2023 IEEE Symposium Series on Computational Intelligence (SSCI)
Mexico City, Mexico. December 5-8, 2023

Fault Identification of Discrete-time unknown Non-linear Systems: A Two-dimensional Convolutional Neural Network Approach

Eduardo Rangel-Carrillo
Sciences and Education Department,
COPSIJAL, Professional College of Information Systems of Jalisco,
Guadalajara, México.
https://orcid.org/0000-0002-4506-6615

Alma Y. Alanis
Computer Sciences Department
CUCEI, University of Guadalajara
Guadalajara, México
https://orcid.org/0000-0001-9600-779X

***Abstract*— A complex system that is governed by several smaller sub-systems whose coordinated functionality allows it to work properly over time can be challenging to analyze for faults on real time by an observer; moreover, if such failing system could work with no obvious signs of fault over time until it becomes catastrophic and clearly identifiable. Because the variables involved in such system's functionality are usually not easily correlated, the different time-series they might generate can be extremely difficult to analyze by conventional means. Lately, 2-dimensional Convoluted Neural Networks (2D-CNN) have been used to introduce artificial intelligence into diagnosis and fault detection with success; however, the systems that so far have benefited from this are mainly those that deal with images, like medical diagnosis using x-ray images, or autonomous driving using real time pictures, although recently, some resent research on robotic sensor fault and signal analysis have been published using 1-dimensional CNN (1D-CNN) for time-domain signals.**

This paper proposes a novel 2D-CNN approach to fault identification of an unknown, discrete-time, non-linear system; by recognizing features that are consistent with a fault in a *signal-image* of several layers. With such *signal-image* being an artificial picture created by combining all the system signals in a single high-layered image format that is recognizable by a conventional 2D-CNN.

This paper also includes the results of its applicability in a fault identification of a three-phase induction motor in a simulation environment and with measurements of a real motor with injected faults.

Keywords—deep learning, fault identification, fault prediction, applied artificial intelligence. Convolutional neural networks non-linear systems.

I. INTRODUCTION

According to Bloch and Geitner in [1], machinery faults reveal a reaction chain of cause and effect, whose end is a performance deficiency. Such causes can by of several types like disturbances, uncertainties, erosion, over charge, fatigue, and even malicious intent. Once a fault has occurred, a process of collecting and analyzing data is started to determine the root cause of such fault; this process is known as *failure analysis* (FA) and it is applied in a wide variety of areas that range from geology, medicine, engineering to electronics and marketing, to mention a few. The very first step of the FA procedure is to determine preliminary fault mode, as Brady states in [2]. This means that for failure analysis to take place, the fault must be first clearly identified. Artificial Intelligence (AI) is increasingly applied in fault identification in a wide variety of problems, and more recently Deep Learning (DL) approach have been used for the prediction of different types of faults that can be related to object recognition, first proposed by Fukishima in his Neocognitron as a pattern recognition mechanism in [3] and then later applied to object recognition by LeCun in [4]. From there, more recent applications have been successfully implemented as a form of fault recognition, like the one shown by Shang *et. al* [5] in there, DL is used to recognize log patterns to detect early warning signals of IT systems fault. Or the one in [6] in where Shaheen & Hakan proposes a CNN to identify fractures in the Anatolian plate. Another AI-based approach for diagnostics is offered by Sadoughi *et al* [7] where they use a 1-Dimensional multichannel CNN to classify the condition of the bearing of a rotating machinery system. Additionally, Pan proposed a Deep CNN solution for sensor and actuator fault diagnosis [8] using a 1D-CNN as a fault diagnosis framework.

In the medicine field, CNN application is even more widely spread, like the work of Park *et al* in [9] in where a CNN is successfully implemented to diagnose appendicitis in patients using CT scan images. Or the approach proposed by Yadav and Jadhav in [10] in which a Deep CNN is used to diagnose pneumonia.

In the other hand, resiliency for artificial systems has caught the attention of scientists. In that direction, Rieger *et al* [1] defines a resilient system as one that maintains an accepted level of operational normalcy in the presence of faults. Recently, model-free control systems based on AI have been successfully applied for resilient control, like the one presented by Alanis in [11]. Such model implemented a recurrent high order neural network (RHONN) for output trajectory tracking and to allow an inductor motor to function with relative normalcy even in the presence of a sensor fault. To be able to either diagnose a fault, root cause it, predict it or control a system on its presence, some form of fault identification needs to be implemented. For simple systems when the presence or absence of any given signal represent a fault, this step might seem trivial; however complex systems may have signals ranging within expected boundaries and present only intermittent peaks or valleys that implies an imminent fault but are difficult to catch on a real time basis, like peaks on glucose levels on a non-diagnosed diabetic person; such signals are clearly an early warning of a diagnose but standard single-time-point measurements are very limited to catch it due to the fact that they neglect glucose dynamics [12]. Even more, a system may have multiple signals to be between boundaries but their dynamics might be off, thus setting the system enroute to a fault scenario, like for example the complex system of an Fuse Deposition Modeling (FDM) 3D-printer in which the extruder temperature and bed temperature might be in range, as well as the stepping motors might be correctly aligned in between them but a seemingly normal cooling fan might be not providing enough cool air to the nozzle and ends up building a clough in the material after several hours of printing [13].

From the above literature review we can see that for complex systems in whose subsystem signal dynamics are strongly intertwined, single-time fault identification forms might be too limited to prevent a catastrophic fault. Our proposed methodology is motivated by previous works and their successful results. This paper implements an AI approach using a 2D-CNN structure that unlike other stablished methods, considers all the available subsystem signals over a significant operational period and their intertwined dynamics to identify a possible fault. The model explicitly recognizes the *features* that appear on a system`s dynamic during training and classify any given sample as either a *fault or non-fault*.

According to above, the main contributions of this paper can be defined as follows:

1. Our work proposes a neural model for fault identification of several intertwined signals and their dynamics.
2. This model proposes the use of conventional 2D-CNN solution to create a *signal-image* that allows for artificial creation of a “picture” that allows the CNN to recognize dynamics by feature recognition, much like they recognize any given object.
3. The applicability of this model is shown by simulation training and experimental results are shown using real inductor motor signals.

This paper is organized as follows:

Section 2 analyze the problem statement of this paper. Section 3 describes the proposed methodology for the design of a neural model for fault identification with a 2D-CNN approach. Section 4 presents the simulation results for an induction motor application. Section 5 shows the results of the proposed methodology applied on a real inductor motor. Finally in Section 6 conclusions are stated and future work is listed

II. Problem statement

In [11], Alanis considers a discrete-time unknown non-linear system, given by

$$x(k+1) = F\big(\bar{x}(k), \bar{u}(k)\big) + d(k),$$
$$y(k) = C\bar{x}(k), \tag{1}$$

where $x \in \mathcal{R}^n$ is the state vector of the system, $u \in \mathcal{R}^m$ is the input vector, $y \in \mathcal{R}^p$ is the output vector, C is the output matrix, $d(k)$ is the disturbance vector, and $F(*)$ is a smooth vector field with $F_i(*)$ being its entries; therefore we can define (1) as follows:

$$x(k) = [x_1(k) \dots x_l(k) \dots x_n(k)]^T$$
$$F(k) = [F_1(k) \dots F_l(k) \dots F_n(k)]^T$$
$$d(k) = [d_1(k) \dots d_l(k) \dots d_n(k)]^T$$
$$x_i(k+1) = F_i\big(\bar{x}(k), \bar{u}(k)\big) + d(k), i = 1, \dots, n.$$
$$y(k) = Cx(k), \tag{2}$$

with

$$\bar{x}(k) = x(k) + \Delta_x,$$
$$\bar{u}(k) = u(k) + \Delta_u,$$

where Δ_x and Δ_u represent state and input uncertainties respectively and they are considered unknown and bounded. A Schematic representation of (1) can be appreciated in Fig. 1:

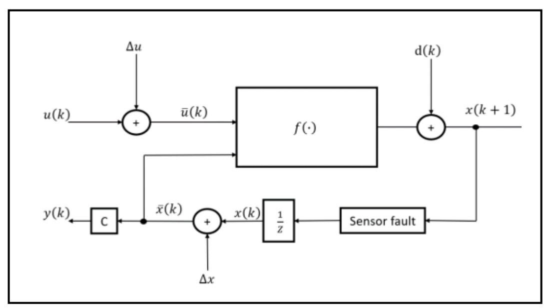


Fig 1. Schematic representation of a discrete-time nonlinear system with disturbances at the state and at the input.

For fault detection purposes, the intrinsic values of the output, input and state vectors and their internal dynamics of system (1) can result in only two possible outcomes: *fault* or *non-fault*, which can be expressed as a classification problem described by

$$G\big(x_i(k+1), y(k)\big) = \begin{cases} failure, & G(*) \in S \\ non-failure, & G(*) \notin S \end{cases} \tag{3}$$

where $G(*)$ represents the dynamics governing all the signals of system (1) and S is the subset of all possible fault modes. Because for a given discrete-time nonlinear system might be too difficult to entirely define *S*, this subset is considered only partially known.

To be able to identify a fault scenario in system (1) using the dynamics shown by (3) it is necessary for the following assumptions to be true:

Assumption 1.- The full state $x(k) = [x_1(k) \dots x_l(k) \dots x_n(k)]$ of system (1) is available.

Assumption 2.- There exists a data-structure in the dynamics of (3) that can be identified as *features* in a multidimensional array.

If the above assumptions are met, then we can say that it is possible to identify a *fault scenario* in system (1) by combining all signal inputs in a multidimensional array, or a t third order tensor given by:

$$\begin{gathered} \mathcal{A}_{lmn}, \\ l, m = 1, \dots, r \\ n = 1, \dots, v \end{gathered} \tag{4}$$

where r represents the total measurements taken from system (1) of any given signal and v represents the total signals available for that same system.

This representation allows us to present tensor data in *slices* as defined by Newman *et al* in [14] and shown in Fig. 2:

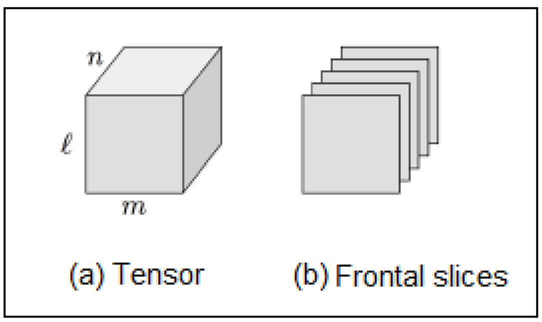


Fig 2. Graphic representation of A Tensor and the frontal data slicing

III. Two-Dimensonal Convolutional Neural Network Approach

The format of the data shown in Figure 2(b) will be referred as *signal-image* and it is suitable to be used with an standard 2D-CNN methodology like the one shown in [4] to recognize an *object* which, in this paper is the *fault;* and in order to do so, the network will recognize *features* which in this case are the dynamics of the signals of system (1).

This approach differs from the methodology using 1D-CNN shown in previous research; the main difference is that 1-dimensional convolution samples all the inputs with the same kernel to create an output matrix, much like the one shown in Fig. 3.

Therefore, when using 1-D CNN any single signal sampling is limited to the size of the kernel, and since modern literature like the one written by Goodfellow *et al.* in [15] defines standard kernel size for CNN to be values of 3x3 or 5x5 in some cases, this could limit the ability of the solution to obtain features of a complex dynamic system, especially when high number of inputs are involved. Another reason to use 2D-CNN is because by using front slices like the ones shown in Fig. 2(b), all of the inputs are separated as independent layers and convoluted among their own time-series, which allows to include more sensors and intertwined dynamics simply by adding more layers to the input.

The schematic representation of the proposed solution is shown in Fig. 4.

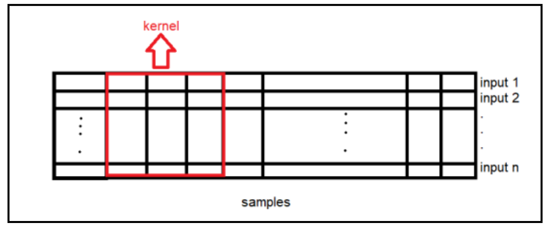


Fig 3. Convolution representation of a 1-dmensional CNN

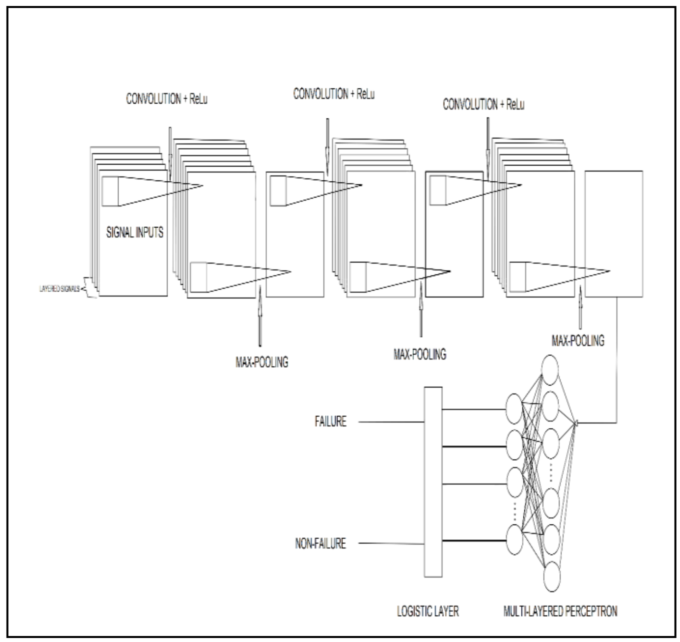


Fig 4. Schematic Diagram of the 2D-CNN approach for fault identification of a discrete-time nonlinear system

IV. Simulation Results

For this section, the simulation of an inductor motor like the one shown in Figure 5 where used. Data simulated corresponds to the well-known Clarke's transformation for a three-phase induction motor [16] and the data extracted for fault identification correspond to 9 signals of the complete model:

Timestamp for the simulation (t), rotor position (pos), angular velocity (ω), *stator alpha-flux* (ψ_α), *stator beta-flux* (ψ_β), *stator alpha-current* (i_α), *stator beta-current,* v_α.-

stator alpha-voltage, v_β.- stator beta-voltage

(5)

Simulation allows to inject several fault modes into the motor model in a safe environment; and for this case more than $\boldsymbol{78x10^3}$ samples were considered to simulate as much as possible the motor state-vector space that includes normal operation as well as several injected types of faults.

Fig. 5 shows the Simulink model and Fig. 6 shows a small sample of the simulated states.

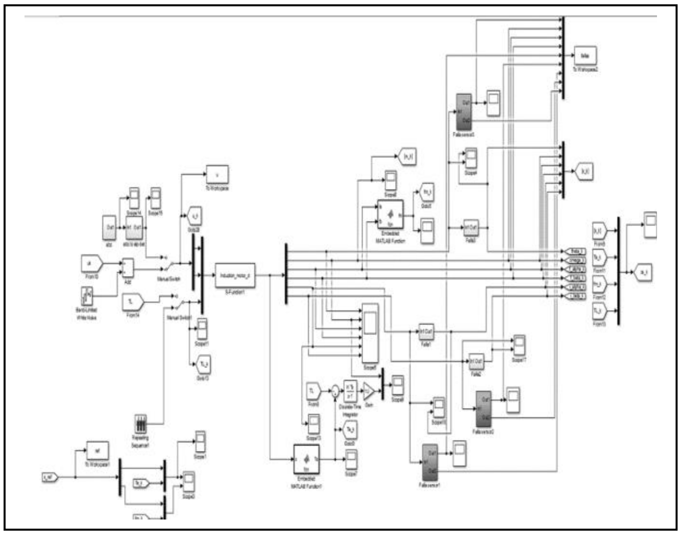


Fig. 5. SIMULINK model of a three-phase motor

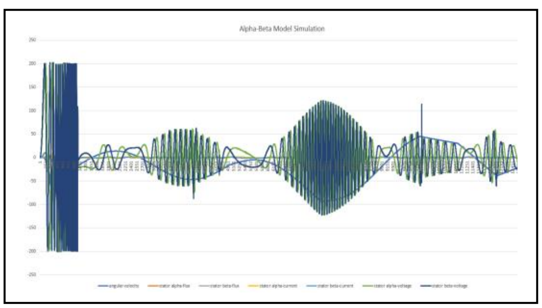


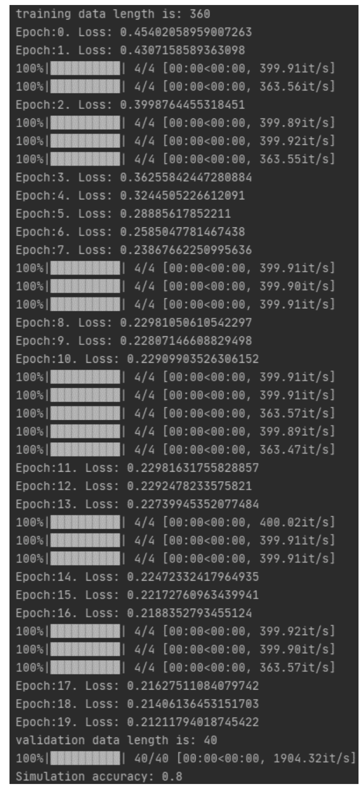
Fig 6. Three-phase motor simulated data extraction.

After extracting the simulation results, they were classified as stated in (3) and then used to train and validate the proposed 2D-CNN model shown in Figure 4. For the experiment to take place, the following configuration of our novel 2D-CNN model for fault identification is shown in Table I.

TABLE I. MODEL CONFIGURATION

Configuration Parameter	Value
Signal-Image Size	25
Signal-Image Dimensional layers	9
First Convolutional layer feature layers	32
Second Convolutional layer feature layers	64
Third Convolutional layer feature layers	128
Kernel size for all layers	3
Max-pooling size for all layers	2
Perceptron First layer size	128
Perceptron hidden layer size	64
Batch Size	100
Epochs	20
Training data set	90%
Validating data set	10%

After running the experiment for a training and validating cycle of 20 epochs, the final loss achieved was 0.2121, and validation resulted in an accuracy of 0.8, the numeric results are shown in Fig. 7(a) and the training loss is presented in Fig. 7(b).



```
training data length is: 360
Epoch:0. Loss: 0.45402058959007263
Epoch:1. Loss: 0.4307158589363098
100%|██████████| 4/4 [00:00<00:00, 399.91it/s]
100%|██████████| 4/4 [00:00<00:00, 363.56it/s]
Epoch:2. Loss: 0.3998764455318451
100%|██████████| 4/4 [00:00<00:00, 399.89it/s]
100%|██████████| 4/4 [00:00<00:00, 399.92it/s]
100%|██████████| 4/4 [00:00<00:00, 363.55it/s]
Epoch:3. Loss: 0.36255842447280884
Epoch:4. Loss: 0.3244505226612091
Epoch:5. Loss: 0.28885617852211
Epoch:6. Loss: 0.2585047781467438
Epoch:7. Loss: 0.23867662250995636
100%|██████████| 4/4 [00:00<00:00, 399.91it/s]
100%|██████████| 4/4 [00:00<00:00, 399.90it/s]
100%|██████████| 4/4 [00:00<00:00, 399.91it/s]
Epoch:8. Loss: 0.22981050610542297
Epoch:9. Loss: 0.22807146608829498
Epoch:10. Loss: 0.22909903526306152
100%|██████████| 4/4 [00:00<00:00, 399.91it/s]
100%|██████████| 4/4 [00:00<00:00, 399.91it/s]
100%|██████████| 4/4 [00:00<00:00, 363.57it/s]
100%|██████████| 4/4 [00:00<00:00, 399.89it/s]
100%|██████████| 4/4 [00:00<00:00, 363.47it/s]
Epoch:11. Loss: 0.22981631755828857
Epoch:12. Loss: 0.2292478233575821
Epoch:13. Loss: 0.22739945352077484
100%|██████████| 4/4 [00:00<00:00, 400.02it/s]
100%|██████████| 4/4 [00:00<00:00, 399.91it/s]
100%|██████████| 4/4 [00:00<00:00, 399.91it/s]
Epoch:14. Loss: 0.22472332417964935
Epoch:15. Loss: 0.22172760963439941
Epoch:16. Loss: 0.2188352793455124
100%|██████████| 4/4 [00:00<00:00, 399.92it/s]
100%|██████████| 4/4 [00:00<00:00, 399.90it/s]
100%|██████████| 4/4 [00:00<00:00, 363.57it/s]
Epoch:17. Loss: 0.21627511084079742
Epoch:18. Loss: 0.21406136453151703
Epoch:19. Loss: 0.21211794018745422
validation data length is: 40
100%|██████████| 40/40 [00:00<00:00, 1904.32it/s]
Simulation accuracy: 0.8
```

Fig 7(a).Simulation results.

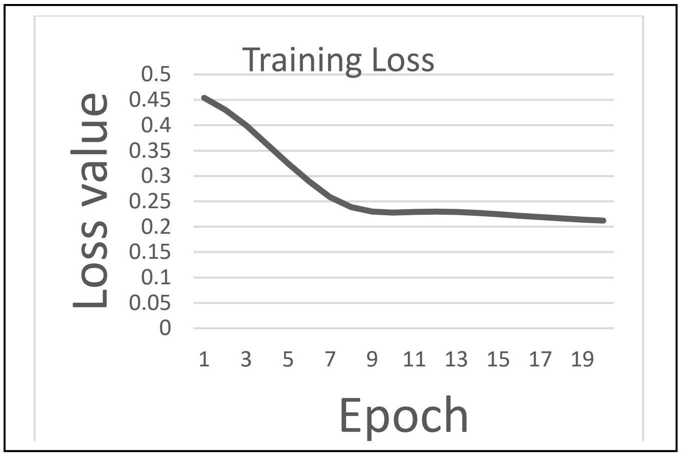


Fig 7(b). Training and validation results

To verify the stability of the network, a Monte Carlo experiment run of 20 cycles was performed on the training and the result was an average Loss of 0.2 which is shown in Fig. 8.

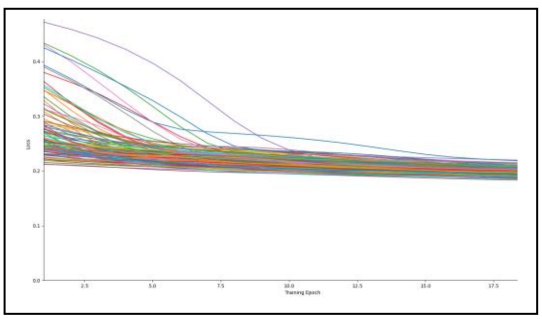


Fig 8. Monte Carlo results for training Loss.

V. REAL SCENARIO RESULTS

After simulation results are obtained, the trained model was used in a real-time scenario with data taken from a real induction motor like the one shown in Fig. 9.

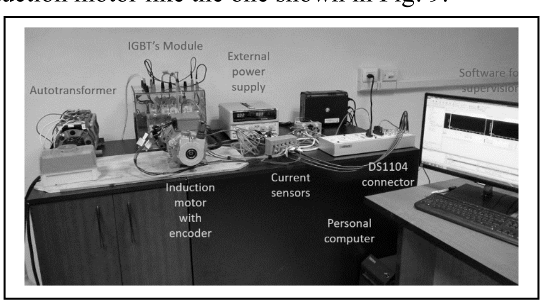
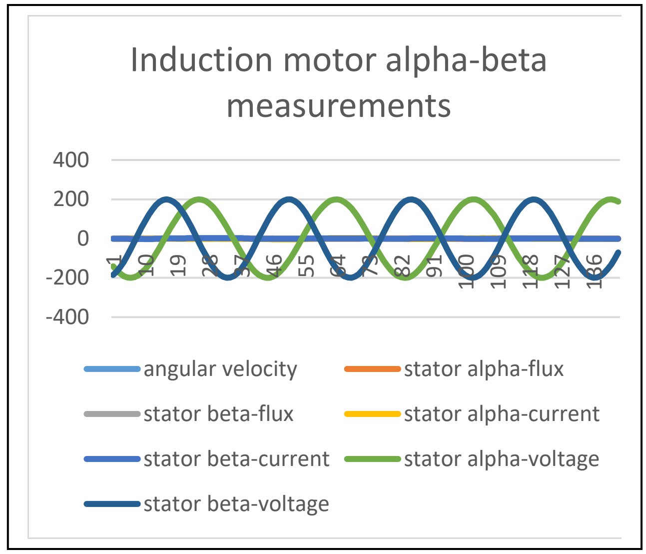


Fig 9. Physical induction motor for the experiment.

For this case more than $16x10^3$ samples were considered, and they include normal operation as well as some real faults. Fig. 10 shows a small sample of the simulated states.

Fig 10. Three-phase motor data measurements sample.



The proposed model of Fig. 4 that was trained with simulation data is now tested with the real measurements and the result was an average ***testing accuracy of 0.6427***.

Finally, to check the stability of our proposed novel model over time, a Monte Carlo experiment run of 100 cycles of complete training, validating, testing phases was performed, and the results are listed in Table II and presented in Fig. 11.

TABLE II. ACCURACY RESULTS

Accuracy	**Simulation Data**	**Real scenario data**
Maximum	0.8750	0.6350
Minimum	0.6531	0.4731
Average	0.7450	0.6427

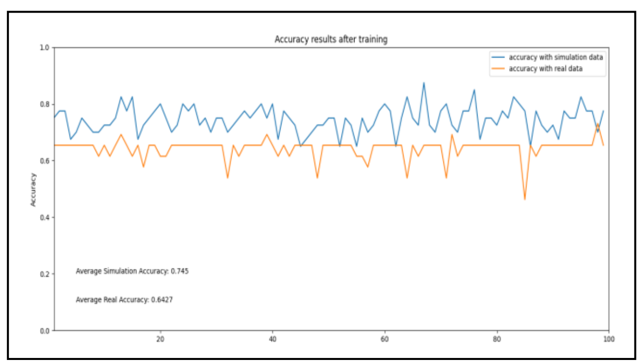


Fig 10. Monte Carlo experiment of 100 cycles of training-validating-testing of the 2D-CNN Model for fault identification

VI. RESULTS DISCUSSION AND CONCLUSIONS

In this work, a novel 2-dimensional CNN approach for fault identification of an unknown non-linear discrete system was presented. The model was then configured for a three-phase inductor motor system in both simulation and real environments showing good training loss and fault identification accuracy results in a simulation scenario and with real-time measurements. Furthermore, the model was tested for stability using Monte Carlo experiments and showed to be a stable solution for the problem statement of this paper, with simulation and real time accuracy variance ***~20 %*** with only one of the cycles performing significantly lower than the random accuracy of **0.5** in the real data scenario and few valleys in accuracy with real data found across the results shown in Figure 12.

The resulting accuracy of the testing scenario with real data was lower than the simulation results since training data did not include any real time measurements for two reasons: the injected faults if the real motor were too limited for training purposes and by excluding the real faults, we could test the general capacities of the model to identify a fault not known before.

The main contribution of this work is that it shows it is possible to achieve successful fault identification of an unknown non-linear discrete system using a conventional 2D-CNN approach and gaining all the benefits of this technology.

Even more, our experiment shows that it is possible to transform several time-series of complex non-linear systems into a 2-dimensional picture format with layers called "*signal-image"* and by doing so, it opens the possibility to take advantage of well-known 2D-CNN models like ResNEt50, InceptionV3, VGG16, etc. that so far have been successfully used for image classification; but now with the presented model, it could be possible to identify faults in unknown non-linear discrete systems in a similar way it is possible to identify an image by its characteristics.

Because of the opportunities that can rise from the implementation of our model, future work will be focused in implementing variants of this model in several non-linear systems in a wide range of applications such as engineering, social or even medical scenarios.

ACKNOWLEDGMENT

Authors of this paper thank the University of Guadalajara and COPSIJAL for their support on this work.

REFERENCES

[1] C. G. Rieger, D. Gertman and M. A. McQueen, "Resilient control systems: Next generation design research," in *IEEE- 2nd Conference on Human System Interactions*, Catania, 2009; DOI: 10.1109/HSI.2009.5091051.

[2] B. G. Brady, "Failure Analysis," The Materials Science and Engineering Department of the State University of New York, New York City, NY, 1999.

[3] K. Fukushima, "Neocognitron," *Biological Cybernetics, Springer-Verlag,* vol. 2, no. 1, pp. 193-202, 1980; DOI: 10.1007/BF00344251.

[4] Y. LeCun and P. e. a. Haffner, "Object Recognition with Gradient-Based Learning," Springer, Red Bank, 1999; DOI: 10.1007/3-540-46805-6_19.

[5] K. Zhang, J. Xu, M. Renqiang, G. Jiang, K. Pelechrinis and H. Zhang, "Automated IT System Failure Prediction: A Deep Learning Approach," in *2016 IEEE International Conference on Big Data (Big Data)*, Washington D.C., USA, 2016; DOI: 10.1109/BigData.2016.7840733.

[6] M. S. A. Shaheen and H. Guneyli, "Improved deep learning (inception) CNN for detecting small tectonic plates of broken Anatolian, Turkey," *All Earth,* vol. 33, no. 1, pp. 52-65, 2021; DOI: 10.1080/27669645.2021.1945523.

[7] M. Sadoughi, A. Downey, G. Bunge, A. Ranawat and H. Chao, "A Deep Learning-based Approach for Fault Diagnosis of Roller Element Bearings," in *ANNUAL CONFERENCE OF THE PROGNOSTICS AND HEALTH MANAGEMENT SOCIETY*, Philadelphia, Pennsylvania, USA., 2018.

[8] J. Pan, L. Qu and K. Peng, "Sensor and Actuator Fault Diagnosis for Robot Joint Based on Deep CNN," *Entropy,* vol. 23, no. 6, 2021; DOI: 10.3390/e23060751.

[9] J. J. Park, K. A. Kim, Y. Nam, M. H. Choi, S. Y. Choi and J. Rhie, "Convolutional-neural-network-based diagnosis of appendicitis via CT scans in patients with acute abdominal pain presenting in the emergency department," *Scientific Reports,* vol. 10, no. 9556, 2020; DOI: 10.1038/s41598-020-66674-7.

[10] S. S. Yadav and S. Jadhav, "Deep convolutional neural network based medical image classifcation for disease diagnosis," *Journal of Big Data volume,* vol. 6, no. 113, 2019; DOI: 10.1186/s40537-019-0276-2.

[11] A. Alanis and J. Alvarez, "Real-time model-free resilient control for discrete nonlinear systems," *Asian Journal of Control,* vol. 1, no. 1, p. 15, 2021; DOI: 10.1002/asjc.2564.

[12] H. Hall, D. Perelman, A. Breschi, P. Limcaoco, R. Kellogg and T. McLaughlin, "Glucotypes reveal new patterns of glucose dysregulation," vol. 16, no. 7, 2018; DOI: 10.1371/journal.pbio.2005143.

[13] S. Aranda, 3D Printing Failures: 2020 Edition: How to Diagnose and Repair ALL Desktop 3D Printing Issues, David Feeney, 2020; ASIN: B081X8PYPQ.

[14] E. Newman, M. Kilmer and L. Horesh, "Image classification using local tensor singular," in *IEEE 7th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 2017; DOI: 10.1109/CAMSAP.2017.8313137.

[15] I. Goodfellow, Y. B. Courville and C. Aaron, Deep Learning, Cambridge, Massachusstes: MIT Press, 2017; DOI: 10.1007/s10710-017-9314-z.

[16] C. J. O'Rourke, M. M. Qasim, M. R. Overlin and J. L. Kirtley, "A Geometric Interpretation of Reference Frames and Transformations: dq0, Clarke, and Park," *IEEE Transactions on Energy Conversion,* vol. 34, no. 4, pp. 2070-2083, 2019; DOI: 10.1109/TEC.2019.2941175.

[17] . Heinz and F. Geitner, Machinery Failure Analysis and Troubleshooting, Houston, Texas: Gulf Publishing Company, 1994; DOI: 10.1016/C2010-0-67104-5.