Emotion detection using deep learning on spectrogram images of the electroencephalogram

Mauro Mezzini

Department of Education, Roma Tre University, 00185 Rome, Italy

Abstract

In this paper it is proposed a deep learning technique for emotion classification using spectrogram images produced from the related electroencephalogram (EEG) signals. The idea is to use the classical, state-of-the-art deep learning techniques for image recognition applied to the spectrogram of the EEG signal. The goal is to detect and recognize the level of valence, arousal, dominance, and likability. Extensive experiments are carried on with different convolutional neural network architectures on the publicly available DEAP dataset in order to find the best possible model with respect to the accuracy of the prediction. A new data augmentation technique on EEG signals has been experimented with and validated. The model has been developed and evaluated by taking a random permutation of the dataset and partitioning it in 80% training, 10% validation, and 10% test. Doing so allowed us to assess the model's ability to recognize an individual's emotion based on the EEG signals of other individuals. Results show that the models can learn and detect emotions with high accuracy at the same level as the state of the art of analogous models already presented in the literature.

Keywords

EEG, Deep learning, CNN, Emotion recognition

1. Introduction

The use of physiological signals for emotion recognition and detection has been the subject of several studies in the recent past. An emotion can be detected and recognized by a self-assessment of the subject or can be inferred from physiological signals. Subjective self-reports, while valuable, could raise validity issues. On the other hand, physiological signals can be used to evaluate emotions objectively. The ability to objectively assess the emotional state of a person could be employed in a wide variety of fields like (but not limited to) education, medicine, and entertainment with applications ranging from medical diagnostic, robotic and automatic assistance of motion impaired persons, validation of recommendation algorithms and, generally speaking, for brain-computer interface (BCI).

An emotion can be defined as a complex psychological state that involves three distinct components: a subjective experience, a physiological response, and a behavioral or expressive response. Various descriptions of these states have been proposed: In one approach, a discrete categorization of emotions is devised as proposed by Ekman et al. [1] and by Plutchik [2] where

Italian Workshop on Artificial Intelligence for Human-Machine Interaction (AIxHMI 2023), November 06, 2023, Rome, Italy

mauro.mezzini@uniroma3.it (M. Mezzini)
0000-0002-5308-0097 (M. Mezzini)

CEUR Workshop Proceedings (CEUR-WS.org)

both proposed several universal basic emotions such as anger, fear, sadness, disgust, surprise, and joy. In this perspective, the emotions could be represented as a graph whose nodes are the emotions and the edges are the links that connect similar emotions, and various algorithms can be employed to explore the properties such graphs[3, 4].

A second perspective of describing emotions is called *dimensional approach* (see Figure 1). In this case, the emotions are described along three dimensions: *valence*, *arousal*, and *dominance*. Valence goes from pleasant (very positive) to unpleasant (very negative). Arousal indicates how much excitement is involved in the feeling of the emotion: from very excited and active, to very calm, bored, and/or sleepy. Dominance indicates the degree of control one feels: from being helpless and weak or without control to feeling in control of everything. We will refer to them in the following as *emotion dimensions*. The dimensional approach allows us to assess emotions quantitatively, and it is a common approach adopted in much of the literature on emotion detection and, therefore, is the approach used in this paper. There can be many ways to asses an emotional dimension: the subject self-assessment, physiological signals, and external signals like facial expression. Physiological signals may range from Galvanic Skin Resistance, Hearth Beat Rate, Respiration Rate, and Electroencephalogram (EEG). The last method attracted much interest from the researchers because it directly obtains signals from where the emotions start to form -the brain- (although the paper [5] hypothesized that emotion responses originates in the heart) and is relatively noninvasive and simple to obtain.

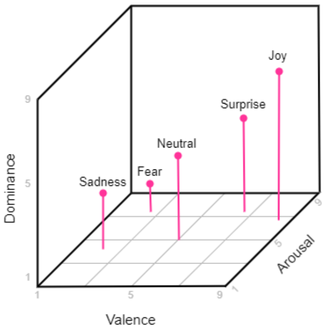

Figure 1: A 3D representation of the emotion dimensional model.

In this work, we developed deep learning models using the DEAP EEG recording dataset [6]. Our approach consists of pre-processing the EEG signal to obtain the spectrogram image for each EEG channel. The research hypothesis is that the spectrogram, treated as an image, can be used to effectively detect the level of emotions. The experiments where conducted in order to determine the best CNN model and in order to select the best hyperparameters to apply to the model.

The spectrogram is a 2-dimensional array S obtained from the EEG signal; each column i of S represents the signal's Power Spectral Densities (PSD) in the time interval i. In other words, $S(j, i)$ is the power of the sinusoid of the signal at frequency j in the time interval i. We obtain a spectrogram image or a 3-dimensional tensor by stacking all EEG channels together. Then, we use the state-of-the-art convolutional neural network (CNN) architectures [7, 8], commonly employed for image detection, to recognize the emotion dimensions described above. In this paper, we adopt the standard methodology of fine-tuning the model on the validation set and then evaluate it on a test set. We take a random permutation of the dataset and partition it into three sets: training, validation, and testing. We made extensive experiments using many combinations of hyperparameters and configurations. We provide a detailed description of the experiments done, accompanied by detailed reports of data results. We also provide the running code on a Colaboratory notebook. The novelty of this work is in the fact that this is (to the best of my knowledge) the first paper that uses the state-of-the-art ResNet architecture in conjunction to the spectrogram images. This approach will consider in a single image the spatial, temporal and the frequency domain feature of the EEG signal thus allowing to employ the power of ResNet architecture in image detection and recognition. Furthermore this allow us to avoid the heavy prepossessing analysis to found (if any) the most relevant EEG channel which gives the most significant response to the emotional stimulus as described in [9].

The work is organized as follows. In Section 2, we briefly cover the most significant related literature. In Section 3, we describe the dataset and the pre-processing algorithms we employed to obtain the tensors used as input to the models. In Section 4, we report the experiments' results and the models' evaluation. In Section 5, we give final remarks and directions for future works.

2. Related works

Emotion detection and recognition have been the subject of several studies in literature [10, 11]. While many use classical machine learning techniques, relatively few use modern deep learning architectures [12]. In [6], a dataset, called DEAP, of EEG signals recorded from 32 participants watching music videos is presented. They obtained a total of 1280 recordings. They computed the PSD of the signal over the trial length. Then, they took the average of the PSD over four frequency bands: theta (4-7 Hz), alpha (8-13 Hz), beta (14-29 Hz), and gamma (30-47 Hz). They determined that positive and negative correlations exist between the band's strength detected in some of the electrodes and the intensity of valence and arousal. They developed a Gaussian naive Bayes model reaching an accuracy of 0.620 on arousal, 0.576 on valence, and 0.554 on liking. In [13], another dataset, AMIGOS, is created similar to the DEAP dataset. In this case, the number of participants was 16, and the number of short videos they watched was 40. They also developed a Gaussian naive Bayes model, which attained an accuracy of 0.576 on valence and 0.592 on arousal by using the EEG signal alone. In [14], a hybrid 1-dimensional CCN + LSTM model is developed for detecting valence and arousal. They make the first experiment by training the model on the data of each single participant. Part of the data of a single participant was used as a training set and the other as a test set. For example, using the DEAP dataset, they took, out of 40 videos, 32 for the training set and the remaining 8 videos as a test set. They did a similar experiment on three datasets: DEAP, DREAMER [15], and DASP [16]. The final accuracy was calculated by taking the average of all the results. They reached an accuracy of 63.02% on valence and 67.34% on arousal when using exclusively the EEG signal. However, it is not explained in detail what kind of features are given in input to the model. In [12], four distinct datasets have been used: DEAP, AMIGOS, DREAMER, and MAHNOB-HCI [17]. They first compute the PSD of each EEG channel of three EEG bands, namely, theta (4-7 Hz), alpha

(7-13 Hz), and beta (13-30 Hz). The PSD was then averaged over the trial length. Then, they plotted the power spectrum heat maps for the three EEG bands using bicubic interpolation to obtain a 2D image according to the standard EEG 10-20 system. These images contain the topographical information for the three frequency bands and are treated as standard images. Their idea is to feed these images to a state-of-the-art convolutional image detector. They used the well-known (albeit relatively old) VGG-16 architecture [8] pretrained on Imagenet [18]. They employed the so-called leave-one-subject-out evaluation and averaged the results over all participants. They obtained, by using the EEG signal alone, the accuracy of 71.09% on valence, 72.58% on arousal, and 74.77% on liking.

It is worth noting that the vast majority (if not all) of the works published in literature employ a 10-fold cross-validation scheme (see [19]). In this work, we also used a more classical evaluation scheme based on the partition of the dataset into three subsets: training, validation, and test sets. In the evaluation scheme containing a validation set, the hyperparameters of the model, the principal component analysis, and all the design choices about the model's effectiveness are chosen using the validation set. During the fine-tuning of the hyperparameters, the best model is chosen based on its performance on the validation set. Once the choice is made, it is tested on a test set to assess the model's performance and objectively evaluate its effectiveness. It is also worth noting that, since the datasets present in literature albeit contain a fair amount of data, their quantity is orders of magnitude less than the quantity of data contained in the analogous dataset commonly employed in the field of image detection [20] and object detection [21, 22] where billions of (labeled) images are easily provided. Therefore, the performances of the deep learning models developed in the literature for emotion detection from EEG signals are less impressive than the analogous performance in image detection and recognition.

3. Methods

3.1. Dataset description and preprocessing

We built and tested the prediction models using one of the best-known online datasets, the DEAP dataset. This dataset contains the EEG signals of 32 individuals that were collected while the subjects watched and listened to music videos taken from YouTube. Each subject was invited to view 40 one-minute videos and then asked to express her emotions on the dimensional model described above. In the following we refer to the EEG signals of one participant watching one video as an *experiment* or *trial*. Additionally, a parameter called *likability* was used to quantify how much the participant liked the stimulus. In the following we treat likability as another emotion dimension. For each emotion dimension, the participant was asked to rate its intensity on a continuous scale between 1 and 9, where 1 stands for minimum intensity, and 9 for maximum intensity. We transformed the continuous scale of each emotion dimension e into a binary value $b(e) \in \{0, 1\}$ so that $b(e) = 0$ if $e < 5$, $b(e) = 1$ if $e \geq 5$.

The EEG signal consists of 32 channels, each corresponding to an electrode that measures the difference in electric potential in the skull area where it is positioned.

We used for all the experimentation ResNet101 and VGG11 architectures, referred in the following as the CNN architectures. The input tensor T to the CNN architecture has three

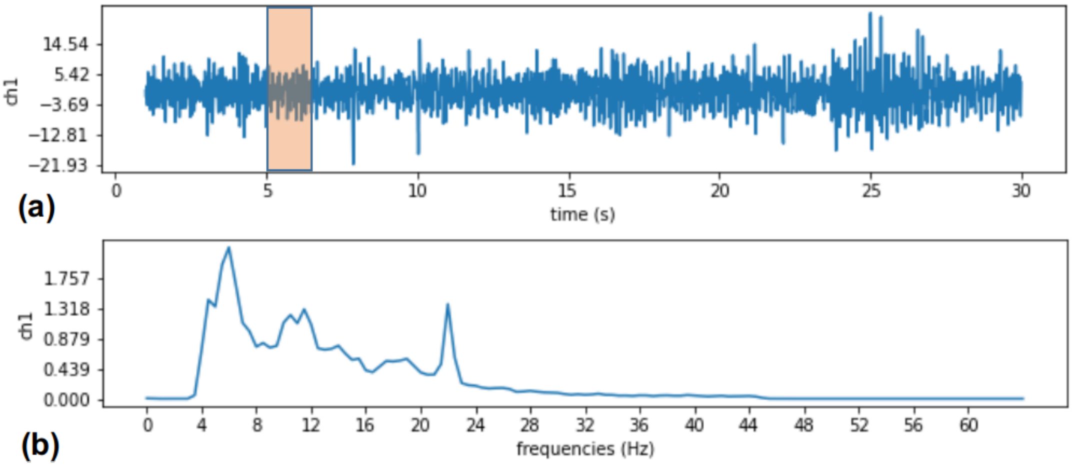

Figure 2: (a) The EEG signal of one channel and (b) the PSD corresponding to the orange windows in the EEG signal (a).

dimensions denoted by the *depth* C, the *height* H and the *width* W. Each data point of the input tensor will be denoted as $T(c, f, t)$. In all subsequent configurations the width of the input tensor is the time.

3.1.1. Spectrogram computation

We divided the raw signal of each of the 32 EEG channels in 34 overlapping segments, called *time segments*, each of which is composed by 256 data points and the amount of overlapping is 32 data points[1]. For each time segment the Discrete Fourier Transform (DFT) is computed with a Tukey window and from it we obtained the PSD (for more details on the definition of PSD and spectrogram please refer to [23, 24, 25]). Since each window is composed by 256 data points, the DFT returns a total of 129 different frequencies. These frequencies are between 0 and 64Hz, since the signal available in the DEAP dataset has been (down)sampled at 128Hz. Since the DEAP dataset's preprocessed data have a bandpass frequency filter from 4.0-45.0Hz, we excluded all the frequencies outside such interval (which are 0), eventually obtain $2(45 - 4) + 1 = 83$ different frequencies. Figure 2 shows: in (a) an example of the original signal and in (b) the PSD of one window.

3.1.2. Hyperparameters configuration

Therefore in its basic configuration the input tensor has dimensions $C = 32$, $H = 83$ and $W = 34$ (see Figure 3 for a heat map of the channels' spectrograms stacked together). We called it the *normal*, configuration. However we developed other two configurations of the input tensor. The second configuration is obtained by the first one by subtracting to each time segment the PSDs of the first 3 seconds of recording, in which the subject has been exposed to a neutral

[1] The amount of overlapping has been chosen by following the best practices of the digital signal processing community.

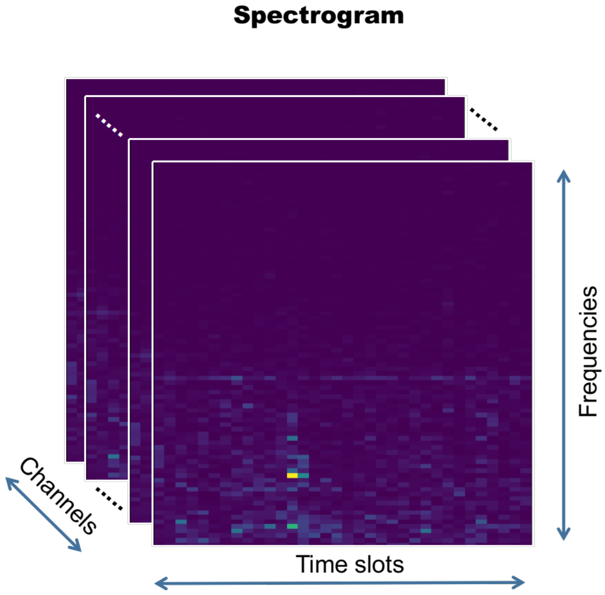

Figure 3: The color plot of the tensor used as input to the models.

stimulus. We called it as the *delta* configuration. A third configuration has been obtained in the following manner. Three frequency intervals have been chosen: theta (4-7 Hz), alpha (7-13 Hz) and beta (13-30 Hz) according to the paper [12]. Then the average of the frequencies in each interval have been computed obtaining an input tensor of height $H = 3$. Then, we swapped the height axis with the depth axis of the tensor, obtaining an input tensor with $C = 3$, $H = 32$ and $W = 34$. This configuration is called *reduced* and allowed us to use a CNN architecture in which the weights were pretrained on Imagenet since 3 is the depth of first convolutional layer of a pretrained model.

We developed three different way of data normalization. In the first, called *fine* normalization, for each time slot t, for each frequency f and for each EEG channel c we computed the maximum $m_1(c, f, t)$ for all the trials. We first compute

$$T_1(c, f, t) = \frac{T(c, f, t)}{m_1(c, f, t)}$$

and after this we computed the average $a_1(c, f, t)$ and the standard deviation $s_1(c, f, t)$ for all the trials using the values of T_1. Then each data point of the normalized tensor U_1 is given as

$$U_1(c, f, t) = \frac{T_1(c, f, t) - a_1(c, f, t)}{s_1(c, f, t)}$$

The second type of data normalization, called *coarse* normalization, is obtained by computing the maximum $m_2(c)$, along all the trials for each EEG channel c. In this case we first computed

$$T_2(c, f, t) = \frac{T(c, f, t)}{m_2(c)}$$

and after this we computed the average $a_2(c)$ and the standard deviation $s_2(c)$ along all the experiments for each EEG channel c using the values of T_2. Then each data point of the normalized tensor U_2 is obtained as

$$U_2(c,f,t) = \frac{T_2(c,f,t) - a_2(c)}{s_2(c)}$$

in the third type of data normalization, called *all* normalization, we computed the maximum m of all data points of the input tensor. Then,

$$T_3(c,f,t) = \frac{T(c,f,t)}{m}$$

and after this we computed the average a and the standard deviation s is of all data points of the tensor T_3. Then each data point of the normalized tensor U_3 is obtained as

$$U_3(c,f,t) = \frac{T_3(c,f,t) - a}{s}$$

Other type of configurations has been obtained by swapping the height axis (frequencies) and the depth axis (channels) of the first two configurations. We call these configurations as obtained by the *swap of frequencies*.

We employed a data augmentation scheme which consists in a random shift of the input tensor, after normalization but before the swap of frequencies, by an amount of up to 15% or 20% or 22% of the width of the tensor. This is motivated by the fact that a particular emotion may be felt by one subject at certain time while other subjects may feel the same emotion slightly before or slightly later. We used, in the backpropagation algorithm, the *cross entropy* (CE) loss function. Generally speaking, the CE gives a measure of the distance between two probability distributions p and q over the same domain $\mathscr{C} = \{1, 2 \ldots, K\}$, i.e. a set of classes. Its definition is

$$H(q,p) = \sum_{i=1}^{K} q(i) \log \frac{1}{p(i)} \tag{1}$$

Let $y = f(x)$ be the K-dimensional output of the CNN architecture f where x is the input tensor and K is the number of classes of the model. Let c_x be the ground truth (GT) class corresponding to x. Applying the softmax function to y we obtain a K-dimensional vector p_y such that $0 \leq p_y(i) \leq 1$ and $\sum_{i=1}^{K} p_y(i) = 1$. The predicted class $\hat{c}_x$, is the class for which $p_y(\hat{c}_x) \geq p_y(i)$ for $i \in \mathscr{C}$. If we interpret p_y as a probability function, the best model should produce a p_y that minimizes the CE between the GT probability distribution $q_y(i)$. It is common to assume that $q_y(i) = 0$ if $i \neq c_x$ and $q_i(i) = 1$ if $i = c_x$. Suppose, for the sake of simplicity that $K = 2$, as in all of our models, and suppose that to the input x, with GT class c_x corresponds a probability $p_y(i) = 0.49$ if $i \neq c_x$ and $p_y(i) = 0.51$ if $i = c_x$. Clearly, in this case, the model produces a p_y which is quite distant to q_y. Nevertheless the model's predicted class is correct. In other words, *any* probability function p_y such that $p_y(c_x) > p_y(i)$ for all $i \neq c_x$ is a correct output of the model regardless its distance from the GT probability q_y. Given these observations, we applied in our experiments the *label smoothing* [26] scheme. In other words we relaxed the

Table 1
For each emotion dimension and each set the quantity and percentage of experiments labeled 1.

Emotion	**Train**		**Val**		**Test**	
	#	%	#	%	#	%
Valence	577	0.563	70	0.547	77	0.602
Arousal	585	0.571	90	0.703	79	0.617
Dominance	628	0.613	76	0.594	91	0.711
Liking	675	0.659	85	0.664	97	0.758

requirement that $q_y(i) = 1$ if $i = c_x$ and $q_y(i) = 0$ otherwise, and assume only that $q_y(c_x) > q_y(i)$ for all $i \neq c_x$. We tested the label smoothing scheme by choosing $q_y(c_x)$ to be equal to 0.75 or 0.9 or 1.0.

Counting all the possible combinations of hyperparameters we obtain a total of 864 different configurations, for each CNN architecture, denoted in the following as *grid points*.

3.2. Data and experimentation setup

We used Pytorch framework [27] for the CNN models and the package Scipy [28] for computing spectrograms and DFTs of the EEG signal. We made publicly available the software we developed in a Colaboratory notebook[2]. In order to asses which configuration gives out the best model we conducted a grid search on the 864 grid points described in the previous section. Both Resnet101 and VGG11 architectures were modified in order to accept in input a tensor that has depth not necessarily equal to 3, which is the fixed default in these architectures since they were originally developed for RGB image detection and recognition. Every model outputs a binary value which represents the high/low value of the corresponding emotion dimension.

We applied a backpropagation algorithm based on SGD, with a batch size of 32, momentum=0.9, weight_decay=0.0005. We employed a warm up scheme in which the initial learning rate is first decreased by a factor of 10^{-3} and then linearly increased in the first 32 epochs. After this warm up, the learning rate for all grid points without a pretrained model is 3.0×10^{-3} and for the pretrained models is 1.0×10^{-3}. In both cases and after warm up, the learning rate is decreased every 20 epochs by a factor of 0.85. The total number of epochs for all grid points is 440.

In order to test the effectiveness of the models we conducted a completely randomized experiments. To do so we computed a random permutation of the 1280 experiments of the dataset. Of the first 256 experiments of the permutation 128 has been used for the validation and the other 128 for test set. The remaining 1024 are used for the training set. In Table 1 are reported for each emotion dimension and each set the quantity and percentage of experiments labeled 1.

We also conducted several k-fold analysis. In the first one we computed a random permutation of the 1280 trials and used a 10-fold analysis by considering 128 trials for the test and the other for the training. Furthermore we conducted a 32-fold analysis by considering 40 trials of a

[2] https://colab.research.google.com/drive/1g3uV0uI7cFlOOew3rzfRz6BzsINgkgRO?usp=sharing

single participant as a test and the others as a training. This last experiment has been conducted because most of the works in the literature report using this kind of evaluation analysis.

The experiment setup is as follow. We first conducted a grid search over all the grid points described above using only the ResNet101 architecture. Based on the results of the grid search we selected the best hyperparameters. Then we train again both ResNet101 and VGG11 on these hyperparameters. During this last training we saved the models which achieved the best accuracy on the validation set in the last 340 epochs. We repeated this training for 30 times overall collecting 30 ResNet101 and 30 VGG11 models for each of the four emotion dimensions. We observed a lot of variability during the training phase even when the learning rate becomes very small and the model overfits the training set. For example, in Figure 4 we have the accuracy plot for both the training and validation set for the best grid point of the model predicting Arousal. This suggested utilizing an ensemble of the models. We tested an ensemble of 30 and 15 of these models all from the same architectures. We also used an ensemble of 60 and 30 models from both CNN architectures.

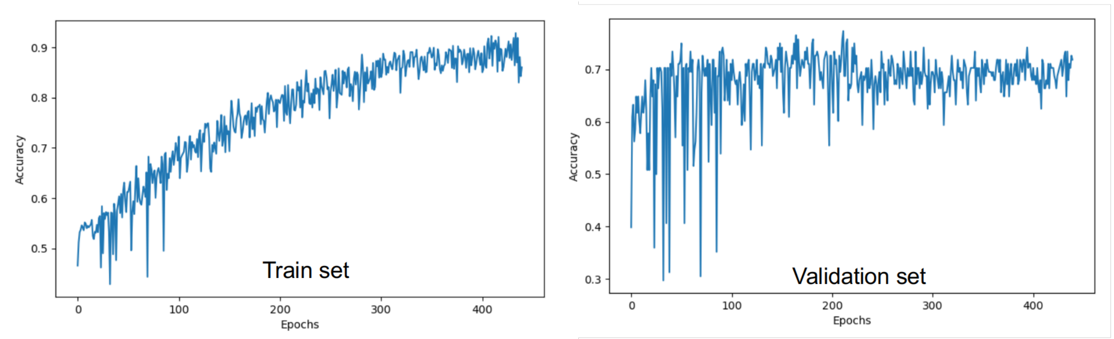

Figure 4: Plot of the accuracy of the training and validation of the best grid point on the Arousal.

4. Results

In all the grid points the training average accuracy is over 0.750 and can easily reach 0.999, a clear sign that the models overfits quite effectively. Furthermore we observed that even when the model overfits and the learning rate is very small, there is still quite a lot of variability in the accuracy on the validation set. For each different type of emotion dimension and for each grid point, we computed the average, standard deviation and the maximum of the accuracy over the last 340 epochs for the validation set. Eventually we selected the hyperparameters reported in Table 2.

In Table 3 we report the results of the ensemble tests. In the ensemble test we run a number n of models and the output is defined by the majority. The values reported are the accuracy obtained using different ensemble configurations. We tested single architecture (ResNet101 and VGG11) with 15 and 30 models ensemble. We also mixed 30 ResNet101 and 30 VGG11 (resp. 15). The row called *average* is the average of the accuracy of the four emotion dimensions. The row *Avg. single models* represents the average of the accuracy over all $n \in \{15, 30\}$ single models.

Table 2
The hyperparameters which were selected after the grid search. The column *average* is the average accuracy on validation set over the last 340 epochs. The column *max* is the maximum accuracy achieved by the model over the last 340 epochs during the training on the validation set.

emotion	pretr.	augment type	augment parameter	swap freq.	spectra type	normaliz.	smooth	average	max
valence	0	translate	0.2	1	normal	coarse	0.90	0.627	0.703
arousal	0	translate	0.22	0	normal	all	0.75	0.704	0.766
dominance	0	translate	0.22	1	normal	fine	1.00	0.703	0.742
liking	0	no augment.	0	1	normal	all	0.90	0.706	0.742

Table 3
The accuracy obtained using different ensemble configurations. Highlighted are the best results on the test set. The row *average* is the average accuracy of the four emotion dimensions. The row *Avg. single models* represents the average of the accuracy over all $n \in \{15, 30\}$ single models. The row *differences* is the difference in percentage between the two previous rows.

Emotion	Resnet101				VGG11				Mixed (ResNet + VGG)			
	30 models		15 models		30 models		15 models		30+30 models		15+15 models	
	Validation	Test	Validation	Test	Validation	Test	Validation	Test	Validation	Test	Validation	Test
Valence	0.664	0.641	0.672	0.656	0.711	0.680	0.688	0.633	0.719	0.672	0.695	**0.703**
Arousal	0.813	0.656	0.813	0.656	0.781	**0.695**	0.781	0.688	0.781	0.688	0.781	**0.695**
Dominance	0.703	**0.695**	0.719	0.688	0.734	0.680	0.703	0.656	0.727	0.664	0.750	0.680
Liking	0.758	0.672	0.750	0.641	0.664	0.672	0.711	0.672	0.734	**0.703**	0.727	0.695
Average	0.734	0.666	0.738	0.660	0.723	0.682	0.721	0.662	0.740	0.682	0.738	0.693
Avg. single models	0.700	0.634	0.699	0.628	0.695	0.643	0.693	0.643	0.698	0.638	0.697	0.640
Difference	3.42%	3.24%	3.95%	3.19%	2.79%	3.84%	2.72%	1.86%	4.20%	4.38%	4.12%	5.35%

Table 4
Accuracy of k-fold evaluations. Highlighted the best results.

Emotion	Rand. 128, ep.>100		Rand. 128, ep.>0		32 part., ep. >100		32 part., ep. >0	
	Resnet	VGG11	Resnet	VGG11	Resnet	VGG11	Resnet	VGG11
arousal	0.661	0.681	0.663	0.682	0.675	0.673	**0.715**	0.712
dominance	0.693	0.696	0.709	0.703	0.691	0.698	**0.732**	0.720
liking	0.659	0.659	0.698	0.691	0.704	0.671	**0.733**	0.716
valence	0.670	0.652	0.673	0.654	0.685	0.672	**0.710**	0.682
average	0.671	0.672	0.686	0.683	0.689	0.679	**0.722**	0.708

The row differences is the difference in percentage between the two previous rows. We can see that in all cases the ensemble of models improve the accuracy of both the validation and the test set. By mixing both architectures we obtain a gain of more than 4% in accuracy.

In Table 4 are reported the results of the k-fold experiments. In the k-fold experiment the dataset is partitioned in k disjoint parts. One part is the test set and the other $k - 1$ will comprise the training set. We did two kind of experiments. In the first one a random partition of the trials is determined; this partition is composed by 10 elements each of which contains 128 trials. In the second case the partition is composed by 32 elements each of which contains the 40 trials of one participant. In all cases we record the maximum accuracy of the model over the test set and averaged this number over all the tests. In the columns containing the label $ep. > 100$, we took the maximum accuracy on the test set restricted over the last 340 epochs of the training because after the epoch 100 the model is more stable. When the column contains the label $ep. > 0$ we took the maximum accuracy on the test set over all the epochs. We see that in the case when $k = 32$ participants and we take the maximum accuracy over all the epochs, we obtain the best accuracy results which are quite comparable to the best already achieved in literature.

5. Conclusion and future works

In this work we developed a new technique for emotion detection using the EEG signal. We stacked together the spectrogram of each EEG channel in order to obtain a 3-dimensional tensor which encloses frequencies, temporal and spatial information. In this way we transformed the emotion detection problem in an image recognition problem. Therefore we were able to use the state-of-the-art convolutional architectures which proved quite effective on image detection. Future works aims to apply the same methodology to the other dataset already available in literature. Another line of research is to integrate to the EEG signals other types of biological signals (ECG, GSR, etc.) since, as reported in literature [12], the combined use of different biological signals significantly increases the accuracy of the models. We built and tested the prediction models using one of the best-known online dataset, the DEAP dataset. However future works will consider to extend this approach to other datasets. One of the challenge in this field, if compared with the field of image detection, is that in the latter there exist datasets containing millions and even billions of images, while the datasets already available of EEG signals contain thousand or even less examples, collected from a few scores of participants. Moreover these datasets are not homogeneous and often collect different type of emotions. Therefore one of the possible future work is to collect a larger database of EEG signal in term of number of experiments and number of participants.

References

[1] P. Ekman, Basic emotions, in: T. Dalgleish, M. J. Powers (Eds.), Handbook of Cognition and Emotion, Wiley, 1999, pp. 4–5.

[2] R. Plutchik, The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice, American Scientist 89 (2001) 344–350. URL: http://www.jstor.org/stable/27857503.

[3] M. Mezzini, On the geodetic iteration number of the contour of a graph, Discrete Applied Mathematics 206 (2016) 211 – 214. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84959904664&doi=10.1016%2fj.dam.2016.02.012&partnerID=40&md5=cdbc7eb1427ce3d059cafbd89a16e548. doi:10.1016/j.dam.2016.02.012, cited by: 6; All Open Access, Bronze Open Access.

[4] M. Mezzini, M. Moscarini, The contour of a bridged graph is geodetic, Discrete Applied Mathematics 204 (2016) 213 – 215. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84948808102&doi=10.1016%2fj.dam.2015.10.007&partnerID=40&md5=2bdf9cf6c0d7bf22875690f2aae3f595. doi:10.1016/j.dam.2015.10.007, cited by: 6.

[5] D. Candia-Rivera, V. Catrambone, J. F. Thayer, C. Gentili, G. Valenza, Cardiac sympathetic-vagal activity initiates a functional brain–body response to emotional arousal, Proceedings of the National Academy of Sciences 119 (2022) e2119599119. URL: https://www.pnas.org/doi/abs/10.1073/pnas.2119599119. doi:10.1073/pnas.2119599119. arXiv:https://www.pnas.org/doi/pdf/10.1073/pnas.2119599119.

[6] S. Koelstra, C. Muhl, M. Soleymani, J.-S. Lee, A. Yazdani, T. Ebrahimi, T. Pun, A. Nijholt, I. Patras, Deap: A database for emotion analysis using physiological signals, IEEE Transactions on Affective Computing 3 (2012) 18–31. doi:10.1109/T-AFFC.2011.15.

[7] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition, in: Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR '16, IEEE, 2016, pp. 770–778. URL: http://ieeexplore.ieee.org/document/7780459. doi:10.1109/CVPR.2016.90.

[8] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations, 2015.

[9] S. Gannouni, A. Aledaily, K. Belwafi, H. Aboalsamh, Emotion detection using electroencephalography signals and a zero-time windowing-based epoch estimation and relevant electrode identification 11 (2021). doi:10.1038/s41598-021-86345-5, funding Information: The authors extend their appreciation to the Deanship of Scientific Research at King Saud University for funding this work through Research Group no. RG-1441-524. Publisher Copyright: © 2021, The Author(s).

[10] S. M. Alarcão, M. J. Fonseca, Emotions recognition using eeg signals: A survey, IEEE Transactions on Affective Computing 10 (2019) 374–393. doi:10.1109/TAFFC.2017.2714671.

[11] A. Craik, Y. He, J. L. Contreras-Vidal, Deep learning for electroencephalogram (eeg) classification tasks: a review, Journal of Neural Engineering 16 (2019) 031001. URL: https://dx.doi.org/10.1088/1741-2552/ab0ab5. doi:10.1088/1741-2552/ab0ab5.

[12] Siddharth, T. . Jung, T. J. Sejnowski, Utilizing deep learning towards multi-modal biosensing and vision-based affective computing, IEEE Transactions on Affective Computing 13 (2022) 96–107. URL: www.scopus.com, cited By :41.

[13] J. A. Miranda-Correa, M. K. Abadi, N. Sebe, I. Patras, Amigos: A dataset for affect, personality and mood research on individuals and groups, IEEE Trans. Affect. Comput. 12 (2021) 479–493. URL: https://doi.org/10.1109/TAFFC.2018.2884461. doi:10.1109/TAFFC.2018.2884461.

[14] R. Agarwal, M. Andujar, S. Canavan, Classification of emotions using eeg activity associated with different areas of the brain, Pattern Recognition Letters 162 (2022) 71–80. URL: https://www.sciencedirect.com/science/article/pii/S016786552200263X. doi:https://doi.org/10.1016/j.patrec.2022.08.018.

[15] S. Katsigiannis, N. Ramzan, Dreamer: A database for emotion recognition through eeg and ecg signals from wireless low-cost off-the-shelf devices, IEEE Journal of Biomedical and Health Informatics 22 (2018) 98–107. doi:10.1109/JBHI.2017.2688239.

[16] A. Baghdadi, Y. Aribi, R. Fourati, N. Halouani, P. Siarry, A. M. Alimi, Dasps database, 2021. URL: https://dx.doi.org/10.21227/barx-we60. doi:10.21227/barx-we60.

[17] M. Soleymani, J. Lichtenauer, T. Pun, M. Pantic, A multimodal database for affect recognition and implicit tagging, IEEE Transactions on Affective Computing 3 (2012) 42–55. doi:10.1109/T-AFFC.2011.25.

[18] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255. doi:10.1109/CVPR.2009.5206848.

[19] A. Topic, M. Russo, Emotion recognition based on eeg feature maps through deep learning network, Engineering Science and Technology, an International Journal (2021).

[20] P. Goyal, M. Caron, B. Lefaudeux, M. Xu, P. Wang, V. Pai, M. Singh, V. Liptchinsky, I. Misra, A. Joulin, P. Bojanowski, Self-supervised pretraining of visual features in the wild, CoRR abs/2103.01988 (2021). URL: https://arxiv.org/abs/2103.01988. arXiv:2103.01988.

[21] R. Balestriero, M. Ibrahim, V. Sobal, A. Morcos, S. Shekhar, T. Goldstein, F. Bordes, A. Bardes, G. Mialon, Y. Tian, A. Schwarzschild, A. G. Wilson, J. Geiping, Q. Garrido, P. Fernandez, A. Bar, H. Pirsiavash, Y. LeCun, M. Goldblum, A cookbook of self-supervised learning, 2023. arXiv:2304.12210.

[22] A. Ferrato, C. Limongelli, M. Mezzini, G. Sansonetti, Using deep learning for collecting data about museum visitor behavior, Applied Sciences (Switzerland) 12 (2022). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85122372298&doi=10.3390%2fapp12020533&partnerID=40&md5=34cbc3ee04afb84f4911d6172acac2f7. doi:10.3390/app12020533, cited by: 11; All Open Access, Gold Open Access, Green Open Access.

[23] M. X. Cohen, Analyzing neural time series data: theory and practice, MIT press, 2014.

[24] P. Welch, The use of fast fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms, IEEE Transactions on Audio and Electroacoustics 15 (1967) 70–73. doi:10.1109/TAU.1967.1161901.

[25] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, Numerical Recipes 3rd Edition: The Art of Scientific Computing, 3 ed., Cambridge University Press, USA, 2007.

[26] M. Mezzini, Empirical study on label smoothing in neural networks, Computer Science Research Notes 2802 (2018) 200–205. URL: www.scopus.com, cited By :1.

[27] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, Pytorch: An imperative style,

high-performance deep learning library, in: Advances in Neural Information Processing Systems 32, Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[28] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, SciPy 1.0 Contributors, SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, Nature Methods 17 (2020) 261–272. doi:10.1038/s41592-019-0686-2.