Violence Detection in Audio: Evaluating the Effectiveness of Deep Learning Models and Data Augmentation

Dalila Durães*, Bruno Veloso, Paulo Novais*

ALGORITMI Centre/LASI, University of Minho, Guimarães (Portugal)

Received 4 March 2023 | Accepted 22 August 2023 | Published 29 August 2023

Abstract

Human nature is inherently intertwined with violence, impacting the lives of numerous individuals. Various forms of violence pervade our society, with physical violence being the most prevalent in our daily lives. The study of human actions has gained significant attention in recent years, with audio (captured by microphones) and video (captured by cameras) being the primary means to record instances of violence. While video requires substantial processing capacity and hardware-software performance, audio presents itself as a viable alternative, offering several advantages beyond these technical considerations. Therefore, it is crucial to represent audio data in a manner conducive to accurate classification. In the context of violence in a car, specific datasets dedicated to this domain are not readily available. As a result, we had to create a custom dataset tailored to this particular scenario. The purpose of curating this dataset was to assess whether it could enhance the detection of violence in car-related situations. Due to the imbalanced nature of the dataset, data augmentation techniques were implemented. Existing literature reveals that Deep Learning (DL) algorithms can effectively classify audio, with a commonly used approach involving the conversion of audio into a mel spectrogram image. Based on the results obtained for that dataset, the EfficientNetB1 neural network demonstrated the highest accuracy (95.06%) in detecting violence in audios, closely followed by EfficientNetB0 (94.19%). Conversely, MobileNetV2 proved to be less capable in classifying instances of violence.

Keywords

Audio, Deep Learning, Human Action Recognition, Machine Learning, Transfer Learning, Violence Detection in a car.

DOI: 10.9781/ijimai.2023.08.007

I. Introduction

As stated by Koritsas in 2009 [1], violence manifests in both verbal and physical aspects. Verbal aggression entails employing disrespectful speech, shouting, or shrieking with the purpose of causing offense or inducing fear. Physical aggression involves physically assaulting or trying to assault others, encompassing actions like striking, slapping, kicking, or utilizing a weapon or any object with the intention of inflicting bodily harm. As outlined in a study [2], almost half (48%) of the individuals who fell victim to interpersonal violence in South Korea in 2015 were fatally injured by sharp instruments like knives, whereas such fatalities attributed to sharp objects were approximately 25%. In the year 2020, the Portuguese Association for Victim Support (APAV) reported a total of 66,408 cases, with 31% attributed to "crimes and other forms of violence." Among these cases, 94% involved acts of violence against individuals [3]. The identification and acknowledgment of violence have been focal points of research interest, particularly within surveillance. The primary aim of detecting and recognizing violence revolves around achieving automated and real-time capabilities, enabling timely assistance to victims [4]. It is crucial to identify and prevent such actions before they escalate into catastrophic situations.

Modern society is placing increasing emphasis on automated surveillance as it helps manage an overwhelming amount of data, including attention bias, and ensures the privacy of those being surveyed [5], [6]. Well-designed surveillance software can process multiple sets of sensor data over an extended period of time without risking disengagement. On the other hand, an extra safeguard for data privacy is a properly auditable system that anonymises or deletes data in cases where violence is not detected.

Most violence recognition methods primarily rely on video detection, which necessitates high-performance hardware and software for recording [7]. An alternative technique for violence detection involves using audio, which can be effectively recognized and classified using deep learning algorithms [8]. Audio signals can be effortlessly captured by microphones, which possess strong capabilities to record human behavior and emotions. Therefore, it is crucial to have a robust audio representation that complements and validates the video's audio quality [9], [10].

Another consideration is that violence detection is often associated with crowd violence detection [11]–[14]. However, in recent times, there has been a notable surge in interest surrounding audio-based violence detection, owing to its capacity to identify and prevent violent incidents and also by the increase of car sharing. Particularly,

* Corresponding author.

E-mail addresses: dad@di.uminho.pt (D. Durães), pjon@di.uminho.pt (P. Novais).

researchers have directed their attention to the detection of violence within vehicles using audio-based methods [5], [7], [15].

Despite the promising prospects of audio-based violence detection inside vehicles, its effectiveness relies on various factors, such as the type of microphone employed, background noise levels, and microphone placement within the car [7], [16], [17]. Nonetheless, leveraging audio-based violence detection in vehicles holds potential to enhance the safety of both passengers and drivers.

According to the search results provided, various models have been employed for audio-based violence detection inside vehicles. These models include the ResNet model utilizing the Mel-spectrogram methodology for audio signals [10], [18], CNN-based Audio Event Recognition [15], ensemble deep learning, and multimodal approaches [19], as well as the application of machine learning (ML) models for detecting violence in video streams [20]. The studies indicate that deep learning techniques, such as artificial neural networks and convolutional neural networks, have demonstrated notable enhancements in the accuracy of audio event classification when compared to traditional feature-based classification methods.

Deep learning [21], a novel approach to data modeling that has gained significant traction in recent years, has led to the development of innovative structures and learning algorithms. These advancements have enabled breakthroughs in areas such as recognition [22], object recognition, and machine translation [23], [24]. In the realm of audio-related tasks, deep learning models have played a pivotal role in enhancing accuracy and robustness across diverse categories. As a result, deep learning has become a fundamental area of research in various fields of knowledge [25].

Notwithstanding the extensive research conducted thus far, the realm of identifying violence within the confines of a vehicle remains severely limited in terms of available studies. This scarcity of research is attributed to the distinctive attributes of the car's interior, which pose challenges to the effectiveness of existing models in yielding favorable outcomes [5]. As audio requires minimal storage, our intention is to carry out a study focused on detecting violence within a car using audio.

A. Main Contributions

We utilized a custom dataset designed specifically for detecting violence within a car environment using audio data. It is worth noting that this paper is an extension of the previously published work [26], with the primary focus being on violence detection within a car.

The main objective of this paper is to present the outcomes of our experiments conducted using in-car audio data and deep learning frameworks for the purpose of violence identification. The dataset used for training and validation serves as the foundation for the results presented in this study. Due to the relatively small size of the dataset, data augmentation techniques were applied to augment its volume.

The research questions to be addressed are as follows: RQ1) Can violence inside a car be effectively detected using audio data and deep learning models? RQ2) Can the use of data augmentation enhance the accuracy of violence detection results? To limit the scope of the study, incidents will be classified solely as either violent or non-violent, without considering the specific type of human action or the nature of the violence involved.

B. Organization

The organization of this document is as follows: Section II, Background, discusses the current state of the field, while Section III, Methods, outlines the Mel Spectrogram concepts, public dataset, In car dataset, pre-processing techniques, algorithms, and training procedures employed. Section IV, Results and Discussion, presents the obtained outcomes and corresponding discussions. Lastly, Section V, Conclusion, offers the final conclusions drawn from the study.

II. Background

Different methodologies adopted in some previously conducted studies on the use of audio in violence detection were explored.

A. Models

The detection of violence inside a car using audio-based methods has garnered significant interest as it holds the promise of enhancing road safety by preventing violent incidents and aiding in criminal investigations. Over time, research in this domain has resulted in the advancement of sophisticated algorithms and techniques that significantly improve the accuracy of identifying violent activities within vehicles.

Audio violence detection offers several advantages over video approaches, particularly in terms of bandwidth, storage, and computing requirements, which are significantly lower [9]. While audio sensors have their limitations, they are relatively minor compared to video cameras. For instance, microphones can have an omnidirectional capability, providing a spherical field of view, unlike video cameras with limited angular views. Additionally, audio event acquisitions can outperform video acquisitions due to the longer wavelength of audio, allowing for acoustic wave reflections when encountering obstacles in the direct path. Moreover, audio processing is not affected by issues like lighting and temperature, unlike video processing [9]. The audio approach also captures a wealth of information that visual data alone cannot represent, including screams, explosions, abusive language, and emotional cues conveyed through sound passages. Despite these advantages, there are still limited applications for violence detection using audio-based methods.

Souto, Mello, and Furtado [27] conducted research on domestic violence and acoustic scene classification using machine learning. The parameters employed for feature extraction and processing in both short and medium terms included MFCC (Mel Frequency Cepstral Coefficients), Energy, and ZCR (Zero Crossing Rate). For classification, they utilized the SVM (Support Vector Machine) technique. The resulting models, post-training, included the MFCC-SVM classifier, the Energy-SVM classifier, and the ZCR-SVM classifier.

In their previous work, Purwins, Virtanen, Schluter, Chang, and Sainath [28] explored audio signal processing methods like Gaussian mixture models, hidden Markov models, and non-negative matrix factorization. However, they found that these traditional methods were often outperformed by deep learning models when sufficient data was available. They applied various techniques such as categorization, audio features, models, data, and evaluation, and conducted cross-domain comparisons with speech, music, and environmental sounds. Additionally, for audio synthesis and transformation, they employed source separation, speech enhancement, and audio generation methods.

Rouas [29], based on public transport vehicles, studied the detection of audio events. For this purpose he created an automatic audio segmentation, which divides an audio signal into several consecutive, almost stationary zones. The developed algorithm detected activity, i.e., ignored the quiet and low noise zones, focusing exclusively on the high noise zones. In this work the SVM model was used.

Crocco [9] conducted a systematic review of surveillance based on the audio signal. In this review, several approaches are presented, namely: i) background subtraction by monomodal analysis; ii) background subtraction by multimodal analysis; iii) audio event classification; iv) source localisation and tracking, especially audio source localisation; v) audiovisual source localisation; and vi) audio source tracking and audiovisual source tracking.

Gavira [30] has presented a device designed to accurately perform the recognition task in urban areas with high noise. The audio was recorded in real urban environments using a current microphone. The strategy was to train a classifier based on temporal and frequency data analysis, and deep convolutional neural networks were used to develop the work.

Hossain [31] proposes a system for emotion recognition through audiovisuals, using two deep networks to extract features and join the features. In addition, it uses Big Data technology to train the emotion network and separate the information based on gender. The proposed system will also use a CNN network for audio signals and a three-dimensional CNN for video signals.

Another study [32] delves into the intricate task of Motivic pattern classification in music audio recordings, with a particular focus on a cappella flamenco cantes. To tackle this, the paper proposes the application of Convolutional Neural Networks (CNN) architectures for intra-style classification of flamenco cantes, utilizing small motivic patterns. The suggested architecture capitalizes on the advantages of residual CNN for feature extraction and incorporates a bidirectional LSTM layer to handle the sequential nature of musical audio data. Sequential pattern mining and contour simplification techniques are employed to extract relevant motifs from the audio recordings, and Mel-spectrograms of these motifs serve as inputs for the various architectures tested. The research investigates the practicality of motivic patterns for automatically classifying music recordings and explores the influence of audio length and corpus size on the overall classification accuracy.

B. Data Augmentation

Related to data augmentation techniques, a study [33] focuses on enhancing the accuracy of animal audio classification through various data augmentation techniques. These techniques involve manipulating the existing audio data to create additional samples, thereby increasing the diversity and size of the dataset. The study investigates different augmentation methods, their impact on model performance, and their ability to mitigate challenges such as limited labeled data. By implementing these augmentation strategies, the paper aims to enhance the robustness and effectiveness of animal audio classification models, ultimately improving their ability to accurately identify and classify animal sounds.

Another work [34] presents a methodology for effectively classifying environmental sounds using a deep convolutional neural network (CNN) that incorporates regularization techniques and data augmentation. The study emphasizes the challenges of environmental sound classification, including limited labeled data and diverse acoustic variations. To address these challenges, the proposed approach involves augmenting the dataset through various techniques and integrating regularization methods into the CNN architecture. The experimental results demonstrate that the combination of data augmentation and regularization enhances the model's ability to accurately classify environmental sounds, making it more robust to variations in acoustic conditions and contributing to improved classification performance.

Also another study [35] introduces a novel technique for augmenting audio data using an evolutionary-based generative approach. The method involves employing evolutionary algorithms to generate new audio samples that are structurally similar to the existing data while introducing variations. By iteratively refining these generated samples, the approach aims to create diverse and realistic audio data that can expand the training dataset for machine learning models. The paper highlights the benefits of this approach in improving the performance of audio-based tasks such as classification and recognition, demonstrating its effectiveness in enhancing model generalization and accuracy through the incorporation of synthetically generated but plausible audio samples.

Finally, a last study [36] presents a method for automating the selection of effective data augmentation techniques to enhance object detection models. It addresses the challenge of selecting appropriate augmentation strategies from a large set of possibilities by utilizing a reinforcement learning framework. The approach involves training a policy network that learns to select augmentation operations based on their impact on the model's performance. This policy network is optimized through reinforcement learning techniques, resulting in a strategy for augmenting the training data that improves the object detection model's accuracy. The paper demonstrates the effectiveness of the approach through experiments, showing that learned data augmentation strategies can lead to significant performance gains in object detection tasks.

The background discussed in this section highlights the progress achieved in the development of methods for identifying violence and the latest enhancements in data augmentation techniques. However, when we narrow our focus to the particular scenario of detecting violence using audio within a vehicle, the existing models are not well-suited, and there is a lack of datasets recorded in such settings. Therefore, our study aims to enhance the effectiveness of violence detection within cars by utilizing audio inputs and a newly captured in-car dataset. Additionally, we emphasize the significance of employing data augmentation techniques to improve the results in this context.

III. Methods

A. Mel Spectrogram

Audio can be converted into an interpretable format by representing it as visual images. The key concept involves transforming the audio signal into visual images, which can then be utilized to extract features either manually or directly fed into a Deep Learning classifier. There exist classifiers that can learn and extract features from these audio-generated images [37].

There are some methods tha can be used to create this images (spectrograms), that represent the audio, and some are: *Short-Time Fourier Transform*, *Chromagram*, *Mel-Spectrogram* [7]. According to the literature by Choi, Fazekas, Cho, and Sandler [38]; Gaviria et al. [30]; Hossain and Muhammad [39]; Purwins et al. [40], each method for audio representation comes with its own set of advantages and disadvantages. Nonetheless, the Mel-Spectrogram method stands out as the most widely utilized approach. Therefore, we have chosen to employ the Mel-Spectrogram method to represent audio in order to test our model.

A mel-spectrogram is a type of spectrogram, which visualizes the frequency content of an audio signal over time. However, instead of using a linear scale for the frequency axis, the mel-spectrogram uses the mel scale. The mel scale is a perceptual scale that is designed to better align with how humans hear and perceive sound [30]. The mel scale was introduced in the 1930s in order to account for the fact that humans do not perceive changes in frequency linearly - that is, changes in pitch at lower frequencies are more noticeable than at higher frequencies. The mel scale is based on this perceptual phenomenon, and is designed so that equal distances on the scale correspond to equal perceived differences in pitch. In practical terms, the mel scale is used to create a filterbank that is applied to the Fourier transform of an audio signal to map it onto the mel scale [40].

So, a mel spectrogram displays the time-frequency distribution of audio, with the frequency axis based on the mel-frequency scale. The process of converting to a mel spectrogram involves computing the Short-Time Fourier Transform (STFT) of the audio signal. This STFT computation transforms the audio from the time domain to the frequency domain. Once in the frequency domain, the y-axis is scaled using a mel-scale [41].

The mel spectrogram displays the successive frequencies (y-axis) over time (x-axis) as well as the different amplitudes (represented by colors and measured in decibels) for each moment (Fig. 1).

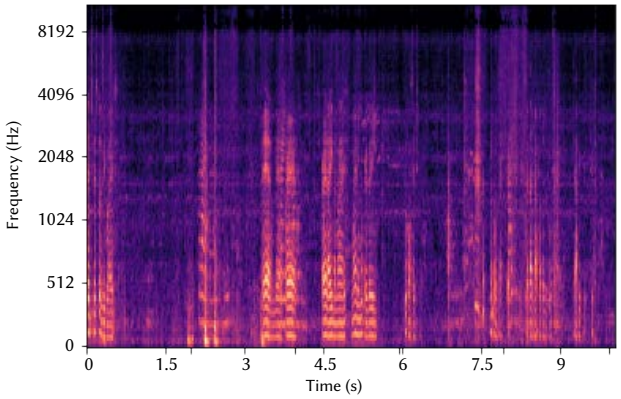


Fig. 1. Representation of a mel spectrogram.

B. Public Datasets

One of the key points in ML is selecting a dataset that has the necessary restrictions of what is intended to be classified. For the problem of this paper, the dataset should have audio entries where each entry would fall into violence or non violence. With this restrictions, we found some datasets worth mentioning.

The XD-Violence dataset is comprised of 4754 videos, with varying degrees of audio availability. It is divided into two categories: violence (2405 videos) and non-violence (2349 videos), totaling 217 hours of footage. The videos that depict violence can be further categorized into six types, including abuse, explosions, car accidents, riots, fights, and shootings. Each video of violence can have 1 to 3 labels, reflecting the significance of each event depicted. The videos come from a variety of sources, including movies, cartoons, video games, news, sports, etc. The XD-Violence dataset is separated into two parts, training and testing. The training section has 3954 videos, while the testing section has 800 videos. In both the training and testing datasets, the six types of violence are present at different points in time within the videos [42] [43].

Nanyang Technological University CCTV-Fights dataset contains 1000 videos obtained from YouTube, some without audio, that display various actions such as pushing, kicking, fighting, etc. It is separated into two categories: CCTV (280 videos captured by surveillance cameras) and NON-CCTV (720 videos captured by dash-cams, cell phones, drones, and helicopters). The CCTV (camera stands for Closed-Circuit Television camera) videos range from five seconds to 12 minutes (average of two minutes), totaling 8.54 hours, while the NON-CCTV videos range from three seconds to seven minutes (average of 45 seconds), adding up to 9.13 hours of footage [44] [45].

Violent Scenes Detection (VSD2014) is widely used when the problem is to detect violence through video or audio. It has two types of videos: clips from Hollywood movies and clips taken from YouTube. The dataset is divided into three groups: "Hollywood: Development", "Hollywood: Test" and "YouTube: Generalization". In terms of the Hollywood group, they selected some movies, and it can go from movies with some violence ("Saving Private Ryan", with 34% frames with violence) to movies without violence ("Legally Blond", with 0% frames with violence). The "Hollywood" group has a total of 63 hours and 55 minutes of movie time (31 movies), while clips from YouTube has a total of two hours and 37 minutes (86 clips) and each clip can last from six seconds to six minutes. The features offered by this dataset are separated into audio and visual features, to make it easier for those without much experience in classification to have a starting point. To complete the dataset, annotations are included for all the content. The annotations identify the start and end frames of each violent segment and are binary in nature. There are seven visual concepts and three audio ones. The visual elements include: fights, blood, fire, knives, car pursuits, and disturbing/bloody images, which may also provide information about the level of intensity. The audio elements include: shots, screams, and explosions. It should be noted that the visual elements provide the start and end of each segment, expressed in terms of frames. Meanwhile, the audio elements are described in terms of seconds for the start and end of each occurrence [46].

The Real Life Violence Situations (RLVS) dataset contains real-world violent scenarios used for research in fields like computer vision. The purpose of the RLVS dataset is to supply a varied and accurate set of violent situations for the purpose of training and evaluating algorithms and systems with the aim of detecting, preventing, and responding to acts of violence. The dataset is comprised of 2000 clips, half of which depict violence and the other half do not. Some of the clips have been manually captured. In an effort to eliminate redundancy of individuals and surroundings, additional videos were taken from the YouTube platform. The lengthy clips have been broken down into shorter ones, ranging from three to seven seconds, with an average duration of five seconds. All of these clips are of high resolution and some of them have no sound. The violent clips depict scenes from places like prisons, schools, streets, etc. The non-violent clips feature individuals participating in activities like playing walking, eating, sports, etc. This dataset includes a wide variety of race, gender and age [47].

A brief summary can be seen on the Table I.

However, despite the existence of several audio-based datasets, none have met the specified constraints for this work. So a group of researchers made their own dataset.

C. In Car Dataset

In order to evaluate the implemented models, a dataset was necessary, but no existing dataset met the specific requirements. Consequently, a team of researchers decided to create their own dataset, capturing video recordings of both violent and non-violent scenarios inside a car, involving real people, and all recorded during the pandemic. The dataset consists of videos, each with accompanying audio, representing 20 distinct scenarios. Among these scenarios, 12 involve violence, including push and punch incidents, different fight scenarios, discussions with physical altercations, sexual harassment situations, and robberies using weapons like knives or guns. One scene depicts one person forcibly looking at another's phone. On the other hand, the remaining 8 scenarios are non-violent, featuring instances such as people hugging, taking photos, fixing hair, sleeping,

TABLE I. Overall Analysis of the Datasets

Dataset	Number of videos	Duration (hours)	Sources	Audio
XD- Violence	4754	217	Movies, cartoons, videogames, news, sport, etc.	Yes
NTU CCTV-Fights	1000	18	Surveillance cameras and mobiles	Yes*
VSD2014	31 Movies + 86 Clips	64+3	Hollywood movies and clips from YouTube	Yes
RLVS	2000	-	Manually recorded and clips from YouTube	Yes*

* Some videos lack sound or only have background music.

sneezing, reading a book, yawning, listening to music, answering calls, coughing, using a notebook, and writing, along with using alcohol gel. Each scenario was recorded with 16 different pairs of actors, and certain scenes include the use of various objects. For each pair, each scenario was recorded twice, P1 is one person and P2 is the other one.

The dataset is comprised of video files, with 494 entries depicting non-violent scenes and 795 depicting violent scenes. Every video file has audio, and that audio can go from the scene in itself or just noise.

The violence scenarios can be described as:

1. A person (P1) requests a kiss; A second person (P2) refuses the kiss; P1 insists; P2 slaps P1; A conflict ensues between the two.
2. P2 is on the phone; P1 approaches; P2 shoves P1; P1 insists on seeing the phone.
3. P2 is sleeping; P1 drinks water from a bottle; P1 throws the bottle at P2; P2 wakes up and shoves P1.
4. P1 and P2 are on the phone. They engage in a dispute, leading to a physical conflict.
5. P1 threatens P2 with a knife; P1 harasses P2 by touching their body.
6. P1 pulls out a knife and points it at P2; P1 stabs P2.
7. P1 draws a gun and points it at P2; P1 shoots P2 with the weapon.
8. P1 greets P2; P1 shows something on the phone and threatens P2 with scissors; P1 robs P2.
9. P1 approaches P2, touches a non-sexual part of P2; P2 slaps P1.
10. P2 threatens to strike P1; P1 behaves in a provocative manner, and P2 slaps P1.
11. P2 performs an obscene gesture; P1 attacks P2 with a closed fist and attempts to strangle him.
12. A discussion with hand gestures, shoves, and punches.

As for the non-violent scenarios, they can be described as follows:

1. P1 is writing in a notebook, while P2 is applying hand sanitizer.
2. P1 answers a phone call; P2 uses a notebook and coughs.
3. P1 drinks and eats; P2 takes pictures.
4. P1 yawns and stretches; P2 puts on the headphones to listen to music.
5. P1 sneezes; P2 reads a book/newspaper/magazine.
6. P1 applies lipstick and arranges her hair; P2 sleeps.
7. P1 asks P2 to take a picture; P2 takes several pictures of him; P2 shows P1 the pictures taken.
8. P1 and P2 talk; P2 cries; P1 and P2 embrace.

D. Pre-Processing

Data pre-processing is a crucial step to reduce the difficulty of learning features of the algorithm [48]. In the section III.C, we talked about the dataset created. This dataset only had 494 videos without violence on it and 795 videos with violence. The data pre-processing follows the flow represented in Fig. 2.

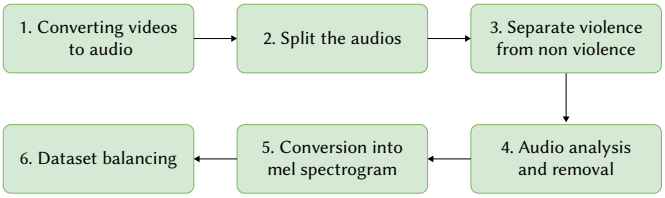


Fig. 2. Pre-processing steps.

As for the first step all the videos had to be converted to audios so we could create the mel spectrograms of each one. In the second step, the audios without violence that had more than 40 seconds were split in half so each entry became two entries in the dataset with the same label. The audio recordings of violent incidents were typically longer than those of non-violent incidents, but the issue was that violence was often not present in the beginning of the audio. The solution involved inspecting each audio individually to determine the start of violence, and using the "pydub" library[1], the audio could be divided into two parts - one representing non-violence, and the other representing violence. During the process of analysing each audio (step 4), it was discovered that some files lacked content and that some audio recordings did not have meaningful information for the data (e.g. audio recordings that only had background noise). These were removed from the dataset. By the end of the fourth step in the workflow, the dataset had 860 audio files of non-violence and 755 audio files of violence, for a total of 1615 audio files. The step five was mixed with step six. We converted every audio into a mel spectrogram that could represent the audio in itself, so all the 1615 were converted and then it was decided that a good approach would be to balance the dataset so we used some entries from the RLVS dataset refered in section III.B. We tried to find the best violence videos in RLVS dataset that could go into our dataset. We found 105 violent videos, and those were converted to audio and then converted into a mel spectrogram to be added to the dataset. Ending this workflow, the dataset had 860 non-violence mel spectrograms and 860 violent mel spectrograms, with a total of 1720 mel spetrograms.

In Fig. 3 it is shown what a mel spectrogram created from an audio with violence and and audio without violence looks like.

[1] https://thepythoncode.com/assistant/transformation-details/cutting-audio-files-in-python-with-pydub/

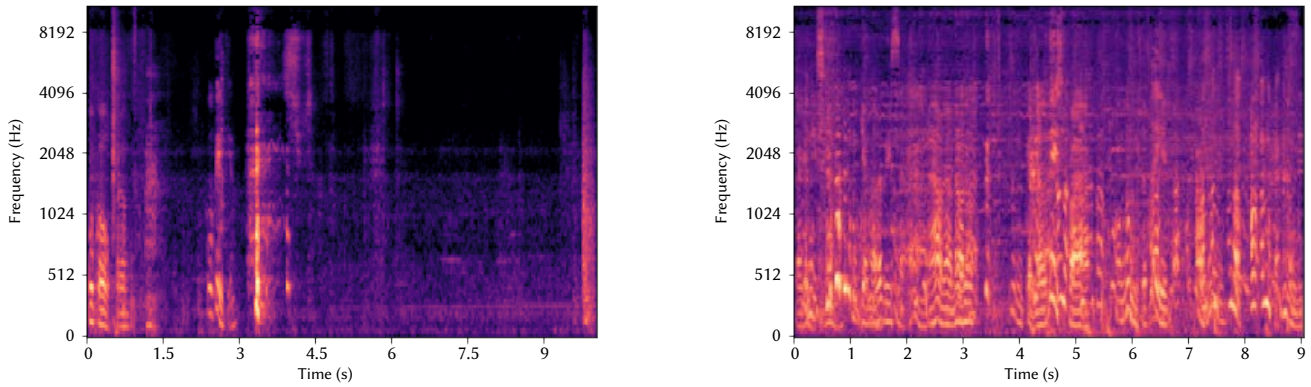
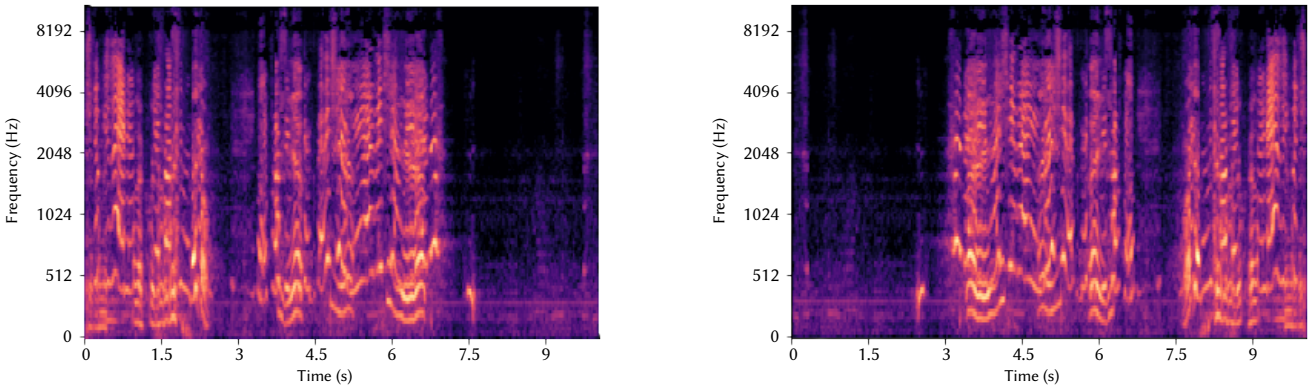


Fig. 3. Mel spectrogram created from (a) an audio without violence and (b) an audio with violence.



(a) Original spectrogram.

(b) Spectogram after flipping.

Fig. 4. Horizontal flipping of a spectrogram.

E. Data Augmentation

In deep learning, there is the notion that large datasets lead to better training which can lead to best accuracies. But collecting enough data to create a suitable dataset can be a challenging task at times. The data augmentation mechanism is frequently employed to generate a large volume of training data by adding synthetic data to the dataset. These synthetic data may consist of copies of existing data but with minor changes or completely new data created from the data already on the dataset [49].

Some examples of data augmentation were:

- **Cropping**: The process involves trimming the image, thus decreasing its input size;
- **Rotation**: Consists in rotating the image, between 1 and 359 degrees;
- **Translation**: This process involves in moving the image along the x-axis or y-axis (left, right, up or down);
- **Flipping**: The image is flipped vertically or horizontally;
- **Scalling or resizing**: The image is resized to a given size;
- **Noise injection**: The process entails adding a matrix of randomly generated values;
- And other methods that are more complex to achieve.

For the problem meant to be solved which was the classification of mel spectrograms, the data augmentation that we used was flipping. With this flipping method we were able to duplicate the number of entries of the dataset, where we performed a horizontal flip in each entry. This resulted in a dataset with 3440 mel spectrograms, 1720 for each class.

This flip method was most useful in violence entries because some of the entries had violence since the start and it would calm down later in the audio. But after this data augmentation, we were able to show that it also can start with a calm environment and then escalate the situation to pure violence. The Fig. 4 ilustrates this last case, where in a) we see a mel spectrogram taken from an audio with violence and in b) this same mel spetrogram after being flipped. Every mel spectrogram had also his axis removed for the final dataset.

F. Algorithms

This section provides an overview of all the algorithms evaluated in this project. The tested algorithms include Convolutional Neural Network (CNN), EfficientNetB0, EfficientNetB1, EfficientNetB2, MobileNet, MobileNetV2, ResNet50, VGG16, VGG19, and Xception. The selection of these models is supported by the findings from the literature review.

Convolutional Neural Network (CNN) is a deep learning algorithm used for image classification. Its architecture was inspired by the human brain. This network can extract features directly from the image without requiring human assistance [50].

EfficientNets are a type of artificial neural networks that take into account the scaling process and the importance of the base network. They feature a unique mechanism called the compound scaling method, which enables the network to be uniformly scaled in terms of depth, width, and resolution. The base network is the EfficientNetB0 (for example, EfficientNetB1 is a scaled version of EfficientNetB0). These networks can achieve better performance than existing CNN models while using less number of parameters [51] . EfficientNetB0, EfficientNetB1, and EfficientNetB2 belong to the EfficientNet family of image classification models. Here are the key distinctions between these three models: i)Depth: EfficientNetB0 has the fewest layers with 20 convolutional layers, while EfficientNetB1 has 23 convolutional layers, and EfficientNetB2 has 26 convolutional layers; ii) Width: As we progress from B0 to B2, the width of the network increases. This means that the number of channels in each convolutional layer is larger in EfficientNetB2 than in EfficientNetB1, and larger in EfficientNetB1 than in EfficientNetB0; and iii) Resolution: The input resolution of EfficientNetB2 is higher than that of EfficientNetB1, and EfficientNetB1 has a higher resolution than EfficientNetB0. Consequently, EfficientNetB2 is better equipped to handle high-resolution images. In general, moving from EfficientNetB0 to B2 results in a model that is deeper, wider, and more capable of processing high-resolution images. However, with each step up the scale, the model also becomes more computationally demanding. The choice of which model to use depends on specific task requirements, including available compute resources and the resolution of the input images [51].

MobileNet was designed for efficient deployment on mobile and embedded devices with limited computational resources. This network is based on a CNN and uses depthwise separable convolutions, which leads to a decrease in the number of parameters when comparing to networks with regular convolutions and with the same depth. This process allows the network to be a lighter neural network [52].

The Residual Network (ResNet) was created to address the issue of the vanishing gradient problem, making it possible to train a network with more than 1000 layers [7].

VGG, an acronym for Visual Geometry Group, is a deep convolutional neural network (CNN) architecture that is composed of multiple layers. This model is used for image classification and has been trained using the ImageNet dataset, making it a popular choice for transfer learning. VGG16 means that the neural network has 16 layers, while the VGG19 has 19 layers [53].

Xception is a network developed by Google, for image classification tasks. It uses the idea of depthwise separable convolution layers that decreases the computacional cost, and it was designed to be a more efficient alternative for the overall Inception architectures [54].

These models have been developed specifically for image classification tasks. EfficientNet is a family of optimized and efficient models that also achieve top-notch accuracy in image classification. MobileNet, on the other hand, is a family of lightweight and fast models, making them ideal for deployment on mobile and embedded devices. ResNet50 is a CNN architecture that cleverly employs skip connections to address the vanishing gradient problem during training, enabling the creation of very deep neural networks without compromising performance. VGG16 and VGG19 are CNN architectures known for their utilization of small 3x3 convolutional filters. While they demonstrate strong performance in image classification tasks, they can be computationally expensive during both training and deployment. Lastly, Xception is a CNN architecture that incorporates depthwise separable convolutions, performing a depthwise convolution followed by a pointwise convolution. This design results in better performance with fewer parameters compared to other architectures.

G. Training Details

As to prepare for the training of the algorithm, the dataset was divided initially into train and test. We decided that 80% of the dataset would be for the training, and 20%for the testing, giving a total of 2752 entries for training and 688 entries for testing (equal distribution between classes). With the necessity of a validation set, we used the 80% for training where 80% of those would be for training and the other 20% would be for validation. Ending the split phase, the train set consisted of 2202 entries, validation set had 550 entries, and the test set had 688.

Table II shows the class distribution between the three sets (train, validation and test set).

TABLE II. Train, Validation and Test Set

Dataset	Violence	Non violence	Total
Train	1101	1101	2202
Validation	275	275	550
Test	344	344	688

All the algorithms used the same callbacks: EarlyStopping[2], ReduceLROnPlateau[3], ModelCheckpoint[4], and TensorBoard[5]. The EarlyStopping was meant to stop the training of the algorithm in case the validation loss was not getting better. It had a patience of 25 for the CNN and 10 for the other algorithms. The ReduceLROnPlateau would reduce the learning rate if the validation loss did not improve; we used a factor of 0.1 for every algorithm, a patience of 10 for CNN and 5 for the rest of the algorithms. The ModelCheckpoint would save the model weights in a file. To visualize all the training done (accuracy and loss during the different epochs) it was used the callback TensorBoard.

The Table III shows all the training details for each algorithm. All of them used Adamax as optimizer, with a learning rate of 0.001. CNN was meant to run for 200 epochs, while the others ran for 40 epochs. Batch size used was 64, with a resize to (150,150) on each entry (mel spectrograms). The last column of the table (EarlyStopping) shows the epoch that the algorithm stopped the training because of the callback EarlyStopping.

As pre-trained networks on *ImageNet* have demonstrated remarkable results across multiple fields such as image classification datasets, object detection, action recognition, and more [55], we decided that all the algorithms would use the weights from training the network on ImageNet dataset. Those weights are available on the python library *Keras*[6].

The training was done on a computer with a GeForce GTX 1070 Ti, 16GB RAM, and a AMD Ryzen 5 2600 as CPU.

IV. Results and Discussion

Table IV shows the best results obtained by the different algorithms, with all values corresponding to the epoch that achieved the best validation loss.

The VGG16 network performed better on the test in terms of accuracy (91.86%) than VGG19 (91.28%). However, it has slightly worse test loss compared to VGG19. Of the transfer learning networks, Xception had the lowest test accuracy at 90.70%, while ResNet50 had a slightly better result at 90.84%. Both had a similar test loss that was around 0.25.

In regards to the MobileNet family of networks, MobileNet achieved superior results in training, validation, and test, even reaching an accuracy of 93.31% on test. Nevertheless, MobileNetV2 also performed well on test with an accuracy of 92.44% when compared to the previously evaluated networks.

The family of EfficientNet achieved the best results, with EfficientNetB1 achieving the highest accuracy in the test (95.06%), followed by EfficientNetB0 with 94.19%. Moreover, in terms of test loss, EfficientNetB1 had the best performance with a loss of 0.1685, followed by EfficientNetB0 with 0.1772. Although EfficientNetB2 had the weakest performance within the family, it still achieved a satisfactory accuracy of 92.88%.

[2] https://keras.io/api/callbacks/early_stopping/

[3] https://keras.io/api/callbacks/reduce_r_on_plateau/

[4] https://keras.io/api/callbacks/model_checkpoint/

[5] https://www.tensorflow.org/

[6] https://keras.io/

TABLE III. Details of the Training for Each Algorithm

Algorithm	Optimizer	Learning Rate	Epochs	Batch	Resize	EarlyStopping
CNN	Adamax	0.001	200	64	(150,150)	82
EfficientNetB0	Adamax	0.001	40	64	(150,150)	28
EfficientNetB1	Adamax	0.001	40	64	(150,150)	31
EfficientNetB2	Adamax	0.001	40	64	(150,150)	21
MobileNet	Adamax	0.001	40	64	(150,150)	23
MobileNetV2	Adamax	0.001	40	64	(150,150)	20
ResNet50	Adamax	0.001	40	64	(150,150)	14
VGG16	Adamax	0.001	40	64	(150,150)	16
VGG19	Adamax	0.001	40	64	(150,150)	15
Xception	Adamax	0.001	40	64	(150,150)	20

TABLE IV. Accuracy and Loss on the Train, Validation and Test Set, for Each Algorithm

Algorithm	Train	Train Loss	Validation	Validation Loss	Test	Test Loss
CNN	89.28	0.2827	90.00	0.2579	89.53	0.2877
EfficientNetB0	95.19	0.1427	91.09	0.2030	94.19	0.1772
EfficientNetB1	95.19	0.1328	91.92	0.1912	95.06	0.1685
EfficientNetB2	91.05	0.2178	91.82	0.2117	92.88	0.2139
MobileNet	92.51	0.2087	89.82	0.2090	93.31	0.1926
MobileNetV2	88.74	0.2367	88.36	0.2457	92.44	0.2054
ResNet50	83.11	0.3583	89.64	0.2436	90.84	0.2535
VGG16	88.33	0.2679	88.00	0.2873	91.86	0.2259
VGG19	87.33	0.3016	86.55	0.3270	91.28	0.2238
Xception	92.14	0.2056	87.82	0.2967	90.70	0.2527

TABLE V. Results From Precision, Recall and F1-Score of the Algorithms

Algorithm	Precision		Recall		F1-Score	
	0	1	0	1	0	1
CNN	0.89	0.90	0.90	0.89	0.90	0.89
EfficientNetB0	0.96	0.92	0.92	0.96	0.94	0.94
EfficientNetB1	0.95	0.95	0.95	0.95	0.95	0.95
EfficientNetB2	0.93	0.93	0.93	0.93	0.93	0.93
MobileNet	0.93	0.94	0.94	0.93	0.93	0.93
MobileNetV2	0.90	0.96	0.96	0.89	0.93	0.92
ResNet50	0.93	0.89	0.89	0.93	0.91	0.91
VGG16	0.92	0.92	0.92	0.92	0.92	0.92
VGG19	0.91	0.92	0.92	0.90	0.91	0.91
Xception	0.92	0.89	0.89	0.92	0.91	0.91

Class 0 represents non-violence inputs; 1 represents violence inputs.

Accuracy and loss are the primary metrics used to evaluate the behavior of various algorithms. However, there are other metrics that assist in the evaluation of algorithms, and these metrics are widely used in the world of ML. The metrics that are often used to evaluate the algorithm are: precision, recall, and f1-score [56].

Table V presents the recall, precision, and f1-score values for each class, with the value 0 representing the non-violence class and the value 1 representing the violence class.

The outcome of an algorithm can fall into four distinct categories, namely, TP (True Positive), TN (True Negative), FP (False Positive), and FN (False Negative). When considering violence entries as "Positive" and non-violence entries as "Negative," TP represents the correctly predicted violence entries, TN denotes the correctly predicted non-violence entries, FP includes the misclassified non-violence entries, and FN comprises the misclassified violence entries. These four categories collectively form a matrix known as the confusion matrix, which effectively reflects the algorithm's performance.

Taking into account the concept of accuracy and the four aforementioned values, the formula is as follows:

$$\frac{TP + TN}{TP + TN + FP + FN} = \frac{\#\ \text{CorrectForecast}}{\#\ \text{Forecast}} \tag{1}$$

In contrast, the precision indicates the number of correct positive forecast (Equation (2)).

$$\frac{TP}{TP + FP} = \frac{\#\ \text{CorrectlyPredictedPositives}}{\#\ \text{PositiveForecasts}} \tag{2}$$

The recall, as stated in Equation (3), represents the count of true positive cases that the algorithm correctly predicted.

$$\frac{TP}{TP + FN} = \frac{\#\ \text{CorrectlyPredictedPositives}}{\#\ \text{TotalPositiveDataset}} \tag{3}$$

Finally, the f1-score (Equation (4)) combines the *precision* with the *recall*, in order to produce a value that represents both weights (*precision* and *recall*) in a balanced way.

$$F1 - \text{Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

EfficientNetB0 achieved the highest precision of 96% for the non-violence class, although its precision for the violence class was not as high. Nevertheless, it had the highest recall for the violence class, with 96%. EfficientNetB1 came in second for precision for the non-violence class with 95%, and the same precision value was obtained for the violence class. Moreover, EfficientNetB1 attained a recall of 95% for both classes, and the f1-score demonstrated identical results of 95%. EfficientNetB2 achieved a value of 93% in all the analyzed fields.

MobileNetV2 had the highest recall for the non-violence class at 96%. However, it had the poorest recall for the violence class, which is an important consideration when choosing which algorithm to use, even though it had the highest precision for the violence class. The MobileNet algorithm presented good results in all fields of the confusion matrix, even though it did not perform the best when compared to all the algorithms.

Although MobileNetV2 had good accuracy and precision for the violence class, this network is not suitable for classifying violent inputs as it has the lowest recall for this class, which is the most important class to classify.

Evaluating all the results obtained in the two tables, the algorithms most suitable for this problem are: EfficientNetB1, EfficientNetB0, and MobileNet.

A. (RQ1) Can Violence Inside a Car Be Effectively Detected Using Audio Data and Deep Learning Models?

Violence inside a car can be effectively detected using audio data and deep learning models. Audio-based violence detection has gained significant attention in recent years, and deep learning models have shown promising results in accurately classifying violent and non-violent audio events.

The results presented in Table 4 demonstrate the high accuracy in detecting violence behavior. As previously mentioned, the models from the EfficientNet family showcased the best performance. When compared to the background, particularly the study by Duraes, Santos, Marcondes, Hammerschmidt and Novais [18], our models yielded superior results. It is important to note that the other background studies have not been specifically applied to the unique environment inside a car.

B. (RQ2) Can the Use of Data Augmentation Enhance the Accuracy of Violence Detection Results?

Data augmentation typically leads to several benefits in the context of deep learning models:i) improved model accuracy, by creating variations in the training data, data augmentation can enhance the accuracy of deep learning models, particularly when dealing with small datasets; ii) increased amount of training data, because obtaining large amounts of labeled data can be challenging and costly; iii) reduced overfitting, because overfitting occurs when a model becomes overly complex and starts fitting noise in the training data instead of the underlying pattern and data augmentation introduces variations to the training data, mitigating overfitting and preventing the model from relying too heavily on a limited number of training examples; iv) better generalization by adding variability to the training data through data augmentation aids deep learning models in generalizing better to new and unseen data, leading to improved performance in real-world scenarios; and v) faster model development, where data augmentation can accelerate the model development process by reducing the time required to collect and label large datasets for training deep learning models.

In comparison with the previous study [26], where data augmentation was not applied, the results presented in this paper show a better increase in performance.

V. Conclusion

The fact that violence is very present in today's society makes the study of violence detection an asset.

Determining how to capture violence is the primary factor that determines the selection of an architecture. Studies have shown that violence can be captured using either video (cameras) or audio (microphones). Since the use of audio to detect violence has more advantages when compared to video, it was decided that audio would be the mechanism to use.

To enable ML architectures to accurately classify audio, it was necessary to find a way to represent all the information contained in it in a compact way (such as an image). Mel spectrograms were utilized to represent audio as images for this task, since this approach is commonly employed and yields good accuracies in audio classification.

Datasets that contained the necessary constraints for the problem were also sought. However, there was no dataset that had all the necessary constraints, so a dataset created by researchers was preprocessed accordingly. A data augmentation process was also applied to the dataset, resulting in a dataset with twice the amount of data.

For the final evaluation, the custom CNN algorithm, EfficientNetB0, EfficientNetB1, EfficientNetB2, MobileNet, MobileNetV2, ResNet50, VGG16, VGG19, and Xception were evaluated. The algorithm that achieved the highest accuracy was EfficientNetB1 with an accuracy of 95.06%, followed by EfficientNetB0 with 94.19%, making the EfficientNetB1 the best algorithm to use in order to detect violence in audio. Additionally, it was found that the worst neural network for classifying violence inputs is MobileNetV2, so it should not be the most suitable for solving the problem at hand.

In future work, the intention is to compare the current approach with other methods, specifically those that involve transforming audio data into text and subsequently analyzing the text. This could involve using techniques such as automatic speech recognition (ASR) to convert the audio content into text transcripts, which can then be further processed and analyzed using natural language processing (NLP) or other text-based analysis methods. By exploring these alternative approaches, researchers aim to gain insights into the effectiveness and suitability of different methodologies for violence detection and potentially discover novel insights from the textual representations of audio data.

Appendix

On Appendix we present Fig. 5 to Fig. 14, which contain the detailed training made during the experience.

Fig. 5 depicted the accuracy and loss training curves over the epochs for CNN model. The model achieved it is best results after 80 epochs.

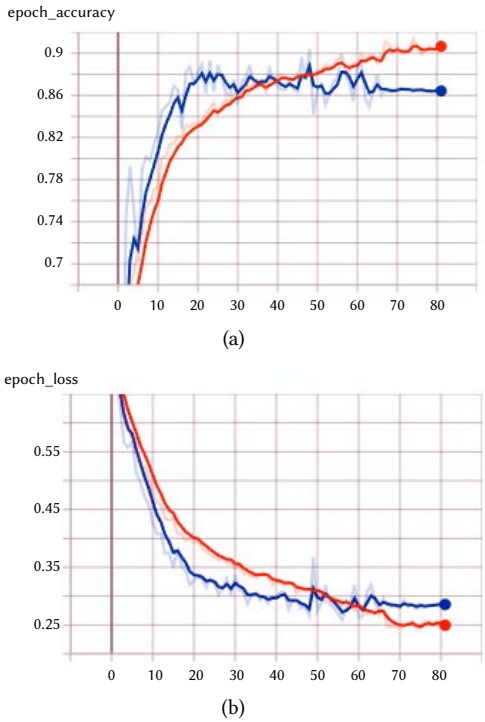


Fig. 5. a) Accuracy and b) loss curve for CNN. Training is represented by the orange line, and validation by the blue line.

Fig. 6 illustrated the accuracy and loss training curves across the epochs for the model EfficientNetB0. The model attained its optimal performance after 26 epochs.

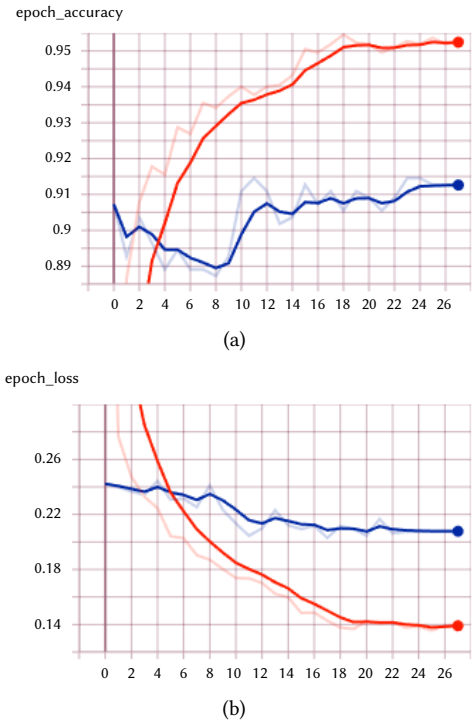


Fig. 6. a) Accuracy and b) loss curve for EfficientNetB0. Training is represented by the orange line, and validation by the blue line.

Fig. 7 displayed the accuracy and loss training curves throughout the epochs for the model EfficientNetB1. The model achieved its best performance after 30 epochs. However, it should be noted that the validation line showed some instability during the training process.

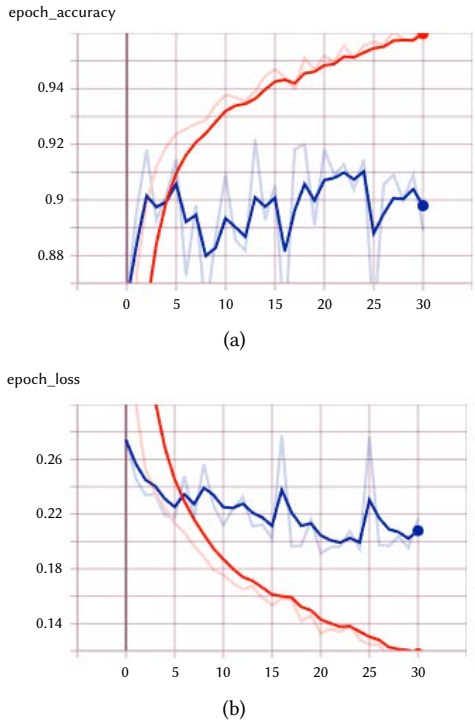


Fig. 7. a) Accuracy and b) loss curve for EfficientNetB1. Training is represented by the orange line, and validation by the blue line.

Fig. 8 depicted the accuracy and loss training curves over the epochs for the model EfficientNetB2. The model achieved its peak performance after 20 epochs. However, it should be acknowledged that the validation line displayed some instability around the 14th epoch during the training process.

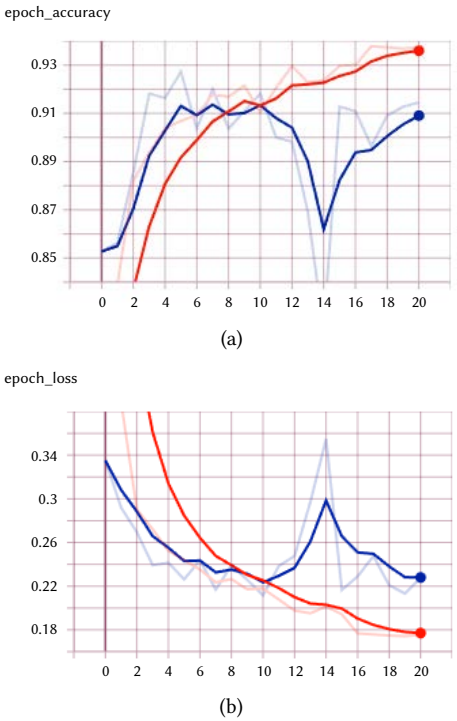


Fig. 8. a) Accuracy and b) loss curve for EfficientNetB2. Training is represented by the orange line, and validation by the blue line.

Fig. 9 presented the accuracy and loss training curves across the epochs for the model MobileNet. The model reached its optimal performance after 22 epochs.

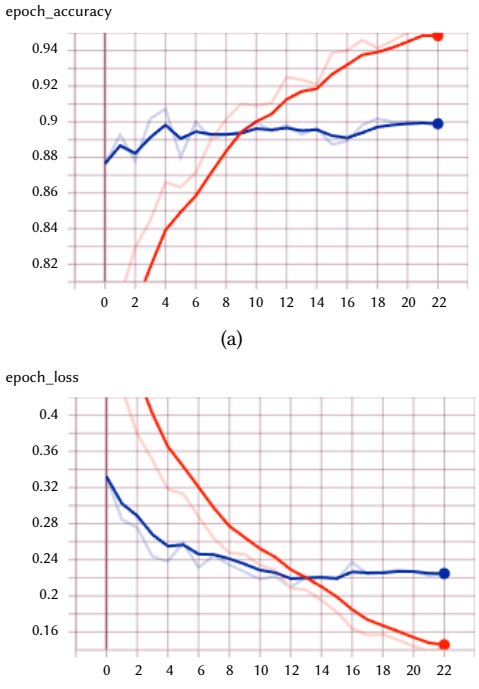


Fig. 9. a) Accuracy and b) loss curve for MobileNet. Training is represented by the orange line, and validation by the blue line.

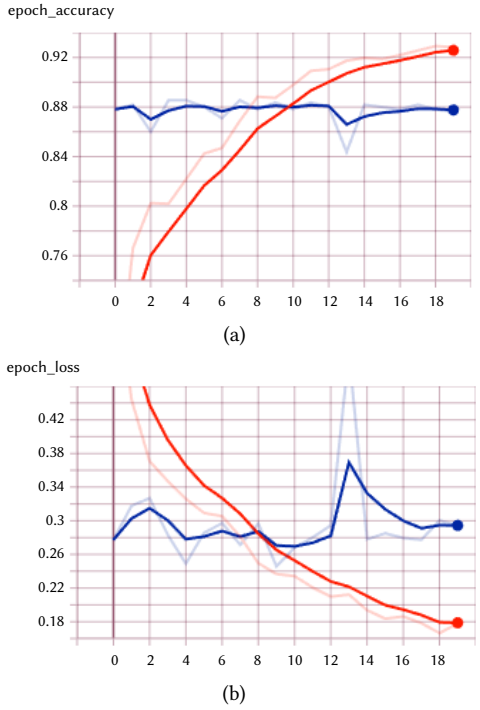


Fig. 10. a) Accuracy and b) loss curve for MobileNetV2. Training is represented by the orange line, and validation by the blue line.

Fig. 10 illustrated the accuracy and loss training curves throughout the epochs for the model MobileNetV2. The model achieved its best performance after 19 epochs. However, it is important to note that the validation line showed some instability around the 13th epoch during the training process.

Fig. 11 displayed the accuracy and loss training curves over the epochs for the model ResNet50. The model achieved its best performance after 12 epochs.

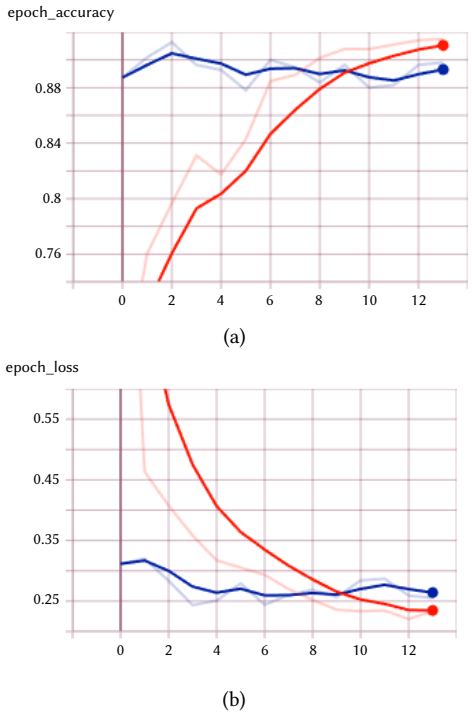


Fig. 11. a) Accuracy and b) loss curve for ResNet50. Training is represented by the orange line, and validation by the blue line.

Fig. 12 showed the accuracy and loss training curves throughout the epochs for the model VGG16. The model achieved its peak performance for accuracy after 14 epochs. However, it is important to note that the validation performance stabilized after 6 epochs.

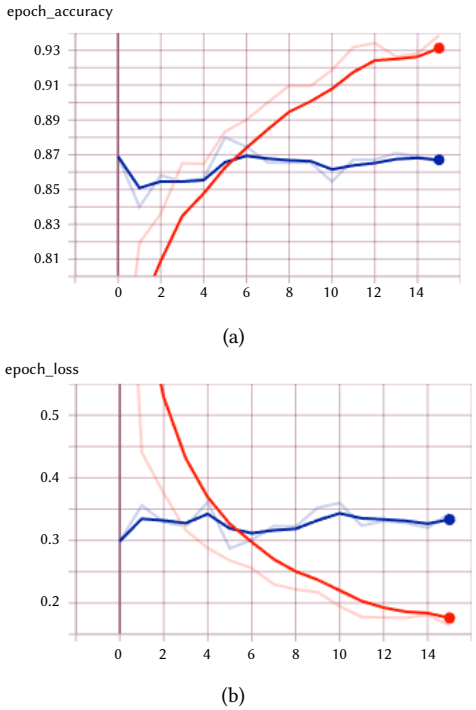


Fig. 12. a) Accuracy and b) loss curve for VGG16. Training is represented by the orange line, and validation by the blue line.

Fig. 13 displayed the accuracy and loss training curves over the epochs for the model VGG19. The model reached its optimal accuracy after 14 epochs.

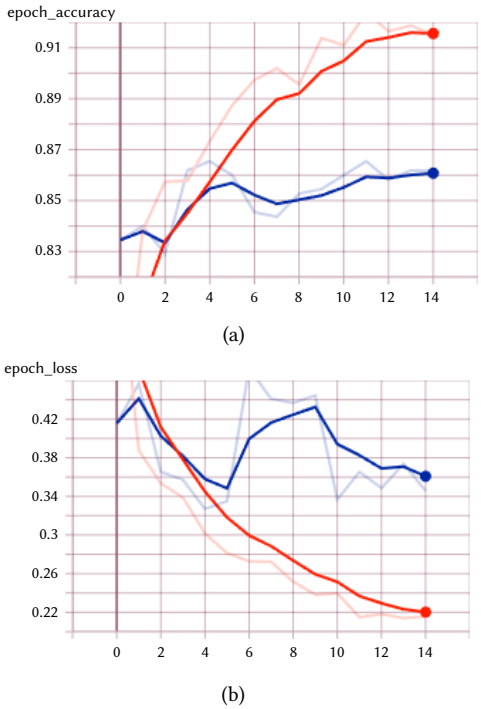


Fig. 13. a) Accuracy and b) loss curve for VGG19. Training is represented by the orange line, and validation by the blue line.

Fig. 14 illustrated the accuracy and loss training curves throughout the epochs for the model Xception. The model achieved its best accuracy after 19 epochs.

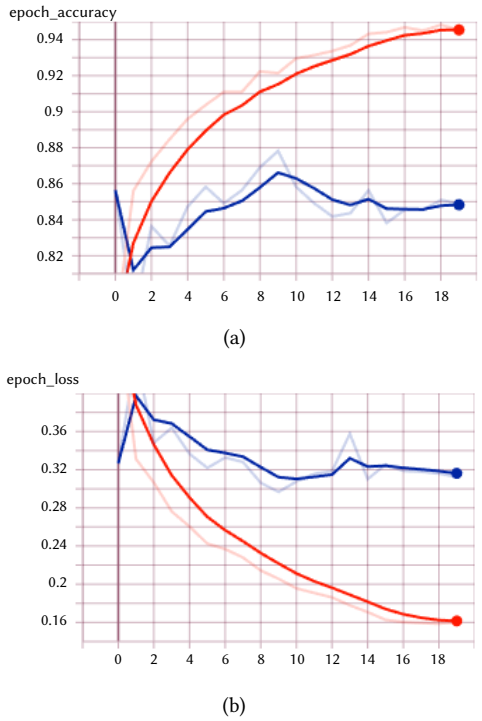


Fig. 14. a) Accuracy and b) loss curve for Xception. Training is represented by the orange line, and validation by the blue line.

Except for the CNN model, which required 80 epochs for training, all the other models needed less than 30 epochs for training. Among them, the VGG16 model had the lowest number of epochs needed for training.

Acknowledgment

This work has been supported by FCT – Fundação para a Ciência e Tecnologia within the R&D Units Project Scope: UIDB/00319/2020.

References

[1] S. Koritsas, M. Boyle, J. Coles, "Factors associated with workplace violence in paramedics," *Prehospital and disaster medicine*, vol. 24, no. 5, pp. 417–421, 2009.

[2] W. So, "Perceived and actual leading causes of death through interpersonal violence in south korea as of 2018," 2019.

[3] APAV, "Estatísticas apav -relatório anual 2020." https://apav.pt/apav_v3/images/pdf/ Estatisticas_APAV_Relatorio_Anual_2020.pdf, 2021. Access at 22/10/2021.

[4] D. Durães, F. Santos, F. S. Marcondes, S. Lange, J. Machado, "Comparison of transfer learning behaviour in violence detection with different public datasets," in *Progress in Artificial Intelligence*, 2021, Springer International Publishing.

[5] D. Durães, F. S. Marcondes, F. Gonçalves, J. Fonseca, J. Machado, P. Novais, "Detection violent behaviors: a survey," in *Ambient Intelligence–Software and Applications: 11th International Symposium on Ambient Intelligence*, 2021, pp. 106–116, Springer.

[6] A. Jan, G. M. Khan, "Real world anomalous scene detection and classification using multilayer deep neural networks," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 8, no. 2, pp. 158–167, 2023, doi: 10.9781/ijimai.2021.10.010.

[7] F. Santos, D. Durães, F. S. Marcondes, N. Hammerschmidt, S. Lange, J. Machado, P. Novais, "In-car violence detection based on the audio signal," in *Intelligent Data Engineering and Automated Learning– IDEAL 2021: 22nd International Conference, IDEAL 2021, Manchester, UK, November 25–27, 2021, Proceedings 22*, 2021, pp. 437–445, Springer.

[8] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, *et al.*, "Cnn architectures for large- scale audio classification," in *2017 ieee international conference on acoustics, speech and signal processing (icassp)*, 2017, pp. 131–135, IEEE.

[9] M. Crocco, M. Cristani, A. Trucco, V. Murino, "Audio surveillance: A systematic review," *ACM Computing Surveys (CSUR)*, vol. 48, no. 4, pp. 1–46, 2016.

[10] D. M. Beltrán-Flores, "Ópera nacionalista ecuatoriana," Master's thesis, 2022.

[11] K. Gkountakos, K. Ioannidis, T. Tsikrika, S. Vrochidis, I. Kompatsiaris, "Crowd violence detection from video footage," in *2021 international conference on content-based multimedia indexing (CBMI)*, 2021, pp. 1–4, IEEE.

[12] T. Senst, V. Eiselein, A. Kuhn, T. Sikora, "Crowd violence detection using global motion-compensated lagrangian features and scale-sensitive video-level representation," *IEEE transactions on information forensics and security*, vol. 12, no. 12, pp. 2945–2956, 2017.

[13] K. Gkountakos, K. Ioannidis, T. Tsikrika, S. Vrochidis, I. Kompatsiaris, "A crowd analysis framework for detecting violence scenes," in *Proceedings of the 2020 International Conference on Multimedia Retrieval*, 2020, pp. 276–280.

[14] T. Hassner, Y. Itcher, O. Kliper-Gross, "Violent flows: Real-time detection of violent crowd behavior," in *2012 IEEE computer society conference on computer vision and pattern recognition workshops*, 2012, pp. 1–6, IEEE.

[15] M. Sharma, T. Gupta, K. Qiu, X. Hao, R. Hamid, "Cnn- based audio event recognition for automated violence classification and rating for prime video content," *Proc. Interspeech 2022*, pp. 2758-2762, 2022, doi: 10.21437/Interspeech.2022-10053.

[16] A. J. Naik, M. Gopalakrishna, "Violence detection in surveillancevideo-a survey," *International Journal of Latest Research in Engineering and Technology (IJLRET)*, vol. 1, pp. 1–17, 2017.

[17] A. M. Yildiz, P. D. Barua, S. Dogan, M. Baygin, T. Tuncer, C. P. Ooi, H. Fujita, U. R. Acharya, "A novel tree pattern-based violence detection model using audio signals," *Expert Systems with Applications*, vol. 224, p. 120031, 2023.

[18] D. Duraes, F. Santos, F. S. Marcondes, N. Hammerschmidt, P. Novais, "Applying multisensor in-car situations to detect violence," *Expert Systems*, p. e13356, 2023.

[19] V. S. Saravanarajan, R.-C. Chen, C. Dewi, L.-S. Chen, L. Ganesan, "Car crash detection using ensemble deep learning," *Multimedia Tools and Applications*, pp. 1–19, 2023.

[20] F. Reinolds, C. Neto, J. Machado, "Deep learning for activity recognition using audio and video," *Electronics*, vol. 11, no. 5, p. 782, 2022.

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

[22] B. Peixoto, B. Lavi, P. Bestagini, Z. Dias, A. Rocha, "Multimodal violence detection in videos," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 2957– 2961, IEEE.

[23] A. S. Arukgoda, *Improving sinhala–tamil translation through deep learning techniques*. PhD dissertation, 2021.

[24] A. Uçar, Y. Demir, C. Güzeliş, "Object recognition and detection with deep learning for autonomous driving applications," *Simulation*, vol. 93, no. 9, pp. 759–769, 2017.

[25] Y. Cho, N. Bianchi-Berthouze, S. J. Julier, "Deepbreath: Deep learning of breathing patterns for automatic stress recognition using low-cost thermal imaging in unconstrained settings," in *2017 Seventh international conference on affective computing and intelligent interaction (acii)*, 2017, pp. 456–463, IEEE.

[26] B. Veloso, D. Durães, P. Novais, "Analysis of machine learning algorithms for violence detection in audio," in *Highlights in Practical Applications of Agents, Multi- Agent Systems, and Complex Systems Simulation. The PAAMS Collection: International Workshops of PAAMS 2022, L'Aquila, Italy, July 13–15, 2022, Proceedings*, 2022, pp. 210–221, Springer.

[27] H. Souto, R. Mello, A. Furtado, "An acoustic scene classification approach involving domestic violence using machine learning," in *Anais do XVI Encontro Nacional de Inteligência Artificial e Computacional*, 2019, pp.

705–716, SBC.
[28] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S.-Y. Chang, T. Sainath, "Deep learning for audio signal processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 206–219, 2019.
[29] J.-L. Rouas, J. Louradour, S. Ambellouis, "Audio events detection in public transport vehicle," in *2006 IEEE Intelligent Transportation Systems Conference*, 2006, pp. 733–738, IEEE.
[30] J. F. Gaviria, A. Escalante-Perez, J. C. Castiblanco, N. Vergara, V. Parra-Garces, J. D. Serrano, A. F. Zambrano, L. F. Giraldo, "Deep learning-based portable device for audio distress signal recognition in urban areas," *Applied Sciences*, vol. 10, no. 21, 2020, doi: 10.3390/app10217448.
[31] M. S. Hossain, G. Muhammad, "Emotion recognition using deep learning approach from audio–visual emotional big data," *Information Fusion*, vol. 49, pp. 69– 78, 2019.
[32] A. Arronte Alvarez, F. Gómez, "Motivic pattern classification of music audio signals combining residual and lstm networks," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, no. 6, pp. 208–214, 2021, doi:10.9781/ijimai.2021.01.003.
[33] L. Nanni, G. Maguolo, M. Paci, "Data augmentation approaches for improving animal audio classification," *Ecological Informatics*, vol. 57, p. 101084, 2020.
[34] Z. Mushtaq, S.-F. Su, "Environmental sound classification using a regularized deep convolutional neural network with data augmentation," *Applied Acoustics*, vol. 167, p. 107389, 2020, doi:10.9781/ijimai.2021.01.003.
[35] S. Mertes, A. Baird, D. Schiller, B. W. Schuller, E. André, "An evolutionary-based generative approach for audio data augmentation," in *2020 IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP)*, 2020, pp. 1–6, IEEE.
[36] B. Zoph, E. D. Cubuk, G. Ghiasi, T.-Y. Lin, J. Shlens, Q. V. Le, "Learning data augmentation strategies for object detection," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, 2020, pp. 566–583, Springer.
[37] L. Nanni, Y. M. Costa, R. L. Aguiar, R. B. Mangolin, S. Brahnam, C. N. Silla, "Ensemble of convolutional neural networks to improve animal audio classification," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2020, pp. 1–14, 2020.
[38] K. Choi, G. Fazekas, K. Cho, M. Sandler, "A tutorial on deep learning for music information retrieval," *arXiv preprint arXiv:1709.04396*, 2017.
[39] M. S. Hossain, G. Muhammad, "Emotion recognition using deep learning approach from audio–visual emotional big data," *Information Fusion*, vol. 49, pp. 69– 78, 2019.
[40] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S.-Y. Chang, T. Sainath, "Deep learning for audio signal processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 206–219, 2019.
[41] D. de Benito-Gorron, A. Lozano-Diez, D. T. Toledano, J. Gonzalez-Rodriguez, "Exploring convolutional, recurrent, and hybrid deep neural networks for speech and music detection in a large audio dataset," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2019, no. 1, pp. 1–18, 2019.
[42] P. Wu, j. Liu, Y. Shi, Y. Sun, F. Shao, Z. Wu, Z. Yang, "Not only look, but also listen: Learning multimodal violence detection under weak supervision," in *European Conference on Computer Vision (ECCV)*, 2020.
[43] W.-F. Pang, Q.-H. He, Y.-j. Hu, Y.-X. Li, "Violence detection in videos based on fusing visual and audio information," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 2260–2264, IEEE.
[44] R.-R. O. S. Lab, "Ntu cctv-fights dataset." https://rose1.ntu.edu.sg/dataset/cctvFights/, 2019. Access 03/02/2023.
[45] M. Perez, A. C. Kot, A. Rocha, "Detection of real-world fights in surveillance videos," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 2662–2666, IEEE.
[46] M. Schedi, M. Sjöberg, I. Mironică, B. Ionescu, V. L. Quang, Y.-G. Jiang, C.-H. Demarty, "Vsd2014: A dataset for violent scenes detection in hollywood movies and web videos," in *2015 13th International Workshop on Content-Based Multimedia Indexing (CBMI)*, 2015, pp. 1–6, IEEE.
[47] M. M. Soliman, M. H. Kamal, M. A. E.-M. Nashed, Y. M. Mostafa, B. S. Chawky, D. Khattab, "Violence recognition from videos using deep learning techniques," in *2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS)*, 2019, pp. 80–85, IEEE.
[48] S. Tang, S. Yuan, Y. Zhu, "Data preprocessing techniques in convolutional neural network based on fault diagnosis towards rotating machinery," *IEEE Access*, vol. 8, pp. 149487–149496, 2020.
[49] C. Shorten, T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of big data*, vol. 6, no. 1, pp. 1–48, 2019.
[50] K. O'Shea, R. Nash, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458*, 2015.
[51] M. Tan, Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, 2019, pp. 6105–6114, PMLR.
[52] D. Sinha, M. El-Sharkawy, "Thin mobilenet: An enhanced mobilenet architecture," in *2019 IEEE 10th annual ubiquitous computing, electronics & mobile communication conference (UEMCON)*, 2019, pp. 0280–0285, IEEE.
[53] J. P. Gujjar, H. P. Kumar, N. N. Chiplunkar, "Image classification and prediction using transfer learning in colab notebook," *Global Transitions Proceedings*, vol. 2, no. 2, pp. 382–385, 2021.
[54] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.
[55] M. Huh, P. Agrawal, A. A. Efros, "What makes imagenet good for transfer learning?," *arXiv preprint arXiv:1608.08614*, 2016.
[56] D. M. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," *arXiv:2010.16061*, 2020, doi: https://doi.org/10.48550/arXiv.2010.16061.

Dalila Durães

Dalila Durães is a Professor of Computer Science at the Department of Informatics, the School of Engineering, the University of Minho (Portugal), and a researcher at the ALGORITMI Centre and LASI Intelligent Systems Associate Laboratory. She is chair of the IEEE in Computational Intelligence Society, Portugal Chapter and a member of the Portuguese Association for Artificial Intelligence (APPIA) social corporate since January 2020 (Secretary of the General Assembly). Dalila is the author of more than 60 scientific publications in international peer-reviewed journals, book chapters, books, and conferences. She is also a member of the editorial board of several international journals. During the past few years, she has served as an expert/reviewer for several conferences and journals. She was also a supervisor of master's and doctoral students and was a member of the jury of several doctoral and master's theses.

Bruno Veloso

Bruno Veloso is 26 years old and he live in Portugal. He studied "Science and Technology" before university, in Portugal, Viana do Castelo, then he enrolled at the University of Minho (Portugal, Braga) in 2015, in the degree of Computer Science. After a year, he decided to change his major to Integrated Master's in Computer Engineering, in which he completed the bachelor's degree in 2020 and he currently finishing the last year of the master's degree (2023), in which he is writing a dissertation related to deep learning. During the master's program, he chose the study profiles of "Data Science" and "Intelligent Systems". With these two profiles, he realized that he would like to pursue a career in machine learning because it is something that belongs to the future and captivates me. Additionally, he is also participating in a research scholarship, in collaboration with two universities and a company, which aims to use machine learning to investigate the success/failure of students. In the future, he would like to find a job in the field of machine learning.

Paulo Novais

Paulo Novais is a Full Professor of Computer Science at the Department of Informatics, the School of Engineering, the University of Minho (Portugal) and a researcher at the ALGORITMI Centre in which he is the leader of the research group ISLab - Synthetic Intelligence lab, and the coordinator of the Portuguese Intelligent Systems Associate Laboratory (LASI). His main research aim is to make systems a little smarter, intelligent and also reliable. He is the co-author of over 400 book chapters, journal papers, conference and workshop papers and books.