POLITECNICO DI TORINO
Repository ISTITUZIONALE

Convolutional networks and transformers for intelligent road tunnel investigations

Original
Convolutional networks and transformers for intelligent road tunnel investigations / Rosso, M.M., Marasco, G., Aiello, S., Aloisio, A., Chiaia, B., Marano, G.C.. - In: COMPUTERS & STRUCTURES. - ISSN 0045-7949. - 275:(2023), p. 106918. [10.1016/j.compstruc.2022.106918]

Availability:
This version is available at: 11583/2972727 since: 2022-11-01T10:35:48Z

Publisher:
Elsevier

Published
DOI:10.1016/j.compstruc.2022.106918

Terms of use:

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

Publisher copyright
Elsevier postprint/Author's Accepted Manuscript

© 2023. This manuscript version is made available under the CC-BY-NC-ND 4.0 license
http://creativecommons.org/licenses/by-nc-nd/4.0/.The final authenticated version is available online at:
http://dx.doi.org/10.1016/j.compstruc.2022.106918

(Article begins on next page)

19 November 2026

Convolutional networks and transformers for intelligent road tunnel investigations

Marco Martino Rosso[a], Giulia Marasco[a], Salvatore Aiello[a], Angelo Aloisio[b,*], Bernardino Chiaia[a], Giuseppe Carlo Marano[a]

[a] *Politecnico di Torino, DISEG, Dipartimento di Ingegneria Strutturale, Edile e Geotecnica, Corso Duca Degli Abruzzi, 24, Turin, 10128, Italy*
[b] *Civil Environmental and Architectural Engineering Department, Università degli Studi dell'Aquila, via Giovanni Gronchi n.18, L'Aquila, 67100, Italy*

Abstract

Visual inspections do not provide a reliable and objective assessment of the conservation state of road tunnels. Although direct tests might represent a valid survey approach, they would often lead to prohibitive costs if performed extensively. Therefore, indirect techniques, such as ground-penetrating radar (GPR), have become fundamental to supporting limited direct tests. The analysis of the GPR tunnel linings profiles is mainly hand-operated. It permits the detection of various tunnel linings defects, characterizing a tunnel's global health state. In the present work, the authors developed an artificial intelligence (AI) based automatic road tunnel defects hierarchical classification framework to improve the efficiency of this powerful indirect surveying method. Adopting the most recent tools in image processing provided by the deep learning (DL) community, the authors proposed a convolutional neural

*Corresponding author.
Email address: angelo.aloisio1@univaq.it (Angelo Aloisio)

network (CNN) with the acknowledged ResNet-50 architecture, initialized through the transfer learning method. For the sake of comparisons, the authors also adopted the state-of-art convolutional EfficientNet architecture. To further improve the proposed framework, the authors investigated how the bidimensional Fourier transform applied as a preprocessing procedure could affect the classification performances of the ResNet-50 model. Finally, to further enhance the classification performance, the state-of-art neural vision transformer (ViT) architecture has been adopted with the transfer learning approach to the currently proposed defects classification framework.
Keywords: Deep learning, Vision Transformers, Road tunnels, Fourier transform, Convolutional Neural Network, Structural Health Monitoring, Ground Penetrating Radar

1. Introduction

The current diagnosis paradigm for the health assessment of road tunnels requires an initial knowledge phase based on original drawings and documentation [1, 2, 3, 4, 5]. This foremost step characterizes the declared project requirements, the initial structural testing reports, and the material. This information is usually combined with the results of periodical visual inspections to know the actual as-built state [6, 7, 8, 9, 10, 11]. During these inspections, there is also a survey of surface irregularities with hammering tests, localization, and quantification of possible degradation flaws. To direct characterize the mechanical properties of the tunnel lining concrete, a certain number

of specimens are usually extracted and tested with destructive compression tests. Regrettably, although direct testing offers very reliable results, the main drawback is related to the fact that it provides a piece of punctual information. To have a quite complete overview of the heterogeneity of the lining state, it would be virtually necessary to perform a very high number of investigations to provide extended coverage of the tunnel's internal surface along with its longitudinal development. This procedure would lead to prohibitive costs. Moreover, these costs would only be related to the initial investigation phase, without any maintenance or restoration interventions. For this reason, non-destructive and indirect techniques have been successfully included in cost-effective diagnosis and maintenance plans to reduce surveying costs. Common current approaches are ground-penetrating radar profiles (GPR), laser scanners, and thermography acquisitions. Likewise, for all the indirect testing methods, their main shortcoming is the need for an accurate calibration to rely on their outcomes entirely. Therefore, the indirect approaches are not a substitute for the direct ones. Still, they support direct investigations, mainly when these latter are limited in number due to budget restrictions.

GPR is a geophysical technique [12] that involves transmitting high-frequency electromagnetic wave impulses inside the material under study using an antenna with a frequency of 10 to 2600 MHz. The dielectric characteristics of the material influence the propagation of such an impulse. A receiver antenna collects the reflected signals to inspect the material in-depth

[12]. The GPR provides, therefore, an image as output, evidencing the presence of anomalies, defects, fractures, etc., overcoming the drawbacks of a direct visual inspection. A GPR inspection output is an image which presents the progressive longitudinal distance from the beginning of the tunnel. Three profiles are usually inspected with two-lane roads to characterize the single tunnel better. In contrast, five profiles are generally examined for three-lane roads, as illustrated in Figure 1. As shown, notwithstanding two different GPR testing profile configurations, in both cases, the attention is mainly focused on the critical segment of the top crown area, which comprises the two lateral haunches (shoulder joints). This area represents the most dangerous zone for road drivers if some concrete chunks from the primary concrete layer detachments fall on the road. When the GPR method for road tunnels is employed, the GPR linings require high skills and very experienced personnel to identify and classify the presence of all different linings defects, as depicted in Figure 2. This approach appears remarkably time-consuming and engineering judgment-based and experience-based only, thus more prone to possible subjective evaluations. In the present work, the authors propose to adopt an automatic procedure to classify the road tunnel linings defects from the output images provided by the indirect GPR testing. Nowadays, Machine learning (ML) and, especially, deep learning (DL) methods are in the spotlight due to their successful applications in solving complex engineering and mathematical problems, e.g. physics-informed neural networks to approximate solutions of partial differential equations [13, 14], and they

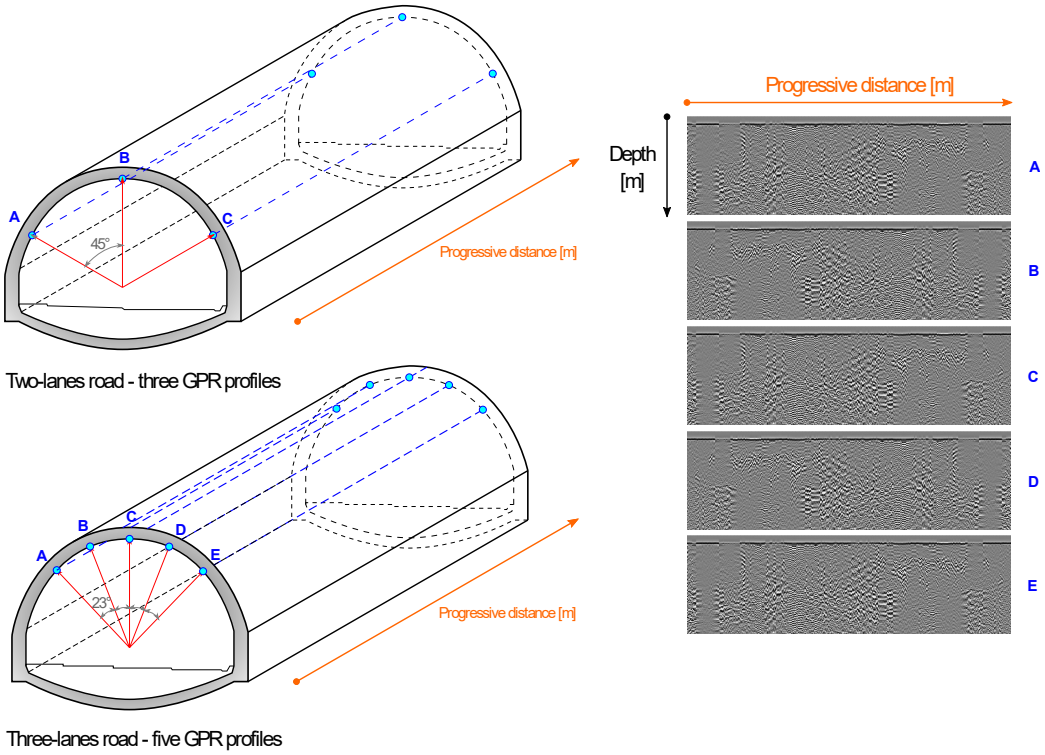


Figure 1: An illustrative example of GPR tunnel linings profiles and their number for different tunnel widths, related to a different number of road lanes.

revealed their innovativeness and potentiality for automatic image processing and classification tasks [15, 16]. Convolutional neural networks (CNNs) proved to be effective tools to accomplish those tasks, and they represent the most widely adopted techniques [17, 18]. Furthermore, to deal with the computational resources required for training complex neural models, the use of deep CNNs that exploit transfer learning processes has been proven effective in many engineering applications [17]. In the present study, the authors proposed a hierarchical multi-level road tunnel linings defects classification from GPR profiles executed foremost with the CNN ResNet-50. To provide

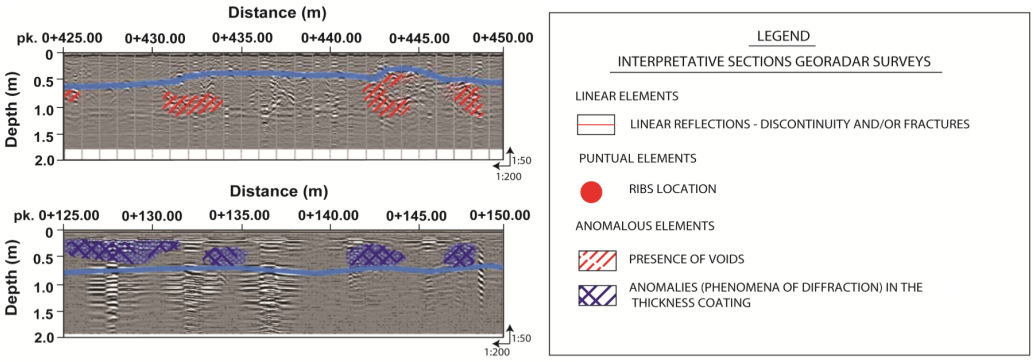


Figure 2: An example overview of GPR tunnel linings profiles defects categorization and labelling process provided by very high skilled and experienced personnel.

a comparison of the performance with a recent state-of-art convolutional architecture, the authors compared the ResNet-50 with the EfficientNet model [19]. Furthermore, for the sake of a more complete classification performance evaluation and a mutual validation purpose of generalization capabilities, the ResNet-50 architecture has been trained on preprocessed images with two-dimensional Fourier transform, thus comparing the two procedures' results. Finally, to further enhance the classification performances, the authors applied a more recent neural model, i.e. the vision transformer (ViT) [20]. The novel contributions of this article can be summarized as follows:

- Improvement of tunnel health diagnosis paradigm with the formulation of an AI-based automatic framework for the hierarchical multi-level road tunnel linings defects classification;
- Adoption of a CNN ResNet-50 with transfer learning approach to accomplish the defects classification working on original images provided

by GPR instrumentation;

- Comparisons of the classification results of the trained CNN model ResNet-50 on original images with a different state-of-art CNN architecture, the EfficientNet, and with another CNN ResNet-50 model trained instead on samples that undergo the bidimensional Fourier GPR images preprocessing;

- Enhancing the CNN neural architecture model with a more recent architecture acknowledged vision transformer and comparing the classification results with the prior cases.

The current paper is organized as follows: in section 2, the proposed road tunnel defects automatic classification framework and available dataset have been described. In section 3 the adopted CNN ResNet-50 is presented and the obtained classification results are reported. In section 4, the authors investigate the effects produced by the bi-dimensional Fourier transform on the original images provided by the GPR. On the other hand, in section 5 the authors investigate how the classification performance changes by adopting the innovative ViT model. Finally, an extended and critical discussion on the results obtained from the three main analyzed cases is extensively argued in section 6.

2. Road tunnel defects automatic classification framework

A complete risk analysis of road tunnels involves several parameters and careful evaluations. At least the following aspects should be considered: traffic conditions, management procedures, surroundings, structural plant equipment, and the structural elements [21]. In the current work, the structural elements have been investigated with indirect non-destructive testing based on the geophysics technique of GPR profiles. The authors focused on an ensemble of the Italian highway panorama road tunnels. These structures date back from the 1960s to the 1980s. The current traditional approach of the GPR profiles data postprocessing retrieved from road tunnels is based on the intervention of expert personnel. Based on a great experience in this field and engineering judgment, it is possible to manually recognize certain patterns in the GPR profile images and provide a defects classification. This procedure is helpful and of great support for inspecting the actual health state of the structural elements of the road tunnels. However, the current paradigm is noticeably time-consuming and, therefore, costly. Moreover, it may be more prone to subjective evaluations, depending on the innate ability or the single operator's experience. In the present study, the authors propose an automatic classification of the road tunnel defects based on an AI-based methodology. To develop this automatic classification, a hierarchical classification tree has been defined as represented in Figure 3. It is based on a classification pattern in which a single defect may be classified, adopting a binary approach at each node of the hierarchical graph. This method re-

sembles the human expert's mental process while recognizing and classifying each defect in the GPR profile. In total, 14 classes have been considered, denoted as Ci with $i = 1, 2, ..., 14$, spread over 6 main levels: Level 1 (C1, C2 folders), Level 2a (C3, C4 folders), Level 2b (C5, C6 folders), Level 3 (C7, C8 folders), Level 4 (C9, C10 folders), Level 5 (C11, C12 folders), and Level 6 (C13, C14 folders). Level 1 is devoted to locating the completely healthy samples (C1) from the ones with generic flaws (C2). Level 2a deals with healthy samples (C3) to potentially locate the ones with the presence of reinforcement bars (C4), characterized by distinctive narrow hyperbolas patterns. Level 2b, performs an initial defects classification by a generic warning mix (C5) which may not easily be categorized from other more specific flaws (C6). The class C6 is further analyzed to locate cracks (C7) in level 3 from other flaws (C8). This latter is further investigated in level 4, to characterize the anomalies in the concrete linings (C9) from the voids defects (C10). In level 5, a more detailed classification provides the image categorization with simple voids (C11) from the others (C12). Finally, this latter class is further analyzed in level 6 to categorize the excavation problems (C13) and the concrete-rock detachments issues (C14).

After that, priorly to proceeding with the training and testing procedure which usually characterize the DL methods, the database has been constructed as illustrated in Figure 4. Each image of GPR profiles for the tunnel at the authors' disposal has been cropped with a constant step size of 5.00 m along the abscissa. The abscissa measures the progressive distance

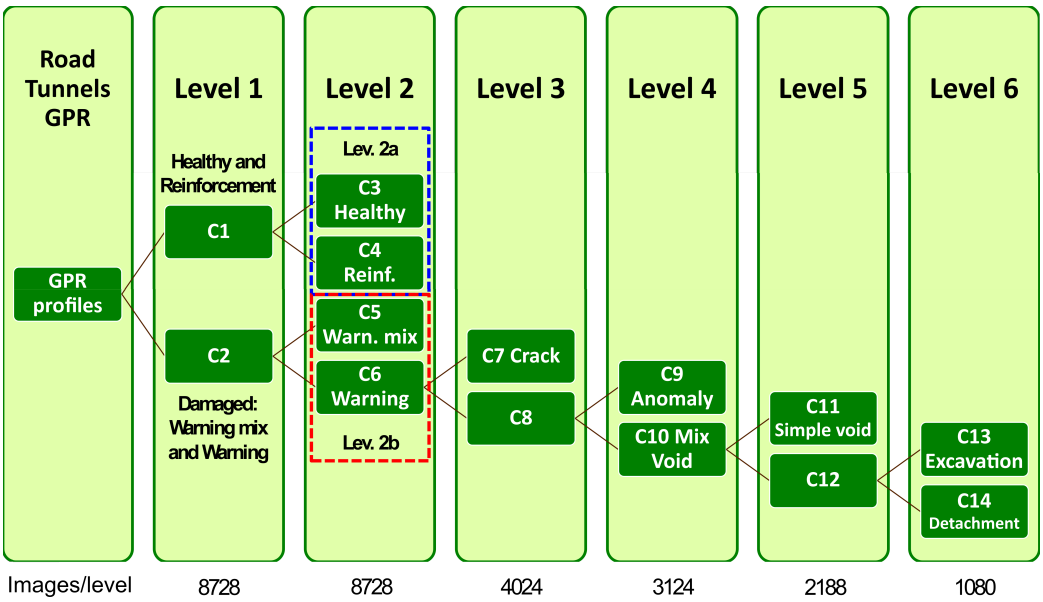


Figure 3: Hierarchical framework for the multi-level tunnel defects GPR profiles classification.

from the beginning of the tunnel under investigation. Each image obtained from the larger one keeps the label of the defects that have been found, and thus moved to each folder. To avoid defects being placed across two different images, the cutting step was occasionally manually altered to provide samples, allowing for a more precise classification. Therefore, seven DL classification models have been analyzed in the present work to accomplish the GPR tunnel defects classification tasks for SHM purposes with the CNN ResNet-50 model and with a state-of-art image-processing advanced DL approach. To analyse the effects of the model architecture on the GPR tunnel defects' classification, the authors compared the results of ResNet-50 with a different state-of-art convolutional architecture, i.e. the EfficientNet. Fur-

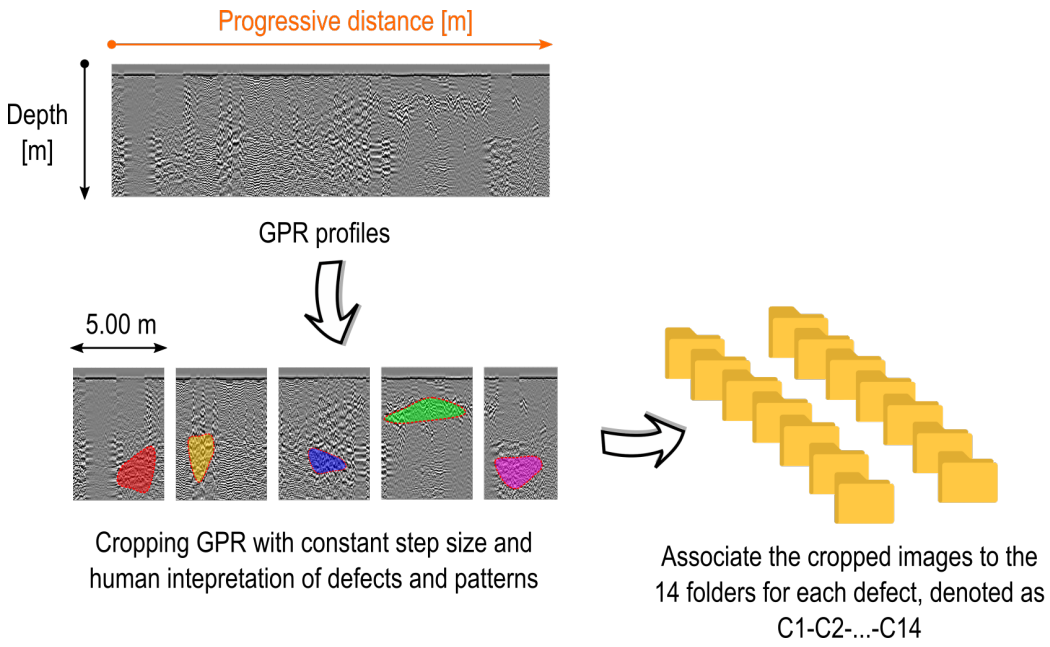


Figure 4: Database construction from GPR road tunnel linings profiles.

thermore, for the sake of completeness, in an attempt to improve the current basic technique, a preprocessing phase has been performed, adopting the bi-dimensional Fourier transform to the GPR sample images, as discussed in the following. For the sake of comparisons, a completely different and novel DL technique is employed for the first time within the road tunnel assessment field, the neural transformer, i.e. the ViT variant. Specifically, referring to Figure 3, the authors focused on level 6 (1080 images in total), level 5 (2188 images in total), level 4 (3124 images in total), level 3 (4024 images in total), level 2a (4130 images in total), level 2b (4598 images in total), and level 1 (8728 images in total). Thereafter, the neural models were trained. Therefore, it would be virtually possible to automatically evaluate the road

tunnel linings' GPR profiles to establish the current structural health status. This technique should lead to higher efficiency since the expert personnel's duty is to check if this automatic procedure succeeded in recognizing all the defects in the image. However, with the traditional CNN method, getting information about the defect localization into the image is not trivial. On the other hand, exploiting the self-attention mechanism, as further explained later in this document, the transformer architecture can evidence which parts of the images contributed to its classification, providing an initial localization indication of the detected defect.

3. Convolutional Neural Network: ResNet-50

The first model employed for road tunnel automatic defect classification is the most widespread DL technique based on CNN algorithms. In several literature studies, better performances in the learning process are usually appointed to deeper neural network models with respect to shallower ones [22, 23]. These models are devoted to dealing with the image data type, and through specific learnable filters, CNNs are able to extract feature

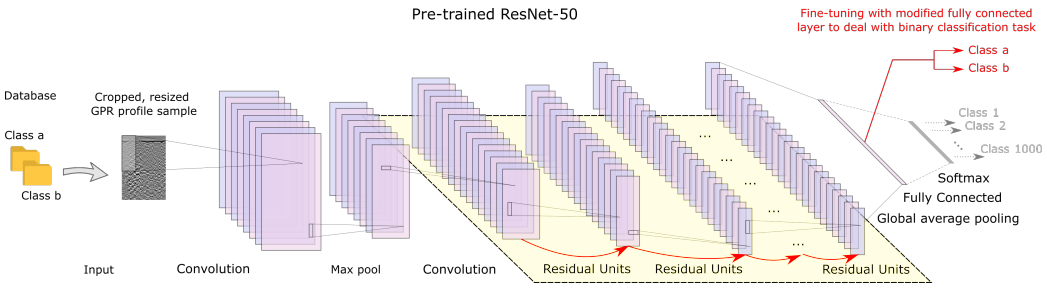


Figure 5: ResNet-50 illustrative example.

maps to accomplish machine learning tasks automatically. CNNs have been explored starting from the 1980s within the brain's visual cortex studies. However, only in the last decade they have gained a leading position in the DL panorama because of new higher computational capabilities and training data availability [24, 25]. Furthermore, transfer learning methodology further contributes to breaking down the computational effort by adopting pre-trained models adapted to specific tasks with fine-tuning approaches [26]. In the current study, the CNN model acknowledged as ResNet-50 has been employed, as depicted in Figure 5. He et al. [27] in 2015 illustrated for the first time the actual potentialities of this model. This model is pre-trained on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) dataset based on 1.3 millions of images as training, 50,000 as validation, and 100,000 as test images [28, 29]. The main feature of the ResNet model is the presence of skip or residual connections. These connections transfer the input information directly to the output of a layer and they are extremely helpful during training, especially at the beginning of the learning process. Indeed, it initially behaves as an identity function and speeds up the learning process even when the weights of some layers are close to zero, which normally slows down the back-propagation with the vanishing gradient issues [24]. The ResNet-50 adopted in the current study has been implemented in MATLAB2021a programming environment [30], by adopting the pre-trained model by [28] trained on ILSVRC public image dataset. The architecture of the ResNet-50 is composed of 50 layers (only counting the convolutional layers and the

fully connected layers). It comprises a stack of residual units, each composed of two convolutional layers, without pooling layers, batch normalization, and rectified linear unit (ReLU) activation function, using 3x3 kernels with stride 1 [31, 24]. Input images are 224x224x3 tensors, considering the common RGB for coloured image codification. Therefore the cropped GPR profile samples have been resized accordingly [18]. The initial pre-trained ResNet-50 was able to classify images into 1000 different object categories with a fully connected layer with 1000 units and a softmax layer. However, it was re-adapted to the current study. The final fully connected layer has been changed to accomplish the binary classification tasks for each hierarchical level presented in Figure 3. Following in-depth the tree depicted in Figure 3, the total number of available samples for each level gradually decreases. Moreover, since each class presents an unbalanced number of images, to train a good classification model, the class forced a balanced approach with the minimum number of samples. Additionally, an appropriate setting of the hyperparameters for the neural model plays an important key role. However, a general, unique, universally acknowledged, efficient, and effective hyperparameter tuning procedure does not exist yet nowadays. Some techniques and procedures have been developed recently, such as exploring the hyperparameter space with a grid search or a random search approach combined with cross-validation procedures [24]. However, a refined search of this type may require a long time and probably a high computational effort. For this reason, for practical reasons, at first, an empirical trial-and-error procedure is generally acknowl-

Table 1: Hyperparameters adopted for CNN ResNet-50.

Hyperparameters	Value
Mini-batch size	32
Learning rate	0.001
Maximum number of epochs	12

edged to be advantageous in order to balance computational efforts and the model's generalization capabilities [22]. Albeit an accurate hyperparameter tuning may lead to more accurate results, for engineering purposes adopting the trivial trial-and-error procedure may appear in most cases sufficiently adequate to reach high accuracy levels, e.g. greater than 90-95% [18]. Therefore, the authors empirically defined the hyperparameters required for the DL model by a trial-and-error approach to achieving the best possible results, as illustrated in Table 1. In detail, the learning rate was set to 0.001, the mini-batch size was chosen as 32, and the maximum number of epochs to contain computational effort and achieve the best performances was set to 12. After the training procedure, the results on the test set are provided in Table 2. The tables show the average values provided by the cross-validation technique with a k-fold equal to 10, as suggested in [32] being a good choice to avoid both significant variance and biased values. According to the values of the tables, the AI-based automatic defects classification pointed out an average accuracy of 0.945 among all the levels of the hierarchical classification tree. The section 6 provides an extended discussion of the results. In the following, the authors provided a comparison with a state-of-art CNN architecture, the EfficientNet [19]. Thereafter, the authors attempted to im-

prove the ResNet-50 model by introducing preprocessing approach to the raw image data of road tunnels' GPR profiles.

3.1. Comparison with state-of-art CNN: EfficientNet

In [19], in an attempt to provide a new systematic scaling-up method for CNN models, the authors designed a new family of state-of-art convolutional networks denoted as EfficientNet. The previous model scaling-up procedures appeared relatively arbitrary and challenging to efficiently balance all the dimensions of a neural network for the available computational resources. On the other hand, the proposed new scaling approach uniformly scales all dimensions of depth, width, and resolution by employing a simple and constant ratio denoted as the compound coefficient. In the present study, the authors compared the previous results of ResNet-50 with this state-of-art convolutional architecture, i.e. the EfficientNet. The current adopted implementation is the pre-trained model provided in Matlab environment denoted as EfficientNet-B0, which presents an architecture with mobile inverted bottleneck convolutional building blocks [19]. Similarly to before, the authors employed the same hyperparameters of Table 1 and re-adapted the final fully connected layers to perform a binary classification for each level of the GPR tunnel defects hierarchical classification tree. After the training procedure, the results of the test set are provided in Table 3. The tables show the average values provided by the cross-validation technique with a k-fold equal to 10. According to the values of the tables, the EfficientNet automatic defects

Table 2: ResNet-50 classification results in terms of confusion matrices for Levels 1, 2a, 2b, 3, 4, 5, and 6.

Level 6

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C13	C14		C13	0.942	0.966	0.954
C13	0.966	0.034	0.9535	C14	0.965	0.941	0.953
C14	0.059	0.941					

Level 5

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C11	C12		C11	0.978	0.988	0.983
C11	0.988	0.012	0.9830	C12	0.988	0.978	0.983
C12	0.022	0.978					

Level 4

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C9	C10		C9	0.894	0.949	0.920
C9	0.949	0.051	0.9180	C10	0.946	0.887	0.915
C10	0.113	0.887					

Level 3

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C7	C8		C7	0.990	0.927	0.958
C7	0.927	0.073	0.9590	C8	0.931	0.991	0.960
C8	0.009	0.991					

Level 2b

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C5	C6		C5	0.900	0.909	0.904
C5	0.909	0.091	0.9040	C6	0.908	0.899	0.904
C6	0.101	0.899					

Level 2a

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C3	C4		C3	0.962	0.984	0.973
C3	0.984	0.016	0.9725	C4	0.984	0.961	0.972
C4	0.039	0.961					

Level 1

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C1	C2		C1	0.920	0.933	0.927
C1	0.933	0.067	0.9260	C2	0.932	0.919	0.925
C2	0.081	0.919					

classification pointed out an average accuracy of 0.917 among all the levels of the hierarchical classification tree. Compared with the results of ResNet-50 in Table 2, it is worth noting that, the EfficientNet reaches almost the same results in terms of accuracy for levels 3 and 4, while slightly improving the results for level 1 and 6. However, it performed worse than ResNet in levels 2b and 5, probably due to arising of overfitting issues that occurred in those levels, as argued in section 6. In summary, these results evidenced that the recent EfficientNet architecture represents a sort of trade-off choice between efficiency and classification accuracy since it was able to reach in virtually all cases almost the same order of magnitude of accuracy levels.

4. Fourier transform preprocessing effects on ResNet-50

The road tunnel inspections with indirect GPR techniques provide tunnel linings profiles as image data types. The automatic defect classification may be solved by adopting machine learning techniques in a data-driven approach, exploiting information in the image input dataset. To extract information in the GPR images, the ResNet-50 uses the automatic feature extraction through specific size kernel filters, which slide through convolution operation on the input image, providing feature maps as results. On the other side, image processing approaches offer nowadays powerful tools and interesting techniques to extract information directly in the preprocessing or postprocessing phase, e.g. image enhancement, image restoration, or image data compression [33, 34, 35]. In this section, the authors focused on the

Table 3: EfficientNet classification results in terms of confusion matrices for Levels 1, 2a, 2b, 3, 4, 5, and 6.

Level 6							
	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C13	C14		C13	0.959	0.963	0.961
C13	0.963293	0.036707	0.9608	C14	0.963	0.958	0.961
C14	0.041707	0.958293					
Level 5							
	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C11	C12		C11	0.961	0.906	0.933
C11	0.906481	0.093519	0.9347	C12	0.911	0.963	0.937
C12	0.037037	0.962963					
Level 4							
	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C9	C10		C9	0.925	0.886	0.905
C9	0.885633	0.114367	0.9070	C10	0.890	0.928	0.909
C10	0.071574	0.928426					
Level 3							
	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C7	C8		C7	0.916	0.990	0.951
C7	0.99	0.01	0.9494	C8	0.989	0.909	0.947
C8	0.091111	0.908889					
Level 2b							
	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C5	C6		C5	0.806	0.817	0.811
C5	0.81709	0.18291	0.8101	C6	0.814	0.803	0.809
C6	0.196945	0.803055					
Level 2a							
	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C3	C4		C3	0.926	0.892	0.909
C3	0.892449	0.107551	0.9107	C4	0.896	0.929	0.912
C4	0.07102	0.92898					
Level 1							
	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C1	C2		C1	0.937	0.955	0.946
C1	0.954964	0.045036	0.9455	C2	0.954	0.936	0.945
C2	0.063923	0.936077					

bidimensional Fourier transform (2D-DFT), given its recent successful applications. Considering at the beginning continuous one-dimensional signals, the Fourier transform permits a domain transformation from the time-space to the frequency domain, or in more general terms, from the input domain to the Fourier domain. The signal can be therefore expressed as the superposition of harmonics components (sine and cosine functions) weighted according to their frequency content. However, working with real signals unavoidably involves working with sampled discrete signals. The Nyquist-Shannon theorem's mathematical framework defines certain limits to the domain mapping produced by the Fourier transform and its inverse transform. Specifically, it poses a relationship between the sampling frequency of the actual signal and the highest reconstructable frequency component in the Fourier domain, denoted as Nyquist Frequency [36, 37]. Without loss of generality, the same concepts of the discrete Fourier transform were easily extended to bi-dimensional signals [38]. Lets denote a digital image of size $M \times N$ pixels as $f(x, y)$ for $x = 0, 1, ..., M - 1$ and $y = 0, 1, 2, ..., N - 1$, the bidimensional discrete Fourier transform (2D-DFT) of $f(x, y)$ is denoted as $F(u, v)$ [33]

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-2\pi i \left(\frac{ux}{M} + \frac{vy}{N}\right)} \tag{1}$$

where $u = 0, 1, 2, ..., M - 1$ and $v = 0, 1, 2, ..., N - 1$. Adopting the Euler's formula, it would be possible to expand the exponential into its sinusoidal components, retrieving the corresponding frequency according the x and y

direction of the image mapped in the frequency domain (or frequency rectangle) and denoted by the variables u and v [33]. The inverse transform (2D-IDFT) is expressed as

$$f(x,y) = \frac{1}{N \cdot M} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{2\pi i \left(\frac{ux}{M} + \frac{vy}{N}\right)} \tag{2}$$

Since, the bijective direct and inverse mapping, the values of $F(u,v)$ are even acknowledged as the Fourier coefficients of the expansion, and they are typically complex numbers [33, 35]. The value of the Fourier transform at the origin of the frequency rectange $F(0,0)$ is numerically equal to MN times the average value of $f(x,y)$. Borrowing the terminology from electrical engineering, this is acknowledged as *dc-component*, which means direct current (i.e. at zero frequency, in contrast with alternate current). Since its complex nature, denoting as $\mathfrak{Re}(u,v)$ and $\mathfrak{Im}(u,v)$ the real and imaginary part respectively of the transform $F(u,v)$, the Fourier spectrum (i.e. the modulus of the complex number) is given by

$$|F(u,v)| = \sqrt{\mathfrak{Re}(u,v)^2 + \mathfrak{Im}(u,v)^2} \tag{3}$$

and the phase angle

$$\phi(u,v) = \arctan \frac{\mathfrak{Im}(u,v)}{\mathfrak{Re}(u,v)} \tag{4}$$

providing a polar representation of the bidimensional Fourier transform

$$F(u,v) = |F(u,v)|e^{i\phi(u,v)} \tag{5}$$

For purposes of visualization, since most of the information is contained in the magnitude, the phase can be neglected. However, when the inverse transform 2D-IDFT is employed, the phase information would be strictly required, otherwise, a corrupted image will be the final result due to the loss of information [37]. Since the magnitude may present very scattered values comparing the largest DC component with respect to the other frequencies, a logarithmic transformation is usually applied to enhance the information contained in low-frequency components:

$$\tilde{F}(u,v) = c\log(1 + |F(u,v)|) \tag{6}$$

where c represents a scale factor, which has been set to unity in the current work. The 2D-DFT may also be computed as a series of $2n$ one-dimensional FT [37], which leads to a computational complexity of $O(n^2)$. However, faster formulations (2D-FFT) have been developed in order to further reduce the computational effort to $O(n\log_2(n))$ [37]. Matlab environment provide the command `fft2` to compute the bidimensional Fourier transform to theroad tunnel GPR profiles images, and the command `fftshift` multiplies $(-1)^{x+y}$ to the images permitting to centering again the dc-component in the origin of the frequency rectangle, operation strictly required due to the fast Fourier

algorithm formulation [33]. Among the many useful properties of the FT, e.g. linearity, translation, rotation, etc. the most important in the present case is related to the convolution theorem [39]. In practice, the convolution operation denoted with the symbol $*$ between the digital image $f(x, y)$ and a filter kernel mask function $h(x, y)$ is given by the following expression

$$f(x, y) * h(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) h(x - m, y - n) \tag{7}$$

In the Fourier domain, the convolution operation translates into a simple product of the Fourier transform of the two functions

$$f(x, y) * h(x, y) \Leftrightarrow F(u, v) H(u, v) \tag{8}$$

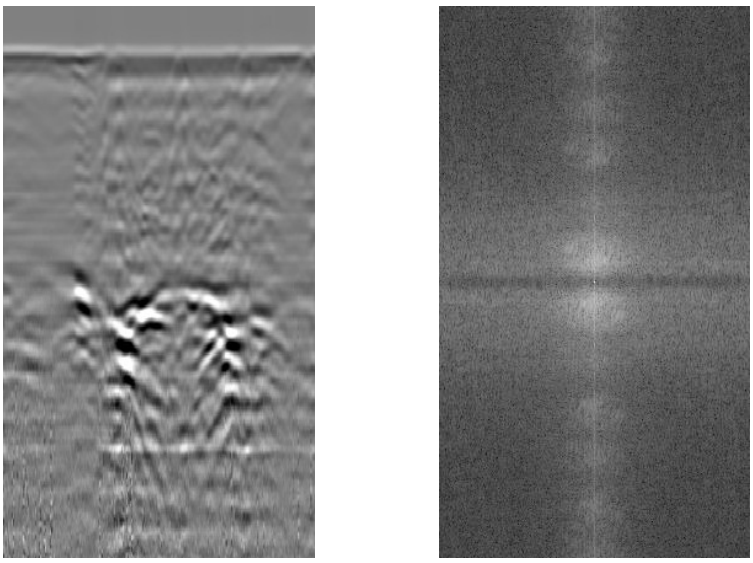
and the double arrow sign underlines the direct and inverse validity, even if for the filtering process only the left-to-right expression is of interest. For this reason, the Fourier transform of the kernel function $H(u, v)$ is also acknowledged as filter transfer function [33, 39], because it apply a certain transformation to every component of the $F(u, v)$. In image processing, digital filters are extremely useful to smooth the image, by suppressing high frequencies in the image, or to detect edges by removing the low frequencies [37]. Filters with convolution operation are the propeller of the automatic feature extraction of CNN. In reality, the CNN automatic feature extraction relies on a correlation operation, which is the process of sliding a filter function on

the image and computing the sum of products at each location, similar to a convolution but reversing the sign into the filter kernel, i.e. rotating the filter of a straight angle

$$f(x,y) * h(x,y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n)h(x+m, y+n) \tag{9}$$

Due to the convolution theorem, the application of the bidimensional Fourier preprocessing to the road tunnel linings GPR profiles images suggested to the authors that the automatic feature extraction would be more efficient for CNN-based automatic defects classification paradigm. Indeed, this kind of preprocessing produced a data and information compression for every image, without jeopardizing the geometric nature of the GPR profiles images (investigated depth along the vertical axis). Moreover, as depicted in the illustrative example of Figure 6, the bidimensional Fourier preprocessing maintains the vertical and horizontal patterns in the input image, preserving them in the most dominant frequency components in the Fourier domain, compressing the information of the periodic components, typically of the GPR profiles in the depth direction, and removing the non-periodic noise effects [39].

In conclusion, the entire database at the author's disposal of GPR road tunnel linings profiles undergoes the preprocessing phase with bidimensional Fourier operation. The newly obtained dataset was used to train the ResNet-50 model, already mentioned in the previous sections and with the same hyperparameter of Table 1. The cross-validation procedure with a k-fold equal



(a) (b)

Figure 6: An explanatory example of bidimensional Fourier preprocessing before (a) and after (b) the preprocessing application on a road tunnel GPR profile sample image belonging to the class C11 (simple voids).

to 10 has been again adopted, and the test set averaged on the ten trained model results are reported in Table 4. According to the first sight of the results shown in the table, the AI-based automatic defects classification with Fourier preprocessing pointed out an average accuracy among all the levels of the hierarchical classification tree less than before. In the present case, the average accuracy settles at 0.856 only, despite the advantages foreshadowed. The authors provide in section 6 an extended discussion of the results.

Table 4: ResNet-50 classification results with bidimensional Fourier pre-processed dataset in terms of confusion matrices for Levels 1, 2a, 2b, 3, 4, 5, and 6.

Level 6

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C13	C14		C13	0.891	0.924	0.907
C13	0.924	0.076	0.9055	C14	0.921	0.887	0.904
C14	0.113	0.887					

Level 5

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C11	C12		C11	0.936	0.857	0.895
C11	0.857	0.143	0.8990	C12	0.868	0.941	0.903
C12	0.059	0.941					

Level 4

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C9	C10		C9	0.861	0.839	0.850
C9	0.839	0.161	0.8515	C10	0.843	0.864	0.853
C10	0.136	0.864					

Level 3

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C7	C8		C7	0.916	0.816	0.863
C7	0.978	0.220	0.8590	C8	0.805	0.910	0.854
C8	0.090	0.910					

Level 2b

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C5	C6		C5	0.779	0.735	0.756
C5	0.735	0.265	0.7630	C6	0.749	0.791	0.769
C6	0.209	0.791					

Level 2a

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C3	C4		C3	0.859	0.793	0.825
C3	0.793	0.207	0.8315	C4	0.808	0.870	0.838
C4	0.130	0.870					

Level 1

	Predicted		Accuracy	Class	Precision	Recall	f1-score
True	C1	C2		C1	0.885	0.879	0.882
C1	0.879	0.121	0.8825	C2	0.880	0.886	0.883
C2	0.114	0.886					

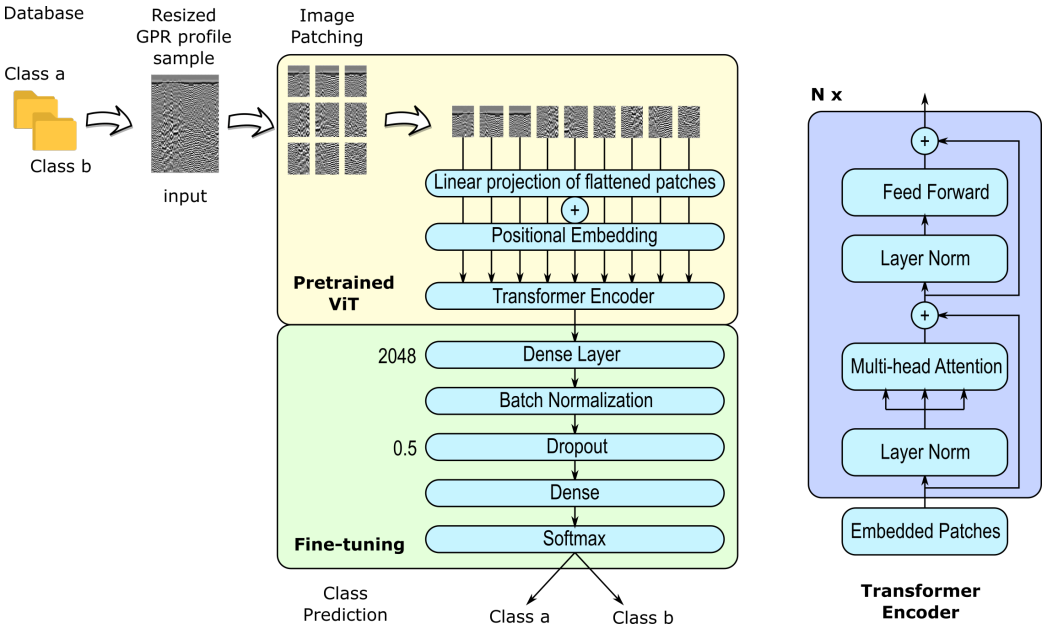


Figure 7: ViT-L16 model architecture visualization, highlighting the pre-trained part and the part involved by the fine-tuning for the current problem and a particular focus on the transformer encoder block.

5. Vision Transformers for tunnel defect classification

In 2017, the research conducted by the Google team entitled "*Attention is all you need*" [40] echoed a revolution in the DL field because, for the first time, the neural transformer architecture appeared. Their very first application was referred to as Natural Language Processing (NLP) tasks, but it has quickly spread like wildfire in many other engineering and scientific applications since their revolutionary approach. The transformer was born to overcome the problematics of the state-of-art DL technique when dealing with long data sequences. Its main peculiarity is the total entrustment to the self-attention mechanism, which provides the network with short and

even very long-range relationships among every element with the others composing the data sequence. With this hidden information, the network can extract deep knowledge from data and produce noticeably promising results for machine learning tasks, e.g. classification problems. Moreover, due to their nature, every block of the transformer may be parallelized to improve computational efficiency. However, since these models are remarkably computational demanding due to the massive number of learnable parameters, it is not easy to train a transformer from scratch without possessing a considerably extended dataset and allocating computational resources for days or, more probably, for weeks. For this reason, transfer learning approaches have proved to be the most promising way to exploit them [17] effectively. Recently, impactful and fruitful studies have been conducted, such as, e.g., the introduction of BERT model (Bidirectional Encoder Representations from Transformers) [41] which exploits only the encoder part of the transformers, or the adaptation of neural transformers to deal with images data types. In October 2020, in [42], the authors presented the Vision Transformer (ViT), a novel architecture to deal with image data. The ViT beat the best CNN models such as ResNet for image classification for a sufficiently large dataset for pre-training [42]. ViT is based solely on the neural transformer architecture's encoder network part, similarly to the BERT model for natural language processing [40]. The model architecture of the ViT adopted for the current automatic defects classification for road tunnel GPR indirect testing is depicted in Figure 7. To deal with images, it was necessary to rethink

the image data as a sequence of data. This was realized by producing a partitioning of the input image, which is subdivided into a finite number n of patches. These patches may overlap or not, and each one is a tensor with shape $d_1 \times d_2 \times d_3$, this latter corresponding to the Red-Green-Blue (RGB) digital image encoding. However, it is always required to define an exact and suitable finite number n of resulting patches considering the resolution of the starting image. To treat these patches as an ordered sequence of elements, a vectorization procedure involves each i-th patch provide a column vector $\boldsymbol{x}_i$, with $i = 1, 2, ..., n$, of dimension $d_1 d_2 d_3 \times 1$, resulting at the end into n vectors. These vectors of flattened patches are fed into a dense layer with shared parameters and linear activation function, producing the hidden embedded representation typical of the transformer architectures denoted as $\boldsymbol{z}_i$. The shared weights of this dense layer, which are learned from training data during the training phase, act as the flattening operation using a linear projection matrix [20]. Since in a sequence, the comparison ordering is extremely important, to maintain the information of the actual position of each patch within the initial image, a positional encoding [40] should be applied. The positional encoding is usually referred to sine and cosine functions at high frequency, which adds spatial information into the network. Then, they are simply summed to the embedding representations $\boldsymbol{z}_i$. In this way, the new representation of the input information $\boldsymbol{z}_i$ captures both the content and the position of the i-th patch. Similar to BERT-based architecture, the ViT relies only on the transformer encoder block, which is repeated N times.

However, in order to accomplish the classification task, BERT introduced an additional token denoted as *[CLASS]* token. Similarly, also in the ViT model, the *[CLASS]* token for classification is fed to an embedding layer producing the vector $\boldsymbol{z}_0$ of the same shape as other embeddings. The sequence of vectors $\{\boldsymbol{z}_i\}_{i=0}^{n}$ are subsequently fed to the neural transformer encoder block, composed of a stack of a multi-head self-attention and dense fully-connected layer blocks, actually employing normalization and skip connections, as depicted in detail in Figure 7. The output of the neural transformer encoder is a new representation of the input vectors $\{\boldsymbol{z}_i\}_{i=0}^{n}$ mapped to a new representations $\{\boldsymbol{c}_i\}_{i=0}^{n}$ which integrates the scaled dot-product attention (the self-attention) [40]. In any case, only the $\boldsymbol{c}_0$ is considered for the classification task, the one referred to the *[CLASS]* token, and the others may usually be neglected. This vector represents the feature vector from the input image. This vector is thus normally fed to a final multi-layer perceptron followed by a softmax classifier. This last layer results in a column vector $\boldsymbol{p}$ of size equal to the number of output classes, representing a probability to belong to each output class.

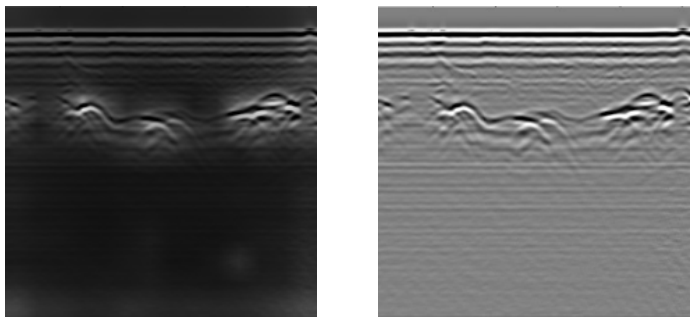
In the current study, a pre-trained ViT model has been considered, and only a fine-tuning of the last classification layers has been performed on the GPR profiles within a transfer learning approach, likewise in [20]. Specifically, the ViT-L16 model has been adopted in the current framework. Similar to BERT, also the ViT may appear in a different form according to the number of learnable parameters involved. The ViT-B16 or ViT-B32 are referred

to as the ViT-Base model, likely the BERT-base version, involving 86M trainable parameters. The ViT-L16 or ViT-L32 are referred to the ViT-Large model, likely the BERT-Large model, involving 307M learnable parameters. The numbers 16 or 32 indicate the number of patches in which the input images are partitioned. Referring to the architecture depicted in Figure 7, the ViT pre-trained model has been frozen and, after it, some final layers have been added to perform the classification task which undergoes the fine-tuning training. A fully connected dense layer with 2048 units has been added, followed by batch normalization and with a dropout probability of 0.5, followed again by a dense layer with softmax activation to provide classification into the binary classification output class analyzed at each level of the hierarchical scheme in Figure 3. According to [43], the pre-trained ViT model has been trained on the public ImageNet-21k, which is composed of 14 million images database with 224x224 pixel resolution distributed in 21843 classes, and then fine-tuned on the public ImageNet 2012 database which, on the other hand, contains 1.3 million images with 224x224 pixel resolution distributed among 1000 classes. Thus, the ImageNet-21k represent a superset of ImageNet [44] which is consequently denoted also as ImageNet-1k. The dataset of GPR profile images has been split, considering 90% of the total images per level to belong to the training set and the resulting 10% belonging to the test set. Since the previous section evidenced that the Fourier preprocessing does not provide substantial improvements in terms of classification accuracy of the model, the authors trained the ViT model only on the raw images without

Table 5: Hyperparameters adopted for ViT model.

Hyperparameters	Value
Mini-batch size	16
Learning rate	0.0001
Maximum number of epochs	20

applying any Fourier transform image pre-processing procedures. For the ViT model, the pre-trained model provided by [43] has been imported based on available python Keras implementation [45]. The input images have been resized to 224x224 pixels, and 16 patches are extracted from the input images accordingly to the ViT-L16 model, batch size has been fixed to 16, and a validation set of 10% of the training set has been used during the training process, the categorical cross-entropy loss [46] has been adopted, and the model for each level has been trained for 20 epochs. Similarly to the CNN model, also for the ViT model the authors adopted an empirical trial-end-error procedure to define the optimal choice of the hyperparameters, whose values are reported in Table 5. These hyperparameters were selected to still guarantee a fair evaluation and comparison with the previous CNN models and also to contain the training computational efforts. Practical criteria and suggestions for hyperparameters trial-and-error tuning schemes are presented e.g. in [24]. Despite its useful generalization properties, cross-validation has not been carried on in this case, since very high computational effort also for the fine-tuning only training approach. However, to control the overfitting, the callbacks with an early stopping criterion have been set, and the checkpoints save model parameters only if the performance of the model sub-



(a) (b)

Figure 8: Attention map visualization example provided by the ViT-L16 model. On the left side (a), the resized sample image belonging to C14 class is showing a concrete layer detachment problem; on the right side (b), the attention map provided by the ViT-L16 model on the same image highlights the defects' presence and its position.

stantially increases, i.e. with a relatively significant reduction of the loss, a substantial increase of accuracy and considering the metrics' trends on the validation set in an attempt to avoid overfitting issues. The results obtained on the test set are illustrated in Table 6 in terms of confusion matrices, precision, recall, and f1-score metrics [24]. According to a very first sight to the results shown in the Table, the AI-based automatic defects classification with the ViT model shows an impressive average accuracy of around 0.981 among all the levels of the hierarchical classification tree, substantially more significant than the three previous analysed cases. The noticeably high accuracy has rewarded the computational efforts reached on the test set, remembering that despite even without cross-validation, the necessary precautions have been considered for overfitting issues. The Discussion section 6 provides an extended discussion and comparisons of the current results.

At the end of the present section, the authors illustrate the great ad-

vantage of a model entirely entrusted to the self-attention mechanism. For instance, the reader may observe an example of self-attention maps in Figure 8, which has been extracted from the current ViT-L16 model for illustrative purposes. The neural model can enhance specific patterns in the image, which helps the classification task and darken the other less important parts. This provides a handy tool to enhance the interpretation of these DL approaches. Furthermore, the specific patterns highlighted from the attention maps may also be helpful for damage localization, especially in the depth direction. These results are encouraging, and this aspect, also considering defects localization, should be further explored in future studies to increase the capabilities of the AI-based road tunnels' automatic defects classification framework to assess the actual tunnel health state.

6. Results comparisons and discussion

In the previous sections, four different models have been trained and tested to propose an AI-based approach for automatic defects classification for road tunnel GPR indirect testing. The first model involves the CNN pre-trained ResNet-50 model trained on GPR samples images obtained by cropping the long profiles with constant step length, providing the results illustrated in Table 2. To compare ResNet-50 with more recent state-of-art convolutional architectures, the EfficientNet has also been trained, providing the results illustrated in Table 3. Furthermore, to improve the ResNet-50 model, two different techniques have been implemented. The foremost act

Table 6: ViT model classification results with original raw GPR profiles dataset in terms of confusion matrices expressed in absolute terms on the test set for Levels 1, 2a, 2b, 3, 4, 5, and 6.

Level 6								
	Predicted		Accuracy	Tot. images/class		Test set support	Precision	Recall	f1-score
True	C13	C14		C13	408	53	1.000	0.981	0.990
C13	52	1	0.9907	C14	672	55	0.982	1.000	0.991
C14	0	55		Total	1080	108			
Level 5									
	Predicted		Accuracy	Tot. images/class		Test set support	Precision	Recall	f1-score
True	C11	C12		C11	1108	115	0.991	1.000	0.996
C11	115	0	0.9954	C12	1080	104	1.000	0.990	0.995
C12	1	103		Total	2188	219			
Level 4									
	Predicted		Accuracy	Tot. images/class		Test set support	Precision	Recall	f1-score
True	C9	C10		C9	936	96	0.989	0.979	0.984
C9	94	2	0.9904	C10	2188	217	0.991	0.995	0.993
C10	1	216		Total	3124	313			
Level 3									
	Predicted		Accuracy	Tot. images/class		Test set support	Precision	Recall	f1-score
True	C7	C8		C7	900	95	1.000	1.000	1.000
C7	95	0	1.0000	C8	3124	308	1.000	1.000	1.000
C8	0	308		Total	4024	403			
Level 2b									
	Predicted		Accuracy	Tot. images/class		Test set support	Precision	Recall	f1-score
True	C5	C6		C5	574	56	0.763	0.804	0.783
C5	45	11	0.9457	C6	4024	404	0.973	0.965	0.969
C6	14	390		Total	4598	460			
Level 2a									
	Predicted		Accuracy	Tot. images/class		Test set support	Precision	Recall	f1-score
True	C3	C4		C3	3638	359	0.989	1.000	0.994
C3	359	0	0.9903	C4	492	54	1.000	0.926	0.962
C4	4	50		Total	4130	413			
Level 1									
	Predicted		Accuracy	Tot. images/class		Test set support	Precision	Recall	f1-score
True	C1	C2		C1	4130	401	0.952	0.948	0.950
C1	380	21	0.9542	C2	4598	472	0.956	0.960	0.958
C2	19	453		Total	8728	873			

on the preprocessing side by performing the bidirectional Fourier transform on the entire image dataset, obtaining a new dataset. This dataset was adopted to re-train the ResNet-50, providing results shown in Table 4. On the other hand, the authors acted on the neural model side, replacing the CNN model with a novel architecture based on the encoder part of the neural transformers. Finally, the vision transformer, ViT, has been trained with the original GPR image dataset, providing the overwhelming results reported in Table 6. For the sake of completeness, to inspect the overfitting issues occurrence, the training convergence curves have been reported in Appendix A. These curves illustrate the accuracy and loss trends for training and validation sets that occurred during the epochs or iterations. The curves for the considered convolutional models (ResNet-50, ResNet-50 with Fourier preprocessing, and EfficientNet), which have been trained on Matlab, have been reported concerning the training iterations. The term iteration refers to a single step performed with the gradient descent algorithm in minimizing the loss function by adopting the mini-batch. Since a cross-validation procedure has been employed for all the adopted convolutional models, to improve the clearness and readability of the convergence curves, the graphs have been arranged with the average curves for training accuracy, training loss, validation accuracy and validation loss among the various k-folds trained models. However, in order to provide more informative graphs without hiding any information due to the averaging operation, the authors provided the shaded areas around the average curves to represent the envelope of the maximum

and minimum curves obtained from the various k-folds models. From these convergence curves, it appears evident how the ResNet-50 model presents a generally good behavior without any overfitting evident phenomena, except for level 1 where the validation loss presents a slightly increasing trend from iteration 425. Even the EfficientNet present in general a good regular behavior in the training curves. However, both EfficientNet and ResNet-50 with Fourier pre-processed data present an evident overfitting behavior in level 2b. This may explain the reasons for the poor results obtained on the test set at this level. Moreover, the ResNet-50 model with Fourier pre-processed data presents slight evidence of overfitting in levels 1 and 4. On the other hand, the ViT model has been implemented in the Python environment and the convergence curves have been reported concerning the training epochs directly. It is worth noting that, despite a maximum number of 20 epochs being set for training the ViT model, the convergence curves stopped early in levels 1, 2a, 2b, and 3. This is because the option of early stopping was set and it was combined with the option which permits saving only the best-found models, providing the possibility to save computational resources if no further improvements occur. In general, the ViT model presents fairly good behavior without any overfitting issues. As a matter of fact, notwithstanding the validation loss showed a peak value in levels 1 and 2a for epochs 10 and 7 respectively, during the next training epochs, the model was able to improve again the learning phase and restore the desired descending trend. Moreover, it is worth noting that from the ViT convergence curves, level 2b

appears still moderately undertrained, and it may virtually be trained for more epochs to further reduce the loss, but requiring a considerable increase of computational effort with respect to the other levels.

With a deeper insight into the results, Figure 9 provides a general comparison of the overall average accuracy produced by the three approaches analyzed in the current study. As anticipated, the ViT transformers exhibit the best level of average accuracy, reaching 98.10%. In contrast, the worse value has been obtained with the Fourier transform preprocessing, limited to 85.6%. Although, from a theoretical point of view, the bidimensional Fourier transform provides a more efficient convolution operation in the Fourier domain, this was not compensated by the classification performance because of probable excessive data, i.e. information, compression in the input image. This provides similar sample images, slowing down the automatic recognition process with the CNN model. To provide the reader with a more comprehensive insight into the results of the above-mentioned tables, the accuracy values for every single level of the proposed hierarchical multi-level classification tree are depicted in Figure 10. The bar graph evidenced the accuracy results for each level for the four analyzed models in the current study. No particular trends are evidenced among the classification levels. However, level 2b exhibits the lowest accuracy value for all the three models taken separately. Level 2b deals with recognising the warning mix from the other defect types (cracks, anomalies, simple voids, excavation, and detachments). The reason behind this aspect may hide behind the class unbalance

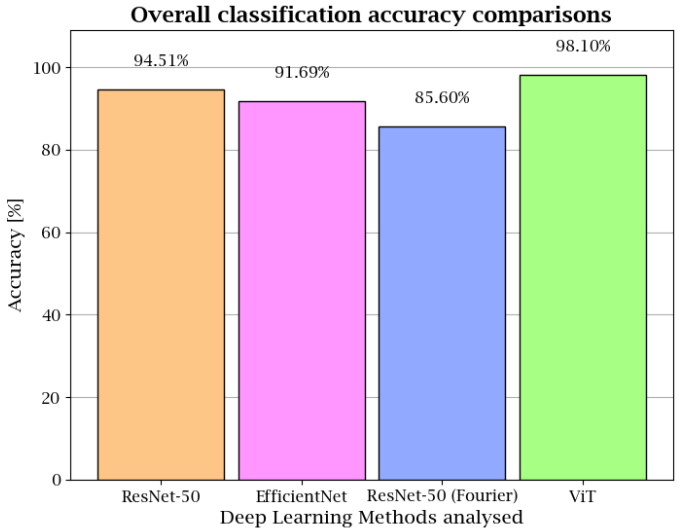


Figure 9: Overall accuracy comparison graph among the various DL trained models.

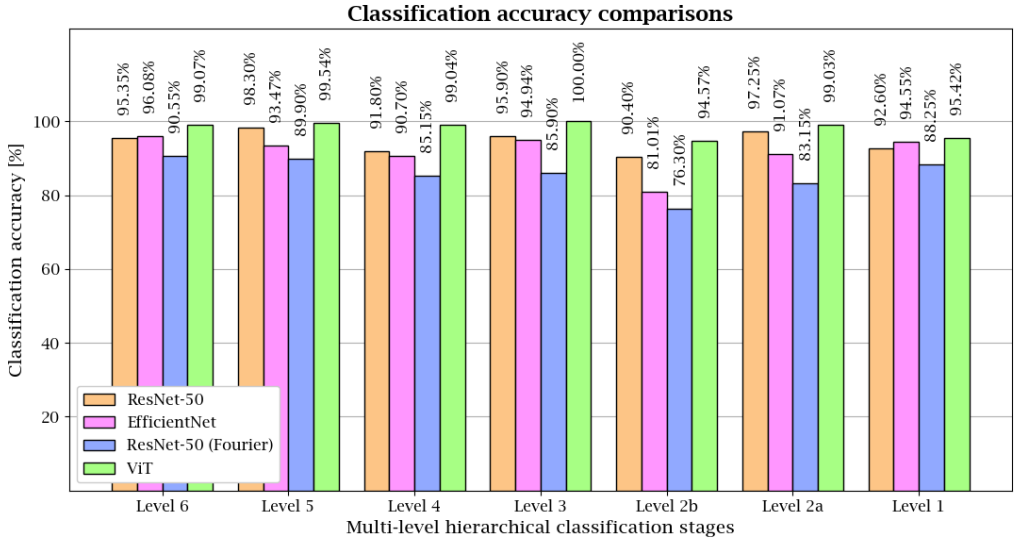


Figure 10: Accuracy comparison graph for each analyzed level in the hierarchical multi-level automatic defects classification.

among the samples belonging to the two different categories involved in the current binary classification. For example, class C5 (warning mix) contains only 574 against class C6, which includes 4024 samples. The class unbalance may be computed considering the absolute difference among the samples belonging to the two classes compared at each binary classification level. In this case, this difference is equal to 3450. For instance, on the contrary, the classes C11 and C12 provide a neglecting value of class unbalance, i.e. 28, since C11 contains 1108 samples against 1080 samples of class C12. The class unbalance issue is probably the main reason for the lower performances of the ViT in the level 2b only. In addition, this fact may also explain the moderate undertraining evidenced in the convergence curve for ViT in level 2b, on the same conditions of training settings (i.e. same hyperparameters) of the other levels. Nevertheless, the ResNet-50 models have been trained considering only a number of training samples at each binary classification level considering the class with the lower number of samples. This ensures that the class unbalance equals zero, preventing the model from being more prone always to classify the most populated class. However, a poor and deterministic choice of the training set in the most populated class may adversely affect the classification performances. For this reason, a random state and shuffling in the definition of the training set was set for each model of the 10 models trained for every classification level due to the 10-fold cross-validation procedure, convincing the authors to have accomplished their best even this last issue. Finally, the precision and recall metrics have been con-

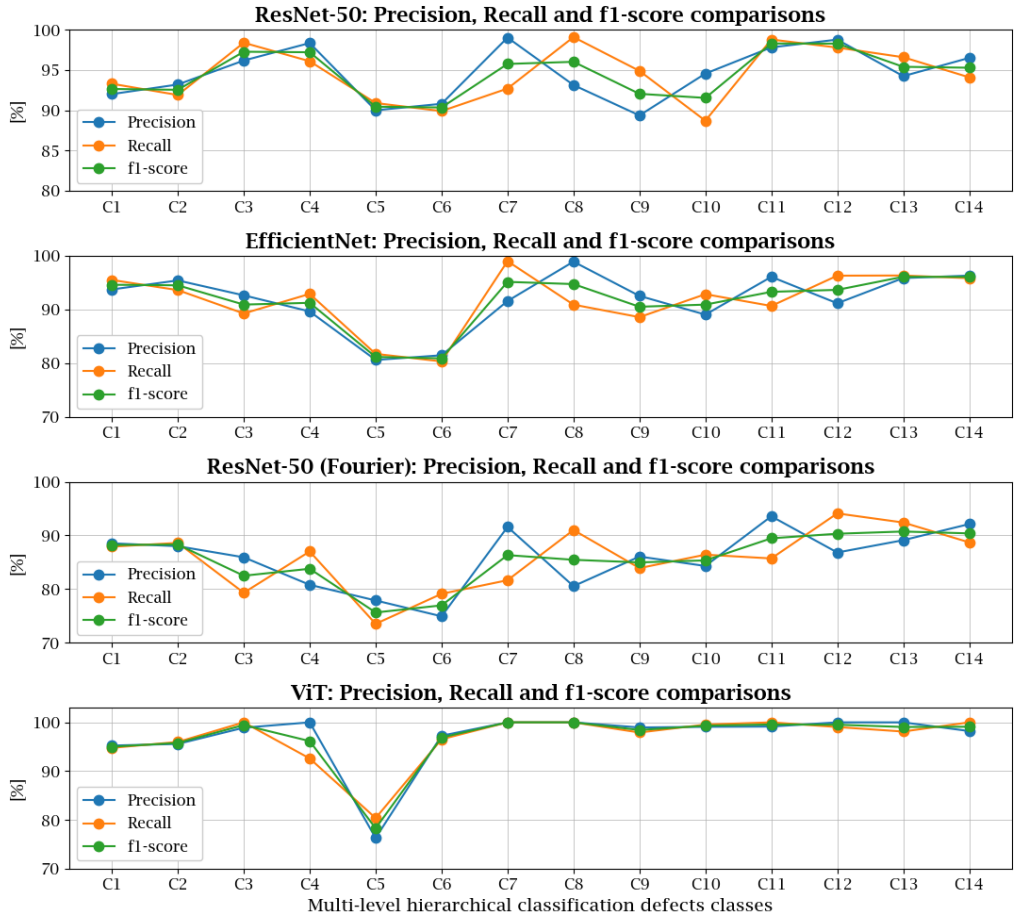


Figure 11: Accuracy comparison graph for each analyzed level in the hierarchical multi-level automatic defects classification.

sidered associated with the confusion matrices of the current trained models. The precision estimates the number of samples that were correctly classified in a certain class over the total number of samples which have been truly associated with that class [47, 48, 49, 46]

$$\text{Precision} = \frac{TP}{TP + TN} \tag{10}$$

where TP indicates the true positive number of samples and TN stands for the true negative ones. On the other side, the recall metric indicates the number of samples correctly associated with a certain class over the number of samples which actually truly belongs to that class [47, 48, 49, 46]

$$\text{Recall} = \frac{TP}{TP + FN} \tag{11}$$

The f1-score represents a sort of harmonic mean of precision and recall metrics, therefore it is widely used as a synthetic evaluation metric of the performances

$$\text{f1-score} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{12}$$

These three metrics are reported in Figure 11, to show their trends synthetically and graphically for the three analyzed models. As already evidenced previously, the above metrics are largely reduced for the ResNet-50 with Fourier preprocessing approach, and they suffer a noticeably reduction in the class C5, the one involved in the most unbalanced level, i.e. the level 2b. Specifically, the ViT model, which exhibits extensively outstanding performances, revealed a remarkable drastic drop in class C5. For a more direct comparison among the current four analyzed DL models, the f1-score has been appointed as a synthetic metric and depicted in Figure 12 for every class of the current hierarchical classification paradigm. In this Figure, the same conclusions can be retrieved more clearly and directly according to this

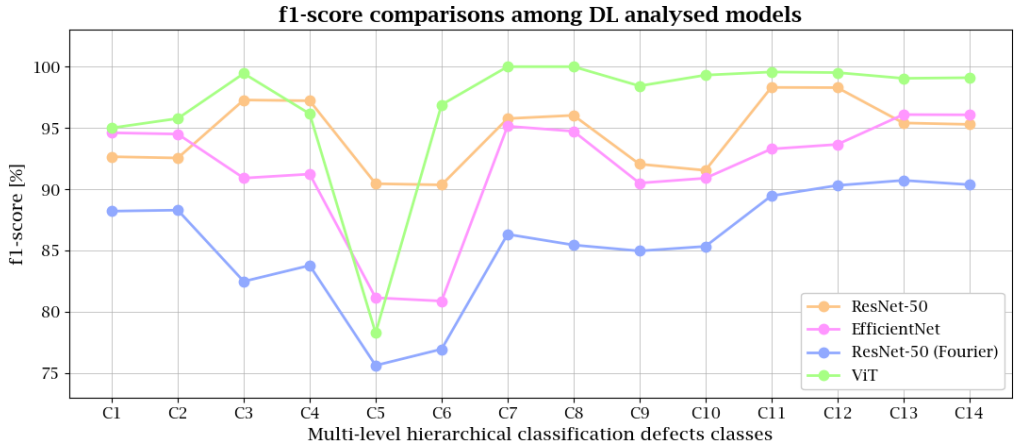


Figure 12: Accuracy comparison graph for each analyzed level in the hierarchical multi-level automatic defects classification.

synthetic and comprehensive graphical representation.

6.1. Neural models interpretability

During the last decades, with the widespread adoption of evermore sophisticated DL network architectures, in scientific community concerns about interpretability of the neural models arose, exacerbating the idea of neural networks as merely black boxes. In fact, it became fairly challenging to provide a clear explanation of what the network learned to classify an image in a specific certain class rather than another one. Thus, in [50], the authors pointed out that, leveraging the global average pooling layer in a certain manner, it is possible to highlight the powerful localization capabilities of CNN models, even when trained for classification purposes only and not specifically for e.g. object detection tasks. They introduced the concept of class activation maps (CAM), i.e. a visual representation of those portions of input images that

mainly contributes to the classification score for a given class. Since the final classification output score is in practice a weighted sum operation performed by a fully connected layer, similarly, the CAM is actually the representation of the activation maps following the last convolutional layer weighted by the weights of the final fully connected layer. Thereafter, other researchers e.g. [51] formalized a gradient-weighted CAM or e.g. the most recent gradient-free Score-CAM method [52]. In the present document, adopting the Matlab implementation of [53], the authors provided in Appendix B a visual inspection of CAM for an example image of the test set belonging to every single class of the proposed GPR defects' classification tree. The CAM are reported both for the models ResNet-50, EfficientNet, and ResNet-50 with Fourier pre-processing technique. The CAM visually pointed out a quite impressive successful learning of the ResNet-50 model with original images to focus on the more characteristic pattern of the GPR images. On the contrary, EfficientNet performed moderately worse than ResNet-50, presenting in general more dispersed activation maps. At first sight, the CAM for ResNet-50 on Fourier pre-processed images may appear as well quite dispersed, without any apparent consideration of the main frequency component pattern. However, considering two classes level by level, e.g. comparing C3 and C4 in level 2a, it seems that the network mainly focuses on central regions of the image belonging to class C4 and, in a complementary manner, it focuses on extremum areas of the images for class C3. A similar pattern of CAM is more evident in C5 and C6 in level 2b, in C7 and C8 in level 3, and in C9 and

C10 in level 4. On the other side, another merit that distinguishes the neural transformers-based architecture, such as the herein analysed ViT model, is the adoption of the attention mechanism. Therefore, the transformers models have already intrinsically incorporated an essential interpretability tool to give the user the possibility to inspect what the network learned, without requiring any further post-processing procedure as before, sometimes time and resources-consuming. Thus, in the Appendix B, the attention maps for ViT reveal tremendous localization capabilities for those parts of the input images which mainly contributed to defining the right output classes, crowning the neural transformers models as one of the nowadays more naturally and reliably interpretable DL models.

7. Conclusions and future remarks

In the present work, the authors proposed an innovative framework to perform an automatic road tunnel's defects classification based on indirect testing with GPR profiles. The long images obtained from GPR linings profiles have been split into sample images 5.00 m long along the progressive distance from the beginning of the road tunnel. This dataset has been investigated by analyzing different AI-based approaches rooted in the Deep Learning (DL) field. The CNN ResNet-50 was first considered working directly on the raw images of GPR data according to the hierarchical multi-level binary classification tree exhibited in Figure 3, providing encouraging results characterized by an average accuracy of 0.945. The authors further investigated two

other variants. The foremost involved maintaining the same neural model but providing a preprocessing technique on the input data according to the bidimensional Fourier transform applied to digital images. Notwithstanding this technique theoretically appearing to improve the efficiency of the convolution operation in the Fourier domain, this technique could not overcome the accuracy of the ResNet-50 trained with the original raw GPR image data, with an average accuracy of 0.856 among various classification levels. It is plausible that the Fourier transform performed an excessive information compression in the input data, reducing the global accuracy among the different levels of the proposed hierarchical multi-level classification technique. On the other hand, the second variant to improve the standard ResNet-50 approach uses the current state-of-art method in the image processing based on the neural transformer architecture. Based on the encoder part of the transformer, the vision transformer provides overwhelming results for the automatic road tunnel defects classification paradigm. The ViT reached an average accuracy of 0.981. The precision, recall and f1-score have also been compared among the three different analyzed techniques, providing good results, but for the C5 class, where the warning mix probably contains features too much similar to the class C6 in level 2b. However, the results mentioned above provide compelling evidence for such an automatic proposed approach's effectiveness in assessing the road tunnel's current health state based on indirect GPR testing. Furthermore, the ViT model can highlight in the attention map the most considerable characteristic patterns in the input images enhancing

their brightness and darkening those parts deemed irrelevant. This feature suggested an encouraging research path to the author, which should be further explored in future studies, involving the defects' localization purposes in conjunction with their identification and categorization. This aspect should impressively increase the capabilities of the AI-based road tunnels automatic defects classification paradigm for road tunnel structural health monitoring. The main limitation of adopting the neural techniques investigated in the present document is probably the high computational effort required, especially for the training phase. However, the recall of the after-training models do not require the same computational effort as the training phase. Nevertheless, since these models demand quite high memory to be initialized before being fully operative, this fact may severely limit the adoption of such techniques for the real-time monitoring phase on small portable electronic devices, such as emerging technologies, e.g. likewise internet of things (IoT) edge devices. Nowadays, for in situ real-time or near-real-time GPR inspections, the operators may still necessitate fairly high computational capabilities offered, e.g., by a sufficiently powerful laptop at least.

Acknowledgments

Computational resources provided by hpc@polito (http://www.hpc.polito.it).

References

[1] M. Hu, Y. Liu, V. Sugumaran, B. Liu, J. Du, Automated structural defects diagnosis in underground transportation tunnels using semantic technologies, Automation in Construction 107 (2019) 102929.

[2] F. Sandrone, V. Labiouse, Identification and analysis of swiss national road tunnels pathologies, Tunnelling and Underground Space Technology 26 (2) (2011) 374–390.

[3] C. Moret, Safety-related regulations in french road tunnels, in: PROCEEDINGS OF THE FOURTH INTERNATIONAL CONFERENCE ON SAFETY IN ROAD AND RAIL TUNNELS, HELD MADRID, SPAIN, 2-6 APRIL 2001, 2001.

[4] G. Marasco, M. M. Rosso, S. Aiello, A. Aloisio, G. Cirrincione, B. Chiaia, G. Marano, Ground Penetrating Radar Fourier Pre-processing for Deep Learning Tunnel Defects' Automated Classification, 2022, pp. 165–176. doi:10.1007/978-3-031-08223-8_14.

[5] V. Dimitrova, M. O. Mehmood, D. Thakker, B. Sage-Vallier, J. Valdes, A. G. Cohn, An ontological approach for pathology assessment and diagnosis of tunnels, Engineering Applications of Artificial Intelligence 90 (2020) 103450.

[6] R. Montero, J. G. Victores, S. Martinez, A. Jardón, C. Balaguer, Past,

present and future of robotic tunnel inspection, Automation in Construction 59 (2015) 99–112.

[7] A. Benedetto, F. Tosti, L. B. Ciampoli, F. D'amico, An overview of ground-penetrating radar signal processing techniques for road inspections, Signal processing 132 (2017) 201–209.

[8] Y. Zan, Z. Li, G. Su, X. Zhang, An innovative vehicle-mounted gpr technique for fast and efficient monitoring of tunnel lining structural conditions, Case Studies in Nondestructive Testing and Evaluation 6 (2016) 63–69.

[9] F. Daneshgaran, L. Zacheo, F. D. Stasio, M. Mondin, Use of deep learning for automatic detection of cracks in tunnels: prototype-2 developed in the 2017–2018 time period, Transportation research record 2673 (9) (2019) 44–50.

[10] R. Elvik, Road safety inspections: safety effects and best practice guidelines, Transportøkonomisk institutt, 2006.

[11] C. Balaguer, R. Montero, J. Victores, S. Martínez, A. Jardón, Towards fully automated tunnel inspection: A survey and future trends, in: ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction, Vol. 31, IAARC Publications, 2014, p. 1.

[12] E. Cardarelli, C. Marrone, L. Orlando, Evaluation of tunnel stability using integrated geophysical methods, Journal of Applied Geo-

physics 52 (2) (2003) 93–102. doi:https://doi.org/10.1016/S0926-9851(02)00242-2.

[13] C. Anitescu, E. Atroshchenko, N. Alajlan, T. Rabczuk, Artificial neural network methods for the solution of second order boundary value problems, Computers, Materials and Continua 59 (1) (2019) 345–359.

[14] E. Samaniego, C. Anitescu, S. Goswami, V. M. Nguyen-Thanh, H. Guo, K. Hamdia, X. Zhuang, T. Rabczuk, An energy approach to the solution of partial differential equations in computational mechanics via machine learning: Concepts, implementation and applications, Computer Methods in Applied Mechanics and Engineering 362 (2020) 112790.

[15] T. Dawood, Z. Zhu, T. Zayed, Deterioration mapping in subway infrastructure using sensory data of gpr, Tunnelling and Underground Space Technology 103 (2020) 103487. doi:https://doi.org/10.1016/j.tust.2020.103487.

[16] W. Al-Nuaimy, Y. Huang, M. Nakhkash, M. Fang, V. Nguyen, A. Eriksen, Automatic detection of buried utilities and solid objects with gpr using neural networks and pattern recognition, Journal of Applied Geophysics 43 (2) (2000) 157–165. doi:https://doi.org/10.1016/S0926-9851(99)00055-5.

[17] C. Feng, H. Zhang, S. Wang, Y. Li, H. Wang, F. Yan, Structural damage

detection using deep convolutional neural network and transfer learning, KSCE Journal of Civil Engineering 23 (10) (2019) 4493–4502.

[18] B. Chiaia, G. Marasco, S. Aiello, Deep convolutional neural network for multi-level non-invasive tunnel lining assessment, Frontiers Of Structural And Civil Engineering (2022). doi:https://doi.org/10.1007/s11709-021-0800-2.

[19] M. Tan, Q. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, in: International conference on machine learning, PMLR, 2019, pp. 6105–6114.

[20] L. Tanzi, A. Audisio, G. Cirrincione, A. Aprato, E. Vezzetti, *Vision Transformer for femur fracture classification* (2021). arXiv:2108.03414.

[21] F. Borghetti, A. Frassoldati, M. Derudi, I. Lai, C. Trinchini, Resilience and emergency management of road tunnels: The case study of the san rocco and stonio tunnels in italy, Saf. Secur. Eng. IX 1 (2021) 81–92.

[22] C. Anitescu, E. Atroshchenko, N. Alajlan, T. Rabczuk, Artificial neural network methods for the solution of second order boundary value problems, Computers, Materials & Continua 59 (1) (2019) 345–359. doi:10.32604/cmc.2019.06641.

[23] H. Guo, X. Zhuang, T. Rabczuk, A deep collocation method for the

bending analysis of kirchhoff plate, Computers, Materials & Continua 59 (2) (2019) 433–456. doi:10.32604/cmc.2019.06660.

[24] A. Géron, Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems, " O'Reilly Media, Inc.", 2019.

[25] D. H. Hubel, T. N. Wiesel, Receptive fields of single neurones in the cat's striate cortex, The Journal of physiology 148 (3) (1959) 574.

[26] A. S. B. Reddy, D. S. Juliet, Transfer learning with resnet-50 for malaria cell-image classification, in: 2019 International Conference on Communication and Signal Processing (ICCSP), IEEE, 2019, pp. 0945–0949.

[27] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778. doi:10.1109/CVPR.2016.90.

[28] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, International journal of computer vision 115 (3) (2015) 211–252.

[29] J. Markoff, For web images, creating new technology to seek and find, New York Times (2012).

[30] MATLAB version 9.10.0.1649659 (R2021a) Update 1, Natick, Massachusetts, 2021.

[31] W. Rawat, Z. Wang, Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review, Neural Computation 29 (9) (2017) 2352–2449. doi:https://doi.org/10.1162/neco_a_00990.

[32] J. Gareth, W. Daniela, H. Trevor, T. Robert, An introduction to statistical learning: with applications in R, Spinger, 2013.

[33] M. Thompson, Digital image processing by rafael c. gonzalez and paul wintz, Leonardo 14 (3) (1981) 256–257.

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

[35] A. K. Jain, Fundamentals of digital image processing, Prentice-Hall, Inc., 1989.

[36] C. R. Farrar, K. Worden, Structural health monitoring: a machine learning perspective, John Wiley & Sons, 2012.

[37] R. Fisher, S. Perkins, A. Walker, E. Wolfart, Hypermedia image processing reference, England: John Wiley & Sons Ltd (1996) 118–130.

[38] J. S. Lim, Two-dimensional signal and image processing, Englewood Cliffs (1990).

[39] R. E. Woods, R. C. Gonzalez, Digital image processing third edition (2021).

[40] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[41] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[42] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16x16 words: Transformers for image recognition at scale, in: International Conference on Learning Representations, 2021.

[43] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, arXiv preprint arXiv:2010.11929 (2020).

[44] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE conference on computer vision and pattern recognition, Ieee, 2009, pp. 248–255.

[45] F. Morales, et al., vit-keras, keras implementation of vit (vision transformer) (2015).
URL `https://github.com/faustomorales/vit-keras`

[46] C. C. Aggarwal, et al., Neural networks and deep learning, Springer 10 (2018) 978–3.

[47] S. Raschka, Python Machine Learning, Packt Publishing - ebooks Account, 2015.

[48] I. Goodfellow, Y. Bengio, A. Courville, Deep learning, MIT press, 2016.

[49] J. P. Mueller, L. Massaron, Deep Learning for dummies, John Wiley & Sons, 2019.

[50] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2921–2929.

[51] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 618–626.

[52] H. Wang, M. Du, F. Yang, Z. Zhang, Score-cam: Improved visual explanations via score-weighted class activation mapping (2019).

[53] K. Itakura, Explainable-ai-interpreting-the-classification-performed-by-deep-learning-with-lime-using-matlab (2021).
URL https://github.com/KentaItakura/Explainable-AI-interpreting-the-classification-performed-by-deep-learning-wit

Appendix A. Convergence Curves

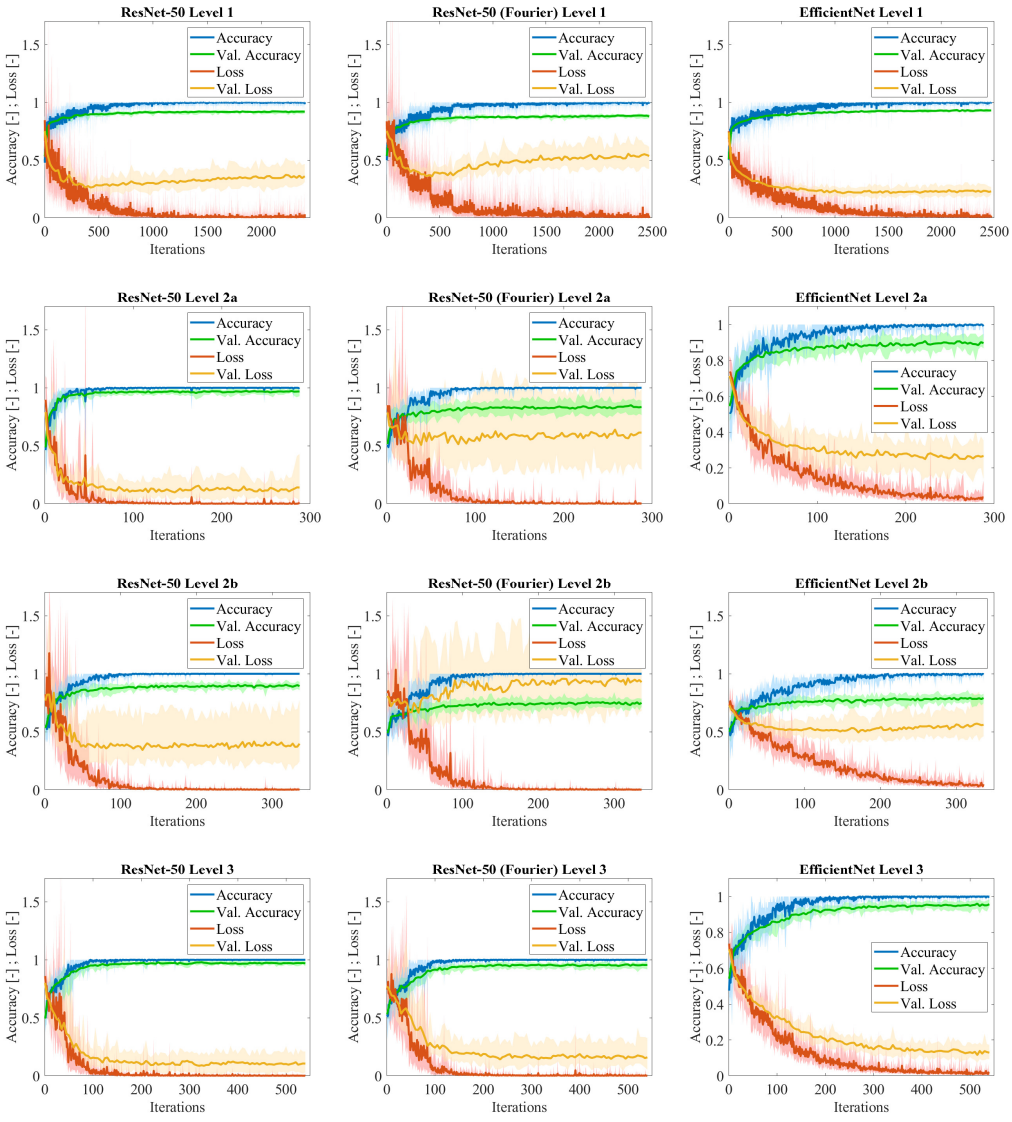


Figure A.13: Accuracy and loss convergence curves during training iterations for ResNet-50, ResNet-50 with Fourier pre-processing and EfficientNet.

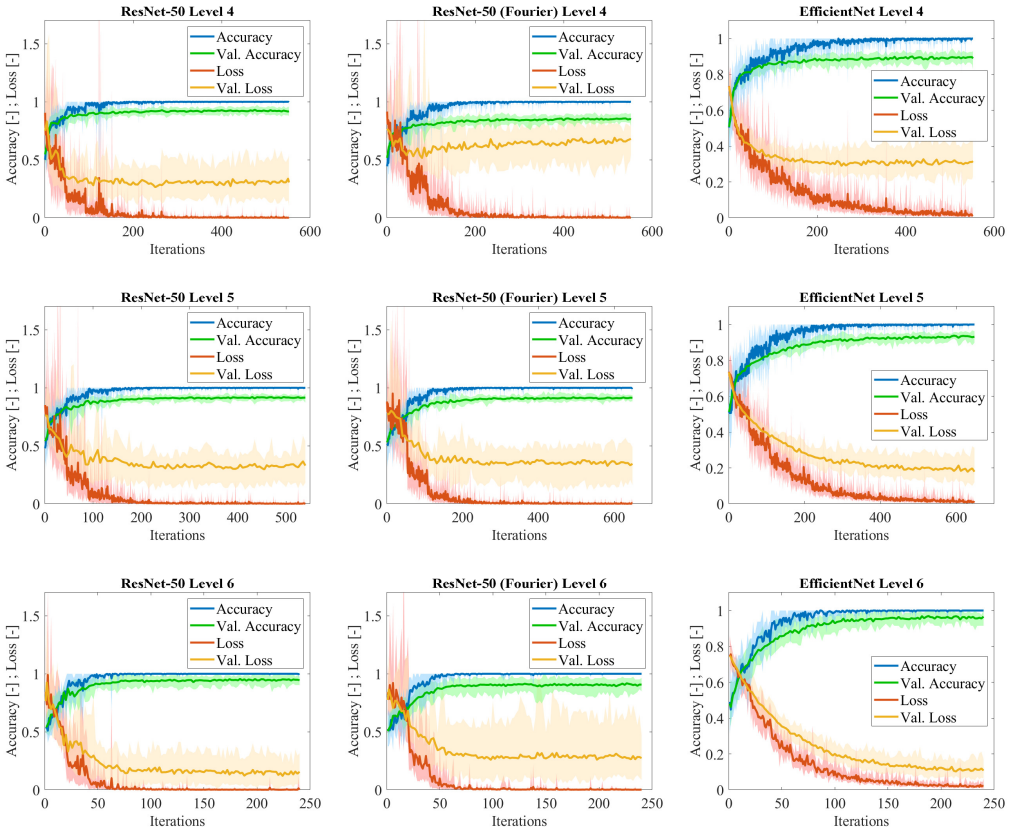


Figure A.13: Accuracy and loss convergence curves during training iterations for ResNet-50, ResNet-50 with Fourier pre-processing and EfficientNet.

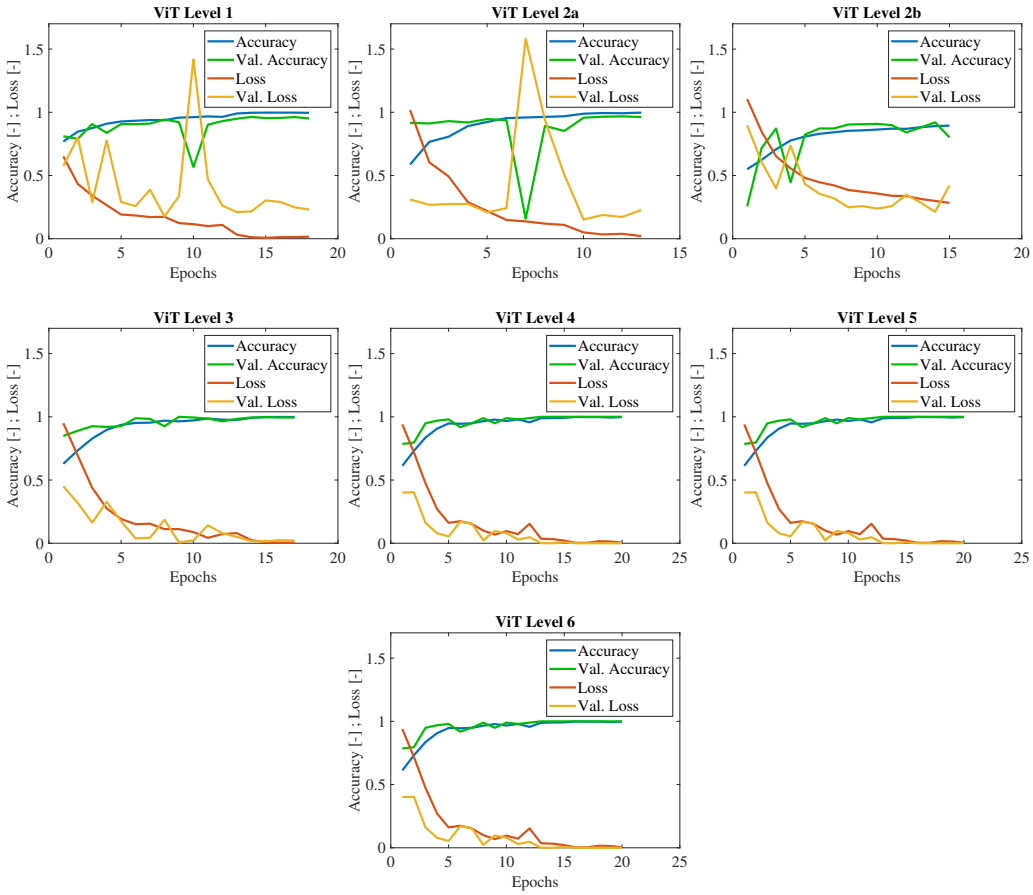


Figure A.14: Accuracy and loss convergence curves during training epochs for ViT model.

Appendix B. Class activation maps and attention maps

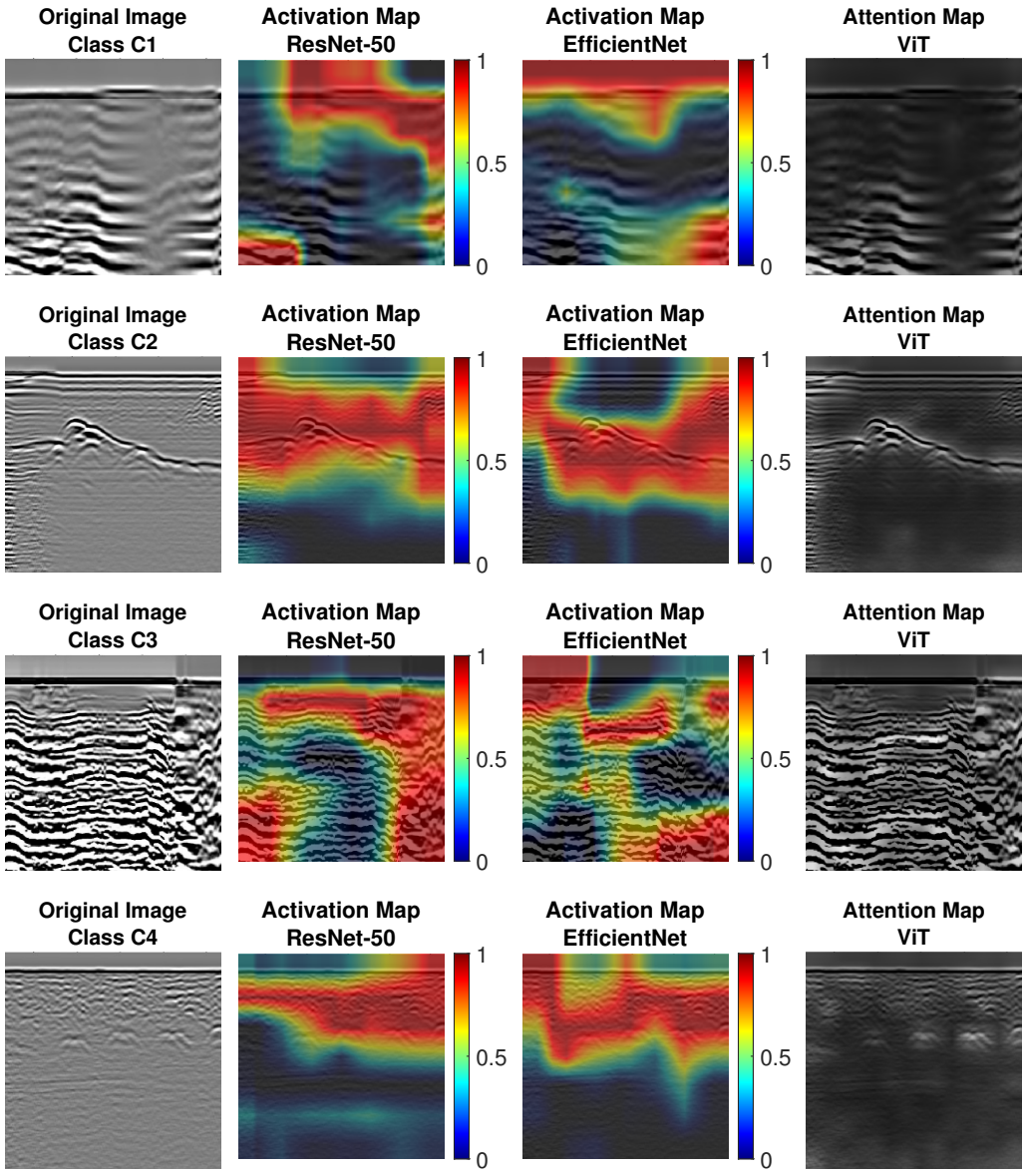


Figure B.15: Class activation maps for ResNet-50 and EfficientNet, and attention maps for ViT compared to original images for each class.

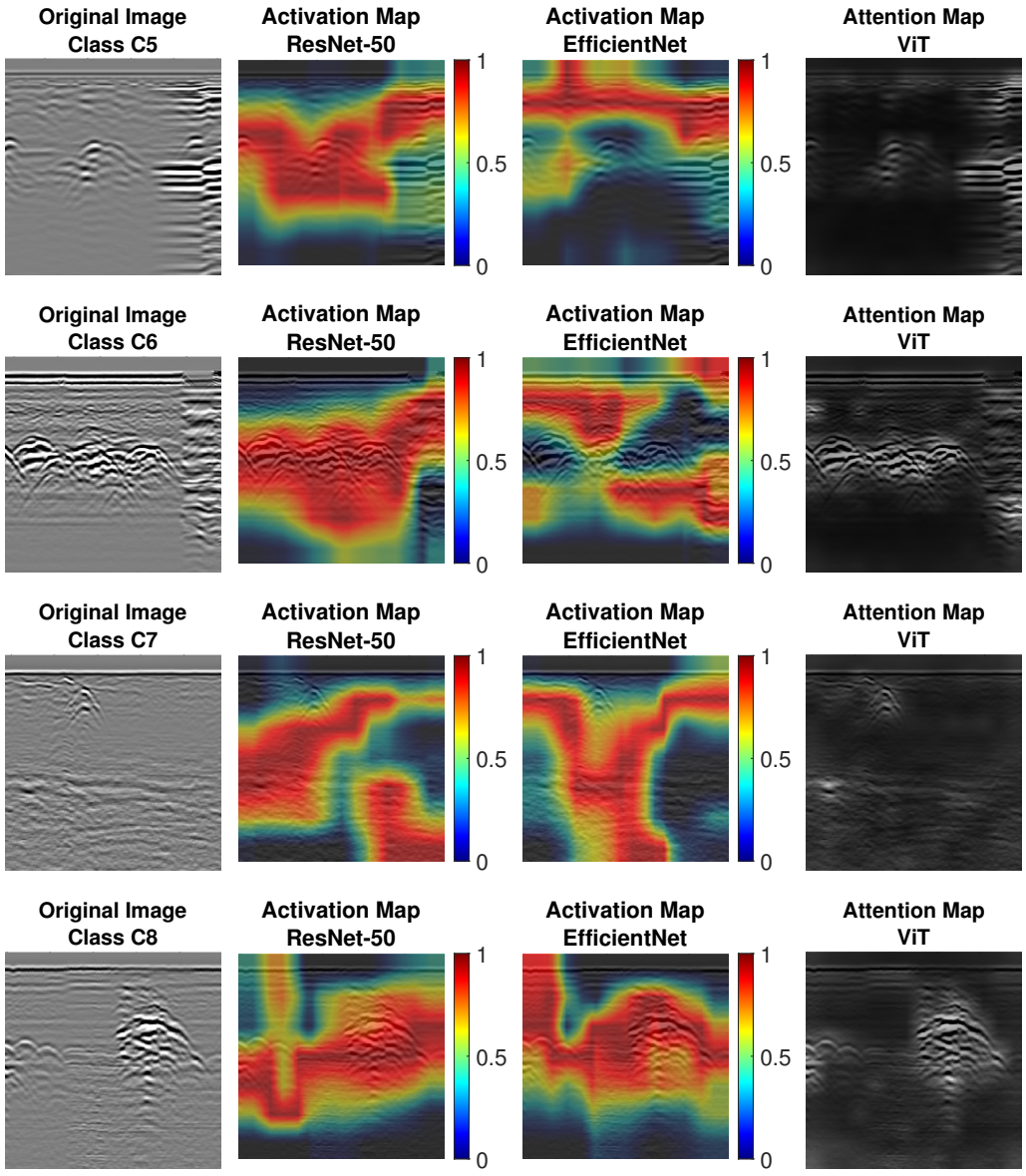


Figure B.15: Class activation maps for ResNet-50 and EfficientNet, and attention maps for ViT compared to original images for each class.

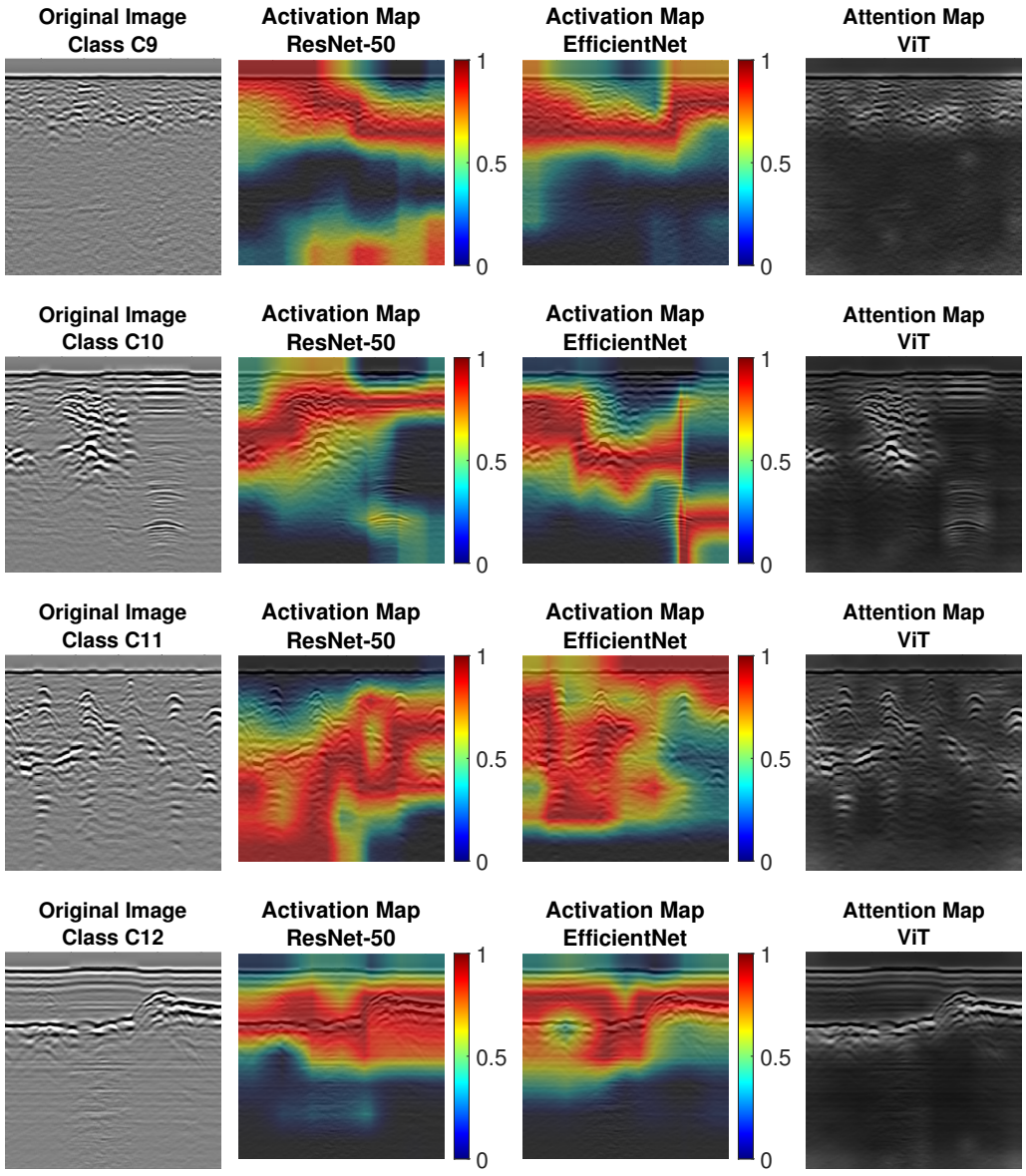


Figure B.15: Class activation maps for ResNet-50 and EfficientNet, and attention maps for ViT compared to original images for each class.

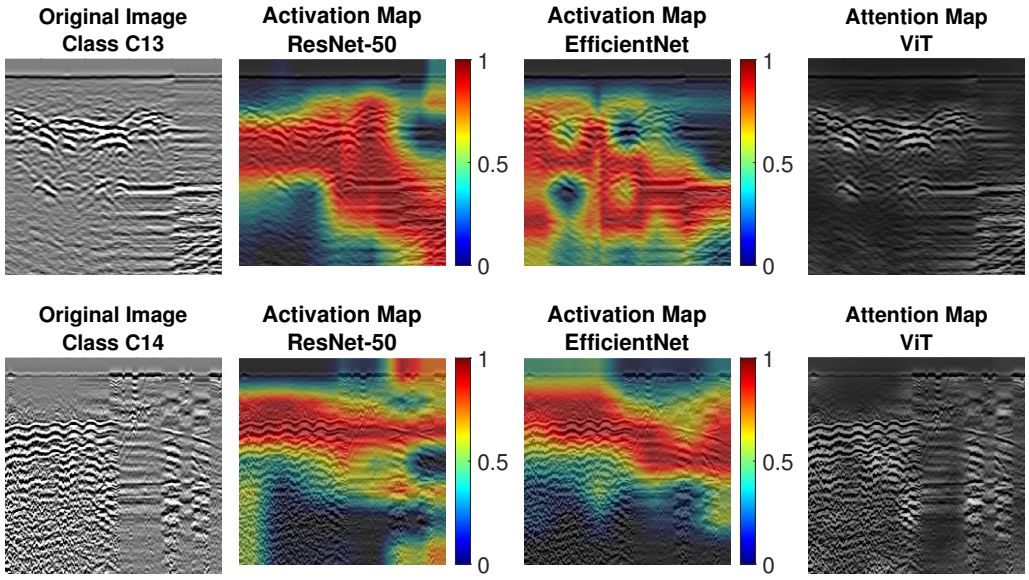


Figure B.15: Class activation maps for ResNet-50 and EfficientNet, and attention maps for ViT compared to original images for each class.

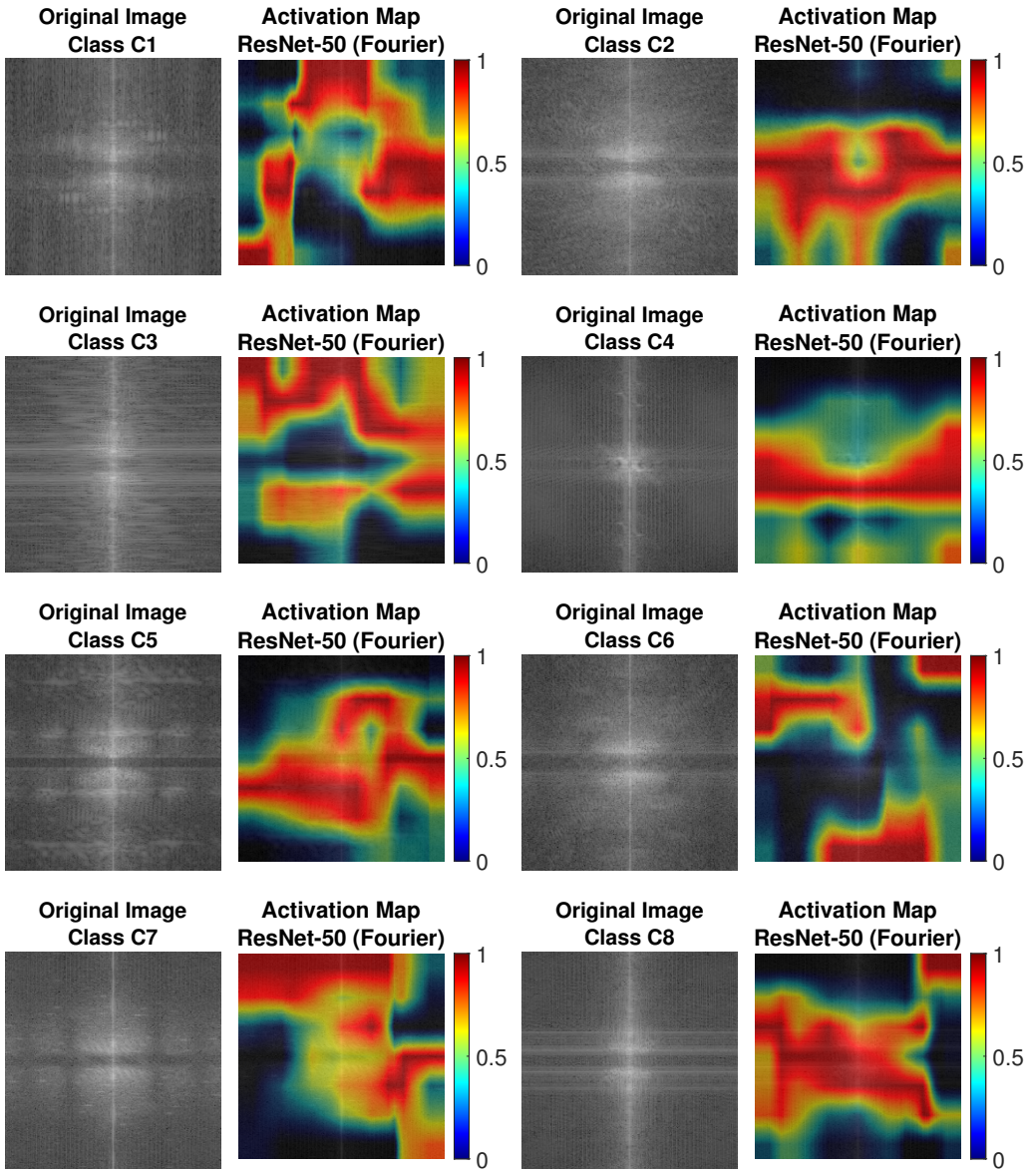


Figure B.15: Class activation maps for ResNet-50 with Fourier pre-processing compared to Fourier pre-processed images for each class.

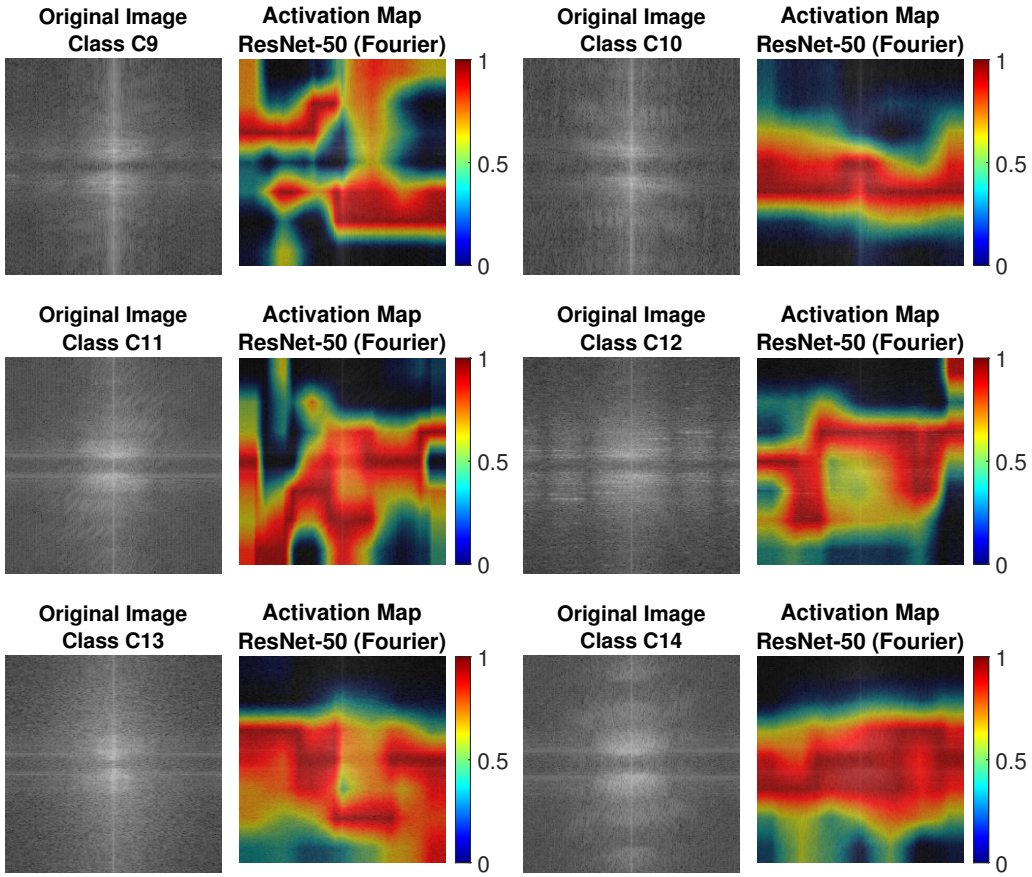


Figure B.15: Class activation maps for ResNet-50 with Fourier pre-processing compared to Fourier pre-processed images for each class.