Structural Health Monitoring of RC structures using optic fiber strain measurements: a deep learning approach

(article starts on next page)

Structural Health Monitoring of RC structures using optic fiber strain measurements: a deep learning approach

Dimitrios F. KARYPIDIS

M.Sc. Student

Chalmers University of Technology

Gothenburg, Sweden
dimkary@student.chalmers.se

Last year student at the Master Program Complex Adaptive Systems and research assistant in machine learning applications in civil engineering

Carlos G. BERROCAL

PhD

Chalmers University of Technology

Gothenburg, Sweden
carlos.gil@chalmers.se

Postdoctoral researcher in the field of durability of RC and structural monitoring currently working on digitalization of transport infrastructure

Rasmus REMPLING

Associate Professor

Chalmers University of Technology

Gothenburg, Sweden
rasmus.rempling@chalmers.se

Researcher in the field of Structural and Construction Engineering with interest in AI based and automated processes.

Mats GRANATH

Associate Professor

University of Gothenburg

Gothenburg, Sweden
mats.granath@physics.gu.se

Background in theoretical condensed matter physics and complex systems and researcher in AI based systems.

Peter SIMONSSON

PhD

Swedish Transport Administration

Luleå, Sweden
peter.simonsson@trafikverket.se

Bridge expert in a client organization and researcher in construction engineering and building processes.

Contact: carlos.gil@chalmers.se

1 Abstract

This paper reports the early findings of an ongoing project aimed at developing new methods to upgrade the current maintenance strategies of the civil and transport infrastructure. As part of these new methods, the use of Machine Learning (ML) algorithms is being investigated to constitute the core of a new generation of more accurate and robust structural health monitoring (SHM) systems for concrete structures. Unlike most of the existing SHM systems, relying on the analysis of the natural frequencies of the structure based on data obtained from accelerometers, the present study uses a distributed optic fiber system to monitor the strain distribution along steel reinforcing bars. The preliminary results of the study indicate that a semi-supervised Deep Autoencoder algorithm (DAE) can successfully quantify the damage attributable to transverse cracks in a reinforced concrete beam subjected to three-point loading. Future applications will feature the determination of crack locations, early detection of reinforcement corrosion as well as other types of damage such as splitting cracks or surface spalling.

Keywords: structural health monitoring, machine learning, deep autoencoders, anomaly detection, concrete structures, distributed optic fiber.

2 Introduction

Recent advancements in digital technology and communications have rendered possible the use of real-time monitoring systems, which constantly receive, process and analyze streams of data obtained from distributed sensor networks. Handling and using big streams of data is a tedious task, which has occupied data scientists for the past years. One of the most promising tools for handling

such tasks is Machine Learning (ML). In recent years, thanks to the increase in computational capacity of modern computers, a particular sub-field of ML called Deep Learning (DL) has dominated the research arena, since it appears the best-known approach to solve prediction and classification tasks [1]. DL has successfully been implemented and is considered to be the state-of-the-art method in a plethora of applications such as image recognition, self-driving cars, machine translation, financial time series prediction etc.

A potential application where DL stands out as a promising tool is Structural Health Monitoring (SHM). SHM is the constant monitoring of structural systems to detect, localize and assess irregularities and defects. Despite that SHM has been successfully implemented in sectors like aerospace and automotive industries, its application in the transport infrastructure has been hindered by the singular nature of civil structures. Current SHM systems still rely on deterministic methods, such as signal processing and Finite Element Analysis (FEA). These techniques, although very practical when used by themselves, suffer from two main issues: (i) non-robustness to noise and (ii) inflexibility. Nevertheless, more recent approaches have applied DL techniques in order to optimize the SHM procedures, most of them falling under the category of object detection (computer vision) and multi-feature classification [2] [3], which sometimes are non-practical to implement in large scale projects.

This paper explores the possibility of developing an anomaly detection system for reinforced concrete elements using DL. The system has been tested on RC beams subjected to 3-point bending where strains are measured via a distributed optical fiber attached to the reinforcement. Subsequently, the strain profiles were fed to a Deep Autoencoder Network (DAE) with different configurations. The results show that, after proper training, the network is able to detect the anomalous states of the beam by measuring the reconstruction error of the preceding observations.

3 Methodology

3.1 Deep Autoencoders

The goal of the current application is to build an anomaly detection system. To that end, one of the most commonly used neural network architectures is the Deep Autoencoder [1](DEA) (Fig. 1). DEA's fall under the category of the semi-supervised ML algorithms, which manipulate neural network architectures to solve the task of representation learning. Its purpose is to reduce the dimensions and keep the essential structural information of the data, similar to Principal Component Analysis. This technique works well when the data is multidimensional, and some of the features are either correlated or non-uniformly significant.
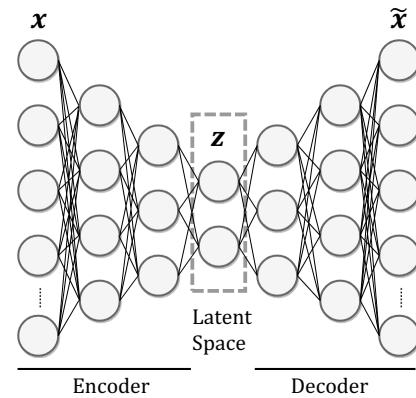

Figure 1. Example of a Deep Autoencoder Network architecture.

The unique property of DEA's compared to conventional artificial neural networks is that the network architecture consists of two parts: the encoder and the decoder. The encoder consists of sublayers of decreasing node number, "squeezing" the initial input x into the smallest sublayer, called bottleneck or latent-space layer. The number of neurons in the bottleneck layer correspond to the number of features we want to compress the data into. The decoder is the part of the network from the bottleneck until the output. The topology of the decoder sublayers is a mirror of the encoder. The purpose of Autoencoders is to map the input data to itself, i.e. minimize the function:

$$J(x, \tilde{x}) \equiv J(x, g(f(x))) \qquad (1)$$

where $x, \tilde{x}$ are the pair of original and reconstructed inputs, g and f are the decoding and encoding functions respectively and $J(x, \tilde{x})$ is the loss

function which must be optimized. In our case we use the typical Mean Squared Error (MSE) function shown below:

$$J(x, \tilde{x}) = \|x - \tilde{x}\|^2 + \lambda \sum_{i=1}^{n} w_i^2 \qquad (2)$$

where the second addition term is the $L2$ regularizer, which inhibits model overfitting without increasing bias significantly.

The bottleneck architecture inhibits the non-useful mapping $g(f(x)) = x$. The efficiency of the network is measured by the reconstruction error produced by feeding unseen data of the dataset.

3.2 Anomaly Detection

Anomaly detection (also outlier detection) refers to a family of techniques that systematically monitor a system in order to identify rare items, events or observations which raise suspicions by differing significantly from the majority of the data [4]. These unusual observations quite often indicate either that our system has reached an anomalous condition or that the newest datapoint is defective, both of which should be examined with care. The use of DAE's has become a staple for anomaly detection tasks. With proper implementation, it is a robust and reliable model. The algorithm is the following:

- Step-1: Examine the data and use an appropriate preprocessing scheme.

- Step-2: Decide which data fall under the normal category.

- Step-3: Train the autoencoder with the normal data only, until convergence.

- Step-4: Feed all the training data into the model (training+validation), in order to obtain the training reconstruction error.

- Step-5: Construct a threshold error according to some statistic derived from the training reconstruction error (a common choice is $T_{rec} = \max_{x \in X_{train}} J(x, \tilde{x})$ [5].

- Step-6: Each sample that has a greater reconstruction error than the threshold, is characterized as an anomaly. With the total reconstruction error profile, we can have a sanity check on the efficiency of our network.

Figure 2. Close up view of the optic fiber sensor attached to a steel reinforcing bar used in the experimental tests.

4 Current implementation

4.1 Main concept

The goal of the current implementation is to examine the possibility of using DAE as a monitoring system in concrete structures. More specifically, the aim is to monitor structures using only measurements of reinforcement strains obtained from a distributed optic fiber sensor, see Fig. 2.

The idea is to model the structure using Finite Element Analysis (FEA), load it until failure with different load cases, gather the features needed for the damage estimation and train the model with the normal states of each loading case.

This approach, while reasonable, does not account for the unavoidable error existing in all data acquired from integrated sensors (imperfect application, device noise etc.) nor the inherent heterogeneity of concrete. Since concrete is commonly modelled as a homogeneous material in FEA, the existing randomness of the material is not well captured by conventional FE, thereby rendering them divergent from the real structure. To tackle this issue, the initial states of the physical element were included into the training data. This step, which acts as a "calibration", can be easily applied to large scale projects and makes the neural network generalize even better.

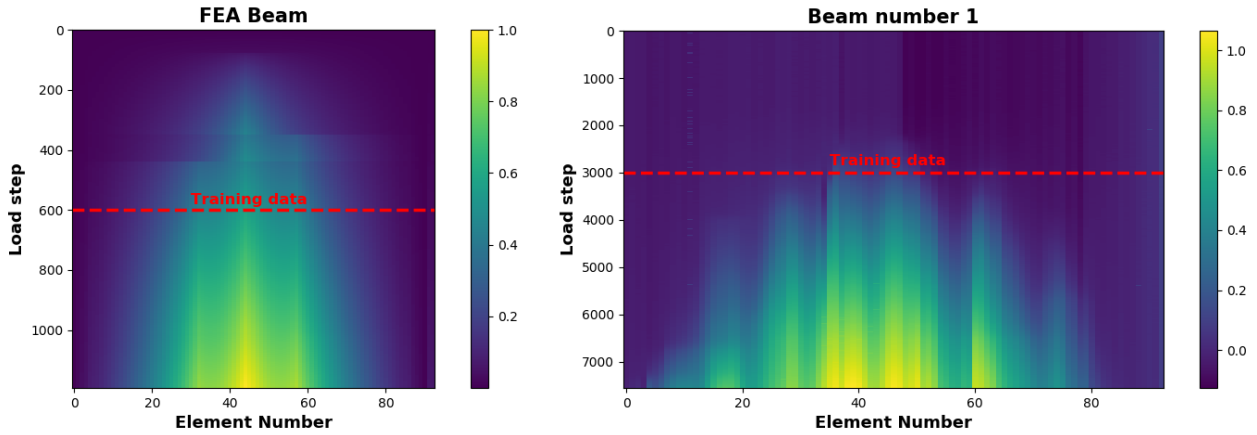*Figure 3. Tested beam after failure.*

Figure 4. On the left, the strain profile of the beam FEA. On the right, the strain profile of one of the tested beams. The red dashed line shows until which state the networks were tested on (see 4.1).

4.2 Set up

The experimental set up consisted of 6 concrete beams with dimensions 90×15×10cm, reinforced with two Ø10mm rebar of B500B steel placed with a concrete cover of 25mm. The concrete had a cube compressive strength of 60 MPa and a tensile splitting strength of 3.5 MPa, both measured at 28 days. For each beam, only one of the longitudinal bars was outfitted with an optic fiber sensor. The signal frequency was 1.25 Hz and spatial resolution was 0.65 mm.

The beams were tested to failure under three-point loading using a displacement-control setup at displacement rate of 1mm/min. Two of the beams were loaded monotonically and four were subjected to cyclic loading. Stirrups were not provided in order to promote shear failure [6], which occurred for all of the beams tested, see Fig.3. Moreover, the monotonic tests were numerically simulated using the commercial FEA software DIANA.

4.3 Training procedure

The training data consisted of the strain states up to until 50% of the total capacity for the results obtained from the FEA, whereas only about the first 15% of the strain states obtained from each experiment were used for training (Fig. 4).

4.4 Preprocessing

The most crucial part of the implementation is the data preprocessing. Feature-wise preprocessing such as z-score standardization and min-max normalization was discarded. The reason is that for the current application all features $F_j, j = 1, ..., n$ have the same physical meaning (strain at some rebar position), thus having the same underlying range while the total range of the values will not be known beforehand in a real application. Consequently, the preprocessing scheme that was tested was a row zero mean centering approach where each feature for each datapoint is transformed as follows:

$$x_{new}^{ij} = x^{ij} - \overline{x}^i, i = 1, ..., m; j = 1, ..., n \quad (3)$$

where m is the number of observations and $\overline{x}^i$ is the mean of all the features in the current observation. In common ML applications, this type of preprocessing is discouraged, since transforming different units with the same device removes a great part of relational information. Nevertheless, for the reasons discussed above, it is an ideal candidate for the current application.

4.5 Damage classification

In bridge condition assessment (BCA) a common practice to assess the level of existing damage in the structure is by creating different criteria or thresholds. Accordingly, three damage thresholds were created by examining the distribution of the strains across the rebar from the analysis: R_0, R_1, R_2

which are the small, significant and hazardous damage thresholds respectively. Thus, we have:

$$R_0 = max\ (Err)\left(1 + \frac{1}{2}\ \widetilde{\omega}\ \right) \qquad (4)$$

$$R_1 = R_0\left(1 + \lambda_1\sqrt{\widetilde{\omega}}\right) \qquad (5)$$

$$R_2 = R_0\left(1 + \lambda_2\sqrt{\widetilde{\omega}}\right) \qquad (6)$$

where:

$$\widetilde{\omega} = \sqrt{\exp(\tilde{\sigma}^2) - 1} \qquad (7)$$

$$\tilde{\sigma} = \sigma(\log\ (Err)) \qquad (8)$$

with σ being the standard deviation, and Err is a vector with all the training reconstruction errors.

Intuitively, $\widetilde{\omega}$ is a custom dispersion, fitting the current application. It must be noted that the above rules were constructed empirically, after trial and error, taking into consideration the dispersity of the reconstruction error resulting from measurements of different noise levels. The coefficient λ_1, λ_2 are multipliers that dictate the sensitivity of the system. In our application, $\lambda_1 = 3$ and $\lambda_2 = 5$. In fact, one could construct arbitrary levels of damage λ_i, depending on the significance of the structure. It must be noted that the proposed thresholds should be used with the training scheme that involves both FEA and the initial states data.
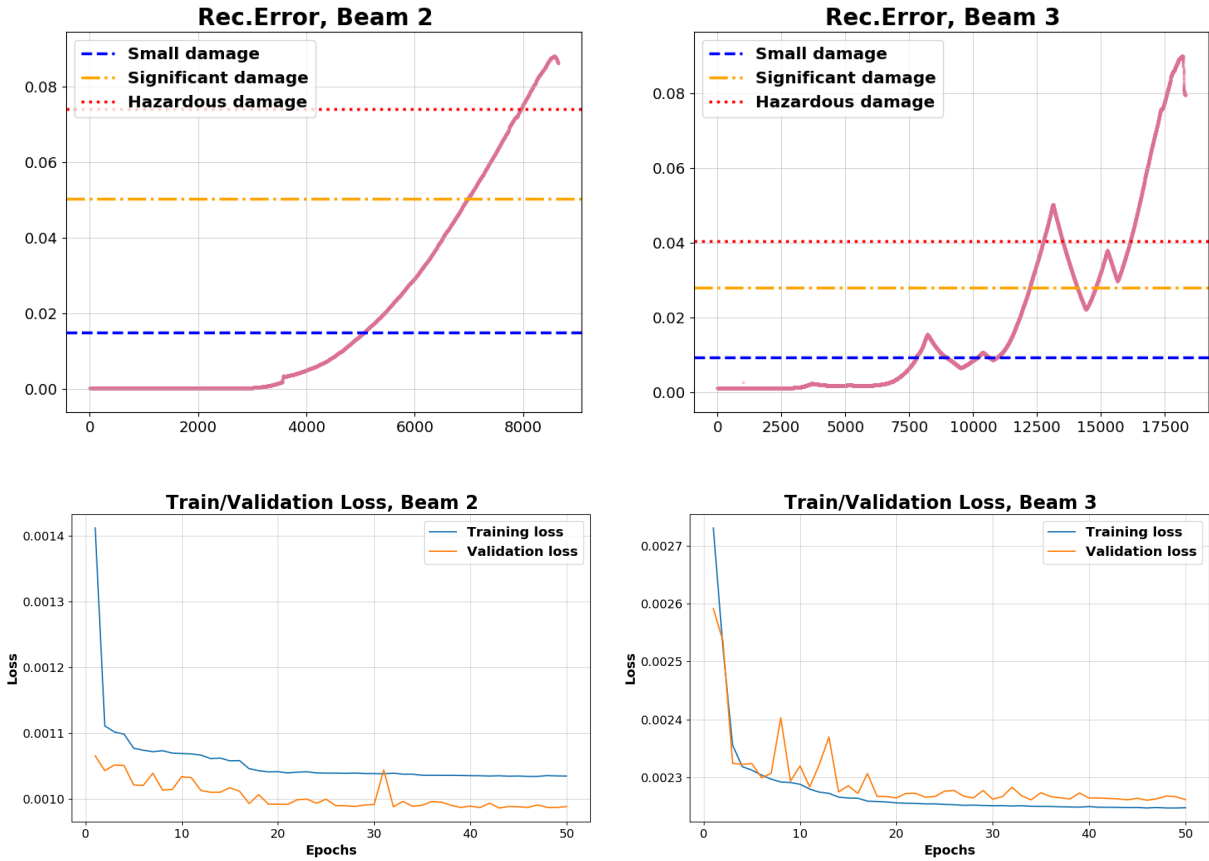

Figure 5. Losses and reconstruction errors for beams 2 and 3, subjected to monotonic and cyclic loading, respectively.

5   Results and Discussion

Fig. 5 presents the results of the DAE for two of the tested beams, one with monotonic and one with

cyclic loading, in terms of reconstruction error and train and validation loss. All the cases studied were tested with various random seeds, in order to verify the robustness of the results. For all the tested beams, the model was able to easily fit the data, which was a more cumbersome task before the

preprocessing scheme described in 4.4. Moreover, in all cases, the damage thresholds classify the damage level successfully, though less efficiently in the heavily noisy beams. It must be noted that the network could capture the states of all the beams, regardless of the type of loading applied.

The Hazardous damage was generally about 70% to 90% of the beam capacity. However, in all cases, the significant damage threshold can efficiently trigger an inspection, thereby avoiding the total collapse of the concrete element.

6   Conclusions

In this research project we have presented the possibility of constructing a DML anomaly detection model that monitors the damage state of a concrete beam, using its strain profile only. The current approach was implemented successfully and could possibly be extended in a plethora of structures. As a future research, more models could be investigated, as well as more complex structural elements with different load patterns. Verifying the effectiveness of the proposed method in various structures is an imperative need before a full-scale application, which could possible revolutionize the field of SHM in bridges.

7   Acknowledgements

8   References

[1] I. Goodfellow, Y. Bengio and A. Courville, Deep Learning, MIT Press, 2016.

[2] 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, no. 2, p. 401–421, 2019.

[3] Y. J. Cha, W. Choi and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks.," *Computer-Aided Civiland Infrastructure Engineering,* vol. 32, no. 5, pp. 361-378, 2017.

[4] A. Zimek and E. Schubert, "Outlier Detection," *Encyclopedia of Database Systems,* 2017.

[5] L. Beggel, P. Michael and B. Bernd, "Robust Anomaly Detection in Images using Adversarial Autoencoders.," *arXiv,* 2019.

[6] J. G. MacGregor and J. K. Wight. "Reinforced Concrete: Mechanics and Design". 6th Edition Harlow: Pearson Education, 2011.