectric positioner using the DL architectures reported in this work.

### *2.1 Data Collection for Piezoelectric Micromanipulator*

The procedure for data collection of the piezoelectric positioner employs an open loop experiment (Soares et al., 2020). The positioner is composed of a piezoelectric actuator with cantilever structure having rectangular section. When applying a voltage to the actuator, it bends. This bending is exploitable as the positioning result (displacement).

The driving voltage signal (input) called $x(t)$ is generated using MATLAB computational environment and a high-voltage-amplifier, with signals ranging between $[-100V, 100V]$. The displacement (output) called $y(t)$ is measured using an optical displacement sensor, the Keyence LK2420. Finally, the dS1104 acquisition board, from dSPACE, serves as converter (DAC and ADC) with $20kHz$ sampling between the computer and the sensor/amplifier.

The voltage $x(t)$ for the training and validation dataset is created using a multi-sine signal

$$x(t) = \sum_{k=1}^{n_f} A \cos\left(2\pi f_k t + \phi_k\right) \tag{1}$$

where $f_k$ and $\phi_k$ are the frequency and phase of each sine component. In total, the signal is composed of $n_f = 500$ components with frequencies equally spaced between $[0Hz, 500Hz]$ and random phases $\phi_k$. The band has been chosen so that it includes resonant modes and typical working frequencies for precise positioning.

The test dataset is composed of twelve sinusoidal signals, with frequencies set to 0.1, 1, 10, 50, 100, 150, 200, 250, 300, 400, 450, and 500 Hz. The multisine dataset has been created so that the hysteretic behavior of the stage is adequately captured, when applied in the band required for the application.

### *2.2 Deep Learning Architectures*

DL has gained great attention from the industry and research community recently due to advances in computer

vision and, most recently, natural language processing (Goodfellow et al., 2016). Nevertheless, recent studies have employed such techniques to perform system identification (Ljung et al., 2020; Soares et al., 2020; Wu and Jahanshahi, 2019b). However, there are still few studies that compare DNN models for this task (Ljung et al., 2020). Therefore, this work compares three distinct DL architectures, FFNN, LSTM, and CNN.

*Feedforward Neural Network* FFNNs are the quintessential deep learning models, also known as multilayer perceptrons (Goodfellow et al., 2016). The term feedforward derives from the information flow of such networks, going from the input $\boldsymbol{x}$ to the intermediate layers, and finally to the output $y$. A network with $l$ layers, therefore, maps the input $\boldsymbol{x}$ to the output $y$ through a cascaded function $y = f(\boldsymbol{x}) = f^{(l)}(\dots f^{(2)}(f^{(1)}(\boldsymbol{x}))\dots)$. Therefore, FFNNs can be considered as NARX models (Ljung et al., 2020). Moreover, such an architecture has showed universal approximation capability (Hornik, 1991).

*Long Short Term Memory* LSTMs are gated RNNs (Hochreiter and Schmidhuber, 1997; Wang, 2017). Recurrent architectures are specialized in processing a sequence of input values $\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(n)}$, which can scale to much longer sequences than non-sequence neural networks (Goodfellow et al., 2016). Nevertheless, such models also present universal approximation capabilities (Schäfer and Zimmermann, 2006). Finally, the gate structure enables the dynamic control of the time scale to be used for integration during the prediction step. As such, LSTMs can be considered as a type of nonlinear state-space (NLSS) model (Ljung et al., 2020).

*Convolutional Neural Network* CNNs are a specialized type of ANN that employs the convolution operation (LeCun et al., 1989). That is, in at least one of the network layers, the convolution operation is performed instead of the usual matrix multiplication. It is mostly used to process data with grid-like topology, like images and time series (Goodfellow et al., 2016). Moreover, the universal approximation capability of CNNs has also been proven (Zhou, 2020). Nevertheless, recent research has detailed equivalences between convolutional networks and system identification models, such as Volterra series and block-oriented models (Wiener, Hammerstein, and Wiener-Hammerstein) (Andersson et al., 2019).

### *2.3 Bayesian Optimization for Hyperparameters Tuning*

Hyperparameter tuning is an important task when developing machine learning models. The careful adjustment of such hyperparameters has a huge impact on the final predictive performance. However, the manual tuning involves a trial and error approach. Therefore, automatic approaches have been mostly used, such as grid and random search. Most recently, researchers have shown that Bayesian optimization can outperform other search approaches for this task (Snoek et al., 2012).

The focus of Bayesian optimization is similar to other optimization approaches, which is to find a set of parameters in $\boldsymbol{H} \in \mathbb{R}^{\mathrm{D}}$ that minimizes (or maximizes) a function $g(\boldsymbol{h})$. However, such an algorithm constructs a probabilistic model for $g(\boldsymbol{h})$ and then exploits it to suggest the next position in $\boldsymbol{H}$ to evaluate the function. The probabilistic model is built using Gaussian processes, resulting in slightly more computation to perform each search, but enabling the use of less iterations to find an optimal solution. Therefore, this algorithm is suitable for expensive functions, such as hyperparameter tuning. The hyperparameters to be optimized in FFNN, LSTM, and CNN architectures are present in Table 1.

## 3. EXPERIMENTS AND RESULTS

This section describes all the experiments and discusses the results. Section 3.1 introduces the experimental procedure; Section 3.2 exposes the hyperparameter optimization results; and Section 3.3 compares the three DL architectures. Finally, all experiments and analyzes were done using Python programming language.

### *3.1 Experimental Procedure*

The experimental procedure is performed according to the following steps.

*Data Handling* During optimization and training, the development set is composed of more than 70,000 samples with multiple simultaneous frequencies. Such a set is randomly split for holdout validation. While the training step uses 90% of the data, the validation step uses only 10%.

Finally, the test set evaluates the optimized models. There are a total of 12 sets with different frequencies (0.1Hz, 1Hz, 10Hz, 50Hz, 100Hz, 150Hz, 200Hz, 250Hz, 300Hz, 400Hz, 450Hz, and 500Hz). Both the inputs ($\boldsymbol{x}$) and outputs ($\boldsymbol{y}$) are scaled between the range $[-1, 1]$, by dividing the outputs and inputs by the maximum displacement ($20\mu m$) and voltage ($100V$), respectively.

*Deep Learning Models* All deep learning models are given the regression matrix formed by input $x(t)$ and the output $y(t)$. The model order was set as 10 for both input and output, same value used in (Soares et al., 2020), obtained by trial and error. Therefore, FFNN receives a total of 20 inputs, while LSTM and CNN receive 10 time steps of 2 features. Table 1 details the adjustable hyperparameters for FFNN, LSTM, and CNN, being kept blank in Table 1 all hyperparameters not applied to the corresponding architeture. In both LSTM and CNN architectures, the specialized layers are used as the first ones, while the number of fully connected layers is the value that complete the maximum number of layers ('Total Layers'). Also, a global max pooling operation is added between the convolution and the fully connected layers. All DNN architectures use the rectified linear unit (ReLU) activation function (Nair and Hinton, 2010), 100 training epochs with a batch size of 128 samples, and the Adam optimizer (Kingma and Ba, 2014). Finally, the loss function is the mean squared error (MSE) for the prediction $\hat{\boldsymbol{y}}$ of the validation signal $\boldsymbol{y}$, computed as follows for the $n$ total samples.

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

Table 1. Hyperparameters for the FFNN, LSTM, and CNN.

| | Hyperparameters | Architecture | | |
|---|---|---|---|---|
| | | FFNN | LSTM | CNN |
| Adjustable | Total Layers | 3 – 5 | 3 – 5 | 3 – 5 |
| | Neurons per Layer | 25 – 100 | 25 – 100 | 25 – 100 |
| | Learning Rate ($10^{-4}$) | 1 – 10 | 1 – 10 | 1 – 10 |
| | LSTM Layers | | 1 – 2 | |
| | CNN Layers | | | 1 – 2 |
| | Kernel Size | | | 3 – 5 |

*Support Vector Machine* This experiment also employs a support vector machine (SVM) (Boser et al., 1992; Cortes and Vapnik, 1995) for comparison purposes. Hyperparameter tuning is also performed by optimizing the kernel function (linear, polynomial, RBF, or sigmoid), the regularization parameter ($0.1 \leq C \leq 100$), the polynomial order ($[2, \ldots, 6]$), and the kernel coefficient ($1e^{-4} \leq \sigma \leq 10$).

*Optimization* The Bayesian optimization algorithm adjusts the hyperparameters of each architecture for 100 iterations and $\alpha = 1e^{-3}$, being $\alpha$ the inverse of the regularization parameter. The objective function is the MSE computed for the validation dataset.

*Evaluation Metrics* This experiment uses two evaluation metrics to analyze the results, the coefficient of determination ($R^2$) and root mean square error (RMSE). The former indicates how accurate the model is by the proximity of the unity value, while RMSE measures the prediction error amplitude, being the best predictions those with RMSE close to zero. Both metrics are computed as follows

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \overline{y})^2} \tag{3}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}} \tag{4}$$

### *3.2 Optimization Results*

All models achieved similar optimization results, with RMSE values close to $0.07\mu m$. The final FFNN model is composed of 4 layers with 100 neurons each, which has been achieved in approximately 10 iterations. For LSTM and CNN only 3 total layers with 2 initial specialized layers are used, with the optimal models achieved in approximately 70 and 55 iterations, respectively. Additionally, the optimal CNN has a kernel size of 5. LSTM presents the lowest number of configured neurons per layer (66), followed by CNN (83). It is important to notice, however, that LSTM presents the highest number of trainable parameters ($58k$) given the more complex architecture of the LSTM layer. Such a model is followed by CNN ($42k$) and FFNN ($33k$). LSTM also demands higher computation during training ($860s$) than the two other models ($75s$), but demands similar computation during prediction ($4ms/step$). Additionally, the total optimization times for the FFNN, LSTM, and CNN were approximately $6k$, $72k$, and $12k$ seconds on a virtual machine with Intel(R) Xeon(R) E-2690 central processing unit (CPU), 120 GB random access memory (RAM), and CENT-OS Linux 7 operating system, respectively.

It is interesting to notice how the best DNN models are not composed of the maximum number of neurons and layers. Of course, this is closely related to the available data and fixed hyperparameters. Nevertheless, such results show that the characterization of the piezoelectric micromanipulator does not necessarily need a high number of layer, but rather a good evaluation and heuristic search method. Moreover, the fast optimization of the FFNN, in contrast to LSTM and CNN, indicates how simpler architectures are more easily optimized.

### *3.3 Comparison of Deep Learning Architectures*

Figure 1 plots the outputs of the three DL architectures in the 12 tested frequencies. The Figure 1 analysis reveals that for lower frequencies ($0.1Hz$ and $1Hz$) all models show difficulty in characterizing the hysteresis. Most specifically, the CNN displays an instability for the lower voltages ($0.1 - 10Hz$). However, the models tend to correctly follow the frequency-dependent hysteresis for higher frequencies. Most noticeably, signals with $250Hz$ and $400Hz$ present nonlinear characteristics that are harder to predict by all models.

Table 2 details the $R^2$ and RMSE scores of the tested models for the 12 frequencies, where $R^2$ values from previous literature are also shown, in the "Ref" column (Soares et al., 2020) and suggests that: i) LSTM is the best model when considering both metrics for achieving higher $R^2$ and lower RMSE, in terms of average; ii) despite the best score for each frequency be distributed among all DL architectures, the CNN presents best $R^2$ and RMSE scores for most frequencies, but worst results for the lowest and highest ones; iii) FFNN yield the same $R^2$ value than LSTM and is considered the second best architecture even presenting a higher RMSE result when compared with CNN, since FFNN has a more constant performance than CNN, which is denoted by the standard deviation of 0.27 and 0.51, respectively; and finally, iv) SVM is not capable of achieving similar results to any of the DNN architectures, presenting the lowest $R^2$ and highest RMSE scores.

Table 2. The $R^2$ and RMSE values for the compared models at each test frequency. Best results are marked in bold.

| Freq. | $R^2$ | | | | | RMSE | | | |
|---|---|---|---|---|---|---|---|---|---|
| (Hz) | Ref | FFNN | LSTM | CNN | SVM | FFNN | LSTM | CNN | SVM |
| 0.1 | 0.987 | **0.988** | 0.987 | 0.984 | 0.929 | **1.181** | 1.213 | 1.389 | 2.888 |
| 1 | 0.989 | 0.989 | **0.989** | 0.985 | 0.939 | 1.098 | **1.085** | 1.280 | 2.597 |
| 10 | 0.991 | **0.992** | 0.991 | 0.991 | 0.947 | **0.907** | 0.999 | 1.002 | 2.356 |
| 50 | 0.998 | 0.996 | 0.997 | **0.999** | 0.956 | 0.650 | 0.563 | **0.314** | 2.121 |
| 100 | 0.998 | 0.996 | 0.998 | **0.999** | 0.961 | 0.618 | 0.479 | **0.274** | 1.992 |
| 150 | 0.997 | 0.997 | 0.998 | **0.999** | 0.966 | 0.580 | 0.473 | **0.358** | 1.884 |
| 200 | 0.995 | 0.995 | 0.997 | **0.997** | 0.971 | 0.725 | 0.587 | **0.581** | 1.793 |
| 250 | 0.986 | **0.988** | 0.983 | 0.979 | 0.967 | **1.191** | 1.400 | 1.577 | 1.953 |
| 300 | 0.996 | 0.994 | 0.998 | **0.998** | 0.972 | 0.863 | 0.556 | **0.500** | 1.886 |
| 400 | 0.988 | 0.990 | 0.989 | **0.995** | 0.897 | 1.319 | 1.341 | **0.953** | 4.138 |
| 450 | **0.997** | 0.995 | 0.997 | 0.995 | 0.855 | 0.980 | **0.825** | 0.990 | 5.326 |
| 500 | 0.986 | 0.992 | **0.994** | 0.985 | 0.756 | 1.446 | **1.216** | 1.901 | 7.765 |
| Mean | 0.992 | **0.993** | **0.993** | 0.992 | 0.926 | 0.963 | **0.895** | 0.927 | 3.058 |

Results for model comparison indicate that LSTM and FFNN are the best DNN architectures for the given task. Interestingly, the previous work also employed FFNN (Soares et al., 2020), but applied grid search for hyperparameter selection. Therefore, the Bayesian optimization used in the present work performs as well as the grid search, according to the $R^2$ metric disposed in Table 2.

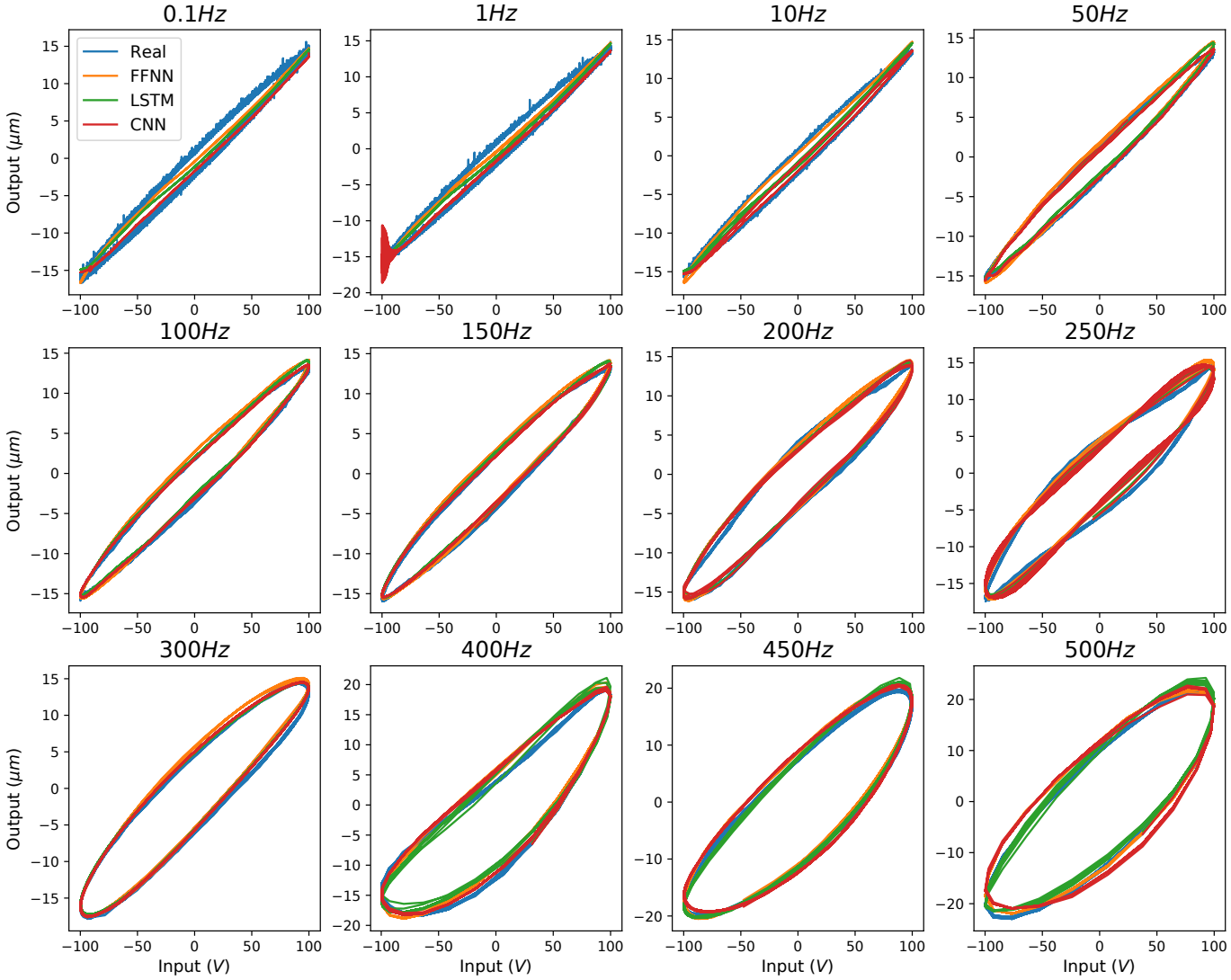


Figure 1. Position prediction for the three tested deep neural network (DNN) architectures.

This result sets the Bayesian search algorithm as a hyperparameter tuning option. Finally, LSTM achieves the best overall results, which might be mostly related to the recurrent architecture, enabling a better handling of sequential data.

## 4. CONCLUSIONS

This work employed three different DNN architectures, FFNN, LSTM, and CNN, for system identification of a hysteretic piezoelectric positioner. Additionally, Bayesian optimization performs hyperparameter tuning to build more accurate models. The models are also compared to SVM and a FFNN from recent literature. Results show that all the DL models outperform the SVM model. Moreover, LSTM presents the best overall results, followed by FFNN, in terms of $R^2$ and RMSE. The CNN architecture was the most variable in terms of $R^2$ and RMSE metrics, presenting instabilities for the lower voltages $(0.1-10Hz)$, following by the best performances in $(50-200, 300-400Hz)$ and a performance drop at higher frequencies.

As a conclusion, DL techniques show great potential as system identification techniques. However, there is still room for improvement. Solutions for the selection of lagged regressors can be included. Future works shall also focus on the architectures by i) combining the methods used to build stronger ensemble models through stacking ensemble techniques, for example; or ii) the use of more complex models due to the noisy character of the signal. Other interesting paths to follow refer to the methodology, such as evaluating the performance iii) not only at different frequencies, but varying the training/test dataset proportion; and iv) with computational cost, in a multiobjective approach.

## REFERENCES

Abdelhady, M. et al. (2017). System identification and control optimization of an active prosthetic knee in swing phase. In *American Control Conference*. Seattle, USA.

Abid, A. et al. (2019). Adaptive system identification and severity index-based fault diagnosis in motors. *IEEE/ASME Trans on Mechatronics*, 24(4), 1628–1639.

Andersson, C. et al. (2019). Deep convolutional networks in system identification. In *Conference on Decision and Control*, 3670–3676.

Ayala, H.V.H. and dos Santos Coelho, L. (2016). Cascaded evolutionary algorithm for nonlinear system identification based on correlation functions and radial basis functions neural networks. *Mechanical Systems and Signal Processing*, 68.

Ayala, H.V.H., Rakotondrabe, M., and dos Santos Coelho, L. (2018). Modeling of a 2-dof piezoelectric micromanipulator at high frequency rates through nonlinear black-box system identification. In *American Control Conference*, 4354–4359.

Billings, S.A. (2013). *Nonlinear system identification: NARMAX methods in the time, frequency, and spatio-temporal domains*. John Wiley & Sons.

Boser, B.E., Guyon, I.M., and Vapnik, V.N. (1992). A training algorithm for optimal margin classifiers. In *Workshop on Computational Learning Theory*, 144–152.

Clévy, C. et al. (2011). *Signal measurement and estimation techniques issues in the micro/nano world*. Springer.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3), 273–297.

De la Rosa, E. et al. (2015). Nonlinear system identification using deep learning and randomized algorithms. In *IEEE International Conference on Information and Automation*, 274–279.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.

Gu, G.Y. et al. (2014). Modeling and control of piezo-actuated nanopositioning stages: A survey. *IEEE Transactions on Automation Science and Engineering*, 13(1), 313–332.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2).

Kingma, D.P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kumpati, S.N., Kannan, P., et al. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, 1(1), 4–27.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.

LeCun, Y. et al. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4).

Ljung, L. (1999). *System identification: theory for the user*. PTR Prentice Hall, Upper Saddle River, NJ.

Ljung, L. et al. (2020). Deep learning and system identification. *IFAC-PapersOnLine*, In press.

Mattos, C.L.C. et al. (2017). Deep recurrent gaussian processes for outlier-robust system identification. *Journal of Process Control*, 60, 82–94.

Nair, V. and Hinton, G.E. (2010). Rectified linear units improve restricted boltzmann machines. In *International Conference on Machine Learning*. Haifa, Israel.

Rakotondrabe, M. (2017). Multivariable classical prandtl-ishlinskii hysteresis modeling and compensation and sensorless control of a nonlinear 2-dof piezoactuator. *Nonlinear Dynamics*, doi. 10.1007/s11071-017-3466-5.

Rakotondrabe, M. (2012). Modeling and compensation of multivariable creep in multi-dof piezoelectric actuators. In *IEEE International Conference on Robotics and Automation*.

Rakotondrabe, M. (2013). Combining self-sensing with an unkown-input-observer to estimate the displacement, the force and the state in piezoelectric cantilevered actuator. In *American Control Conference*.

Rakotondrabe, M. (2014). Piezoelectric systems for precise and high dynamic positioning: design, modeling, estimation and control. *HDR habilitation, University of Franche-Comté*.

Schäfer, A.M. and Zimmermann, H.G. (2006). Recurrent neural networks are universal approximators. In *International Conference on Artificial Neural Networks*.

Schrauwen, B., Verstraeten, D., and Van Campenhout, J. (2007). An overview of reservoir computing: theory, applications and implementations. In *Proceedings of the 15th European Symposium on Artificial Neural Networks*, 471–482. Bruges, Belgium.

Snoek, J., Larochelle, H., and Adams, R.P. (2012). Practical bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, 2951–2959.

Soares, M.P., Rakotondrabe, M., and Ayala, H.V.H. (2020). Deep learning applied to data-driven dynamic characterization of hysteretic piezoelectric micromanipulators. *IFAC WC*.

Subudhi, B. and Jena, D. (2011). Nonlinear system identification using memetic differential evolution trained neural networks. *Neurocomputing*, 74(10), 1696–1709.

Tangirala, A.K. (2018). *Principles of system identification: theory and practice*. Crc Press, Boca Raton, FL, USA.

Wang, G. and Chen, G. (2017). Identification of piezoelectric hysteresis by a novel duhem model based neural network. *Sensors and Actuators A: Physical*, 264, 282–288.

Wang, Y. (2017). A new concept using lstm neural networks for dynamic system identification. In *American Control Conference*, 5324–5329.

Worden, K. et al. (2007). Identification of pre-sliding and sliding friction dynamics: Grey box and black-box models. *Mechanical Systems and Signal Processing*, 21(1).

Wu, R.T. and Jahanshahi, M.R. (2019a). Deep convolutional neural network for structural dynamic response estimation and system identification. *Journal of Engineering Mechanics*.

Wu, R.T. and Jahanshahi, M.R. (2019b). Deep convolutional neural network for structural dynamic response estimation and system identification. *Journal of Engineering Mechanics*.

Wu, Y. et al. (2020). Gated recurrent unit based frequency-dependent hysteresis modeling and end-to-end compensation. *Mechanical Systems and Signal Processing*, 136. 106501.

Xie, S. and Ren, J. (2019). Recurrent-neural-network-based predictive control of piezo actuators for precision trajectory tracking. In *American Control Conference*, 3795–3800.

Zhao, G. et al. (2019). System identification of the nonlinear residual errors of an industrial robot using massive measurements. *Robotics and Computer-Integrated Manufacturing*.

Zhou, D.X. (2020). Universality of deep convolutional neural networks. *Applied and Computational Harmonic Analysis*, 48(2), 787–794.