

Development Of A Neural Network Embedding For Quantifying Crack Pattern Similarity In Masonry Structures

Rózsás, Árpád; Slobbe, Arthur; Huizinga, Wyke; Kruithof, Maarten; Giardina, Giorgia

Publication date

2021

Document Version

Final published version

Published in

12th International Conference on Structural Analysis of Historical Constructions

Citation (APA)

Rózsás, Á., Slobbe, A., Huizinga, W., Kruithof, M., & Giardina, G. (2021). Development Of A Neural Network Embedding For Quantifying Crack Pattern Similarity In Masonry Structures. In P. Roca, L. Pelà , & C. Molins (Eds.), *12th International Conference on Structural Analysis of Historical Constructions : SAHC 2021, Online event, 29 Sep - 1 Oct, 2021* (pp. 1905-1916). International Centre for Numerical Methods in Engineering, CIMNE. http://congress.cimne.com/SAHC2020/frontal/doc/Ebook_SAHC2020.pdf

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

DEVELOPMENT OF A NEURAL NETWORK EMBEDDING FOR QUANTIFYING CRACK PATTERN SIMILARITY IN MASONRY STRUCTURES

ÁRPÁD RÓZSÁS^{1*}, ARTHUR SLOBBE¹, WYKE HUIZINGA², MAARTEN
KRUIHOF² AND GIORGIA GIARDINA³

¹Department of Structural Reliability
Netherlands Organization for Applied Scientific Research (TNO)
Stieltjesweg 1, Delft, The Netherlands
e-mail: [arpad.rozsas, arthur.slobbe]@tno.nl (*corresponding author)

²Department of Intelligent Imaging
Netherlands Organization for Applied Scientific Research (TNO)
Oude Waalsdorperweg 63, Den Haag, The Netherlands
email: [wyke.pereboom, maarten.kruihof]@tno.nl

³Department of Civil Engineering and Geosciences
Delft University of Technology
Stevinweg 1, 2628 CN Delft, The Netherlands
email: g.giardina@tudelft.nl

Keywords: Masonry Structure, Crack Patterns, Similarity Measure, Machine Learning, Deep Neural Network

Abstract. *The degree of similarity between damage patterns often correlates with the likelihood of having similar damage causes. Therefore, deciding whether crack patterns are similar is one of the key steps in assessing the conditions of masonry structures. To our knowledge, no literature has been published regarding masonry crack pattern similarity measures that would correlate well with assessment by structural engineers. Hence, currently, similarity assessments are solely performed by experts and require considerable time and effort. Moreover, it is expensive, limited by the availability of experts, and yields only qualitative answers. In this work, we propose an automated approach that has the potential to overcome the above shortcomings and perform comparably with experts. At its core is a deep neural network embedding that can be used to calculate a numerical distance between crack patterns on comparable façades. The embedding is obtained from fitting a deep neural network to perform a classification task; i.e., to predict the crack pattern archetype label from a crack pattern image. The network is fitted to synthetic crack patterns simulated using a statistics-based approach proposed in this work. The simulation process can account for important crack pattern characteristics such as crack location, orientation, and length. The embedding transforms a crack pattern (raster image) into a 64-dimensional real-valued vector space where the closeness between two vectors is calculated as the cosine of their angle. The proposed approach is tested on 2D façades with and without openings, and with synthetic crack patterns that consist of a single crack and multiple cracks.*

1 INTRODUCTION

1.1 Motivation and background

Every year in the Netherlands, house owners report hundreds of issues concerning cracks in their masonry dwellings. Addressing these problems requires identifying the most likely causes of the observed damage. In these structures, similar causes often manifest through similar crack patterns; thus, quantification of crack pattern similarities is instrumental in understanding masonry damage. Currently, the assessment of similarities between masonry crack patterns relies entirely on experts' judgment. This implies several limitations:

- The quality of the evaluation strongly depends on the experience of the assessor, who needs to correlate the observed crack pattern with previously analyzed and recorded ones;
- The assessment is affected by the expert's understanding of global mechanisms and local conditions;
- Currently, it is not clear how the human brain performs the crack pattern comparison and experts have difficulties in verbalizing and formalizing their decision;
- The process is expensive, limited by the availability of experts, and provides only qualitative assessment.

This leads to a general lack of objectivity in crack analysis as different experts can provide substantially different interpretations for similar crack patterns.

Recent developments in deep neural network embedding have the potential to overcome some of these limitations by providing a tool for automated damage classifications and quantitative measures of crack pattern similarities. While significant effort has been devoted to crack detection and characterization [1,2,3], a consultation with structural masonry experts (forensic engineers and computational mechanics experts) identified a lack of automated tools that could be applied to the practice of damage assessment. Furthermore, only few authors have investigated automated methods to connect masonry damage patterns to their most likely causes [4], and the identification of masonry damage through pattern recognition remains a challenge.

1.2 Aim and objectives

This work addresses the need for a systematic approach to damage classification and quantification of crack pattern similarities in masonry structures by using deep neural network embedding. We focus on two major challenges (Figure 1):

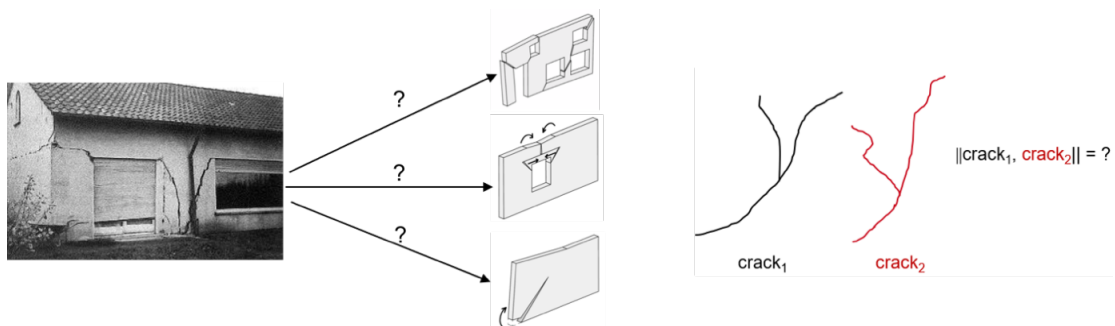


Figure 1: A visual representation of the two major challenges addressed in this paper: crack pattern classification (left) and crack pattern similarity quantification (right).

1. The automated classification of observed crack patterns into damage cause classes (crack pattern archetypes);
2. The quantification of similarities between crack patterns in terms of structural masonry damage in a way that correlates strongly with masonry experts' assessment.

Having an accurate and quantitative similarity measure would enable us to solve both challenges and perform three tasks with high practical relevance:

- Classification of crack patterns into archetypes, such as those proposed by de Vent [5], by calculating the distances between an observed crack pattern and archetype crack patterns. This would allow for a quick and simplified assessment.
- Identification of damage in not directly observed/measured structural components, such as wooden piles, by calculating the distance between an observed crack pattern and simulated (e.g. finite element analysis) crack patterns. This would allow for a more in-depth assessment and understanding the causes of observed damage.
- Automated calibration of finite element models to experts based on matching the similarity between predicted and observed crack patterns.

This paper contributes to develop a fast, cheap, and automated procedure for the assessment of the structural integrity of cracked (damaged) masonry structures by generating synthetic crack patterns using Markov-walks and incorporating expert judgment. The proposed approach is demonstrated on 2D façades with and without an opening, and with crack patterns that consist of a single crack and multiple cracks.

2 METHODOLOGY

The core of our proposed approach is a deep neural network embedding that can be used to calculate a numerical distance between crack patterns on comparable façades. An embedding is a mathematical function that maps a higher dimensional representation to a lower dimensional one while retaining selected essential characteristics of the original representation. In the context of this paper, an embedding maps crack pattern images to a real-valued vector space (i.e., lower dimensional than that needed to represent the images). The challenge is to find such an embedding that produces close-by vector representations for those crack pattern images that are similar by structural engineering standards.

Our working hypothesis is two-fold: first, we assume that crack pattern images and their damage cause/type labels contain sufficient information to find the above described embedding; second, we assume that the embedding can be found by using a deep neural network (a flexible mathematical construct).

The following approach is used to test our hypothesis:

1. Data acquisition: crack pattern images and their damage cause labels.
2. Model definition and fitting: define the mathematical model of the deep neural network and fit its parameters using the data.
3. Model evaluation: comparing the predictions of the fitted model to the damage cause labels for data points not used for the fitting.

Because current deep learning networks are data intensive, we would need at least a few hundreds of crack patterns and corresponding damage causes to test our hypothesis. The highest quality data are real-world cases but those are expensive and time consuming to collect, especially as reliable damage cause labels are needed. Moreover, in real-world cases, multiple damage causes are often present and are difficult to separate even by experts. An alternative

route is to generate synthetic data. This can be done using computational physics to solve the mathematical representation of physical systems (labeled as physics-based approach in this paper), e.g. finite element analysis to solve partial differential equations. Fortunately, our understanding of masonry structures is sufficient to perform crack analysis of them with good accuracy. Still, this approach is relatively time consuming due to the large number and large variety of structures and damage causes needed to be implemented with parameterized models and to perform the analyses and post-processed. Therefore, a cheaper and simpler synthetic data generating approach is used first, which is labeled as a statistics-based approach in this paper and based on Markov-walks guided by rules that encode expert knowledge. Here, a statistics-based synthetic data generating process is used to find the embedding and evaluate its performance. If successful, then it would encourage the acquisition of more realistic but more expensive data sources.

Because the statistics-based approach generates synthetic data and has a random component, an arbitrary number of crack patterns can be simulated once the algorithm is available. Compared with the physics-based simulation approach, it is expected to generate synthetic data an order of magnitude faster; however, the quality of the data is poorer, i.e. less representative of reality.

An approach is proposed and implemented to generate statistical crack patterns resulting from a total of twelve damage causes and for five different façades with and without openings (Section 3). A convolutional neural network (CNN) is fitted to half of the data with the task of classifying the crack patterns into damage cause classes. The first part of the fitted CNN is used as a 64-dimensional embedding of the crack pattern images where the closeness between two vectors is calculated based on the angle between them (Section 4-5). The closeness is interpreted as the similarity between crack pattern images and compared with our experts' judgment.

3 GENERATION OF SYNTHETIC CRACK PATTERNS

A statistics-based simulation approach is developed in order to generate synthetic crack patterns for a wide range of masonry façades and, hence, quickly test the potential of various crack pattern similarity measures. This approach includes an algorithm that generates lines in a pre-defined bounding box (i.e. the geometry of a façade) by means of Markov-walks. The algorithm is generalized so that it can handle a parametric input of the façade and the cracks. This allows control over the dimensions of the façade, and the number, dimensions and position(s) of the openings (doors and windows). It also enables control of the number of cracks, the crack initiation point(s), the crack angle(s), the crack length(s), and how jagged a crack is. Furthermore, structural engineering considerations are imposed in order to increase the resemblance to reality. Though the statistics-based simulation approach is not based on first physical principles, it is able to account for important crack pattern characteristics such as crack location, orientation, and size.

Subsection 3.1 explains the details of the crack pattern generation in masonry wall panels without openings and with a single crack. Subsection 3.2 presents a more realistic case, where crack patterns are generated in longitudinal masonry façades that resemble typical existing dwellings in the Netherlands. These patterns are based on frequently observed damage.

3.1 Case 1: masonry wall panels without openings

To support structural damage diagnosis in masonry structures, de Vent [5] introduced 60 types of crack patterns along with their possible damage causes. In case 1, eight of these are selected as listed in Figure 2. For consistency, the numerical IDs from [5] are used here as well when referring to crack pattern archetypes. The eight archetypes share the following characteristics:

- a single wall panel is considered with a length over height ratio of 1.5;
- a single major crack characterizes the pattern;
- no openings are present;
- a crack is assumed to have unit and constant width.

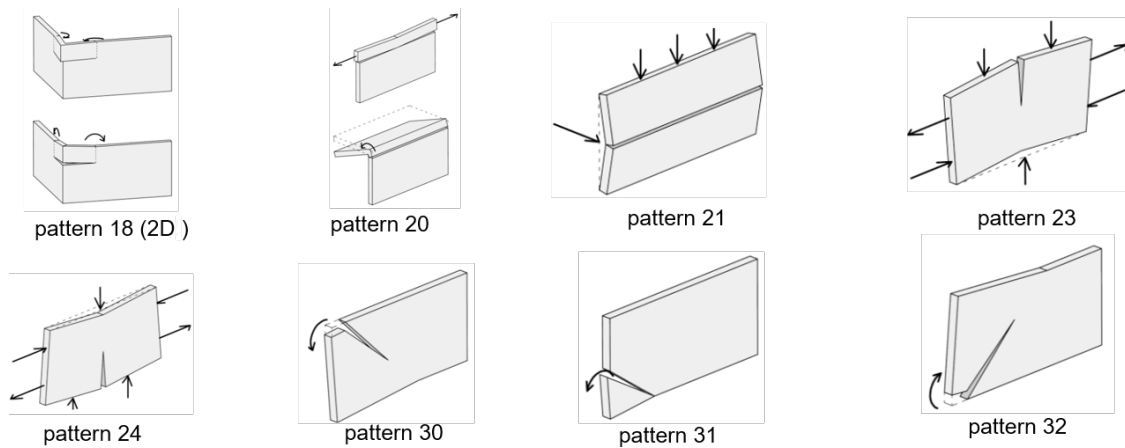


Figure 2: Illustrations of the selected eight crack pattern archetypes [5].

Figure 3 shows the crack parametrization in the wall panels for a normalized (unit square) wall. Each crack is defined by the following parameters: the coordinates of the crack initiation point (x_0, y_0) , the crack angle (α) , the crack length (l_{crack}) , and jaggedness of the crack (σ_{crack}) . The first three parameters are randomly and uniformly sampled from an interval that is specified for each crack pattern archetype. For illustrative purposes, the input values of these parameters for the crack pattern archetypes 23, 30, and 31 are summarized in Table 1. In the simulation, the panel is discretized into disjoint rectangles and a crack can propagate only through the cells of this grid, one cell at a time/step. A crack starts from the cell at (x_0, y_0) , a random step is taken in either vertical or horizontal direction in a way that the expected global crack angle is α . The average absolute deviation from the expected direction is controlled by σ_{crack} , i.e., 0.5: highest deviation, 0: no deviation. Illustrative realizations of this simulation are presented in Figure 4, one for each case 1 pattern archetype. The simulated patterns illustrate that the proposed statistical approach can simulate realistic-looking and unique (due to the random component) crack patterns. In principle, the current implementation of the algorithm allows for the generation of an unlimited number of unique realizations for the eight selected crack pattern archetypes.

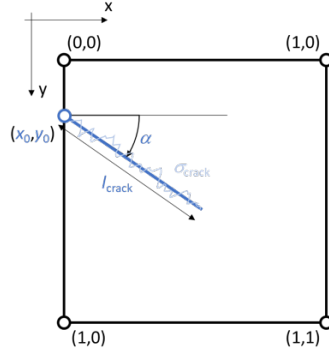


Figure 3: Crack parametrization in the wall panels for a normalized (unit square) wall.

Table 1: Input values in the unit square for the crack pattern archetypes 23, 30 and 31. For explanation of the parameters, see Figure 3.

Pattern ID	x_0, y_0^*	α [deg] *	l_{crack}^*	σ_{crack}
23	[0.45, 0.55], [0, 0]	[87, 93]	[0.4, 0.6]	0.5
30	[0.05, 0.20], [0, 0]	[20, 70]	[0.3, 0.5]	0.5
31	[0, 0], [0.60, 0.80]	[20, 70]	2^\dagger	0.5

* the lower and upper bounds of the uniform distribution

† to ensure that the crack goes through the entire wall

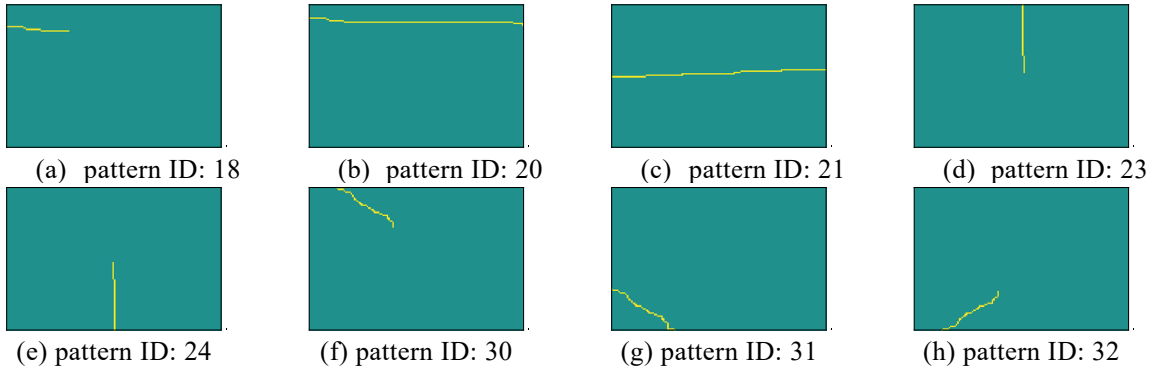


Figure 4: One random crack pattern realization for each damage cause (pattern id) for case 1 (façade with openings). See Figure 2 for the architectural representation of the matching crack patterns.

3.2 Case 2: masonry façades with openings

Every year in the Netherlands, house owners report hundreds of issues concerning cracks that appear in their masonry dwellings. In general, there are two main causes for these cracks: (i) uneven settlements; and (ii) constrained deformation due to temperature effects. The second cause mainly occurs in older masonry dwellings, where dilatation joints are poor or even not present, due to the heating of the roof or when the foundation constrains the façade.

In case 2, random crack pattern are generated in the longitudinal masonry façade of typical Dutch dwellings. Four different geometries are evaluated (Figure 5). One of them is detailed in

Figure 6, showing one door opening and two window openings in the façade. Four crack pattern archetypes are considered based on the following damage causes:

1. Cracking due to uneven settlements with large settlements at the left side (pattern ID: 101);
2. Cracking due to uneven settlements with large settlements at the right side (pattern ID: 102);
3. Cracking due to uneven settlements with large settlements at the middle (pattern ID: 103);
4. Cracking due to high temperature of the roof (pattern ID: 201).

Damage causes 1 and 3 typically result in crack pattern archetypes with a single major crack, whereas damage causes 2 and 4 result in crack pattern archetypes with multiple cracks (we consider two cracks). As in case 1, the cracks are assumed to have unit and constant widths.

The crack pattern realizations are made in a similar fashion as for case 1. For illustrative purposes, parametrization of the two cracks in the façade subjected to damage cause 2 is shown in Figure 7. The parameters of the two cracks are distinguished by the numbers "1" and "2" in their subscripts. Table 2 provides the input values of the crack parameters. The specified intervals of the crack initiation points are relative to the red dots that are indicated in the corners of the right window. The specified intervals of the crack lengths are relative to the length of a fully developed crack (i.e. a crack that has been propagated to the edge of the façade).

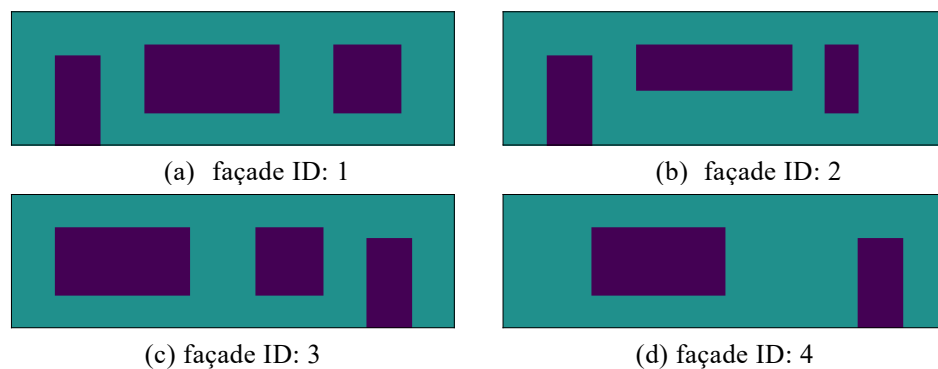


Figure 5: Overview of the considered façades for case 2. All façades have the same bounding rectangle dimensions. See also Figure 6.

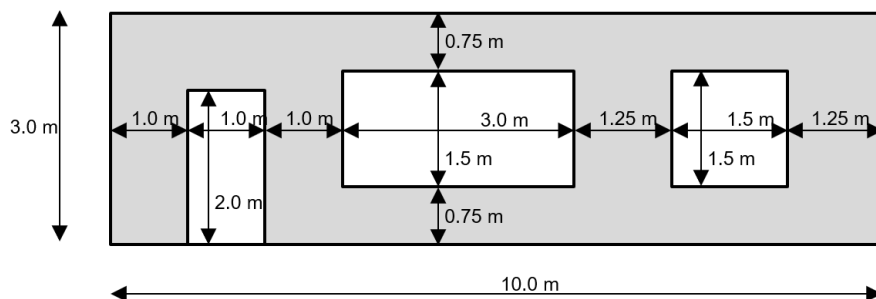


Figure 6: The geometry of the longitudinal façade of a typical Dutch masonry dwelling.

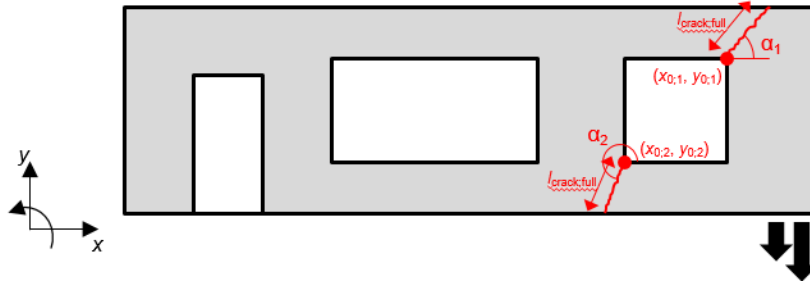


Figure 7: Crack parametrization in the longitudinal façade of a typical Dutch masonry dwelling for damage cause 2.

Table 2: Input values for crack pattern ID 102. For explanation of the parameters, see Figure 7.

$x_{0;1}, y_{0;1}$ [cm] *	$[-80, 0], [-20, 0]$ from indicated corner in Figure 7
$x_{0;2}, y_{0;2}$ [cm] *	$[0, 80], [0, 0]$ from indicated corner in Figure 7
α_1 [deg] *	$[0, 70]$
α_2 [deg] *	$[225, 270]$
$l_{crack;1}$ *	$[0.4, 1.0] \cdot l_{crack;full}^\dagger$
$l_{crack;2}$ *	$[0.4, 1.0] \cdot l_{crack;full}^\dagger$
σ_{crack}	0.5

* the lower and upper bounds of the uniform distribution

† to ensure that the crack goes through the entire wall

Figure 8 shows one illustrative realization for each damage cause. The four images demonstrate that the algorithm is able to generate reasonable crack patterns from a structural engineering point of view, though no physics is explicitly included.

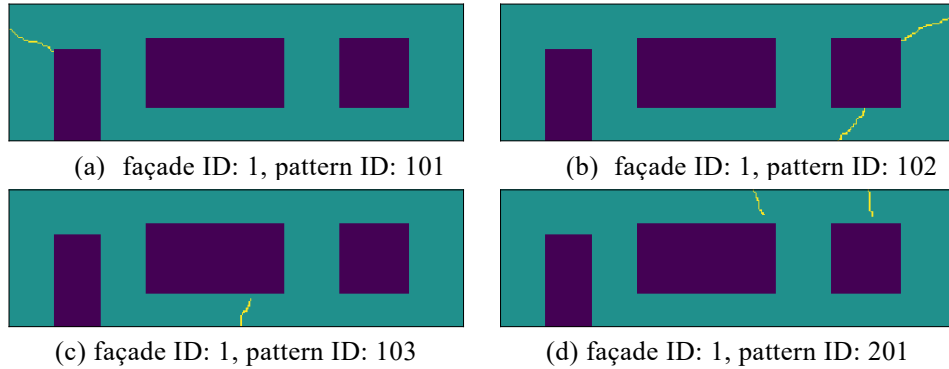


Figure 8: One random crack pattern realization for each damage cause (pattern ID) for façade 1 of case 2 (façades with openings).

4 DEEP NEURAL NETWORK

For the analysis of the crack pattern similarity, we used an artificial neural network (ANN) [6]. An ANN is a computational model consisting of nodes that are organized into layers. Each node in a certain layer transmits a weighted signal to the other units in the next layer. A unit in the next layer sums these signals, applies a non-linear function, and sends the result to the units

in the next layer. See Figure 9 for a schematic representation of an ANN.

The model weights are free parameters that are optimized for a certain task. Examples of such tasks are image classification, object detection, or language translation [7]. The performance of the model on a task and dataset is mathematically expressed in an objective function. Under a given dataset, optimization of this objective function with respect to the free parameters is called “fitting” of the model (or “training” in the machine learning community). In our case, we fit the network to classify simulated crack patterns into known crack pattern archetypes. The input of our network is a simulated image of a crack pattern and the output of the model is the class of the crack pattern.

We used a convolutional neural network (CNN) to classify the crack patterns [8]. A CNN takes an image as its input and applies convolutional filters to this image. The values of these convolutional filters are also free model parameters. The filter sizes are hyperparameters and should be set by the user.

The design of the network we used is shown in Figure 10 and consists of eight layers. This model contains $\approx 3 \cdot 10^6$ free parameters. We constructed the network in such a way that the input images are embedded into a 64-dimensional space by the first seven layers. The embedding is used by the last layer to predict the crack class. We used the output of this embedding to calculate the cosine similarity of the different crack images, which we used as a proxy for crack pattern similarity. In other words, the closer the images are in the embedding space, the more similar the crack patterns with respect to their class.

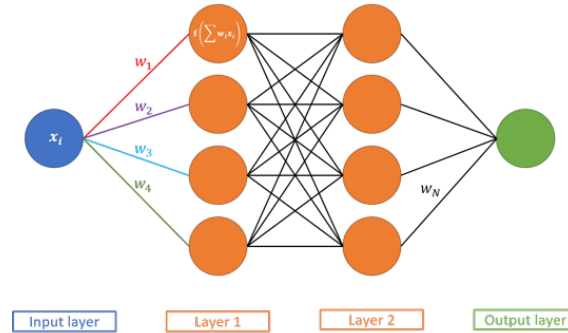


Figure 9: Schematic representation of an ANN. The computational units are represented by the coloured circles. Each of the N connections has a weight w_i and each unit applies a non-linear function to the sum of its weighted input(s): $f(\sum w_i x_i)$.

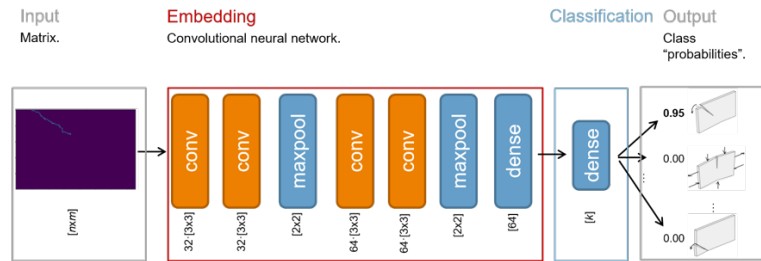


Figure 10: Overview of the neural network used in this work. conv: convolutional layer; maxpool: maxpool layer; dense: fully connected layer.

5 CLASSIFICATION RESULTS AND CRACK PATTERN SIMILARITY

As explained in Section 4, the model is fitted to classify crack pattern images into crack pattern archetypes. We used a randomly chosen subset of the data for fitting and the rest is reserved for testing. The experiments were performed on two data sets: a single façade without openings (case 1) and four façades with openings (case 2). The two sections below describe the results.

5.1 Results for case 1: masonry wall panels without openings

For this case, a total of 4,000 cracks were simulated, with 500 images per class. Here, 2,000 images were used to fit the model and the remaining 2,000 were used for testing. The classification accuracy on the test set is $> 99\%$, which means that the model can almost perfectly classify the eight simulated patterns in their corresponding classes.

We visualize the 64-dimensional embedding space in 2D with a method called t-distributed Stochastic Neighbor Embedding (t-SNE) [9]. This is a widely used method to visualize high dimensional data on a low dimensional subspace. The t-SNE plot in Figure 11 shows that the classes are nicely clustered and perfectly separable in the t-SNE embedding and, therefore, are also separable in the 64-dimensional embedding, which is expected given the high classification accuracy.

We computed the pairwise similarity between all 2,000 simulated crack patterns in the test set with the cosine similarity of the 64-dimensional vectors. This similarity is visualized as a lower-triangular matrix and shown in Figure 12. The rows and columns are sorted on class labels and, as expected, a high similarity is found in blocks at the diagonal of this matrix. A low similarity is found outside these blocks, i.e. between crack patterns of different classes.

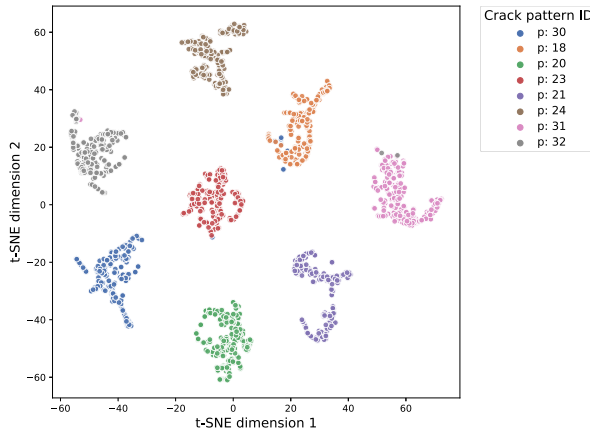


Figure 11: t-SNE scatter plot of the 64-dimensional embedding of case 1 test set crack patterns (façade without openings).

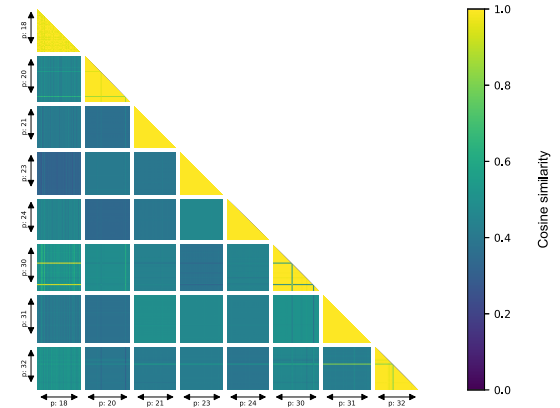


Figure 12: Pairwise cosine similarity of the test set of case 1 (façade without openings).

5.2 Results for case 2: 2D longitudinal façade of typical Dutch dwellings

For this case, a total of 8,000 cracks were simulated: 2,000 images per class, 500 for each unique façade-crack pattern archetype pair. Here, 4,000 images were used to fit the model and 4000 for testing.

Similar to the crack patterns of a masonry wall panel without openings, the classification

accuracy on the crack patterns in the test set of case 2 is >99%. With these data, the model can also almost perfectly classify the eight simulated patterns in their corresponding classes.

The t-SNE plot in Figure 13 shows that also these classes are perfectly separable. Similar to the crack patterns of a masonry wall panel without openings, the pairwise similarity is high for crack patterns of the same class and low for crack patterns from different classes (Figure 14).

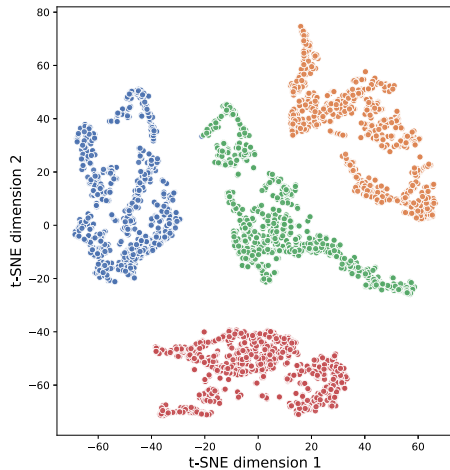


Figure 13: t-SNE scatter plot of the 64-dimensional embedding of case 2 test set crack patterns (façades with openings).

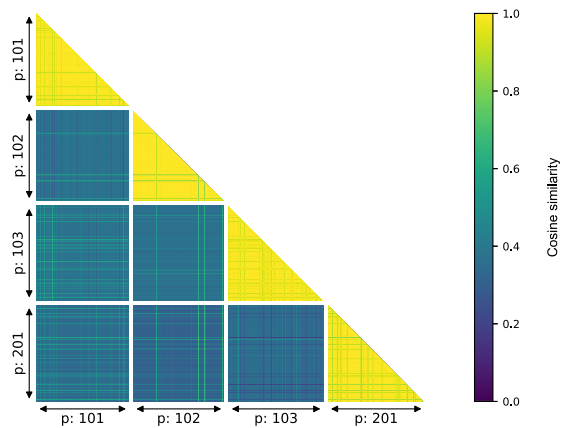


Figure 14: Pairwise cosine similarity of the test set of case 2 (façades with openings). Note that all façades with the same crack pattern ID are assigned to the same group.

6 CONCLUSIONS

This paper addressed the lack of quantitative procedures for the assessment of masonry structure conditions by proposing an automated tool to measure the similarity between crack patterns. We used a deep neural network embedding to calculate a numerical distance between different crack patterns of similar façades. Through Markov-walks guided by expert knowledge, we generated synthetic data that were used to find an embedding and evaluate its performance. We considered twelve damage causes and five façade models, with and without openings, and with single and multiple cracks. The fitted neural network performs exceptionally well in classifying crack patterns that were not used for the fitting: >99% classification accuracy for all considered test cases. The similarity measures calculated using the fitted embedding are in an almost perfect agreement with the experts for between crack pattern class comparisons. A worse performance is observed for within crack pattern class comparisons, i.e. the embedding predicts a very similar low difference for all crack pattern pairs within the same class, while more diversity in the similarity values was expected based on the generated patterns. This is likely due to fact that no information regarding within class similarities are used for fitting the neural network. While the relatively simple simulation process may skew the evaluation of the neural network performance, these results are an encouraging first step towards the development of an automated assessment of damaged masonry structures.

Acknowledgements. We are grateful to Huibert Borsje (TNO) for the valuable discussions on damaged masonry structures and Wessel de Zeeuw (TNO) for his contribution to implementing the computer code to perform the presented analyses.

REFERENCES

- [1] Silva, W. and Schwerz de Lucena, D. Concrete cracks detection based on deep learning image classification. In: The Eighteenth International Conference of Experimental Mechanics, Vol. II (2018), pp. 5387.
- [2] Chaiyasarn, K., Khan, W., Sharma, M., Brackenbury, D., and DeJong, M. Crack detection in masonry structures using convolutional neural networks and support vector machines. In: Proceedings of the 35th ISARC, Berlin, Germany (2018), pp.118-125.
- [3] Dung, C. V. and Anh, L. D. Autonomous concrete crack detection using deep fully convolutional neural network. *Automation in Construction* (2019) **99**:52-58.
- [4] Napolitano, R. and Glisic, B. Methodology for diagnosing crack patterns in masonry structures using photogrammetry and distinct element modeling. *Engineering Structures* (2019) **181**:519-528.
- [5] de Vent, I. *Prototype of a diagnostic decision support tool for structural damage in masonry*. PhD thesis, Delft University of Technology (2011).
- [6] Wang, S.-C. Artificial Neural Network. In: *Interdisciplinary Computing in Java Programming*, Springer US, Boston,MA, (2003), pp. 81-100.
- [7] Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., and Alsaadi, F. E. A survey of deep neural network architectures and their applications. *Neurocomputing*, (2017) **234**:11-26.
- [8] Arel, I., Rose, D. C., and Karnowski, T. P. Deep machine learning - a new frontier in artificial intelligence research. In: *IEEE Computational Intelligence Magazine*, (2010) **5**(4):13–18.
- [9] van der Maaten, L. and Hinton, G. Visualizing data using t-SNE. *Journal of Machine Learning Research*, (2008) **9**:2579–2605.