IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 2, 2022, accepted June 10, 2022, date of publication June 14, 2022, date of current version June 17, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3183116

An Embedded System for Acoustic Data Processing and AI-Based Real-Time Classification for Road Surface Analysis

ALESSIO GAGLIARDI, V. STADERINI, AND S. SAPONARA, (Senior Member, IEEE)
Department of Information Engineering, University of Pisa, 56126 Pisa, Italy

Corresponding author: Alessio Gagliardi (alessio.gagliardi@phd.unipi.it)

This work was supported in part by MIUR through the Dipartimento di Eccellenza Crosslab Project, and in part by POS FESR Toscana 2014–2020 SURFace Project at the University of Pisa.

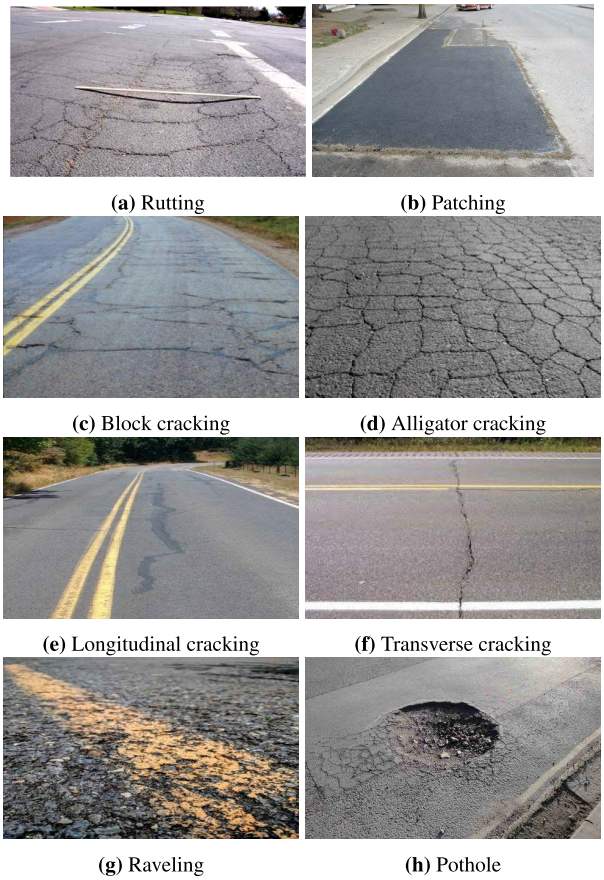
ABSTRACT The current roadway monitoring is expensive and not systematic. This paper proposes a new system able to evaluate the pavement quality of road infrastructure. The embedded system records and processes the acoustic data of the wheel-road interaction and classifies in real-time roadways' health thanks to integrated AI solutions. The measurements to produce the dataset to train a convolutional neural network (CNN) were collected using a vehicle operating at different cruise speeds in the area of Pisa. The dataset is composed by acoustic data belonging to several typologies of roads: dirty or grass roads, high roughness surfaces and roads with cracks or potholes. The raw audio signals were split, labelled, and converted into images by calculating the Mel spectrogram. Finally, the authors designed a tiny CNN with a size equal to 18 kB able to classify between four different classes: good quality road, ruined road, silence and unknown. The CNN architecture achieves an accuracy of about 93% on the original model and 90% on the quantized one. Quantization permits to convert the final architecture into a suitable form to be deployed on a low-complex embedded system integrated in the tyre cavity. In addition, a custom board was designed to act as IoT node thanks to a Bluetooth Low Energy communication towards smartphones and/or car infotainment systems. These systems, featured with GPS, guarantee to obtain real-time maps service of road quality. At authors' knowledge, this is the first real-time and fully integrated solution at the state of the art for road pavement quality analysis and classification on acoustic data.

INDEX TERMS Road surface classification, convolutional neural network, audio processing, embedded system, image recognition.

I. INTRODUCTION

Road surface is an essential component of roadways. The main requirements a roadway should meet are evenness, tyre road friction, carrying capacity and low noise level. However, this infrastructure is subject to permanent stress and needs to be repaired or renewed in order to ensure the substance and utility value. Maintaining a good road surface quality is a major challenge for governments around the world [1]. In fact, ruined surfaces are responsible for car accidents, poor driving quality as well as environmental noise. In addition, traffic noise and noise caused by motor vehicles are even considered as a serious health problem today [2], [3]. Nowadays, most steps of the evaluation are done manually by an inspector who drives along the road, collects raw data, identifies the type of defects and their location, and calculates a specific index for road surface condition (International Roughness Index - IRI) [4]. Since the current procedure is a subjective and labour intensive process, it is an ideal candidate for automation. The rapid growth of vehicles and traffic accidents caused by road pavement anomalies highlights the necessity to invest in finding new systems to evaluate road health. The profile of a surface can be described by the texture wavelength λ that illustrates the different lengths of periodical structures in the profile. Surface texture is the nature of a surface in terms of lay, roughness and waviness where: lay is the direction of the predominant surface pattern determined by the production method used; surface roughness measures

The associate editor coordinating the review of this manuscript and approving it for publication was Jason Gu.



(a) Rutting **(b)** Patching

(c) Block cracking **(d)** Alligator cracking

(e) Longitudinal cracking **(f)** Transverse cracking

(g) Raveling **(h)** Pothole

FIGURE 1. Road surface defects.

the total spaced surface irregularities; waviness is a measure of surface irregularities with a spacing greater than the one of surface roughness. It is said *microtexture* when λ is less than 0.5 mm, *macrotexture* when it is between 0.5 and 50 mm, *megatexture* if the wavelength is from 50 mm to 0.5 m and *unevenness* if it is between 0.5 and 50 m [5].

According to the Texas Department of Transportation (TxDOT) Pavement Management Information System Rater's Manual [6], pavement distresses for asphalt sections are mainly categorized into eight types as shown in Figure 1: rutting, patching, block, alligator, longitudinal and transverse cracking, ravelling and potholes. The central importance of a periodical road surface monitoring and maintenance is due to the fact that it can increase the life span of roads from 15 up to more than 30 years [7].

As detailed in Section II-A, the state of the art does not offer a low-cost and integrated device that performs real-time classification of road surface anomalies. To overcome this issue, in this paper we propose a novel system for road quality classification by implementing a real-time Artificial Intelligence algorithm on a low-complex custom embedded system. Hereafter, the paper is organized as follows: Section II deals with state of the art and innovative contributions of this work. Section III presents data collection and processing, and discusses the neural network architecture used for road quality classification. Section III also contains a description of the firmware and hardware purposely designed for this project. Section IV shows the experiment results, the implementation on the final custom board and discusses about the future works. Conclusions are reported in Section V.

TABLE 1. Comparison of different sensors.

Sensor	Defect	Cost
Profilometer	All defects	High
Camera	Surface defects	Medium
Inertial	Elevation defects	Low
Acoustic	Surface type	Low

II. STATE OF THE ART AND MISSION OF THE PAPER

A. RELATED WORKS

In the last years, researchers have presented several methods to address the problem of road anomalies detection. The number of devices designed for this task is increasing and mainly includes road profilometers, cameras and inertial sensors. There are also few works based on acoustic sensors that represent a new research field. Table 1 shows a brief comparison of different sensors used in the literature for road quality detection. Each system can be combined with machine learning or physical model providing different outputs, e.g. evaluation index or defect type.

The technologies focused on automated road surface monitoring can be divided into different groups based on the output [8] as reported below.

- *Presence*: it answers the question whether a defect exists in the given data or not.
- *Detection*: it identifies the exact position of the defect within the street.
- *Measurement*: it provides the spatial measurements of the defect, e.g. width and depth of pothole.

Road profilometers are devices used to calculate parameters, which describe longitudinal and transverse evenness and the cross fall of a road surface. An example of this application is proposed by Sjogren *et al.* in [9] where profilometers are applied to measure rut depth. Unfortunately, these systems are very expensive (over 3000 $) and require to be mounted on special vehicles involving additional costs.

Another method to detect anomalies is based on cameras and image recognition tools. A camera is placed outside a vehicle and captures real-time 2D images. Then, these are elaborated to get information, like the type of defect and its size, for instance. Balcerek *et al.* [10] propose a classifier of road surfaces based on CNN that determines the general condition of surfaces. The development of new systems is going in the direction of replacing traditional cameras with smartphone cameras. This has the main advantages of reducing costs and facilitating the spreading of the technology. Also Rateke *et al.* [11] propose a surface types and quality classifier based on CNN and present their own dataset collected with a low-cost camera. Varadharajan *et al.* [12] and

Tedeschi *et al.* [13] use a mobile Android device to perform real-time detection of anomalies, e.g., potholes and cracks. However, this methodology has some drawbacks, in particular high costs in exchange for high accuracy and influence of bad weather, shadow, and light variations on the results. The cheapest systems for the estimation of road surfaces are based on inertial sensors. These are mainly implemented in case of potholes recognition or severe anomalies. An advantage respect to the use of cameras is that the data obtained with inertial sensors have a smaller size and are easier to be stored. Nevertheless, this technology is not yet commonly applied from governments, and this is due to lack of reliable and adaptable models. In fact, there are many factors contributing to the final output of inertial sensors (sensory, vehicular, driving, and environmental properties) that must be taken into account. This makes their development really challenging. A device based on inertial sensors is Pothole Patrol [14] that was deployed on taxis to detect and report the surface conditions of roads.

Nowadays, many studies are intended to apply smartphone's inertial sensors because it makes the technology easily applicable in a widespread manner and does not require additional costs. An example of it is Nericell [15], a system thought to be used by people in their normal course to monitor roads and traffic conditions. The disadvantage is that it requires a very complicated hardware and software setup with low final performance. Many studies are available about the correlation between road surface quality and tyre-road noise. *Close Proximity* method (CPX) [16] is the name of a methodology based on test-tyre rolling on the road with measuring microphones located close to the tyre surface. Tyre-road noise is mainly due to the combination of *airborne* noise and *structure-borne* noise. The first causes noise at frequencies higher than 1 kHz and is related to the compression of the air trapped within the tread of the rolling tyre. The other is caused by the contact of tyre with surface defects and covers frequencies lower than 1 kHz. A system that uses this method to distinguish between wet and dry road surface is proposed by Alonso *et al.* [17]. Unfortunately, this has some downsides, for example, the influence of environmental noise and the risk of damaging the device for its proximity to the wheel. Considering that the road texture causes a displacement of the tyre carcass and induces a noise field into the tyre torus, a different type of acoustic sensor was developed. In the work presented by Masino *et al.* [18], a microphone is placed inside the tyre cavity, and this guarantees great advantages. In fact, the tyre cavity is a reverberation room and insulates the mic from external disturbs. The system proposed in this paper aims to overcome the limitations of what is described in the state of the art: performing acquisition and classification in real-time using a low-cost integrated device.

B. OUR CONTRIBUTION

The aim of this work is to develop a road classifier system capable of providing a real-time assessment of the infrastructure. The focus of this paper is the development of innovative artificial intelligence techniques able of autonomously learning from the acoustic data acquired through an integrated system. The algorithm is deployed on an electronic board mounted on the rim flange of a vehicle, linked to a microphone that is installed inside the tyre cavity and equipped with components for Bluetooth Low Energy (BLE) data transmission. One of the main differences between the system proposed by Masino *et al.* [19] and the system proposed in this work is that here the classification is executed real-time directly on the microcontroller. Another innovative aspect is the employment of a convolution neural network that receives Mel spectrogram as input and classifies the road health. Based on the input, the classifier distinguishes the roadway surface typology, and the detected label is sent via BLE to external apparatus for further processing. This is a great advantage respect to devices in which raw data are sent to a server. It assures higher speed, lower power consumption and safeguards from data loss. Other benefits of this methodology are related to the fact that the acoustic signals recorded inside the tyre cavity are not influenced by external environmental noise and the measurements can take place in every light condition. Moreover, the purposely designed board is tiny (3 cm $\times$ 4 cm) and low-cost ($\sim$50€ per each board prototype). This makes the system easily and widely applicable. Finally, an important contribution of this work is the creation of a new dataset containing the tyre noise due to the interaction between wheel and road surfaces. The set of audios is mainly acquired with several measurement campaigns and partly generated by data augmentation. The files are related to road surface typologies such as: good quality, roads covered in grass, dirty roads, potholes, and bad quality, e.g., cracks or high roughness roadways. The main contributions of this paper are summarized as following:

- a new low-cost device based on CNN for real-time road defects classification based on acoustic signals;
- a modified condensed microphone sensor placed inside tyre cavity achieving a perfect insulation from external noise and weather condition;
- design of different CNN architectures taking into account the memory footprint and execution constraints required by the embedded system.

III. MEASUREMENT SYSTEM

The required dataset to train the artificial neural network model was created in two phases. Firstly, several recordings were collected. In particular, the road surfaces considered are: good quality, bad quality, such as cracks or high roughness streets, potholes, dirty and covered in grass. Then, these acoustic data were processed with the intention of obtaining a balanced set.

A. DATA COLLECTION

The acquisition of the signals took place in Pisa province. To achieve this, it was employed a proto-board based on the components listed below:

- CUI Devices cma-4544pf-w electret microphone placed inside the tyre cavity that was modified for measuring higher sound pressure values;
- Raspberry Pi 3 model B attached to the tyre rim that samples and locally stores the data.

The front-end involved in this phase is similar to the one used in [20] and will be implemented in the final embedded system. The Raspberry Pi 3 board is equipped with a sound card for single-channel sampling at 44.1 kHz of the audio signal. To make the model more robust, it was decided to collect the audio at different constant cruise speeds: $\sim$ 30, 40, 50 km/h and each was repeated three times to get further information. Speed of 50 km/h was adopted as a reasonable maximum speed to perform the inspection of city streets. The collection of audio signals at several cruise speeds is useful to generalize the ability of the classifier to detect anomalies at speeds in a neighbourhood of 40 km/h. In fact, to keep constant the velocity is not easy when driving on a busy road. During the campaigns, all done in condition of dry road surface, a total of about 60 minutes of recordings were collected. All the equipment needed for the measurements was mounted on a Mercedes-Benz Vito, used as a mobile laboratory. Additional sensors were employed: an encoder to know the exact cruise speed; an auxiliary synchronized GoPro to film the crossed street. This was important to correctly link the recorded audio with the relative label. The labelling stage is crucial and was carefully performed since a wrong coupling between the signal and the related road surface would determine bad performance of the final model.

B. DATA PROCESSING

Once the data were collected, a pre-processing phase was performed: each record was cut to make it 1 second long and downsampled from 44 kHz to 16 kHz, which is the sampling frequency used by the microcontroller responsible for the classification. From the acquired files, a dataset was created by dividing the tracks into folders where each has the name of the category the files belong to. The whole dataset consists of four classes: *silence*, *unknown*, including both dirty and grass road, *good quality* and *ruined* roads composed by bad quality roadways and potholes. For each class, data augmentation was executed to increase the number of available records and make the model more robust. In particular, we applied the time shift by moving each wave track forwards and backwards with different time factor. In this way, it was obtained a balanced dataset: every folder containing the same number of samples and having the same characteristics. These operations were performed with both Python scripts and Audacity. At this point, further audio processing was conducted to not provide raw data to the model. This has the main advantages of reducing the time required to train the model and simplifying the design of the neural network. In Figure 2, it is shown the spectrum of each considered road surface. As we can see, the main components of the signals appear concentrated in a frequency range lower than 1 kHz.

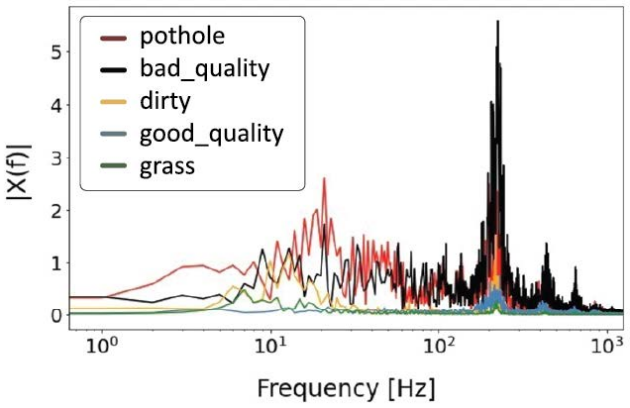


FIGURE 2. FFT spectrum of each considered road surface.

In Figure 2 it is possible to notice a peak at a frequency around 200Hz. This behaviour is consistent with the Equation 1 [21]:

$$f_n = n * \frac{c}{\pi} * \frac{1}{R_{in} + R_{out}} \rightarrow f_1 \approx 215Hz \tag{1}$$

where:

- $n \in \mathbb{N}$ is the mode order;
- c = 343 ms^{-1} is the speed of sound in air at 20°C;
- R_in = 0.191 m is the inner tyre radius;
- R_out = 0.317 m is the outer tyre radius.

In view of this and of the fact that the application must be deployed on a microcontroller, it was decided to transform each acoustic signal into Mel spectrograms. Mel scale is a technique inspired by the way humans perceive frequency in sounds based on pitch. Our hearing works in such a way that we perceive frequencies on a scale that is not linear, paying more attention to low frequencies than to high ones. This means that it is easier to detect differences in low frequencies. This is what Mel scale achieves. It is a logarithmic transformation of the frequency of a signal in which sounds at the same distance on the Mel scale are perceived as being at the same distance by humans. The formula to convert f Hertz into f_{mel} mels is [22]:

$$f_{mel} = 2595 * log_{10}\left(1 + \frac{f}{700}\right) \tag{2}$$

Mel spectrograms are actually preferred for dimension reduction respect different spectrogram representation achieving nearly identical classification accuracy with less model memory required [23]. In consideration of this, one second-length audio was divided into sub-frames that have short interval (30ms) and then windowed by applying the Hann window to avoid discontinuities. The Hann window can be mathematically represented as follows: :

$$w_n = 0.5 * \left(1 - cos\frac{2\pi n}{N-1}\right) \tag{3}$$

where N represents the number of samples in each frame. To compute the Fast Fourier Transform of the frame, it was

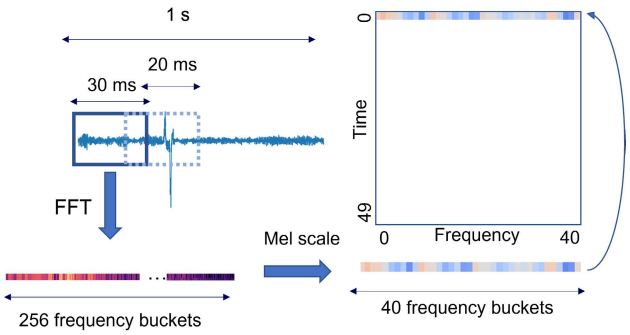


FIGURE 3. Features extraction from audio track.

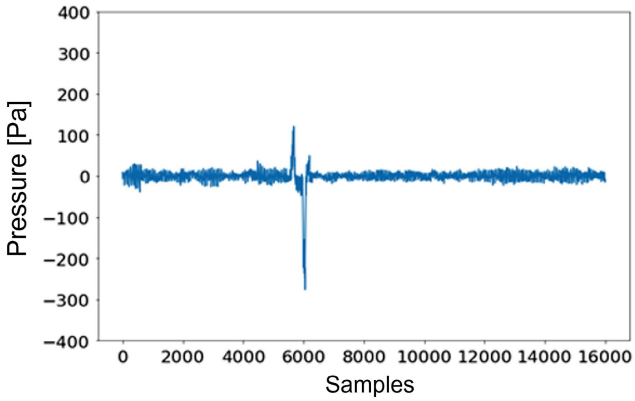


FIGURE 4. Acoustic wave generated from to the interaction between a vehicle's wheel and a pothole.

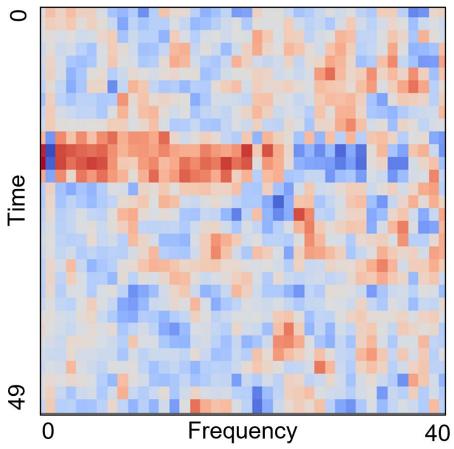


FIGURE 5. Spectrogram of the wave reported in Figure 4. This image evidences the presence of a pothole in correspondence of the 19th row on y-axes.

performed the following operation:

$$X_i(k) = \sum_{n=1}^{N} x_i(n) * w_n(n) * e^{-j2\pi kn/N} \tag{4}$$

where i represents the number of frame and K represents the length of the FFT. Then, the power spectrum was calculated as Equation 5:

$$P_i(k) = \frac{1}{N} |X_i(k)|^2 \tag{5}$$

At this point, it was taken the absolute value of the complex Fourier transform, and the result was squared. This process generates 256 frequency buckets that were averaged by applying Mel frequency scale and obtaining 40 downsampled buckets as in Equation 2. Thanks to these steps, lower frequencies were characterized by a higher resolution and this is coherent with the type of the signals the authors are working with. The process was repeated 49 times striding the window of 20 ms each time as depicted in Figure 3.

The presented steps generate spectrograms with a shape of 49 rows and 40 columns and appear translated respect to traditional spectrograms. This methodology has already been presented in literature [24] and used for other types of audio applications [25], [26]. In Figure 4, it is depicted the acoustic wave corresponding to a pothole as it is easy to notice the pressure peak around the samples 6000. By calculating its spectrogram, as described above, this translates as an increase in energy and it is shown in Figure 5 by the red dark color at row 19.

The extracted spectrograms were treated as images and thus given input to the neural network that is responsible for learning their features. The fact of providing to the CNN a spectrogram instead of raw data has the benefit of reducing the size of the input from 16000 samples to 1960 corresponding to a matrix with 49 rows and 40 columns.

C. NEURAL NETWORKS DESIGN

In a preliminary phase, two convolutional neural networks were designed and compared: the first has a lightweight architecture and is referred to with the term *Tiny*; the other, characterized by an architecture more complex, is inspired by *cnn-trad-fpool3* [27]. In this paper, we will refer to the second architecture with the term *Conv* for the sake of brevity. The Conv model is composed by two convolutional layers with 64 filters each, one maxpooling layer and one fully-connected layer. The presence of the two convolutional and the maxpooling layers assures very good achievements and contributes to the regularization of the model and to greater efficiency in training and evaluation time. Both models accept as input a 49 × 40 image corresponding to the size of the spectrogram discussed above. Unfortunately, this involves that the Conv model has a size of ∼ 30 MB, being infeasible the deployment on limited memory embedded devices. Conversely, the so called Tiny model has only one level of convolution and is less deep than the Conv model. This is why it occupies a memory size of ∼ 18 kB making it a perfect candidate to be used in conjunction with microcontrollers. The reduced size is due to the fact that there is just one convolutional layer and the maxpooling layer is missing. Moreover, the size of the kernel is reduced, and the stride step is increased. Once the CNN has been trained and the results

have been assessed as satisfactory, the classifier is frozen, converted to a quantized vector and saved in a file ready to be integrated with MCU firmware. This last step was possible thanks to the TensorFlow Lite library [28]. Artificial neural network architectures are shown in Figure 6. It is easy to note that the *cnn-trad-fpool3* is slightly bigger and deeper than the so called *tiny* model. Both models have a 1×4 vector as output, which means that they are able to classify between four different membership classes. The number of classes and thus the output vector of the classifiers can be modified during the design phase and especially in relation to the training data available. We chose to use categorical cross-entropy loss function in our study, and it can be computed by using the following formula:

$$Loss = \sum_{i=1}^{N} \left(y_i * log(\hat{y}_i + (1 - y_i) * log(1 - \hat{y}_i) \right) \quad (6)$$

where the variable $\hat{y}$ is the neural network's prediction, and the variable y is the expected output. The variable N represents instead the number of elements in the training and validation set.

D. FIRMWARE

To make the application working on an embedded device, an ad-hoc firmware was developed. The overall idea is to acquire the tyre cavity noise due to the interaction road surface-wheel and to digitize it. Then, to transform the audio samples into a matrix representing a spectrogram. Such spectra will be provided to a pre-trained AI model that will predict the output class. Figure 7 shows how microphone, board and external devices are related.

In particular, the mic is allocated inside the tyre cavity, and it is connected to the ADC of the MCU. To ensure a fast sampling rate, the I2S protocol with dedicated DMA buffers was used. The DMA controller allows for streaming sample data without requiring the CPU to copy each data sample. The first block on the Figure 7 is the *audio provider* that has the task of allowing the communication between the device's mic hardware and the microcontroller. Then, raw audio data are elaborated by the *feature provider* that converts the data into spectrograms. These represent the inputs of the classifier that was already trained on a computer. In fact, it would not be possible to train a NN on a microcontroller. At this point, *TF Lite interpreter* runs the TF Lite model making inference. This process consists in the generation of a set of scores based on the input of the CNN. Each of these values provides information about how likely the analyzed sample belongs to a specific class. Considering the fact that inference is run multiple times per second, the *recognizer commands* aggregates the results and determines, on average, the output of the classification. Averaging the results of multiple inferences is a useful and common technique when dealing with time-series data. In fact, the recogniser calculates the average score for each class over the last three inferences and decides whether it is high enough to count as a

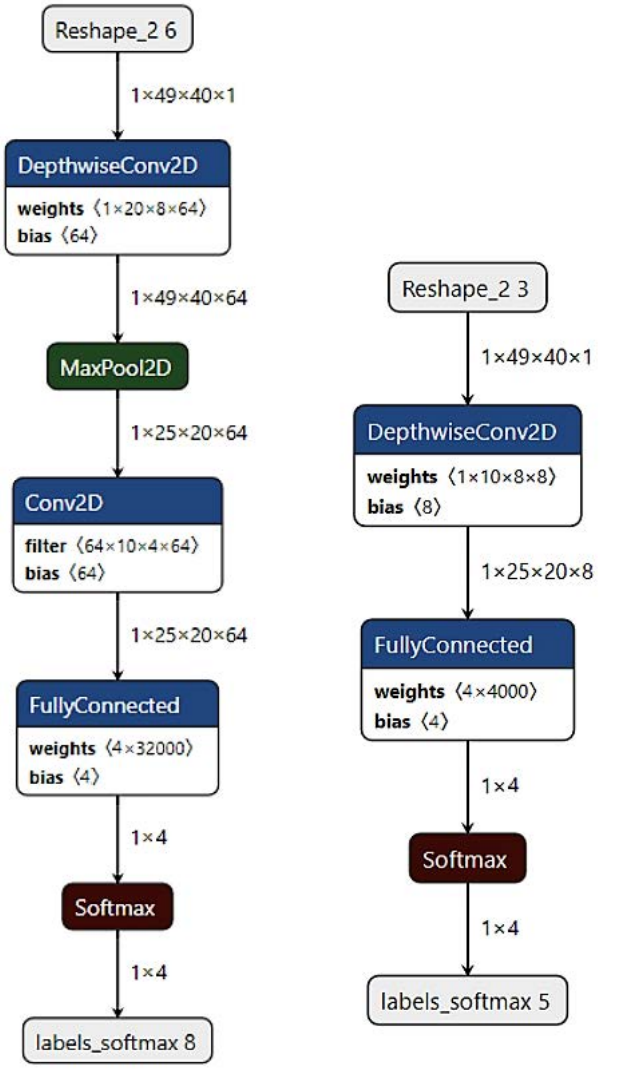


FIGURE 6. Artificial neural network architectures: (left) *cnn-trad-fpool3* model; (right) "Tiny" convolutional model.

detection. Finally, a *command responder* is used to recognize the detected label and send it via Bluetooth Low Energy to an external device. This has many advantages, e.g., it is not required a continuous audio streaming and additional operations can be performed with a machine that is more powerful than a microcontroller. The idea is to develop an app that receives the label, links it with the current position of the vehicle and notifies users of the anomaly colouring a map on a mobile app. Obviously, the data can only be transmitted via BLE after pairing the smartphone with the embedded system (as Figure 16). In an experimental phase, a firmware was tested by providing mocked spectra to the neural network in order to verify its robustness. This made it possible to improve the architecture without requiring to connect the microphone to the circuit board. We underline that the firmware was designed in such a way that it always possible to be modified and improved to detect more classes or to refer to other noise indices.

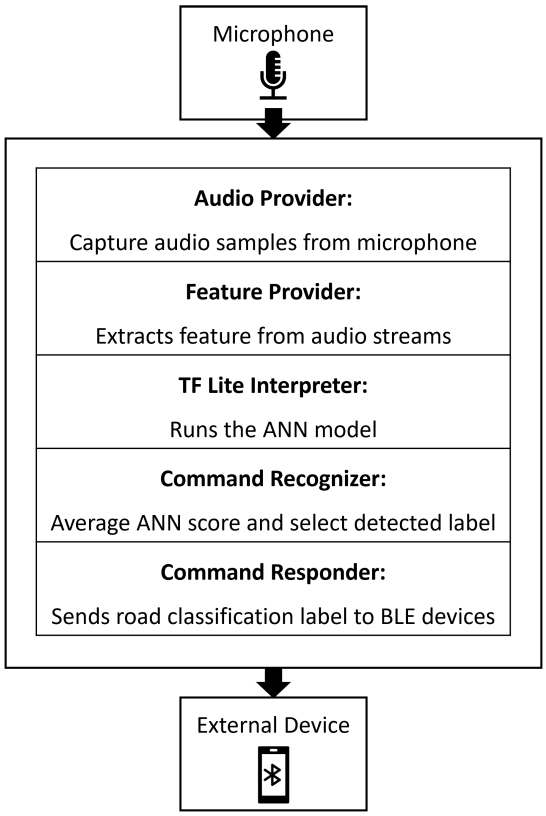


FIGURE 7. Firmware flowchart.

E. HARDWARE

As regards the hardware components, a proper electronic board was developed. A block diagram of the components is depicted in Figure 8. The core of the board is the ESP32-WROOM-32D module [29], selected because it supports TensorFlow Lite and most of the libraries used for data processing. The internal microcontroller is an Xtensa dual-core microprocessors 32-bit LX6 ultra low power at 40 nm technology equipped with 520 kB of SRAM, 4 MB of SPI FLASH memory and onboard antenna. These characteristics make the ESP32 module particularly suitable for IoT and AI applications, such as this. The microphone inside the tyre is connected to the board via an SMA connector.

The acoustic signal from the microphone first passes through an active second-order Sallen-Key bandpass filter with a frequency range of 1 Hz to 16 kHz. Two single-package operational amplifiers are employed for this purpose using a Texas Instruments TL072IDR [30]. Then, the filtered signal is sampled by the 12-bit SAR ADC internal to the ESP32 chip at 16 kHz sampling rate. Three chips are responsible for power management. The first is the MAX77757 from Maxim Integrated [31], which enables battery charging and control. This component is connected to a USB-C port and to a LiPo battery and is configured to deliver a voltage of 4.2-3.5 V. This voltage is converted by the LDO TPS746-Q1 into 3.3 V to supply the ESP32 module and the positive rail of the active filter. Finally, the Linear Technology LT1614 [32] chip is used to supply a voltage of -9 V. In fact, the cma-4544pf-w electret microphone requires a negative voltage to measure high sound pressure of approximately 150 dB with a low total harmonic distortion as discussed in [20]. In addition, there are test points, two button switches, LEDs indicators and header placed on the board. The RESET and BOOT button are used to reboot or enter the ESP32 in write mode respectively. The LEDs indicate correct microcontroller power supply, battery charge status and can be programmed by the user. A serial connector is present on the board and allows the transfer of the firmware to the board via an external FTDI module. All these components are mounted on the two faces of the PCB (as Figure 8). In particular, on the top face of the board prototype, it is placed the ESP32 module with buttons, led, connectors and some ICs; on the bottom face, USB-C connector and the Department of Information Engineering logo (DII) can be observed.

IV. IMPLEMENTATION RESULTS

The system was evaluated in three different phases. Firstly, the two CNNs were trained and compared by observing their results. Then, it was selected the model that achieved the best performance, and it was quantized. Furthermore, it was calculated the model accuracy on both floating point and quantized model as final comparison. The classifier was lastly integrated into the embedded firmware and its functioning was tested by using an Espressif dev board. The application was then deployed on the specifically designed board to test the execution on the real hardware.

A. TESTS

The Tiny and Conv CNNs were trained to classify four different classes: *silence*, *unknown*, *good quality* and *ruined* road, a fourth class that contains both potholes and bad quality roads characterized by cracks and high roughness. The dataset consists of about 240 audio tracks per class for a total of about 1000 observations. As explained above, each audio track must be converted into a spectrogram and thus into a 2D image to be classified. The original dataset was split as follow: 80% train data, 10% validation data and 10% test data. The training of the models was carried out by Python and TensorFlow using Adam as optimiser. For which concern the hyperparameters, we configured a batch size of 64, a learning rate of 0.001, $beta\_1 = 0.9$, $beta\_2 = 0.999$, epsilon=1e-07, with the intention of allowing fast convergence of the model. Figure 10 shows the training process accuracy vs number of iteration of the Tiny model for both training and validation sets. Figure 11 shows instead the values of the loss function again for the Tiny model. The plots of Conv model's accuracy and loss are not presented for the sake of brevity as they are very similar to those already reported. Indeed, the outcomes of the CNNs reach a training accuracy of 99 % and a validation accuracy equal to 92.7 % almost for both models. Additionally, cross entropy loss function was evaluated to have information about the quality of the learning. The final loss score was lower than 0.2 in both models and it is considered to be satisfactory.

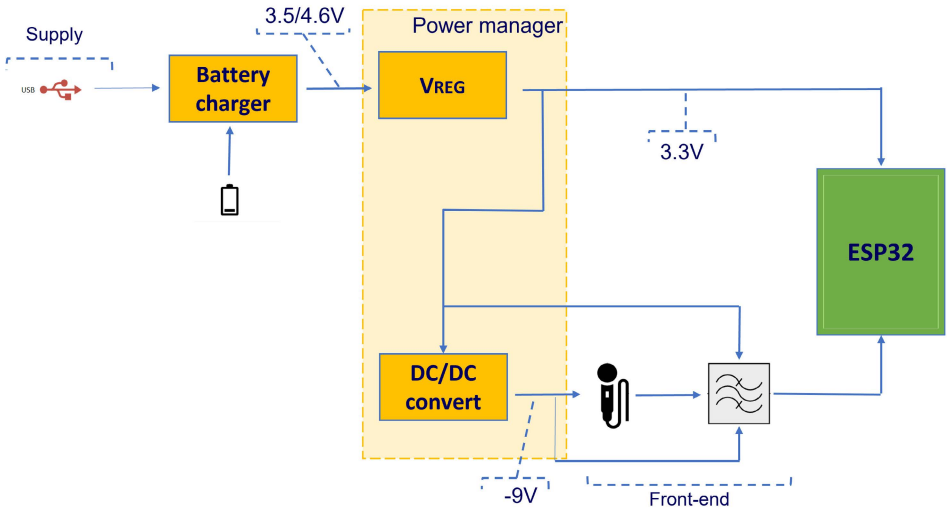


FIGURE 8. Block diagram of custom electronic board.

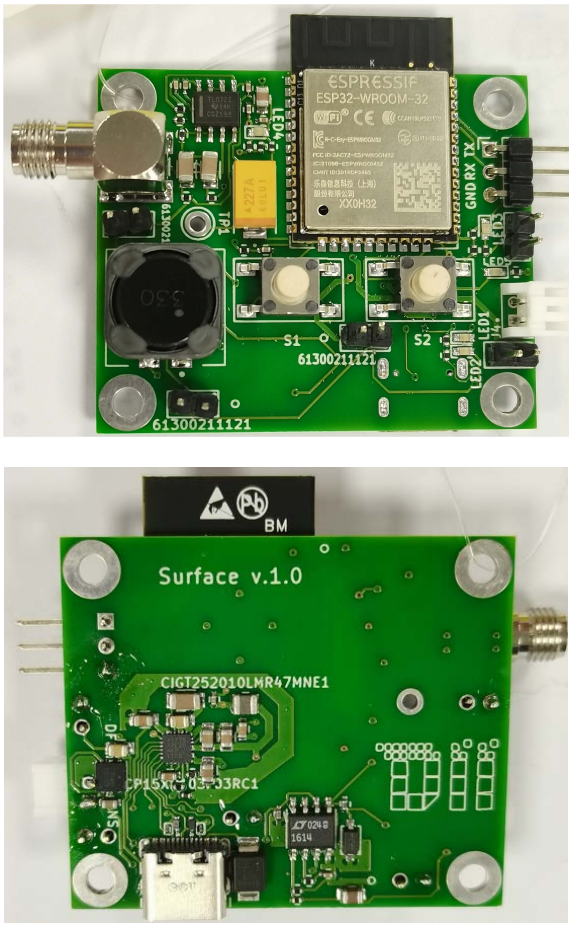


FIGURE 9. Top and bottom view of the purposely designed board.

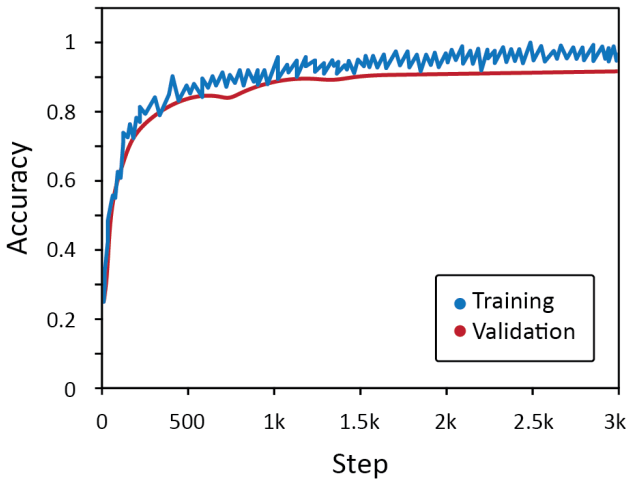


FIGURE 10. Tiny training accuracy: 99% - Tiny validation accuracy: 92.7%.

At the end of the training, the models were evaluated on the test dataset. The confusion matrices for both Conv and Tiny models are shown in Figure 12 and Figure 13 respectively. Confusion matrix is the most common way to evaluate the performance of a prediction or classification model. As expected, Conv model has an overall accuracy of 94% on test set while the Tiny model achieve the 91%.

For multi-class tasks, recall, precision and F1-score are also computed [33] and reported in Table 2. Again, the Conv model generally performs better than the Tiny model. Both models get 100% on all metrics in the detection of the class "silence" while the Conv model is the most proficient in the detection of the class "good_quality". Regarding the average and weighted accuracy each model gets the same value for each class as the test set is balanced.

However, the Tiny model was selected to be deployed on the embedded system because it requires less memory space and achieves reasonable performance. Before to transfer the model to the microcontroller, it was quantized and its confusion matrix was computed once again to guarantee that the quantization did not affect the performance. An accuracy of

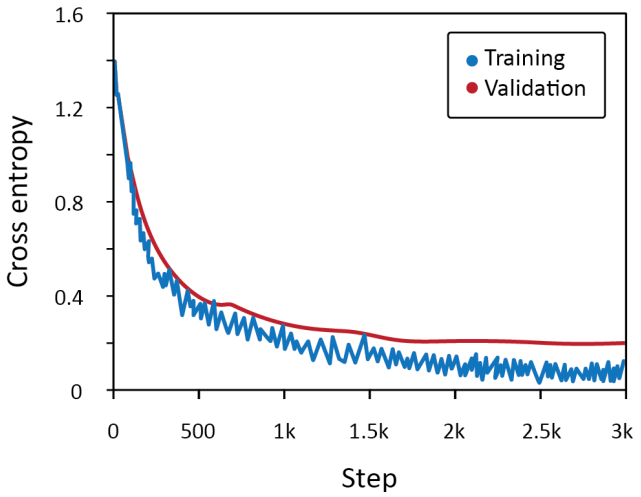


FIGURE 11. Tiny training cross entropy: 0.059 - Tiny validation cross entropy: 0.2.

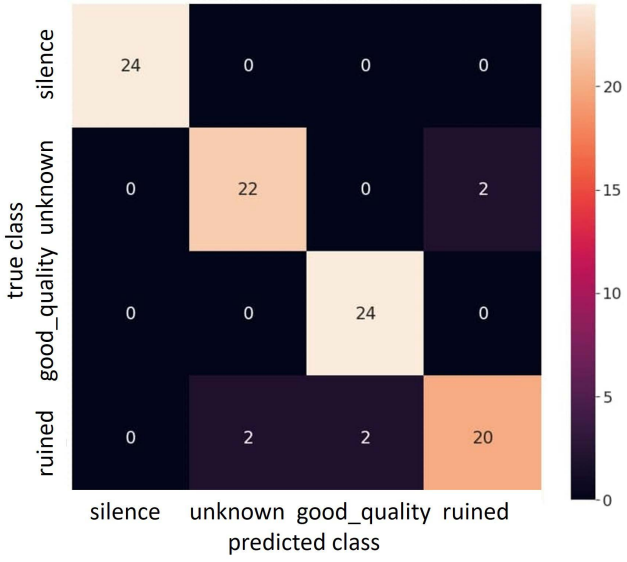


FIGURE 12. Conv CNN confusion matrix on test set: accuracy of 94%.

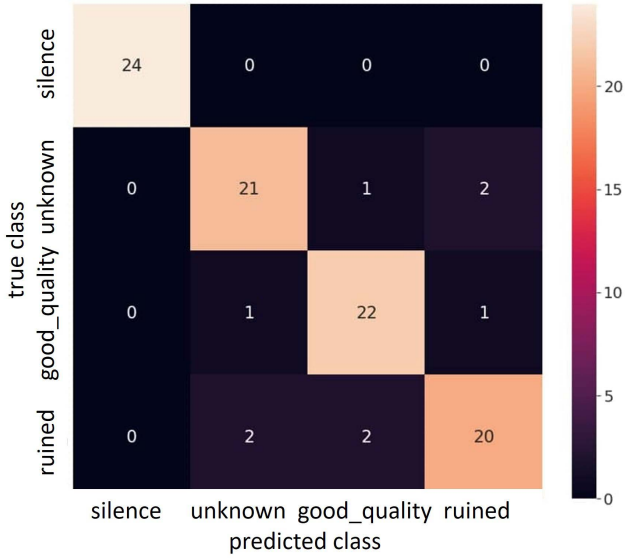


FIGURE 13. Tiny CNN confusion matrix on test set: accuracy of: 91%.

TABLE 2. Performance metrics on test set.

Conv CNN model			
	precision	recall	F1-score
silence	1.000	1.000	1.000
unknown	0.917	0.917	0.917
good_quality	0.923	1.000	0.960
ruined	0.909	0.833	0.870
accuracy			0.938
macro avg	0.937	0.938	0.937
weighted avg	0.937	0.938	0.937
Tiny CNN model			
	precision	recall	F1-score
silence	1.000	1.000	1.000
unknown	0.875	0.875	0.875
good_quality	0.880	0.917	0.898
ruined	0.870	0.833	0.851
accuracy			0.906
macro avg	0.906	0.906	0.906
weighted avg	0.906	0.906	0.906

91% on the test set was obtained on the quantized model, confirming the assumptions discussed above.

B. EXPERIMENTS

An Espressif development board was initially employed to check that the firmware was properly working and transmitting the predicted labels in real-time. Some audio files were extracted from the test set, converted into quantized vectors and used as inputs of the model. By observing the classifier prediction on mocked data input, it was possible to verify the reliability of the final model on the real hardware. As was foreseeable, the board processes the data and simultaneously detects the correct class of the received input vectors. In addition, we checked the functionality of BLE. As expected, the BLE application operates as required: in fact, it waits for pairing request from a client and then, it starts sending the detected labels to the external apparatus. This stage was fundamental to proof that the whole system performs the required tasks and the process can be executed without having issues.

The last phase consists in evaluating the real-time processing by connecting the protoboard to the microphone placed inside the tyre cavity. To this end, we simulated the interaction ruined road - wheel by hitting the tyre with a hammer. The result is consistent with expectations: as the tyre receives the hammer blow, the label predicted is ‘‘ruined road’’; conversely, the label is ‘‘silence’’. In Figure 14, it is shown the signal acquired by the serial port after the hammer hit. The plot reports the number of samples, on the x-axis, and the voltage (in mV), on the y-axis. As we can see, the acquired signal is in the range of 800-2400 mV. We expect that the

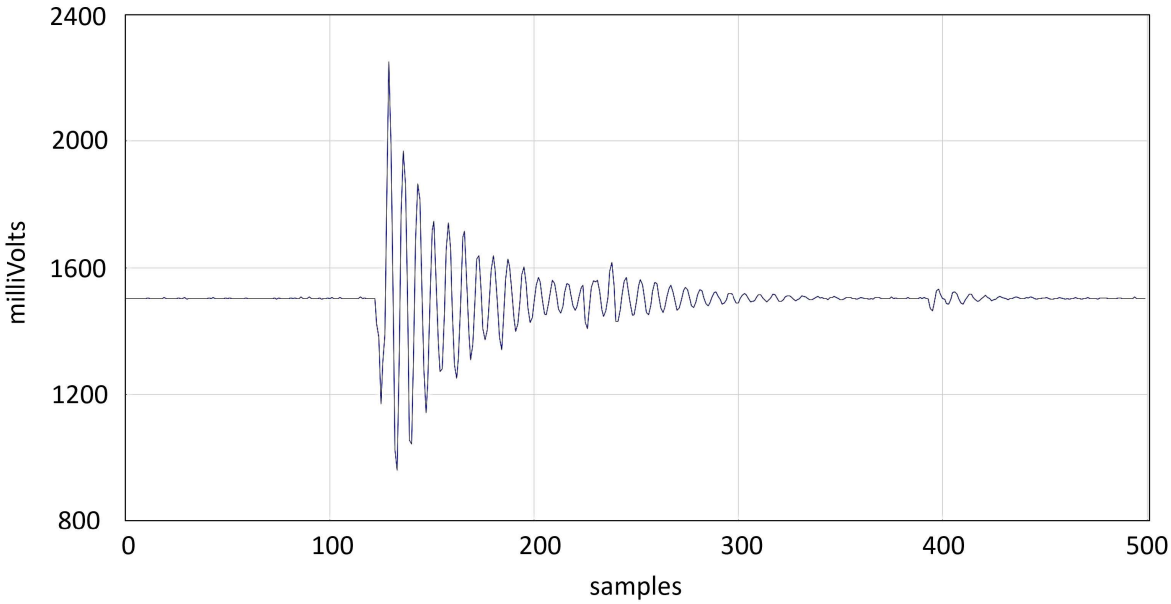


FIGURE 14. Plot of the hammer blow sampled by the ADC of the custom board.

system can measure sound pressure levels of approximately 161 dB with a total harmonic distortion of −63.4 dB as demonstrated in [20]. It is worth remembering that the ADC of the ESP32 module samples in a range of 0-3.3 V and therefore, the acquired signal is well below the saturation threshold.

Then, the wheel was mounted on the vehicle and further preliminary tests were performed. During the motion, the board kept the Bluetooth connection with a smartphone device sending the detected label. In Figure 15, we can observe the system mounted on the vehicle wheel. Figure 16 shows a smartphone screenshot of the label transmitted via BLE (it was chosen to use a free smartphone app called ''nRF Connect for Mobile'' from Nordic). We underline that the proposed system is able to communicate with every BLE equipped devices. So that it would also be possible to transmit information to a computer or to the car infotainment system through easy steps: search the list of BT devices, look for the device with the name ''Surface-ESP32'' and then click on ''Connect''. Within the characteristics of the service defined in the firmware, the current value, in this case ''ruined road'', will be visible.

During the experiments, it was measured the execution time of the application while working in real-time. In particular, we measured the inference time, which represents the time spent by the neural network to calculate the classification value given an input, and the processing time, which represents the time needed from the sampling of the signal to the transmission of the label via BLE. Results are shown in Table 3. As expected, most of the processing time is required by the TF Interpreter to generate the classification result: on average, the inference time is about 208 milliseconds while the total processing time is just over 216 ms. These metrics

FIGURE 15. Device fixed to the rim.

are calculated approximately out of 100 values, where a low variability of values is also observed. In fact, in real time, about 4 inferences are made per second and therefore about 4 packets are transmitted via BLE per second.

Experimental tests were also executed to get information about the power consumption of the board. The system is powered by a 3.7 V LiPO battery with a capacity of 3000 mAh housed underneath the board. It drains an average of 90 mA when in idle state, i.e., without BLE pairing. Conversely, it draws on average about 115 mA during BLE transmission.

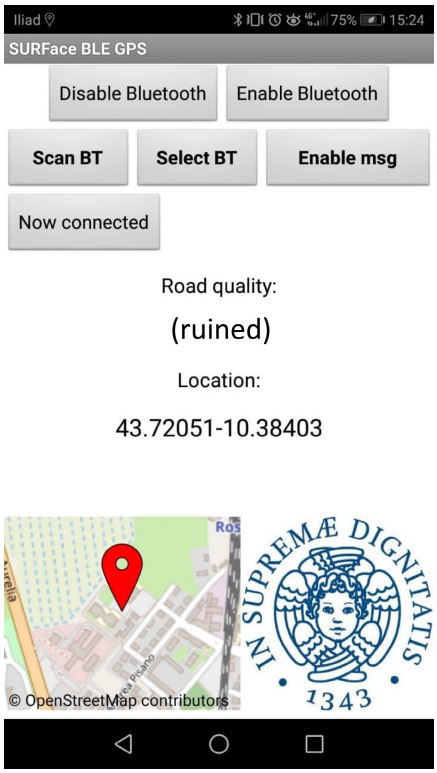


FIGURE 16. Interface of Surf-ace prototype application: it receives the label sent by the microcontroller via BLE and matches it with the smartphone position obtained through its GPS.

TABLE 3. Execution time performance.

	Inference	Processing
Min time (ms)	208.00	215.00
Max time (ms)	209.00	218.00
Mean time (ms)	208.17	216.05
Variance	0.13	0.28
Standard deviation	0.37	0.54
Count	100	100

We can therefore state that the maximum consumption of the device is less than 0.45 W, and that a 3000 mAh battery guarantees packet transmission for more than 26 hours. This is a great result considering that road monitoring sessions usually last only a few hours in a day. Moreover, the device can be easily detached and recharged at the domestic power supply in short time.

C. FUTURE WORKS

The preliminary obtained results are very encouraging. However, new acquisition campaigns are necessary to enrich the dataset and make the convolutional neural network more robust. The new dataset will consist of audio tracks acquired directly from the device developed and presented in this work. As already said, in this study, it was employed a Raspberry Pi 3 for data acquisition and it is worth noting that its analogue front-end is the same as that used in our custom board. A retraining phase will be executed on the new dataset, and it will be also possible to increase the number of classes distinguished by the classifier. The accuracy of the system will be evaluated by collecting the online prediction achieved on the vehicle in motion. Finally, a distinct smartphone application will be developed with the idea of acquiring GPS coordinates as well as road quality. In this way, it will be possible to create a web platform containing the map of roads and their quality.

V. CONCLUSION

In this work, we have presented an application for real-time road surface monitoring based on AI tools. The algorithm is designed to be implemented on a microcontroller board equipped with a microphone that captures sounds inside the cavity of a tyre. Preliminary experimental results show that the device is capable of detecting the quality of the asphalt with an accuracy of 91% on the test set. This demonstrates the suitability of the proposed Tiny architecture for this application and of the Mel inspired spectrogram as input to detect road health. The presented approach takes advantage of innovative techniques. In fact, the deployment of AI system settled on embedded system is a cutting-edge technology, focus of many current research. The fact of having a light, low-power and low-cost device is a great advantage respect to the current state of the art and commercial technologies. Moreover, the methodology adopted in the present paper is not susceptible to light condition or environmental noise. The algorithm has been tested both on Espressif developed board as well as the purposely designed board. The algorithm works as expected and performs real-time processing, classification and BLE communication achieving very promising preliminary results. The performed experiments allow us to have high expectations for future steps. In particular, new acquisition campaigns are planned with the designed device. In this way, it will be possible to improve the robustness of the model and expand the membership categories. The final objective is the development of an app for smartphones and a web platform containing information on roads location and their health available as a service.

ACKNOWLEDGMENT

The authors would like to thank M. Notini for custom hardware design and Ipool srl for data acquisition support.

REFERENCES

[1] R. Frisoni, "EU road surfaces: Economic and safety impact of the lack of regular road maintenance," Tech. Rep., 2014.
[2] G. Belojević, B. Jakovljević, and O. Aleksić, "Subjective reactions to traffic noise with regard to some personality traits," *Environ. Int.*, vol. 23, no. 2, pp. 221–226, 1997.
[3] G. L. Bluhm, N. Berglind, E. Nordling, and M. Rosenlund, "Road traffic noise and hypertension," *Occupational Environ. Med.*, vol. 64, no. 2, pp. 122–126, Oct. 2006.
[4] P. Múčka, "International roughness index specifications around the world," *Road Mater. Pavement Des.*, vol. 18, no. 4, pp. 929–965, Jul. 2017.
[5] L. Fontes, P. Pereira, and J. Pais, "Improvement of the functional pavement quality with asphalt rubber mixtures," Tech. Rep., May 2021.
[6] *Pavement Management Information System Rater's Manual*, Texas Dept. Transp., Austin, TX, USA, 2016.

[7] J. R. M. Acurio, "Incorporating risk and uncertainty into pavement network maintenance and rehabilitation budget allocation decisions," M.S. thesis, Dept. Civil Eng., Texas A&M Univ., College Station, TX, USA, 2014.
[8] S. C. Radopoulou and I. Brilakis, "Automated detection of multiple pavement defects," *J. Comput. Civil Eng.*, vol. 31, no. 2, Mar. 2017, Art. no. 04016057.
[9] L. Sjögren and T. Lundberg, "Design an up to date rut depth monitoring profilometer, requirements and limitations," Statens väg-och Transportforskningsinstitut, Linköping, Sweden, Tech. Rep., 2005.
[10] J. Balcerek, A. Konieczka, K. Piniarski, and P. Pawlowski, "Classification of road surfaces using convolutional neural network," in *Proc. Signal Process., Algorithms, Architectures, Arrangements, Appl. (SPA)*, Sep. 2020, pp. 98–103.
[11] T. Rateke, K. A. Justen, and A. Von Wangenheim, "Road surface classification with images captured from low-cost camera–road traversing knowledge (RTK) dataset," *Revista de Informática Teórica e Aplicada*, vol. 26, no. 3, pp. 50–64, Nov. 2019.
[12] S. Varadharajan, S. Jose, K. Sharma, L. Wander, and C. Mertz, "Vision for road inspection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Mar. 2014, pp. 115–122.
[13] A. Tedeschi and F. Benedetto, "A real-time automatic pavement crack and pothole recognition system for mobile Android-based devices," *Adv. Eng. Inform.*, vol. 32, pp. 11–25, Apr. 2017.
[14] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan, "The pothole patrol: Using a mobile sensor network for road surface monitoring," in *Proc. 6th Int. Conf. Mobile Syst., Appl., Services*, 2008, pp. 29–39.
[15] P. Mohan, V. Padmanabhan, and R. Ramjee, "Nericell: Rich monitoring of road and traffic conditions using mobile smartphones," in *Proc. ACM Sensys*. Raleigh, NC, USA: Association for Computing Machinery, Nov. 2008, pp. 323–336.
[16] G. de León, L. G. D. Pizzo, L. Teti, A. Moro, F. Bianco, L. Fredianelli, and G. Licitra, "Evaluation of tyre/road noise and texture interaction on rubberised and conventional pavements using CPX and profiling measurements," *Road Mater. Pavement Des.*, vol. 21, no. 1, pp. S91–S102, Sep. 2020.
[17] J. Alonso, J. López, I. Pavón, M. Recuero, C. Asensio, G. Arcas, and A. Bravo, "On-board wet road surface identification using tyre/road noise and support vector machines," *Appl. Acoust.*, vol. 76, pp. 407–415, Feb. 2014.
[18] J. Masino, J. Pinay, M. Reischl, and F. Gauterin, "Road surface prediction from acoustical measurements in the tire cavity using support vector machine," *Appl. Acoust.*, vol. 125, pp. 41–48, Oct. 2017.
[19] J. Masino, M. Luh, M. Frey, and F. Gauterin, "Inertial sensor for an autonomous data acquisition of a novel automotive acoustic measurement system," in *Proc. IEEE Int. Symp. Inertial Sensors Syst. (INERTIAL)*, Mar. 2017, pp. 98–101.
[20] J. Masino, B. Daubner, M. Frey, and F. Gauterin, "Development of a tire cavity sound measurement system for the application of field operational tests," in *Proc. Annu. IEEE Syst. Conf. (SysCon)*, Apr. 2016, pp. 1–5.
[21] L. G. Del Pizzo, F. Bianco, A. Moro, G. Schiaffino, and G. Licitra, "Relationship between tyre cavity noise and road surface characteristics on low-noise pavements," *Transp. Res. D, Transp. Environ.*, vol. 98, Sep. 2021, Art. no. 102971.
[22] D. O'Shaughnessy, *Speech Communications: Human and Machine*. Reading, MA, USA: Addison-Wesley, 1987.
[23] K. Choi, G. Fazekas, M. Sandler, and K. Cho, "A comparison of audio signal preprocessing methods for deep neural networks on music tagging," in *Proc. 26th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2018, pp. 1870–1874.
[24] L. Muda, M. Begam, and I. Elamvazuthi, "Voice recognition algorithms using mel frequency cepstral coefficient (MFCC) and dynamic time warping (DTW) techniques," 2010, *arXiv:1003.4083*.
[25] F. Ye and J. Yang, "A deep neural network model for speaker identification," *Appl. Sci.*, vol. 11, no. 8, p. 3603, Apr. 2021.
[26] P. Warden and D. Situnayake, *TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers*. Sebastopol, CA, USA: O'Reilly Media, 2019.
[27] T. N. Sainath and C. Parada, "Convolutional neural networks for small-footprint keyword spotting," in *Proc. 16th Annu. Conf. Int. Speech Commun. Assoc.*, Sep. 2015.
[28] M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning," in *Proc. 12th USENIX Symp. Oper. Syst. Design Implement. (OSDI)*, 2016, pp. 265–283.
[29] *ESP32 WROOM 32D & ESP32-WROOM 32U*, Espressif, Singapore, Version 2.2, 2021.
[30] *TL07xx Low-Noise FET-Input Operational Amplifiers, SLOS080S*, Texas Instrum., Dallas, TX, USA, 2021.
[31] *MAX77757 3.15A USB Type-C Autonomous Charger With JEITA for 1-Cell Liion/LiFePO$_4$ Batteries*, Maxim Integr., San Jose, CA, USA, Revision 2, 2021.
[32] *LT1614 Inverting 600 kHz Switching Regulator*, Linear Technol., Milpitas, CA, USA, 2021.
[33] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, 2009.

ALESSIO GAGLIARDI received the M.Sc. degree in robotics and automation engineering from the University of Pisa, Italy, in 2018. He is currently pursuing the Ph.D. degree with the Department of Information Engineering. His master's thesis concerned the modeling and analysis of the V2x 802.11p protocol by developing simulation software to study the reliability of vehicle communication in different operating scenarios. In 2016, he worked at Texas Instrument as an Application and Support Engineer in Freising, Germany. His research interests include image processing, artificial intelligence, and embedded systems for automotive and industrial applications.

V. STADERINI received the M.S. degree *(cum laude)* in robotics and automation engineering from the University of Pisa, Italy, in 2021. Her master's thesis was focused on the development of AI algorithms to be implemented on embedded systems for the real-time classification of road surface anomalies based on tyre noise. She is currently pursuing the Ph.D. degree with the Automation and Control Centre, Complex Dynamical Systems Group, Austrian Institute of Technology, Vision. Her research aims to merge vision and robotics in path planning and to obtain improvements from the use of reinforcement learning tools.

S. SAPONARA (Senior Member, IEEE) received the master's degree *(cum laude)* and the Ph.D. degree from the University of Pisa. In 2012, he was a Marie Curie Research Fellow at IMEC. He is currently a Full Professor in electronics with the University of Pisa. He is also an IEEE Distinguished Lecturer and the Co-Founder of a special interest group on the IoT of both IEEE CAS and SP societies. He is also the Director of the I-CAS Laboratory, Crosslab Industrial IoT, and Summer School Enabling Technologies for IoT. He is an Associate Editor of several IEEE and Springer journals. He has coauthored more than 300 scientific publications and 18 patents. He is the leader of many funded projects by EU and by companies like Intel, Magneti Marelli, Ericsson, and PPC.