



# Generalized Morse Wavelets parameter selection and transfer learning for pavement transverse cracking detection

Erick Axel Martinez-Ríos\*, Rogelio Bustamante-Bello, Sergio A. Navarro-Tuch

Tecnológico de Monterrey, School of Engineering and Sciences, Mexico City, 14380, Mexico



## ARTICLE INFO

### Keywords:

Continuous wavelet transform  
Generalized Morse wavelets  
Mother wavelet selection  
Transfer learning  
Convolutional neural networks  
Pavement cracking detection

## ABSTRACT

Monitoring road surface anomalies is crucial to avoid potential harm to pedestrians, vehicles, and vehicle users. These factors have motivated the development of systems that employ signal processing and machine learning algorithms to detect these irregularities automatically by sensing the vehicle's vibration. The wavelet transform (WT) has been frequently used for signal representation and feature extraction to solve signal classification problems. However, selecting a mother wavelet to perform the WT is challenging due to the need for methods that help to validate the time-frequency representation. Furthermore, machine learning algorithms require a sizeable sample size to train them, and setting their parameters is considered a time-consuming task. This paper proposes to compute the Continuous Wavelet Transform (CWT) through Generalized Morse Wavelets (GMWs) on vehicle's vertical acceleration data for pavement transverse cracking detection. The parameters gamma and beta of the GMWs and the CWT's voices per octave (VPO) were selected based on two metrics, the Heisenberg area and the Average Reconstruction Mean Squared Error. The scalogram generated through the CWT and GMWs was used to fine-tune pre-trained convolutional neural networks (CNNs) through transfer learning, such as GoogLeNet, SqueezeNet, and ResNet18. The proposed methodology set 1 VPO for the CWT, while the parameters gamma and beta of GMWs were set as 3 and 1.3333, respectively. The 5-fold cross-validation results of fine-tuning the CNNs showed that SqueezeNet provided a higher average validation sensitivity ( $84.3290 \pm 5.5704$ ) than GoogLeNet and ResNet18; however, the average validation specificity of SqueezeNet ( $92.7061 \pm 7.2403$ ) was lower.

## 1. Introduction

Maintaining the road infrastructure in an optimal condition is crucial to correct and prevent the possible degradation of the road surface (Lekshmpathy et al., 2021; Varona et al., 2020). Road surface anomalies can be a product of pavement age, weather conditions, traffic loads, and lack of maintenance (Martinez-Ríos et al., 2022). This degradation could include cracks, potholes, speed bumps deterioration, and grooves. In addition, these deviations present in the road surface could cause potential harm to vehicles, vehicle users, and pedestrians (Celaya-Padilla et al., 2018). Likewise, they can affect the energy consumption of automobiles. Furthermore, the maintenance of roads has financial costs for the government to repair these irregularities (Martinez-Ríos et al., 2022). The traditional approach in which the road's surface is monitored is through human observation performed by employing Pavement Condition Index Surveys (PCI) (ASTM, 2023). However, these surveys are prone to human errors and risk the health of road technicians (Martinelli et al., 2022; Shaghil and Khalafallah, 2018; Yang and Zhou, 2021). The above factors have motivated the development of alternative methods to automatically detect and classify these road defects.

In this regard, recent studies have proposed using classification techniques based on signal processing and machine learning algorithms to detect or classify these anomalies by sensing the vehicle's vibration or through image processing and computer vision techniques (Sattar et al., 2021; Kim et al., 2022). For example, vibration-based methods detect and classify road surface anomalies by sensing the vehicle's vertical acceleration while it passes over the anomaly with the help of accelerometers (Sattar et al., 2018). Vibration-based structural damage has been developed since structural damage can change the dynamical characteristics of structures, such as natural frequencies, mode shapes, and damping ratios (Yang and Zhou, 2021). This approach has gained interest due to its cost-effectiveness, the low computational power needed to process accelerometer data, and potential real-time execution.

A typical strategy of vibration-based methods for road surface anomaly detection and classification is to employ classical signal processing techniques to extract features or statistics from a fixed time window of the vibration signal by considering different signal representations. These signal representations typically include the time

\* Corresponding author.

E-mail addresses: [A01331212@tec.mx](mailto:A01331212@tec.mx) (E.A. Martinez-Ríos), [rbustama@tec.mx](mailto:rbustama@tec.mx) (R. Bustamante-Bello), [snavarro.tuch@tec.mx](mailto:snavarro.tuch@tec.mx) (S.A. Navarro-Tuch).

domain, frequency domain, and time–frequency domain representations (Celaya-Padilla et al., 2018; Andrades et al., 2020; Ferjani and Ali Alsaif, 2022). In addition, standard features include the computation of statistical moments in both the time and frequency domains, minimum and maximum values, root-mean-square, median, or range values. Nevertheless, these diverse sets of signal representations and features that have been proposed made it challenging to select a particular representation to extract features to detect and classify specific road surface anomalies. Thus, a typical approach is to compute and test different sets of features and choose the set that provides the best performance to solve the classification task (Martínez-Ríos et al., 2022).

In the case of time–frequency representations, one of the main challenges that involve their use is to select the appropriate type of method (e.g., Short-Time Fourier Transform (STFT), Wavelet Transform, or Hilbert–Huang Transform) and their parameterization to represent the time–frequency content of a signal correctly (Boashash, 2016). The above is influenced by the lack of methodologies and systematic frameworks that help to validate the obtained time–frequency representation and the diverse set of parameters that can be controlled from the available techniques (Hlawatsch and Auger, 2013).

For instance, in the case of the STFT, the challenge is to select the type of window and its length that best represent the time–frequency content of a signal, or in the case of the wavelet transform, the challenge lies in the adequate selection of a mother wavelet used to perform the signal decomposition (Durak and Arikán, 2003; Guo et al., 2022; Rhif et al., 2019). The above is also associated with the diverse wavelet types available in the literature, such as the Morlet, Daubechies, Meyer, Shannon, and Haar wavelets, which makes it challenging to select one for a particular application (Lilly and Olhede, 2012). Moreover, a typical method is to employ the STFT spectrogram or the Continuous Wavelet Transform (CWT) scalogram to solve classification tasks by using them as input into convolutional neural networks (CNNs). However, there is a lack of discussion on how the parametrizations of time–frequency methods should be defined based on the characteristics of the signal under study as examined by Guo et al. (2022) and Martínez-Ríos et al. (2023).

Moreover, different machine learning techniques have been considered to detect and classify road surface anomalies based on vehicle vibrations, ranging from classical machine learning algorithms (e.g., random forest, logistic regression, and support vector machines) to deep learning techniques such as CNNs or recurrent neural networks. These methods and algorithms have achieved relatively high accuracy, as discussed by Martínez-Ríos et al. (2022). Nevertheless, there is a lack of publicly available datasets, and the experimental settings reported in the literature are heterogeneous, making it difficult to compare the results that authors have reported (Martínez-Ríos et al., 2022). Additionally, machine learning techniques rely upon large quality labeled datasets to train and validate the algorithms. However, collecting sufficient quality training data could be costly, challenging, and inefficient (Zhou et al., 2023). Furthermore, setting an adequate architecture of deep learning techniques can be time-consuming since their parameter setting is viewed as an art and could suffer from overfitting when trained with small datasets (Pasqualotto and Zigliotto, 2022; Li et al., 2022a).

This paper proposes to employ the Generalized Morse Wavelets (GMWs) and select its parameters based on two metrics, the Heisenberg area, and the Average Reconstruction Mean Squared Error (ARMSE), to perform the CWT on vehicles' vertical acceleration data for pavement transverse cracking detection. GMWs is a family of analytical wavelets proposed by Lilly and Olhede (2012), which provides a systematic framework to select a wavelet for general-purpose use. Furthermore, the obtained scalogram representation was used to fine-tune pre-trained CNNs following the transfer learning framework. Transfer Learning was chosen to avoid requiring a large dataset and setting the CNN parameters from zero, which are typical disadvantages of deep neural networks (Li et al., 2022a; Wan et al., 2021). Hence, this study focuses

on two major aspects. The first concerns selecting an appropriate wavelet to generate a scalogram of vehicle vertical acceleration signals associated with pavement transverse cracking and pavement sections without cracks. The second is to classify the generated scalograms of the vehicle's vertical acceleration signals in pavement sections with transverse cracks and without cracks by fine-tuning pre-trained CNNs. As far as the authors' knowledge, such a study has not been carried out. A general overview of the methodology followed in this paper is presented in Fig. 1. The main contributions of this study are listed below.

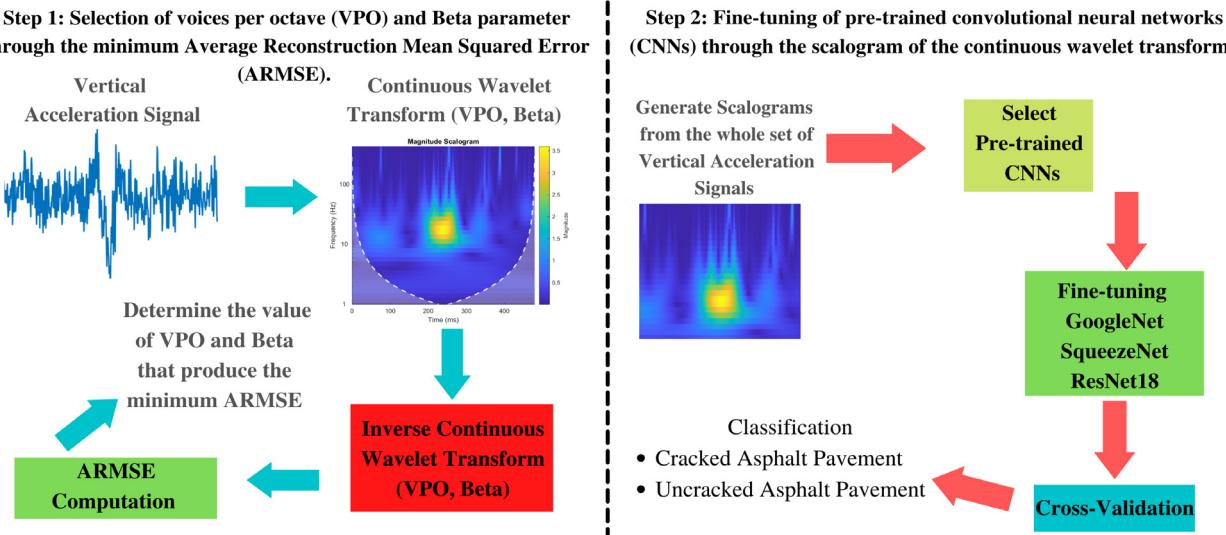
- The Heisenberg area and the ARMSE are used as reference metrics to select the parameters of the GMWs and the number of voices per octave (VPO) of the CWT and thus provide a basis for selecting a wavelet to compute the CWT. The ARMSE is used for setting the value of beta of the GMWs and the number of VPO. The Heisenberg area is used to define the value of gamma of GMWs.
- GMWs are employed to compute a scalogram of the vehicle's vertical accelerometer data used as input to CNNs for pavement transverse cracking detection.
- Transfer Learning is used for pavement transverse cracking detection through CNNs based on the vehicle's vertical acceleration data.
- Pre-trained CNNs such as GoogLeNet, SqueezeNet, and ResNet18 are fine-tuned using 5-fold cross-validation to apply transfer learning for pavement transverse cracking detection using the scalograms of vehicle's vertical acceleration signals as training data.

The rest of this paper is organized as follows. Section 2 shows the research related to the present study. Section 3 presents the dataset, a general overview of the background of the algorithms employed, and the methodology used to carry out the present study. Section 4 presents the results obtained via the proposed methodology, while Section 5 shows the analysis and discussion of the results. Section 6 discusses the limitations of the study. Finally, Section 7 shows the conclusions and future research directions.

## 2. Literature review

This section presents a summary and brief analysis of studies employing time–frequency representations for road surface anomaly detection and classification based on vehicle vibrations for feature extraction or signal representation. The standard assessment of road surface is performed via visual inspection based on the guidelines provided by ASTM International in the standard known as the ASTM D6433 - 20: Standard Practice for Roads and Parking Lots Pavement Condition Index Surveys (ASTM, 2023). This standard proposes guidelines to evaluate the condition of pavement distress, such as potholes, rutting, bumps, bleeding, corrugations, and longitudinal and transverse cracking. The severity of the road surface is quantified by calculating the PCI. The PCI gives an overview of a road's pavement condition. It is scored between 0 and 100, where 100 means a newly paved road, while a deteriorated road will score 0 (Piryonesi and El-Diraby, 2021). Table 1 shows the PCI classes proposed by the ASTM International to evaluate the pavement condition (Piryonesi and El-Diraby, 2020). However, this method is predisposed to human errors, is time-consuming, and can put the health of road technicians at risk (Martinelli et al., 2022; Shaghil and Khalafallah, 2018). Hence, these drawbacks have influenced the development of alternative procedures in which road surface can be assessed. One of these methods is assessing the road surface by analyzing the acceleration or vibration of vehicles while traveling on the road via the use of accelerometers.

Generally, the literature on road surface anomaly detection and classification based on vehicle vibrations has employed algorithms based on thresholds techniques, feature extraction, and deep learning as presented and extensively discussed by Sattar et al. (2018), Kim et al.



**Fig. 1.** The overall methodology applied in this study. The first step consists of determining the value of voices per octave and the beta parameter of Generalized Morse Wavelets that produce the minimum ARMSE from the vehicle's vertical acceleration signals. The second step consists of fine-tuning pre-trained convolutional neural networks through the scalograms generated with the value of voices per octave and beta that produced the minimum ARMSE for pavement transverse cracking and uncracked pavement classification.

**Table 1**  
PCI classes proposed by ASTM International for evaluating the pavement condition.

| PCI range | Class        |
|-----------|--------------|
| 85–100    | Good         |
| 70–85     | Satisfactory |
| 55–70     | Fair         |
| 40–55     | Poor         |
| 25–40     | Very poor    |
| 10–25     | Serious      |
| 0–10      | Failed       |

(2022), and Martínez-Ríos et al. (2022) to detect and classify potholes, speed bumps, cracks or straight roads. In the case of feature extraction approaches, features in the time or frequency domain of vibration data are used to train classical machine learning classifiers. These features can include the computation of statistical moments, maximum and minimum values, or range values computed by analyzing a window of the vibration signal of the vehicle (Sattar et al., 2021).

Time-frequency representations have been used individually or in combination with time and frequency domain features for road surface anomaly classification based on vehicle vibrations for training classical machine learning classifiers. One of the main characteristics of time-frequency representations is that they provide insight into how the frequency content of a signal changes with time, which has motivated their use to solve signal classification tasks. When working with deep learning algorithms such as CNNs, the 2D mapping of the vibration data or 1D signal produced by the time-frequency method is used as input into the algorithm without requiring a feature extraction stage as classical machine learning methods (Martínez-Ríos et al., 2022; Baldini et al., 2020).

One instance of these approaches is the study presented by Basavaraju et al. (2020) that utilized accelerometer data sampled from smartphones and considered its three axes to assess road surface anomalies. Time and frequency domain features were tested in combination with time-frequency domain features computed through the wavelet transform utilizing the Morlet and Daubechies wavelets. The machine learning algorithms trained with the computed features were a support vector machine and a decision tree. Nonetheless, a multilayer perceptron trained with the raw acceleration data provided a higher performance with an accuracy of 92.12%.

Similarly, Wu et al. (2020) computed the discrete wavelet transform for pothole detection by sensing the vehicle's vibration through the

built-in smartphone sensors. Different wavelet types, such as Daubechies, Haar, Symlet, and Reverse Biorthogonal 3.1 were tested. The above led to the calculation of 144 features in the wavelet domain for each tested wavelet. In addition, time and frequency domain features were also considered, comprising 126 features by considering both domains. According to the authors, a random forest with a reported recall of 75% and a precision of 88.5% provided the best performance by employing only features in the time and frequency domain, since the proposed wavelet domain features produced a lower performance.

Likewise, Ferjani and Ali Alsaif (2022) proposed using the features of the time, frequency, and time-frequency domains to classify anomalies on the road's surface, such as potholes and speed bumps, by processing the three axes of the accelerometer. The algorithms tested were a support vector machine, a multilayer perceptron, and a decision tree. The mother wavelet was a second-order Daubechies wavelet to compute the time-frequency domain features. The authors reported an accuracy of 94% by training a decision tree. Besides, Li et al. (2019) employed a third-order Daubechies wavelet to compute the CWT on smartphone accelerometer data. The generated features from the CWT were used for the detection and size estimation of potholes. The detection was performed with a clustering technique known Density-Based Spatial Clustering of Applications with Noise. This process led to obtaining an accuracy of 94.44%.

Yang et al. (2021) proposed to detect transverse cracking in asphalt pavement sections by processing vehicle's vertical acceleration data and computing time, frequency, and wavelet domain indices, which were used to train eight support vector machine models. According to the authors, 2292 pavement sections were studied, and the best model reported an accuracy and F1-score of 97.25% and 85.2%, respectively. The CWT was used to compute the wavelet domain indices; however, the authors did not report the mother wavelet employed to compute the CWT, which limits the reproducibility of the results.

Baldini et al. (2020) presented an extensive analysis of the influence of the parameters of the STFT in road surface anomaly classification while analyzing inertial measurement unit data for the classification of potholes, cracks, and speed bumps. This study tested different window lengths, types, and overlaps while computing the STFT to produce a spectrogram that can be used as input to a CNN to identify road surface anomalies. The authors reported an accuracy of 97.2% by following this procedure. However, one of the drawbacks of CNNs is that the tuning process of their parameters, such as the weight initialization, number of

**Table 2**

Summary of the works that have employed time–frequency domain representations for road surface anomaly classification performed through vibration-based techniques. The accuracy results reported were computed considering the entire sample without distinguishing between individual road surface anomalies.

| Author                        | Classified road anomalies                                     | Features   | Machine learning algorithm   | Performance      |
|-------------------------------|---|--|------------------------------|------------------|
| Basavaraju et al. (2020)      | Cracks<br>Potholes<br>Smooth roads                            | Time and Frequency domain features.<br><br>Wavelet domain features computed through a Morlet, and Daubechies order 6 and 10 wavelets.                          | Multilayer perceptron        | Accuracy: 92.12% |
| Li et al. (2019)              | Bumps<br>Potholes   | Continuous Wavelet Transform computed with a 3rd order Daubechies wavelet.   | Density-Based Clustering     | Accuracy: 94.44% |
| Baldini et al. (2020)         | Potholes<br>Cracks<br>Patches<br>Speed bump<br>Rumble strips  | Short-time Fourier transform spectrogram   | Convolutional neural network | Accuracy: 97.20% |
| Wu et al. (2020)              | Potholes  | Time and Frequency domain features.<br><br>Discrete wavelet transform features computed with Daubechies, Symlet, Haar, and Reverse Biorthonormal 3.1 wavelets. | Random forest                | Accuracy: 95.7%  |
| Yang et al. (2021)            | Crack asphalt<br>Pavement                                     | Continuous Wavelet Transform based features  | Support vector machine       | Accuracy: 97.25% |
| Ferjani and Ali Alsaif (2022) | Potholes<br>Metal bumps<br>Asphalt bumps<br>Worn out roads    | Time and Frequency domain features.<br><br>Discrete wavelet transform features computed with a 2nd order Daubechies wavelet.                                   | Decision tree                | Accuracy: 94.00% |
| Martinelli et al. (2022)      | No pavement distress<br>Potholes/Manholes<br>Fatigue cracking | Time-frequency domain features obtained through the Short-time Fourier transform.  | Support vector machine       | Accuracy: 91.9%  |

layers, and number of convolutions in each layer, need to be set empirically; added to this, there is the computational power, and training time required to set the parameters of the network adequately. Furthermore, the STFT has resolution limitations in the time–frequency plane that could mitigate the adequate representation of a signal (Mallat, 2008; Fugal, 2009; Durak and Arikán, 2003).

Ultimately, the study of Martinelli et al. (2022) employed the STFT to extract features from the time–frequency plane to evaluate pavement distresses such as potholes, manholes, rutting, and cracking by using accelerometer sensors placed on the vehicle. The proposed features included the coefficient of variation and entropy over the time segments of the STFT. The tested machine learning algorithms included a support vector machine, a decision tree, and a k-nearest neighbor. A cubic support vector machine achieved the best performance with positive predictive values of 97%, 84%, and 97% for potholes and manholes, fatigue cracking, and no-distress, respectively. Contrary to Baldini's study, Martinelli's work did not perform an extensive analysis of the parameterization of the STFT that was taken into account to compute the proposed features.

Table 2 summarizes the methodologies and results reported in each reviewed study. In addition, the road surface anomalies classified through the vehicle's vibration data are shown in the same Table. The works that only considered one road surface anomaly make a comparison by considering the vehicle's vibration data associated with the absence of the road surface anomaly. Furthermore, the road surface anomalies considered in the related research were labeled using manual distress surveys or through visual inspection. The accuracy results presented in Table 2 for each reviewed study were computed considering the entire sample with road surface anomalies. The performance for recognizing each road surface anomaly analyzed through the vehicle's vibration data in the respective studies can be consulted in the references in the same Table for further detail.

This literature review shows that an exhaustive discussion has yet to be carried out when computing features for road surface anomaly detection and classification through time–frequency analysis techniques applied to the vehicles' vibration data. A common approach is to perform trial and error tests, such as the case of Baldini's study, which presents an analysis of how the STFT parameters influence the performance of a CNN or Wu's work that tests different wavelet types to perform the discrete wavelet transform and compute wavelet domain features. However, an analysis performed a priori that expands on the considerations that led to testing a particular wavelet has not been carried out. The above is related to discussing the underlying properties of the mother wavelet that were considered for performing the signal decomposition of the vibration data of the vehicle while passing over the road surface anomaly. In some cases, the mother wavelet was not reported as in the case of Yang's study, or the parameterization of the time–frequency method was not extensively discussed as in Martinelli's work. A common choice is the Daubechies family; nevertheless, different orders of Daubechies wavelets have been employed without performing a justification related to its selection.

Finally, training machine learning techniques require a sizeable dataset, and tuning the hyperparameters can be time-consuming (Alzubaidi et al., 2021). The above was also mentioned by Basavaraju et al. (2020). Thus, there is an opportunity to test transfer learning methods to avoid setting models from zero and requiring a large sample size to train machine learning algorithms (Xiao et al., 2019). For instance, Baldini's study needed 9600 vibration signal segments to train a CNN. There is also an opportunity to highlight the advantages and disadvantages of using transfer learning for detecting and classifying road surface anomalies based on vehicle vibrations as examined by Martínez-Ríos et al. (2022).

### 3. Materials and methods

This section presents the dataset used to carry out the present work and an overview of the theoretical background of the algorithms and methods used to develop this study. Furthermore, the general methodology followed in selecting the parameters of the GMWs to compute the scalogram of the vertical acceleration signals is explained. Finally, the process of applying the transfer learning framework by fine-tuning pre-trained CNNs for pavement transverse cracking detection based on vehicle's vertical acceleration signals is presented.

#### 3.1. Dataset

To compare the proposed methodology presented in this study with the related research, it was decided to employ a previously sampled and labeled public dataset for road surface anomaly detection based on vehicle vibrations. Hence, the vehicle's vertical acceleration data collection and road surface anomaly labeling procedure were not part of the present study. The vehicle vertical acceleration dataset used in this study was collected in Shanghai, China, on three roads in the same city. This dataset can be consulted from the Mendeley Data open repository of Zhou (2019). The primary purpose of this dataset is to detect transverse cracking and uncracked sections in asphalt pavement based on the vehicle's vertical acceleration. According to ASTM International, transverse cracks cross the pavement at a roughly right angle to its center line or lay down direction. Transverse cracks can be produced by the shrinkage of the asphalt concrete caused by low temperatures, hardening of the asphalt, or reflective cracks produced under the surface. This type of crack typically has nothing to do with loading (ASTM, 2023). Table 3 summarizes the main characteristics of the dataset used in this study.

The vehicle's vibration data was collected using a piezoelectric analog accelerometer model CT1100L, a data acquisition device model MCC USB-231, a notebook computer, and a current adapter. The sampling frequency of the accelerometer was set as 1280 Hz (Zhou, 2019). The accelerometer was placed underneath the front passenger seat in the tire's suspension knuckle. The accelerometer sampled the vertical acceleration sensed in the tire suspension knuckle. The criteria for setting the accelerometer in this position was to avoid the vertical acceleration signal being affected by the vibration isolation system of the vehicle. Moreover, in this position, the sensor only recognized the vibration of the front wheel, and the influence of the rear wheel was reduced (Yang et al., 2021). Furthermore, according to the review of road surface anomaly detection and classification systems based on vibration-based techniques presented by Martínez-Ríos et al. (2022), the vehicle's vertical acceleration has been frequently used to detect and classify road surface anomalies by employing machine learning algorithms and provides a better performance compare to employing all the three axes of the accelerometer. The use of the vehicle's vertical acceleration data was also supported by the work of Sattar et al. (2018). Additionally, processing only one axis of the accelerometer is less computationally expensive than analyzing the accelerometer sensor's vertical, transversal, and horizontal axes simultaneously. Fig. 2 shows a sketch of the placement of the accelerometer sensor in the vehicle based on the description provided in the work of Yang and Zhou (2021).

The analyzed asphalt pavement sections had a length of 4 m. The dataset authors identified and labeled the transverse cracks in the asphalt pavement by employing a manual distress survey. The cracks had a width of 2 to 13 mm, and the spacing between cracks ranged from 7 to 35 m. To avoid the influence of adjacent cracks on the vibration signals, the sections used to collect the data were previously assessed through the same manual distress survey. While collecting the dataset, the vehicle's speed was restricted to three velocities 30, 40, and 50 km/h (Yang and Zhou, 2021). These velocities were considered since the testing conditions were in an urban area. This procedure collected 327 vertical acceleration signal segments associated with transverse

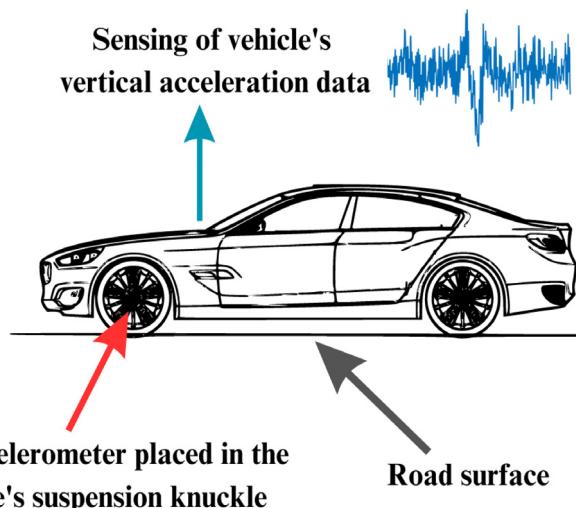


Fig. 2. Illustration of the accelerometer sensor placement in the vehicle for sensing the vehicle's vertical acceleration while traveling on the road surface.

Table 3

Summary of the main characteristics of the dataset.

| Dataset characteristics  |   |
|--|---|
| Sample size (N)  | 327   |
| Accelerometer model  | Piezoelectric accelerometer CT1100L                       |
| Sensor placement   | Suspension knuckle of tire under the front passenger seat |
| Data acquisition card  | MCC USB-321   |
| Vehicle's vertical acceleration signals units                          | m/s <sup>2</sup>  |
| Number of signal segments associated with pavement transverse cracking | 109 (33%)   |
| Number of signal segments associated with uncracked pavement           | 218 (67%)   |
| Sampling frequency   | 1280 Hz   |
| Range of vehicle velocities  | 30 to 50 km/h   |
| Range of the width of pavement cracks                                  | 2 to 13 mm  |
| Range of space between pavement cracks                                 | 7 to 35 m   |

cracks and uncrack pavement sections. The collected signal segments had different lengths, such as 126 signals with a length of 369 samples, 143 signals with a length of 461 samples, and 58 signals with 615 samples.

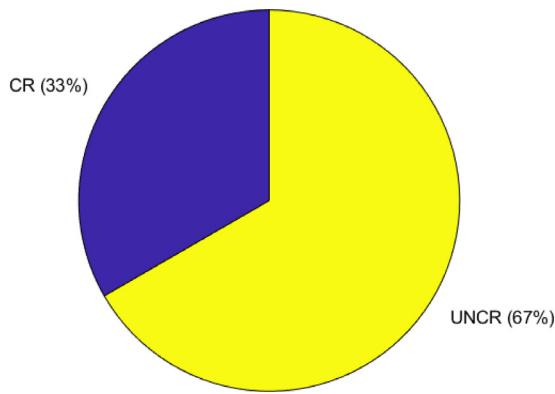
Fig. 3 shows the distributions of the two classes available in the dataset. It can be appreciated that the class distribution of the vehicle's vertical acceleration signals of the dataset is unbalanced since there is a higher percentage of signals associated with uncracked sections (67 %) compared to the signals associated with transverse cracking sections (33 %). Further detail relating to the data collection process presented in this section can be consulted in the works of Yang and Zhou (2021), Zhou (2019), and Yang et al. (2021).

#### 3.2. Continuous wavelet transform and generalized morse wavelets

This section summarizes the background related to the CWT and the wavelet family used in this study, known as the GMWs. The above is done to provide an overview of the techniques employed and examined in this study.

##### 3.2.1. Continuous wavelet transform

Time-frequency analysis refers to techniques, algorithms, and methods that help to quantify the changes in the frequency content of a



**Fig. 3.** The class distribution of the classes available in the dataset. CR refers to the percentage of the vehicle's vertical acceleration signals associated with transverse cracking asphalt pavement sections, while UNCR refers to the percentage of the vehicle's vertical acceleration signals associated with uncracked pavement sections.

signal over time (Hlawatsch and Auger, 2013). The above is accomplished by creating a 2D mapping of the original time series that provides time and frequency localization. A classical approach to construct this 2D mapping of a signal's time and frequency content is the STFT. However, since the Fourier transform is applied to a fixed window length, the STFT has a fixed resolution in both time, and frequency (Durak and Arıkan, 2003).

The wavelet transform counters the fixed resolution of the STFT by allowing multiple window lengths to change across frequencies (Brunton and Kutz, 2019). Contrary to the STFT, the wavelet transform employs the mother wavelet function to decompose the signal in time and frequency. This process is shown in Eq. (1) (Mallat, 2008).

$$Wf(u, s) = \langle f(t), \psi_{u,s}(t) \rangle = \int_{-\infty}^{\infty} f(t) \frac{1}{\sqrt{s}} \psi^*(\frac{t-u}{s}) dt \quad (1)$$

where  $Wf$  is the wavelet transform of a square-integrable function  $f(t) \in L^2(\mathbb{R})$ , and  $\psi_{u,s}(t) \in L^2(\mathbb{R})$  represents the mother wavelet used to perform the wavelet transform for a particular scale  $s$ , and translation  $u$ . The scale parameter allows controlling the frequency localization of the wavelet transform, while the translation parameter controls the time domain localization. In addition, a wavelet must have a zero mean as expressed in Eq. (2) (Daubechies, 1992).

$$\int_{-\infty}^{\infty} \psi(t) dt = 0 \quad (2)$$

Furthermore, this function must satisfy the admissibility condition ( $C_\psi$ ), presented in Eq. (3).

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\Psi(\omega)|^2}{|\omega|} d\omega < \infty \quad (3)$$

where  $\Psi(\omega)$  is the Fourier transform of the mother wavelet  $\psi(t)$ . When this condition is met, it is possible to obtain a stable inverse transform.

To perform the wavelet transform across frequency and time, a dictionary ( $D$ ) of wavelets is obtained by translating and scaling the mother wavelet, as shown in Eq. (4).

$$D = \left\{ \psi_{u,s}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-u}{s}\right) \right\}_{u \in \mathbb{R}, s \in \mathbb{R}^+} \quad (4)$$

The coefficients obtained for each scale and translation received the name of wavelet coefficients, which represent the degree of similarity or correlation between  $f(t)$ , and the wavelet  $\psi_{u,s}(t)$  for each translation  $u$  and scale  $s$ . Since computing the wavelet transform can be understood as computing correlations between  $f(t)$  and  $\psi_{u,s}(t)$  across frequency scales, the mother wavelet should have some degree of similarity with  $f(t)$ ; thus, its selection must be made with care.

As presented before, the admissibility condition assures that the function  $f(t)$  can be reconstructed from the obtained wavelet coefficients. The inverse transform is shown in Eq. (5) (Mallat, 2008).

$$f_r(t) = \frac{1}{C_\psi} \int_0^{\infty} \int_{-\infty}^{\infty} Wf(u, s) \frac{1}{\sqrt{s}} \psi\left(\frac{t-u}{s}\right) du \frac{ds}{s^2} \quad (5)$$

where  $f_r(t)$  is the reconstructed version of  $f(t)$  from the wavelet coefficients, and  $C_\psi$  is the admissibility condition. The variable  $s$  refers to the scale parameter, and  $u$  is the translation parameter.

The term CWT refers that the scale and translation parameters are varied continuously rather than referring to a continuous function of time. On the other hand, the so-called discrete wavelet transform varies the translation and scale parameters with a process called dyadic sampling. The dyadic sampling procedure involves selecting the scales and translation parameters with powers of two. This process is presented in Eq. (6), where  $j$  is the scale parameter, and  $k$  is the translation parameter, both parameters assumed integer values (Mallat, 2008; Fugal, 2009).

$$\psi_{j,k}(t) = \frac{1}{\sqrt{2^j}} \psi\left(\frac{t-k2^j}{2^j}\right) \quad (6)$$

### 3.2.2. Generalized morse wavelets

This section presents an overview of the theoretical background of GMWs originally presented by Lilly and Olhede (2012). Other relevant papers that can be consulted are the works of Lilly and Olhede (2009, 2010). Furthermore, a summary of the applications of GMWs in different fields and areas of opportunity for selecting its parameters are presented in the study of Martinez-Ríos et al. (2023).

GMWs are time–frequency localized filters with vanishing support for negative frequencies. The vanishing support at negative frequencies makes the GMWs analytical wavelets. Analytical wavelets are preferred to analyze the oscillatory behavior of frequency transients, while non-analytical wavelets are preferred to enhance sharp signal transitions in the time–frequency plane (Mallat, 2008). GMWs avoid issues associated with non-analytical wavelets, such as the development of artifacts and interferences in the time–frequency plane (Lilly and Olhede, 2009). The frequency domain representations of GMWs can be seen in Eq. (7) (Lilly and Olhede, 2012).

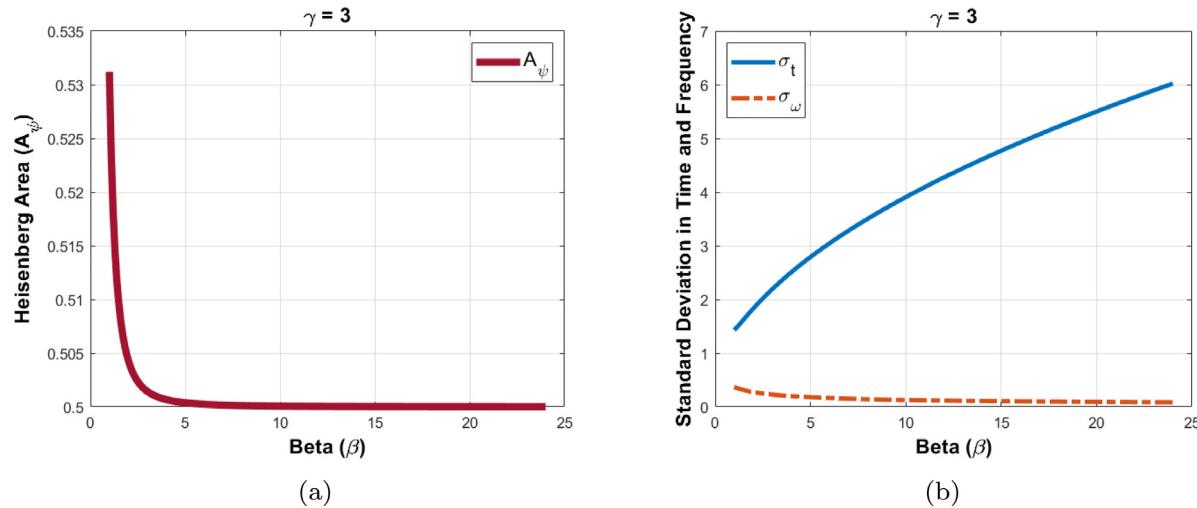
$$\Psi_{\beta,\gamma} = \int_{-\infty}^{\infty} \psi_{\beta,\gamma}(t) e^{-i\omega t} dt = U(\omega) a_{\beta,\gamma} \omega^\beta e^{-\omega^\gamma} \quad (7)$$

The term  $\Psi_{\beta,\gamma}$  is the frequency domain representation of the GMWs;  $\psi_{\beta,\gamma}(t)$  represent the time domain representation of the GMWs;  $a_{\beta,\gamma}$  is a normalization constant;  $U(\omega)$  is a unit step function; which represents the vanishing support at negative frequencies. The  $\beta$  parameter controls the time domain decay of the wavelet; while  $\gamma$  controls the wavelet frequency domain decay. For a GMW to be valid, gamma and beta must be greater than zero ( $\gamma > 0, \beta > 0$ ). Moreover, the normalization constant is equivalent to  $a_{\beta,\gamma} \equiv 2(e\gamma/\beta)^{\beta/\gamma}$  with  $e$  representing Euler's number (Lilly and Olhede, 2010).

GMWs receive their name due to two essential properties. First, when  $\gamma = 1$  GMWs are equivalent to a solution to the Schrödinger equation analyzed by Morse (1929). Second, for certain gamma values, this wavelet is equivalent to other wavelet families. For instance, for  $\gamma = 2$ , GMWs are equivalent to the Derivative of Gaussian family of wavelets. Moreover, for  $\gamma = 1$ , GMWs are equivalent to the Cauchy family. Finally, for  $\gamma = 3$ , the Airy wavelets can be obtained. The Airy wavelets are the most closely gaussian and symmetric wavelets (Olhede and Walden, 2002; Lilly and Olhede, 2010, 2009). These characteristics make the GMWs a recommended starting point for general-purpose use.

As can be appreciated, the selection of a wavelet is reduced to determining the values of gamma and beta. Lilly and Olhede (2012) proposed selecting the values of these parameters by analyzing the Heisenberg area, a metric that indicates the degree of energy concentration or time–frequency localization of a wavelet. The Heisenberg area can be computed based on the expression shown in Eq. (8).

$$A_\psi \equiv \sigma_t \sigma_\omega \quad (8)$$



**Fig. 4.** Behavior of the Heisenberg area and time and frequency domain standard deviations of the Generalized Morse Wavelets for increasing values of  $\beta$ , and  $\gamma = 3$ . (a) Shows the behavior of the Heisenberg area of Generalized Morse Wavelets for increasing values of  $\beta$  and  $\gamma = 3$ . (b) Shows the behavior of the standard deviation of the Generalized Morse Wavelets in the time and frequency domains for increasing values of  $\beta$  and  $\gamma = 3$ . The blue line is the time domain standard deviation, and the dash red line is the frequency domain standard deviation.

The term  $\sigma_t$  is the standard deviation in the time domain of the GMW, while the term  $\sigma_{\omega}$  is the standard deviation in the frequency domain of the GMW. Heisenberg's uncertainty principle states that a function cannot be arbitrarily compact in time and frequency. The aforementioned suggests that it is difficult to locate finite oscillation transients simultaneously in time and frequency (Moca et al., 2021). This leads to the fact that the Heisenberg area has a lower bound of one-half as expressed in Eq. (9) (Mallat, 2008).

$$\sigma_t \sigma_{\omega} \geq \frac{1}{2} \quad (9)$$

Gaussian functions can approximate this lower-bound of the Heisenberg area (Wojczyński, 2011). In the case of GMWs and considering that this wavelet family is defined as Gaussian functions, the Heisenberg area lower-bound is approximated when  $\gamma = 3$ . Fig. 4(a) shows the behavior of the Heisenberg area of GMWs for  $\gamma = 3$ , and increasing values of  $\beta$ . Notice that by setting  $\gamma = 3$ , the Heisenberg area of GMWs approximates the value of one-half, and for values of  $\beta$  greater than five, the Heisenberg area is practically equal to 0.5 (Martínez-Ríos et al., 2023).

Considering that the Heisenberg area is defined as the product between the standard deviations of a wavelet in the time and frequency domains, it is possible to study its behavior independently for the case GMWs. Fig. 4(b) shows the time and frequency domain standard deviation behaviors. As can be appreciated, there is a trade between the degree of localization in both time and frequency of a wavelet by increasing or decreasing the values of  $\beta$ . Furthermore, it is possible to appreciate that the standard deviation in the time domain of GMWs is more sensitive to the changes of  $\beta$  compared to the frequency domain standard deviation when  $\gamma = 3$ . Thus, its selection must be chosen carefully since it influences the wavelet's time and frequency localization degree.

Additionally, the GMWs can be expressed in terms of the dimensionless duration of the wavelet. This duration is defined in terms of  $\beta$  and  $\gamma$  as shown in Eq. (10).  $P_{\beta,\gamma}$  controls the number of oscillations of the wavelet in the time domain within its central window.

$$P_{\beta,\gamma} = \sqrt{\beta\gamma} \quad (10)$$

Considering the above, GMWs can be represented in terms of the duration  $P_{\beta,\gamma}$  as shown in Eq. (11). When  $P_{\beta,\gamma}$  is squared, the term receives the name time-bandwidth product ( $P^2 = \beta\gamma$ ).

$$\Psi_{P,\gamma} = \int_{-\infty}^{\infty} \psi_{P,\gamma}(t) e^{-i\omega t} dt = U(\omega) a_{P,\gamma} \omega^{\frac{P^2}{\gamma}} e^{-\omega^2} \quad (11)$$

In this work, gamma ( $\gamma$ ) was defined based on the Heisenberg area lower-bound, while beta ( $\beta$ ) and the number of VPO (i.e., the number of wavelets within a frequency octave) were selected based on the ARMSE. Martínez-Ríos et al. (2023) showed that the typical parameterization of GMWs employed in the literature is setting  $\gamma = 3$ , and  $\beta = 20$ , without performing a justification related to its selection and that in some studies, the parameterization was not reported. Hence, the purpose of employing the ARMSE as a reference metric is to enable a basis for selecting  $\beta$  and the number of VPO. The calculation process of the ARMSE is described in the following sections. The scalogram of the vehicle's vertical acceleration signals generated by the selected GMWs and CWT parameterization will be used as input to fine-tune the pre-trained CNNs.

### 3.3. Transfer learning

Despite machine learning having solved a prominent vast of problems and real-world applications, it has specific weaknesses that complicate its execution. Machine learning expects that numerous training samples of the same distribution will be available for training, validation, and test datasets. Nonetheless, gathering sufficient data could be expensive, tedious, and unrealistic (Zhuang et al., 2021). Transfer Learning tries to overcome these obstacles by transferring knowledge between different domains instead of only solving an isolated task as the classic machine learning paradigm. A graphical representation of the transfer learning methodology can be seen in Fig. 5. First, a model is trained for a source domain, known as the source task. Consequently, the weights or parameters of the pre-trained model are fine-tuned to adjust the pre-trained model based on the distribution of the target domain or target task (Himeur et al., 2023).

Transferring knowledge involves using two different definitions, the concept of domain and the concept of task. A domain  $D$  is conformed by a feature space  $\mathcal{X}$  and marginal probability distribution  $P(X)$  over the feature space. Besides,  $X$  is defined as  $X = \{x_1 \dots x_2\} \in \mathcal{X}$ . Considering the domain,  $D = \{\mathcal{X}, P(X)\}$ , a task  $T$  consists of a label space  $\mathcal{Y}$  and a conditional probability distribution  $P(Y|X)$  that is commonly learned from the label training data of pairs  $x_i \in \mathcal{X}$  and  $y_i \in \mathcal{Y}$ . Thus, a task can be defined as  $T = \{\mathcal{Y}, P(Y|X)\}$ .

Moreover, considering the source domain  $D_S$ , the source task  $T_S$ , the target domain  $D_T$ , and the target task  $T_T$ , the main purpose of transfer learning is to learn the target conditional probability distribution  $P(Y_T|X_T)$  in  $D_T$  with the knowledge acquired from the source domain and source task, where  $D_S \neq D_T$  or  $T_S \neq T_T$ . Most of the time,

# Transfer Learning

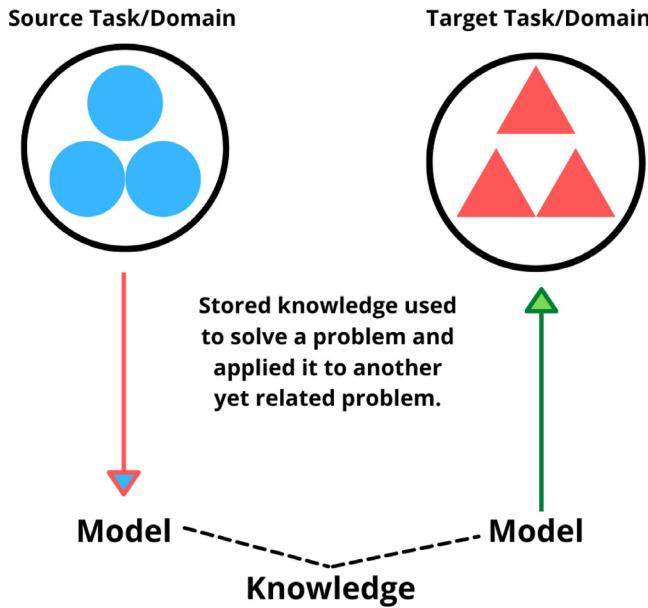


Fig. 5. Graphical representation of the transfer learning methodology in machine learning.

it is assumed that the number of labeled target samples is exponentially fewer than that of labeled source samples. These definitions allow for the description of four transfer learning scenarios (Pan and Yang, 2010).

- The feature space of the source and target domains are different. This implies that  $\mathcal{X}_S \neq \mathcal{X}_T$ .
- The marginal probability distributions of the source and target domain are distinct. This means that  $P(X_S) \neq P(X_T)$ .
- The label spaces are different between the source and target tasks. This denotes that  $\mathcal{Y}_S \neq \mathcal{Y}_T$ .
- The conditional probability distributions of the source and target tasks are distinct. This indicates that  $P(Y_S|X_S) \neq P(Y_T|X_T)$ .

The target domain of this study is the distribution of the vehicle's vertical acceleration signals associated with transverse cracking and uncracked pavement sections. The source domain is the distribution of the ImageNet dataset since the pre-trained CNNs were initially trained with that dataset. These pre-trained CNNs will be fine-tuned with the vehicle's vertical acceleration signals represented through the wavelet transform scalogram generated through GMWs (Morid et al., 2021).

## 3.4. Convolutional neural networks

CNNs are feedforward neural networks that can extract features from data by employing convolution operations. Compared to traditional machine learning algorithms, CNNs can automatically extract features from data relevant to solve the classification task (Li et al., 2022b). This type of neural network architecture has gained popularity for image recognition tasks. The overall structure of this type of neural network is presented in Fig. 6. The architecture of a CNN comprises three main elements convolution layer, a pooling layer, and a fully connected layer.

Since an image is in color, it has three channels, and each of these channels can be represented as a three-dimensional array or feature map. The convolution layer consists of a large number of convolution filters. Each filter determines if an image contains a specific local

feature by employing the convolution operation between the input image feature maps and the convolution filters. These filters often have a low dimensionality, such as  $3 \times 3$ ,  $5 \times 5$ , and  $7 \times 7$  grids (Lopez Pinaya et al., 2020). The network's training determines the weights of these convolution filters through backpropagation and gradient descent. The input image's areas that resemble the convolution filter are highlighted in the convolved image. This convolution process is described in Eq. (12).

$$y_j^l = g\left(\sum_i x_i^{l-1} * w_{ij}^l + b_j^l\right) \quad (12)$$

where  $l$  represents the network's layer number;  $j$  is the index of the output feature map;  $i$  is the index of the input feature map;  $*$  represents the convolution between the  $j$ th kernel  $w_{ij}^l$  with the  $x_i^{l-1}$  input feature map at the  $l$ th layer; finally,  $g$  is a nonlinear activation function (Liang et al., 2022). The ReLU activation function is commonly used in the convolved image; however, other widespread use activation functions are the sigmoid and hyperbolic tangent (Gu et al., 2018). Eq. (13) shows the ReLu activation function, where  $z$  is the result of the convolved input image with the convolution filter or kernel.

$$g(z) = \max(0, z) \quad (13)$$

On the other hand, the pooling layer is used to reduce the dimensionality of the convolved image. This process can also be interpreted as a sub-sampling operation. A typical strategy to perform the pooling stage is by applying the max pooling operation, which consists of taking the convolved image's maximum value in a small window size typically of  $2 \times 2$  (James et al., 2013). Finally, after applying the pooling operation, the results of the last pooling stage are flattened and fed to a fully connected layer, as shown in Fig. 6.

In this work, pre-trained CNNs were fine-tuned following the transfer learning framework for pavement transverse cracking detection and using as input the scalogram of the vehicle's vertical acceleration data generated from the CWT and GMWs.

## 3.5. Preprocessing steps

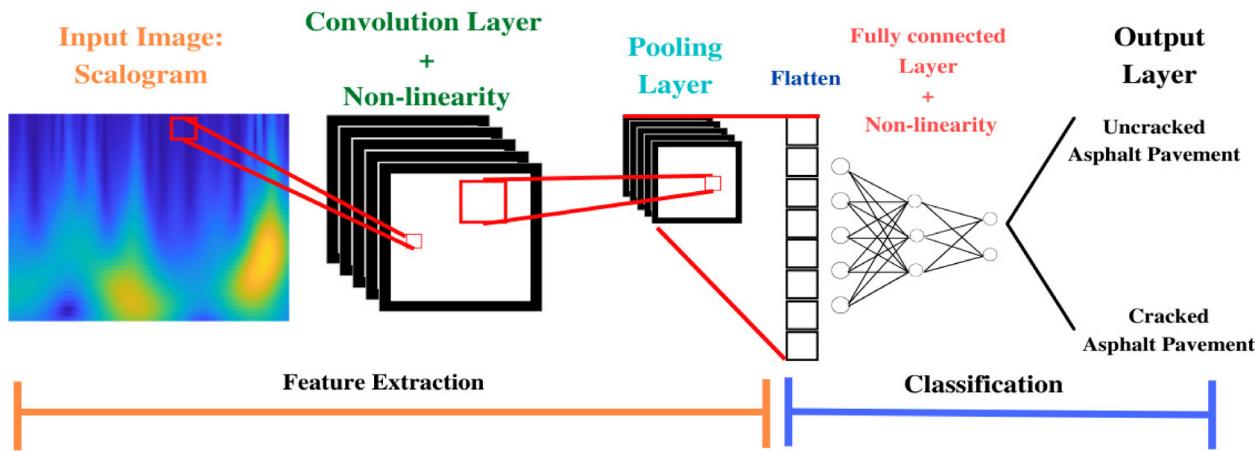
Before computing the CWT of the vehicle's vertical acceleration signals available in the dataset to fine-tune the pre-trained CNNs, a denoising stage was performed on each signal recording. As the first step, the mean of each signal segment was removed following the expression shown in Eq. (14).

$$X_{ij} = Y_{ij} - \frac{1}{n_i} \sum_{j=1}^{n_i} Y_{ij} \quad (14)$$

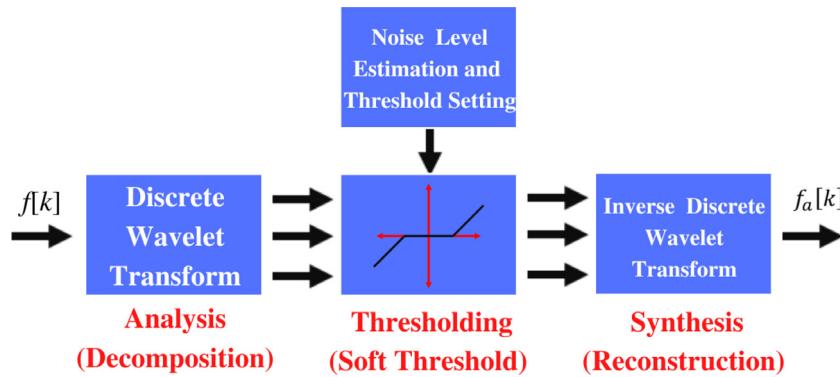
where  $X_{ij}$  is the  $j$ th data point of the  $i$ th signal of the dataset of size  $N$  after removing the mean;  $Y_{ij}$  is the  $j$ th data point of the original  $i$ th raw signal without removing the mean. Finally,  $n_i$  is the size of the  $i$ th signal.

On the other hand, the high-frequency noise presented in the signals was removed with wavelet de-noising following the methodology and parameterization presented by Yang et al. (2021). The general strategy for performing a wavelet de-noising approach is presented in Fig. 7. The first step consists in computing the discrete wavelet transform at a predefined number of decomposition levels to generate the details and approximation coefficients. Consequently, a noise estimation level and thresholding rule (e.g., hard or soft thresholding) are determined to threshold the obtained wavelet coefficients. Finally, the thresholded wavelet coefficients are used to compute the inverse discrete wavelet transform and obtain the de-noised signal (Pan et al., 1999).

The noise estimation technique was the universal thresholding procedure selected along with the soft thresholding rule (Donoho, 1995). The wavelet used to conduct the discrete wavelet transform was a Symlet of order 3 (i.e., Sym3). The decomposition level was chosen as 4. Fig. 8 shows an example of the de-noised vehicle's vertical acceleration signal performed with wavelet-de-noising and the original raw vehicle's vertical acceleration signal.



**Fig. 6.** The general structure of a convolutional neural network. In this illustration, the input image is the scalogram of the Continuous Wavelet Transform from the vehicle's vertical acceleration signal. At the same time, the output is the transverse cracking and uncracked asphalt pavement section labels associated with the vehicle's vertical acceleration data.



**Fig. 7.** Graphical representation of a wavelet de-noising approach. The first step consists in computing the discrete wavelet transform on the raw signal at a pre-defined decomposition level. Next, the obtained wavelet coefficients are thresholded based on the noise estimation level and thresholding rule. Finally, the signal is reconstructed with the inverse discrete wavelet transform.

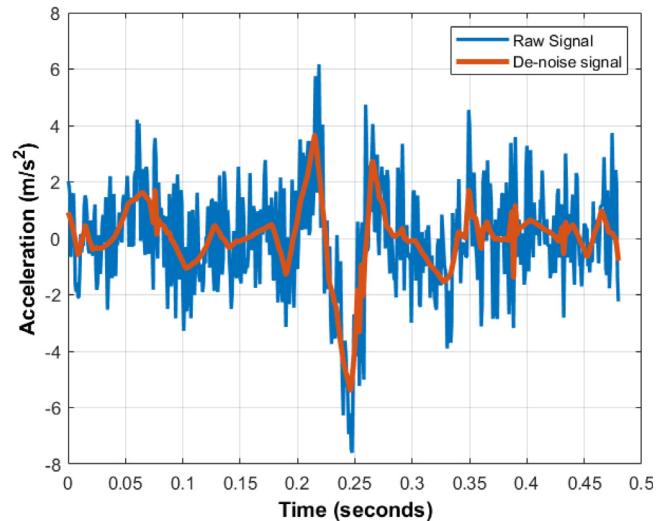
### 3.6. Computation of the average reconstruction mean squared error and selection of beta and voices per octave

To select the beta value of the GMWs and the CWT's number of VPO, this study proposed to use the Reconstruction Mean Squared Error (MSE). The gamma parameter was set as  $\gamma = 3$  since GMWs approximate Heisenberg area lower-bound for this value, as shown in Fig. 4(a). The reconstruction error is measured as a MSE between the original signal and the reconstructed version obtained through the inverse CWT and the wavelet coefficients obtained from CWT for a particular value of  $\beta$  and number of VPO. The above can be appreciated in Eq. (15).

$$MSE_i = \frac{1}{n_i} \sum_{j=1}^{n_i} (X_{ij} - \hat{X}_{ij})^2 \quad (15)$$

where  $n_i$  is the signal's length;  $j$  is a single sample of the signal of length  $n_i$ ;  $X_{ij}$  is the  $j$ th data point of the original  $i$ th vehicle's vertical acceleration signal measured through an accelerometer;  $\hat{X}_{ij}$  is the approximated  $j$ th data point of  $i$ th signal after applying the inverse CWT (as presented in Eq. (5)) and considering the wavelet coefficients obtained for  $\gamma = 3$ , a particular value of  $\beta$ , and a specific number of VPO; the term  $MSE_i$  corresponds to the Reconstruction MSE of the  $i$ th signal of the set of size  $N$ .

In this case, the values of  $\beta$  and the number of VPO are chosen to reduce the Reconstruction MSE. The error is measured since it is expected that an adequate wavelet selection could provide an accurate time-frequency representation of the original signal when transformed



**Fig. 8.** The blue signal represents the raw vehicle's vertical acceleration signal related to transverse cracking in asphalt pavement. The red line shows the de-noised vehicle's vertical acceleration signal generated through wavelet de-noising.

through the CWT; thus, the inverse CWT could provide a correct signal reconstruction based on the obtained wavelet coefficients since they represent the results of the correlation between the signal of interest and a wavelet across scales (see Eq. (1)).

Moreover, the MSE of the reconstruction error was chosen since it is commonly used to measure the performance of autoencoders and compression tasks of 1D signals, images, and videos (Liu et al., 2019; Radünz et al., 2022). Nevertheless, to consider the error for all the sampled signals, the ARMSE of all the signals of the dataset was computed as shown in Eq. (16).

$$ARMSE = \frac{1}{N} \sum_{i=1}^N MSE_i \quad (16)$$

where the term  $N$  in Eq. (16) is the number of signal segments available in the dataset;  $MSE_i$  is the Reconstruction Mean Squared Error of the  $i$ th signal of the set of signals of size  $N$ ; finally, the term  $ARMSE$  refers to the Average Reconstruction MSE computed by considering the individual Reconstruction MSE of each signal  $i$  available in the dataset.

The ARMSE of the signals set was computed using the 327 available vehicle's vertical acceleration signal segments following Eq. (16), and the individual Reconstruction MSE for each signal was calculated following Eq. (15). This was computed for integer values of the time-bandwidth product ( $P^2$ ) with the help of the MATLAB 2022a implementation of GMWs, the CWT, and inverse CWT. The time-bandwidth product was controlled instead of directly selecting  $\beta$  since the MATLAB 2022a implementation only allows the setting of  $\gamma$ , and the time-bandwidth product ( $P^2$ ). On the other hand, gamma was fixed with  $\gamma = 3$ , since, as mentioned previously, when  $\gamma = 3$  GMWs approximate Heisenberg's area lower-bound. The number of VPO was also controlled to determine an adequate scale resolution while computing the CWT for pavement transverse cracking detection based on the vehicle's vertical acceleration signals.

Integer values of the time-bandwidth product were chosen to lower the computational cost of computing the Reconstruction MSEs of the set of signals for each possible value of the time-bandwidth product within the interval of 3 to 120. The set of values of  $P^2$  were integer values ranging from 3 to 60. The above implies that the lowest value that  $\beta$  can assume is  $\beta = 1$ , while the highest value is  $\beta = 20$  by considering the relation shown in Eq. (10). A lower-bound of the time-bandwidth product was set as  $P^2 = 3$ , since based on Eq. (10) and considering  $\gamma = 3$ , the lowest value that  $\beta$  can assume is  $\beta = 1$ , lower values of  $\beta$  have a very long time decay, which is useless in practice (Lilly and Olhede, 2010). An upper bound of 60 for the time-bandwidth product was chosen since it is a common value used in the literature when the GMWs are used, as discussed by Martínez-Ríos et al. (2023).

In addition, the number of VPO was set for integer values ranging from 1 to 10. A lower value of VPO implies a lower scale resolution in the scalogram of the CWT; nonetheless, the computational cost is lower. On the other hand, a higher value of VPO is related to a better scale resolution but at the cost of a higher computational cost (Bentley and McDonnell, 1994).

Algorithm 1 shows a pseudocode of the process followed in computing the ARMSE described previously. For the set of computed ARMSEs for each proposed value of  $P^2$  and VPO, the minimum ARMSE was computed. Finally, based on the computed ARMSEs, a statistical analysis was conducted to determine the difference in the minimum ARMSE obtained between each value of VPO by performing a One-Way Analysis of Variance (ANOVA) and multiple comparisons test. The minimum ARMSEs values were determined with the help of the  $min$  function of MATLAB 2022a.

### 3.7. Transfer learning applied through convolutional neural networks

Transfer learning was applied by fine-tuning the weights of pre-trained CNNs trained for the ImageNet challenge; the selection was taken by considering the depth and number of parameters of the pre-trained CNNs as a reference. This criterion led to select the ResNet18, GoogLeNet, and SqueezeNet. Table 4 shows the depth of the neural networks, the number of parameters, and the size of the input images

### Algorithm 1 ARMSE Computation

```

1: Create a 1D array ( $k$ ) of integer values for the time-bandwidth
   product ( $P^2$ ) ranging from 3 to 60.
2: Create a 1D array ( $z$ ) of integer values for the number of voices per
   octave (VPO) ranging from 1 to 10.
3: Create an array containing the  $N$  vehicle's vertical acceleration
   signals.
4: Create an empty array ( $E$ ) of size  $length(k) \times length(z) \times N$  to save
   the individual Reconstruction MSEs for each  $P^2$ , VPO and signal  $i$ .
5: Create an empty array ( $A$ ) of size  $length(k) \times length(z)$  to save the
   ARMSE for each  $P^2$ , and VPO.
6: for each,  $VPO \in z$  do
7:   for each,  $P^2 \in k$  do
8:     for each,  $i \in \{1, \dots, N\}$  do
9:       Compute the CWT based on the current  $i$  signal, VPO
         value, and  $P^2$  value.
10:      Save the obtained wavelet coefficients of the signal  $i$ .
11:      Compute the inverse CWT of the current  $i$  signal based
         on the current VPO value, the current  $P^2$ , and wavelet coefficients
         of the  $i$  signal.
12:      Compute the Reconstruction MSE according to Eq. (15)
         of the current  $i$  signal.
13:      Save the computed Reconstruction MSE in the array  $E$ .
14:    end for
15:    Compute the ARMSE according to Eq. (16) based on the  $N$ 
         dataset signals for the current VPO and  $P^2$ .
16:    Save the computed ARMSE in the array  $A$ .
17:  end for
18: end for
19: Compute  $min(A)$ .
20: Save the index in which the minimum value of matrix  $A$  occurs.

```

**Table 4**

Characteristics of the selected pre-trained convolutional neural networks used for transfer learning.

| Convolutional neural network | Depth | Number of parameters (millions) | Input image size |
|------------------------------|-------|---------------------------------|------------------|
| GoogLeNet                    | 22    | 7.0                             | 224 × 224        |
| ResNet18                     | 18    | 11.7                            | 224 × 224        |
| SqueezeNet                   | 18    | 1.24                            | 227 × 227        |

that these types of CNNs work with Pannu et al. (2020) and Chakrapani and Sugumaran (2023).

As depicted in Fig. 6 the scalogram generated through the CWT of the vehicle's vertical acceleration signals is used as input to fine-tune the parameters of the pre-trained CNNs. This scalogram is generated by selecting the value of  $\beta$  of the GMWs and the value of VPO that produce the minimum ARMSE, as explained in the previous section.

For this study, a 5-fold cross-validation was selected to test the performance of the algorithms; the results obtained in each fold were registered and averaged to show the performance of CNNs. This implies that the dataset was randomly divided into five parts where four parts, or 80% of the dataset, are used for training, and the last part (i.e., the 20% of the dataset), is used for validation. The process is repeated five times, where each time, the training and validation folds are interchanged between the five partitions of the dataset. Finally, the performance obtained in each of the validation folds is averaged.

Fig. 9 illustrates the 5-fold cross-validation process. Cross-validation was chosen as the validation technique since the sample size is relatively small ( $N = 327$ ). 5-folds were considered to estimate the generalization performance of the proposed CNNs without requiring a large training time and computational power, which increase with a higher number of cross-validation folds (James et al., 2013; Marcot and Hanea, 2021). A schematic representation of applying transfer

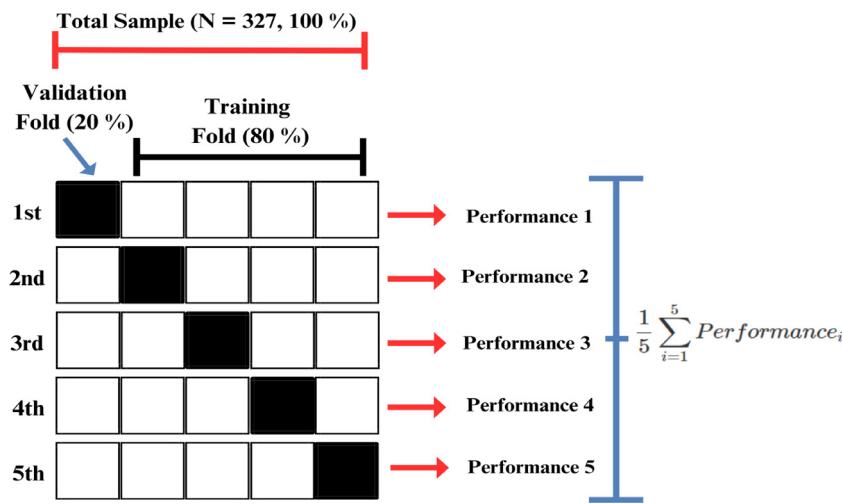
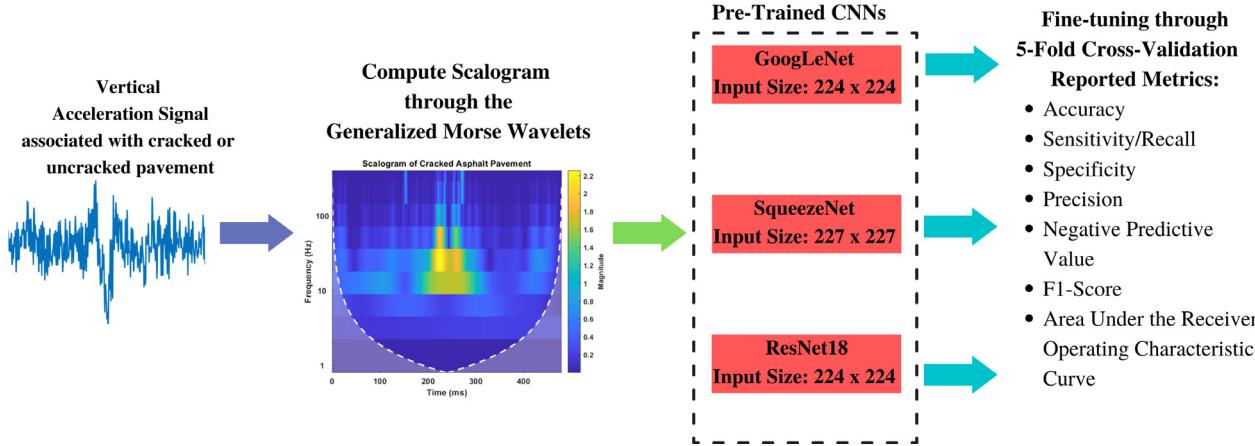


Fig. 9. Workflow for the 5-fold cross-validation process used in this study.

### Transfer Learning applied through pre-trained CNNs



**Fig. 10.** Overall workflow to apply transfer learning for pavement transverse cracking detection by using the scalogram of the Continuous Wavelet Transform as input to pre-trained convolutional neural networks. First, the filtered vehicle's vertical acceleration signals are converted into a scalogram through the Generalized Morse Wavelets and the Continuous Wavelet Transform. Then, the generated scalogram image is resampled to be used as input into the GoogLeNet, SqueezeNet, and ResNet18. Finally, 5-fold cross-validation is used to test the performance of fine-tuning the convolutional neural networks.

learning through the selected pre-trained CNNs for pavement transverse cracking detection via the vehicle's vertical acceleration employed in this work is presented in Fig. 10. A brief description of the selected pre-trained CNNs is shown below.

#### 3.7.1. GoogLeNet

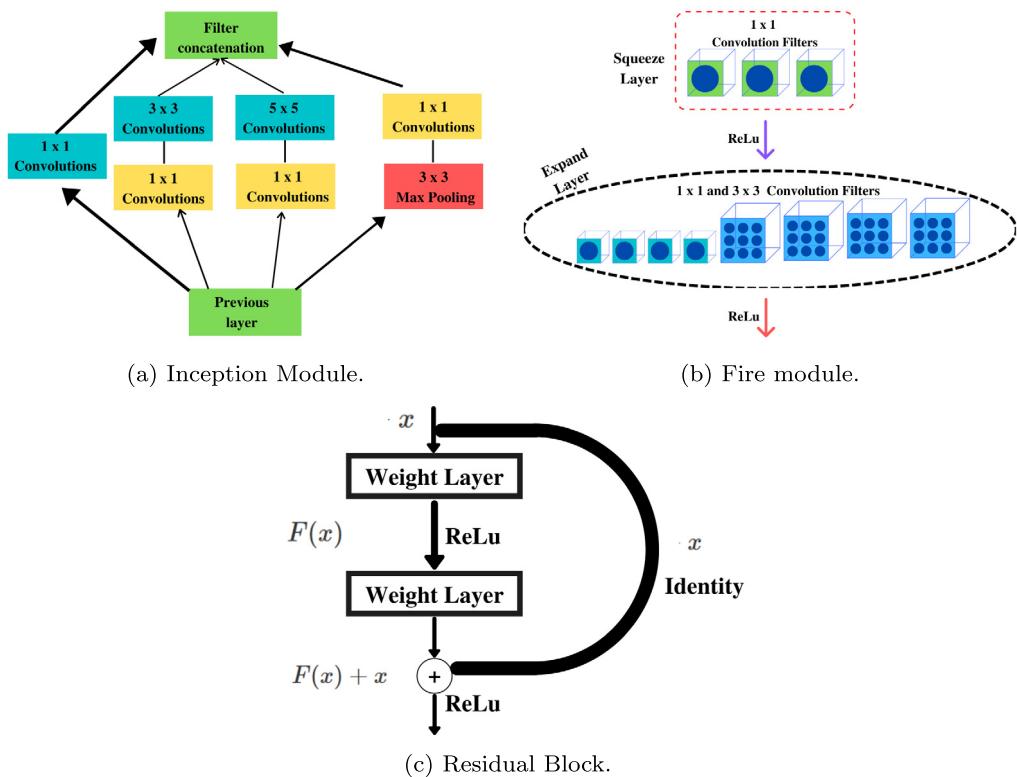
GoogLeNet is a deep CNN that consists of 22 layers. This CNN architecture won the ImageNet Large-Scale Visual Recognition Challenge in 2014. The main characteristic of GoogLeNet is that it is composed of Inception modules. An Inception module consists of  $1 \times 1$ ,  $3 \times 3$ , and  $5 \times 5$  convolution filters and a  $3 \times 3$  max pooling layer performed in parallel at the input rather than sequentially as traditional CNNs. Consequently, the output of each convolution filter and the max pooling layer are stacked together to produce a final output. The reason for performing different convolution sizes in parallel is to handle objects at different scales better. In addition, the GoogLeNet employs  $1 \times 1$  convolutions at the middle of the architecture to reduce the number of parameters. The structure of the Inception module can be appreciated in Fig. 11(a). Finally, at the end of the architecture, the GoogLeNet uses a global average pooling technique to reduce the number of learning parameters to zero. The above is achieved by taking  $7 \times 7$  feature maps which are averaged to produce  $1 \times 1$  feature maps (Szegedy et al., 2015).

#### 3.7.2. SqueezeNet

The SqueezeNet architecture is a deep CNN of 18 layers that intends to reduce the number of parameters of a CNN as much as possible. The above is done with the purpose of generating a CNN architecture that can be embedded in limited memory hardware platforms such as field-programmable gate arrays, can be easily communicated across servers while performing distributed training, and requires less bandwidth to be exported from the cloud to an autonomous car. The reduction of the parameters is performed through what is called a Fire module. The Fire module comprises two layers, a squeeze layer whose output is fed to an expand layer. The squeeze layer employs  $1 \times 1$  convolution filters to reduce the number of input elements, while the expand layers use  $1 \times 1$  and  $3 \times 3$  convolution filters in parallel for multi-scale learning and concatenating. The structure of the Fire module proposed in the SqueezeNet can be appreciated in Fig. 11(b). Besides, the SqueezeNet downsamples late in the network so that the convolution filters have large activation maps to increase the network's accuracy (Iandola et al., 2016; Wang et al., 2019).

#### 3.7.3. ResNet18

The ResNet architecture, where the term ResNet stands for residual network, is characterized by a skip connection that, as its name



**Fig. 11.** Basic modules and building blocks of the pre-trained convolutional neural networks used in this study. (a) Shows the Inception module used in the GoogLeNet. (b) Shows the Fire module used in the SqueezeNet. (c) Shows the residual block used in the ResNet18.

implies, skips one or more layers. The block used in a residual neural network, also known as a residual block, is illustrated in Fig. 11(c). This architecture allows training very deep neural networks of over a hundred layers, which usually can lead to a better performance in the training set. One of the main advantages of residual networks is that while a plain neural network, in theory, can reduce the training error, the deeper it is in reality, the training error gets worse since the optimization algorithm struggles more to find an optimal value for the weights. On the other hand, the residual network has the advantage of dealing with the problem of vanishing and exploding gradients due to their skip connection. This performance is attributed to their ability to learn the identity function through the skip connection. This skip connection or short-cut neither increases the number of parameters nor the computational complexity of the network (He et al., 2016).

## 4. Results

### 4.1. Execution environment

The computing of the ARMSE to determine the parameters of the GMWs and CWT and the cross-validation process used to fine-tune the pre-trained CNNs for pavement transverse cracking detection based on the vehicle's vertical acceleration data were performed in an NVIDIA Graphics Processing Unit (GPU) GeForce GTX 1050 and 8.00 GB of RAM. The MATLAB 2022a implementation of the selected pre-trained CNNs was used to fine-tune them using the scalograms of the vehicle's vertical acceleration signals. In addition, the available MATLAB functions to compute the CWT and the inverse CWT were also used to generate the scalogram employed as input into the CNNs.

### 4.2. Results of generalized morse wavelets and continuous wavelet transform parameter selection

Considering the methodology presented in Section 3, the algorithm 1, and the relation shown in Eq. (10), the behavior of the ARMSE of the

vehicle's vertical acceleration signals for increasing values of  $\beta$  and VPO can be appreciated in Fig. 12(a). On the other hand, Fig. 12(b) shows the detailed behavior of the ARMSE for values of  $\beta < 4$ . Furthermore, Table 5 shows the values of  $\beta$  and  $P^2$  that produce the minimum ARMSE according to the behavior shown in Figs. 12(a) and 12(b) for each curve of VPO. It is possible to observe that the values of  $P^2$  and  $\beta$  that produce the minimum ARMSE are similar or, in some cases, equal across the curves of VPO. Nevertheless, as shown in Table 5, increasing the number of VPO increases the minimum ARMSE.

Fig. 12(c) shows the surface of the ARMSE based on beta values and VPO. In addition, the minimum ARMSE of the surface is highlighted through a red dot. In this case, the ARMSEs obtained for each curve of VPO presented in Table 5 can be interpreted as local minima of the surface presented in Fig. 12(c), while the minimum ARMSE shown as a red dot can be interpreted as the global minimum of the surface. This minimum ARMSE occurs when  $\beta = 1.3333$  and the number of VPO equals 1.

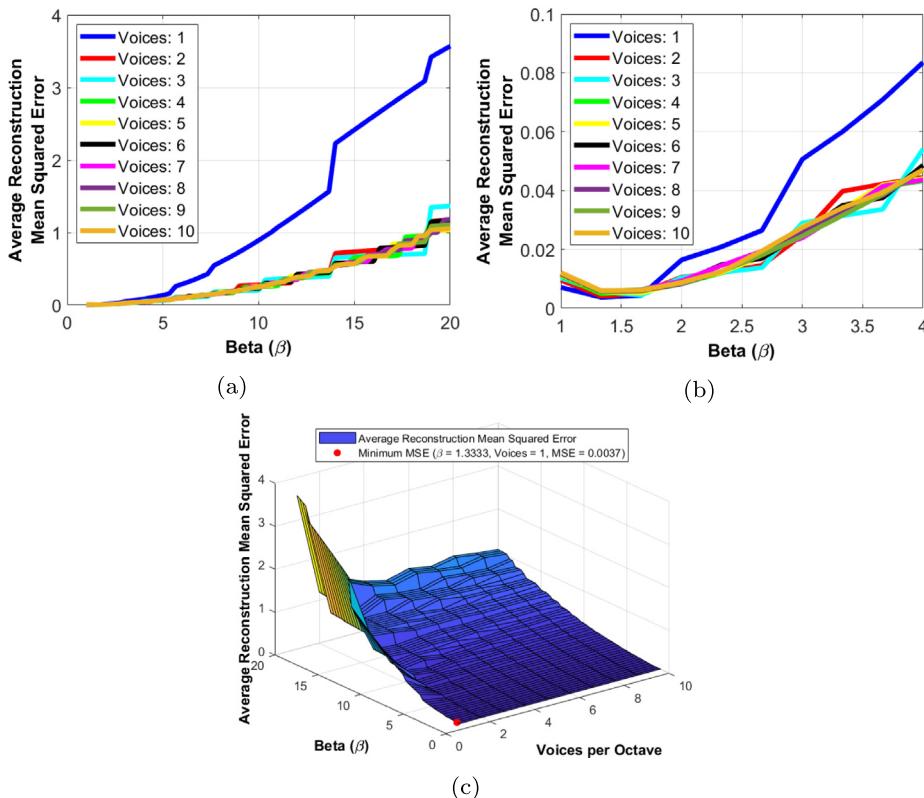
Based on the behavior shown in Fig. 12(b) and the ARMSEs presented in Table 5, it appears that the minimum ARMSE is similar for each curve of VPO that was tested and occur at similar values of  $\beta$ . Considering that the ARMSE is computed based on the individual Reconstruction MSEs of each signal in the dataset, it was decided to test statistically the difference of the minimum ARMSE obtained for constant values of VPO and varying values of  $\beta$ . Fig. 13(a) shows the box plots of the individual Reconstruction MSEs of the signals in the dataset that produce the minimum ARMSE for each curve of VPO presented in Figs. 12(a) and 12(b). It is possible to notice outliers in the Reconstruction MSE; the number of outliers and the percentage they represent from the total sample size is shown in Table 5. The behavior of the distribution of the Reconstruction MSEs implies that it is not symmetric and has a positive skew.

Considering the above, a log transformation was applied to the Reconstruction MSEs to determine the statistical difference between the minimum ARMSE for each curve of VPO presented in Fig. 12(a). The log

**Table 5**

Minimum ARMSE for each curve of VPO with the corresponding time-bandwidth product and beta that generate it. Besides, the number of Mean Squared Error outliers that produced the minimum ARMSE for each curve of VPO and their percentage before and after applying the log transformation are presented.

| Voices per octave | Time-bandwidth product ( $P^2$ ) | Beta ( $\beta$ ) | ARMSE  | Reconstruction Mean Squared Errors outliers | Reconstruction Mean Squared Errors percentage of outliers | Log reconstruction Mean Squared Errors outliers | Log reconstruction Mean Squared Errors percentage of outliers |
|-------------------|----------------------------------|------------------|--------|---|---|---|---|
| 1                 | 4                                | 1.3333           | 0.0037 | 28  | 8.5627  | 1   | 0.3058  |
| 2                 | 4                                | 1.3333           | 0.0041 | 31  | 9.4801  | 3   | 0.9174  |
| 3                 | 5                                | 1.6667           | 0.0044 | 21  | 6.4220  | 3   | 0.9174  |
| 4                 | 4                                | 1.3333           | 0.0049 | 31  | 9.4801  | 4   | 1.2232  |
| 5                 | 5                                | 1.6667           | 0.0052 | 26  | 7.9511  | 4   | 1.2232  |
| 6                 | 4                                | 1.3333           | 0.0056 | 31  | 9.4801  | 4   | 1.2232  |
| 7                 | 4                                | 1.3333           | 0.0057 | 32  | 9.7859  | 3   | 0.9174  |
| 8                 | 4                                | 1.3333           | 0.0057 | 28  | 8.5627  | 4   | 1.2232  |
| 9                 | 4                                | 1.3333           | 0.0057 | 30  | 9.1743  | 4   | 1.2232  |
| 10                | 4                                | 1.3333           | 0.0060 | 31  | 9.4801  | 4   | 1.2232  |



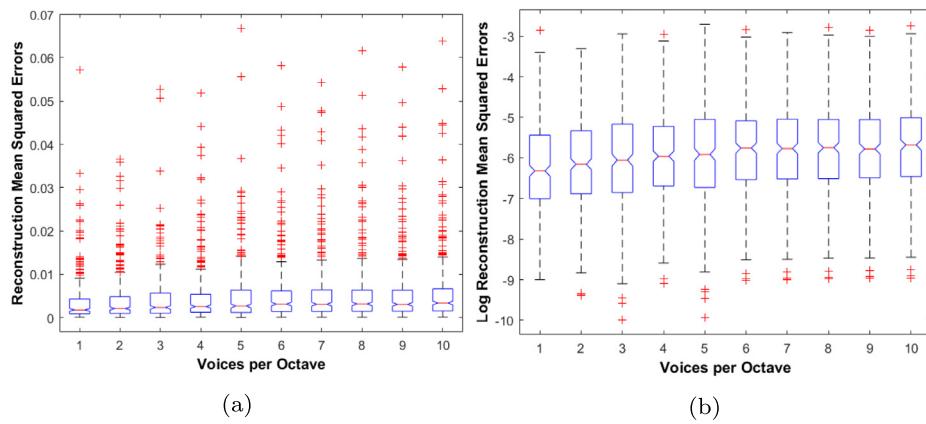
**Fig. 12.** The behavior of the ARMSE of the Continuous Wavelet Transform computed with the Generalized Morse Wavelets for values of  $\beta$  ranging from 1 to 20, and integer values of VPO ranging from 1 to 10. (a) Shows the behavior of the ARMSE of the Generalized Morse Wavelets for increasing values  $\beta$  ranging from 1 to 20 and values of VPO ranging from 1 to 10. (b) Shows the behavior of the ARMSE of the Generalized Morse Wavelets for increasing values of  $\beta$  ranging from 1 to 4 and values of VPO ranging from 1 to 10. (c) Shows the surface of the ARMSE depending on the beta parameter of the Generalized Morse Wavelets and the number of VPO for the vehicle's vertical acceleration signals used for transverse cracking asphalt pavement detection. The red dot indicates the position of the generated surface's minimum ARMSE.

transformation is commonly used to fulfill the normality requirements of parametric statistical tests of data that does not have a symmetric distribution (Hammouri et al., 2020; Feng et al., 2013; Keene, 1995). This comparison between the minimum ARMSEs obtained for different values of VPO was made to determine if the increase in the scale resolution of the CWT significantly impacts the minimum ARMSE since the minimum values are similar or, in some cases, equal, as shown in Fig. 12 and Table 5 and occur at similar values of  $\beta$  across values of VPO.

The boxplots in Fig. 13(b) show the distribution of the Reconstruction MSEs after applying the log transformation. By making this process, the number of outliers has been reduced, and the distribution of the Reconstruction MSEs appears symmetric and normally distributed. After applying the log transformation, the exact number of outliers and their percentage can be appreciated in Table 5.

A One-Way ANOVA and multiple comparisons test were executed to test the difference in the minimum ARMSEs for each curve of VPO. Nonetheless, a One-Way ANOVA requires that data is normally distributed and that the variances between groups are the same (Wu and Hamada, 2011). Thus, a Shapiro-Wilk test was used to test the normality of the log Reconstruction MSEs, and Bartlett's test was used to test the equal variance of the compared groups.

The Shapiro-Wilk test is used to test the normality of a dataset. The null hypothesis states that the data comes from a normally distributed population, while the alternative hypothesis states that the data does not come from a normally distributed population (Ghasemi and Zahedi, 2012). The results of the Shapiro-Wilk test can be appreciated in Table 6 along with the test statistic and  $p$ -value and considering a significance level of  $\alpha = 0.05$ . The significance level of 0.05 was chosen since it is frequently used in practice (Ross, 2017; Riffenburgh and Gillen, 2020). It is possible to appreciate that the  $p$ -values are far



**Fig. 13.** Box plots of the Reconstruction Mean Squared Errors that produce the minimum ARMSE for each curve of VPO. (a) Shows the box plots of the Reconstruction Mean Squared Errors of the dataset vehicle's vertical signals for the values of  $\beta$  that produce the minimum ARMSE for each curve of VPO. (b) Shows the box plots of the log Reconstruction Mean Squared Errors of the dataset vehicle's vertical signals for the values of  $\beta$  that produce the minimum ARMSE for each curve of VPO.

**Table 6**

Shapiro–Wilk test results used to assess the normality and equal variances of the groups of log Reconstruction MSEs that produce the minimum ARMSE for each curve of VPO.

| Shapiro–Wilk test |                |         |
|-------------------|----------------|---------|
| $\alpha = 0.05$   |                |         |
| Voices per octave | Test statistic | p-value |
| 1                 | 0.9956         | 0.4881  |
| 2                 | 0.9968         | 0.7727  |
| 3                 | 0.9941         | 0.2089  |
| 4                 | 0.9973         | 0.8615  |
| 5                 | 0.9950         | 0.3171  |
| 6                 | 0.9975         | 0.9008  |
| 7                 | 0.9973         | 0.8729  |
| 8                 | 0.9976         | 0.9226  |
| 9                 | 0.9972         | 0.8594  |
| 10                | 0.9977         | 0.9309  |

greater than the confidence level of 0.05. Thus, the null hypothesis that data comes from a normally distributed population is not rejected. This suggests that the log transformation helps to obtain a symmetric distribution in the Reconstruction MSEs.

On the other hand, Bartlett's test was conducted to test if the variances across the log Reconstruction MSEs that produced the minimum ARMSE for each curve of VPO are equal. The null hypothesis of Bartlett's test states that the variances of different data samples are the same. In contrast, the alternative hypothesis states that there are at least two data samples with unequal variances (Lim and Loh, 1996). The results of Bartlett's test and Bartlett's statistic can be appreciated in Table 7 with a significance level of 0.05. Based on the p-value, the null hypothesis of equal variance cannot be rejected, implying that the variances across the log Reconstruction MSEs that produce the minimum ARMSE are equal across VPO.

Considering that the compared log Reconstruction MSEs come from normally distributed data and that the compared groups have equal variances, the One-Way ANOVA was performed. The One-Way ANOVA and multiple comparisons test results are presented in Table 8. The null hypothesis of a One-Way ANOVA states that the mean across groups is equal, while the alternative hypothesis states that at least two groups have different means. The null hypothesis of equal means can be rejected according to the obtained p-value and considering a significance level of 0.05. Finally, a multiple comparisons test was performed to determine the groups of Reconstruction MSEs for each curve of VPO with a statistically significant difference in their means.

A modified significance level is considered following the Bonferroni correction for the multiple comparisons test to reduce the Type 1 error (i.e., the rejection of the null hypothesis when it is true) (Andrade,

**Table 7**

Bartlett's test results used to assess the equal variances of the groups of log Reconstruction MSEs that produce the minimum ARMSE for each curve of voices per octave.

| Bartlett's test      |       |         |                    |
|----------------------|-------|---------|--------------------|
| $\alpha = 0.05$      |       |         |                    |
| Voices per octave    | Count | Mean    | Standard deviation |
| 1                    | 327   | -6.2276 | 1.10197            |
| 2                    | 327   | -6.1367 | 1.1574             |
| 3                    | 327   | -6.0692 | 1.20191            |
| 4                    | 327   | -5.938  | 1.12364            |
| 5                    | 327   | -5.944  | 1.22733            |
| 6                    | 327   | -5.793  | 1.12202            |
| 7                    | 327   | -5.7832 | 1.12797            |
| 8                    | 327   | -5.7821 | 1.11612            |
| 9                    | 327   | -5.7775 | 1.11771            |
| 10                   | 327   | -5.7273 | 1.11742            |
| Pooled               | 3270  | -5.9179 | 1.14203            |
| Bartlett's statistic |       |         | 7.646              |
| Degrees of freedom   |       |         | 9                  |
| p-value              |       |         | 0.5700             |

2019; Sedgwick, 2012). The modified significance level is obtained using Eq. (17), where  $K$  is the number of comparisons of the multiple comparisons test,  $\alpha$  is the significance level of the One-Way ANOVA, and  $\alpha^*$  is the modified significance level. Based on this correction, the modified significance level for the multiple comparisons test is  $\alpha^* = 0.05/45 = 0.0011$ . The results of the multiple comparisons test are shown in Table 8.

$$\alpha^* = \alpha/K \quad (17)$$

This analysis shows that a minimum ARMSE of the vehicle's vertical acceleration signals can be obtained by setting  $\gamma = 3$ ,  $\beta = 1.3333$ , and the number of VPO equal to 1. Even though one VPO produces the lowest scale resolution of the CWT, the minimum ARMSE obtained is statistically different compared to the ones obtained for VPO between 6 and 10, as shown in Table 8, which produce a higher ARMSE as shown in Table 5. In the case of the number of voices between 2 and 5, the minimum ARMSE difference is not statistically significant. Thus, it was decided to employ the lowest scale resolution to fine-tune the CNNs since the computational cost of computing the CWT is lower compared to employing higher values of VPO (Bentley and McDonnell, 1994).

Fig. 14 shows an example that compares the vehicle's vertical acceleration signal associated with a transverse crack in asphalt pavement with the reconstructed one generated through the inverse CWT by considering  $\gamma = 3$ ,  $\beta = 1.3333$ , and the number of VPO to 1. It is possible to notice that the original signal and the reconstructed one overlap for most of their amplitude changes.

**Table 8**

Results of the One-Way ANOVA test and the multiple comparisons test to determine the difference between the minimum ARMSEs of the vehicle's vertical acceleration signals obtained for different values of VPO. The significance level for the One-Way ANOVA is  $\alpha = 0.05$ . The modified significance level of the multiple comparisons test is  $\alpha^* = 0.0011$ . In bold are shown the significant p-values.

| One-Way ANOVA             |                  |                    |                 |             |                              |
|---------------------------|------------------|--------------------|-----------------|-------------|------------------------------|
| Source of variation       | Sum of squares   | Degrees of freedom | Mean of squares | F-Statistic | p-value                      |
| Between groups            | 90.25            | 9                  | 10.0275         |             |                              |
| Within groups             | 4251.77          | 3260               | 1.3042          | 7.69        | <b>2.93454<sup>-11</sup></b> |
| Total                     | 4342.02          | 3269               | Not applicable  |             |                              |
| Multiple comparisons test |                  |                    |                 |             |                              |
| Group A (Voices)          | Group B (Voices) | Lower limit        | A-B             | Upper limit | p-value                      |
| 1                         | 2                | -0.37344           | -0.090878       | 0.19168     | 0.9913                       |
| 1                         | 3                | -0.44097           | -0.15841        | 0.12415     | 0.75204                      |
| 1                         | 4                | -0.57219           | -0.28963        | -0.0070682  | 0.039219                     |
| 1                         | 5                | -0.56611           | -0.28355        | -0.00098642 | 0.048358                     |
| 1                         | 6                | -0.71717           | -0.43461        | -0.15205    | <b>4.9744<sup>-5</sup></b>   |
| 1                         | 7                | -0.72692           | -0.44436        | -0.1618     | <b>2.858<sup>-5</sup></b>    |
| 1                         | 8                | -0.72805           | -0.44549        | -0.16293    | <b>2.6783<sup>-5</sup></b>   |
| 1                         | 9                | -0.73261           | -0.45005        | -0.16749    | <b>2.0571<sup>-5</sup></b>   |
| 1                         | 10               | -0.78289           | -0.50033        | -0.21777    | <b>9.3017<sup>-7</sup></b>   |
| 2                         | 3                | -0.3501            | -0.067536       | 0.21502     | 0.99911                      |
| 2                         | 4                | -0.48131           | -0.19875        | 0.083809    | 0.43913                      |
| 2                         | 5                | -0.47523           | -0.19267        | 0.089891    | 0.48668                      |
| 2                         | 6                | -0.62629           | -0.34373        | -0.061173   | 0.0046668                    |
| 2                         | 7                | -0.63605           | -0.35349        | -0.070927   | 0.0030306                    |
| 2                         | 8                | -0.63718           | -0.35462        | -0.072056   | 0.0028804                    |
| 2                         | 9                | -0.64173           | -0.35917        | -0.076612   | 0.0023414                    |
| 2                         | 10               | -0.69201           | -0.40945        | -0.12689    | <b>0.00019594</b>            |
| 3                         | 4                | -0.41377           | -0.13121        | 0.15135     | 0.90441                      |
| 3                         | 5                | -0.40769           | 0.12513         | 0.15743     | 0.92732                      |
| 3                         | 6                | -0.55876           | -0.2762         | 0.006363    | 0.061761                     |
| 3                         | 7                | -0.56851           | -0.28595        | -0.0033912  | 0.044548                     |
| 3                         | 8                | -0.56964           | -0.28708        | -0.0045195  | 0.04285                      |
| 3                         | 9                | -0.5742            | -0.29164        | -0.0090757  | 0.036548                     |
| 3                         | 10               | -0.62447           | -0.34191        | -0.059354   | 0.0050503                    |
| 4                         | 5                | -0.27648           | 0.0060818       | 0.28864     | 1                            |
| 4                         | 6                | -0.42754           | -0.14498        | 0.13758     | 0.83708                      |
| 4                         | 7                | -0.4373            | -0.15474        | 0.12782     | 0.77701                      |
| 4                         | 8                | -0.43842           | -0.15587        | 0.12669     | 0.76947                      |
| 4                         | 9                | -0.44298           | -0.16042        | 0.12214     | 0.73793                      |
| 4                         | 10               | -0.49326           | -0.2107         | 0.071861    | 0.3508                       |
| 5                         | 6                | -0.43362           | -0.15106        | 0.1315      | 0.80074                      |
| 5                         | 7                | -0.44338           | -0.16082        | 0.12174     | 0.73511                      |
| 5                         | 8                | -0.44451           | -0.16195        | 0.12061     | 0.72701                      |
| 5                         | 9                | -0.44906           | -0.1665         | 0.11606     | 0.69343                      |
| 5                         | 10               | -0.49934           | -0.21678        | 0.065779    | 0.30936                      |
| 6                         | 7                | -0.29231           | -0.0097542      | 0.27281     | 1                            |
| 6                         | 8                | -0.29344           | -0.010883       | 0.27168     | 1                            |
| 6                         | 9                | -0.298             | -0.015439       | 0.26712     | 1                            |
| 6                         | 10               | -0.34828           | -0.065717       | 0.21684     | 0.99928                      |
| 7                         | 8                | -0.28369           | -0.0011284      | 0.28143     | 1                            |
| 7                         | 9                | -0.28824           | -0.0056845      | 0.27688     | 1                            |
| 7                         | 10               | -0.33852           | -0.055963       | 0.2266      | 0.99981                      |
| 8                         | 9                | -0.28712           | -0.0045562      | 0.278       | 1                            |
| 8                         | 10               | -0.33739           | -0.054834       | 0.22773     | 0.99984                      |
| 9                         | 10               | -0.33284           | -0.050278       | 0.23228     | 0.99992                      |

#### 4.3. Results of transfer learning applied through convolutional neural networks

This work used a 5-fold cross-validation strategy to measure the performance of the fine-tuned CNNs trained with the scalogram of the vehicle's vertical acceleration signals generated through the GMWs parameters that produce the minimum ARMSE. The metrics used to measure and compare the performance of the fine-tuned CNNs were accuracy, precision, recall, specificity, negative predictive value (NPV), and F1-score derived from a binary confusion matrix (Martínez-Ríos et al., 2022). The proportion of correctly identified observations among all the tested observations is known as the accuracy as expressed in Eq. (18). Where TP refers to the true positives, TN are the true negatives, FP are the false positives, and FN are the false negatives. Moreover, specificity (see Eq. (19)) is the percentage of successfully categorized true negative data points, and sensitivity or recall (see Eq. (20)) shows the percentage of accurately recognized real positive

cases. The ratio of true positives to true positives plus false positives is known as precision, as shown in Eq. (21). The NPV (see Eq. (22)) is the proportion of negative predicted results that are true negatives (Luque et al., 2019).

Besides, the F1-score is the harmonic mean between precision and recall, as presented in Eq. (23). A value of 1 of F1-score indicates a better classifier, while a value closer to 0 indicates a lower performance (Chicco and Jurman, 2020). In this study, the positive class (TP) is the vehicle's vertical acceleration signals associated with the transverse cracking asphalt pavement sections. The negative class (TN) is the vehicle's vertical acceleration signals associated with uncracked pavement sections.

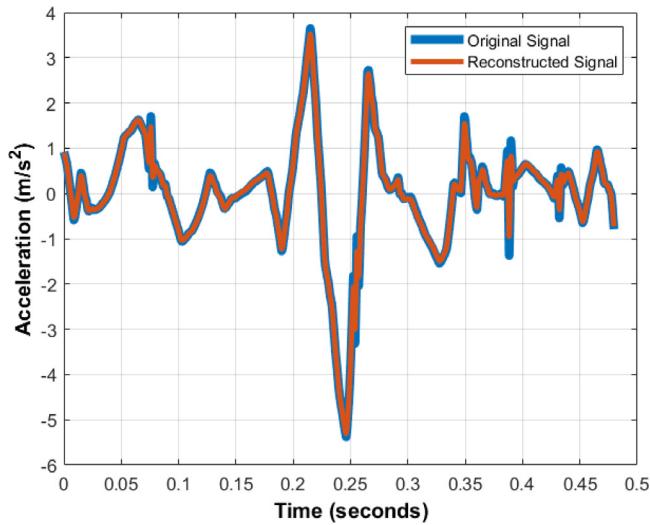
$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (18)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (19)$$

**Table 9**

5-fold cross-validation results of fine-tuning the GoogLeNet for pavement transverse cracking detection based on vehicle's vertical acceleration signals.

| Fold               | Training accuracy % | Validation accuracy % | Validation sensitivity % | Validation specificity % | Validation precision % | Validation NPV % | Validation F1-score % | Validation AUC | Training time |
|--------------------|---------------------|-----------------------|--------------------------|--------------------------|------------------------|------------------|-----------------------|----------------|---------------|
| 1                  | 99.2337             | 81.8182               | 77.2727                  | 84.0909                  | 70.8333                | 88.0952          | 73.9130               | 0.9236         | 6 min 16 s    |
| 2                  | 100.0000            | 95.3846               | 90.9091                  | 97.6744                  | 95.2381                | 95.4545          | 93.0233               | 0.9757         | 5 min 59 s    |
| 3                  | 100.0000            | 86.1538               | 66.6667                  | 95.4545                  | 87.5000                | 85.7143          | 75.6757               | 0.9318         | 5 min 57 s    |
| 4                  | 99.2366             | 87.6923               | 72.7273                  | 95.3488                  | 88.8889                | 87.2340          | 80.0000               | 0.9598         | 6 min 1 s     |
| 5                  | 96.5517             | 89.3939               | 77.2727                  | 95.4545                  | 89.4737                | 89.3617          | 82.9268               | 0.9669         | 5 min 51 s    |
| Average            | 99.0044             | 88.0886               | 76.9697                  | 93.6047                  | 86.3868                | 89.1720          | 81.1078               | 0.9516         | 6.0133 min    |
| ±                  | ±                   | ±                     | ±                        | ±                        | ±                      | ±                | ±                     | ±              | ±             |
| Standard deviation | 1.4234              | 4.9536                | 8.9253                   | 5.4074                   | 9.1827                 | 3.7542           | 7.5463                | 0.0227         | 0.1547 min    |



**Fig. 14.** Comparison of the original vehicle's vertical acceleration signal and reconstructed version obtained from the inverse CWT by considering one VPO, a  $\gamma = 3$ , and a  $\beta = 1.3333$ . The blue line shows the original vehicle's vertical acceleration signal, and the red line shows the reconstructed version from the wavelet coefficients and the inverse CWT.

$$\text{Recall/Sensitivity} = \frac{TP}{TP + FN} \quad (20)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (21)$$

$$NPV = \frac{TN}{TN + FN} \quad (22)$$

$$F1 - score = \frac{2(Precision)(Recall)}{Precision + Recall} \quad (23)$$

Another considered metric was the Area Under the Receiver Operating Characteristic Curve (AUC). The Receiver Operating Characteristic (ROC) Curve shows a classification model's sensitivity/specificity trade-off for any classification threshold. This curve allows the ranking and selection of classification models under particular user needs, frequently linked to differential error costs and accuracy demands. A perfect classifier is represented by an AUC of 1, whereas a random classifier represents a value of 0.5. Since the AUC is calculated using the entire ROC curve, which includes all possible classification thresholds, the AUC is a robust metric used to assess the effectiveness of score classifiers (Melo, 2013).

The Transfer Learning approach was used to adjust the weights of pre-trained CNNs based on the scalogram images generated from the vehicle's vertical acceleration signals available in the dataset. The scalograms were generated by setting the number of VPO equal to 1, the parameters gamma and beta of the GMWs were set to 3 and 1.3333, respectively, which correspond to a time-bandwidth product ( $P^2$ ) of 4 as shown in Table 5. These parameters produce the minimum ARMSE,

as shown previously. Fig. 15 shows the generated scalogram of the vehicle's vertical acceleration signal associated with a transverse cracking in the pavement section and an uncracked pavement section used to train each of the selected CNNs. The maximum number of epochs was defined as 25 for each fold in the 5-fold cross-validation strategy. The learning rate for all networks was set as 0.0001. Stochastic gradient descent with momentum was used to optimize the networks' weights during training. The momentum was set as 0.9, and the mini-batch size was 15.

For the GoogLeNet fine-tuning, the network's last layer was replaced with a fully connected layer with an output size of 2 by considering the number of classes considered in this study; this process was repeated for SqueezeNet and ResNet18. The input data for the GoogLeNet must be resampled to an input size of  $224 \times 224$ . Additionally, the input data must be normalized with zero center normalization.

The results for each of the 5-folds of the cross-validation fine-tuning process of the GoogLeNet can be noticed in Table 9. In addition, Table 9 shows the average metrics when considering the results of each fold with the corresponding standard deviation. The training time for each fold is presented in the same Table. Moreover, the ROC curves of the fine-tuned CNNs are presented in Fig. 16. Fig. 16(a) shows the validation ROC curve of each fold with the corresponding validation AUC of the GoogLeNet. The training and validation accuracies and the training and validation loss are presented in Fig. 17 to assess the overfitting and underfitting in each cross-validation fold for the GoogLeNet fine-tuning process.

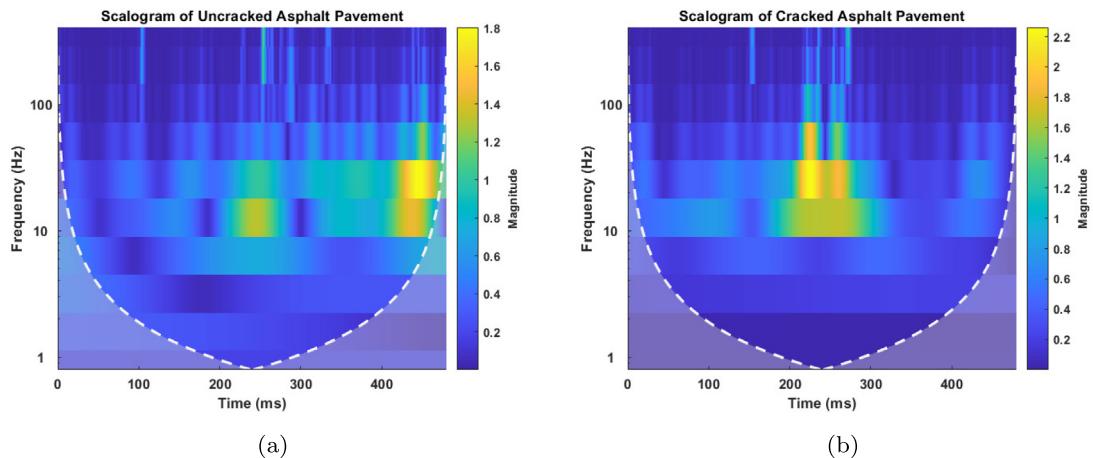
In the case of the SqueezeNet, it is necessary to resample the input images to a size of  $227 \times 227$ . Moreover, the input data must be normalized with zero center normalization. The 5-fold cross-validation results of fine-tuning the SqueezeNet for pavement transverse cracking detection can be appreciated in Table 10. Fig. 16(b) shows the validation ROC curves for each fold with the corresponding AUC of the SqueezeNet. The training and validation accuracies and the training and validation loss for each of the five folds considered in this study are presented in Fig. 18 to assess the overfitting and underfitting of the fine-tuning of the SqueezeNet.

Finally, the results of fine-tuning the ResNet18 through 5-fold cross-validation can be appreciated in Table 11. For this CNN architecture, it is necessary to resample the scalogram image into a size of  $224 \times 224$ . In addition, it is necessary to normalize the input with a z-score normalization. Furthermore, Fig. 16(c) shows the validation ROC curves for the cross-validation folds with the corresponding AUC. In addition, Fig. 19 shows the training and validation accuracy and loss curves for each validation fold to evaluate the overfitting and underfitting of the CNN.

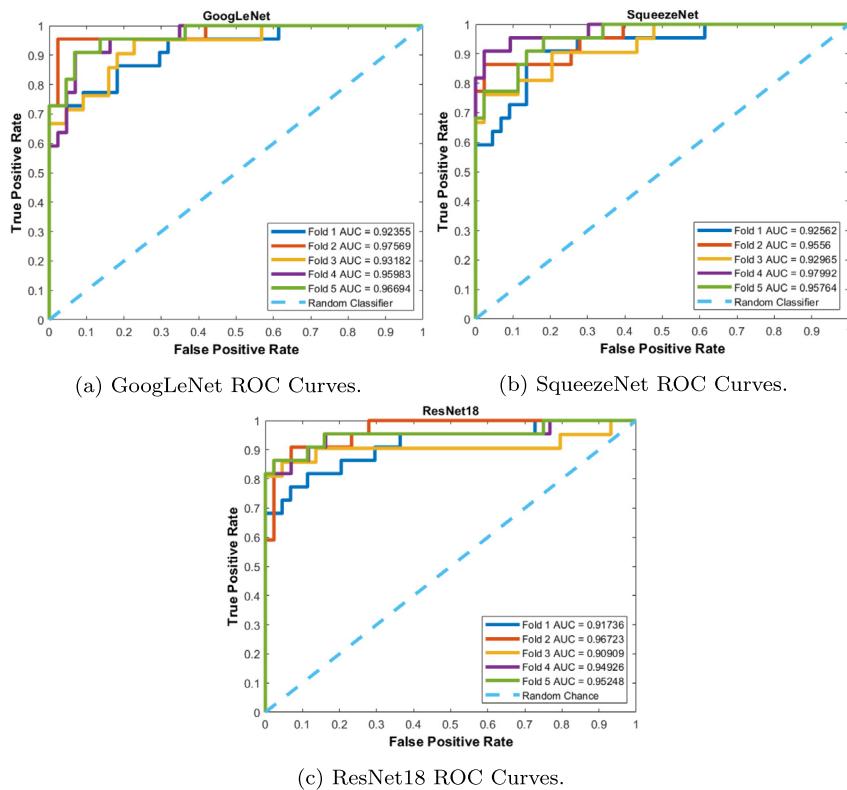
## 5. Discussion

### 5.1. Generalized morse wavelets parameter selection and continuous wavelet transform computation

The behavior that can be appreciated in Fig. 12 shows that the ARMSE of the vehicle's vertical acceleration signals is more sensitive to the changes of  $\beta$  compared to the modifications of VPO. The above can



**Fig. 15.** Examples of the produced scalograms of the vehicle's vertical acceleration signal used for pavement transverse cracking detection by considering the parameters of the Generalized Morse Wavelets as  $\gamma = 3$ , and  $\beta = 1.3333$ , while the VPO were defined as 1. (a) Shows the scalogram of the vehicle's vertical acceleration signal associated with an uncracked asphalt pavement section. (b) Shows the scalogram of the vehicle's vertical acceleration signals associated with transverse cracking in the asphalt pavement.

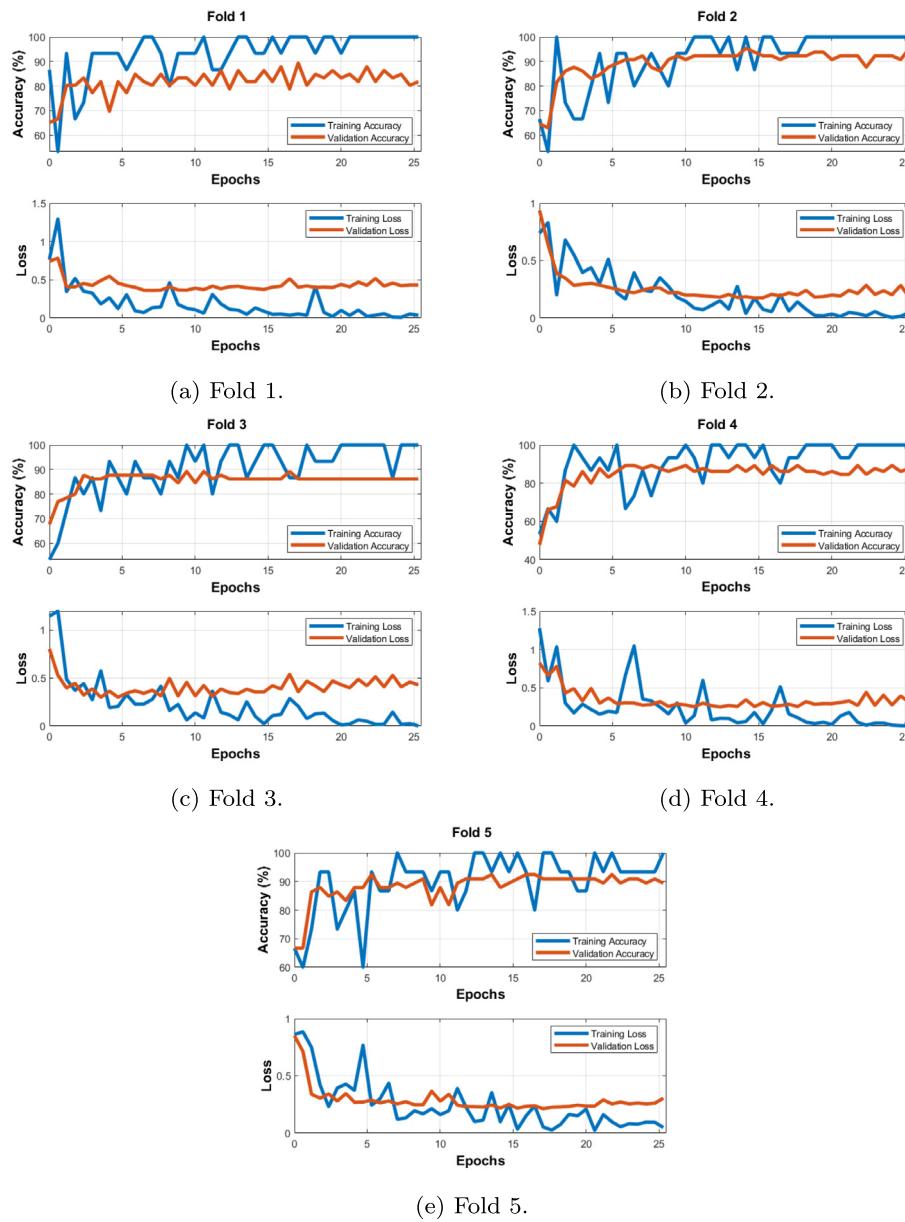


**Fig. 16.** ROC curves for the GoogleNet, SqueezeNet, and ResNet18 for each cross-validation fold. The corresponding Area Under the ROC curve is shown for each fold. (a) Shows the validation ROC curves with the corresponding validation AUC of the GoogLeNet. (b) Shows the validation ROC curves with the corresponding validation AUC of the SqueezeNet. (c) Shows the validation ROC curves with the corresponding validation AUC of the ResNet18.

**Table 10**

5-fold cross-validation results of fine-tuning the SqueezeNet for pavement transverse cracking detection based on vehicle's vertical acceleration signals.

| Fold               | Training accuracy % | Validation accuracy % | Validation sensitivity % | Validation specificity % | Validation precision % | Validation NPV % | Validation F1-score % | Validation AUC | Training time |
|--------------------|---------------------|-----------------------|--------------------------|--------------------------|------------------------|------------------|-----------------------|----------------|---------------|
| 1                  | 99.2337             | 84.8485               | 90.9091                  | 81.8182                  | 71.4286                | 94.7368          | 80.0000               | 0.9256         | 2 min 25 s    |
| 2                  | 99.2366             | 93.8462               | 86.3636                  | 97.6744                  | 95.0000                | 93.3333          | 90.4762               | 0.9556         | 2 min 42 s    |
| 3                  | 100.0000            | 90.7692               | 76.1905                  | 97.7273                  | 94.1176                | 89.5833          | 84.2105               | 0.9297         | 2 min 44 s    |
| 4                  | 98.4733             | 92.3077               | 81.8182                  | 97.6744                  | 94.7368                | 91.3043          | 87.8049               | 0.9799         | 2 min 43 s    |
| 5                  | 100.0000            | 87.8788               | 86.3636                  | 88.6364                  | 79.1667                | 92.8571          | 82.6087               | 0.9576         | 2 min 42 s    |
| Average            | 99.3887             | 89.9301               | 84.3290                  | 92.7061                  | 86.8899                | 92.3630          | 85.0201               | 0.9497         | 2.6533 min    |
| ±                  | ±                   | ±                     | ±                        | ±                        | ±                      | ±                | ±                     | ±              | ±             |
| Standard deviation | 0.6388              | 3.5959                | 5.5704                   | 7.2403                   | 10.9349                | 1.9792           | 4.1592                | 0.0223         | 0.1330 min    |



**Fig. 17.** Training and validation accuracy and training and validation loss for each cross-validation fold related to the fine-tuning of the GoogLeNet. (a) Results of Fold 1. (b) Results of Fold 2. (c) Results of Fold 3. (d) Results of Fold 4. (e) Results of Fold 5.

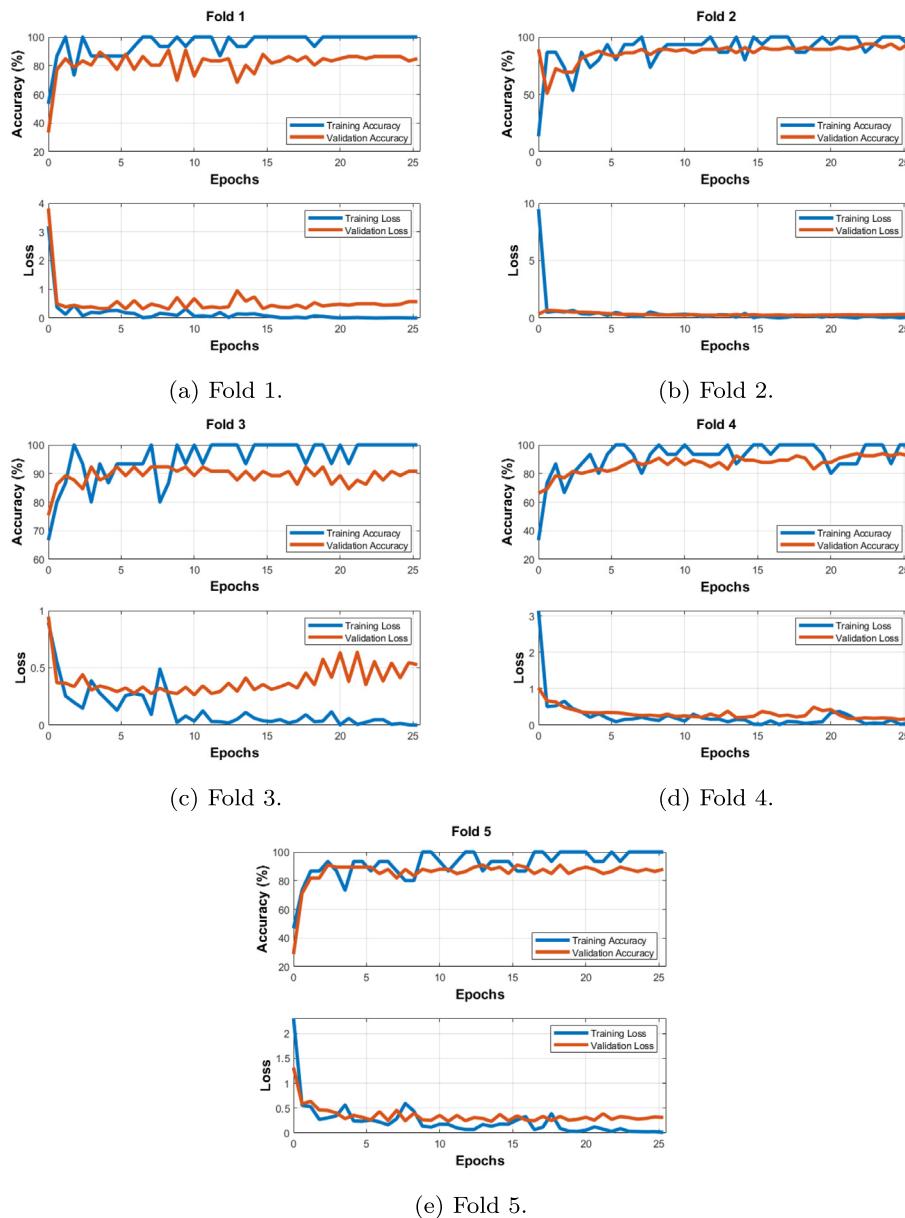
**Table 11**

5-fold cross-validation results of fine-tuning the ResNet18 for pavement transverse cracking detection based on vehicle's vertical acceleration signals.

| Fold               | Training accuracy % | Validation accuracy % | Validation sensitivity % | Validation specificity % | Validation precision % | Validation NPV % | Validation F1-score % | Validation AUC | Training time |
|--------------------|---------------------|-----------------------|--------------------------|--------------------------|------------------------|------------------|-----------------------|----------------|---------------|
| 1                  | 98.8506             | 86.3636               | 72.7273                  | 93.1818                  | 84.2105                | 87.2340          | 78.0488               | 0.9174         | 6 min 16 s    |
| 2                  | 99.6183             | 92.3077               | 81.8182                  | 97.6744                  | 94.7368                | 91.3043          | 87.8049               | 0.9672         | 5 min 30 s    |
| 3                  | 99.6183             | 90.7692               | 71.4286                  | 100.0000                 | 100.0000               | 88.0000          | 83.3333               | 0.9091         | 5 min 17 s    |
| 4                  | 99.6183             | 93.8462               | 81.8182                  | 100.0000                 | 100.0000               | 91.4894          | 90.0000               | 0.9493         | 5 min 25 s    |
| 5                  | 99.6169             | 92.4242               | 81.8182                  | 97.7273                  | 94.7368                | 91.4894          | 87.8049               | 0.9525         | 6 min 9 s     |
| Average            | 99.4645             | 91.1422               | 77.9221                  | 97.7167                  | 94.7368                | 89.9034          | 85.3984               | 0.9391         | 5.7233 min    |
| ±                  | ±                   | ±                     | ±                        | ±                        | ±                      | ±                | ±                     | ±              | ±             |
| Standard deviation | 0.3432              | 2.8848                | 5.3547                   | 2.7836                   | 6.4460                 | 2.1060           | 4.7707                | 0.0247         | 0.4513 min    |

be attributed to the sensitive changes of the time domain localization or time domain standard deviation of the GMWs to changes of  $\beta$  as presented in Fig. 4(b). Contrary to the common use of GMWs in the literature, where the parameters beta and gamma are set as  $\beta = 20$  and  $\gamma = 3$ , respectively, and the number of VPO is not frequently reported as explained by Martinez-Ríos et al. (2023). This study shows

that by computing the ARMSE and analyzing its behavior based on changes in  $\beta$  and the number of VPO used to compute the CWT, it can be noticed that an incorrect selection can provide an inaccurate time-frequency representation of the signals under study. The above is because the inverse CWT and the ARMSE are computed based on the wavelet coefficients obtained for a particular parameterization of the



**Fig. 18.** Training and validation accuracy and training and validation loss for each cross-validation fold related to the fine-tuning of the SqueezeNet. (a) Results of Fold 1. (b) Results of Fold 2. (c) Results of Fold 3. (d) Results of Fold 4. (e) Results of Fold 5.

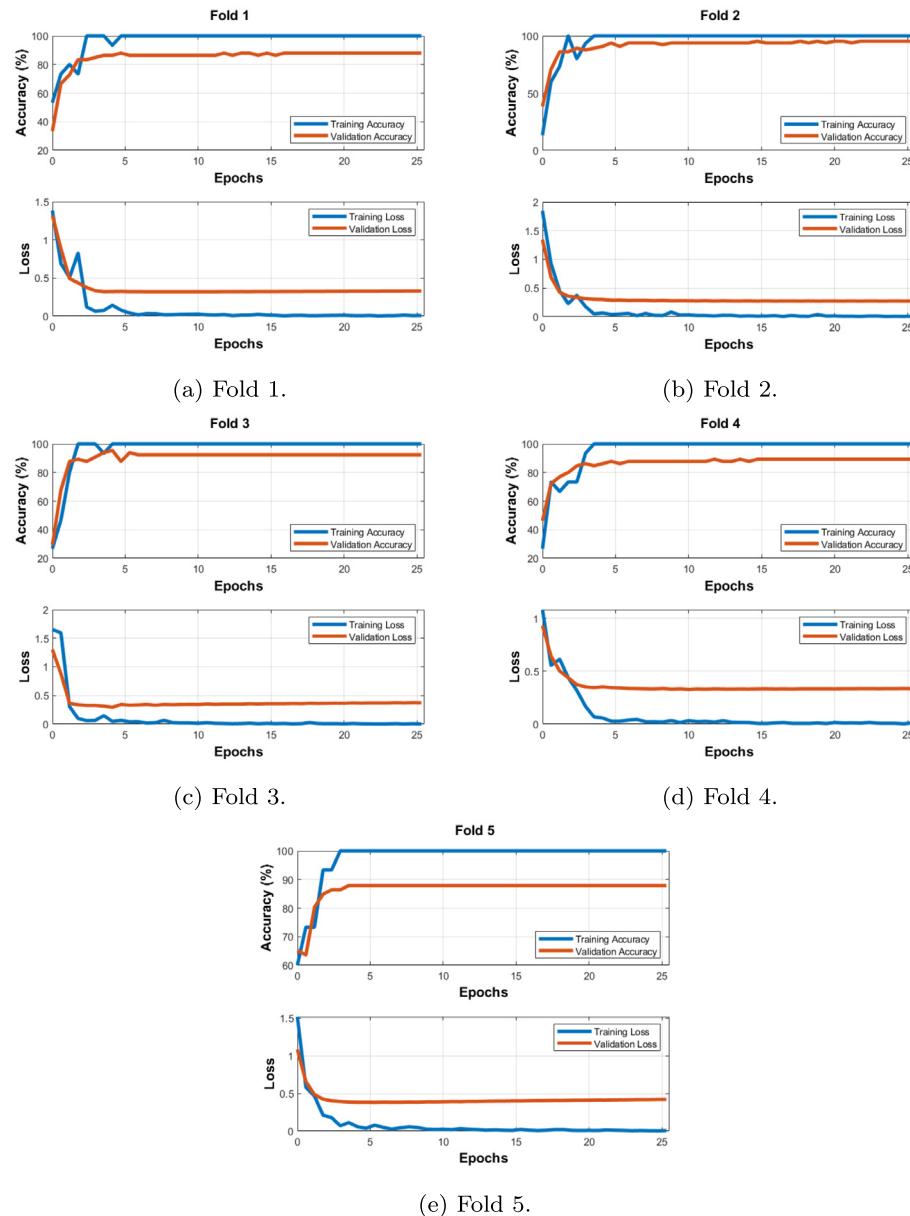
GMWs and CWT. Furthermore, based on the behavior shown in Fig. 12, it can be appreciated that when  $\beta = 20$ , the ARMSE is greater compared to lower values of  $\beta$ . The above highlights again the implications of the adequate parameterization of a mother wavelet when computing the CWT.

Additionally, as seen in Fig. 15, the scalograms of the vehicle's vertical acceleration signal associated with pavement transverse cracking and uncracked pavement can be distinguished in their energy distribution presented in the time-frequency plane when the parameters of the GMW are set as  $\beta = 1.3333$ , and  $\gamma = 3$  and choosing one VPO (which produces the minimum ARMSE). Moreover, the chosen  $\beta$  value is associated with a lower time domain standard deviation of GMWs (see Fig. 4(b)), which suggests a better time localization than selecting a higher value of  $\beta$ . In this case, it can be noticed that the vehicle's vertical acceleration signal associated with an uncracked pavement section has its energy distributed across frequencies between the range of 1 to 100 Hz (see Fig. 15(a)). On the other hand, the energy of a vehicle's vertical acceleration signal associated with a cracked pavement section (see Fig. 15(b)) is concentrated in the time interval

of 200 to 300 ms and within the frequency range of 10 to 100 Hz. This difference in the energy distribution of the vehicle's vertical acceleration while passing over pavement transverse cracks and uncracked pavement sections was also observed and discussed by Yang and Zhou (2021). The results mentioned above help to support the selection of the proposed parameterization to generate the scalogram used to fine-tune the pre-trained CNNs. These differences between both scalograms can be appreciated based on the proposed GMWs parameterization and despite selecting the lowest scale resolution of the CWT (i.e., one VPO). However, selecting a higher value of VPO could provide a better understanding of the frequency transitions of the signal across time.

## 5.2. Fine-tuning of pre-trained convolutional neural networks

The results of fine-tuning the pre-trained CNNs by employing as input the scalogram of the CWT computed by setting  $\gamma = 3$ ,  $\beta = 1.3333$ , and one VPO can be observed in Table 9 for the GoogLeNet, in Table 10 for the SqueezeNet, and in Table 11 for the ResNet18. Based on these



**Fig. 19.** Training and validation accuracy and training and validation loss for each cross-validation fold related to the fine-tuning of the ResNet18. (a) Results of Fold 1. (b) Results of Fold 2. (c) Results of Fold 3. (d) Results of Fold 4. (e) Results of Fold 5.

results, it can be observed that the SqueezeNet has a lower training time in each of the cross-validation folds that were considered compared to the ResNet18 and GoogLeNet. The lower training time can be attributed to the number of parameters of the SqueezeNet that are lower than the ones for the GoogleNet and ResNet18, as depicted in Table 4. Additionally, this CNN was initially designed to be easily embedded in low-memory hardware platforms, contributing to having a lower number of parameters and thus reducing its training time (Iandola et al., 2016).

Comparing the validation accuracy of the three pre-trained CNNs that were fine-tuned, it can be appreciated that ResNet18 achieved a better performance in terms of average validation accuracy with a value of  $91.1422 \pm 2.8848$  compared to the other tested CNNs. Nevertheless, since the dataset is unbalanced, as presented in Fig. 3 where the percentage of pavement sections with transverse cracks is lower than that of uncracked pavement sections, the average validation accuracy could lead to biased interpretation of the performance of the CNNs. Thus, other metrics, such as sensitivity and specificity, could provide

better insight related to the performance of the fine-tuned CNNs for detecting uncracked and cracked asphalt pavement sections based on the vehicle's vertical acceleration.

Regarding average validation sensitivity, SqueezeNet achieved a more significant performance with a value of  $84.3290 \pm 5.5704$  than GoogLeNet and ResNet18; however, the specificity of the SqueezeNet was lower with a value of  $92.7061 \pm 7.2403$ . In general, it can be appreciated that the sensitivity was lower than the specificity for each CNN; this could be attributed to the unbalanced proportion of the dataset since machine learning techniques tend to have a better performance on the over-represented class and lower performance on the under-represented class (Shi and Zhang, 2021). In the case of the SqueezeNet, the percentage of both average validation sensitivity and specificity are closer compared to the more significant difference between average validation sensitivity and specificity presented by the GoogLeNet and ResNet18 as shown in Tables 9, 10, and 11. Otherwise, the average validation F1-score is similar between SqueezeNet ( $85.0201 \pm 4.1592$ ) and ResNet18 ( $85.3984 \pm 4.7707$ ), with the average validation F1-score of ResNet18 being slightly higher than SqueezeNet. Nonetheless,

**Table 12**

Comparison of the results presented in this study with the literature that used the same dataset.

| Author             | Features  | Machine learning algorithm | Performance %  |
|--------------------|---|----------------------------|--|
| Yang et al. (2021) | Time domain<br>Frequency domain<br>Wavelet domain | Support vector machine     | Accuracy: 97.25<br>F1-score: 85.25   |
| This study         | Scalogram of CWT by employing the GMWs            | GoogLeNet                  | Accuracy: $88.0886 \pm 4.9536$<br>Sensitivity: $76.9697 \pm 8.9253$<br>Specificity: $93.6047 \pm 5.4074$<br>F1-score: $81.1078 \pm 7.5463$<br>AUC: $0.9516 \pm 0.0227$ |
| This study         | Scalogram of CWT by employing the GMWs            | SqueezeNet                 | Accuracy: $89.9301 \pm 3.5959$<br>Sensitivity: $84.3290 \pm 5.5704$<br>Specificity: $92.7061 \pm 7.2403$<br>F1-score: $85.0201 \pm 4.1592$<br>AUC: $0.9497 \pm 0.0223$ |
| This study         | Scalogram of CWT by employing the GMWs            | ResNet18                   | Accuracy: $91.1422 \pm 2.8848$<br>Sensitivity: $77.9221 \pm 5.3547$<br>Specificity: $97.7167 \pm 2.7836$<br>F1-score: $85.3984 \pm 4.7707$<br>AUC: $0.9391 \pm 0.0247$ |

SqueezeNet presented a lower standard deviation in its average validation F1-score than ResNet18, as shown in the results of Tables 10 and 11.

On the other hand, it is essential to consider that the reported sensitivity and specificity results consider a 0.5 threshold. Thus, to evaluate the behavior of the CNNs for every possible classification threshold, the validation ROC curve for each cross-validation fold was computed as shown in Fig. 16. Based on the behavior of the ROC curves, it can be observed that the CNNs, despite the unbalanced proportion of the dataset and its small size, have a better performance than that of a Random Classifier depicted in a dashed blue line since the validation ROC curves are more proximate to the top-left corner. The validation AUC of each ROC curve was computed, and the average AUC is presented in Tables 9, 10, and 11 for each of the selected pre-trained CNNs. In this case, it can be appreciated that the AUC for each validation ROC curve and the average AUC are more significant than 0.9. The above suggests that despite the difference in sensitivity and specificity between the proposed CNNs, their performance can be considered acceptable (Mandrekar, 2010). Comparing the three proposed CNNs, GoogLeNet provided a slightly greater average validation AUC with a value of  $0.9516 \pm 0.0227$  than SqueezeNet and GoogLeNet.

Ultimately, analyzing the training and validation curves for each of the folds shown in Figs. 17, 18, and 19, it can be noticed that there is a small degree of over-fitting for specific cross-validation folds since the training curves tend to achieve greater values of training accuracy compared to the accuracy values obtained for the validation curves. This over-fitting can be attributed to the depth of the CNNs, the number of parameters each network has, and the sample size of signals used to fine-tune the CNNs. Nonetheless, despite this slight overfit of the CNNs, the validation results in terms of AUC were more significant than 0.9 for all of the considered cross-validation folds (see Fig. 16). The over-fitting problem could be mitigated with regularization or data augmentation techniques or by training the networks with a greater sample size (Santos and Papa, 2022; Iwana, 2021).

### 5.3. Comparison with the literature

Since the pavement transverse crack detection dataset based on vehicle's vertical acceleration published by Yang et al. (2021) and Yang and Zhou (2021) used in this study has recently been published, there is a small number of articles in the literature analyzing it with which to perform a homogeneous comparison. Thus, this section presents a comparison and discussion by comparing this study's results with the work of Yang et al. (2021). A comparison between the results obtained in this study and Yang's is presented in Table 12.

In the study of Yang et al. (2021), it was proposed to train eight support vector machine models with time domain, frequency domain, and wavelet domain features derived from the vehicle's vertical acceleration signals. The metrics the authors considered to evaluate the performance of the eight models were accuracy and F1-score. The best model attained reported values of 97.25% for accuracy and 85.25% for F1-score. However, one of the drawbacks of accuracy and F1-score is that it could lead to a biased interpretation of the classifier's performance as they are biased by the majority class and do not provide insight into the performance of the negative class (Chicco and Jurman, 2020). Hence, one of the drawbacks of Yang's work is the need for reported metrics that help to evaluate the classifier's performance holistically. Furthermore, since the dataset used in this study is unbalanced other metrics, such as sensitivity and specificity, can provide a better acuity of the performance of the CNNs for each class.

Based on the reported results shown in Tables 12, it can be appreciated that the SqueezeNet and ResNet18 provide an average validation F1-score which is similar to the F1-score reported in Yang's work. Nevertheless, the average validation accuracy obtained in this work is lower. Furthermore, in the case of the negative class (i.e., the vehicle's vertical acceleration signals associated with uncracked asphalt pavement), it is complicated to compare with the results reported in Yang's study since, as previously stated, metrics such as the NPV or specificity were not reported. Thus, the majority class could have influenced the high accuracy reported by Yang.

In terms of interpretability, this study and Yang's opted for non-linear and black-box classifiers such as CNNs and support vector machines to perform pavement transverse cracking detection based on the vehicle's vertical acceleration (Singh et al., 2019; Rai, 2020). Nonetheless, the CNNs used in this work are less interpretable due to their depth and number of parameters compared to a support vector machine (see Table 4). Otherwise, one of the focuses of this study was to take advantage of the pre-defined structure of pre-trained CNNs, without requiring an exhaustive evaluation to determine the adequate architecture and weights initialization that deep CNNs have or correct feature extraction and feature selection that classical machine learning techniques need to perform (Martínez-Ríos et al., 2022; James et al., 2013).

Nevertheless, to fine-tune the proposed CNNs weights, specific hardware, such as GPUs, is essential to reduce the training time of the type of deep neural networks used in this study; these hardware requirements could not be affordable to all authors. In this case, hardware does not limit classical machine learning techniques since their computational cost is lower than CNNs. On the other hand, if a limited hardware platform is a criterion to select a classification model over another of the test CNNs, the SqueezeNet could be an option due to its

lower number of parameters (see [Table 4](#)) and low training time (see [Table 10](#)), [Iandola et al. \(2016\)](#).

Finally, as presented in [Section 2](#) and [Table 2](#), the wavelet transform or STFT for feature extraction and signal representation of vehicle vibration data to detect and classify road surface anomalies has already been studied. Different wavelet types, such as Morlet wavelets and Daubechies wavelets of order second, third, sixth, or tenth, have been employed. In some cases, the type of wavelet used was not reported. However, little discussion has been performed related to the properties that were taken into account to select a particular type of wavelet to perform the signal classification, which contributes that its selection appearing to be arbitrary, as pointed out by [Lilly and Olhede \(2012\)](#). On the contrary, this study chose the parameters of the GMWs based on the Heisenberg Area for the gamma parameter of GMWs and the minimum ARMSE to determine the beta parameter and the number of VPO of the CWT.

Compared to the work of [Baldini et al. \(2020\)](#) in terms of methodology, the parameters of the STFT were determined based on the CNN performance when training it with the spectrograms of the vehicle vibration data produced by varying the window size, window overlap, and window type. Baldini's methodology could be time-consuming and require a high computational cost since the proposed CNN must be trained multiple times for each parameterization of the STFT. [Wu et al. \(2020\)](#) also employed a trial-error approach to select a wavelet for pothole detection based on the classifier performance; however, the properties of the wavelet that were considered to select and test it were not discussed. On the contrary, in this study, the selection of the parameterization of the CWT was determined a priori before conducting the training stage of the proposed CNNs by computing the ARMSE as a reference metric for wavelet selection in combination with the Heisenberg area as explained in [Section 3](#).

## 6. Limitations of the study

One of the study's main limitations is the reduced sample size and unbalanced proportion in the distributions of the evaluated classes. The small sample size could have influenced the slight degree of overfitting presented in some cross-validation folds while fine-tuning the pre-trained CNNs, as shown in [Figs. 17, 18, and 19](#). On the other hand, the unbalanced proportion in the distribution of the classes influenced the minor performance in the sensitivity of each CNNs compared to the specificity as shown in [Tables 9, 10, and 11](#). Furthermore, methods to deal with the unbalanced proportion in the dataset classes could be explored for future work since they were not considered in this study ([Pereira and Saraiva, 2021](#)). Moreover, another limitation of the dataset used in this study was that the vehicle's vertical acceleration signals used to fine-tune the proposed CNNs were exclusively collected from a single vehicle; however, the dataset could have a different distribution if other vehicle types were considered since the frequency response of the vehicle could be influenced due the characteristics of the system such as the vehicle's mass, the suspension structure, the tire's pressure, springs, mechanical linkage, and shock absorbers ([Zhang et al., 2021](#)).

Besides, in this work, the pre-trained CNNs were fine-tuned with a fixed learning rate of 0.0001, a mini-batch size of 15, a max number of epochs of 25, and the optimizer was the stochastic gradient descent with momentum. Nonetheless, a different set of training options could produce an improvement in the reported results. Thus, different sets of learning rates, mini-batch sizes, number of max epochs, and optimizers (i.e., RMSprop or Adam) could be tested similarly to the study of [Chakrapani and Sugumaran \(2023\)](#) that was focused on faults diagnosis of dry friction clutches.

Additionally, despite the relatively high performance that the proposed CNNs achieved in terms of AUC, it is important to highlight certain disadvantages of CNNs. The training time for the CNNs could be time-consuming despite using a GPU, as in the present work. Moreover,

the interpretability of deep CNNs could be a potential drawback of this approach since the mechanisms through which CNN makes a decision could be considered abstract or a black-box model ([Haar et al., 2023](#)). In this case, less complex classifiers such as decision trees, linear discriminant analysis, or logistic regression could be preferred ([James et al., 2013](#)). Otherwise, the computational cost of CNNs could be an obstacle that mitigates their deployment in limited hardware platforms. Nonetheless, CNNs such as SqueezeNets are CNNs that were initially designed to be easily embedded into limited hardware platforms such as field-programmable gate arrays ([Iandola et al., 2016](#)).

In addition, this work was mainly focused on performing a simulation of mother wavelet selection and CNNs training for pavement transverse cracking detection based on the associated vehicle's vertical acceleration signals. However, a physical and hardware implementation of this study's methodology must be performed to validate the approach in real scenarios or with other datasets. The above could consider other types of road surface anomalies, such as manholes or potholes, which are potential road surface anomalies that could mitigate the performance of the CNNs that were fine-tuned in this study. Another limitation of this work is that only pavement transverse cracking was examined. In contrast, other studies had datasets with other road surface anomalies (see [Table 2](#)). Additionally, the vehicles' vertical acceleration data was collected from cracks separated by at least 7 m, as shown in [Table 3](#). The above implies a minimal separation between the road surface anomalies for accurate detection. Besides, the vehicle's vertical acceleration signals associated with the pavement transverse cracks were sampled by considering cracks of width 2 to 13 mm; nevertheless, low severity (i.e., transverse cracks with a width < 2 mm) cracks may not produce a different frequency response in the vehicle compared to having no crack ([Yang and Zhou, 2021](#)).

Furthermore, the selection of the parameters of the GMWs and CWT in this study has certain limitations. The setting of gamma was based on the Heisenberg Area lower-bound as proposed by [Lilly and Olhede \(2012\)](#) and presented in [Fig. 4\(a\)](#), and it is a common reference metric used in the studies that have used this wavelet family ([Martínez-Ríos et al., 2023](#)). Nevertheless, the setting of  $\beta$  according to the minimum ARMSE was limited to values produced by the relation provided in Eq. (10) for integer values of the time-bandwidth product ( $P^2$ ) ranging from 3 to 60 and considering  $\gamma = 3$ . Still,  $P^2$  can assume other types of values aside from integer values, which can increase the resolution of the values that  $\beta$  can take. In this case, integer values were selected to lower the computational cost of computing the Reconstruction MSE for each vehicle's vertical acceleration signal in the dataset. The above brings the opportunity to compute the ARMSE for a better resolution of values of  $\beta$ .

Regarding the values of VPO for computing the CWT, this parameter can only take integer values ranging from 1 to 48 through the MATLAB 2022a implementation. For this study, the number of VPO ranged from 1 to 10. The limit to 10 VPO was set to lower the computational cost of computing the ARMSE and based on the behavior of the ARMSE shown in [Fig. 12](#) where it is appreciated that similar behavior is obtained despite increasing the value of VPO for values ranging 2 and 10, except for the case of one VPO. Finally, it could be possible to test optimization algorithms that try to minimize the ARMSE as defined in Eqs. (15) and (16), which could lead to a better selection of  $\beta$  or VPO without analyzing the behavior of the reconstruction error as presented in this study.

Another potential area of improvement related to vehicle vibration data for road surface anomaly detection is to estimate the degree of distress on the road surface. The above could be performed by generating datasets that associate vehicle vibration data with road surface anomalies and measures of the degree of severity in the road pavement, such as the PCI established by ASTM international ([ASTM, 2023](#)). However, as presented in [Section 2](#), no studies or datasets have tried to estimate the PCI through vehicle vibration signals.

## 7. Conclusions and future work

This study proposed to use the ARMSE in combination with the Heisenberg area as a reference metric to set the parameters of the GMWs and CWT to produce a scalogram of the vehicle's vertical acceleration data to fine-tune pre-trained CNNs for pavement transverse cracking detection. This process led to setting  $\gamma = 3$  based on the Heisenberg area and computing the ARMSE to determine its minimum led to set  $\beta = 1.3333$ , and the number VPO equal to 1. The generated scalogram produced by the established parameterization of the CWT was used to fine-tune pre-trained CNNs such as GoogLeNet, SqueezeNet, and ResNet18. From these three CNNs, SqueezeNet had the best performance in terms of average validation sensitivity with a value of  $84.3290 \pm 5.5704$ . However, the validation specificity of the SqueezeNet was lower than that of GoogLeNet and ResNet18. In terms of average validation AUC, the three networks provided a value greater than 0.9 by training them with the generated scalogram.

Future work can explore methods to balance the classes of the dataset and increase the number of samples through data augmentation techniques to improve the performance of the selected pre-trained CNNs. Besides, the methodology proposed in this study could be tested for different datasets in order to test the effectiveness of the approach for other data distributions or road surface scenarios. Furthermore, a hardware implementation of the methodology could be performed to evaluate the hardware requirements needed to generate a correct physical implementation of the method. Additionally, the use of vehicle vibration data to estimate the PCI could be explored. Moreover, experiments could be performed that test different training options of the selected CNNs, such as the learning rate value and the optimization techniques to train the CNNs. Finally, the selection of the parameters of CWT based on the ARMSE used in this work could be performed with other signal representation techniques to determine their parameters, such as the STFT, the Tunable-Q Wavelet Transform, or discrete wavelet transform for pavement transverse cracking detection based on vehicle vibration data.

## CRediT authorship contribution statement

**Erick Axel Martínez-Ríos:** Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Project administration, Resources, Supervision, Writing – original draft, Writing – review & editing. **Rogelio Bustamante-Bello:** Supervision, Funding acquisition. **Sergio A. Navarro-Tuch:** Writing – review & editing.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Data can be consulted in the open repository of Zhou (2019).

## Acknowledgment

The work of Erick Axel Martínez-Ríos was supported by a scholarship awarded by Tecnológico de Monterrey and Consejo Nacional de Ciencia y Tecnología (CVU: 1010770).

## References

- Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M.A., Al-Amidie, M., Farhan, L., 2021. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* 8 (1), 53. <http://dx.doi.org/10.1186/s40537-021-00444-8>.
- Andrade, C., 2019. Multiple testing and protection against a type 1 (false positive) error using the Bonferroni and Hochberg corrections. *Indian J. Psychol. Med.* 41 (1), 99–100. [http://dx.doi.org/10.4103/ijpsympsymp.499\\_18](http://dx.doi.org/10.4103/ijpsympsymp.499_18).
- Andrades, I.S., Castillo Aguilar, J.J., García, J.M.V., Carrillo, J.A.C., Lozano, M.S., 2020. Low-cost road-surface classification system based on self-organizing maps. *Sensors* 20 (21), <http://dx.doi.org/10.3390/s20216009>, URL: <https://www.mdpi.com/1424-8220/20/21/6009>.
- ASTM, D., 2023. Standard practice for roads and parking lots pavement condition index surveys. <http://dx.doi.org/10.1520/D6433-20>.
- Baldini, G., Giuliani, R., Geib, F., 2020. On the application of time frequency convolutional neural networks to road anomalies' identification with accelerometers and gyroscopes. *Sensors* 20 (22), <http://dx.doi.org/10.3390/s20226425>, URL: <https://www.mdpi.com/1424-8220/20/22/6425>.
- Basavaraju, A., Du, J., Zhou, F., Ji, J., 2020. A machine learning approach to road surface anomaly assessment using smartphone sensors. *IEEE Sens. J.* 20 (5), 2635–2647. <http://dx.doi.org/10.1109/JSEN.2019.2952857>.
- Bentley, P., McDonnell, J., 1994. Wavelet transforms: an introduction. *Electron. Commun. Eng. J.* 6 (4), 175–186. <http://dx.doi.org/10.1049/ecej:19940401>.
- Boashash, B., 2016. Time Frequency Signal Analysis and Processing: A Comprehensive Reference, first ed. Elsevier.
- Brunton, S.L., Kutz, J.N., 2019. Dimensionality reduction and transforms. In: *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge University Press, pp. 1–2.
- Celaya-Padilla, J.M., Galván-Tejada, C.E., López-Monteagudo, F.E., Alonso-González, O., Moreno-Báez, A., Martínez-Torteya, A., Galván-Tejada, J.I., Arceo-Olague, J.G., Luna-García, H., Gamboa-Rosas, H., 2018. Speed bump detection using accelerometric features: A genetic algorithm approach. *Sensors* 18 (2), <http://dx.doi.org/10.3390/s18020443>, URL: <https://www.mdpi.com/1424-8220/18/2/443>.
- Chakrapani, G., Sugumaran, V., 2023. Transfer learning based fault diagnosis of automobile dry clutch system. *Eng. Appl. Artif. Intell.* 117, 105522. <http://dx.doi.org/10.1016/j.engappai.2022.105522>, URL: <https://www.sciencedirect.com/science/article/pii/S0952197622005127>.
- Chicco, D., Jurman, G., 2020. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genom.* 21 (1), 6. <http://dx.doi.org/10.1186/s12864-019-6413-7>.
- Daubechies, I., 1992. Ten Lectures on Wavelets. Society for Industrial and Applied Mathematics, <http://dx.doi.org/10.1137/1.9781611970104>, URL: <https://pubs.siam.org/doi/abs/10.1137/1.9781611970104>, arXiv:<https://pubs.siam.org/doi/pdf/10.1137/1.9781611970104>.
- Donoho, D., 1995. De-noising by soft-thresholding. *IEEE Trans. Inform. Theory* 41 (3), 613–627. <http://dx.doi.org/10.1109/18.382009>.
- Durak, L., Arikan, O., 2003. Short-time Fourier transform: two fundamental properties and an optimal implementation. *IEEE Trans. Signal Process.* 51 (5), 1231–1242. <http://dx.doi.org/10.1109/TSP.2003.810293>.
- Feng, C., Wang, H., Lu, N., Tu, X.M., 2013. Log transformation: application and interpretation in biomedical research. *Stat. Med.* 32 (2), 230–239. <http://dx.doi.org/10.1002/sim.5486>, URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/sim.5486>, arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1002/sim.5486>.
- Ferjani, I., Ali Alsaif, S., 2022. How to get best predictions for road monitoring using machine learning techniques. *PeerJ Comput. Sci.* 8, e941. <http://dx.doi.org/10.7717/peerj.cs.941>.
- Fugal, D.L., 2009. *Conceptual Wavelets in Digital Signal Processing: An in-Depth, Practical Approach for the Non-Mathematician*. Space & Signals Technical Pub..
- Ghasemi, A., Zahediasl, S., 2012. Normality tests for statistical analysis: A guide for non-statisticians. *Int. J. Endocrinol. Metab.* 10 (2), 486–489. <http://dx.doi.org/10.5812/ijem.3505>.
- Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroury, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., Chen, T., 2018. Recent advances in convolutional neural networks. *Pattern Recognit.* 77, 354–377. <http://dx.doi.org/10.1016/j.patcog.2017.10.013>, URL: <https://www.sciencedirect.com/science/article/pii/S0031320317304120>.
- Guo, T., Zhang, T., Lin, E., López-Benítez, M., Ma, F., Yu, L., 2022. A review of wavelet analysis and its applications: Challenges and opportunities. *IEEE Access* 10, 58869–58903. <http://dx.doi.org/10.1109/ACCESS.2022.3179517>.
- Haar, L.V., Elvira, T., Ochoa, O., 2023. An analysis of explainability methods for convolutional neural networks. *Eng. Appl. Artif. Intell.* 117, 105606. <http://dx.doi.org/10.1016/j.engappai.2022.105606>, URL: <https://www.sciencedirect.com/science/article/pii/S0952197622005966>.
- Hammouri, H.M., Sabo, R.T., Alsaadawi, R., Kheirallah, K.A., 2020. Handling skewed data: A comparison of two popular methods. *Appl. Sci.* 10 (18), <http://dx.doi.org/10.3390/app10186247>, URL: <https://www.mdpi.com/2076-3417/10/18/6247>.
- He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 770–778. <http://dx.doi.org/10.1109/CVPR.2016.90>.
- Himeur, Y., Al-Maadeed, S., Kheddar, H., Al-Maadeed, N., Abualsaud, K., Mohamed, A., Khattab, T., 2023. Video surveillance using deep transfer learning and deep domain adaptation: Towards better generalization. *Eng. Appl. Artif. Intell.* 119, 105698. <http://dx.doi.org/10.1016/j.engappai.2022.105698>, URL: <https://www.sciencedirect.com/science/article/pii/S0952197622006881>.
- Hlawatsch, F., Auger, F., 2013. Time-Frequency Analysis. In: ISTE, Wiley, URL: <https://books.google.com.mx/books?id=tOeeJyP951QC>.

- Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K., 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size. <http://dx.doi.org/10.48550/ARXIV.1602.07360>, URL: <https://arxiv.org/abs/1602.07360>.
- Iwana, S., 2021. An empirical survey of data augmentation for time series classification with neural networks. *PLOS ONE* 16 (7), 1–32. <http://dx.doi.org/10.1371/journal.pone.0254841>.
- James, G., Witten, D., Hastie, T., Tibshirani, R., 2013. *An Introduction to Statistical Learning*, Vol. 112. Springer.
- Keene, O.N., 1995. The log transformation is special. *Stat. Med.* 14 (8), 811–819. <http://dx.doi.org/10.1002/sim.4780140810>.
- Kim, Y.-M., Kim, Y.-G., Son, S.-Y., Lim, S.-Y., Choi, B.-Y., Choi, D.-H., 2022. Review of recent automated pothole-detection methods. *Appl. Sci.* 12 (11), <http://dx.doi.org/10.3390/app12115320>, URL: <https://www.mdpi.com/2076-3417/12/11/5320>.
- Lekshminipathy, J., Velayudhan, S., Mathew, S., 2021. Effect of combining algorithms in smartphone based pothole detection. *Int. J. Pavement Res. Technol.* 14 (1), 63–72. <http://dx.doi.org/10.1007/s42947-020-0033-0>.
- Li, W., Huang, R., Li, J., Liao, Y., Chen, Z., He, G., Yan, R., Grylliias, K., 2022a. A perspective survey on deep transfer learning for fault diagnosis in industrial scenarios: Theories, applications and challenges. *Mech. Syst. Signal Process.* 167, 108487. <http://dx.doi.org/10.1016/j.ymssp.2021.108487>, URL: <https://www.sciencedirect.com/science/article/pii/S088832702100830X>.
- Li, X., Huo, D., Goldberg, D.W., Chu, T., Yin, Z., Hammond, T., 2019. Embracing crowdsensing: An enhanced mobile sensing solution for road anomaly detection. *ISPRS Int. J. Geo-Inf.* 8 (9), <http://dx.doi.org/10.3390/ijgi8090412>, URL: <https://www.mdpi.com/2220-9964/8/9/412>.
- Li, Z., Liu, F., Yang, W., Peng, S., Zhou, J., 2022b. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Trans. Neural Netw. Learn. Syst.* 33 (12), 6999–7019. <http://dx.doi.org/10.1109/TNNLS.2021.3084827>.
- Liang, P., Wang, W., Yuan, X., Liu, S., Zhang, L., Cheng, Y., 2022. Intelligent fault diagnosis of rolling bearing based on wavelet transform and improved ResNet under noisy labels and environment. *Eng. Appl. Artif. Intell.* 115, 105269. <http://dx.doi.org/10.1016/j.engappai.2022.105269>, URL: <https://www.sciencedirect.com/science/article/pii/S0952197622003281>.
- Lilly, J.M., Olhede, S.C., 2009. Higher-order properties of analytic wavelets. *IEEE Trans. Signal Process.* 57 (1), 146–160. <http://dx.doi.org/10.1109/TSP.2008.2007607>.
- Lilly, J.M., Olhede, S.C., 2010. On the analytic wavelet transform. *IEEE Trans. Inform. Theory* 56 (8), 4135–4156. <http://dx.doi.org/10.1109/TIT.2010.2050935>.
- Lilly, J.M., Olhede, S.C., 2012. Generalized Morse Wavelets as a superfamily of analytic wavelets. *IEEE Trans. Signal Process.* 60 (11), 6036–6041. <http://dx.doi.org/10.1109/TSP.2012.2210890>.
- Lim, T.-S., Loh, W.-Y., 1996. A comparison of tests of equality of variances. *Comput. Statist. Data Anal.* 22 (3), 287–301. [http://dx.doi.org/10.1016/0167-9473\(95\)00054-2](http://dx.doi.org/10.1016/0167-9473(95)00054-2), URL: <https://www.sciencedirect.com/science/article/pii/0167947395000542>.
- Liu, J., Li, Q., Han, Y., Zhang, G., Meng, X., Yu, J., Chen, W., 2019. PEMFC residual life prediction using sparse autoencoder-based deep neural network. *IEEE Trans. Transp. Electrif.* 5 (4), 1279–1293. <http://dx.doi.org/10.1109/TTE.2019.2946065>.
- Lopez Pinaya, W.H., Vieira, S., Garcia-Dias, R., Mechelli, A., 2020. Chapter 10 - Convolutional neural networks. In: Mechelli, A., Vieira, S. (Eds.), *Machine Learning*. Academic Press, pp. 173–191. <http://dx.doi.org/10.1016/B978-0-12-815739-8.00010-9>, URL: <https://www.sciencedirect.com/science/article/pii/B9780128157398000109>.
- Luque, A., Carrasco, A., Martín, A., de las Heras, A., 2019. The impact of class imbalance in classification performance metrics based on the binary confusion matrix. *Pattern Recognit.* 91, 216–231. <http://dx.doi.org/10.1016/j.patcog.2019.02.023>, URL: <https://www.sciencedirect.com/science/article/pii/S00313203193000950>.
- Mallat, S., 2008. *A Wavelet Tour of Signal Processing: The Sparse Way*. Elsevier Science, URL: <https://books.google.com.mx/books?id=5qzeLJluLoC>.
- Mandrekar, J.N., 2010. Receiver operating characteristic curve in diagnostic test assessment. *J. Thorac. Oncol.* 5 (9), 1315–1316. <http://dx.doi.org/10.1097/JTO.0b013e3181ec173d>, URL: <https://www.sciencedirect.com/science/article/pii/S1556086415306043>.
- Marcot, B.G., Hanea, A.M., 2021. What is an optimal value of k in k-fold cross-validation in discrete Bayesian network analysis? *Comput. Statist.* 36 (3), 2009–2031. <http://dx.doi.org/10.1007/s00180-020-00999-9>.
- Martinelli, A., Meocci, M., Dolfi, M., Branzi, V., Morosi, S., Argenti, F., Berzi, L., Consumi, T., 2022. Road surface anomaly assessment using low-cost accelerometers: A machine learning approach. *Sensors* 22 (10), <http://dx.doi.org/10.3390/s22103788>, URL: <https://www.mdpi.com/1424-8220/22/10/3788>.
- Martínez-Ríos, E.A., Bustamante-Bello, M.R., Arce-Sáenz, L.A., 2022. A review of road surface anomaly detection and classification systems based on vibration-based techniques. *Appl. Sci.* 12 (19), <http://dx.doi.org/10.3390/app12199413>, URL: <https://www.mdpi.com/2076-3417/12/19/9413>.
- Martínez-Ríos, E.A., Bustamante-Bello, R., Navarro-Tuch, S., Perez-Meana, H., 2023. Applications of the Generalized Morse Wavelets: A review. *IEEE Access* 11, 667–688. <http://dx.doi.org/10.1109/ACCESS.2022.3232729>.
- Melo, F., 2013. Area under the ROC curve. In: Dubitzky, W., Wolkenhauer, O., Cho, K.-H., Yokota, H. (Eds.), *Encyclopedia of Systems Biology*. Springer New York, New York, NY, pp. 38–39. [http://dx.doi.org/10.1007/978-1-4419-9863-7\\_209](http://dx.doi.org/10.1007/978-1-4419-9863-7_209).
- Moca, V.V., Bárzan, H., Nagy-Dábácan, A., Mureşan, R.C., 2021. Time-frequency super-resolution with superlets. *Nature Commun.* 12 (1), 337. <http://dx.doi.org/10.1038/s41467-020-20539-9>.
- Morid, M.A., Borjali, A., Del Fiol, G., 2021. A scoping review of transfer learning research on medical image analysis using ImageNet. *Comput. Biol. Med.* 128, 104115. <http://dx.doi.org/10.1016/j.compbiomed.2020.104115>, URL: <https://www.sciencedirect.com/science/article/pii/S0010482520304467>.
- Morse, P.M., 1929. Diatomic molecules according to the wave mechanics. II. Vibrational levels. *Phys. Rev.* 34, 57–64. <http://dx.doi.org/10.1103/PhysRev.34.57>, URL: <https://link.aps.org/doi/10.1103/PhysRev.34.57>.
- Olhede, S., Walden, A., 2002. Generalized Morse Wavelets. *IEEE Trans. Signal Process.* 50 (11), 2661–2670. <http://dx.doi.org/10.1109/TSP.2002.804066>.
- Pan, S.J., Yang, Q., 2010. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* 22 (10), 1345–1359. <http://dx.doi.org/10.1109/TKDE.2009.191>.
- Pan, Q., Zhang, L., Dai, G., Zhang, H., 1999. Two denoising methods by wavelet transform. *IEEE Trans. Signal Process.* 47 (12), 3401–3406. <http://dx.doi.org/10.1109/78.806084>.
- Pannu, H.S., Ahuja, S., Dang, N., Soni, S., Malhi, A.K., 2020. Deep learning based image classification for intestinal hemorrhage. *Multimedia Tools Appl.* 79 (29), 21941–21966. <http://dx.doi.org/10.1007/s11042-020-08905-7>.
- Pasqualotto, D., Zigliotto, M., 2022. Increasing feasibility of neural network-based early fault detection in induction motor drives. *IEEE J. Emerg. Sel. Top. Power Electron.* 10 (2), 2042–2051. <http://dx.doi.org/10.1109/JESTPE.2021.3115170>.
- Pereira, J., Saraiva, F., 2021. Convolutional neural network applied to detect electricity theft: A comparative study on unbalanced data handling techniques. *Int. J. Electr. Power Energy Syst.* 131, 107085. <http://dx.doi.org/10.1016/j.ijepes.2021.107085>, URL: <https://www.sciencedirect.com/science/article/pii/S0142061521003240>.
- Piryonesi, S.M., El-Diraby, T.E., 2020. Data analytics in asset management: Cost-effective prediction of the pavement condition index. *J. Infrastruct. Syst.* 26 (1), 04019036. [http://dx.doi.org/10.1061/\(ASCE\)IS.1943-555X.0000512](http://dx.doi.org/10.1061/(ASCE)IS.1943-555X.0000512).
- Piryonesi, S.M., El-Diraby, T.E., 2021. Examining the relationship between two road performance indicators: Pavement condition index and international roughness index. *Transp. Geotech.* 26, 100441. <http://dx.doi.org/10.1016/j.tregeo.2020.100441>, URL: <https://www.sciencedirect.com/science/article/pii/S2214391220303299>.
- Radünz, A.P., Bayer, F.M., Cintra, R.J., 2022. Low-complexity rounded KLT approximation for image compression. *J. Real-Time Image Process.* 19 (1), 173–183. <http://dx.doi.org/10.1007/s11554-021-01173-0>.
- Rai, A., 2020. Explainable AI: from black box to glass box. *J. Acad. Mark. Sci.* 48 (1), 137–141. <http://dx.doi.org/10.1107/s11747-019-00710-5>.
- Rhif, M., Ben Abbes, A., Farah, I.R., Martínez, B., Sang, Y., 2019. Wavelet transform application for/in non-stationary time-series analysis: A review. *Appl. Sci.* 9 (7), <http://dx.doi.org/10.3390/app9071345>, URL: <https://www.mdpi.com/2076-3417/9/7/1345>.
- Riffenburgh, R.H., Gillen, D.L., 2020. *Statistics in Medicine*. Academic Press.
- Ross, S.M., 2017. *Introductory Statistics*. Academic Press.
- Santos, C.F.G.D., Papa, J.P., 2022. Avoiding overfitting: A survey on regularization methods for convolutional neural networks. *ACM Comput. Surv.* 54 (10s), <http://dx.doi.org/10.1145/3510413>.
- Sattar, S., Li, S., Chapman, M., 2018. Road surface monitoring using smartphone sensors: A review. *Sensors* 18 (11), <http://dx.doi.org/10.3390/s18113845>, URL: <https://www.mdpi.com/1424-8220/18/11/3845>.
- Sattar, S., Li, S., Chapman, M., 2021. Developing a near real-time road surface anomaly detection approach for road surface monitoring. *Measurement* 185, 109990. <http://dx.doi.org/10.1016/j.measurement.2021.109990>, URL: <https://www.sciencedirect.com/science/article/pii/S0263224121009192>.
- Sedgwick, P., 2012. Multiple significance tests: the Bonferroni correction. *BMJ* 344 (jan25 4), e509. <http://dx.doi.org/10.1136/bmj.e509>.
- Shaghilil, N., Khalafallah, A., 2018. Automating highway infrastructure maintenance using unmanned aerial vehicles. In: *Construction Research Congress 2018*. pp. 486–495. <http://dx.doi.org/10.1061/9780784481295.049>, URL: <https://ascelibrary.org/doi/abs/10.1061/9780784481295.049>, arXiv:<https://ascelibrary.org/doi/pdf/10.1061/9780784481295.049>.
- Shi, Q., Zhang, H., 2021. Fault diagnosis of an autonomous vehicle with an improved SVM algorithm subject to unbalanced datasets. *IEEE Trans. Ind. Electron.* 68 (7), 6248–6256. <http://dx.doi.org/10.1109/TIE.2020.2994868>.
- Singh, N., Singh, P., Bhagat, D., 2019. A rule extraction approach from support vector machines for diagnosing hypertension among diabetics. *Expert Syst. Appl.* 130, 188–205. <http://dx.doi.org/10.1016/j.eswa.2019.04.029>, URL: <https://www.sciencedirect.com/science/article/pii/S0957417419302593>.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition. CVPR*, pp. 1–9. <http://dx.doi.org/10.1109/CVPR.2015.7298594>.
- Varona, B., Monteserin, A., Teyseyre, A., 2020. A deep learning approach to automatic road surface monitoring and pothole detection. *Pers. Ubiquitous Comput.* 24 (4), 519–534. <http://dx.doi.org/10.1007/s00779-019-01234-z>.
- Wan, Z., Yang, R., Huang, M., Zeng, N., Liu, X., 2021. A review on transfer learning in EEG signal analysis. *Neurocomputing* 421, 1–14. <http://dx.doi.org/10.1016/j.neucom.2020.09.017>, URL: <https://www.sciencedirect.com/science/article/pii/S0925231220314223>.

- Wang, A., Wang, M., Jiang, K., Cao, M., Iwahori, Y., 2019. A dual neural architecture combined SqueezeNet with OctConv for LiDAR data classification. Sensors 19 (22), <http://dx.doi.org/10.3390/s19224927>, URL: <https://www.mdpi.com/1424-8220/19/22/4927>.
- Woycynski, W.A., 2011. A First Course in Statistics for Signal Analysis. Birkhäuser, Boston. <http://dx.doi.org/10.1007/978-0-8176-8101-2>.
- Wu, C.J., Hamada, M.S., 2011. Experiments: Planning, Analysis, and Optimization. John Wiley & Sons.
- Wu, C., Wang, Z., Hu, S., Lepine, J., Na, X., Ainalis, D., Stettler, M., 2020. An automated machine-learning approach for road pothole detection using smartphone sensor data. Sensors 20 (19). <http://dx.doi.org/10.3390/s20195564>, URL: <https://www.mdpi.com/1424-8220/20/19/5564>.
- Xiao, D., Huang, Y., Qin, C., Liu, Z., Li, Y., Liu, C., 2019. Transfer learning with convolutional neural networks for small sample size problem in machinery fault diagnosis. Proc. Inst. Mech. Eng. C 233 (14), 5131–5143. <http://dx.doi.org/10.1177/0954406219840381>, arXiv:<https://doi.org/10.1177/0954406219840381>.
- Yang, Q., Zhou, S., 2021. Identification of asphalt pavement transverse cracking based on vehicle vibration signal analysis. Road Mater. Pavement Des. 22 (8), 1780–1798. <http://dx.doi.org/10.1080/14680629.2020.1714699>, arXiv:<https://doi.org/10.1080/14680629.2020.1714699>.
- Yang, Q., Zhou, S.-s., Wang, P., Zhang, J., 2021. Application of signal processing and support vector machine to transverse cracking detection in asphalt pavement. J. Cent. South Univ. 28 (8), 2451–2462. <http://dx.doi.org/10.1007/s11771-021-4779-6>.
- Zhang, Q., Hou, J., Duan, Z., Jankowski, L., Hu, X., 2021. Road roughness estimation based on the vehicle frequency response function. Actuators 10 (5), <http://dx.doi.org/10.3390/act10050089>, URL: <https://www.mdpi.com/2076-0825/10/5/89>.
- Zhou, S., 2019. Vehicle vibration data. <http://dx.doi.org/10.17632/3DVPJY4M22.1>, URL: <https://data.mendeley.com/datasets/3dvpjy4m22/1>.
- Zhou, Y., Ji, A., Zhang, L., Xue, X., 2023. Sampling-attention deep learning network with transfer learning for large-scale urban point cloud semantic segmentation. Eng. Appl. Artif. Intell. 117, 105554. <http://dx.doi.org/10.1016/j.engappai.2022.105554>, URL: <https://www.sciencedirect.com/science/article/pii/S0952197622005449>.
- Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., He, Q., 2021. A comprehensive survey on transfer learning. Proc. IEEE 109 (1), 43–76. <http://dx.doi.org/10.1109/JPROC.2020.3004555>.