POLITECNICO DI TORINO
Repository ISTITUZIONALE

Compact Convolutional Transformer Fourier analysis for GPR tunnels assessment

Original
Compact Convolutional Transformer Fourier analysis for GPR tunnels assessment / Melchiorre, J., Rosso, M.M., Cirrincione, G., Marano, G.C.. - (2023). (International Conference on Control, Automation and Diagnosis (ICCAD) Roma, Italia 10-12 May 2023) [10.1109/ICCAD57653.2023.10152455].

Availability:
This version is available at: 11583/2980534 since: 2023-07-19T19:46:01Z

Publisher:
IEEE

Published
DOI:10.1109/ICCAD57653.2023.10152455

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IEEE postprint/Author's Accepted Manuscript

©2023 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

27 November 2026

Compact Convolutional Transformer Fourier analysis for GPR tunnels assessment

Jonathan Melchiorre*, Marco Martino Rosso*, Giansalvo Cirrincione†, and Giuseppe Carlo Marano*‡
*Department of Structural, Geotechnical and Building Engineering, Politecnico di Torino, Turin, Italy.
Email: jonathan.melchiorre@polito.it, marco.rosso@polito.it, giuseppe.marano@polito.it
†Senior Member, IEEE, Lab. LTI, University of Picardie Jules Verne (UPJV), Amiens, France.
Email: giansalvo.cirrincione@u-picardie.fr
‡College of Civil Engineering, Fuzhou University, Fuzhou 350108, China.

***Abstract*—Road tunnels are critical infrastructures that require regular maintenance and inspection to ensure the safety of users. However, traditional inspection methods can be costly, time-consuming, and may not provide an objective assessment of the tunnel's condition. To overcome these challenges, researchers have recently developed newly promising approaches that combine tunnel's indirect testing, e.g. ground-penetrating radar (GPR), with the potentialities of artificial intelligence (AI). This innovative technique allows for the automatic detection and classification of defects in tunnel linings, providing a faster, more efficient, and more reliable method for assessing the health of road tunnels.**

This study explores the application of one of the state-of-the-art deep learning models, i.e. the compact convolution transformer (CCT), to classify defects in GPR images of tunnel linings. The dataset used in this study consists of inspections of tunnels built between the 1960s and 1980s. Two CCT models were trained on filtered and unfiltered datasets and then compared to assess the effects of noise on the identification task.

I. INTRODUCTION

Civil engineering infrastructure heritage is currently facing safety concerns due to aging and degradation [1], specifically bridges [2] and tunnels [3]. As a result, a growing interest and research efforts were devoted to structural health monitoring (SHM), direct testing, and indirect ones. Also acknowledged as non-destructive testing, the latter has gained popularity due to their increased reliability nowadays, less invasive and fast investigations, which can result in less maintenance and rehabilitation costs [4]. Non-destructive testing techniques (NDT) are commonly used in SHM to monitor structural integrity without causing damage to the structure [5]. These techniques include ultrasonic testing, acoustic emission, radiography, thermography, and visual inspections, among others. They have proven to be effective in detecting damage at early stages and monitoring the progress of degradation over time. To improve the efficiency and productivity of SHM surveys with the use of NDT and increase the objectivity of results, several researchers have developed automated procedures [6], [7].

In this manuscript, a non-destructive structural testing technique based on ground penetrating radar (GPR) has been explored [8]. The technique relies on the emission of electromagnetic wave impulses into the material under study and the collection of reflected back signals by a receiver antenna. This procedure allows for an in-depth inspection of the material. The output of the GPR is an image that can reveal the presence of anomalies, defects, fractures, and other features that may not be easily identifiable through direct visual inspection. Within the traditional GPR-based tunnels' investigations experienced personnel is required to identify and classify the possible defects detected in GPR images [9]. Their interpretation may be complex, and requires a thorough understanding of the underlying physics and the properties of the materials being tested [10]. Additionally, GPR data can be affected by a range of factors, including the moisture content of the material, the presence of metallic objects, and the geometry of the structure being tested.

Machine learning (ML), particularly deep learning (DL), has recently become a powerful tool for automatic image processing and classification tasks [11]. The authors herein presents an application of a recent state-of-the-art DL technique for defect classification in tunnel lining [12]. Specifically, this study inspected tunnels built between the 1960s and 1980s by acquiring data with GPR. The acquired data were then filtered using a two-dimensional Fourier transform technique and used for training a compact convolution transformer (CCT) model. This model was chosen due to its compact size, which offers advantages in terms of portability and the ability to be installed on smaller capacity and computing devices. The model was trained to perform hierarchical multi-level defect classification of road tunnels, considering 14 different classes that represent various types of damage that may be present in tunnel linings. Thus, the proposed model was trained on two different datasets. The first dataset was composed of images acquired directly from the GPR, while the second dataset consisted of the same images preprocessed using a bidimensional Fourier transform filtering technique. The use of the bidimensional Fourier transform technique makes virtually possible reducing background noise and increasing the contrast in the images. The two resulting models were then compared to determine the effectiveness of the proposed filtering techniques.

This study is organized as follows. In section II, we present the techniques used for image preprocessing, which involve the use of bidimensional Fourier transform. In section III, we report on the application of GPR for monitoring tunnel lining. In section IV, we briefly describe the compact convolution

transformer model that we used for the defect classification task. In section V, we report the results of applying the presented models on the dataset of real tunnel images.

II. BIDIMENSIONAL FOURIER PRE-PROCESSING

Bidimensional Fourier preprocessing is a technique used in digital image processing to enhance images by removing noise and improving contrast [13]. It involves taking the Fourier transform of an image and filtering out undesired frequencies before performing an inverse Fourier transform to obtain the processed image. Every image can be seen as a bidimensional signal and expressed as the superposition of harmonics components. Considering a digital image of size $M \times N$ pixels as $f(x, y)$ for $x = 0, ...M-1$ and $y = 0, N-1$, the bidimensional discrete Fourier transform (2D-DFT) can be used to retrieve the corresponding frequencies (u, v) according to the (x, y) directions [14]:

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-i2\pi\left(\frac{u}{M}x + \frac{v}{N}y\right)} \quad (1)$$

Once the image has been expressed in the frequency domain using the 2D discrete Fourier transform, the convolution theorem can be used to filter the signal using a kernel mask function $h(x, y)$:

$$F^{-1}[H(u,v)F(u,v)] = F^{-1}[H(u,v)] * F^{-1}[F(u,v)] \quad (2)$$

where $F(u, v)$ is the Fourier transform of the input image, $H(u, v)$ is the Fourier transform of the kernel mask function, F^{-1} denotes the inverse Fourier transform, and $*$ denotes the 2D convolution operation. The filtered image can then be obtained by taking the inverse Fourier transform of the product $H(u, v)F(u, v)$ in the frequency domain [15].

This operation allows the application of a filter kernel to the image in the frequency domain, which can be used to selectively enhance or suppress specific frequency components in the image. This technique is commonly used for noise reduction, edge detection, and image enhancement, and can be a powerful tool in digital image processing. This type of preprocessing compresses data and information for each image without compromising the geometric nature of the GPR profile images. In addition, the bidimensional Fourier preprocessing preserved the vertical and horizontal patterns in the input image by retaining them in the dominant frequency components in the Fourier domain. This resulted in the compression of information related to periodic components, which are typically present in GPR profiles in the depth direction, and the removal of non-periodic noise effects.

III. ROAD TUNNEL ASSESSMENT WITH GPR

GPR is a non-destructive testing method used for defect characterization in engineering materials. GPR is a geophysical technique that uses an antenna to transmit high-frequency electromagnetic waves into a material. The antenna typically has a frequency range of 10 to 2600MHz. The dielectric properties of the material affect the propagation of the waves, which are then collected by a receiver antenna to produce an image that represents the in-depth of the medium [8]. This non-destructive testing method can detect anomalies, defects, fractures, and other features that may not be visible, making it useful for damage detection and classification. In the case of road tunnels, GPR is commonly used to inspect the linings for defects that could pose a safety risk to drivers, particularly in the top crown area which includes the two lateral haunches. However, this inspection method typically requires skilled and experienced personnel to manually interpret the GPR images and identify different types of defects, which can be time-consuming and subjective. In this paper, an automatic procedure to classify the road tunnel lining defects through the use of machine learning techniques on images provided by GPR testing is presented. The technique involves the adoption of a deep learning model called Compact Convolution Transformer (CCT) to perform a binary classification task. The classification has been accomplished in a hierarchical multi-level procedure, as reported in Figure 1. This process enabled a more thorough classification of the anomalies. The total number of available samples for each level progressively reduces as following the tree in Figure 1 in depth. Additionally, because each class offers an uneven amount of data, the class with the fewest samples was compelled to adopt a balanced strategy in order to build a strong classification model.

To categorize the samples, 14 classes have been identified (Figure 1), labeled Ci ($i = 1$ to 14), and divided into six main levels. Level 1 includes folders C1 and C2, and is focused on identifying completely healthy samples (C1) versus those with generic flaws (C2). Level 2a aims to identify healthy samples (C3) that may contain reinforcement bars (C4). On the other hand, Level 2b focuses on classifying potential defects using a generic warning mix (C5), which can be more challenging to distinguish from specific flaws (C6). Class C6 is examined more closely in Level 3 to identify cracks (C7) and differentiate them from other types of flaws (C8). Level 4 focuses on characterizing anomalies in the concrete linings (C9) and distinguishing them from void defects (C10). In Level 5, a more detailed classification is performed to differentiate simple voids (C11) from other types (C12). Finally, in Level 6, the latter class is further analyzed to categorize excavation problems (C13) and issues with concrete-rock detachments (C14).

In the current study, a dataset of tunnel lining defects was used, consisting of structures dating from the 1960s to the 1980s. After human experts recognized the defects, the GPR profiles were cut at a constant step of 5.00m along the horizontal axis to obtain sample images for classification. These samples were labeled based on the human experts' defect recognition phase. To ensure that defects were not split between two images, the cutting step was occasionally manually adjusted to provide clearer samples for classification.

IV. COMPACT CONVOLUTION TRANSFORMER

To address the tunnel defects classification problem, the authors focused on a state-of-art DL techniques acknowledged

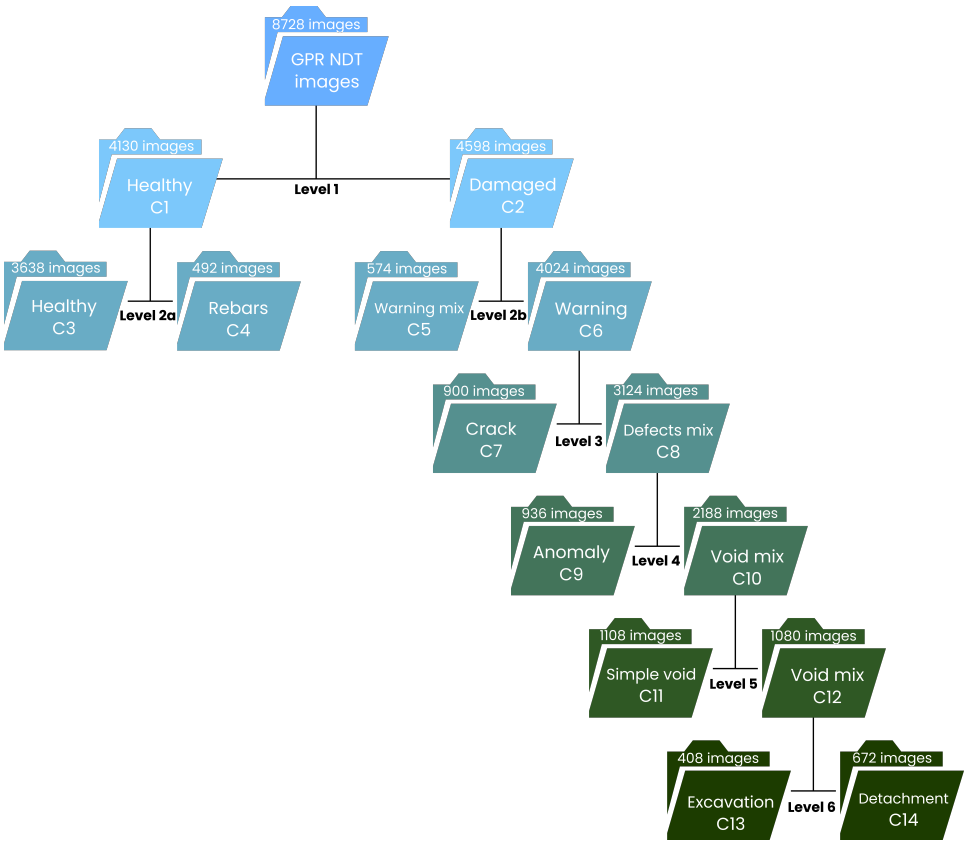


Fig. 1. GPR data classification with CCT model.

as neural transformers [16]. In order to compare the GPR defects classification outcomes with and without bi-dimensional Fourier pre-processing, the last DL model adopted in the current work is the Compact Convolutional Transformer (CCT). This model was developed in [17], starting from the baseline model of Vision Transformer (ViT) [18] and introducing some substantial modifications. In literature, the transformers deserved the title of data-hungry models [17], [19] because of their main drawback of almost prohibitive computational effort for training from scratch in applications with a limited dataset size [20]. Therefore, the CCT model was developed to alleviate this issue [17]. In CCT, a convolutional tokenization procedure has been set in place of the ViT patching method. In this way, it is no more strictly required to set the specific image resolution size and an appropriate and arbitrary fixed number of patches. Adopting the right number of conventional convolutional blocks, it is possible to realize an embedding input to properly feed the subsequent transformer encoder blocks. This procedure is expected to provide a more efficient compact image embedding representation [17]. Indeed, on one side, this new embedding process reflects the desirable properties of CNNs, i.e. hierarchical feature extractor with local information preservation, weight sharing, and efficiency. On the other hand, the CTT permits the adoption of the powerful framework of the self-attention mechanism to process long-range and global dependencies. Another key difference between CCT and ViT is related to the dropping of the BERT-based class token while promoting a sequence pooling approach. This latter method leverages the information of the entire scored output sequence and it does not refer only to a single class token as in the ViT. The sequence pooling method helped to enhance the efficiency of the final dense layers accountable for the classification task [17]. The current implementation relies on the CCT base model provided in the Keras python environment [21], [22]. The considered CCT model was composed of two convolutional layers and $N = 2$ number of transformer encoder blocks. Figure 2 illustrates the empirical trial-and-error hyperparameters set adopted to train the current CCT model.

V. FOURIER RESULTS COMPARISONS

Concerning the CCT model described in section IV, also on this occasion, the authors have split the dataset with a proportion of 90% for the training set and 10% for the test set. The authors trained the CCT using 20 epochs and the mini-batch size of 16. Despite the ViT model is normally employed as a pre-trained model, training the CCT model from scratch represents a feasible solution considering the total learnable parameters 407,107 of the current CCT model with respect to training the current ViT model from scratch (305,413,122 total learnable parameters in the case of training from scratch). The authors employed the base model of the CCT provided

TABLE I

CONFUSION MATRICES AND CLASSIFICATION METRICS FOR CCT MODEL TRAINED WITH RAW IMAGE DATA.

Level 1	Predicted		**Accuracy**			**85.22%**		
True	C1	C2	Class	Nr img/class	Test support	Precision	Recall	f1-score
C1	324	77	C1	408	401	86.17%	80.80%	83.40%
C2	52	420	C2	672	472	84.51%	88.98%	86.69%
Level 2a	Predicted		**Accuracy**			**92.74%**		
True	C3	C4	Class	Nr img/class	Test support	Precision	Recall	f1-score
C3	355	4	C3	408	359	93.18%	98.89%	95.95%
C4	26	28	C4	672	54	87.50%	51.85%	65.12%
Level 2b	Predicted		**Accuracy**			**89.13%**		
True	C5	C6	Class	Nr img/class	Test support	Precision	Recall	f1-score
C5	35	21	C5	408	56	54.69%	62.50%	58.33%
C6	29	375	C6	672	404	94.70%	92.82%	93.75%
Level 3	Predicted		**Accuracy**			**87.59%**		
True	C7	C8	Class	Nr img/class	Test support	Precision	Recall	f1-score
C7	64	31	C7	408	95	77.11%	67.37%	71.91%
C8	19	289	C8	672	308	90.31%	93.83%	92.04%
Level 4	Predicted		**Accuracy**			**85.62%**		
True	C9	C10	Class	Nr img/class	Test support	Precision	Recall	f1-score
C9	72	24	C9	408	96	77.42%	75.00%	76.19%
C10	21	196	C10	672	217	89.09%	90.32%	89.70%
Level 5	Predicted		**Accuracy**			**75.34%**		
True	C11	C12	Class	Nr img/class	Test support	Precision	Recall	f1-score
C11	91	24	C11	408	115	75.21%	79.13%	77.12%
C12	30	74	C12	672	104	75.51%	71.15%	73.27%
Level 6	Predicted		**Accuracy**			**76.85%**		
True	C13	C14	Class	Nr img/class	Test support	Precision	Recall	f1-score
C13	32	21	C13	408	53	88.89%	60.38%	71.91%
C14	4	51	C14	672	55	70.83%	92.73%	80.31%

TABLE II

CONFUSION MATRICES AND CLASSIFICATION METRICS FOR CCT MODEL TRAINED WITH BI-DIMENSIONAL FOURIER PRE-PROCESSED IMAGE DATA.

Level 1	Predicted		**Accuracy**			**79.84%**		
True	C1	C2	Class	Nr img/class	Test support	Precision	Recall	f1-score
C1	317	84	C1	408	401	77.51%	79.05%	78.27%
C2	92	380	C2	672	472	81.90%	80.51%	81.20%
Level 2a	Predicted		**Accuracy**			**88.62%**		
True	C3	C4	Class	Nr img/class	Test support	Precision	Recall	f1-score
C3	354	5	C3	408	359	89.39%	98.61%	93.77%
C4	42	12	C4	672	54	70.59%	22.22%	33.80%
Level 2b	Predicted		**Accuracy**			**87.83%**		
True	C5	C6	Class	Nr img/class	Test support	Precision	Recall	f1-score
C5	0	56	C5	408	56	-	0.00%	-
C6	0	404	C6	672	404	87.83%	100.00%	93.52%
Level 3	Predicted		**Accuracy**			**84.86%**		
True	C7	C8	Class	Nr img/class	Test support	Precision	Recall	f1-score
C7	41	54	C7	408	95	85.42%	43.16%	57.34%
C8	7	301	C8	672	308	84.79%	97.73%	90.80%
Level 4	Predicted		**Accuracy**			**81.15%**		
True	C9	C10	Class	Nr img/class	Test support	Precision	Recall	f1-score
C9	55	41	C9	408	96	75.34%	57.29%	65.09%
C10	18	199	C10	672	217	82.92%	91.71%	87.09%
Level 5	Predicted		**Accuracy**			**65.30%**		
True	C11	C12	Class	Nr img/class	Test support	Precision	Recall	f1-score
C11	89	26	C11	408	115	64.03%	77.39%	70.08%
C12	50	54	C12	672	104	67.50%	51.92%	58.70%
Level 6	Predicted		**Accuracy**			**50.93%**		
True	C13	C14	Class	Nr img/class	Test support	Precision	Recall	f1-score
C13	1	52	C13	408	53	50.00%	1.89%	3.64%
C14	1	54	C14	672	55	50.94%	98.18%	67.08%

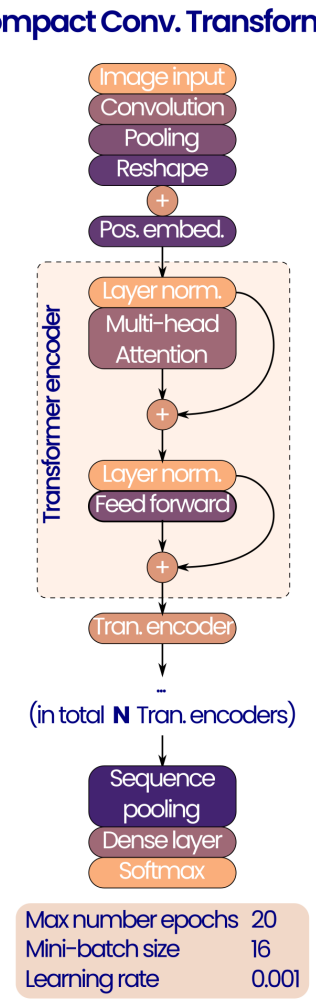


Fig. 2. Illustrative representation of the Compact Convolutional Model adopted and hyperparameters adopted in the present study.

in the Keras python environment [21], [22], composed of two initial convolutional layers and $N = 2$ number of transformer encoder blocks, as illustrated in Figure 2. The authors have not actuated the k-fold cross-validation method for the CCT model. Table I reports the confusion matrices of the averaged classification results expressed in absolute terms, i.e. the number of samples from the test set of the raw GPR samples dataset which has been predicted for each class. The table illustrates the level of overall accuracies and the class metrics precision, recall, and f1-score. Notwithstanding the training conditions may appear limited for training the CCT from scratch, the results in table I have revealed still fairly good behavior with accuracy values above 75% in all the cases. The worst accuracy value of 75.34% has been reached in level 5 whereas the best accuracy value of 92.74% has been recorded in correspondence of level 2a. Averaging all the levels of accuracies, the CCT model trained with the raw dataset, i.e. without any Fourier pre-processing, reached a global classification accuracy of 84.64%. By contrast, Table II reports the confusion matrices of the averaged classification results expressed in percentages for the CCT models trained with the bi-dimensional Fourier pre-processed GPR samples dataset. In this case, the worst level is the last one, presenting the worst accuracy value of 50.93%. Observing in detail the absolute number of predicted samples, it is evident how the CCT biased learned to classify every sample only toward class C14. However, the number of images per class and even the test set supports seem fairly balanced in size. Therefore, the authors believe that one of the more probable reasons for this dreadful result may be a consequence of a severely negative effect of the Fourier pre-processing in level 6, even probably combined with the under-training of the CCT model. The same explanation could be stated for level 5 in which the accuracy stacked only to 65.30%. In the other remaining levels, the CCT has still revealed a good accuracy greater than 79% nonetheless, still reaching an appreciable maximum accuracy value of 88.62% in level 2a. Despite level 2b presenting an interesting accuracy value of 87.83%, observing in detail the absolute number of predicted samples, or to the precision, recall, and f1-score metrics of class C5, also, in this case, the CCT has miserably failed to correctly classify the samples, with a biased tendency toward the class C6 only. However, in this circumstance, it appears quite evident that the most probable reason is related to the unbalanced size of the two classes, and consequently even the test set. For future studies, a possible solution could be forcing a balanced training approach by the class with the minimum number of samples. However, averaging all the levels of accuracies, the CCT model trained with the bi-dimensional Fourier pre-processed dataset reached a less global classification accuracy of 76.93%, with an average reduction of 7.71% with respect to the counterpart CCT trained with the raw dataset. Again, the above-mentioned results demonstrated that, notwithstanding the envisaged advantages of adopting the Fourier pre-processing technique on the GPR sample images, also the CCT model is not able to reach the accuracy levels of the training with the raw GPR dataset. In summary, even in the case of CCT models, the Fourier pre-processing procedure resulted in nefarious effects on the classifiers' ability. This is also virtually exacerbated by a combination of underfitting issues, under-trained models, and even an excessive trivial or compact architecture such as the CCT base model.

Lastly, in order to demonstrate a possible presence of overfitting or underfitting issues during the training phase of all the CCT trained models with and without the Fourier pre-processed dataset, the authors analyzed the convergence curves, which were not reported in the current manuscript due to page limitations. These graphs show the trend of the loss, the accuracy, the validation loss, and the validation accuracy during the training epochs. Focusing on the CCT model trained with raw GPR images dataset, the validation loss curves exhibit global descending trends in all the levels, proving the absence of any overfitting phenomena. However, these curves manifested a possible underfitting, testifying to the under-training of the CCT models which virtually required an increase in the training epochs. Conversely, for the CCT

models trained with the Fourier pre-processed dataset, the validation curve trends revealed the total absence of overfitting occurrences, whilst testifying serious underfitting issues in virtually all the cases, especially in level 2b, level 5, and level 6.

VI. CONCLUSION

This manuscript focuses on the application of a CCT DL model for the classification of defects in tunnel linings on GPR images. The dataset used for this task is composed of images acquired through ground-penetrating radar inspections of tunnels built between the 1960s and 1980s. To enhance image quality and reduce background noise, a two-dimensional Fourier transform-based filtering technique is employed. The study compares the performance of two neural models: one trained on the filtered dataset and another on the unfiltered dataset. The results provide insight into the effectiveness of the proposed approach for automated tunnel lining defect classification. The CCT model was preferred over ViT in this case because it has fewer learnable parameters, which makes it more suitable for use with the limited dataset available for training. This is because with fewer parameters, the model is less likely to overfit the data and is more likely to generalize well to new, unseen data. In addition, the smaller number of parameters makes the CCT model more efficient in terms of computational resources and memory usage, which is important for portability and the possibility of being installed on smaller capacity devices. The models trained in this study appear to perform well considering the limited amount of data available for training. It was unexpected, however, that the model trained on the unfiltered dataset performed better overall compared to the model trained on the filtered dataset. This suggests that the filtering technique using the two-dimensional Fourier transform may not always be necessary or beneficial for defect classification in tunnel linings using GPR data. Further research is needed to investigate this finding and to explore the potential use of CCT for defect classification in other types of infrastructure.

VII. ACKNOWLEDGMENTS

ADDitively Manufactured OPTi-mized Structures by means of Machine Learning" ADDOPTML (ntua.gr) (No: 101007595) belonging to the Marie Skłodowska-Curie Actions (MSCA) Research and Innovation Staff Exchange (RISE) H2020-MSCA-RISE-2020. Their support is highly acknowledged.

REFERENCES

[1] M. Lei, L. Liu, C. Shi, Y. Tan, Y. Lin, and W. Wang, "A novel tunnel-lining crack recognition system based on digital image technology," *Tunnelling and Underground Space Technology*, vol. 108, p. 103724, 2021.

[2] B. Chiaia, G. Ventura, C. Z. Quirini, and G. Marasco, "Bridge active monitoring for maintenance and structural safety," in *International Conference on Arch Bridges*. Springer, 2019, pp. 866–873.

[3] S. Bhalla, Y. Yang, J. Zhao, and C. Soh, "Structural health monitoring of underground facilities – technological issues and challenges," *Tunnelling and Underground Space Technology*, vol. 20, no. 5, pp. 487–500, 2005.

[4] A. Schiavi, G. Niccolini, P. Tarizzo, G. Lacidogna, A. Manuello, and A. Carpinteri, "Analysis of acoustic emission at low frequencies in brittle materials under compression," in *Proc. of SEM Annual Conference & Exposition on Experimental and Applied Mechanics, Albuquerque, 1–4 June 2009*, 2009.

[5] J. Melchiorre, A. Manuello Bertetto, M. M. Rosso, and G. C. Marano, "Acoustic emission and artificial intelligence procedure for crack source localization," *Sensors*, vol. 23, no. 2, p. 693, 2023.

[6] M. M. Rosso, G. M. Marasco, L. T. Tanzi, S. A. Aiello, A. A. Aloisio, R. C. Cucuzza, B. C. Chiaia, G. C. Cirrincione, and G. C. M. Marano, "Advanced deep learning comparisons for non-invasive tunnel lining assessment from ground penetrating radar profiles," in *ECCOMAS Congress 2022-8th European Congress on Computational Methods in Applied Sciences and Engineering*, 2022.

[7] G. Marasco, M. M. Rosso, S. Aiello, A. Aloisio, G. Cirrincione, B. Chiaia, and G. C. Marano, "Ground penetrating radar fourier pre-processing for deep learning tunnel defects' automated classification," in *International Conference on Engineering Applications of Neural Networks*. Springer, 2022, pp. 165–176.

[8] E. Cardarelli, C. Marrone, and L. Orlando, "Evaluation of tunnel stability using integrated geophysical methods," *Journal of Applied Geophysics*, vol. 52, no. 2, pp. 93–102, 2003.

[9] A. G. Davis, M. K. Lim, and C. G. Petersen, "Rapid and economical evaluation of concrete tunnel linings with impulse response and impulse radar non-destructive methods," *NDT & E International*, vol. 38, no. 3, pp. 181–186, 2005.

[10] W. Al-Nuaimy, Y. Huang, M. Nakhkash, M. Fang, V. Nguyen, and A. Eriksen, "Automatic detection of buried utilities and solid objects with gpr using neural networks and pattern recognition," *Journal of Applied Geophysics*, vol. 43, no. 2, pp. 157–165, 2000.

[11] T. Dawood, Z. Zhu, and T. Zayed, "Deterioration mapping in subway infrastructure using sensory data of gpr," *Tunnelling and Underground Space Technology*, vol. 103, p. 103487, 2020.

[12] M. M. Rosso, G. Marasco, S. Aiello, A. Aloisio, B. Chiaia, and G. C. Marano, "Convolutional networks and transformers for intelligent road tunnel investigations," *Computers & Structures*, vol. 275, p. 106918, 2023.

[13] C. Solomon and T. Breckon, *Fundamentals of Digital Image Processing: A practical approach with examples in Matlab*. John Wiley & Sons, 2011.

[14] M. Thompson, "Digital image processing by rafael c. gonzalez and paul wintz," *Leonardo*, vol. 14, no. 3, pp. 256–257, 1981.

[15] R. E. Woods and R. C. Gonzalez, *Digital image processing third edition*, 2021.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[17] A. Hassani, S. Walton, N. Shah, A. Abuduweili, J. Li, and H. Shi, "Escaping the big data paradigm with compact transformers," *arXiv preprint arXiv:2104.05704*, 2021.

[18] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy

[19] W. Wang, J. Zhang, Y. Cao, Y. Shen, and D. Tao, "Towards data-efficient detection transformers," *arXiv preprint arXiv:2203.09507*, 2022.

[20] L. Tanzi, A. Audisio, G. Cirrincione, A. Aprato, and E. Vezzetti, "Vision transformer for femur fracture classification," *Injury*, 2022.

[21] S. Paul *et al.* (2021) Compact convolutional transformers. [Online]. Available: https://github.com/keras-team/keras-io/blob/master/examples/vision/cct.py

[22] F. Chollet *et al.*, "Keras," https://keras.io, 2015.