XLIII CILAMCE
21-25, November, 2022
FOZ DO IGUAÇU - PR - BRAZIL

Structural Health Monitoring of prestressed concrete beams using different CNNs architectures

Diego G. de Lucena, Francisco A. das Neves[1], Cláudio José Martins[2]

[1]*Dpt. of Civil Engineering, Federal University of Ouro Preto (UFOP)*
Address, Zip-Code, State/Province, Country
fassis@ufop.edu.br
[2]*Dpt. of Civil Engineering, Federal Center for Technological Education of Minas Gerais (CEFET-MG)*
Address, Zip-Code, State/Province, Country
somebody3@somewhere.com

Abstract. There are several damage detection techniques that use signal data and can evaluate and ensure the safety of a structure. Recently, machine-learning algorithms have been used to help classify and detect damages as well as extract structural features from signal data. Learning algorithms have the advantage of using raw data or data with minimum pre-processing as input. Convolution neural networks (CNN) is a supervised learning technique that uses a combination of filters and pooling layer to extract features and classify these types of data simultaneously. The performance of CNNS can vary drastically due to the architecture and the input shape of the input selected. In this study it is compared different CNNs proposed by the literature in order to identify damage presented in a set of eight identical prestressed concrete beams. Tests with sixteen 1D and 2D different CNNS were conducted with results of accuracy, recall and f1-score varying from 50% to 99%.

Keywords: Convolutional Neural Networks, Prestressed Concrete, Structural Health Monitoring

1 Introduction

The structural aging has a direct adversely influence on stability, stiffness and life cycle performance of any structure.This aging effects may be caused by environmental, human-induced or operational factors [1, 2]. In order to perform the structural health monitoring (SHM) of structures several different methods have been proposed in the literature, but usually vibration based methods are the most common ones. These methods can be categorized into parametric and nonparametric. Nonparametric methods uses statistical techniques to identify damages on the structures directly from signal data, while parametric methods uses system identification algorithms to extract the modal parameters from the measured response data [2, 3].

The development in data acquisition equipment for SHM systems not only made them more precise, but also drastically increased the size of the data set theses systems can now record. These new, larger and more precise data set pushed for intelligent algorithms to be adopted in order to detect and monitor damage in the analysed structures [4–6]. Recently studies have been focusing in automated methods to identify structural damages, they are called data-driven methods. Data-driven algorithms are normally implemented as a pattern recognition problem and uses techniques such as Neural Networks (NN), genetic algorithms, support vector machine, and more recently Convolutional Neural Networks (CNN) to deal with these large databases [5–7].

Different NNs have been applied to Civil Engineering problems in order to assist damage detection in structures. Zhang et al. and Eisenbach et. al. 2016, 2017 used neural networks to identify cracks and fissures on the surface of roads, Cha, Choi and Büyüköztürk 2017 used images to find different damages in concretes, Cha et al. 2018 evaluated videos in order to find damages on the surface of steel structures, Kang and Cha 2018 used drone images to cracks in concrete floors and Bao et al.2019 and Tang et al. 2019 used NN to reconstruct and/or detect anomalies in data signals.

Convolutional neural networks are a specific type of neural networks that uses two special layers, the convolution and the pooling layers. The CNNs gained a lot of attention in recent works related to image processing and patter recognition, as it can be used for different types of data. These special layers are capable to make a domain-aware regularization of a multichannel input, such as images and videos [15].

As the precision level of a CNN model in relation of a specific data can varies drastically, this work aims to review eight different CNN architectures proposed by different authors in order to analyse which variables can affect the prediction results of the models. The database selected consists of accelerations of eight prestressed concrete beams before and after the application of the prestressing forces. All CNNs were implemented in Python 3 using the TensorFlow and Keras open source libraries.

2 Overview of Convolution Neural Networks

CNNs are a type of machine learning algorithms that was inspired by how the visual cortex system of mammals would discern, recognize and memorize different patterns. The main features of a CNN, as discussed before, is the utilization of pooling and convolution layers in their structures, but they also have a final block of layers called fully connected layer that has the same function of a regular NN [2, 3].

The main difference of a CNN and other NNs is then the utilization of at least one convolution layer and one pooling layer, instead of the multiplication of the array of neurons [5, 16]. The most important consequences of this use is that: CNNs will learn smaller sets of filters that can be applied to the hole input (shared weights); the creation of a maximum array that reduces the size of the data representation and computation required for the next layer (spatial/temporal subsampling); and the filters generated by the CNN learn patterns from specific regions of the previous layers, allowing the network to learn complex patterns from combination of simple local operations (local receptors field) [16].

Despite these differences, the CNNs will use the gradient descent, the back propagation and the variations of these two algorithms in a similar manner as the usual NN does.The figure 1 is a representation of a regular CNN.

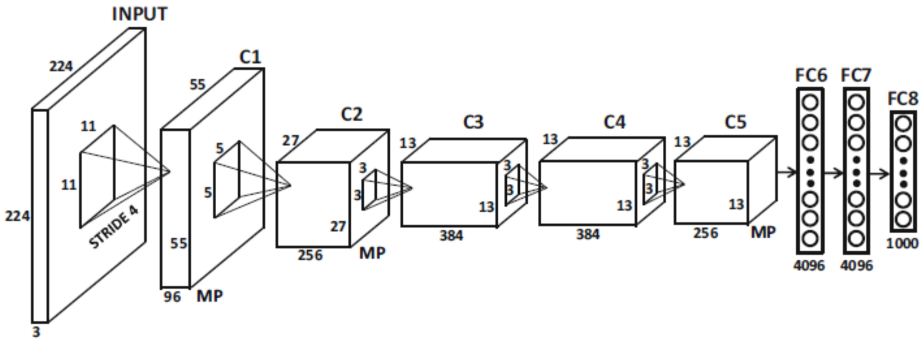

Figure 1. Example of a basic CNN architecture

2.1 Convolution layer

The convolution layers is in general the responsible for the major computational task in the CNN [4]. The input data for every convolution layer is convolved with kernels, which are abstraction of filters that will be responsible to learn specific patters from different regions of the input [6].

Convolution is normally defined as a transformation between two real valued functions, and can be described as[17]:

$$f(i) = \int_{-\infty}^{\infty} s(n)k(i-n)dn \tag{1}$$

In CNNs the equation 1 above can be rewritten as:

$$S(i,j) = \sum_{m}\sum_{n} I(i+m, j+n)K(m,n) \tag{2}$$

where I is the input matrix of the currently convolution layer, K is the kernel. Note that the feature map S is constructed applying the same kernel throughout the length and width of the input [5, 17, 18]. The kernels are initiated randomly and biases are added to them, resulting in unique feature maps. The convolution of the kernel through the hole input helps to detect local high correlated groups as well as making the filters statistically invariant to position[7, 18].

2.2 Pooling layer

The pooling layers is a downsample filter that is applied throughout the entire input of the layer, the layer it can be setted to extracted the minimum, maximum or any other values relevant from the data [17, 18].

The is no learning process in the pooling layer, the main purpose of the pooling layer is to eliminate non-maximum (or minimum) values, while reducing the data size and the computation required for the next layers [16, 18].

The pooling layer is also responsible for reducing the variance of the input data so it is not reflected on the outputs, making a synthesis of the small parts of the input data it filtered before[5, 6]. The most used pooling layer is know as the maxpooling layer.

3 Description of the prestressed concrete beams

The structures analysed are eight prestressed concrete beams built in an industrial environment for a highway bridge by Diego 2019. One of the beam in the storage area is shown in the figure 2

Figure 2. Prestressed concrete beam in storage

In this study the concept of a positive damage was used, as the stiffness of the beams increase when the presstressing forces are applied to the beams.The beams were evaluated then before and after the applications of the prestress forces.

Two triaxial accelerometers were used to record the data, one of them was placed in the middle of the beam and the other on 2/3 of the beam spam. A 30 minute time sample was recorded for each state of the beam before and after the prestress is applied. The sample frequency selected was 1 kHz, this value was selected in order to ensure all modal frequencies could be evaluated.

4 Experimental results

The data set acquired consists of 8 acceleration data recordings of the beams before the presstress forces are applied and 8 recordings after, with 1.8E6 point of data per acquisition. These two types of data consists of the two classes that will be considered during the tests.

In order to evaluate the influence of different CNNs configurations and inputs have in the prediction result of for the two classes proposed. Eight different CNNs will be evaluated. Abdeljaber et al. 2017, Zhang et al. 2019, Khodabandehlou, Pecan and Fadali 2019, Azimi and Pekcan 2020, Liu et al. 2020, Park et al. 2020, Puruncajas, Vidal and Tutivén 2020 and Rezende et al. 2020. Table show a resume of the CNN proposed by the authors

Table 1. Precision, recall and f1-score of the evaluated CNNs

	Adeljaber et al. (2017)	Zhang et al. (2019)	Khodabandehlou, Pecan and Fadali (2019)	Azimi and Pekcan (2020)
Type of CNN	1D	1D	2D	1D
Number CONV layers	2	1	5	9
Number of kernels	32, 64	5	4, 8, 16, 32, 32	16, 16, 16, 32, 32, 32, 64, 64, 64
Shape of kernels	3x1	10x1	3x3	3x1
Pooling layers	0	1	5	3
Shape of layers	-	3x1	2x2	2x1
FC layers	2	2	4	3
Number of neurons	10, 1	40, 2	60, 60, 60, 2	10, 10, 2
Input size	1000	1000	6000	1000
Input shape	1000x1	1000x1	200x180	1000x1

Table 2. Precision, recall and f1-score of the evaluated CNNs

	Liu et al. (2020)	Park et al. (2020)	Puruncajas, Vidal and Tutivén (2020)	Rezende et al. (2020)
Type of CNN	1D	2D	2D	1D
Number CONV layers	2	2	6	1
Number of kernels	32, 64	10, 20	32, 64, 168, 256, 128, 64	100
Shape of kernels	5x1	12x12, 4x4	5x5	224x1
Pooling layers	1	2	0	1
Shape of layers	5x1	6x6, 4x4	-	2
FC layers	1	1	4	3
Number of neurons	2	2	32, 16, 4, 2	10, 10, 2
Input size	1000	1200	1000	1000
Input shape	1000x1	75x96	40x25x6	1000x1

It must be noted that all many of the CNNs evaluated have different input shapes and sizes as well as different data sets in some cases. The architecture proposed by each author differs from another in the number of convolution and pooling layers, number of layers and neurons in the fully connected layers as well as the activation function used throughout the hole CNN. Some of the CNNs evaluated have other different layers that are not specific of the CNN, such as batch normalization and dropout layers. To better understand the CNN proposed by each author we refer to the articles cited here.

To evaluate the CNNs it was used three different metrics of evaluating NN, the precision, recall and f1-score. The equation for each metric is present in the equation respectively.

$$Precision = \frac{TruePositives}{(TruePositives + FalsePositives)} \tag{3}$$

$$Recall = \frac{TruePositives}{(TruePositives + FalseNegatives)} \tag{4}$$

$$f1 - score = \frac{2 * Precision * Recall}{(Precision + Recall)} \tag{5}$$

Table 3 presents the results of precision, recall and f1-score achieved by each CNN evaluated in this study. The values between precision, recall and f1-score do not change drastically for the same CNN, which means that they are not biased to a specific class of the problem and are classifying both the classes properly.

Table 3. Precision, recall and f1-score of the evaluated CNNs

	Precision	Recall	f1-score
Abdeljaber et al. (2017)	0.6280	0.6280	0.6279
Azimi and Pekcan (2020)	0.9092	0.9063	0.9062
Khodabandehlou, Pecan and Fadali (2019)	0.2483	0.4983	0.3315
Liu et al. (2020)	0.9694	0.9694	0.9694
Park et al. (2020)	0.9744	0.9743	0.9743
Puruncajas, Vidal and Tutivén (2020)	0.9405	0.9387	0.9387
Rezende et al. (2020)	0.9473	0.9473	0.9472
Zhang et al. (2019)	0.7105	0.7096	0.7091

The comparison of the parameters between the CNNs evaluated show that the input shape and size as well as the CNN architecture influence the classification problem. It can also be noted that the depth of the CNN, the number and complexity of the layers, is not necessarily a parameter that is proportional to the quality of the results. Khodabandehlou, Pecan and Fadali 2019 proposed the CNN with the most numbers of convolution layers and achieved the worst performed with a f1-score of 33.15%..

The best results for the data set evaluated were achieved by the CNN proposed by Park et al. 2020 with a 97.43% of f1-score. This means that the CNN was capable to positively distinguish between the beam before and after the prestress been applied to the structure almost in 98% for eight similar beams.

The type of CNN, if it is either a 1D or 2D CNN did not influenced the performance. The two best CNNs are from Park et al. 2020 and Liu et al. 2020 which are a 2D and a 1D CNN respectively.

5 Conclusions

In this work a series of CNNs were tested for data set consisted of 16 acceleration recordings, 8 before and 8 after the prestress forces were applied to the studied structure. The test were taken in order to evaluate the viability of the CNNs to detect this type of positive damage and to understand the parameters that can affect the classification process.

The results shows that a medium size CNN will present better results, probably due to a scaling difficulty to solve the algorithm as the CNN complexity grows with their size and depth. As mentioned, the type of CNN, 1D or 2D, did not influence in a significant matter to be considered a relevant parameter.

Finally, it has been showed that CNNs are cable to detect and distinguish between states of a structure, that their level of complexity can affect the results of the study and that it may be possible to define a specific set of parameter of a CNN so it can be used to multiple problems.

Acknowledgements. The authors are grateful for the financial support of the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), Fundação de Amparo à Pesquisa do Estado de Minas Gerais (Fapemig) e Universidade Federal de Ouro Preto (UFOP).

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] B. K. Oh and J. Kim. Optimal architecture of a convolutional neural network to estimate structural responses for safety evaluation of the structures. *Measurement*, vol. 177, pp. 109313, 2021.

[2] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, and D. J. Inman. Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks. *Journal of Sound and Vibration*, vol. 388, pp. 154–170, 2017.

[3] O. Abdeljaber, O. Avci, M. S. Kiranyaz, B. Boashash, H. Sodano, and D. J. Inman. 1-d cnns for structural damage detection: Verification on a structural health monitoring benchmark data. *Neurocomputing*, vol. 275, pp. 1308–1317, 2018.

[4] T. Liu, H. Xu, M. Ragulskis, M. Cao, and W. Ostachowicz. A data-driven damage identification framework based on transmissibility function datasets and one-dimensional convolutional neural networks: verification on a structural health monitoring benchmark structure. *Sensors*, vol. 20, n. 4, pp. 1059, 2020.

[5] M. Azimi and G. Pekcan. Structural health monitoring using extremely compressed data through deep learning. *Computer-Aided Civil and Infrastructure Engineering*, vol. 35, n. 6, pp. 597–614, 2020.

[6] de S. W. F. Rezende, de J. d. R. V. Moura, R. M. F. Neto, C. A. Gallo, and V. Steffen. Convolutional neural network and impedance-based shm applied to damage detection. *Engineering Research Express*, vol. 2, n. 3, pp. 035031, 2020.

[7] Y. Zhang, Y. Miyamori, S. Mikami, and T. Saito. Vibration-based structural state identification by a 1-dimensional convolutional neural network. *Computer-Aided Civil and Infrastructure Engineering*, vol. 34, n. 9, pp. 822–839, 2019.

[8] L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu. Road crack detection using deep convolutional neural network. In *2016 IEEE international conference on image processing (ICIP)*, pp. 3708–3712. IEEE, 2016.

[9] M. Eisenbach, R. Stricker, D. Seichter, K. Amende, K. Debes, M. Sesselmann, D. Ebersbach, U. Stoeckert, and H.-M. Gross. How to get pavement distress detection ready for deep learning? a systematic approach. In *2017 international joint conference on neural networks (IJCNN)*, pp. 2039–2047. IEEE, 2017.

[10] Y.-J. Cha, W. Choi, and O. Büyüköztürk. Deep learning-based crack damage detection using convolutional neural networks. *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, n. 5, pp. 361–378, 2017.

[11] Y.-J. Cha, W. Choi, G. Suh, S. Mahmoudkhani, and O. Büyüköztürk. Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, n. 9, pp. 731–747, 2018.

[12] D. Kang and Y.-J. Cha. Autonomous uavs for structural health monitoring using deep learning and an ultrasonic beacon system with geo-tagging. *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, n. 10, pp. 885–902, 2018.

[13] Y. Bao, Z. Tang, H. Li, and Y. Zhang. Computer vision and deep learning–based data anomaly detection method for structural health monitoring. *Structural Health Monitoring*, vol. 18, n. 2, pp. 401–421, 2019.

[14] Z. Tang, Z. Chen, Y. Bao, and H. Li. Convolutional neural network-based data anomaly detection method using multiple information for structural health monitoring. *Structural Control and Health Monitoring*, vol. 26, n. 1, pp. e2296, 2019.

[15] C. C. Aggarwal and others. Neural networks and deep learning. *Springer*, vol. 10, pp. 978–3, 2018.

[16] M. A. De Oliveira, A. V. Monteiro, and J. Vieira Filho. A new structural health monitoring strategy based on pzt sensors and convolutional neural network. *Sensors*, vol. 18, n. 9, pp. 2955, 2018.

[17] Y.-z. Lin, Z.-h. Nie, and H.-w. Ma. Structural damage detection with automatic feature-extraction through deep learning. *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, n. 12, pp. 1025–1046, 2017.

[18] Y. Duan, Q. Chen, H. Zhang, C. B. Yun, S. Wu, and Q. Zhu. Cnn-based damage identification method of tied-arch bridge using spatial-spectral information. *Smart Struct. Syst., Int. J*, vol. 23, n. 5, pp. 507–520, 2019.

[19] de D. G. Lucena. Influência das forças de protensão nos parâmetros dinâmicos de vigas em concreto. Master's thesis, Centro Federal de Educação Tecnológica de Minas Gerais, Belo Horizonte, MG, 2019.

[20] H. Khodabandehlou, G. Pekcan, and M. S. Fadali. Vibration-based structural condition assessment using convolution neural networks. *Structural Control and Health Monitoring*, vol. 26, n. 2, pp. e2308, 2019.
[21] H. S. Park, J. H. An, Y. J. Park, and B. K. Oh. Convolutional neural network-based safety evaluation method for structures with dynamic responses. *Expert Systems with Applications*, vol. 158, pp. 113634, 2020.
[22] B. Puruncajas, Y. Vidal, and C. Tutivén. Vibration-response-only structural health monitoring for offshore wind turbine jacket foundations via convolutional neural networks. *Sensors*, vol. 20, n. 12, pp. 3429, 2020.