
CHALMERS
UNIVERSITY OF TECHNOLOGY

# Modelling of Auditory Salience by use of Acoustic Features, Deep Neural Networks and Brain Signal Analysis

Master's thesis in Applied Acoustics

JOEP WULMS

Department of Applied Acoustics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2022

Master's thesis 2022

# Modelling of Auditory Salience by use of Acoustic Features, Deep Neural Networks and Brain Signal Analysis

JOEP WULMS


Department of Applied Acoustics
Chalmers University of Technology
Gothenburg, Sweden 2022

**Modelling of Auditory Salience by use of Acoustic Features, Deep Neural Networks and Brain Signal Analysis**

JOEP WULMS



Supervisor: Clara Borelli, PhD student in Image and Sound Processing Lab (ISPL), Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano.
Second supervisor: Augusto Sarti, Full Professor in Music & Acoustic Engineering, Politenico di Milano.
Examiner: Jens Ahrens, Associate Professor in Applied Acoustics at the Department of Architecture and Civil Engineering, Chalmers University of Technology.

Master's Thesis 2022
Department of Applied Acoustics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2022

Modelling of Auditory Salience by use of Acoustic Features, Deep Neural Networks and Brain Signal Analysis
JOEP WULMS
Department of Applied Acoustics and Image and Sound Processing Lab
Chalmers University of Technology and Politecnico di Milano

# Abstract

Auditory salience is the property by which a sound stands out from its surrounding, a phenomenon that people's hearing system is dealing with at almost every moment of the day. It is the reason our attention alters from one sound source to another sound source, or instrument to instrument when listening to a music piece. The term salience is widely used in the field of perception and cognition to describe any feature of a stimulus that stands out from the rest for a variety of reasons. It can be influenced by emotional, motivational, or cognitive elements and it is not always linked to physical characteristics of the stimuli, like intensity, temporal- or frequency contrast. Salient sound events might be represented, for example, by your phone's message rings, a melody played on the piano or a loud passing car in quiet environment. In neuroscience the attention mechanism towards salience is split up in two parts. One is stimulus-driven attention and it is referred as bottom-up attention whereas cognitive-driven attention is known as top-down attention. Bottom-up attention in auditory salience is studied extensively, however top-down is less.
This thesis work explores a relative new approach to investigate the bottom-up and top-down attention towards auditory salience, by using deep neural network techniques and EEG brain signals analysis. Deep neural networks have been related to the human perceptual visual salience, early layers of the deep neural networks resemblances with physical features of an image (bottom-up attention), whereas the deeper latter layers with higher-level semantic properties (top-down attention). Therefore this project adopts a similar approach, focusing instead on auditory salience.
To assess auditory salience a computational model was built based on three frameworks. One frameworks deals with extracting acoustic features from song audio using the Python Librosa library. The second framework is based on the existing pre-trained convolutional neural network VGGish to resemble the deep neural network. And lastly, activity of EEG brain signals was analysed and represented through compact descriptors. We then computed and analysed the correlation between each pair of the three framework's outputs.
All three frameworks utilized data from the public open OpenMIIR database, which contains 12 song stimuli, 10 participants and EEG recordings while those songs were played to the subjects.
The result of the correlations between the three frameworks showed various correlations values in a patterned manner for the different network layers, which is in line with prior expectations. However, the overall correlation values are suspiciously high, which should therefore be interpreted loosely. Nevertheless, a preliminary computational model was developed, which with appropriate modifications could be used for further studies.

Keywords: auditory, salience, music, perception, acoustic, features, bottom-up, top-down, attention, CNN, EEG.

# Acknowledgements

I could not have undertaken this journey without the support of the Image and Sound Processing Lab in Polimi and the Division of Applied Acoustics in Chalmers, they made it possible to carry out my topic of interested for this thesis work. I would like to thank my supervisors Clara Borrelli and Augusto Sarti, without their assistance and dedicated involvement in every step throughout the process, this paper would have never been accomplished. I would like to thank you very much for your support and understanding over the last year. I wish to express my gratitude to my examiner Jens Ahrens for approving the topic and guarding my progress throughout. In addition I would like to thank Alessandra Calcagno of the Bioengineering department and Sebastian Gonzalez of ISPL for sharing there knowledge of EEG signals and how to pre-process them.

Thanks to my family and friends in The Netherlands, Sweden and Italy that supported all through my thesis work and studies. Your help, encouragement and kind words are unforgiving.

And last but absolutely not the least, I am extremely grateful to Chalmers University of Technology and Politecnico di Milano for accepting and welcoming me to study at their universities in the beautiful cities Gothenburg in Sweden and Milano in Italy. That includes of course all the great professors, assistants, staff and students that I met during my two years studying abroad. It was an amazing passage in my life and many happy memories will remain with me for good. A great proost - cheers - skål - Prost - saluti to all of you who have become a part of my life.

Joep Wulms - Gothenburg, December 2022

# Contents

# List of Figures

# List of Tables

# 1
# Introduction

In our daily lives, man is exposed to a multitude of various sounds simultaneously, each with different wave characteristics, context and directivity. Our auditory system is tasked with sorting the sound events, attending and identifying sound objects of interest, while ignoring irrelevant ambient backgrounds. A crucial component in analyzing acoustic situations is the role of attention, where sensory and cognitive resources are influential. These two resources, sensory and cognitive in auditory system context, have been studied intensively in the fields of neuroscience, psychology and human behaviour studies over the last decades [42]. However, a clear understanding of how these two aspects are processed in the brain is not yet accomplished.

An interesting phenomenon that covers the whole topic of auditory attention, with sensory and cognitive aspects included, is auditory salience. Salience describes by which an object stands out from a scene. Think of a loud car that is passing by or a phone is ringing. The sound source, the car or phone, to some degree catches your attention because the scene where the sound is present is contrasting itself. If the same sound events were present in an environment were multiple of the same sounds are active, let say highway or busy call center, the particular sound events would get less your attention.

The same phenomenon of auditory salience occurs in music, when listening to a piece where several instruments and vocals are present, one will be more prominent than the other. The reason that a particular instrument or melody gets your attention is again related to the sensory and cognitive resources. An example for a sensory based source is a highly tonal sound from an electronic keyboard. The wave characteristics of a tonal tune is time- and frequency-invariant, it is perceived as a very clear pure tone over a period of time. When this tone of the keyboard key is played together with sounds that are time and frequency variant sounds, for example a snare drum or maracas, typical percussion instruments which have transient characteristics, the tonal sound will stand out from the percussion sounds. Merely because the wave characteristics for both sounds are very different. In case of the melody of a song, it could catch your attention not because of its physical properties but because you recognize the melody from your past memories to it. This is an example where instead of sensory aspect of the auditory system, the cognitive aspect plays its role. The two different aspects of auditory attention is known in the literature as bottom-up and top-down attention. Whereas bottom-up attention is based on low-level features of a sound, i.e., in music a drum playing the rhythm and top-down attention relies on high-level features, i.e., chord progressions or music

genre [37].

Salience is present for each of five our senses, hearing, sight, tactile, olfactory and gustatory. The sense that is studied most extensively is sight and it is the bases of the scientific field called computer vision. This field addresses how computers can gain high-level understanding from digital images or videos, a trending business since rise of smartphones, social media platforms and surveillance applications [12]. The tool that is most commonly used in computer vision to investigate salience is the eye-tracking device. Through eye-tracking scientists have studied human behaviour when a participant was presented with certain images or video clips [50]. By tracking the eye movement the scientist were able to learn bottom-up aspects as well as top-down attention process of the visual system of the brain. With the information gathered from these experiments, researchers were able to create so called salience maps. These are heatmaps where each color corresponds to a salience level for different regions of an image or video. A practical example of the use of salience maps, such maps can be used to improve a website layout to increase your products sells, or to create a effective commercial on television. A second possible application is video's compression. By knowing which regions are of less interest per frame, the pixel density can be reduced of this area, leading to a smaller storage size [24].

Auditory salience is studied less extensively compared to visual salience. However, there are few computational models invented over the last 15 years that aims at defining auditory salience maps [62] [36]. Instead of an image or video, a spectrogram is utilized to mark the regions of salience. Although, these models are mostly based on bottom-up attention and exclude the top-level attention. The reason top-level attention is not included is because, in contrast to visual salience, eye-tracking devices cannot be applied to investigate cognitive behaviour of the auditory system. State-of-the-art research therefore started to investigate in brain activity. This new active research area is known as brain decoding, defined as the process of mapping brain activities to the stimuli that generated them. Recently applied by cognitive sciences studies to research top-level processes, and therefore an opportunity to study auditory salience for both bottom-up as well top-down attention [67].

Before heading to the actual project of this thesis work, one more topic will be introduced now, which is the most rapid growing technology of the last 5 to 10 years and greatly invested by the largest companies such as Google, Meta, Microsoft, Apple [26]. The technology is artificial intelligence (AI), with its sub fields machine learning (ML) and deep learning (DL). The term artificial intelligence refers to the simulation of human intelligence processes by machines, especially computer systems. Its sub fields ML/DL concentrates on learning patterns from vast amount of data, supervised or unsupervised. Since the increase of computational power and available data of the last years, these algorithms are able to perform better classification tasks, game plays such as chess, and simulations of physics than humans can do [29]. Relating to this work, the latest studies have shown that deep neural networks are somehow related to the human perceptual salience, given that these network tries to predict high semantic properties of music [45].

With the current findings in the field of auditory salience, bottom-up and top-down attention, brain decoding and machine learning and deep learning techniques, this thesis work attempts to develop a possible computational model that predicts both bottom-up and top-down attention in the context of an auditory system. This model is a deep neural network, which are known with the ability to predict high semantic properties of text and speech [68] [41]. The type of sound that this work will focus on is music, therefore it is interesting to see if these deep neural networks are can predict high semantic properties of music as well, this could be then related to the human perceptual salience. The choice for music is based on two reasons, first is that there are little studies out that analyzed auditory salience in music, mostly natural soundscapes or speech are used. Secondly, this work is in collaboration with the department of Image and Sound Processing Lab of Politecnico di Milano, where one of their main research areas is Music Information Retrieval. With their knowledge and expertise a good foundation is already there to start with the thesis project.

The relevance of this computational model is to get a better insight of how the auditory system operates, and to predict whether a certain elements in a music excerpt got salient elements. With this knowledge it is possible to improve the listener experience while listening to music, while it be in a concert hall, watching a movie in the cinema or listening at home via your speakers. By knowing which elements in music gets the listeners attention, music could be created in a way it it plays with the listener experience with salient and non-salient objects. It could even be combined with the visual aspect, for example in games, movies or commercials.

This 1 Introduction Chapter will continue with the sections 1.1 Background, 1.2 Related work and state-of-the-art, 1.3 Goal study and finally an 1.4 Thesis Outline.

## 1.1 Background

The Background Section will elaborate on some of the terms stated in the 1 Introduction Section by placing them in a context. This will result in a better understanding of the phenomenons itself.

### 1.1.1 Sound, two definitions

A famous philosophical thought experiment exposes the two definitions of sound, and summarizes the paragraph: "If a tree falls in a forest and no one is around to hear it, does it make a sound?" Here the questions that raises are regarding the observation and perception of sound.

Sound can be described by two separate definitions, one is in the nature of acoustics and the other in the nature of auditory or hearing. In the acoustic nature, sound is a physical phenomenon, vibrations of air, which can be measured by instruments other than our ears. Here different attributes of the waves can be observed, such as the wave characteristics which includes frequency, amplitude and phase. Relevant

entities for us humans is the capacity of our auditory system, the frequency range of hearing from 20 Hz to roughly 20 kHz, in means of intensity (amplitude or sound level) the scale is up from 10 dB to 140 dB. The second definition of sound is our perception of air vibrations. Perception (seeing or hearing) is our interpretation of sensory information coming from our senses, such as our eyes and ears, and tries us to represent and understand our environment. For this definition of sound it is not possible to measure the physical attributes of sound, since it is not a physical thing. However, us human share more or less the same experience of sound, and it can give descriptions to it, i.e,. how loud or sharp a sound sounds.

### 1.1.2 Bottom-up and top-down attention

The human is exposed to a multitude of sounds simultaneously during the entire day. Luckily, the brain is not processing each of these sounds with equal intensity. This is due to the fact that the brain capacity is limited in process-power at a given time [40]. An average human brain consist of 86 billion neurons interconnected by synapses [28]. Apart from the hearing sense, the brain needs to process as well the sight, tactile, olfactory and gustatory senses, which are as well continuously bombarded with a multitude of sensory inputs. A reduction of sensory impressions processes would therefor result in lower energy demands. But because the need of filtering, the brain has to decide which sounds are of relevance and which are not. The decision of attention for the different incoming sounds is a process of selection, in the literate known as selective attention (Broadbent, 1958 [18], (Deutsch , 1963 [14]) (Treisman, 1960 [13]). Among these conceptual schemes of selective attention, the debate is whether the moment of selection occurs in an early or late stage in the auditory system. In practice this means if irrelevant stimuli are filtered, meaning totally discarded, or just attenuated [18][13]. The different theories on selective attention have been changed over the years and for the sake of this thesis these different paradigms are not further analysed. In any case, the concept of bottom-up and top-down attention, strictly related to selective attention, will be elaborated. To do so, a few questions regarding relevance and irrelevance sounds are stated and answered in the following section.

When an object is relevant and gets our attention? Does it depend on the physical characteristics of the object itself? Or does it depends in what context this object is observed? What context means in this sense?

To answer these questions psychologists, neuro-scientists and scientist in the field of computer vision and audition have done relevant research on salience and selective attention. From these studies it is possible to conclude that our attention towards a stimulus is directed/driven by two separated processes. One process is a stimulus-driven or bottom-up process, here the physical attributes of a object matters to catch our attention. This bottom-up attention process is involuntary and, no matter our state, the focus will go towards that object. An example of such a object is an alarm going off, a loud explosion at the street or your phone's messages

alert ping. These sound objects have certain physical characteristics that draw our attention immediately and cannot be ignored. They are based on abrupt changes, transitions and abnormalities in the stream of sound events. The stimulus-driven attention is referred as a low-level process, since it does not required complex processes but rather a fast process, with other words, you react almost immediately to an unexpected event. In the area of music, features as pitch, timbre, loudness, reverberation, spatial location and tone duration are the low-level components.

The counter process that controls selective attention is driven by intentional attention, described in the literature as top-down attention. Top-down attention is motivated by the cognitive functions and perceptual capability of the human which entails a high-level process. Top-down attention is influenced by factors as intention, emotions and prior experience towards a stimulus [35]. The human perception creates meaning and understanding towards music and language. Elements in music such as consonance and dissonance tones could lead to pleasant and unpleasant feelings, this by placing them in a certain consecutive manner where the listener gets moved by the sound. This is not caused by the low-level components of the sounds itself, but our prior experience with music. The sounds are related with our emotions and play then a role with our feelings. In language words are a combination of successive vowels and consonants, low-level features, but become meaningful when the combination is known. The vowels and consonants become a word with semantics. Several studies concludes that the main difference is that top-down guided selection processes are slower processed in the brain compared to bottom-up mechanism [8].

As what can be understood from bottom-up and top-down attention with the given information above, is that they are closely related to each other and that they both play a role of the human perception. Bottom-up attention is more related to the physical aspect of a sound, which is relative easily measurable. While looking at measuring top-down attention, this is extremely difficult, since the process happens within the brain. Therefore new techniques are developed to get information on how top-down attention works, hence this study focus on one of these new techniques, deep neural networks.

## 1.2 Related work and state-of-the-art

In this section we present the relevant studies that have attempted to develop a computational model of salience. Starting with the first ever design model by Itti and Koch, up to the state-of-the-art models that include brain decoding and EEG signals.

### 1.2.1 Salience maps

The earliest work that attempted to build a computational model of salience was focused on visual attention and it is from L. Itti and C. Koch [39] with the title *A Model of Saliency-Based Visual Attention for Rapid Scene Analysis*. Their approach was based on visual features such as colors, intensity and orientation of images. The

model represents only bottom-up saliency and does not include top-down attention. The model takes an 2D image as input and builds an saliency map based on the given features and image. The model is presented in Figure 1.1. A saliency map is a layer that highlights salient areas within an image. The first step in the model is to extract the described visual features, in parallel, along various scales. Subsquently the scales were then compared using a center-surround mechanism to obtain feature maps, here the *courser* scale is the surround and the *finer* scale the center. These feature maps are then combined into a 2D saliency map, see Figure 1.2 for their example. The great benefit of these saliency maps is that it is directly visible which areas of the image are more salient compared to other regions within the image, having a similar concept for audio is harder and therefor probably less researched.

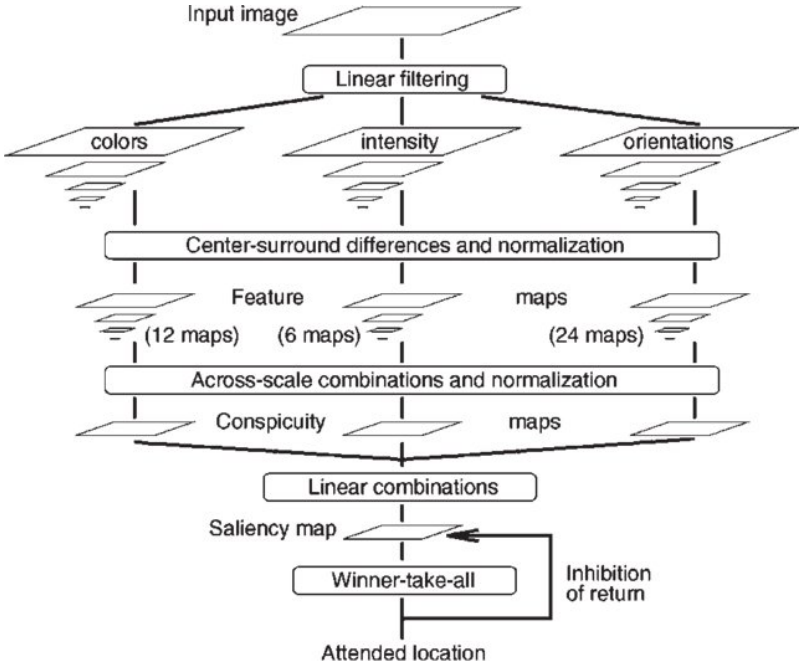


**Figure 1.1:** The general architecture of the Itti-Koch saliency map [39].

In the footsteps of the Itti-Koch model for mapping visual saliency, a similar model is made for auditory saliency in 2005 by C. Kayers et al. [10]. The model is depicted in Figure 1.3. Where as the input of Itti-Koch model is a digital image, Kayers model employed a sound wave. The model is adjusted to fit for the auditory scene, the sound wave is mapped into a intensity image by applying frequency analysis, the energy per frequency is plotted against time. Further, the feature extraction is done based on the auditory features intensity, frequency contrast and temporal contrast in a parallel manner. By applying dyadic Gaussian pyramids [23] spatial scales were created and by comparing the scales with the center-surround mechanism the feature maps were created. These maps are normalized using an asymmetric sliding window extending into the past and future in a manner consistent with psychoacoustical masking effects to obtain a feature-independent scale. Finally, saliency maps from each individual features are combined, in analogy to the idea of feature integration [9].

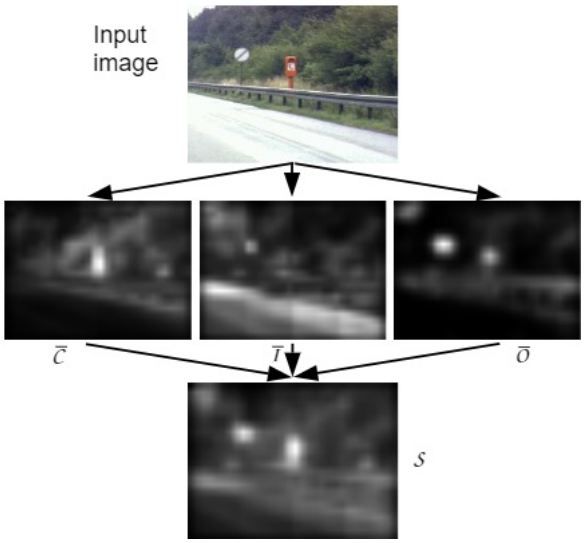


**Figure 1.2:** Example of Itti-Koch saliency map applied on an digital image. The letters $C$, $I$ and $O$ are the features; colors, intensity and orientation, respectively. $S$ is the final saliency map.

One of saliency map's objective is to predict which sensory events will capture our attention. The proposed auditory model analyzes sounds in the time-frequency domain and thus "localizes" important events in these dimensions. It can be suggested that these dimensions are important to consider, sounds frequencies are decomposed in the early auditory processing and attention can be directed to frequency and temporal position.

Second the saliency map serves as a model of how cortical areas extract sound features from a sensory scene and combines the inputs to attention. There is an ongoing debate of how the cortical areas are structured and these saliency map models could support new hypotheses.

A new approach to an auditory salience model is invented by Tsuchida and Cottrel [61], their model was not based on neurally-inspired and with the response of many feature maps (like Itti-Kochs), but based on a natural statistic model (SUN) proposed by Zhang [62]. Same as Kayser's model is a visual saliency model transformed to the auditory domain, the same counts for Tsuchidas model. The ASUN, AUditory Salience Using Natural statistics, model uses one single feature map, which is learned by using Independent Components Analysis of natural sounds, see Figure 1.4. The model focuses on the idea that novelty attract attention, salience at any point is based on the rarity of feature responses at that point. The extension from SUN to ASUN involves using realistic auditory features instead of visual ones, and short-term temporally local statistics are combined with long-term statistics from SUN. The SUN has both bottom-up and top-down components, however for the auditory domain the latter component is omitted. Bottom-up saliency of point $x$ in

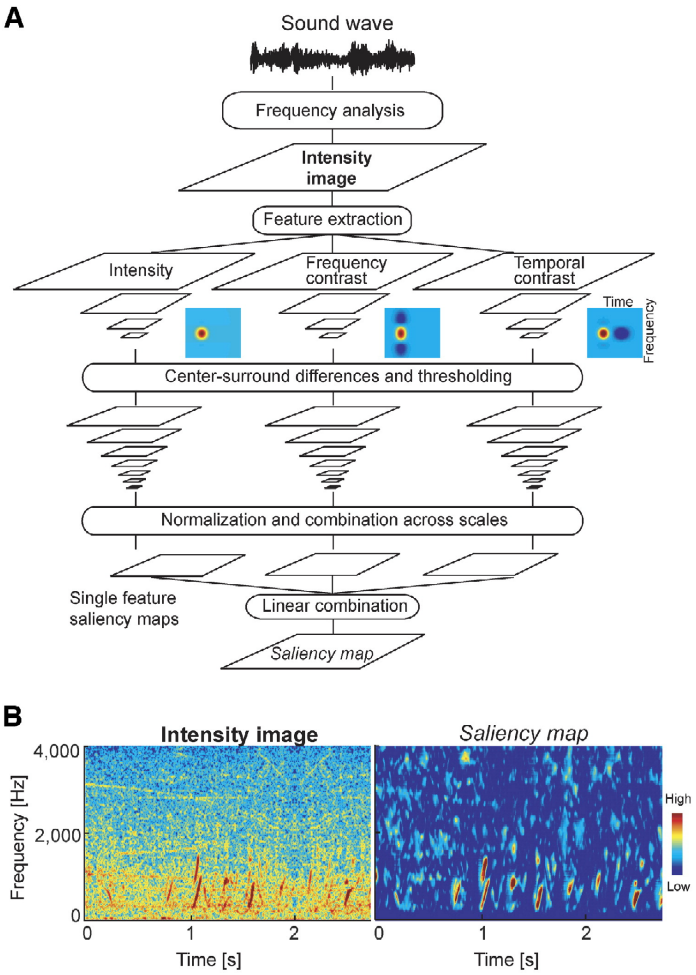


**Figure 1.3:** Auditory Saliency map model by Kasyer 2005 [10]. Features extraction is applied on the sound wave that is represented in the frequency-time domain. Main auditory features as intensity, frequency contrast and temporal contrast were applied in a parallel manner to define a saliency map from the intensity image.

the image at time $t$ as:

$$s_x(t) \propto -\log P(F_x = f_x) \tag{1.1}$$

where, $f$ counts for the vector of feature values whose probability is computed based on prior experience. In other words this can be seen as "self-information" of features, it implies that the value of rare features will draw attention.

Kayser's model is extended with two more features, beside the auditory features; intensity, frequency contrast and temporal contrast, orientations and pitch distribution were added by Kalinli and Narayanan [36]. Their model was specified for *speech* attention, prominent syllable and word locations were tested in an unsupervised manner. The results were compared well by human performance, it achieved 75.9% accuracy at the syllable level, and 78.1% accuracy at word level.

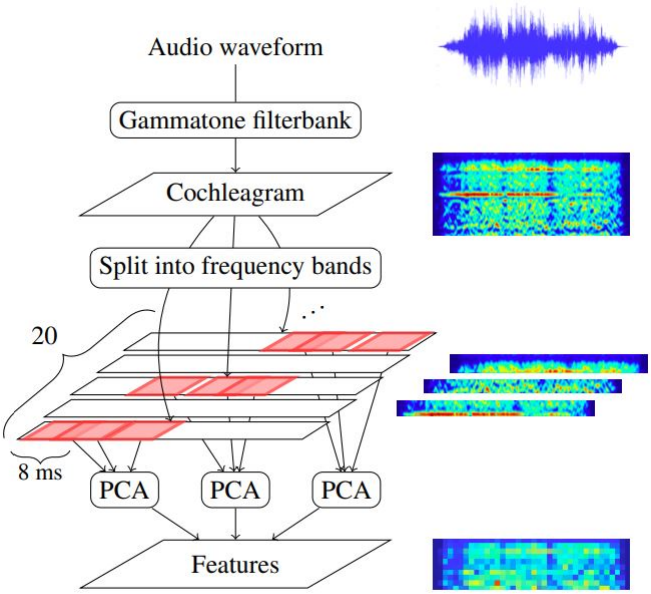


**Figure 1.4:** Schematic for feature transformation pipeline of the Auditory Saliency Using Natural stasitics (ASUN) model. The Input signals are waveforms which are first converted to smoothed cochleagram. Each band is seperated in 20 parts of 8 msec batches. Finally the batches are reduced in dimensions by applying PCA.

### 1.2.2 CNN based models

In the computer-vision field new salience prediction methods have been developed to improve the prediction accuracy of visual salience. With the improving computational power and vast increasing amount of available data over the last decade, researchers tended to apply artificial intelligent more often. The great benefit from artificial neural networks as the Convolutional Neural Network (CNN), the most widely used deep learning method for image processing applications, discriminant visual features can be extracted from the 2-D images. CNN uses a hierarchy of convolutional filters to apply multiple nonlinear transformations, see Chapter 2 Theory for more in-depth theory of CNN's. These deep CNN models achieve an even higher classification accuracy than previously hand-crafted features by humans. Examples of applications that utilizes CNN models are self-driving cars, scene classification, object detection and semantic segmentation [44].
The last few years CNN models are not only used for computer vision tasks, but as well for computer audition. Mel-spectrograms, spectrograms are taken as input for the CNN model, and promising results are be seen by Yu et al. [57] and Huang et al. [31]. However these are still in their infancy and more research is required.

### 1.2.3 Eye tracking

A device that gave the research on visual salience prediction models a great boost is the eye-tracking application. An eye-tracking device is capable of following the gaze of participant in a certain experimental setting. A study that applied this device

was to detect where a person is looking at when browsing through a specific website. Advertisement agencies utilized the results from these experiments to improve the likelihood of a person clicking on their commercial [6]. The gaze of a person is direct related to their selected attention. The gaze can be distinguished between overt and covert attention, where overt is where the eyes move towards something that attracted its attention and covert towards you want to pay attention on. A number of studies have concluded that overt shift of attention are primarily associated with the execution of saccadic eye movements (Findlay, 1997, Mioli et al., 2001)[20, 11]. Saccadic targeting is controlled by saliency local properties, in combination with the task in mind, behavioral goals or intention. This suggests that eye-tracking devices can be utilized as top-down attention measure of visual objects. Consequently many researchers employed this fact of top-down measure and improved their computational models for salience in visual stimuli. [43].

As eye-tracking is a good measure for top-down attention in visual aspect, it is not applicable for auditory salience. The sound object that attracts the person's attention is not shown in a physically way. The only visible reaction of a persons attention to sound is when the person tries to localize the sounds direction. Your premier movement is to turn your head towards the events location to 'see' what occurs and get a better hearing. But when multiple sounds are presented this cannot be appointed.

One study (Huang, 2017) [32] attempted to correlate pupil dilation with salient sound events. The size of eye's pupils cannot be consciously controlled, hence it can be seen as a measure of the brain activity or of sort. The measure of increasing and decreasing the diameter of a persons eye pupil is used in various researcher fields. One general field is emotion research, where the persons emotional reaction is examined, this can be reaction to a conversation, music, film or other sound event. However the fact that there were correlations found between some salient sound events and eye dilations, it cannot be indicated as good measure. A theory could be that the increasing loudness accounts for the increase of pupil, were this is not a direct correlation to salient.

### 1.2.4 Activation of CNN analysis by hand-crafted acoustic features

In the study, "*A Case Study of Deep-Learned Activations via Hand-Crafted Audio Features* by a O. Slizovskaia et al. [48], they present a method for analysis of activations of audio convolutional neural networks by use of hand-crafted acoustic features. These hand-crafted features are well understood in the literature, namely loudness and onset rate. As dataset to analyse features they use user-generated recordings of different musical instruments performances. Furthermore they utilize state-of-the-art VGG-style architectures: CNN Audiotagger, VGGish and Music Motivated CNN. All three receive a mel-spectrum as the input to the neural network model, consist of blocks of convolutional and max-pooling layers, and dense layers.

In the computer vision field CNN's learn boundaries in the first layer of a neural network, and more complex concepts in the subsequent deeper layers. They hypothesize that a similar behaviour is present for audio-based CNNs, the first layers can occasionally learn some of the hand-crafted features in a similar matter. The goal of their work is to investigate if those features learned by the pre-trained networks are possible to identify.

From their study it can be concluded that even for the complex pre-trained models that construct features in a very different way than traditional methods, the relationship between hand-crafted features and activations provide insights for better understanding of internal representations of CNNs.

### 1.2.5 Aligned activations of CNN with brain activity of human visual cortex

Recent breakthroughs in artificial intelligence have discovered principles relating to neural processing, specifically vision. Previous research has shown that the hierarchy of human visual areas and layers of deep convolutional neural networks (DCNN) trained on visual object recognition are closey linked. In a paper published in Nature by I.Kuzovivk et al. [33] they go a step further and utilize DCNN to see which frequency bands correlate with increasingly complicated feature transformations in the ventral visual pathway. Therefore they evaluate the alignment between the DCNN and signals at different frequency bands using intracranial depth recordings from 100 patients. The growing complexity of visual feature representations in DCNN is matched by gamma activity (30–70 Hz). These findings reveal that the DCNN's activity captures the basic aspects of biological object detection in the frequency domain as well as in space and time. These findings show that artificial intelligence systems have the potential to advance our understanding of the brain. From this work it is interested if a similar relationship between DNNs and human auditory areas can be shown, by looking at the brain activity for different frequency bands. And instead of using images as input, audio could hypothetically give similar results.

### 1.2.6 EEG based models

EEG, brain activity recordings, has been used to identify perceived melodies with success. 10 participants listened to 7 short song clips with a length between 3.26s and 4.36s in a study by Schaefer et al. [51]. Each stimulus was shown 140 times in randomized back-to-back sequences of all stimuli for single-trial classification. They were able to accurately classify the ERPs of single trials using a quadratically regularized linear logistic-regression classifier with 10-fold cross-validation. Within each subject, the accuracy ranged from 25 to 70 percent. They achieved between 35 and 53 percent accuracy using the same classification approach across all subjects.

Another closely related field of research is the reconstruction of auditory stimuli from EEG recordings. EEG recorded while listening to natural speech contains traces of the speech amplitude envelope, according to Deng et al. [55]. To improve the signal's strength and correctly identify audible phrases, they used independent component analysis (ICA) and a source localisation technique. They reported statistically significant single-sentence categorization performance for two of eight subjects using their technique, with greater performance when numerous sentences were merged for a longer trial period.

From the above described studies about visual and auditory salience it can be concluded that bottom-up salience prediction is well understood, but top-down attention parameters are hardly understood. The computer-vision branch found its solution in eye-tracking devices as top-down attention indicator, the computer-audition came with a new approach through EEG brain signals.

## 1.3 Goal study

The goal of this thesis project is to present potential method to analyze auditory salience and auditory attention by use of a computational model. This computational model is based on machine learning algorithms, more specific artificial neural networks, that imitate brain's own network structure. From this it can be possible to find correlations between how the brain and the neural network processes sound.

### 1.3.1 Aims

The following aims are created to this thesis.

- Is it possible to build a computational model that can predict auditory salience in music for both bottom-up and top-down attention?
- What is the relationship between activations of DNNs layers, low-level and high-level acoustic features, and EEG brain signals?

### 1.3.2 Limitations

The artificial neural network used to learn acoustic features from audio streams is not designed and trained by ourself in this thesis work, instead an existing pre-trained network is utilized. This to save time and be certain that the network fulfills the requirements of extracting features sufficient. The hand-crafted acoustic features are developed with use of Python public libraries that contain tools to perform music and audio analyses. These libraries contain several implementation of most general acoustic features that can be easily applied. Due to the Covid-19 situation and to reduce time, the experiment to gather subjects brain activity while listening to a various of music excerpts is not carried out too, therefore data is taken from online public databases. These databases consist of well documented experiments and are provided with complete source codes and observation data. Moreover studies that

utilized these databases are mentioned, this makes it convenient to know how to approach this data and what can be done for further studies.

### 1.3.3 Collaboration

The establishment of this thesis work was in collaboration with the departments of Sound and Image Lab and Electronics, Information and Bioengineering at Politecnico di Milano. Both shared their specific knowledge about the topics as music information retrievel, artificial neural networks, digital signal processing and preprocessing EEG signals. In addition their workstations were available to perform the vast and heavy computations. Lastly to mention is the department of Research Focus Cognitive Sciences of the University of Potsdam, were the OpenMIIR database was established.

## 1.4 Thesis outline

The outline of this report is as follows, Chapter 2 Theory, is devoted for more in-depth theortical background information of for example EEG signals, salience maps, machine learning and aritifical neural networks. Chapter 3 Methodology describes the realization of the computational auditory salience model, including a 3.4 Implementation Section. Here the description of which tools are utilized, for example the pre-processing steps of EEG brain signals, or acoustic features with the Librosa audio library in Python. The outcome of the computation model and the final correlation analyses are presented in Chapter 4 Results. This section is supported by graphs and tables, which will be commented. This is followed by the Chapter 5 Discussion, here the results are discussed and compared with other studies and findings. Lastly, an overall conclusion of the method used, outcome of the correlation analyses, will be given in the Chapter 6 Conclusion. Additional detailed information is found in the appendices. Here the source code and remaining graphs will be added.

# 2
# Theory

The theory behind the components of the model and related terms are described in this chapter. The first section is devoted to brain activity detection technique EEG, brain regions and the functions of the different frequency bands. Since this work is not part of the neuroscience field, the theory will be superficial. The follow up section dives more in-depth of salience in vision and salience maps, this to get a better understanding of how to computational models deals with salience. The third section focuses on artificial intelligence with its sub-fields machine learning and deep learning. The latter two fields uses techniques that are utilized for extracting acoustic features. The acoustic features itself is the last section of the this Chapter 2 Theory.

## 2.1 EEG

In the neuroscience field the brain activity is extensively studied. Multiple are techniques developed and utilized to capture activity in the brain, each technique with its own pros and cons. The most used, cheap and simple to execute technique is electroencephalography (EEG) and will be explained in this section.

The brain consists of billions of neurons and synapses which interconnected these neurons. These synapses act as gateways of inhibitory or excitatory activity. When a synaptic activity occurs, neurotransmitters are binding to receptors on the postsynaptic membrane, a subtle electrical impulse is generated and referred to as a postsynaptic potential. The detection of a single neuron burst is difficult and unreliable if there is no direct contact with it. However, whenever thousands of neurons fire in the same region of the brain and at the same time, the electrical field they generate is strong enough to go through tissue and bone. This makes it possible to measure brain activity at the surface of the skull.

Electroencephalography is the technique that captures neuronal electrical activity generated by the brain via multiple electrodes positioned on the scalp surface. The quantity of the signal is in millivolt and measured in different frequency bands. It mostly records signals from small areas of the brain surrounding each electrode. EEG is primarily measuring postsynaptic potentials or changes in the membrane potential. Typical amount of electrodes used for measurements are 64 or more, these are mounted in a cap that the participant wear over the head. The benefit of the cap is that the position of the measurement points are fixed, this results in

more consistent data collection for each respondent.

EEG provides an image of electrical activity in the brain represented as waves of varying frequency, amplitude, and shape. The time resolution of these waves is high, in the order of milliseconds. EEG can be used to measure brain activity that occurs during an event, or to measure spontaneous brain activity. An brain activity that occurs in association with an event, is sometimes called the event-related potential. This makes EEG useful to let participants do an experiment while recording the brain signals, and find relation between the two. In addition EEG is used for a variety of clinical applications, such as neurological disorders as seizures and epilepsy or to monitor sleep for the diagnosis of sleep disorders.

A drawback for EEG is the spatial resolution, as the electrodes only catch the activity of the brain at the surface of the brain. It is difficult where the signal was produced and deeper regions are barely detectable.

## 2.2 Brain areas and its functions

There are four major lobes of the cerebral cortex which each has their own functions. The four cortex are frontal, temporal, occipital and parietal cortex [58], see Figure 2.1.

The frontal cortex is the part which is located in the front of the skull, just behind the forehead. This part of the brain is relatively enlarged compared to most other mammals. The basic function of the frontal cortex is the executive, it plans our future, helps us sustain control, and monitor our behaviour. In the temporal cortex is associated creating and preserving both conscious and long-term memory. It plays a role in visual and sound processing in the comprehension of written and spoken language. In the middle and top of the skull lays the parietal cortex and is concerned with integrating information that comes from external sources and internal sensory feedback from our bodies. It is responsible for synthesizing all these sources of information into a coherent representation of how our body relates to the environment, and how all things as objects or persons in the environment relate to us spatially. For example a task like eye or hand movement as well as eye-hand coordination would not be possible without parietal cortex. In addition it processes, stores and retrieves size, shape and orientation of objects. The fourth major lobe, occipital cortex, is the center for visual processing and is located at the back of the head. Other functions are distance and depth perception, object and face recognition, color determination, and memory formation.

### 2.2.1 EEG frequency bands

The signal that is received by the electrodes is a mix of underlying base frequencies, which are related to certain cognitive, affective or attentional states. The frequency

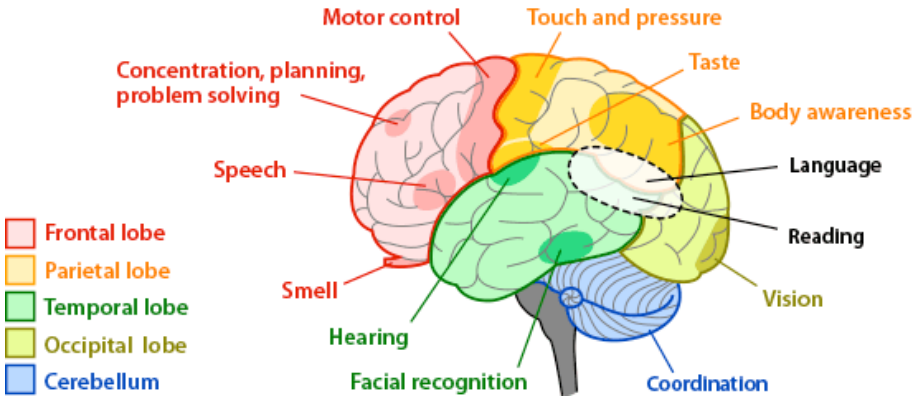


**Figure 2.1:** The four cortex and its functions depicted: frontal, temporal, occipital and parietal. Plus the cerebellum. *Source: Askabiologist* [58]

patterns change over time when the brain is in a particular state and gives insight into cognitive processes. The oscillating electrical voltages in the brain are divided in five main frequency bands, which are widely recognized brain waves. The five brain waves frequency ranges and their brain states are shown in table 2.1 and Figure 2.2 below.

| Frequency band | Frequency | Brain states |
|---|---|---|
| Delta ($\delta$) | 0.5-4 Hz | Sleep |
| Theta ($\theta$) | 4-8 Hz | Deeply relaxed, inward focused |
| Alpha ($\alpha$) | 8-12 Hz | Very relaxed, passive attention |
| Beta ($\beta$) | 12-35 Hz | Anxiety dominant, active, external attention, relaxed |
| Gamma ($\gamma$) | >35 Hz | Concentration |

**Table 2.1:** Characteristics of the five main brain waves.

## 2.3 Salience in vision

The vast majority of research in salience have been conducted in the visual domain. The reason why visual salience has been studied more than auditory salience can be explained by the following points. The concept of visual salience is more easily definable since the primitives for vision are better understood. Another reason is the evaluation of visual salience is easier to implement as there is physical correlation, eye-gaze [49]. People tend to look to salience stimulus, consequently eye-gaze can be utilized as physical measure, therefor eye-tracking device finds its solution. It has been shown that eye-tracking correlate with stimulus salience, which allows to quantify visual salience. In this section, the current research that has been carried out until now and the motivation for studying visual salience will be provided.

In the early studies to salience the focus was aimed to stimulus-driven attention. The objects or stimulus that was looked at gets a person attention because it stands out

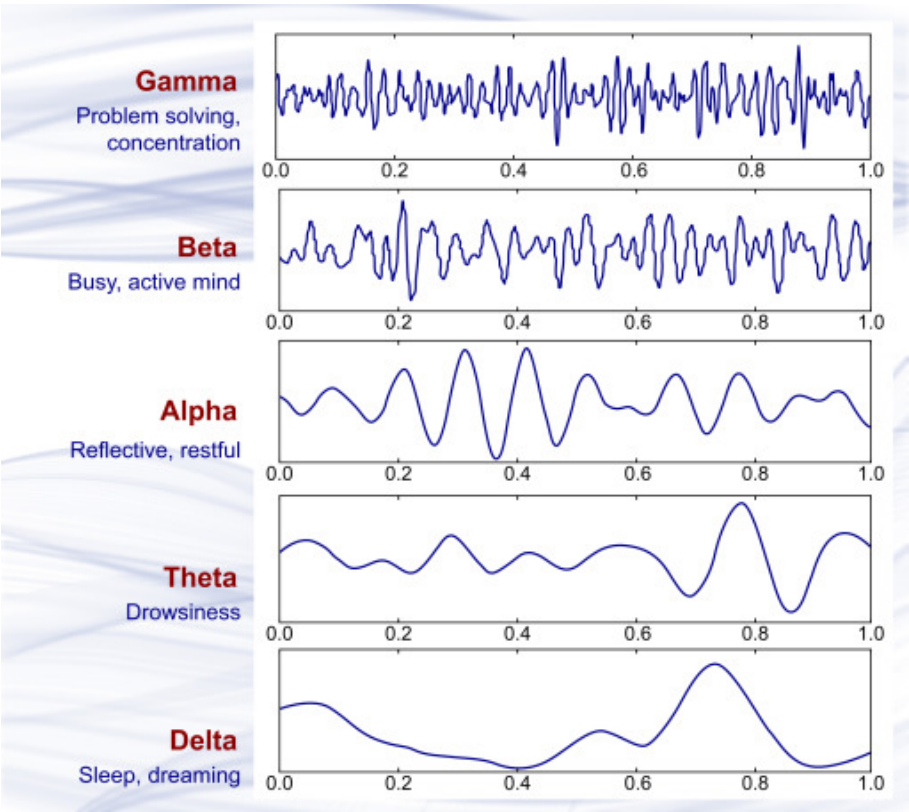


**Figure 2.2:** Five brain wave frequency bands and its characteristics. *Source: Abhang 2016* [2].

of the rest in a visual scene [34]. Certain charactertics of the visual aspect of both the stimulus and scene makes it more salient. The mechanisam of the brain that is leading to this is bottom-up attention. In this attentional process the brain reacts almost immediately to the incoming sensory information and focus subsequently on that particular object. During this process the other not important information is reduced, so only a small portion is observed. This process is often referred as "selective attention". However, user-driven factors from the top down can significantly alter or even completely override the bottom-up deployment of attention to the salient location.

The physical property or visual characteristics of why an stimulus is salient object depends fully on its surroundings. Salience is the consequence of an interaction of a stimulus with other stimuli. This can be explained by a straight-forward example, consider that a color-blind person is looking at the same physical scene as a person with normal vision, the experience of visual salience is completely different. There are less obvious example where observers have slight different view of salience, this could be various expertise of the persons, mood, or the participant was told to look for a specific thing.

To examine visual salience of stimuli experiments several have been conducted. One of the experiments is to show the subject different pictures of a drawing. These

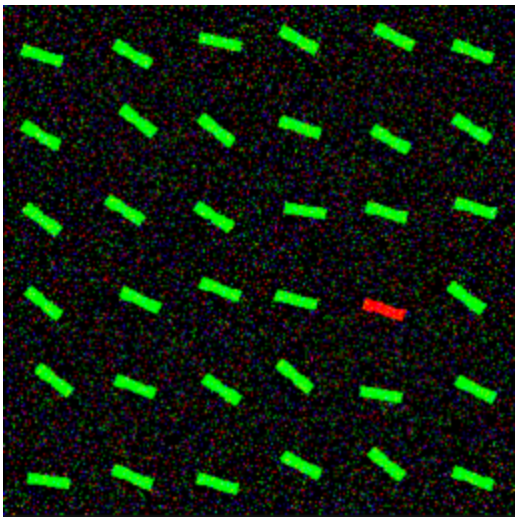

**Figure 2.3:** Visual salience example of the pop-out effect by bottom-up attention. *Source: Delmotte* [15]

pictures can be photo's of real scene or artificial made. To illustrate such an experiment the following drawings are used.

In Figure 2.3 are arrays depicted with the same shape but one with different color. The array with the red color strongly pops-out of the scene effortlessly and immediately attracts attention. Various studies have suggested that in simple frames like Figure 2.3 that no scanning occurs. The attention is directly draw to the salient, read "red" array, item, nevertheless how many other items are in present. [9].

When the red array was colored green, or all the green arrays colored red, no single salient object could be chosen. This means color is only a salient property if the surrounding has different properties. Another property could be local visual, or shape. If one of the green array was position in a vertical manner instead of horizontal like the rest, this would have been the pop-out object.

The outcome of these examples suggests that the image is processed all at one to determine salience, this counts for every location and to orient towards to the salient stimulus location.

In the image of Figure 2.4 there is again one bar that is different from the rest. However by the design of the other bars, it is less obvious visible because it is not defined by a single unique feature. The bar that is conflicting from the rest is the vertical red bar. This follows that salience does not guide you directly towards a potentially interesting items, but by carefully scanning.

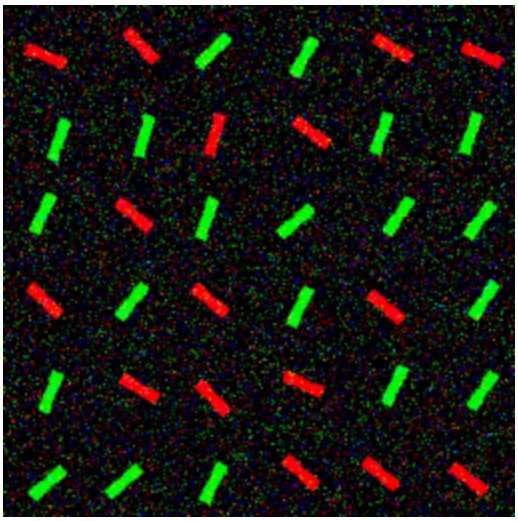

**Figure 2.4:** Visual salience example where the pop-out effect does occur little. *Source: Delmotte* [15]

## 2.4 Salience maps - auditory / visual

The brain interacts with the visual world by responds via a collection of parallel neural pathways beginning in the retina [53]. Certain pathways perform selective modulation of the visual signal, highlighting features and locations that contain relevant or important information.

To evaluate stimulus of visual salience, models are developed which 'maps' highlighted features and location. Different approaches are attempted to get as accurate map as possible. Two types map have been proposed. One type is the salience map, which is based on low-level visual features such as color, brightness, orientation, shape and motion. The second map is known as priority map. In addition to the bottom-up salience map it integrates task-driven information, top-down attention.

Salience maps are computational models that describes low-level visual features such as colour, luminance or shape. These features are combined into a single global map representing the relative salience of each point on the map. Certain areas on the map contain more components that are associated with visual attractiveness than other parts of the map. High and lower salient areas are denoted on the map. The first concept of salience map is original proposed by Koch & Ullman [38], and later implemented by Itti et al. [39].

## 2.5 Introduction AI

Over the last decade Artificial Intelligence (AI) has been a trending topic and implemented in studies over all kind of scientific fields, such as computer science, data science, biology, physics, neuroscience and economics etc. AI is defined as the ability of

machines to exhibit intelligent behavior through smart and self-learning algorithms. The term artificial intelligent was introduced in the fifties of the last century and the first algorithms were build in the sixties and seventies [52]. The popularity of AI during that period was high and many papers were published, however real applications stayed out. In the following decades the interest in AI was relatively low, only up to five to six years ago AI got a boost in its practise due to the availability of vast amount of data together with the rapid grow of computational power, the annual AI index report by arXiv shows a sixfold grow of AI-related publications on arXiv [66] from 2015 till 2021. Nowadays AI is implemented in many daily used products such as smartphones, self-driving cars, social media and streaming services, for example your recommendation lists of Spotify or Netflix. This section will cover more in-depth understands of machine learning and deep learning which includes artificial neural networks.

## 2.6 Machine Learning vs. Neural Networks

Artificial Intelligent is a collection term which includes machine learning, deep learning and neural networks. Each of these sub genres share the same goal of getting a understanding of data by the use of a machine, but differ in how each algorithm learns. Each field is a sub-field of the former.

The way in which machine learning and deep learning differ is the approach of feature extraction. Feature extraction can be seen as finding certain patterns in the data. As for a dataset that consist of many pictures, a feature could be to find images with a particular color. In ML a human expert determines the color feature to understand the difference between data, this is known as feature engineering. In the case of deep learning, these features are not implemented by human intervention, but is learned by the algorithm itself, feature learning. This is were neural networks come in. Deep learning and neural networks show accelerating progress in areas, such as speech recognition, computer vision and natural language processing.

## 2.7 Artificial Neural Network

Neural networks, or artificial neural networks (ANNs), are comprised with components that mimic the structure of our brain. The human brain consist on average of 86 billion neurons [28], and each of these neurons are connected with synapses. In the computer science they are referred as *input nodes* $x_1, x_2, x_3, ..., x_n$ and *connections* with their weights $w_1, w_2, w_3, ..., w_n$, respectively. In Figure 2.5 a schematic drawing of a biological neuron and an artificial neuron are depicted. The biological neurons are extremely complex cells, but their basic computational nature in terms of inputs and output is relatively simple. Each neuron sums up its electrical load inputs from its dendrites, and if the total electrical impulse strength exceeds a certain *firing threshold*, the neuron fires a new impulse. This impulse distributes the signal to other neurons, which can lead to new firing impulses. A similar event occurs for the artificial neurons, nodes, in a neural network [21]. The output node receives

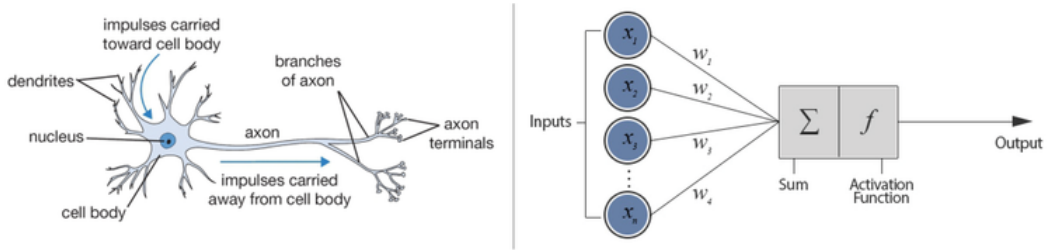


**Figure 2.5:** Schematic drawing of a biological neuron versus an artificial neural network. Source: DataCamp [64]

values from the input nodes, with connection weights $w_n$, and they are summed up to a *weighted sum*, see right drawing in Figure 2.5.

The node can be defined with its input $\boldsymbol{x} \in \boldsymbol{X}$ and output $\boldsymbol{y} \in \boldsymbol{Y}$:

$$y = f(\sum_{i=1}^{D} w_i * x_i), \tag{2.1}$$

$$\boldsymbol{y} = f(\boldsymbol{x}; \boldsymbol{w}), \tag{2.2}$$

where $\boldsymbol{w} \in \boldsymbol{W}$ are the weights.

Subsequently, the threshold for the node to send information forward is set by an *activation function*. The activation function, also called a *transfer function*, determines the value of the node output. It ensures that values that are passed on lie within a tunable, expected range. The most simple form of activation function is a *step function*, which gives as output a 1 value when total input exceeds the given threshold, otherwise its output is of value 0, see equation below:

$$f(x) = \begin{cases} 0 & \text{if } x < 0, \\ 1 & \text{if } x \geq 0, \end{cases}$$

The step function is a discontinues function, the output is not a smooth line, this can cause problems in the mathematical processing. Therefore a continuous variant like the Sigmoid function is preferred, also know as the logistic function, given below in 2.3:

$$\sigma(t) = \frac{1}{1 + e^{-\beta t}} \tag{2.3}$$

$$y_i = \frac{e^{\zeta_i}}{\sum_{j \in L} e^{\zeta_i}}, \tag{2.4}$$

where L is the set of neurons in the output layer. The functions are depicted in Figure 2.6.

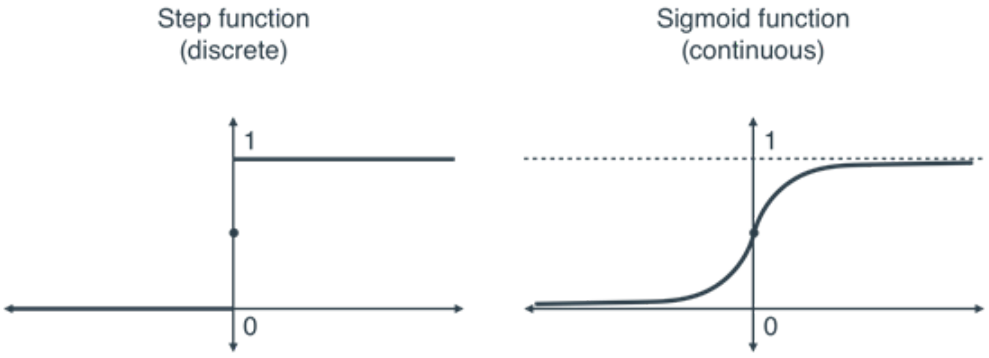


**Figure 2.6:** Left: The step function used to build discrete nodes. For any negative input it outputs a value of 0 and a value of 1 for any input that is positive or zero. It has a discontinuity at zero. Right: The Sigmoid function, or logistic function, utilized to build continuous nodes. Values less than 0.5 are for outputs for negative inputs, and values greater than 0.5 for positive inputs. At zero it outputs 0.5. It is continuous and differentiable for any point.

The result of the given above, is an output node, which is associated with the transfer function of the weighted sum of the input nodes. One more important additional parameter to mention is the *bias*, this can be noted as the weight associated with an additional input node which value is set to 1 permanently. The bias value is of importance because it allows to shift the activation function to the left or right, which can affect the learning success. In machine learning *parameters* are referred to weights and biases, the variables that makes prediction.

In practise the artificial network consist of at least three layers of nodes, an input layer, an output layer and one or more *hidden* layers, see Figure 2.7. The input layer receives the raw data from the external world, this could be images or digital text for example. The hidden layers act as the learning machine through performing nonlinear transformations of the input data. Finally, the output layer produces the result. The network size differ from a couple nodes to a few hundreds to thousands, depending on the task and computational power. Most modeling assumes that the layers are *fully connected*, meaning all the units or nodes of one layer are connected to all the nodes of their neighbouring layers. The word "deep" in deep learning is referred to the depth of layers in an artificial neural network. A neural network is consider a deep learning network when it consist of more than three layers, a basic network two or three layers. The neural network can be seen as a system that is trying to interpret, understand, perceive, translate, learn, remember, recognize data that is given as input.

#### 2.7.0.1 Learning process

The core function of an artificial neural network is its learning process that is performed by *backpropagation*, this can be more easily explained with *supervised learning*, one of the most common used learning algorithm. In supervised learning tasks,

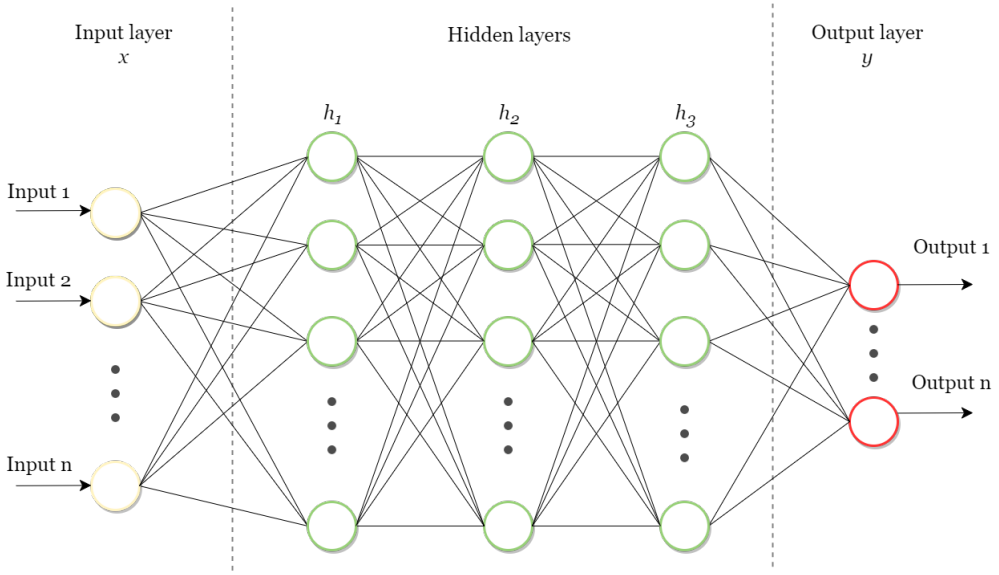


**Figure 2.7:** Schematic drawing of a basic artificial neural network with input nodes $x$, hidden layers $h$ and output nodes $y$.

the training data includes both the input and the desired results. The network tries by adjusting the weights values of the hidden layer nodes, to find the pattern between the input data and the desired output result. An example of a supervised learning task can be a training set with thousands of digitally handwritten numbers from 0 to 9 in image format, lets assume 64x64 pixels. Within the dataset the numbers are *labelled* with their correct values or *targets*, number values. The network tries to find patterns of each of these input images, the curve, circles, vertical and horizontal lines, and so on, all the different features of the images. By training the network it extracts features from the number images and tries to find a match with the target. The final result is a probability factor, the network tries to find its best match according to statistics. For example the handwritten number 0 has 98% chance to be of the value null, whereas the other values have a probability of less than two percent.

The neural network is provided with the training data, therefor preprocessing steps are made to train the network well. First step is to split dataset up in two parts, a *training set* and a *test set*, in particular cases a third part which is the *validation set*. The training set contains most of the data, a common set is divided in $70\% - 30\%$ train vs. test data. During the training phase of a network, the network is learning, adjusting and fine-tuning itself to get closer to the intended output. Here gets backpropagation its function to optimize the weights of the hidden layers by compare with the intended output to the actual output using a *loss function*. Multiple train rounds are performed to calibrate the network weight values, the loss function compares the output result with the intended output value and gives an error value as result, also known as a *cost*. This error value is then used during backpropagation to adjust the "wrong" weights between the neurons by calibrating them. A commonly used cost or loss function is the Mean Square Error (MSE), here below is the MSE given:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2.5}$$

where $y_i$ are the observed values, $\hat{y}_i$ the predicted value and $n$ the number of data points. The MSE is the average of the squared difference between the predicted values and the actual values. Here where predictions are far out of range of the actual value are penalized much in comparison with less deviated values. In other words, backpropagation aims to minimize the cost function by adjusting the parameters, weights and biases, of the network. The level of adjustment is determined by *gradients* of the loss function with respect to those parameters. The computation of gradients will be explained in the following section.

#### 2.7.0.2 Gradient descent

The neural network model learns by minimizing a specified cost function, here minimizing is performed by *gradient descent*. Gradient descent is an efficient optimization algorithm that tries to find a local or global minima of a given function.
To find a local minimum of a function the gradient or *direction* is required. In case for a neural network the direction refers to how the parameters should be adjusted to further reduce the cost function. As the training rounds iterates, it converges towards a minimum. The pace of convergent is determined by the *step size* or *learning rate*, moving in small steps leads to a higher probability to get closest to the local minimum. Moving in larger steps, the minimum can be found quicker, however too large steps can lead to divergence, with other words no exact minimum can be found.

To illustrate the gradient descent the following hypothesis, parameters and cost function are set to get to the main goal which is reaching the local minimum of a function. The case or hypothesis for this illustration is a linear regression task, given as:

$$h_\theta(x) = \theta_0 + \theta_1 x \tag{2.6}$$

where $\theta_0$ and $\theta_1$ are the parameters of the linear regression function and $x$ coordinate in x-y plane. The goal is to learn a function $h : X \longrightarrow Y$, where $h(x)$ is the "best" prediction for the corresponding value of y. The value of y depends on the parameter values $\theta_0$ and $\theta_1$ and are chosen to fit best with the a straight line mapped on the given data points.

Depict a graph with several data points in the x-y plane, the training set. A arbitrary chosen straight line is mapped on the training set to should pass through all the points. By applying the MSE cost function the distance or *error* from each of the data point to the straight line is calculated. Further, the position of the straight line is adjusted to obtain the minimum MSE. Instead of moving the line in

a stochastic manner around to find the best fit, the gradient descent algorithm is applied. Formally, it is given as follows:

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_i} J(\theta_0, \theta_1) \quad (\text{for} \quad j = 0 \quad \text{and} \quad j = 1) \tag{2.7}$$

where $\alpha$, is the learning rate and $J(\theta_0, \theta_1)$ the cost function. The cost function quantifies the error between the actual values and the predicted values, and present it in the form of a single real number. The function for the linear regression case in the x-y plane is:

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \Big( h_\theta(x^{(i)} - y^{(i)}) \Big)^2 \tag{2.8}$$

The next step is to solve the partial derivative of the cost function until it convergences. Therefore the $\theta_j$ is chosen for $j = 0$ and $j = 1$, an arbitrary learning rate $\alpha$ and implemented in equation 2.7. The chosen learning rate, the step size of the gradient descent, has its constraints; to reach a local minimum efficiently, $\alpha$ has to be set appropriately, neither too high nor too low. Depending on the initial point, where the prediction value starts on the graph, it could end up at different points. Typically, learning rate values are chosen manually in the range of $0.1, 0.01$ or $0.001$. The derivative of the gradient descent is given as:

$$\begin{aligned} \theta_0 : &= \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \Big( h_\theta(x^{(i)} - y^{(i)}) \Big) \\ \theta_1 : &= \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \Big( h_\theta(x^{(i)} - y^{(i)}) \Big) \cdot x^{(i)} \end{aligned} \tag{2.9}$$

where $\theta_0$ and $\theta_1$ before the assignment operator are the new positions, and $\theta_0$ and $\theta_1$ after the assignment operator the old position. $\frac{1}{m} \sum_{i=1}^{m} \Big( h_\theta(x^{(i)} - y^{(i)}) \Big)$ is the derivative of the MSE loss function. Both equations are repeated until a local minimum is found. The influence of the learning rate $\alpha$ is depicted in Figure 2.8. A too high learning rate value $\alpha$ leads to divergence, a too low value will lead to convergence, but with a slow pace. A value that lies between both is preferred.
A few more words are to be said to complete the description of the learning process of the neural network to a dataset. As mentioned the neural network is trained on the dataset, which is split up in three parts, training, validation and test data. The goal of the first computation round on the training data is to train the network. It gets an understanding of the input data and the desired output by applying backpropagation. Here a cost function is introduced to find the error values of the weights of the nodes, and subsequently to be adjusted to a more correct value. To

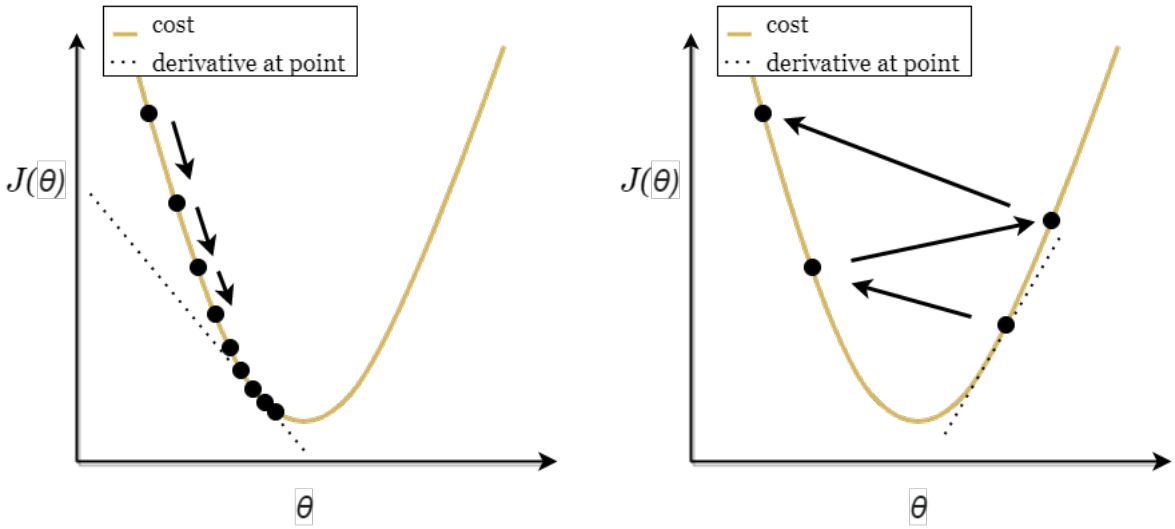


**Figure 2.8:** Left: graph of the gradient descent algorithm where learning rate $\alpha$ is set to a value that leads to convergence, a local minimum. Right: $\alpha$ is too high, values start to bounce from the curve and the error keeps increasing infinitely, divergence. No local minimum of the cost function will be found.

minimize the error the gradient descent algorithm is performed, the derivative of the cost function is taken with the right parameters and a specified learning rate. Once the model has achieved a certain learning result after several iterations of training, *epochs*, the model trains on the next data set, the validation set. Instead of usage of the training data to update the weights of the neural network, the validation data is used to assess the performance. And at last the model is ran on the test set to check if the network achieved prediction well. This set, validation and test, is not 'seen' by the model yet, and thus a method to check if the model can be applied on new data. Eventually, the goal of a neural network is the understand unseen data and predict what the outcome is, this is known as the neural network ability to *generalize*. For example the ability of a model to predict whether the image shows a dog or a cat, on images that are not used for any training purposes. It can occur that the model develops a too well prediction of the training data, that it cannot recognize the dog or cat in new given data, the neural network is *overfitting*. Overfitting is a undesired outcome and the model or the training data needs to be adjusted. The opposite can occur as well, the model is not able to find the right features to predict the outcome, known as *underfitting*. This leads to a model that is not able to find correlations between known data and new data, only to a few data points at most. To get a model well generalized a few methods can be applied, either to the training data or to the model itself. The most common methods that are applied are as follows.

The most simple solution to tackle overfitting is *early stopping*, the training procedure is stopped when after a certain amount of epochs, the convergence rate decreases minimal or the *validation loss* increases. Validation loss is the same metric as training loss, except it indicates how well the model fits the new data. Training loss is the indicator for how well the model fit to training data. After each epoch the values of training loss and validation loss are tracked, to detect whether a model

is causing overfitting or underfitting, and subsequently adjustments are applied if needed. Other common approach is the employment of *dropouts*, it randomly drops out neurons from the network during training in each iteration. When different sets of neurons are dropped out it is equivalent to training different neural networks. Each of the different networks will overfit differently, thus the net effect of dropout will be to reduce overfitting. This can also be seen as creating a more simplified neural network. A simplified network means less features can be learned and thus a more generalization of the data. A similar method is the usage of *regularization*, which penalizes high-valued regression coefficients. This leads to a reduction of parameters and shrinks (simplifies) the model too. Instead of adjusting the nodes and weights to prevent the model from overfitting, the training data can be changed as well to achieve the same result by applying data augmentation. Data augmentation means simply increasing the size of data points in the dataset. More data means more different data to learn from by the neural network, and causes a more generalization. Popular image augmentation techniques are mirroring, scaling, rotating, adding noise or changing brightness et cetera. This way many more images can be made from the same initial dataset. It is cheap and quick technique to increase the dataset size to tackle overfitting of the network. There are various neural networks models, one of them is the convolutional neural network (CNN) which will be described in the following section.

## 2.8 Convolutional Neural Networks

In deep learning, convolutional neural networks are a type of artificial neural networks that can take images as input. It can find patterns in images that can for example be translated to object classes. These CNNs are also used with audio as input, therefore the audio is most often transformed into spectrograms. This section will explain more in-depth understanding of CNNs.

Convolutional neural networks generally consist of two primarly building components, convolutional layers and pooling layers. The input of CNN are images, for the feature extraction task of this thesis the CNN based network (VGGish) uses spectrograms of audio as input, these can also be seen as images. The input is constructed as a 3-dimensional frame, with the width and height referred as the outline of the frame, and the depth the represents the number of channels. In case of a RGB image, three channels are corresponding to the red (R), green (G) and blue (B) components of the respective color, which each their digital values, usually 0-255.

The most typical feature of a CNN is convolution operations, where only a portion of the neurons in previous layer of the network contributes to the activations of each neuron in the next layer. This operation is performed by a kernal or filter. The size of the kernel is much smaller than the image dimensions, a common used size is 3x3. During the convolution operations moves the kernel over the image in a systematic manner, from start to end, and uses matrix multiplications to get the values for feature maps. Feature map is here the activations of a layer, see Appendix B the activations depicted. The advantage of this operation is the reduction of weights,

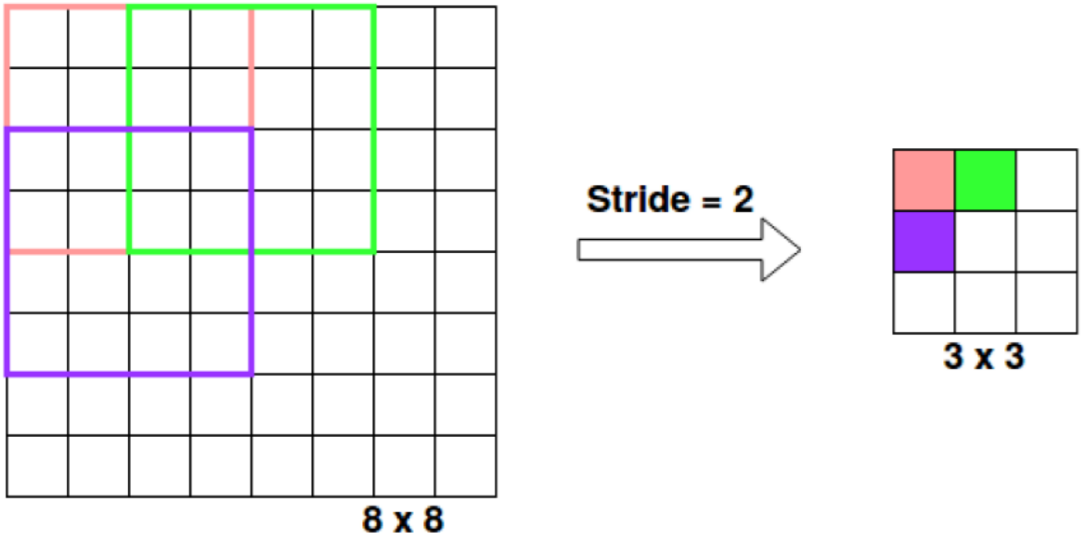


**Figure 2.9:** Example sketch of convolutional operation with strides. The 8x8 layer turns into a 3x3 after a stride 2 convolution. The filter size is 4x4 and the color boxes from the plot on the left correspond to the respective color boxes at the right.

since kernel is only selection a part of the image. Occasionally, zero-padding can be applied to add zeros around the borders of the input image, this is done to control the spatial size of the output. The strides in a CNN are the number of moved pixels by the kernel after each computation, a stride equal to 1 means that the kernel moves 1 pixel at the time, the output size will remain the same if zero-padding is applied. When the stride is greater than 1, the output size decreases. An example of a convolutional operation with stride 2 is given in Figure 2.9.

The pooling layer is the second main component of a CNN, these are commonly used between two convolution layers. The pooling operations reduces the dimension of each channel of the 3-dimensional frame at a given scale, therefore the number of and computations is greatly reduced too. Kernels with a square size of 2x2 along with a stride value of 2, is normally used for downsampling. The pooling layers takes the average value over the four neurons. After a pooling operation process with filter size 2x2 and a stride size of 2, the height and width of the input image will become half the original size. The only value that will remain the same is the depth, the color values, see Figure 2.10 for example of a pooling operation.

It is necessary to add a fully connected layer in a CNN, this layer will combine the local features learned by the convolutional and pooling layers into one output. With other words it is a way of learning non-linear combinations of the features. From this output can for example the classification operation be applied. Before the fully connected layer a flattening operation is applied, this converts the output of the previous layers into a single vector.

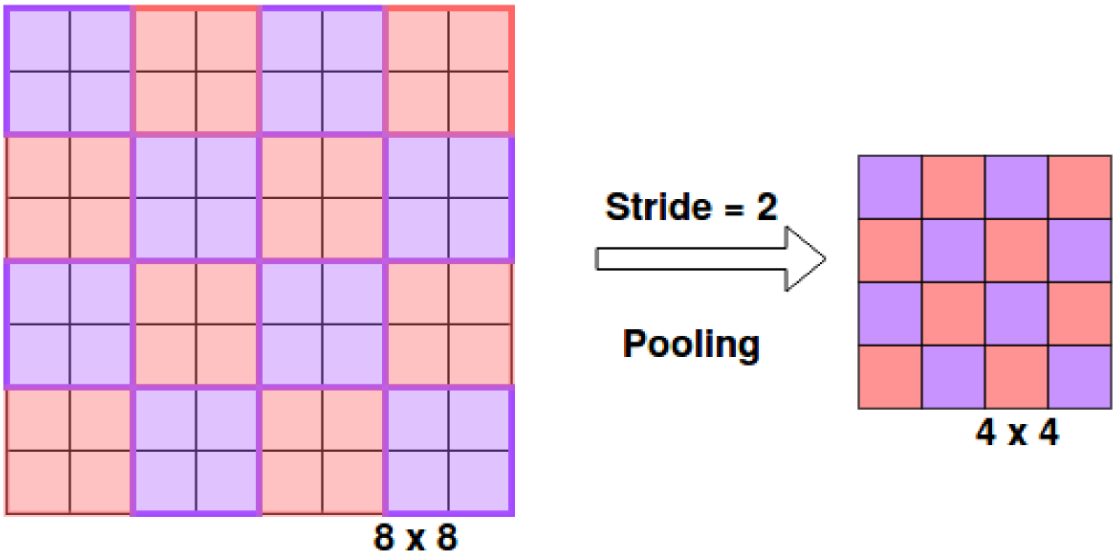


**Figure 2.10:** Example sketch of applying pooling operation with stride 2. The 8x8 layer comes a 4x4 after pooling with filter size 2x2.

## 2.9 Machine learning

Machine learning (ML) is a subfield of artificial intelligence, which can be defined as the capability of a machine to imitate intelligent human behavior. With the addition of vast amount of data and strong computational power these algorithm perform certain task better than a human being, such as object detection in images or audio, medical diagnosis, speech recognition, predictive analytics et cetera. The focus for this section will be on ML for audio, subjects as digital signal processing, feature engineering, filter banks, mel-scale and spectrograms will pass by.

Typically, building machine learning models to perform classification task, describe or synthesize audio involves modeling task with digital audio samples as input data. These digital audio samples are analog signals $x(t)$ that are converted to a digital signal $x[n]$ and stored by *filtering, sampling* and *quantization*. Filtering sets a limit to the frequency content of the signal in a interval $[0, B]$ with use of low-pass and high-pass filtering. The filtered analog signal is digitally sampled at a sampling rate of, at least, $f_s = 2B$ to avoid *aliasing*. Aliasing is the effect that causes different signals to become indistinguishable, or aliases of one another, when sampled. Finally, the digital signal is quantized so that the amplitude can only take values within a predefined set according to the bit depth, the resolution of each sample.

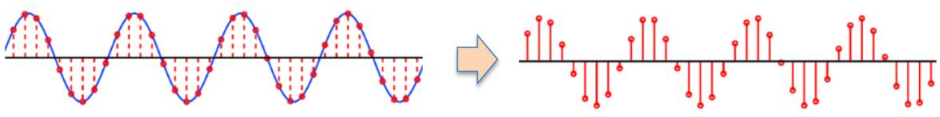

**Figure 2.11:** From an analog signal to a digital signal by sampling.

The audio signal is commonly represented in the time-frequency domain, whereas

in the time-domain the y-axis component is the amplitude of the signal, and in the frequency-domain, where each different frequencies are presented, the y-axis component is the magnitude representation. The most commonly used frequency-domain representation is the *spectrogram*. Both, time-domain and the spectrogram representation are given in 2.12, above the signal time versus amplitude [-], and below the signal time versus frequency [Hz]. The spectrogram gives a more easily interpretation of the characteristics of a signal, with respect to the time-domain plot. The spectrogram is usually depicted as a heat map, an image with the intensity of frequencies shown by varying color or brightness. The frequency and amplitude axes are either in linear or logarithmic scale, depending on its purpose. For audio the common practise is to represent the amplitude axis in decibels [dB]. These spectrograms are used extensively in fields of music, linguistics, sonar, seismology and speech processioning and others [47].

The operation to build spectrograms from a digital audio signal is performed on the bases of the mathematical Fourier series. To switch between both domains the Fourier Transform (FT) and its inverse version inverse Fourier Transform (iFT) are utilized. However, this operation is applicable for continuous signals, for the case of a digital audio signal the Discrete Fourier Transformation (DFT) is utilized. An algorithm that can perform this task is the Short-Time Fourier Transform (STFT), the long duration signal is divided in windowed time frames with fixed length $N$ and computed the DFT for each frame:

$$X[l,k] = \sum_{n=0}^{N-1} x_l[n] \cdot e^{-\frac{j2\pi}{N}kn} \quad k = 0, ..., N-1 \tag{2.10}$$

where $x_l[n]$ is a the windowed frame from a chunk of the original signal. This is obtained by multiplying the signal with a window $w[n]$ and a hop-size of $R$ between windows:

$$x_l[n] = w[n]x[n + lR], \quad 0 \leq n \leq N-1 \tag{2.11}$$

In the spectrogram there is a trade-off between temporal and spectral resolution, in which the feature engineer has to taken account for. Depending on the window size of the STFT the resolution can be determined, a larger window $w[n]$ results in a higher frequency resolution of that windowed signal, in the contrary the time resolution decreases.

In brief the STFT divides the signal in time frames of equal length, further the DFT computes or detects what the magnitude of the frequencies are within that frame are present, and gives as result a magnitude value. This operation is executed over the full length of the audio signal, and each computed DFT is then stacked in a temporal dimension, this results in a 2D representation (time-frequency domain) of the signal. This time-frequency domain representation is a spectrogram and an example is given in Figure 2.12 below.

In the field of machine learning for audio several alternative spectrograms have been introduced, each with a different purpose for a particular task. The basic

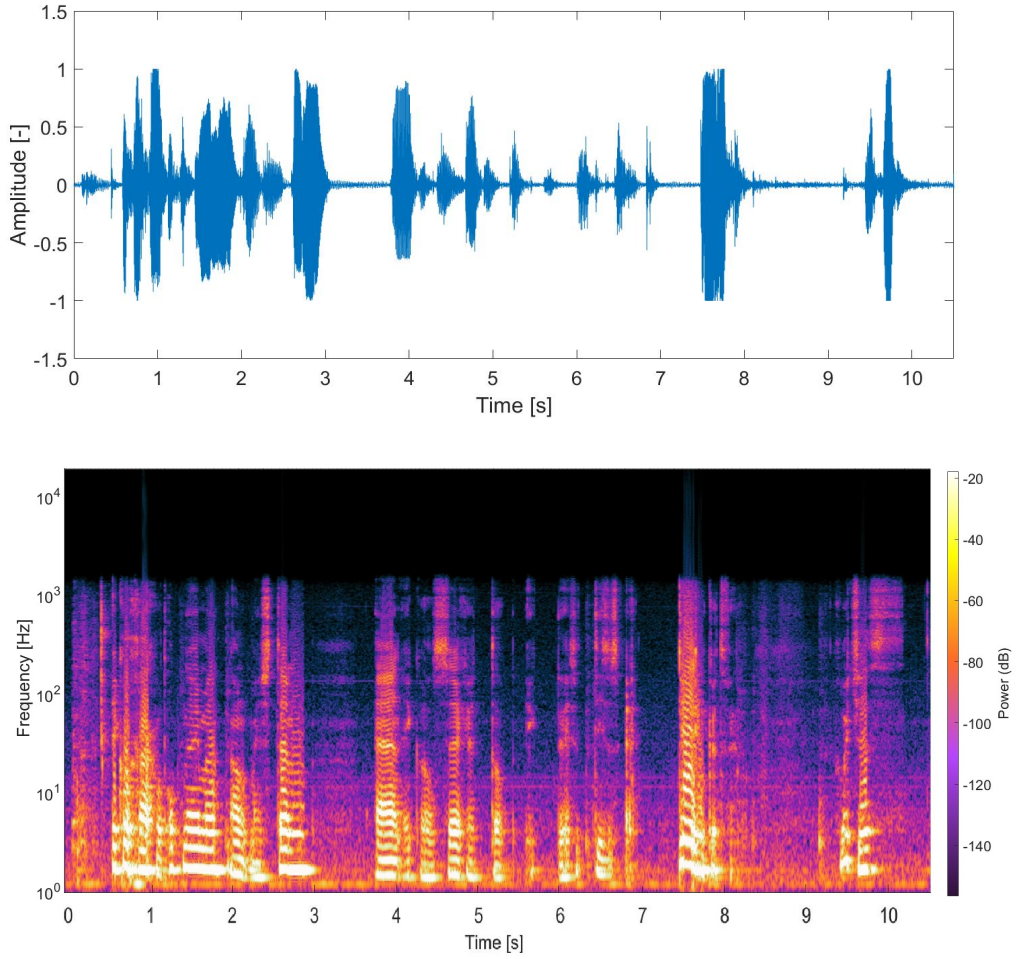


**Figure 2.12:** Above: Recording of a voice memo represented in the time-domain. Below: Spectrogram of the voice memo by applying the Short-Time Fourier Transform algorithm, for each position in time the frequencies magnitude of that time position is depicted in dB.

spectrogram based on STFT is a linear scaled, the frequencies are equally spaced over the axis. The human auditory system catches sound waves in a logarithmic manner with auditory filters in the cochlea, the range of hearing is from the low frequencies of 20 Hz up to 20000 Hz. Due to the logarithmic scale differences in lower frequency range are perceived more easily than in the higher frequency range. At 100 Hz to 110 Hz can be distinguished clearly, whereas 10000 Hz and 11000 Hz is hardly heard as two separate sounds. The auditory filters are known as critical bands and each consist of different center frequencies.

### 2.9.1 Mel-spectrogram

The mel scale is a set of pitches that human hearing perceives as being equally spaced apart. The interval in hertz between mel scale values (or simply mels) increases as frequency rises. The name mel comes from the word melody, and it denotes that the scale is based on pitch comparisons. The mel spectrogram remaps the values in

hertz to the mel scale.

The linear audio spectrogram is best for applications in which all frequencies are equally important, whereas mel spectrograms are better for applications in which human hearing perception must be modeled. Audio classification applications can also benefit from Mel spectrogram data.

In two ways, a mel spectrogram differs from a linearly scaled audio spectrogram. The first is that a mel spectrogram renders frequencies above a certain threshold in a logarithmic manner. The vertical space between 1,000 and 2,000Hz, for example, is half of the vertical space between 2,000Hz and 4,000Hz in a linearly scaled spectrogram. The distance between those ranges in the mel spectrogram is roughly the same. This scaling is similar to human hearing, in which we can distinguish between similar low frequency sounds more easily than similar high frequency sounds. The second way is that the output of a mel spectrogram is calculated by multiplying frequency-domain values by a filter bank.

The filter bank is constructed using a series of overlapping triangular windows at evenly spaced mels. Each window or filter has a response of 1 at the center frequency and decreasing linearly towards 0 until it reaches the center frequencies of the two adjacent filters where the response is 0. In a mel spectrogram, the number of elements in a single frame is equal to the number of filters in the filter bank.

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{2.12}$$

$$f = 700(10^{\frac{m}{2595}} - 1) \tag{2.13}$$

$$H_m(k) = \begin{cases} 0 & k < f(m-1) \\ \dfrac{k - f(m-1)}{f(m) - f(m-1)} & f(m-1) \leq k < f(m) \\ 1 & k = f(m) \\ \dfrac{f(m+1) - k}{f(m+1) - f(m)} & f(m) < k \leq f(m+1) \\ 0 & k > f(m+1) \end{cases} \tag{2.14}$$

The linear audio spectrogram and the mel spectrogram of the same linearly increasing and decreasing tone are shown in the image below. The tone begins at 20Hz and rises to 22,050Hz before returning to 20Hz. The image shows that the audio spectrogram represents the objective signal, but the mel spectrogram reflects human perception, with a flattening of the curve and reduced differentiation between high frequencies.

### 2.9.2 Acoustic features

Audio feature engineers have established many hand-crafted acoustic features for the application of machine learning in audio. The acoustic features can be utilized for

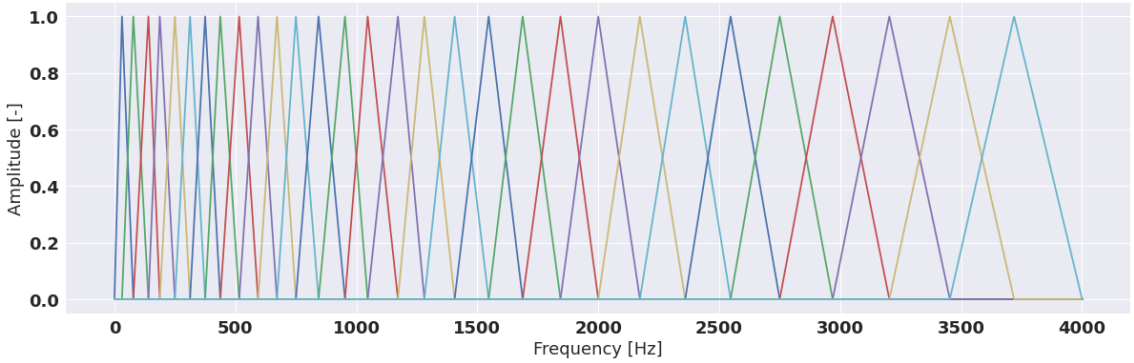


**Figure 2.13:** Mel filter banks. Filter banks try to capture the energy at each critical frequency band of the human hearing and roughly approximates the spectrum shape.

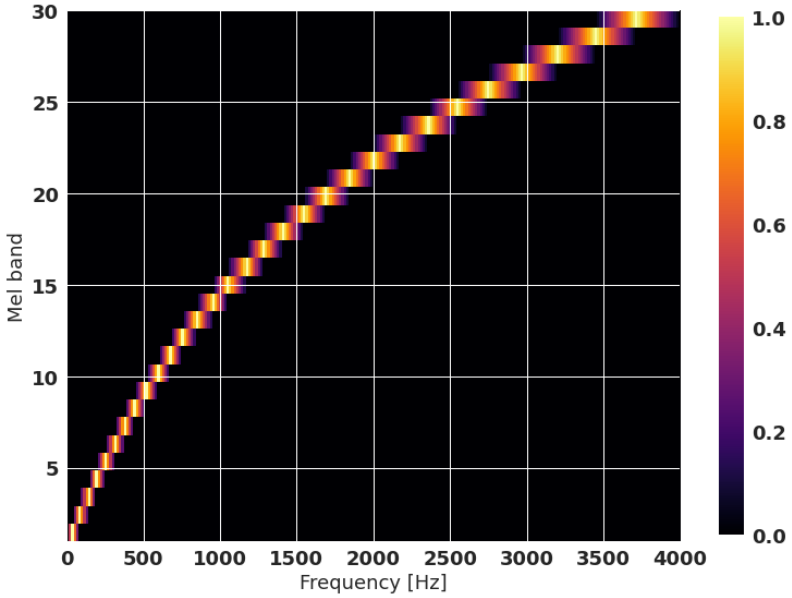


**Figure 2.14:** Mel filter banks depicted in a spectrogram.

classification tasks, for example speech recognition, natural soundscape detection, object detection or for music information retrieval tasks. In the music information retrieval field performed studies are beat or rhythm detection, key and chords analyses or harmonies and melody classifications. To extract information from audio the acoustic features are divided in two groups, time-domain and frequency domain audio features, which will be described in this section.

Short-term features versus mid-term features. The STFT algorithm is used to represent the audio signal in the frequency-domain, here the Fourier Transformation is applied which divides the signal into short-term frames (windows). Over these short-term frames, features can be extracted which result in a sequence of $F$ generated feature vectors per audio signal. In some cases, it can be desired to extract features in a longer time-scale scenario, where several short-term frames are concatenated to retrieve new information, with use of statistics. A MIR task that utilizes mid-term features is music genre classification, where a song is categorized whether it is a jazz, pop, rock or classical piece of music.

#### 2.9.2.1 Time-domain features

A time-domain graph shows how the signal changes with time and its amplitude size. These two characteristics are the bases for the time-domain audio features. In this section the features amplitude envelope, root-mean-square and zero-cross rate are described.

**Amplitude envelop**

The Amplitude Envelope (AE) refers to the changes in amplitude of a sound over time and attempts to extract the maximum amplitude from each frame and connect them, the amplitude of a signal represents its energy, and can be related to the loudness. The first step for retrieving the AE of a signal is to divide the signal individual windows and determine the maximum amplitude within each window. The maximum amplitude in each window is then plotted over time.

The AE can be used for onset detection, or detecting the start of a sound. This could be someone speaking or external noise in speech processing applications, but it could also be the beginning of a note or instrument in music information retrieval (MIR). Another property of sound that can be studied with AE is the timbre of a tone. Since every tone of a instrument has a different amplitude envelope over time, the timbre can be identified.

**Zero-crossing rate**

The Zero-Crossing Rate (ZCR) is the rate in which the amplitude sign of a signal changes from positive to zero to negative or from negative to zero to positive within a time frame. An audio signal is most often expressed with a mean of 0 and amplitude

range of 1 to -1. The ZCR is widely used in the natural language processing and MIR field for speech recognition and classify percussive sounds, respectively.

$$zcr = \frac{1}{T-1}\sum_{t=1}^{T-1} 1_{\mathbb{R}_{<0}}(s_t s_{t-1}) \tag{2.15}$$

where $s$ is a signal of length $T$ and $\mathbb{R}_{<0}$ is a indicator function.

#### 2.9.2.2 Frequency-domain features

Frequency-domain features are extracted from the spectrogram representation of a signal, these spectrogram can be in STFT, linear, non-linear, mel-scale et cetera. Depending on the purpose of the analysis.

**Spectral centroid**

Spectral centroid is a measure of spectral position, or in other words the center of 'gravity' of the spectrum. Perceptually, it has the impression of the brightness of a sound, see equation 2.16.

$$C_i = \frac{\sum_{k=1}^{Wf_L} kX_i(k)}{\sum_{k=1}^{Wf_L} X_i(k)} \tag{2.16}$$

**Spectral bandwidth**

The spectral spread represents the measure of spectral shape, it is commonly associated with the bandwidth of the signal. Individual tonal sounds with isolated peaks have usually a low spectral spread, noise-like signals a large spectral bandwidth, see formula 2.17.

$$S_i = \sqrt{\frac{\sum_{k=1}^{Wf_L}(k-C_i)^2 X_i(k)}{\sum_{k=1}^{Wf_L} X_i(k)}} \tag{2.17}$$

**Spectral flux**

The spectral flux is useful as measure for distinguishing signals whose spectrum changes quickly from signals whose spectrum changes slowly. To compute the spectral flux two successive frames are compared in spectral change, by taking the squared difference between the normalized magnitudes of the spectra of the successive short-term windows, see formula 2.18 below.

$$Fl_{(i,i-1)} = \sum_{k=1}^{Wf_L} (EN_i(k) - EN_{i-1}(k))^2 \tag{2.18}$$

$$EN_i(k) = \frac{X_i(k)}{\sum_{l=1}^{Wf_L} X_i(l)} \tag{2.19}$$

where $EN_i(k)$ is the $k$th normalized DFT coefficient at the $i$th frame. Spectral flux can be used as indicator whether the signal is music or speech, since the local spectral changes are more present in speech signals due to the rapid alternation among phonemes. Some phonemes are quasi-periodic, whereas other are of a more noisy nature [60].

### Spectral flatness

Spectral flatness is a quantification how much tone-like a sound is, as opposed to being noise-like. A high spectral flatness, closer to 1, indicates that the signal is similar to white noise. Spectral flatness is also known as tonality coefficient and typically measured in decibels. It is defined as the ratio of the geometric mean to the arithmetic mean of a power spectrum, see below equation 2.20.

$$Flatness = \frac{\sqrt[N]{\prod_{n=0}^{N-1} x(n)}}{\frac{\sum_{n=0}^{N-1} x(n)}{N}} = \frac{\exp\left(\frac{1}{N}\sum_{n=0}^{N-1} \ln x(n)\right)}{\frac{1}{N}\sum_{n=0}^{N-1} x(n)} \tag{2.20}$$

where $x(n)$ represents the magnitude of bin number $n$. The tonality coefficient has been utilized in birdsong research, when testing similarity between two excerpts [59]. Furthermore, it has been used in the analysis of EEG diagnostics and research [60].

### Spectral rolloff

The roll-off frequency is the frequency at which a certain percentage (cutoff) of the spectrum's total energy is contained. It is possible to tell the difference between harmonic (below roll-off) and noisy sounds using the roll-off frequency (above roll-off). See equation 2.21 below.

$$\sum_{k=1}^{m} X_i(k) = C \sum_{k=1}^{Wf_L} X_i(k) \tag{2.21}$$

## 2.10 Normalized cross-correlation

Cross-correlation is a measure of the similarity of two sequences as a function of the displacement of one relative to the other. The normalized cross-correlation formula [65] is useful to find correlation between two vectors $x[n]$ and $y[n]$ . The vectors should be of equal length. Normalizing makes it possible to compare two vectors of different value ranges, it will more look at the trend of the vectors than actual values. The normalized cross-correlation equation is given below 2.22.

$$norm\_corr(x, y) = \frac{\sum_{n=0}^{n-1} x[n] * y[n]}{\sqrt{\sum_{n=0}^{n-1} x[n]^2 * \sum_{n=0}^{n-1} y[n]^2}} \tag{2.22}$$

# 3
# Methodology

This section is devoted to the methodology of this thesis work, containing the research approach, the computational model, an overview of Huang's [31] work and adjustments made to it to fit this thesis project. Huang's work is the basis of this thesis work. Next the implementation of the project is described including the used dataset, acoustic features, deep neural network VGGish, EEG signal pre-processing steps and tools, the network surprisal and the correlation analyses method applied.

## 3.1 Research approach

This research work started with the question whether it was possible to conduct an analysis on how the human processes music with respect to attention, bottom-up and top-down attention. In the literature research part it came forward that artificial neural networks are able to predict higher semantics properties of text and speech. Higher semantics properties are top-down attention characteristics and were retrieved from the last layers of a neural network. The first layers of the network were related to low-level features, these can be referred as bottom-up attention attributes. From these findings of the neural work it can be concluded that it was possible to analyse audio salience from the bottom-up and top-down attention approach. The next step was how to setup such a network, what data was required, what were the parameters to be set and what metrics were needed to come down to reliable result?

## 3.2 Computational Model

In sake of the limited time of this thesis work and the inexperience in the field, the work of this project was based on a recently, 2018, published paper by Huang et al. [31], *"Connecting Deep Neural Networks to Physical, Perceptual, and Electrophysiological Auditory Signals"*. The questions in the previous section were largely answered by Huang's work, e.g. what input data was needed, how did the computational model look like and how could the output of the model be evaluated? All of this was drawn together in a flowchart to get an better overview, see Figure 3.1. The flowchart shows the computational model consisting of mulitple frameworks. The frameworks are acoustic features extractor, pre-trained deep neural network VGGish, EEG signals preprocessor and the correlation tool.

The largest change of this project compared to Huang's work is the input data type, instead of using natural soundscapes, this work focuses fully on musical audio data.

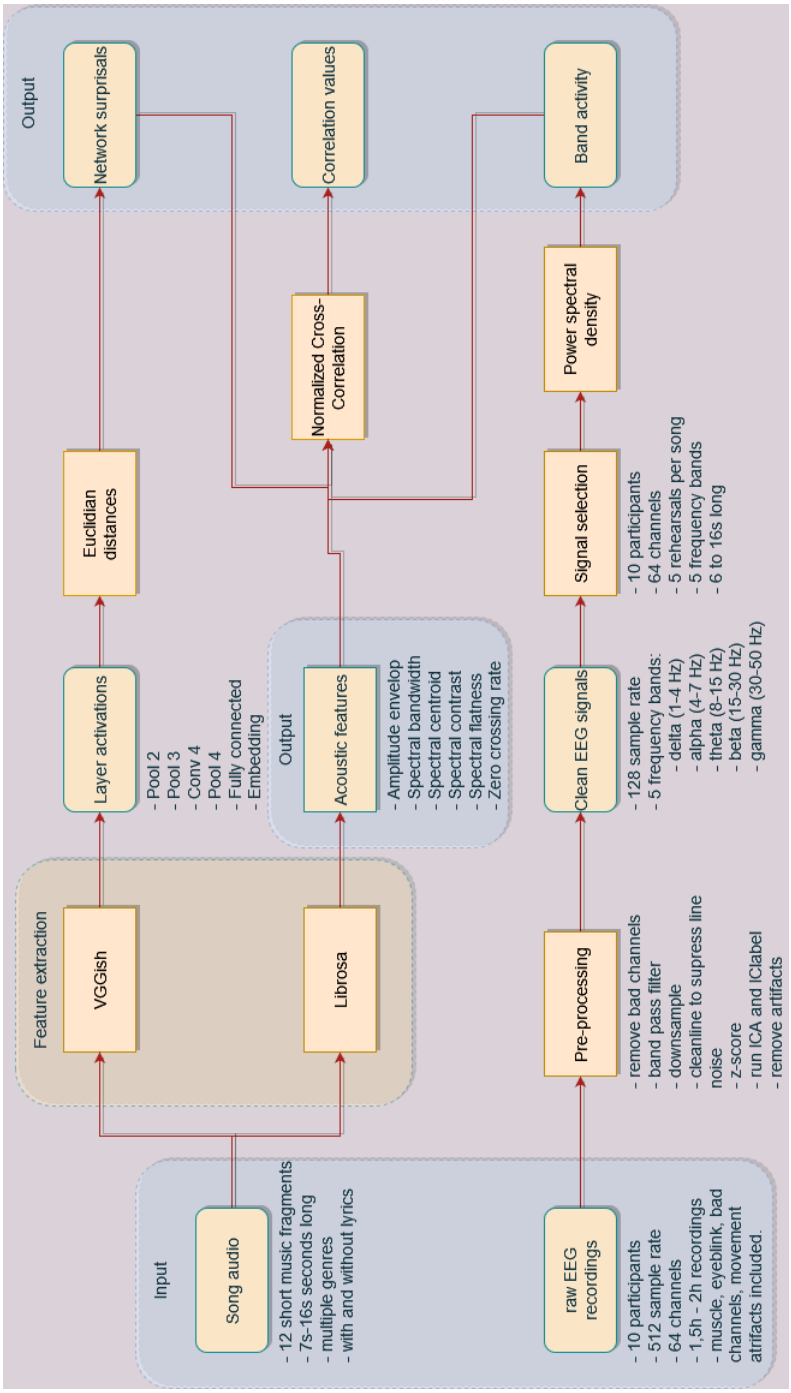


**Figure 3.1:** Pipeline of the computational model. Input and output data are shown in blue boxes, operations in rectangular boxes, and the data type in rounded boxes are shown.

### 3.2.1 Brief overview Huang's work

Huang's work consists of collecting data from their own conducted listening experiment. Collecting and creating acoustic features from sound fragments of their build database. Analyzing and annotating behavioral salience from the sounds, music and speech fragment used in the experiment. EEG brain signals recorded while the participants were listening to the various audio fragments. From these EEG signals the brain activity per song, channel and brain region was known, the differences in brain activity was taken as indicator of the presence of auditory salience. More explanations are described further in this chapter. Designing, building and training a deep neural network with audio as input. Defining the *network surprisal* in a deep neural network (CNN) based on activation patterns within a layer of the neural network. More about network surprisal at the end of this chapter. Second to last, setting up a method to find the correlation between the acoustic features, behavioural salience, brain wave activities and the network surprisal of the CNN. Finally a metric to state prediction of salient events.

As it was be concluded from the steps described above, it was unrealistic to conduct such a work with the given limited time period, COVID-19 situation and the inexperience with the tools, tools such as to execute a full listening experiment, pre-processing EEG signals and training CNN's, therefore short-cut solutions were made. The following section will describe which approach was taken to make the work more feasible.

## 3.3 Method modifications

One of the most important and also the most time consuming part of conducting a research with training deep neural networks is data collection. Data used for training should consist a sufficient amount, clean, well ordered and labelled dataset. Due to COVID-19, experiments with participants became problematic, therefore it was appointed to retrieve the data from existing datasets. A wide range of datasets are public available, as well a dataset (OPENMIIR) from an experiment with participants listening to music pieces while EEG recordings where made. The dataset is annotated and structured, and music pieces were included. More details of the dataset will be described further in this chapter.

Acoustic features, such as loudness, brightness or spectral flatness, are commonly known features in the literature. Both Huang's work and this report uses existing acoustic features from known literature work. These acoustic features can be taken from several software libraries, instead of using the NSL MATLAB TOOLBOX like Huang's work, this work utilized Python's Librosa library for audio processing tools [17].

The EEG brain recordings from the OPENMIIR dataset were raw 64 channel signals, in order to utilize this data it should be cleaned first by performing pre-processing steps. The dataset came with scripts to pre-process the EEG data, including down-

sampling, applying bandpass filters, re-reference data, reject noisy components and epoch the data. However the scripts relied on several deprecated libraries which were unable to be restored, and thus pre-processing with these scripts was not possible. Therefore pre-processing the raw EEG data was performed by own implementation. The software used to clean the raw EEG data was EEGLab, together with aid from the Bioengineering department of Polimi professor-assistent Alessandra Calcagno, this was realized.

The main part of the research was designing and creating a computational model, more specific a convolutional neural network that can take audio samples of music as input. To safe time and take a way the uncertainty if the model was reliable enough, an existing network is used. There are a few pre-trained networks trained on audio fragments out open for public, one of the most commonly used is the VGGish network pre-trained by Google. The model can be downloaded with the trained weights, and via transfer learning the model can be adjusted to fit this research work. Another benefit of this model that it is available as well in Matlab as Python, where the latter will be utilized.

The validation of all the outcomes of the computations, acoustic features, EEG signals and network surprisal, was imitated from Huang's paper. Few adjustments were made to make it more suitable for the song fragments of the OPENMIIR dataset, for example the delay of the network surprisal was set to 1 second instead of 3 seconds.

The following sections will describe the implementations of the research approach, with more detailed description of the adjustments of the method.

## 3.4 Implementation

### 3.4.1 OpenMIIR database

The database used for this thesis work, OpenMIIR [56], is a public dataset made available for the music information retrieval (MIR) research field, more specific for the sub-field music imagery information retrieval (MIIR). The OpenMIIR dataset is a result of ongoing joint work between the Owen Lab and the Music and Neuroscience Lab at Brain and Mind Institute of University of Western Ontario [63].

The aim of the dataset is to retrieve information from brain activity recorded during music cognition, such as listening to or imaging music fragments. This requires a highly inter-disciplnary endeavor from expertise in MIR, as well in cognitive neuroscience and psychology.

The data of the OpenMIIR database consist of several music fragments and EEG records of subjects listening and imagining these music pieces. The music stimuli are 12 short plays taken from well-known songs, each 7s-16s long, see Table 3.1 at the next page for an overview. The stimuli were chosen from various genres and systematically span several musical dimensions such as tempo, meter and the presence

of lyrics. This way, it possible to apply various retrieval and classification operations from the data. However, the music excerpts contained several clicks in the beginning of the track, these were cut-off to get clean data for the VGGish neural network.

The EEG recordings were obtained using a Biosemi Active Two 64-electrode array, initially with a sample rate of 512 Hz. The size of each recording is approximately 700 Mb. The 12 songs can be divided in three groups:.

- 4 records of songs with lyrics
- 4 records same song without lyrics
- 4 instrumental pieces

The dataset consist of four different experiments, where for this work only the Stimulus perception preceded by cue clicks is used:

1. Stimulus perception preceded by cue clicks
2. Stimulus imagination preceded by cue clicks
3. Stimulus imagination without cue clicks
4. Stimulus imagination without cue clicks, with feedback

The raw EEG recordings were not clear of noise from example, muscle movements, bad channels or bad recordings sections. Therefore it was required to clean the data with pre-processing tools for EEG signals, what tool and how this was done is explained in the EEG pre-processing section of this chapter.

| ID | Name | Meter | Length | Tempo |
|---|---|---|---|---|
| 1 | Chim Chim Cheree (lyrics) | 3/4 | 13.3s | 212 BPM |
| 2 | Take me out to the ballgame (lyrics) | 3/4 | 7.7s | 189 BPM |
| 3 | Jingle Bells (lyrics) | 4/4 | 9.7s | 200 BPM |
| 4 | Mary Had a Little Lamb (lyrics) | 4/4 | 11.6s | 160 BPM |
| 11 | Chim Chim Cheree (no lyrics) | 3/4 | 13.5s | 206 BPM |
| 12 | Take me out to the ballgame (no lyrics) | 3/4 | 7.8s | 185 BPM |
| 13 | Jingle Bells (no lyrics) | 4/4 | 9.0s | 200 BPM |
| 14 | Mary Had a Little Lamb (no lyrics) | 4/4 | 12.2s | 160 BPM |
| 21 | Emperor Waltz | 3/4 | 8.3s | 175 BPM |
| 22 | Harry Potter Theme | 3/4 | 16.0s | 166 BPM |
| 23 | Star Wars Theme | 4/4 | 9.2s | 104 BPM |
| 24 | Eine Kleine Nachtmusik | 4/4 | 6.9s | 140 BPM |
|  | mean |  | 10.4s | 176 BPM |

**Table 3.1:** List of songs in the OPENMIIR database.

### 3.4.2 Acoustic features

In this section the method to construct acoustics features for the songs of the OPENMIIR database will be explained. The features are used to analyze acoustic markers

of auditory salience. To extract such acoustic features the spectral, temporal and spectro-temporal attributes of the audio fragments are analyzed.

The feature-extraction method compute short-term spectral parameters, such as the mel-frequency ceptstral coefficient (MFCC) and spectrum, but as well the temporal modulation of the signals amplitude. Therefore simple statistics, such as mean, variance and distribution are conducted to determine the acoustic features. The feature extraction is performed using the Python[1] programming language, with specific the Librosa[2] library, a package for music and audio analysis.

The acoustic features utilized in this work are:

- Amplitude envelope
- Spectral bandwidth
- Spectral centroid
- Spectral contrast
- Spectral flatness
- Zero-crossing-rate

The first and the last acoustic feature, amplitude envelope and zero-crossing-rate, are temporal features and the other spectral features. The mathematical formula of these acoustic features can be found in the Theory chapter. The results of acoustic features from the songs of the OPENMIIR database are shown in the Results chapter.

### 3.4.3 VGGish

The VGGish network is a pretrained convolutional neural network model built by Google. The architect of this network is inspired by the well-known VGG networks used for image classification. VGGish main usage is a feature extractor for audio fragments, but can also used as audio classification, and is commonly used most because of its great performance, even for its age. The dataset where the network is trained on is the AudioSet dataset, which consists of 2.1 million videos, 5.8 thousand hours of audio and 527 classes of annotated sounds [22]. The weights of pre-trained model are publicly available and can be downloaded. The VGGish network and its weights for this work were taken from the Github page of Tensorflow [30], a free and open-source software library for machine learning and artificial intelligence.

The VGGish model structure can be found in Table 3.2. The input of the model are 96x64 dimensional mel spectrogram non-overlapping segments of audio clips. There are four blocks of max-pooling layers and two-dimensional convolution in the VGGish architecture. The embedding vector is generated by the final max-pooling layer, which is followed by two fully connected layers of 4096 units each and a final

[1] https://www.python.org/
[2] https://librosa.org/

| Layer | Activation Size |
|---|---|
| input | $1 \times 96 \times 64$ |
| $64 \times 3 \times 3$ conv, stride 1 | $64 \times 96 \times 64$ |
| $2 \times 2$ maxpool, stride 2 | $64 \times 48 \times 32$ |
| $128 \times 3 \times 3$ conv, stride 1 | $128 \times 48 \times 32$ |
| $2 \times 2$ maxpool, stride 2 | $128 \times 24 \times 16$ |
| $256 \times 3 \times 3$ conv, stride 1 | $256 \times 24 \times 16$ |
| $256 \times 3 \times 3$ conv, stride 1 | $256 \times 24 \times 16$ |
| $2 \times 2$ maxpool, stride 2 | $256 \times 12 \times 8$ |
| $512 \times 3 \times 3$ conv, stride 1 | $512 \times 12 \times 8$ |
| $512 \times 3 \times 3$ conv, stride 1 | $512 \times 12 \times 8$ |
| $2 \times 2$ maxpool, stride 2 | $512 \times 6 \times 4$ |
| flatten | $1 \times 12288$ |
| fully connected I | $1 \times 4096$ |
| fully connected II | $1 \times 4096$ |
| output | $1 \times 128$ |

**Figure 3.2:** Structure of the VGGish model consisting of convolutional, max pooling, flatten and full connected layers. A total of 72 million parameters contains the model.

fully connected layer of 128 units. The fully connect layer of 128 units is known as the embedding layer. ReLU activation function [3] is utilized by each and every fully connected and convolution layer. The model consist of a total 72 million parameters.

The correlation analyses between the network surprisal and the acoustics features and EEG energy bands is performed using only the pool 2, pool 3, conv 4, pool 4, full connected layer 1 and the embedding layer. This is based on Huang's work were they made a choice for these layers. The chosen layers are equally spread over the network, so it might be possible to see a trend from the early layers up to the deeper and last layer in the correlation analyses.

### 3.4.4 EEG pre-processing

The cortical activity recordings in the OpenMIIR database were exploited by EEG. The EEG recordings are raw records, meaning artifacts such as bad electrodes, eye blinks, muscle movement, etc. are included. For further usage for this studies the EEG data should be cleaned in order to be used. The most common used software to analyze and edit EEG signals is EEGLab analysis tool in MATLAB, together with the plugins ICLabel v1.3 and Viewprops v1.5.4. The pre-process steps are explained in the following section.

The raw EEG in the OpenMIIR dataset is saved in the FIF file format used by MNE and MNE-python [16]. However the MNE software is not commonly used in the field of EEG processing, therefore little documentation is available. That is why

EEGLab is chosen as tool to analyze and edit the raw data for this studies. A script is provided by the dataset to convert the FIF files into MAT file format so it can be loaded in MATLAB.

#### 3.4.4.1 Channels

The first step when the data is loaded into EEGLab is to check whether the channels are right. Remove channels that are useless and set the location of the channels and their names right. The raw data is provided with 4 EOG channels that recorded the eye muscle movements, which are not of interest, and are removed. The location of each channel is of importance to check whether the position of the electrode is on the right spot on the scalp of the subject. The locations of the electrodes are given by their label name and polar coordinates file *.loc*. The locations with their code names and their relation with the brain regions can be seen in Figure 3.3.

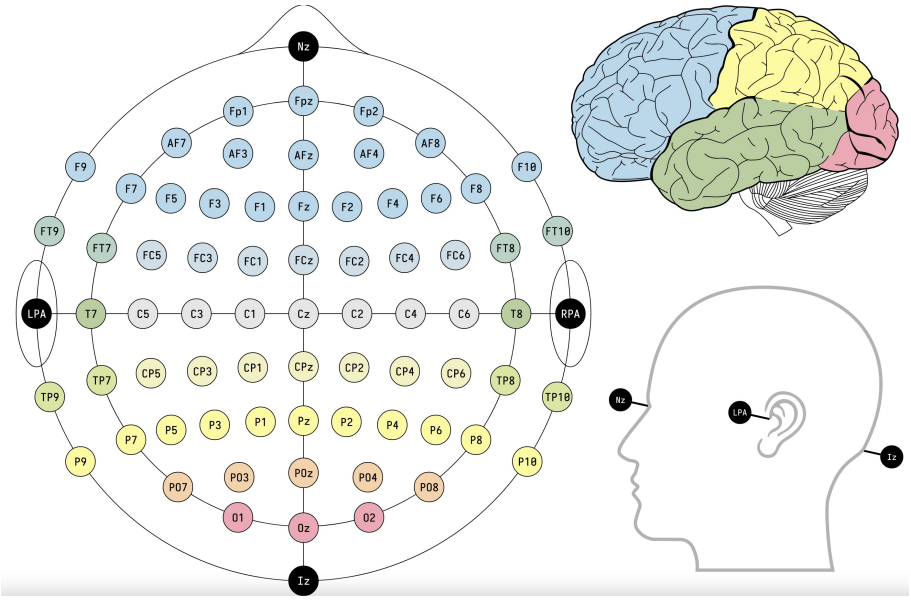


**Figure 3.3:** Locations of the channels shown from a topview of a human head. Each channel has it code name referring to the brain region and coordination, even numbers on the left hemisphere, odd on the right.

#### 3.4.4.2 Band pass filter and downsampling

The 64 channels are loaded with their locations, the following step is to apply a FIR filter to remove frequencies of the signal that are not of interest. The electrodes can detect frequencies that are out of the scope of brainwave frequencies, which lay between 1 Hz and 110 Hz. A high pass filter of 1 Hz and low pass filter of 50 Hz are applied for each of the channels.

The sampling rate is reduced in order to reduce the disk storage and memory usage. The original data with sampling rate 512 Hz is downsampled to 128 Hz. Which is above the low pass filter frequency to avoid aliasing.

#### 3.4.4.3 Bad channel and data detection

Each of the signals can be analyzed with the *Plot* $\Longrightarrow$ *Channel data (scroll)* option in the EEGLab menu. A new window pop-ups where the 64 channels are presented with their EEG data. It is possible to scroll through data from start to end to see whether the signal does not align with a "normal" looking EEG signal. While inspecting the data it can be noted that certain channels have a "flat line", this is the result of a bad connected electrode which give the wrong signal to the computer. It can also happen that the fluctuations are not in parallel with its neighbour channels, this is also an indication of a bad channel. A second option, instead of looking to the amplitude of the signal over time, is to look at the spectrogram where the power spectral density is depicted against frequency in Hz, see Figure 3.4. The signals that have a deviation compared to the collective channels have to be manually removed.

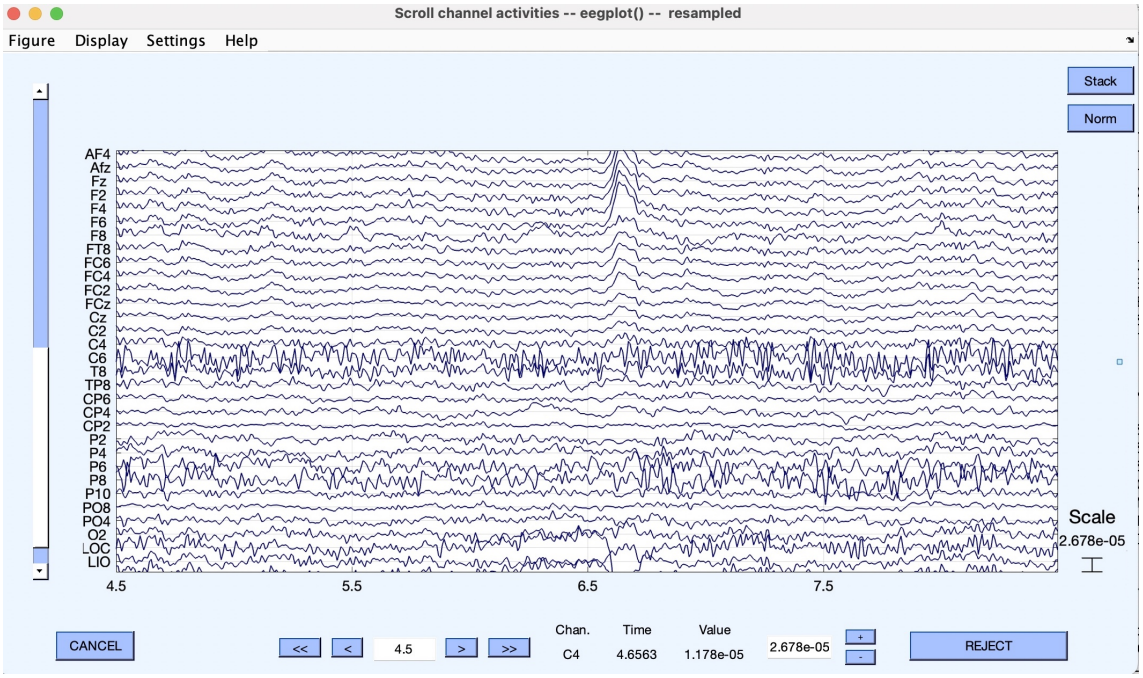


**Figure 3.4:** 32 EEG signals depicted in the EEGLab software tool. X-axis in msec and y-axis the annotated EEG electrodes. Noisy signals, e.g. C6 and T8 are recognizable and at t=6.6 some event is present. Event could be eye movement, or an event-related potential.

Within EEGLab and its plugins it is possible to use automated rejection tools for bad channels or data. However these work not perfectly and therefore self-analyzing and editing is required. The downside by removing channels or parts of the data is that potential information might be lost lost as well. Instead of removing the full data portion, an independent component analysis (ICA) can be applied which can save the potential information. The use of ICA is explained in the following section.

#### 3.4.4.4 ICA and IClabel

Independent Component Analysis (ICA) is used to subtract artifacts embedded in the data (eye blinks, muscle or eye movements) without removing the affected data

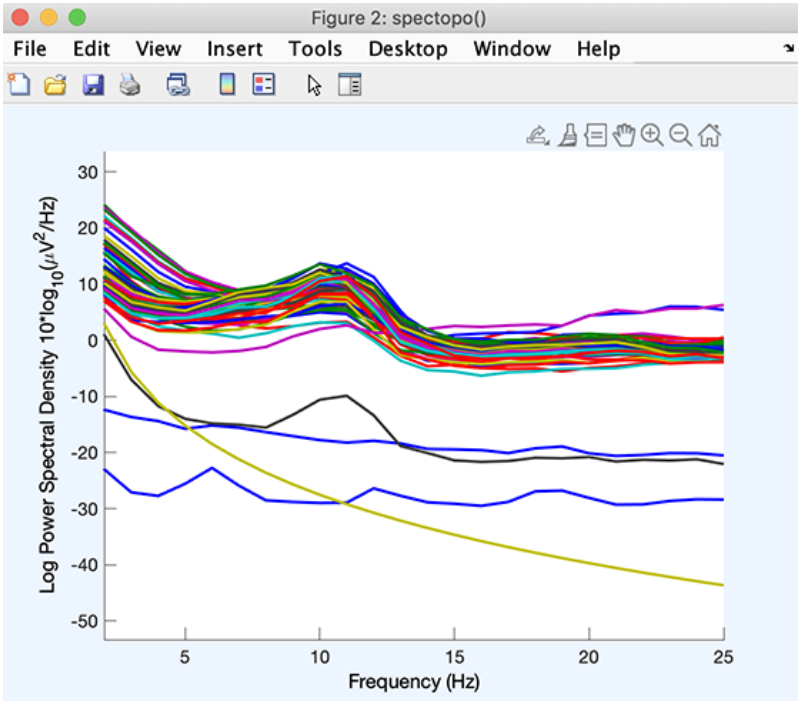


**Figure 3.5:** Spectrum plot of 64 channels of the EEG electrodes. Lines that are off from the average, are bad or broken EEG electrodes and needs to be removed.

parts. Different ICA decomposition algorithms can be chosen in EEGLab: Infomax ICA, Jadar, SOBI, which each have a different approach to calculates independent components in the EEG data. The independent components are then compared with well-known artifacts to determine if it components are for use or not, for example if it are brain waves or eye blink movements.

The Infomax ICA algorithm is ran for each subject, after the previous pre-process steps are made. To compare the found independent components with well-known components the IClabel plugin is applied. IClabel labels components on brain, muscle, eye, heart, line noise, channel noise or other. By setting a threshold value for each of the labels, components that exceed this threshold will be removed. The chosen thresholds are given in Table 3.2:

| | **Brain** | **Muscle** | **Eye** | **Heart** | **Line noise** | **Channel noise** | **Other** |
|---|---|---|---|---|---|---|---|
| **min** | 0 | 0.7 | 0.7 | 0.7 | 0.7 | 0.7 | 0.9 |
| **max** | 0.05 | 1 | 1 | 1 | 1 | 1 | 1 |

**Table 3.2:** Thresholds for the different components that could be present in the EEG signals, and will be filtered with the ICA algorithm.

#### 3.4.4.5 Interpolate and re-reference

The removed channels or components provide that data is missing which can lead to misleading data visualization of the EEG signals, i.e. scalp maps. By retrieving

channel information of the LOCS file, removed channels after are detected. The interpolate tool can be found by *Tools* $\longrightarrow$ *interpolate the data* in the EEGLab menu to restore rejected channels.

An important step is to re-reference the EEG recordings, since EEG signals are relative changes in voltages to a reference electrode. Most typically used reference electrode is one mastoid, noted as TP10 or CZ, the most central electrode. Since the OpenMIIR dataset did not reference for all the subjects to the same electrode, average re-referencing is applied. This way multiple references are tried by EEGLab and the best fit is applied to re-reference all the channels.

### 3.4.5 Frequency band activity

In this study the frequency band activity is of interested for the five different EEG frequency bands. The five bands are delta (1-4 Hz), theta (4-7 Hz), alpha (8-15 Hz), beta (15-30 Hz) and gamma (30-50 Hz). These bands are collected in MATLAB after the pre-processing steps are executed. The bands are retrieved by using the *bandpass()* MATLAB command which consist of a minimum-order filter with a stopband attenuation of 60 dB. The delay that is introduced by the filter is compensated. The sample rate is set equal to the downsampled EEG data of 128 Hz.

After the band pass filtering, the frequency bands are z-score normalized for each channel, to set the each band to a common scale. Next step is to convert the time signal to energy. This is performed by multiplying the signal with its conjugate.

Instead of using the bandpass function, the filter-Hilbert method [4] is as well applied to check whether the filtering was done reasonable. Both methods showed the same results, therefore the bandpass function method was taken for this project. The steps of retrieving frequency band activity are shown in Figures 3.6, 3.7 and 3.8, 3.9. In the first two figures the original EEG signal of channel 1 is shown for the sample number 5000 till 10000. In this segment it can be clearly seen that some brain activity is present, there is no constant noise. The top plot of Figure 3.6 shows the time signal where all the frequencies are present, with the two other plots the delta and theta. Figure 3.7 shows the other three plots of the frequency bands, alpha, beta and gamma. The latter two Figures 3.8, 3.9 show the frequency bands when the z-score and multiplication with its conjugate are applied. From these signals the band activity was taken to find correlation with the other outputs of the computational model, network surprisal and acoustic features.

### 3.4.6 Network surprisal

This section describes the chosen method for determining the activity change in the neural network layers of the VGGish network over time. The change of activation patterns within a layer of the CNN is defined as "network surprisal". See activations of the layers in Appendix B. It represents an estimate of variability in the response pattern across all nodes of a given layer in the VGGish network and it quantifies

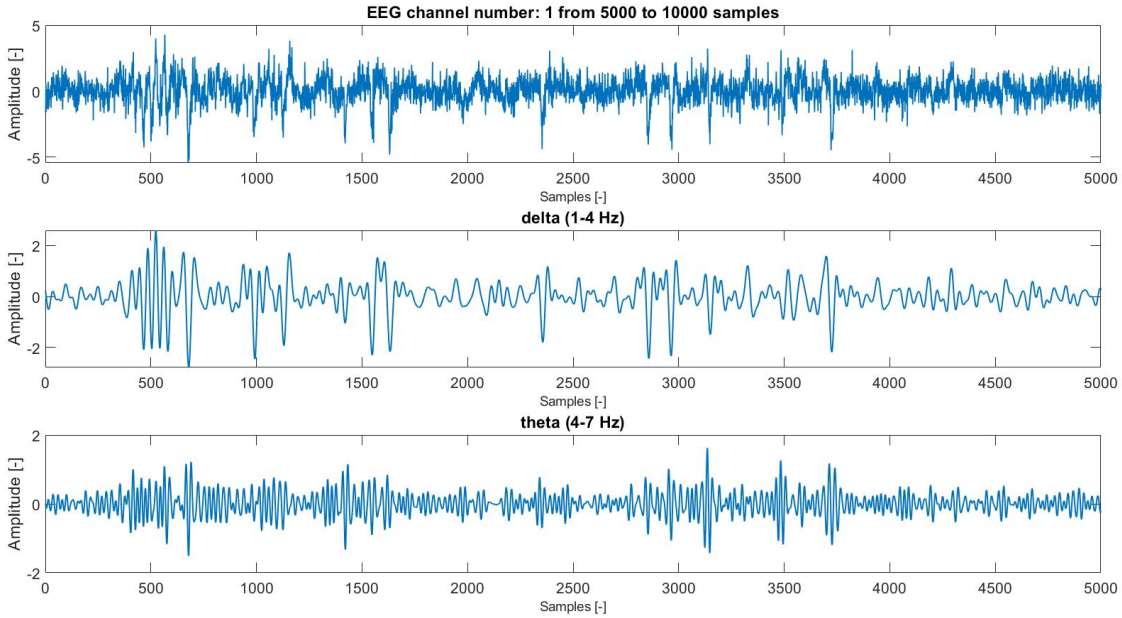


**Figure 3.6:** Frequency bands activity of EEG channel 1 from sample number 5000 to 10000. Top plot shows all frequencies combined, second and third plot show frequencies filtered for the delta (1-4 Hz) and theta (4-7 Hz) range.

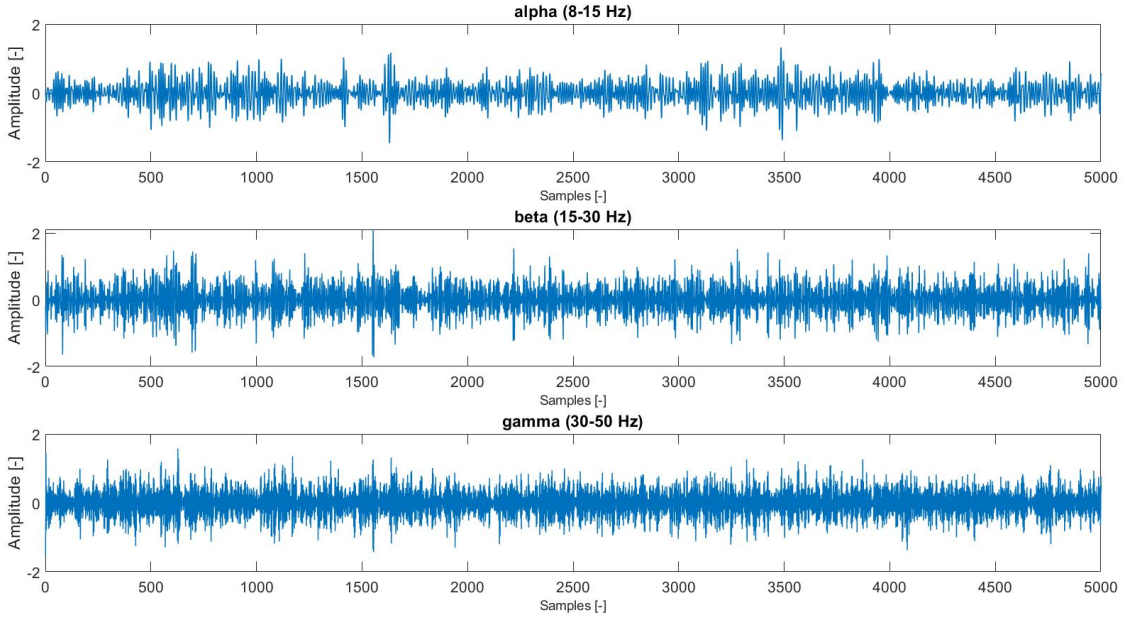


**Figure 3.7:** Frequency bands activity of EEG channel 1 from sample number 5000 to 10000. Frequency bands alpha (8-15 Hz), beta (15-30 Hz) and gamma (30-50 Hz) are shown.

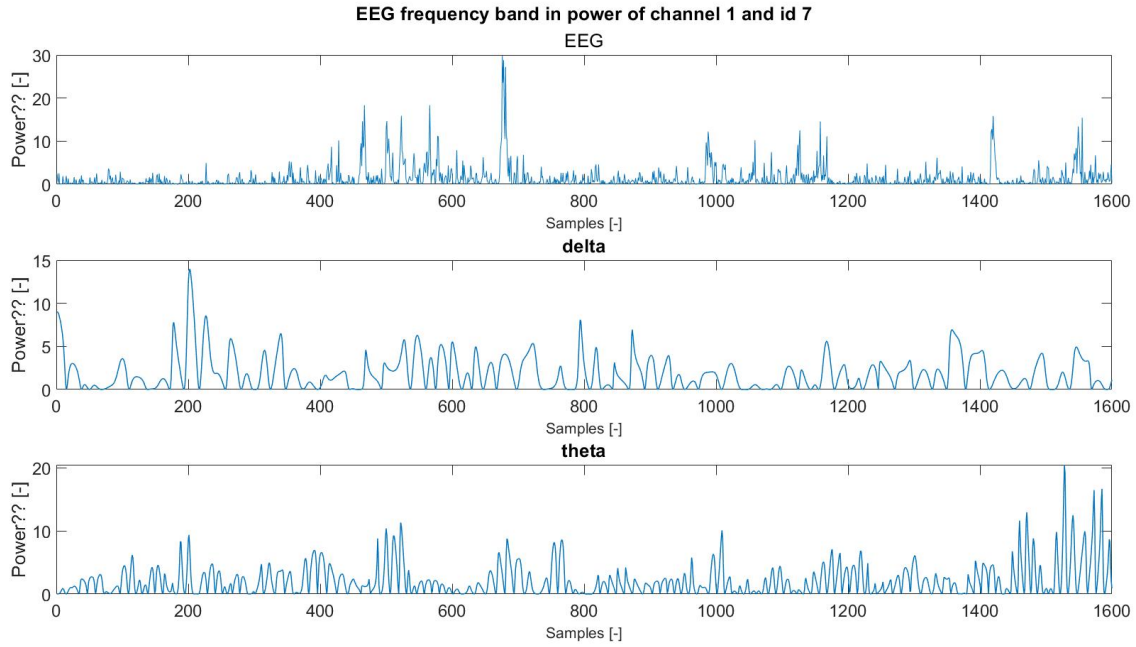


**Figure 3.8:** Frequency bands activity of EEG channel 1 and song ID 7. All frequencies together (top signal) and the delta and theta frequencies bands (middle and bottom signals) are shown. These signals were used as input for the correlation computations with network surprisal and acoustic features.

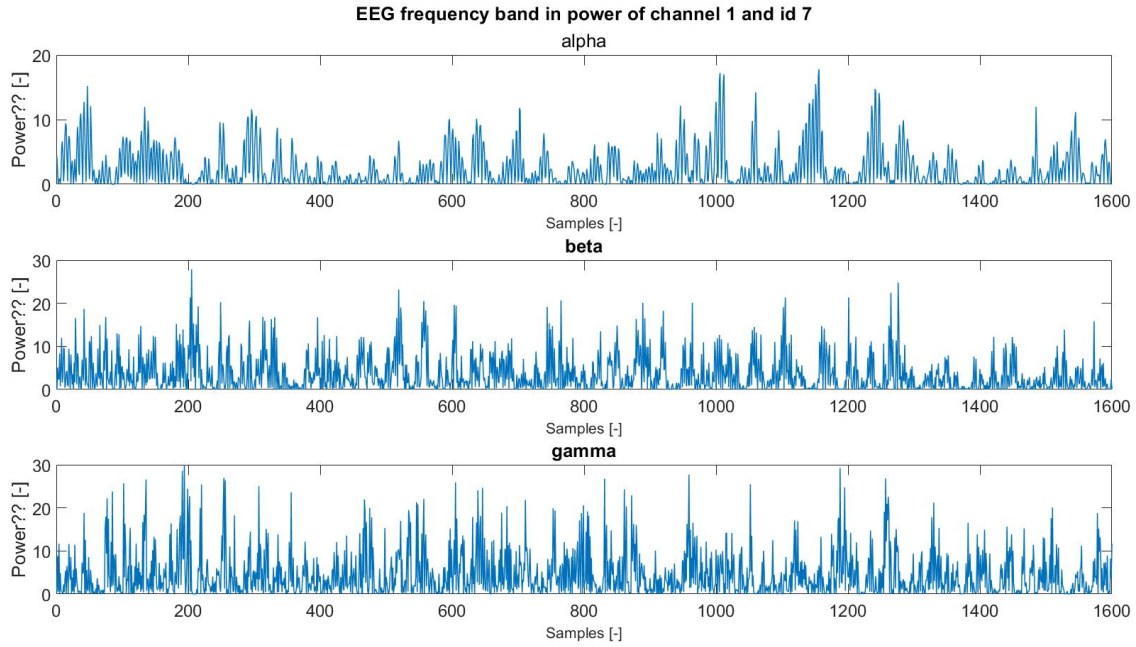


**Figure 3.9:** Frequency bands activity of EEG channel 1 and song ID 7. Alpha, beta and gamma bands are shown. These signals were used as input for the correlation computations with network surprisal and acoustic features.

the surprising activity at a given moment when compared with the activity of the previous time period. To establish a metric for network surprisal the Euclidean distance between activity in a layer at a given time bin and the average activation of the previous x layers is computed. This method is a common metric for evaluating dissimilarity in neural network activity, see Krizhevsky et al. [1].

### 3.4.7 Correlation analyses

The music audio excerpts, EEG, and CNN data have all been reduced to low-dimensional features. The audio is represented by six distinct acoustic measures, the energy in six distinct frequency bands for the 64 channel EEG measurements, and the surprise measure for the multichannel outputs from the six distinct layers of the CNN. The correlation between these metrics and the activations of the neural network is then examined. Also the resolution of the data is reduced, this due the fact that the input size of the VGGish network was fixed. The input size was 96 frames of 10 msec, or about 1 second. To do further correlation analyses between the audio, CNN and EEG signals, all the lengths of the outputs should be equal. Therefore the EEG energies, acoustic features were adjusted to match the resolution of the CNN output of 1 sec length.

Normalized cross correlation is used to compare each layer of the neural network to fundamental acoustic features, and energy in EEG frequency bands. The normalized cross-correlation between the network surprise and the other continuous (acoustic and neural) signals is computed with a maximum delay time of -3s to +3s, with steps of 1s following these pre-processing steps of Rao Yarlagadda, 2010 [27]. The normalized correlation is defined as a sliding dot-product of these two signals normalized by the product of their standard deviation. As the correlation between the network surprise and each of the corresponding signals, the one with the highest correlation coefficient within a window of less than one, two or three seconds, depending on the step size of the delay, is chosen.

# 4
# Results

The result section sets out the computational model outcomes, including a statistical analysis utilizing figures and tables. As shown in Figure 3.1 the outcome parts will be addressed in this section, starting with the acoustic features composed by the Librosa library [46], followed by network surprisal of learned from the audio songs and the pre-trained VGGish neural network. Then the band activities for different brain regions and frequencies will be depicted, together with some comparisons through statistics. Finally, the correlation between network surprisal and acoustic features, and between network surprisal and band activity will be profoundly examined.

## 4.1 Acoustic features

The following acoustic features are computed for each song individually:

- Amplitude envelope
- Spectral bandwidth
- Spectral centroid
- Spectral contrast
- Spectral flatness
- Zero-crossing-rate

The amplitude envelope and the zero-cross-rate are time-domain based features and are shown for song with 02 in Figure 4.1 and Figure 4.2, respectively. The amplitude envelope, related to the loudness of an audio, is depicted in combination with the time-domain plot of the signal. Whereas the time signal lays between the range of +0.6 and -0.7 amplitude for over a time period of 7.8 seconds. Low amplitude values corresponds to low loudness values, which can be tracked by the yellow contour line.

The other time-domain based feature is the zero-cross-rate, this feature is depicted in Figure 4.2. The highest zero-cross-rate values can be found between 0.8s and 1.4s, and around 3.8s.

The next four figures, 4.3 to 4.6 are acoustic features computed in the frequency-domain of the sound signals. The first spectral feature is the spectral contrast, Figure 4.3, where 7 frequency bands are illustrated on the y-axes and the time on the x-axes for song ID 22. The heatmap shows that the most spectral energy is located in the top frequency band, the band with the highest frequencies, and

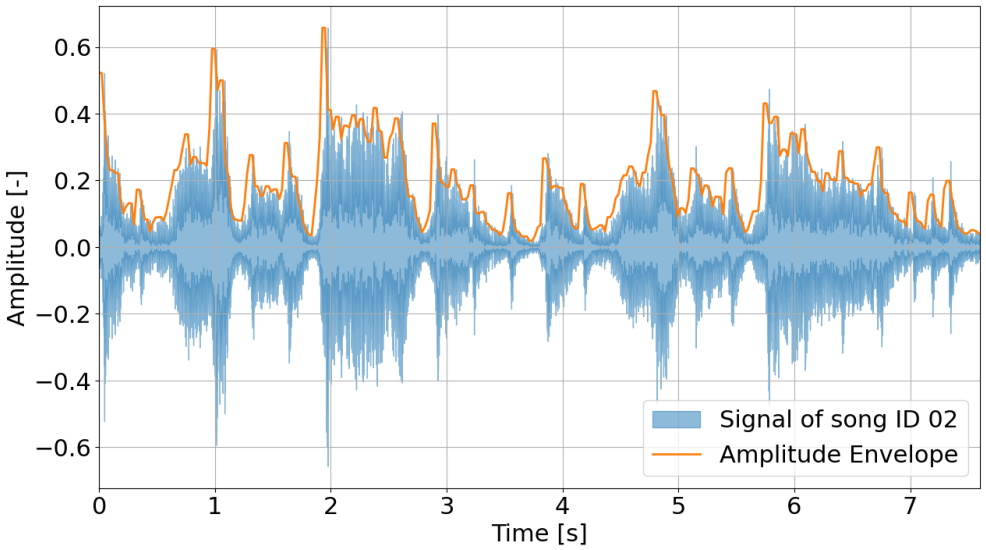


**Figure 4.1:** Time-domain plot of the audio signal of song with ID 02 in blue, the amplitude envelope, also known as loudness is depicted in yellow.

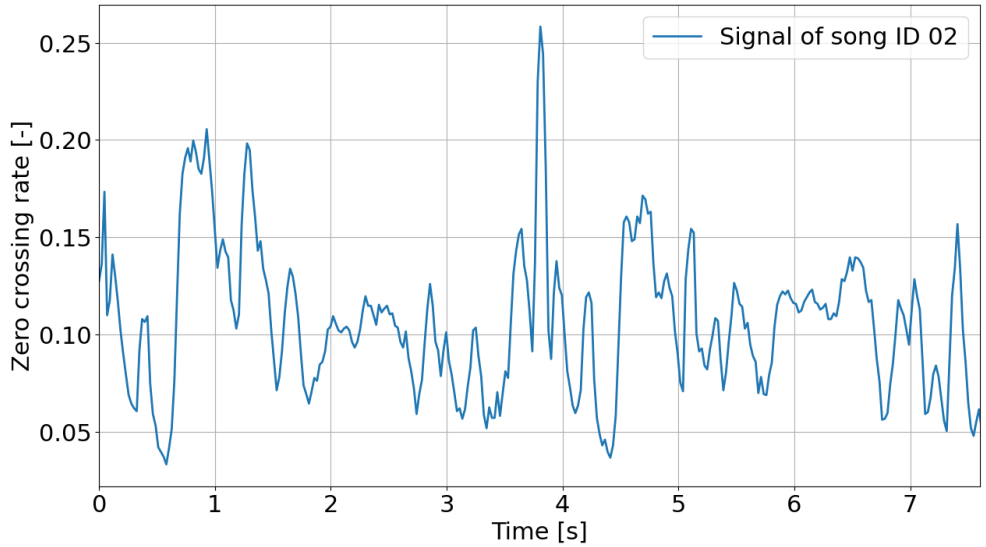


**Figure 4.2:** Zero-cross-rate plot of song ID 02. Higher ZCR values means more broadband noise in the signal, which can be correlated to percussive sounds.

the lowest in the second band. The middle bands show various energies over time, changing from band to band.

The spectral centroid, or centrum of mass of the spectrum, can be seen in Figure 4.4 in combination with the spectrogram of the same sound signal of song with ID 02. The center of mass has its equilibrium around 2000 Hz, with a maximum of 4100

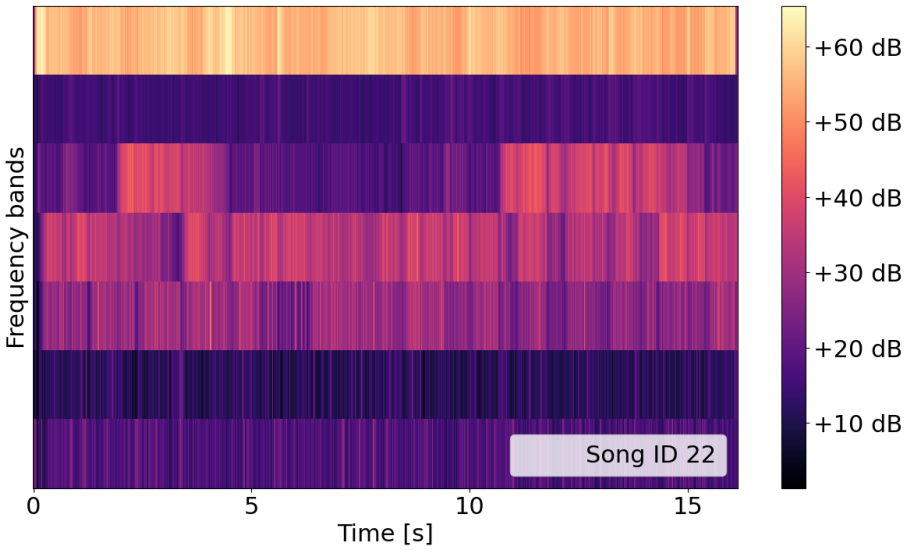


**Figure 4.3:** Spectral contrast plot of song ID 22 for different frequency bands, the top bands are related to high frequencies.

Hz at 3.9s and a minimum around 1020 Hz at 4.6s.

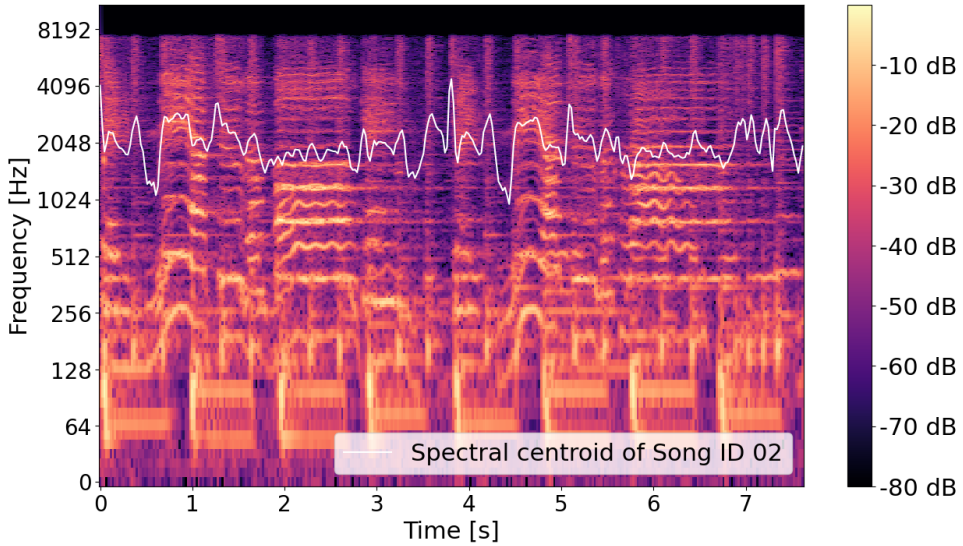


**Figure 4.4:** Spectral centroid plot of song ID 02, depicted by the white line that is on top of the spectrogram. The center of mass of the spectrum is around 2048 Hz, with maximum 4100 Hz and minimum 1024 Hz.

The fifth acoustic feature is the spectral bandwidth, shown in Figure 4.5. The frequency range of the bandwidth is from 1400 Hz up to 2600 Hz, with its several peaks, 3 in the beginning of the song at 0s, 0.5s and 1.4s. Four peaks are found in

the middle between 3.2s and 4.5s, and one more at 5.1s. At the end of the song the bandwidth has its peaks around 2500 Hz, for about 1s long.

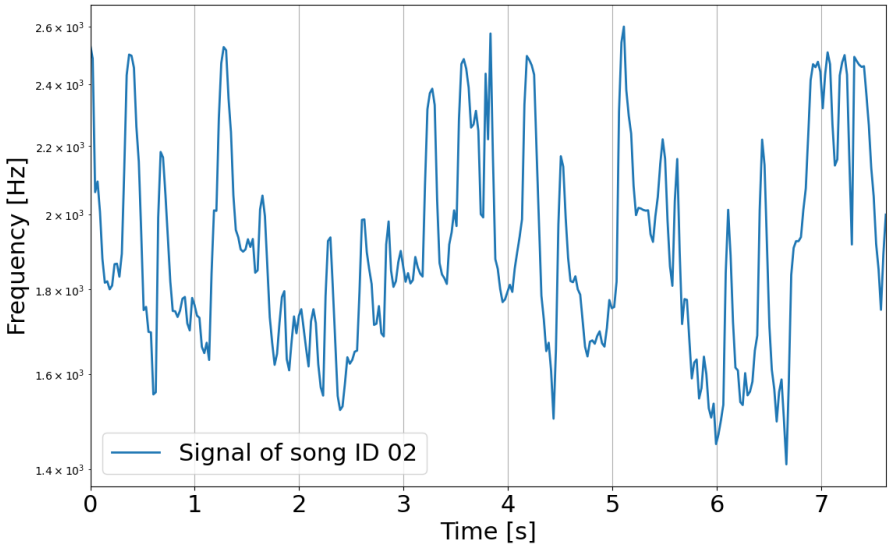


**Figure 4.5:** Spectral bandwidth plot of song ID 02 with maximum and minimum of 2600 Hz and 1400 Hz, respectively.

The last acoustic feature that will be presented is the spectral flatness, depicted in Figure 4.6. The flatness rate fluctuates between the values 0 and 0.28. The highest peak can be found at 3.9s, with four lower peaks at 0.75s, 1.4s, 5.1s and 7.6s.

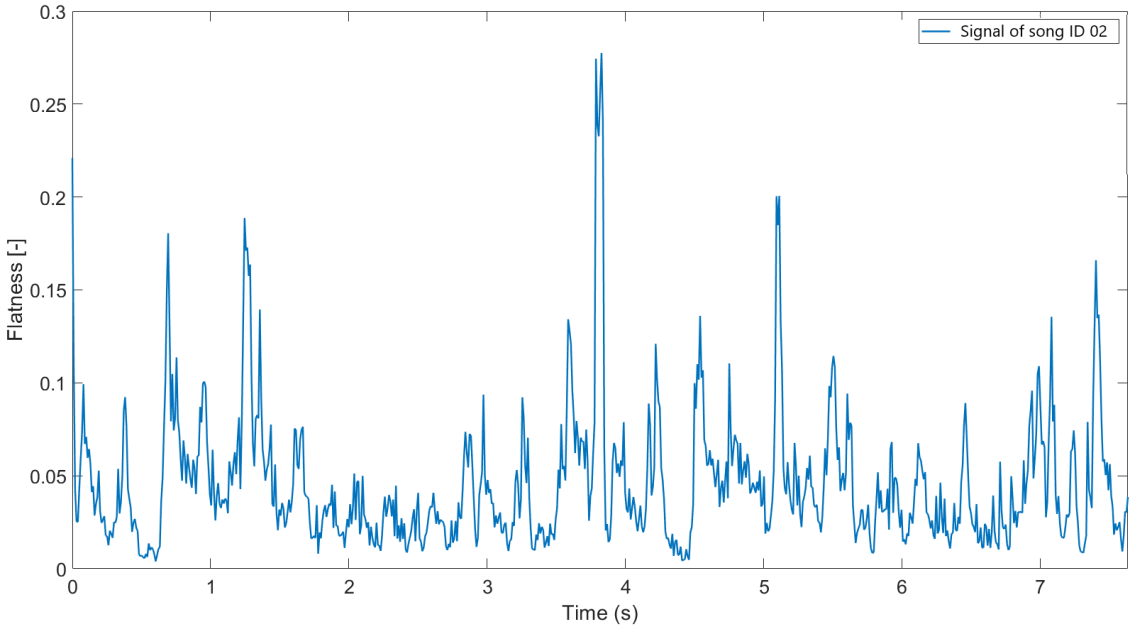


**Figure 4.6:** Spectral flattness plot of song with ID 02 ranging between 0 and 0.28. 5 peaks are present with its highest at 3.8s.

## 4.2 Network surprisal

This section shows the results of the network surprisal of the VGGish network. Each song is fed to the network with chunks of 0.96s of the song signal, with gives as outcome one value per frame of 0.96s. For simplicity 1 frame is referred as 1 second. The layers of interest, Pooling layers 2, 3, and 4, convolutional layer 4, fully connected layer 1 and the embedding layer are looked at. The network surprisal is computed for three different history lengths, 1 second, 2 seconds and 3 seconds. The explanation of the history length can be found in section 3.4.7. In Figure 4.7 the network surprisal is depicted for song with ID 01. Each layer is scaled in order to do the comparison between the layers. A flat line means there is no change in activation level of the current frame compared to its previous frames, with other words there is no "surprise". The graph shows that for each network layer fluctuations are present, meaning changes of activations and thus "surprises". The first frame, 0, has value zero for each graph and layer, this is due the fact that the first frame computes its relative change with the previous frame(s). However, there is no data before frame 0, to solve this problem additional frames are inserted at the beginning. The additonal frames are copies of frame 0, therefore the activation levels are the same for both frame 0 and the history frames. This results in a network surprisal of value 0, meaning no change in activation level.

The three graphs show a similar behaviour for each history length (1s, 2s, 3s). The fully connected layer shows the highest fluctuations for each plot. For most of the layers the highest surprisal value is at frame 1 when a history length of 2 and 3 seconds is chosen, for the case of history of 1 second this is at frame 3 and 8.

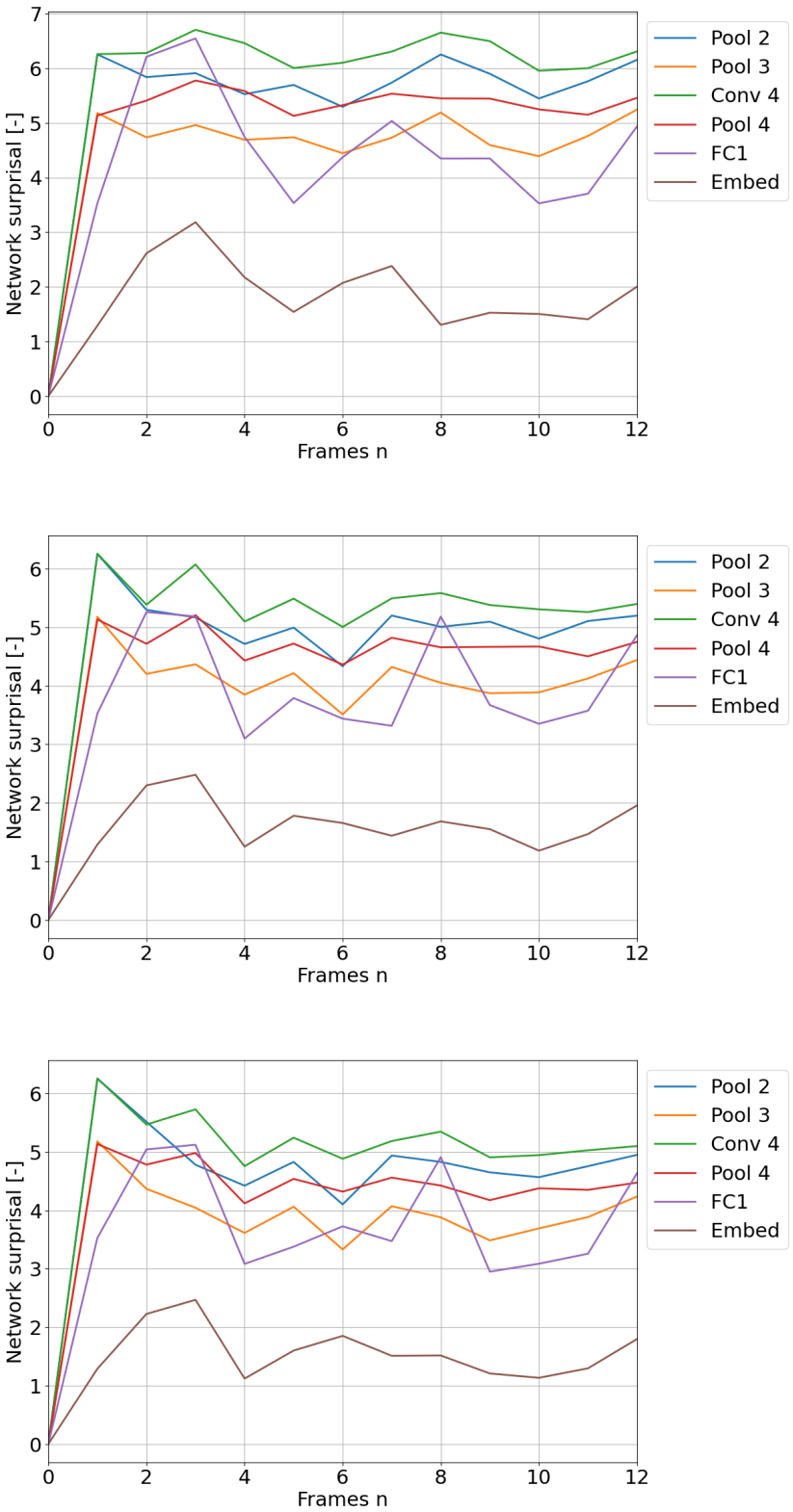


**Figure 4.7:** Network surprisals of the same song, ID 01, but with different history lengths, 1s (top), 2s (middle) and 3s (bottom). History lengths refers to average activations of the past x time lengths.

The following results treated in this chapter are based on network surprisal with a history length of 1 second. The reason to choose 1 second over the other two histories is that there is little difference between the three settings. Another reason is the fact that the songs are short, only 7 to 16 seconds, and they contain high dynamics in the sound envelope. High dynamics means in this sense that within a second there are multiple changes happening in the frequency as well in the time-domain. This is less true for example nature soundscapes sounds, think of recordings made at the beach, river or highway. The sound envelope of these examples do not change that rapidly over time and frequency domain, compared to music. A shorter history is therefore preferred.

## 4.3 Band activity

The band activity for the five frequency bands of the EEG signals are computed and shown in this section. In addition the comparison between the bands and the brain regions are depicted.

### 4.3.1 Band activity per frame

In a similar manner as the network surprisals of the VGGish layers are the activity per frequency band computed for the EEG signals. Instead of looking at the relative activity like for the network surprisals, the absolute values are computed for the brain signals. The band activity is shown for two different fragments, see Figure 4.8, one of song with ID 11, subject P04 for frequency band alpha and only looking at the channels located at the temporal lobe. These are the channels closest to the ears. The activity for this fragment is at its highest peak 700 $V^z/Hz$ and lowest around 0 $V^z/Hz$. The channels with the higher energies are 01, 02, 03, and 33. The second fragment, shown in the figure on the right of 4.8, displays lower maximum energies per channel, with a maximum of 310 $V^z/Hz$ of channel 60. However the lowest energy is 75 $V^z/Hz$ of channel 57, with the other channels energies in between. As a side note, not all the channels of the parietal lobe are displayed, the first 8 channels are omitted. The reason for this is for the sake of the visualization of the plot, too many channels would made the figure unreadable. The omitted channels have energies of 100-200 $V^z/Hz$ or lower.

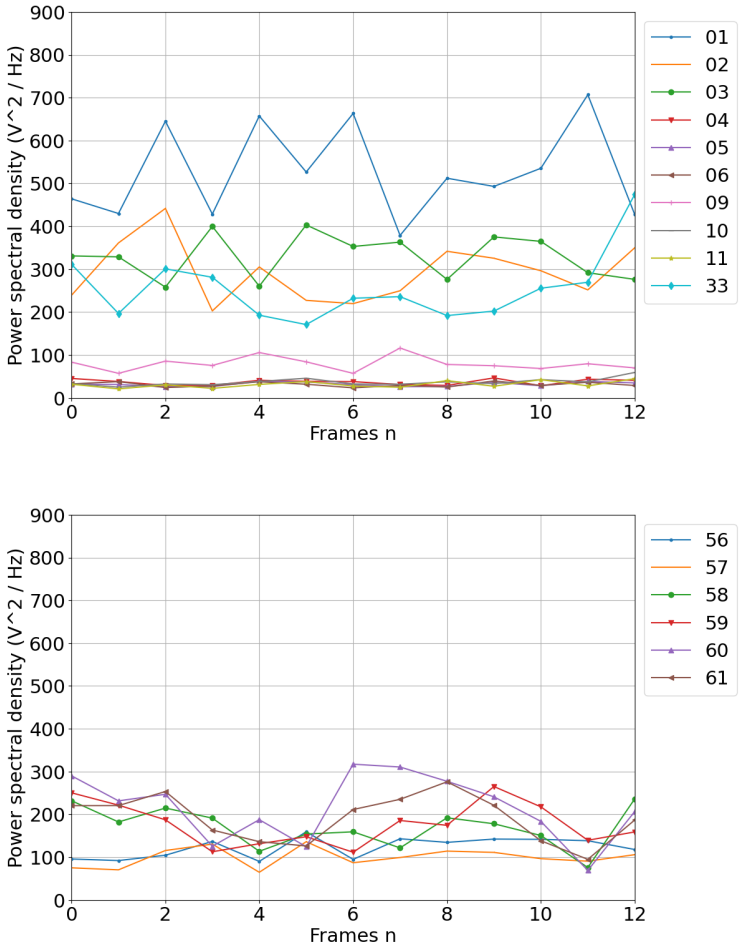


**Figure 4.8:** Band activities in for two cases. Top: Song ID 11, subject P04, frequency band alpha and temporal lobe. Bottom: Song ID 01, subject P01, frequency band gamma and parietal lobe. It shows that the alpha band has higher activity than gamma band for most channels.

### 4.3.2 Band activity per frequency band

Instead of looking at a single case of band activity, the activities of each participant for the same song are added up and combined in one figure. This is applied for song with ID 11 and can be seen in Figure 4.9, where on the left the delta frequency band is depicted and on the right the alpha band. The selected channels are of the temporal lobe.

The distribution of energies for both regions show a similar trend, the energies of the channels lay between 0 and 600 $V^z/Hz$. The delta band has a broader distribution of the energies per channel, where the most channels have power spectral density of 80 - 100 $V^z/Hz$. The alpha band has the highest count of channels with energies around 0-10 $V^z/Hz$, up to 340 channels. The other channels have a distribution of power spectral density between 100 and 600 $V^z/Hz$.

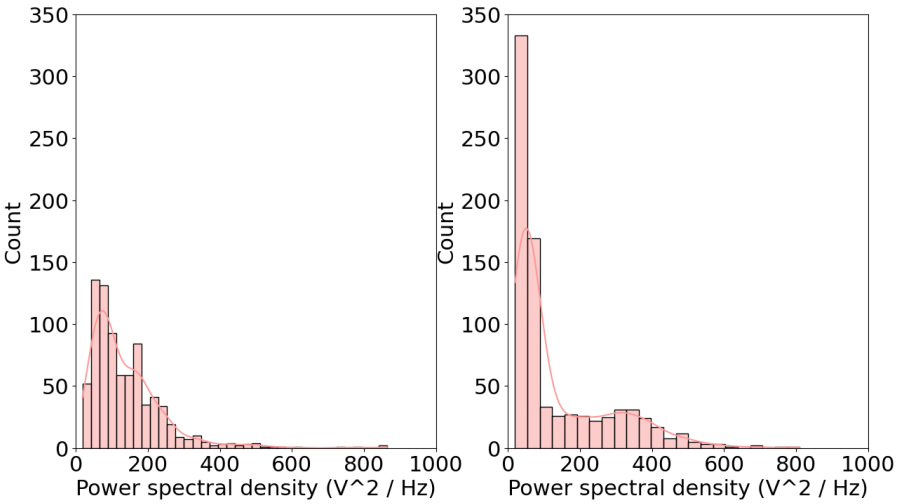


**Figure 4.9:** Histogram of two bands, left: delta, right: alpha of the same song ID 11 and same brain region: temporal lobe.

### 4.3.3 Band activity per brain region

As comparison of band activity between the different brain regions, four boxplots are depicted in Figure 4.10 and 4.11 for song with ID 01 with all the subjects recordings combined. The first Figure 4.10 shows the band activity of the frontal lobe (left) and parietal lobe (right), in case of frontal lobe the lowest activity is present at the theta band, for parietal it is the delta band. 50 percent of the data for each band lies between 50 and 200 $V^2/Hz$ for both brain regions, where theta and beta band shows the highest average band activity for the parietal lobe. The outliers of both brain regions are at maximum 900 $V^2/Hz$, this can be related to artifacts or bad EEG data, since only a few data points has these results.

The temporal lobe and the occipital lobe are depicted in Figure 4.11, where the temporal lobe shows the lowest average band activities of all four lobes, fluctuating around 75 $V^2/Hz$. The most active band is the delta, theta and gamma band in the occipital lobe.

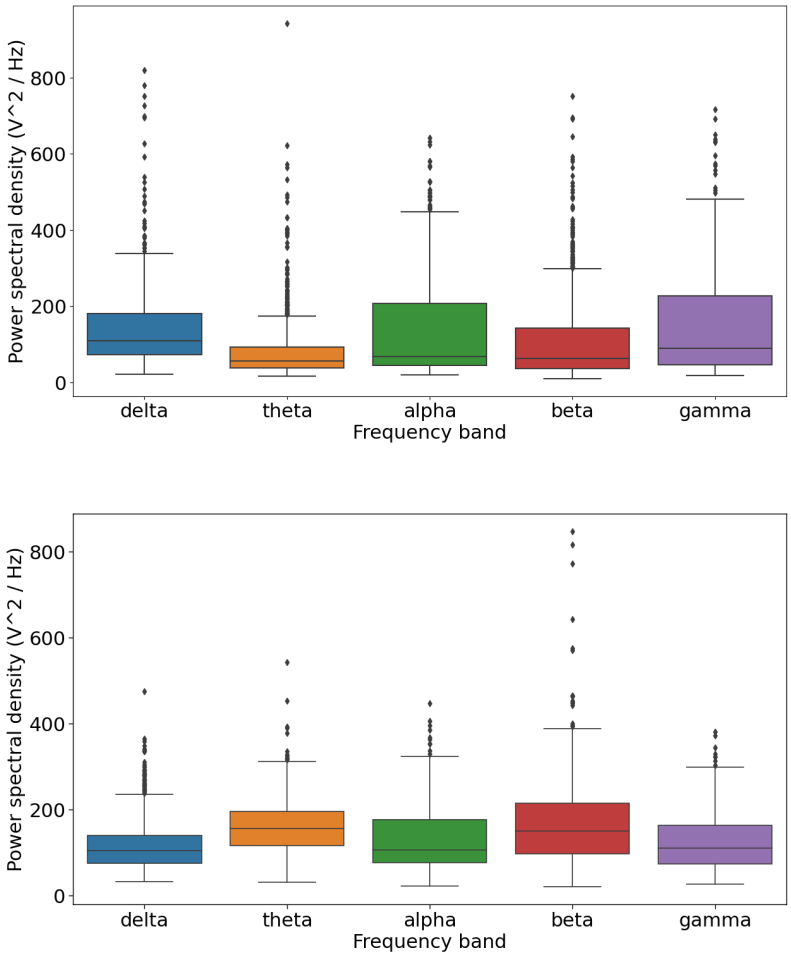


**Figure 4.10:** Band activity compared between two brain regions of the same song ID 01, top: frontal lobe, bottom: parietal lobe. The parietal lobe shows a higher activity for the theta and beta band than the frontal lobe.

## 4.4 Correlation: network surprisal and acoustic features

This section will show the correlation between network surprisal of the VGGish deep neural network layers and the hand-crafted acoustic features. First the comparison will be made between the different delays that are applied for the normalized cross-correlation, with 0s, 1s, 2s and 3s delays. Based on this result one of the delays is chosen and used for the correlation computations.

### 4.4.1 Delay comparison

The normalized-cross correlation is applied in two ways, one by delaying the network surprisal vector, the other by delaying the acoustic features vector. The reasoning

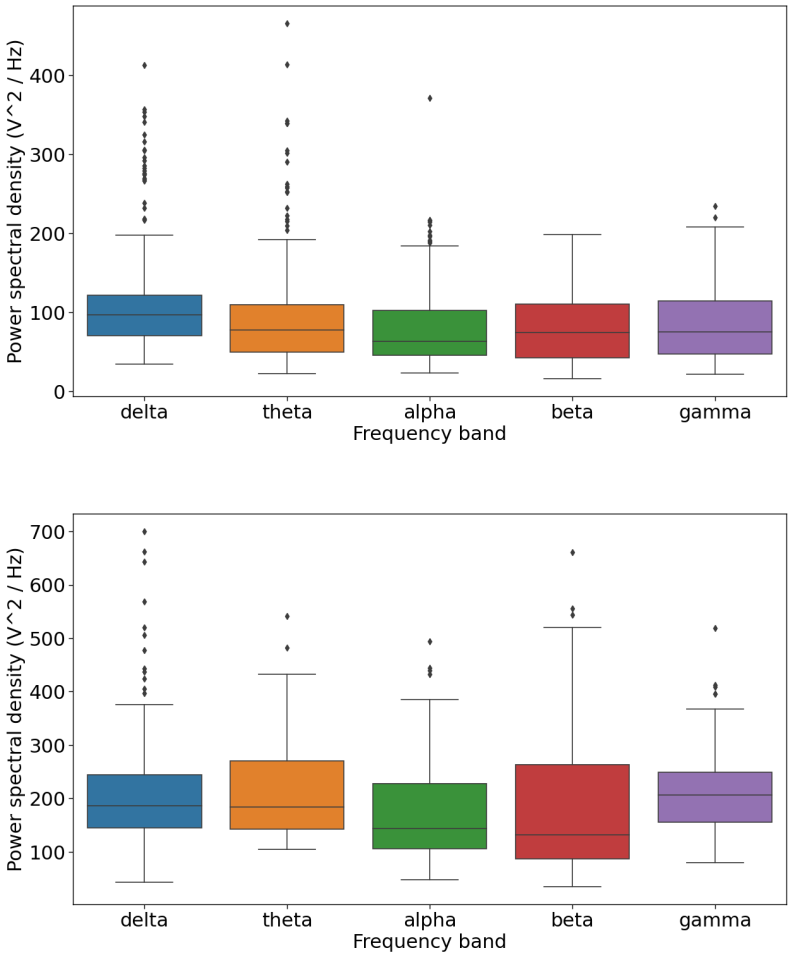


**Figure 4.11:** Band activity compared between two brain regions of the same song ID 01, top: temporal lobe, bottom: occipital lobe. The temporal lobe shows a higher overall activity than the occipital lobe.

behind the use of delays is based on [31] and explained in the 3 Methodology Chapter. On the left graph in Figure 4.12 the delays are shown when the network surprisal vector was changed. When a delay of 0s is applied, the correlation is slightly below zero. In case for delay 1, 2 and 3 frames, 50 percent of the correlation are between 0.20 and 0.60. When looking at the correlations when the acoustic features vector is delayed, see Figure 4.12 on the right, the correlation with a delay of 0 frames is exactly the same as for the figure on the left, as expected. However for the delay of 1, 2 or 3 frames, the correlations lay all above a correlation value of 0.60, with 75 percent of all correlation values higher than 0.80. Based on the results of these two plots, it is assumed that that delaying the acoustic features gives wrong correlation values. Therefore the delay of 1 frame is chosen for further computations, delay 2 and 3 are relatively large delays since the stimuli used are only 6 to 16 frames, hence 1 frame delay fits the most. A delay of 0 frames was only taken in account as reference, but not as possible further usage.

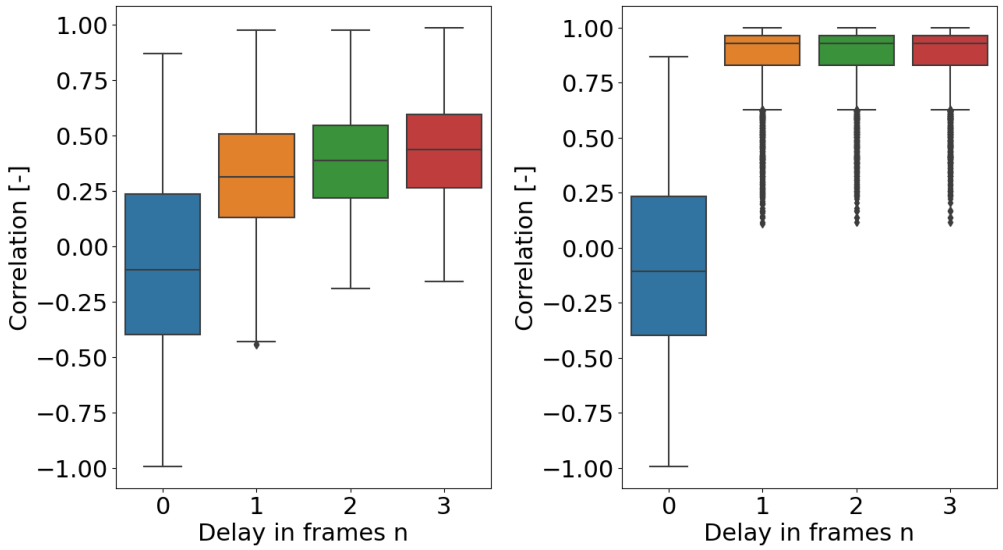


**Figure 4.12:** Comparison between different delays of when applying the normalized cross-correlation between network surprisal and acoustic features. Left: Network surprisals are delayed 0s-3s. Right: Acoustic features are delayed 0s-3s. It shows that delay of 1 sec of network surprisal is more reasonable result.

### 4.4.2 Correlation

The correlations between the acoustic features and network surprisal for all songs combined are shown in Figure 4.13. The highest average correlations can be found for the amplitude envelope and spectral contrast of frequency band 7 features. The lowest are spectral centroid, spectral constrast band 2, and spectral flatness, with their median around 0.18 correlation. The other features have their median fluctuating between 0.2 and 0.4.

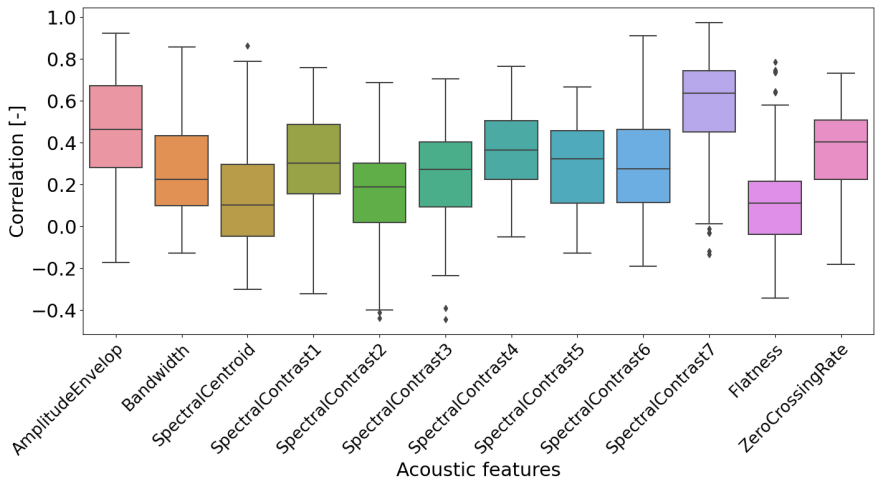


**Figure 4.13:** Comparison between correlations the different acoustic features for all the songs and participants combined.

Finally, the acoustic features are compared with the activations of the VGGish neural network layers when trained on the OpenMIIR song data, this is depicted in a correlation matrix shown in Figure 4.14. The correlations values lay between 0.016 and 0.59. What can be notice is that the highest values of correlation for each individual feature are assigned to the earlier layers of the network, and the lowest values at the fully connected and embedding layer. As exception the spectral bandwidth and spectral contrast band 2, here are the highest values in the latter layers of the neural network. The difference between the highest and the lowest correlation value are for most layers maximum 0.1, except for spectral contrast layer 4 where the correlation difference is 0.14. For two acoustic features the correlation stays more or less the same over the six network layers, the correlations differs maximum 0.3, this counts for spectral bandwidth and spectral centroid. Most of the correlation values are doubtful high, especially the values of the amplitude envelope, spectral contrast 4 and 7, and zero-crossing-rate. This will be further analyzed in the Discussion section 5.

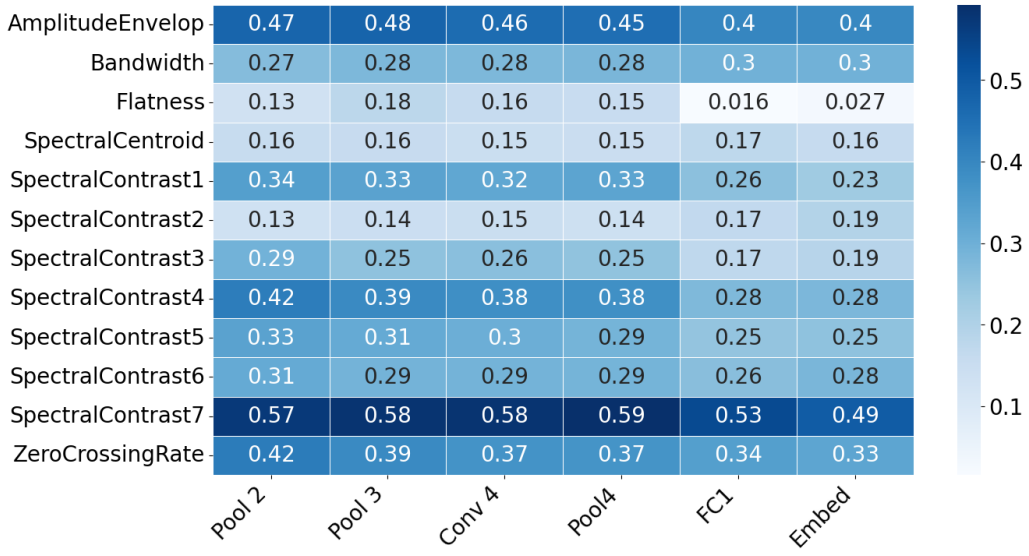


**Figure 4.14:** Correlation matrix between each feature and the VGGish network layers of interest.

## 4.5 Correlation: network surprisal and EEG

The correlation between the network surprisals and the EEG band activities will be shown in this section. The five frequency bands, delta, theta, alpha, beta and gamma are compared with the network layers of interest, which are Pooling 2, 3 and 4, convolution layer 4, fully connected 1 and embedding layer. To examine the correlations matrices, one song for one subject is chosen, see Figure 4.15 and 4.16 below. The correlation matrices for the frontal lobe and the parietal lobe brain

regions are depicted in the first Figure 4.15. What can be seen in the matrices is that the highest correlation are found in the theta band for the frontal lobe, and the theta and beta band for the parietal lobe. The low frequencies, delta theta and alpha, are almost constant over all the network layers, with a slight dip of 0.1 to 0.2 in the middle layers, convolution 4 and pool 4. The beta and gamma band, containing 15-50 Hz frequencies, show a higher correlation in the early layers, pooling 2 and 3, and a lower correlation in the last layers, fully connected 1 and embedding. Look at the parietal lobe, the distribution is similar for the delta and alpha band as for the frontal lobe, whereas the theta layer is different. Here the correlations are slightly higher, 0.4, and start with highest correlation at pooling layer 2 with a value of 0.29, and descents to 0.26 at the embedding layer. The beta band correlation values are high for the first 5 layers, between 0.28 and 0.3, and is the lowest at the sixth layer, 0.26. The gamma band stays almost even for over all the layers, with values between 0.19 and 0.21.

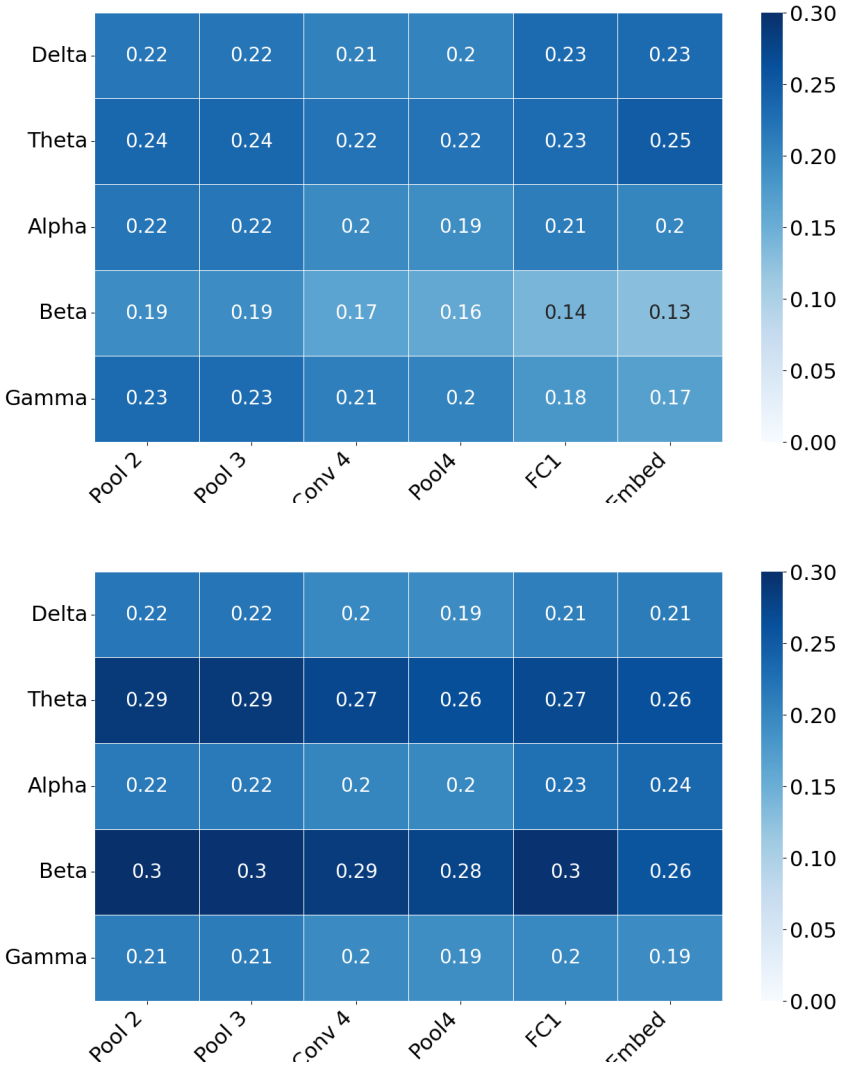


**Figure 4.15:** Correlation matrix between each frequency band and the layers of the VGGish network. Top: Frontal lobe, bottom: Parietal lobe.

The correlation matrices for the temporal lobe and the occipital lobe can be found in Figure 4.16, where the temporal lobe contains the highest correlations of the two. The temportal lobe shows high correlation values for the delta lobe, the lowest frequencies of 1-4 Hz, and stays almost equal over the all the layers, with values between 0.28 and 0.3. The other four bands consist of lower correlation values, ranging from 0.16 to 0.24, but stays as well almost even over all the layers. The lowest correlations can be found for the gamma band for the convolutional 4 and pooling 4 layer. The occipital lobe has the lowest correlations in the delta band instead, ranging from 0.17 to 0.13 in the last layer of the network. The theta, alpha and gamma band have equal correlations over the 6 layers fluctuating between 0.21 and 0.23, however the beta band descents starting from 0.21 down to 0.17 for the last two layers of the network.

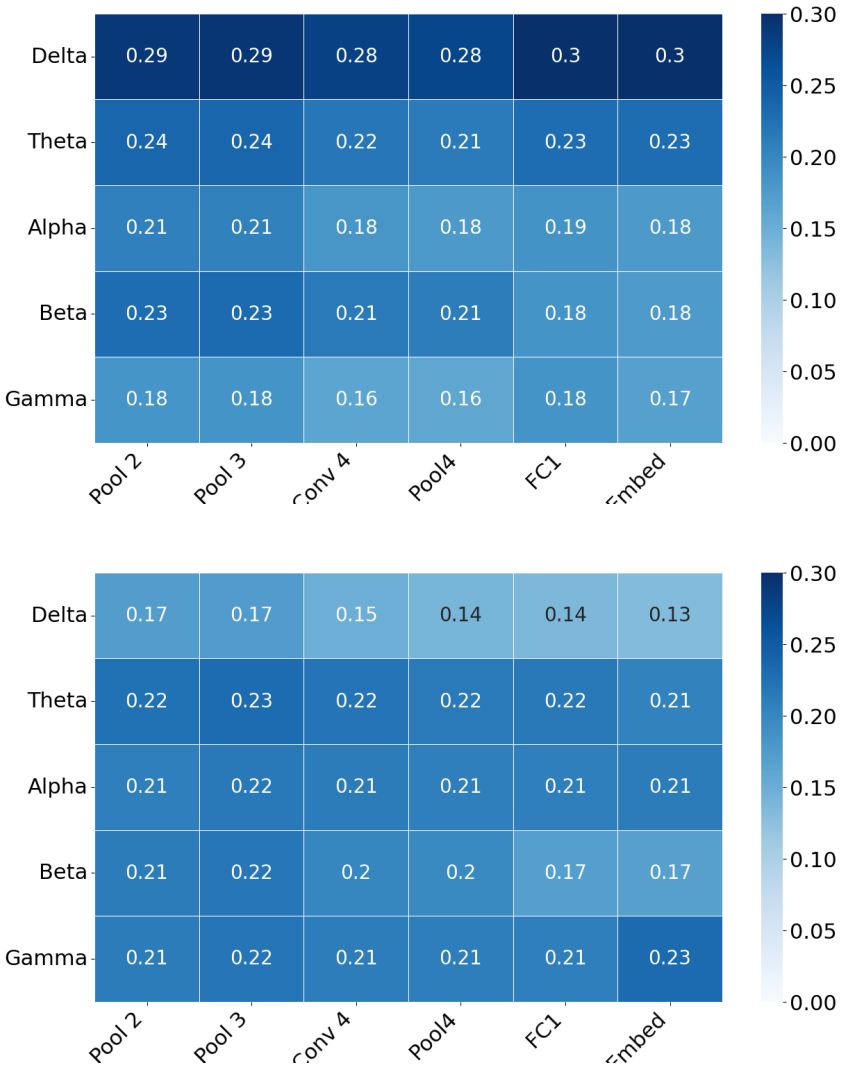


**Figure 4.16:** Correlation matrix between each frequency band and the layers of the VGGish network. Top: Temporal lobe, bottom: Occipital lobe.

# 5
# Discussion

This research aimed to set up a computational model that makes it possible to analyze auditory salience for both bottom-up and top-down attention with use of music excerpts. Based on correlation analysis between activations of deep neural network layers, brain activity per frequency band and hand-crafted acoustic features it can be concluded that relationships can found, however the results in this work should be taken very tentatively. This chapter provides a reflection on the research process. The design's limitations and potential outcomes, as well as the implications for interpreting the results, are discussed. Lastly, recommendations for future studies are appointed.

The hypothesis of the correlation analyses between the acoustics features, frequency band activities of brain waves and the activity of the artificial neural network layers met with the results of this thesis work. Early layers of the neural network was expected to have higher correlations with low-level features, related to bottom-up attention, whereas the deeper layers should have shown higher correlation values with high-level features, related top-down attention. Correlation matrices show similar trends, matching the results from Huangs work and the hypothesis of how the brain processes audio. However, the values of the all the correlation matrices are very close to each other. Deviations lay between values of 0.03 to 0.12, where correlation values lay between 0.0 and 1.0. Compared to Huangs work the correlation value found in this work are smaller, Huangs work show deviations up to 0.4. But also most of his outcomes of the correlations have deviations values similar to this work. When looking at the absolute values of the correlation matrices, it stands out that all the values have around 0.2. Meaning all the layers activities, acoustic features and EEG band activities show correlations. This is contradicting the expectations, it was estimated that the largest part of the correlation analyses have values close to 0, meaning no correlation, and a few values a little higher than 0. This is true for Huangs result, where the mean lays around 0.05. Concluded from this it could mean that there was a scaling issue that caused outcomes have a mean around 0.2 instead of 0.05.

A few words are now given that are related to research process, why the results should be taken very tentatively.

The first reason is related to used data from the OpenMIIR dataset. The dataset contained only ten participants, which was in terms of databases a very small amount. Second the stimuli utilized in the experiment were only a few seconds

long, 6s to 16s. This makes it sensitive for artifacts drastically, a gently error during the listening of the songs can disturb the whole song. Fortunately the participants listens to the same piece five times, which reduces the influence of errors. The other reason what had great influence on the results is the fact that the VGGish pre-trained neural network only works with low-quality spectrograms and utilizes chunks of audio signal of almost 1 second long. A very coarse resolution since it is known that music is very dynamic in within this period of time. When compared to natural sounds, such as waves of the sea hitting the beach producing sound, is known as predictable over a 1 second of time. This does not count for most music songs. Due to this relative long frame the vector that was computed by model for the embedding layer, is related to 1 second long data. This means that the vector cannot tell what happened at time stamp 0.1s or at 0.7s for example, since it makes a summary for the whole frame of 1 second. As noted above, the OpenMIIR music stimuli are short, as consequence only a few values are computed per song. That makes the results very coarse. As last note on the VGGish model and its input frame sizes, the songs from the OpenMIIR database are very dynamic in the spectral as temporal domain. When the model makes summarizes of chunks of 1 second song data, a lot of the temporal information is not traceable. Hereby much of the song data has been lost for this reason, and thus are the final results fuzzy.

The second framework based on VGGish pre-trained neural network model to compute network surprisal was a relative novel and not frequently applied approach. The setup of the model was relative simple since it was pre-trained and easily accessible model via a multiple platforms. Next the parameters to adjust the model was restricted, because of the pre-trained weights are only suited for default settings. The only difficulty was to shape the input data in the same format as the pre-trained model, and creating a method to retrieve activations from individual layers. Subsequently, when the data was pre-processed correctly and activations could be retrieved per layer the network surprisals could be computed. The method to do this was based on Huang [31] work, however it was not a clear description of how they did this. In addition the idea of adding up the activations values in layers to correspond them to the activity of a layer was more of an assumption. By a fact the hidden layers of a deep neural network are a black box, especially larger models as the VGGish network, this makes it harder or impossible to know what is what. Fortunately, two papers were found that performed a similar approach by using the Euclidean distances [48] [33], from which the latter was published in Nature.

The last framework was build in order to retrieve the band activity per frequency band of the EEG signals. To achieve this the EEG data provided by the OpenMIIR database had to be pre-processed first. This was not an easy task since the notebooks that were included with the EEG data did not work. These notebooks consisted all the steps of pre-processing, but since the Python libraries related to the notebooks were deprecated and missing, these notebooks could not be used. Therefore pre-processing EEG signals was mostly carried out by self study and the consult of the Bioengineering department of Polimi. Due to the lack of experience with EEG signals and pre-processing them, there is room for mistakes, however

the EEG signals retrieved from participants during the experiments were already fairly clean (read almost no signal artifacts), this makes the the pre-processed EEG signals reliable. The second step in redeeming the band activity was to filter the EEG signals into frequency bands, this without losing information. Three methods have been tried, wavelet transformation, bandpass filter function from Matlab and designing own bandpass filters using the Hilbert-method. The latter two methods gave the same results, validating that both functions work properly.

### 5.0.1 Recommendations

At last, some recommendations for further research on designing a computation model for auditory salience focusing on bottom-up and top-down attention. Firstly, most important part of training a neural network is to get descent data, this means sufficient, clean, well annotated data with great variety of classes. Without good data, a model will not be able to give great results. So, this can be acknowledged by searching for better public available datasets online, or gathering own data by examine a listening test. Secondly, instead of using an existing artificial neural network that comes with its limits, it could be desired to build an own model that meets the requirements of the research work, for this case it means having an input size of less than 1 second length. Or find a way to bypass the limit of the existing network. The third recommendation is related to the quality of the data, the music in the OPENMIIR dataset was rather complex. This makes it hard to find a direct relation between the results of the correlation analyses and the specific element in the music that could be related to salience. Since the music was complex, meaning multiple instruments, tempo and chords present in one song, it is hard to filter out the direct cause. By taking less complex music, relations could be easier directed to each other. As final suggestion for continuing this thesis work would be to do more in-depth preliminary research, this to get more a solid bases on the tools, data and theory used.

# 6
# Conclusion

This research aimed to set up a computational model that makes it possible to analyze auditory salience for both bottom-up and top-down attention with use of music excerpts. Based on correlation analysis between activations of deep neural network layers, brain activity per frequency band and hand-crafted acoustic features it can be concluded that relationships can found, however the results must be interpreted with caution.

Following Huangs [31] methodology for building the computational model for auditory salience provided a sufficient foundation. Comparing the results of Huangs and this work, similar trends were found for the correlation between the neural network activity and acoustic features, meaning this model is able to detect low-level acoustic features from music excerpts. Although the results of the correlation values between neural network activity and energy in EEG frequency bands did not match with Huangs work, neither there could be other conclusions made of it. It therefore failed to predict top-down attention of auditory salience.

This is assumed to be related with the used OPENMIIR dataset and EEG signals pre-processing implementation, they were far from ideal or well executed. Looking at the dataset it was small, had short music excerpts, and contained complex music pieces. This all in combination with the large fixed input size of the VGGish neural network resulted in a potentially great loss of useful information, which influences the correlation analyses immensely. On top, the pre-processing implementation of the EEG signals was performed with a lack of experience. To get more statistically robust results, these issues should be taken care of.

Nevertheless the idea of combining artificial neural networks, EEG signals and acoustic features is highly interesting development for retrieving higher semantics of music. With more in-depth preliminary research in auditory salience, higher qualitative data and more carefully executed implementation of artificial neural networks and EEG signals this could be attractive for further research in top-down attention in auditory salience.

# Bibliography


[1] G.E. Hinton A. Krizhevsky, I. Sutskever. ImageNet classification with deep convolutional neural networks. *Assosciation for Computing Machinery*, 60, june 2012.

[2] Priyanka A. Abhang, Bharti W. Gawali, and Suresh C. Mehrotra. Chapter 2 - technological basics of eeg recording and operation of apparatus. In Priyanka A. Abhang, Bharti W. Gawali, and Suresh C. Mehrotra, editors, *Introduction to EEG- and Speech-Based Emotion Recognition*, pages 19–50. Academic Press, 2016.

[3] Abien Fred Agarap. Deep learning using rectified linear units (relu). *CoRR*, abs/1803.08375, 2018.

[4] Milad Azarbad, Hamed Azami, Saeid Sanei, and Ataollah Ebrahimzadeh. A time-frequency approach for eeg signal segmentation. *Journal of AI and Data Mining*, 2:63–71, 06 2014.

[5] Yasaman Bagherzadeh, Daniel Baldauf, Dimitrios Pantazis, and Robert Desimone. Alpha synchrony and the neurofeedback control of spatial attention. *Neuron*, 105(3):577–587.e5, 2020.

[6] Sylwester Białowąs and Adrianna Szyszka. *Eye-tracking in Marketing Research*, pages 91–104. 01 2019.

[7] R Burns, T.; Rajan. Combining complexity measures of eeg data: multiplying measures reveal previously hidden information. *F1000Research*, 2015.

[8] H. E. Egeth C. E. Connor and S. Yantis. Attention, awareness, and the perception of auditory scenes. *Current Biology*, 14:850–852, 2012.

[9] M. Lippert C. Kayser, C. I. Petkov and N. K. Logothetis. A feature-integration theory of attention. *Elsevier, Cognitive Psychology*, 12:97–136, 1980.

[10] M. Lippert C. Kayser, C. I. Petkov and N. K. Logothetis. Mechanisms for allocating auditory attention: An auditory saliency map. *Current Biology*, 15:222–228, 2005.

[11] S. Siri K. Sosta S. Cappa C. Maioli, I. Benaglio. The integration of parallel and serial processing mechanisms in visual search: Evidence from eye movement recordings. *European Journal of Neuroscience*, pages 364–372, 2001.

[12] Hsuan-Chu Chen, Chun-Chia Wang, and Jason Hung. Employing eye tracking to study visual attention to live streaming: A case study of facebook live. *Sustainability*, 14:7494, 06 2022.

[13] Deutsch J. A. Deutsch D. Contextual cues in selective listening. *Journal of Experimental Psychology*, pages 12 (4), 242–248., 1960.

[14] Deutsch J. A. Deutsch D. Attention: Some theoretical considerations. *Psychological Review*, pages 70, 80–90., 1963.

[15] V. Delmotte. *Computational Auditory Saliency*. PhD thesis, Georgia Institute of Technology, Georgia, GT, December 2012.

[16] MNE Developers. Audioset database.

[17] Librosa development team. Librosa python library.

[18] Broadbent D. E. Perception and communication. *London: Pergamon Press.*, 1958.

[19] Cherry EC. Some experiments on the recognition of speech, with one and with two ears. *The Journal of the Acoustical Society of America.*, pages 25(5):975–979., 1953.

[20] J.M. Findlay. Saccade target selection during visual search. *Vision Research*, pages 617–631, 1997.

[21] Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016.

[22] Google. Audioset database.

[23] R. Goodman H. Greenspan, S. Belongie. Overcomplete steerable pyramid filters and rotation invariance. *IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 222–228, 1994.

[24] Hadi Hadizadeh and Ivan Bajic. Saliency-aware video compression. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 23, 09 2013.

[25] S. Handel. Listening: An Introduction to the Perception of Auditory Events. *MIT Press*, 1989.

[26] Karen Hao.

[27] Simon Haykin and Barry Van Veen. *Signals and systems*. John Wiley & Sons, 2007.

[28] Suzana Herculano-Houzel. The remarkable, yet not extraordinary, human brain as a scaled-up primate brain and its associated cost. *Proceedings of the National Academy of Sciences*, 109(supplement_1):10661–10668, 2012.

[29] Sean D. Holcomb, William K. Porter, Shaun V. Ault, Guifen Mao, and Jin Wang. Overview on deepmind and its alphago zero ai. In *Proceedings of the 2018 International Conference on Big Data and Education*, ICBDE '18, page 67–71, New York, NY, USA, 2018. Association for Computing Machinery.

[30] Xianzhi Du Yeqing Li Abdullah Rashwan Le Hou Pengchong Jin Fan Yang Frederick Liu Jaeyoun Kim Hongkun Yu, Chen Chen and Jing Li. TensorFlow Model Garden. `https://github.com/tensorflow/models`, 2020.

[31] Elhilali Huang, Slaney. Connecting deep neural networks to physical, perceptual, and electrophysiological auditory signals. *Frontiers in Neuroscience*, pages , month = , volume = , url = https://europepmc.org/article/med/30154688„ 2018.

[32] Elhilali M. Huang, N. Auditory salience using natural soundscapes. *J. Acoust. Soc. Am.*, pages 617–631, 2017.

[33] Mathilde Petton. Ilya Kuzovkin, Raul Vicente. Activations of deep convolutional neural networks are aligned with gamma band activity of human visual cortex. *Nature, Communications Biology*, 2018.

[34] L. Itti. Visual salience. *Scholarpedia*, 2(9):3327, 2007. revision #72776.

[35] DM Weintraub JS. Snyder, MK Gregg and C. Alain. Attention, awareness, and the perception of auditory scenes. *Frontiers in Psychology*, 2012.

[36] O. Kalinli and S. Narayanan. A saliency-based auditory attention model with applications to unsupervised prominent syllable detection in speech. *Proceedings of InterSpeech*, 8:1941–1944, August 2007.

[37] Constantinidis C. Katsuki, F. Bottom-up and top-down attention: different processes and overlapping neural systems.. *The Neuroscientist : a review journal bringing neurobiology, neurology and psychiatry*, 2014.

[38] Christof Koch and Shimon Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. In *Matters of intelligence*, pages 115–141. Springer, 1985.

[39] E. Niebur L. Itti, C. Koch. A Model of Saliency-based Visual Attention for Rapid Scene Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998.

[40] N. Lavie. Capacity limits in selective attention: Behavioral evidence and implications for neural activity. *American Psychological Association.*, 1960.

[41] Yi Lin and Po-Jang Hsieh. Neural decoding of speech with semantic-based classification. *Cortex*, 154:231–240, 2022.

[42] Grace W. Lindsay. Attention in psychology, neuroscience, and machine learning. *Frontiers in Computational Neuroscience*, 14, 2020.

[43] Guillaume Lio, Roberta Fadda, Giuseppe Doneddu, Jean-René Duhamel, and Angela Sirigu. Digit-tracking as a new tactile interface for visual perception analysis. *Nature Communications*, 10, 11 2019.

[44] Darrell T. Long J, Shelhamer E. Fully convolutional networks for semantic segmentation. *Piscataway: IEEE*, page 3431–3440, 2015.

[45] Jianxun Lou, Hanhe Lin, David Marshall, Dietmar Saupe, and Hantao Liu. Transalnet: Towards perceptually relevant visual saliency prediction. *Neurocomputing*, 494:455–467, 2022.

[46] Colin Raffel Dawen Liang Daniel PW Ellis Matt McVicar Eric Battenberg McFee, Brian and Oriol Nieto. librosa: Audio and music signal analysis in python. *In Proceedings of the 14th python in science conference*, 2015.

[47] Students of PSY 3031 and Edited by Dr. Cheryl Olman. *Introduction to Sensation and Perception.* University of Minnesota Libraries Publishing, 2022.

[48] Gloria Haro Olga Slizovskaia, Emilia Gómez. A case study of deep-learned activations via hand-crafted audio features. *arXiv*, 2020.

[49] Derrick Parkhurst, Klinton Law, and Ernst Niebur. Modeling the role of salience in the allocation of overt visual attention. *Vision Research*, 42(1):107–123, 2002.

[50] Shafin Rahman, Sejuti Rahman, Omar Shahid, Md. Tahmeed Abdullah, and Jubair Ahmed Sourov. Classifying eye-tracking data using saliency maps, 2020.

[51] Y. Blokland R.S. Schaefer, J. Farquhar. Name that tune: Decoding music from the listening brain. *NeuroImage*, 2015.

[52] Stuart J. Russell and Peter Norvign. Artificial Intelligence, A Modern Approach. *Prentice-Hall, Inc*, 1995.

[53] Peter H. Schiller. Parallel information processing channels created in the retina. *Proceedings of the National Academy of Sciences*, 107(40):17087–17094, 2010.

[54] Luis G. Serrano. *Grokking Machine Learning.* Manning, 2021.

[55] Ramesh Srinivasan Siyi Deng and Michael D'Zmura. Cortical signatures of heard and imagined speech envelopes. *Department of Cognitive Sciences, University of California at Irvine*, 2013.
[56] S. Stober. Openmiir database.
[57] Yu Su, Jingyu Wang, Ke Zhang, Kurosh Madani, and Xianyu Wang. Computational modelling auditory awareness. pages 160–167, 01 2018.
[58] Brett Szymik. What's your brain doing?
[59] Nottebohm F. Ho C. E. Pesaran B. Mitra P. P. Tchernichovski, O. A procedure for an automated measurement of song similarity. Animal Behaviour. *Elsevier*, 2000.
[60] TheodorosGiannakopoulosAggelosPikrakis. *Introduction to Audio Analysis, a MATLAB approach*. Academic press, 2014.
[61] T. Tsuchida and G. W. Cottrell. Auditory saliency using natural statistics. *CogSci*, 14:1048–1053, 2012.
[62] T. Tsuchida and G. W. Cottrell. Bayesian framework for saliency using natural statistics. *Journal of Vision*, 8:1048–1053, 2012.
[63] Western. Brain and mind institute.
[64] Karlijn Willems. Keras tutorial: Deep learning in python.
[65] R. K. Rao Yarlagadda. *Analog and Digital Signals and Systems*. Springer Publishing Company, Incorporated, 1st edition, 2009.
[66] Daniel Zhang, Saurabh Mishra, Erik Brynjolfsson, John Etchemendy, Deep Ganguli, Barbara Grosz, Terah Lyons, James Manyika, Juan Carlos Niebles, Michael Sellitto, Yoav Shoham, Jack Clark, and Raymond Perrault. The ai index 2021 annual report, 2021.
[67] Han J. Jiang X. et al. Zhao, S. Decoding auditory saliency from brain activity patterns during free listening to naturalistic audio excerpts. *Neuroinform*, page 309–324, 2018.
[68] Yi Zheng, Qitong Wang, and Margrit Betke. Deep neural network for semantic-based text recognition in images. *CoRR*, abs/1908.01403, 2019.

# A

# Appendix - EEG

## A.1 EEG Pre-processing

### A.1.1 ICLabel

ICLabel is a tool in EEGLab that provides an estimation of the type of each of the independent components: brain, muscle, eye, heart, line noise, channel noise and other. For each component it calculates a probability in percentage, as well the location on the head where the component is most prominent. The ICLabel tool is shown in Figure A.1 and in Figure A.2.

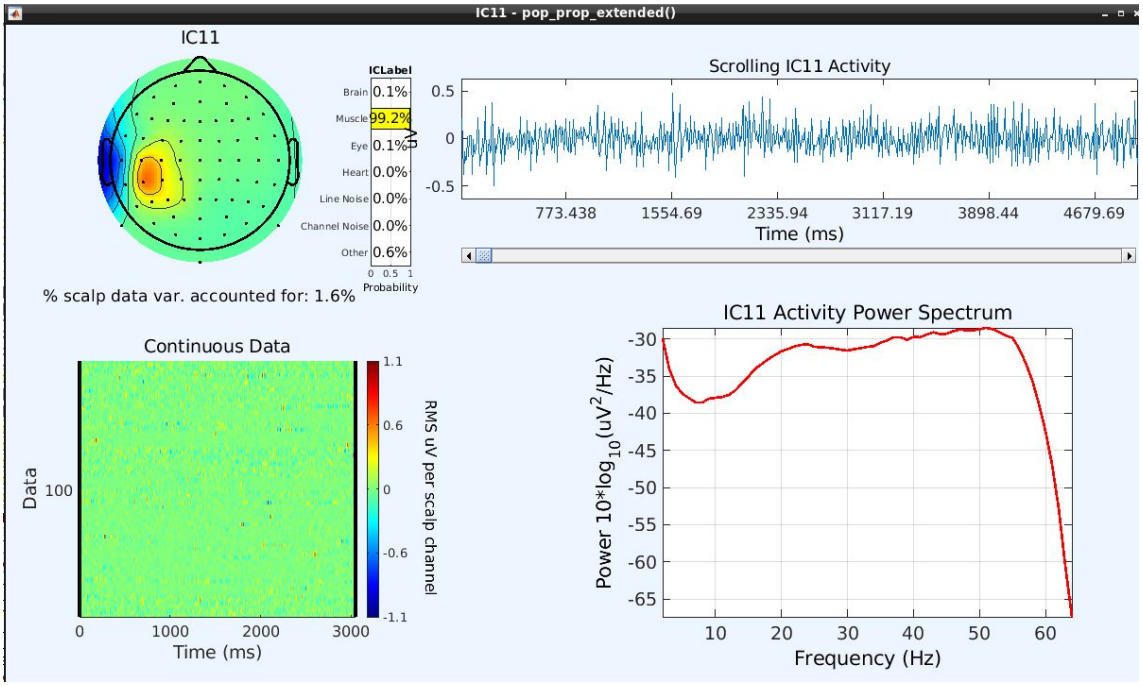


**Figure A.1:** ICLabel detection prominent unwanted indepented components.

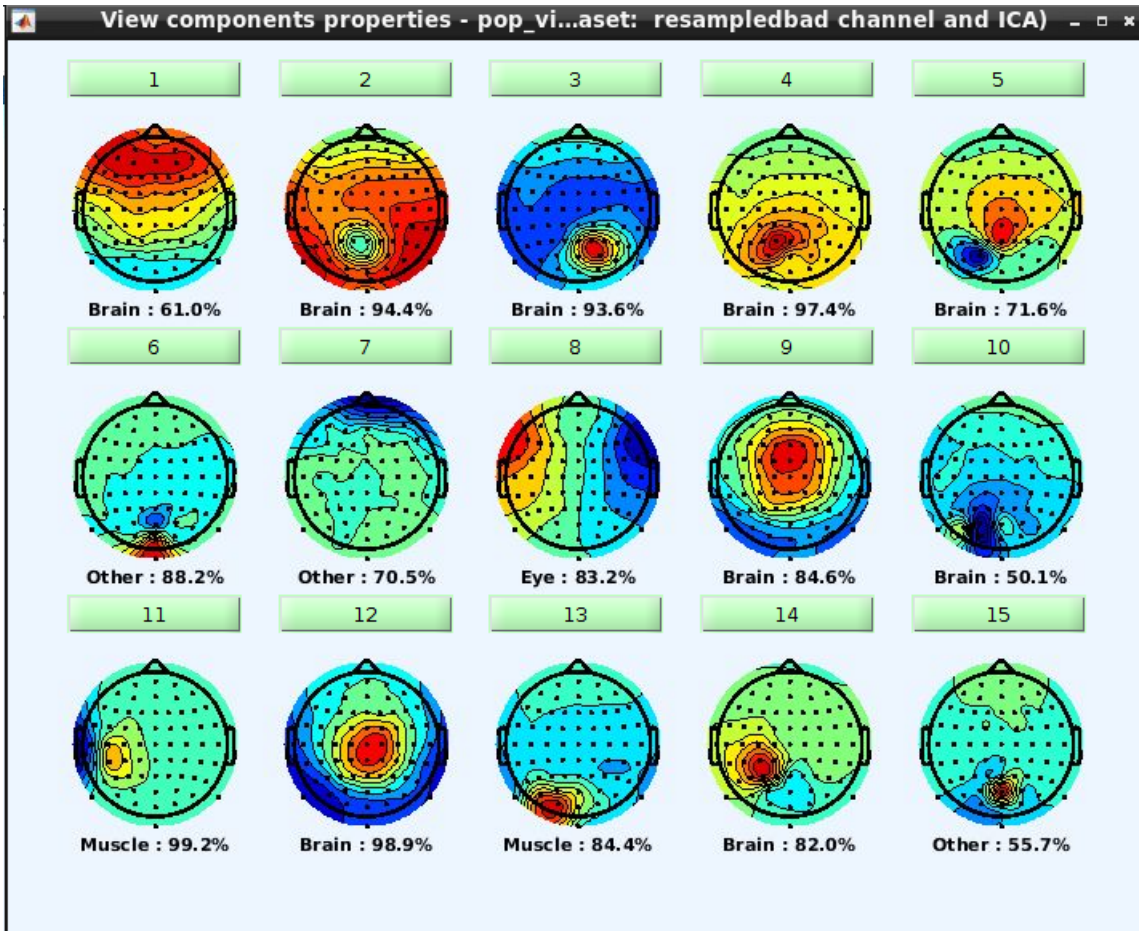


**Figure A.2:** Overview of the detected independent components by the IClabel algorithm in EEGLab software.

# B

# Appendix - CNN

## B.1 Activations of VGGish network layer

With Keract tool in Keras you can visualize the activations of each layer of a convolutional neural network. Here the VGGish network layers activation are shown for song ID 01.

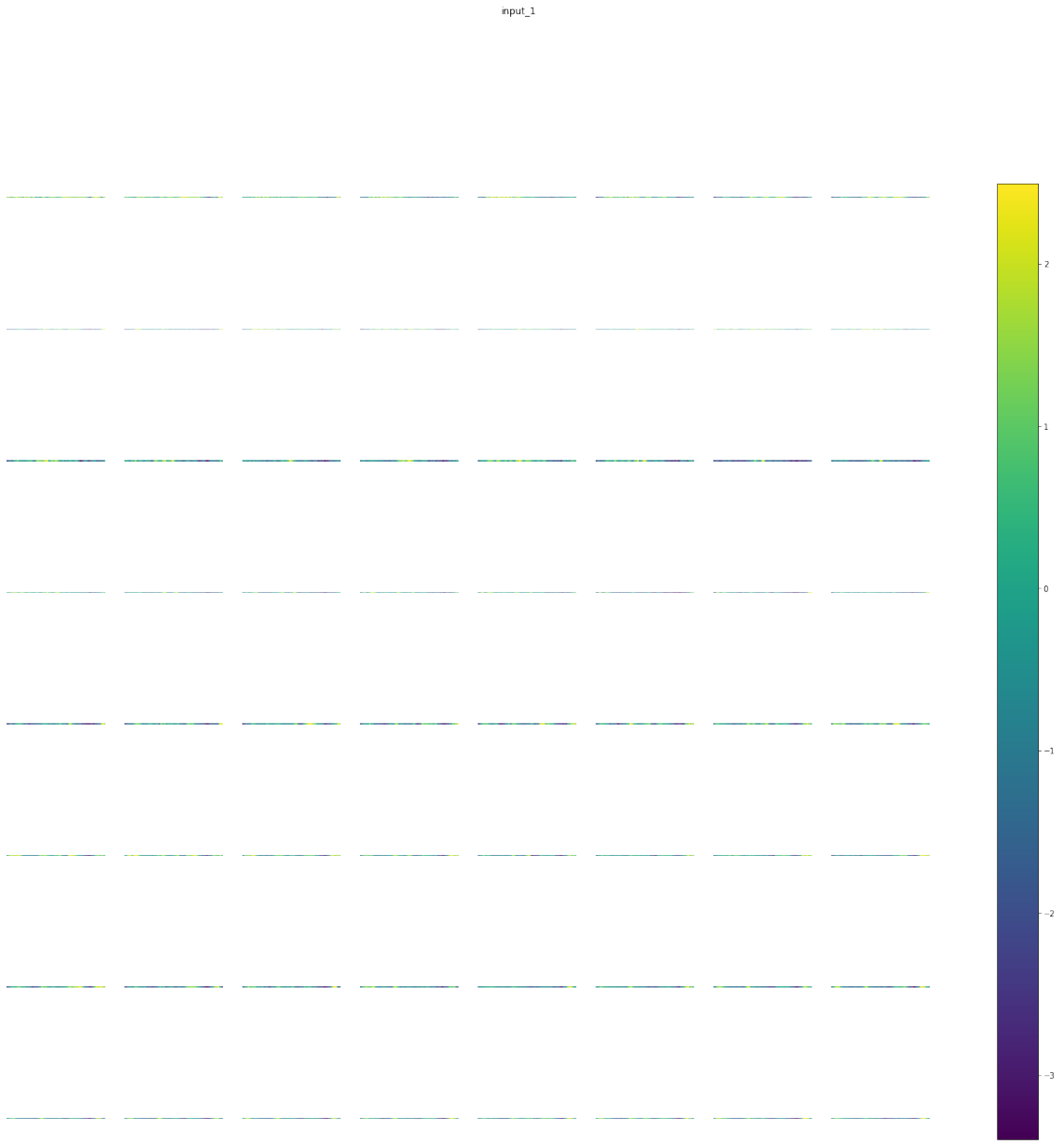


**Figure B.1:** Activations of VGGish layer: Input

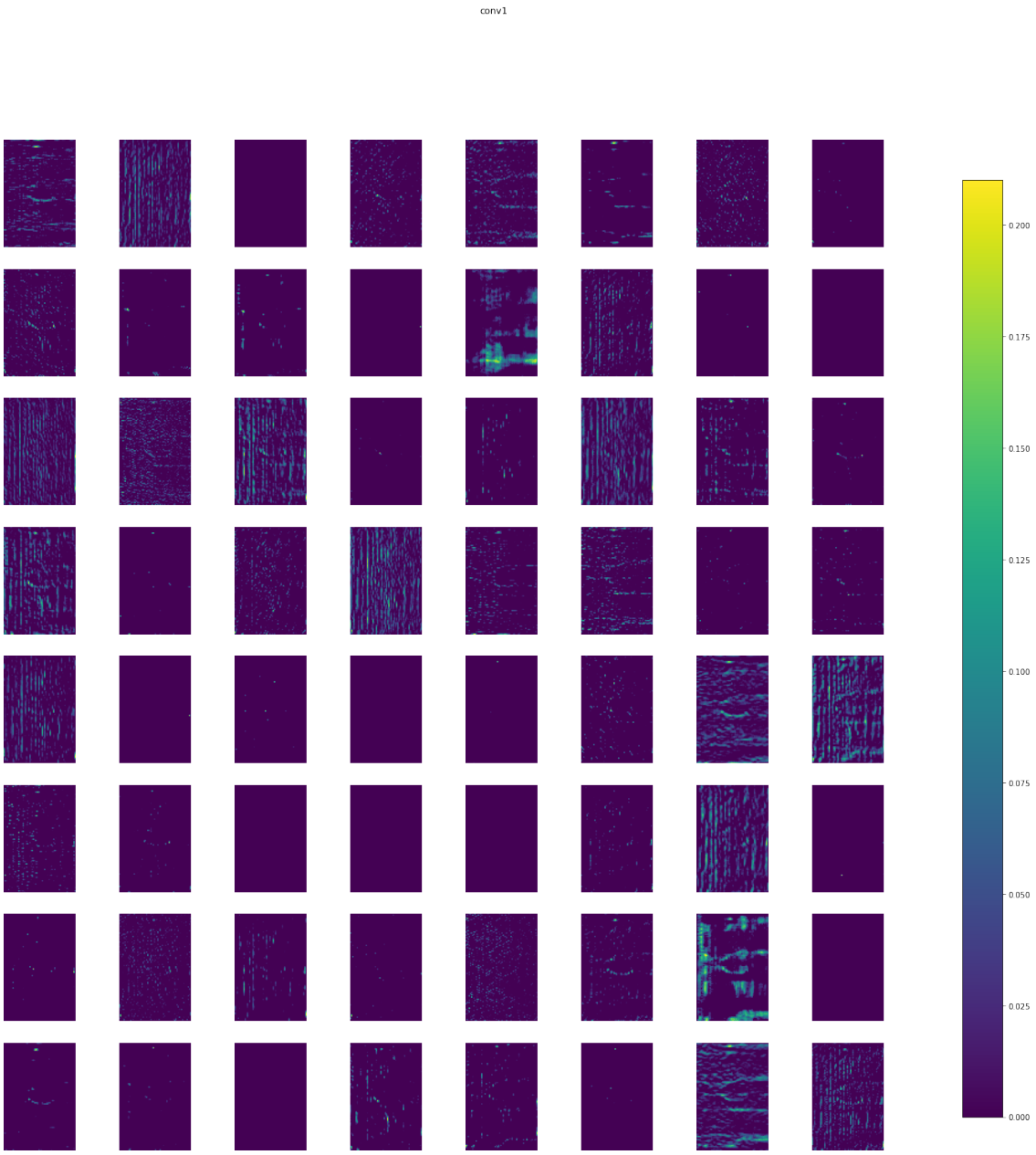


**Figure B.2:** Activations of VGGish layer: Conv1

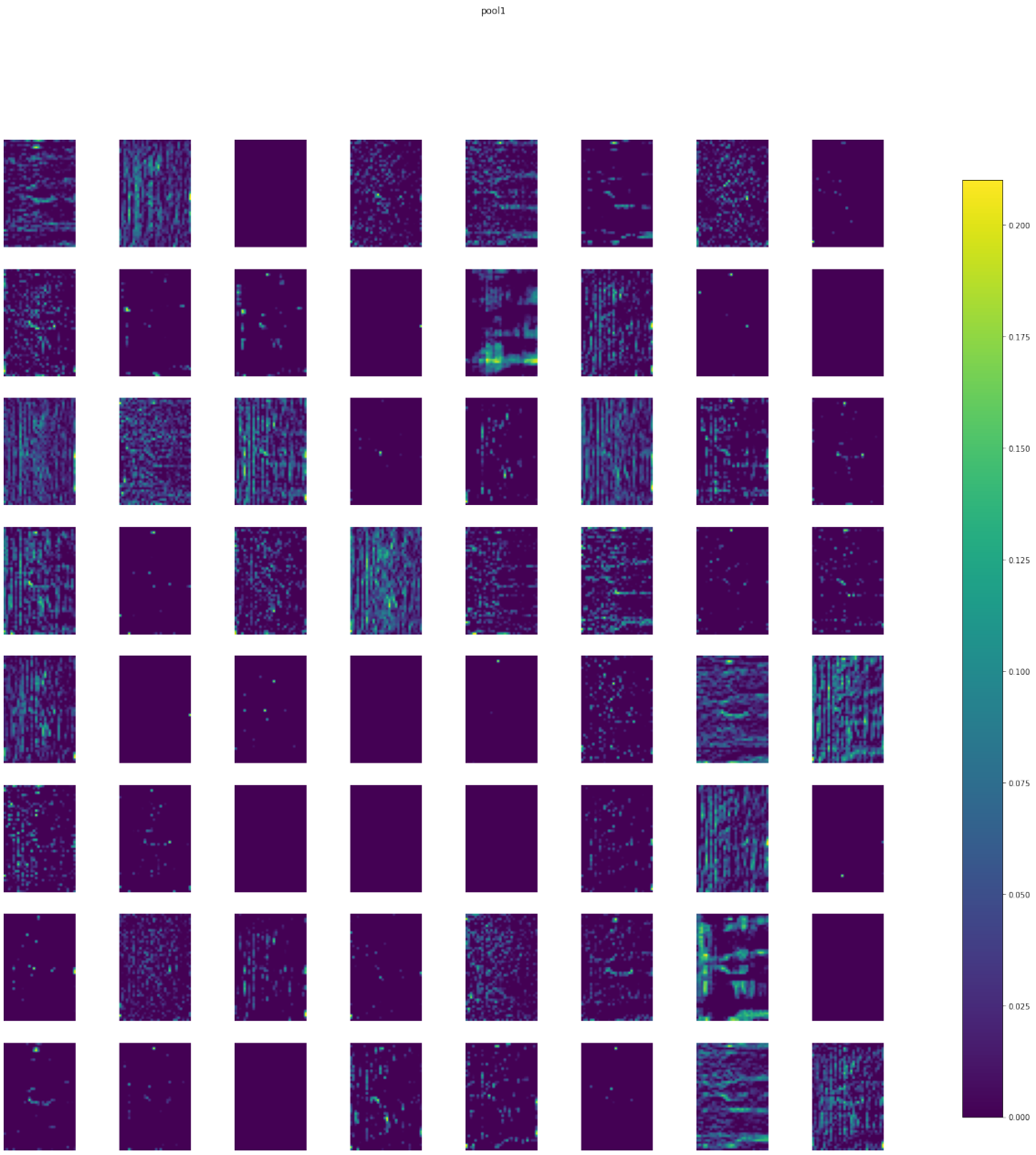


**Figure B.3:** Activations of VGGish layer: Pool1

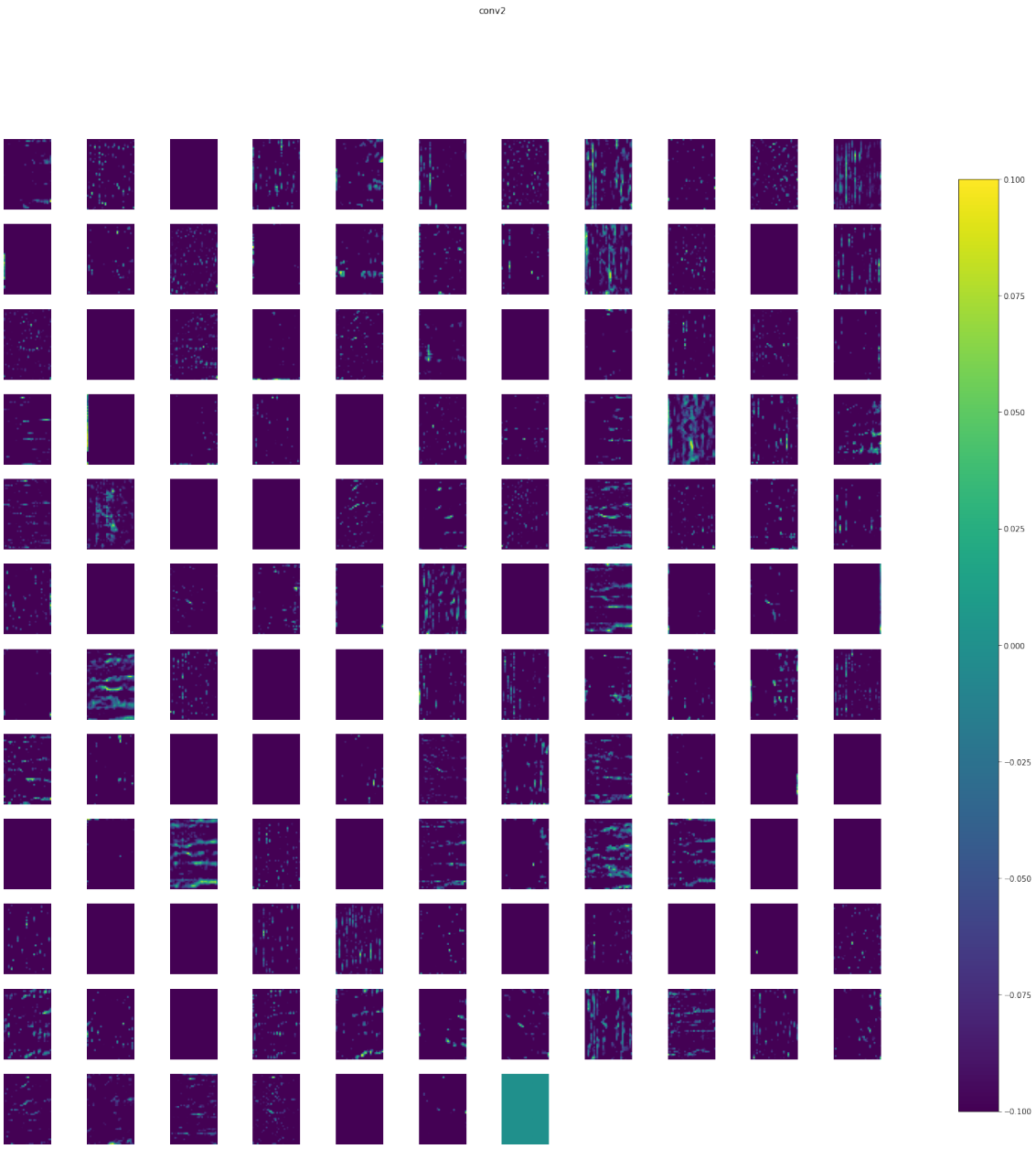


**Figure B.4:** Activations of VGGish layer: Conv2

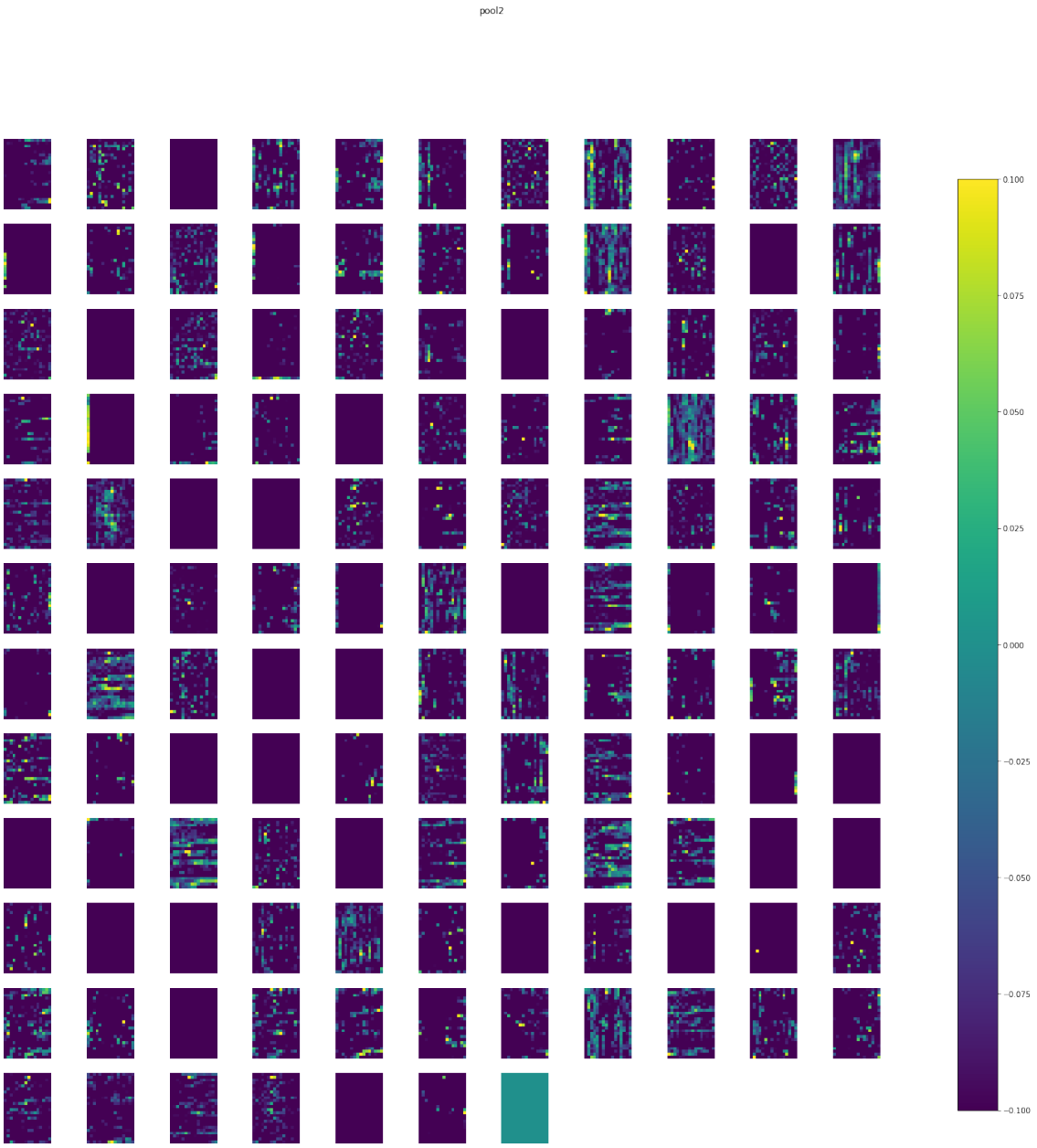


**Figure B.5:** Activations of VGGish layer: Pool2

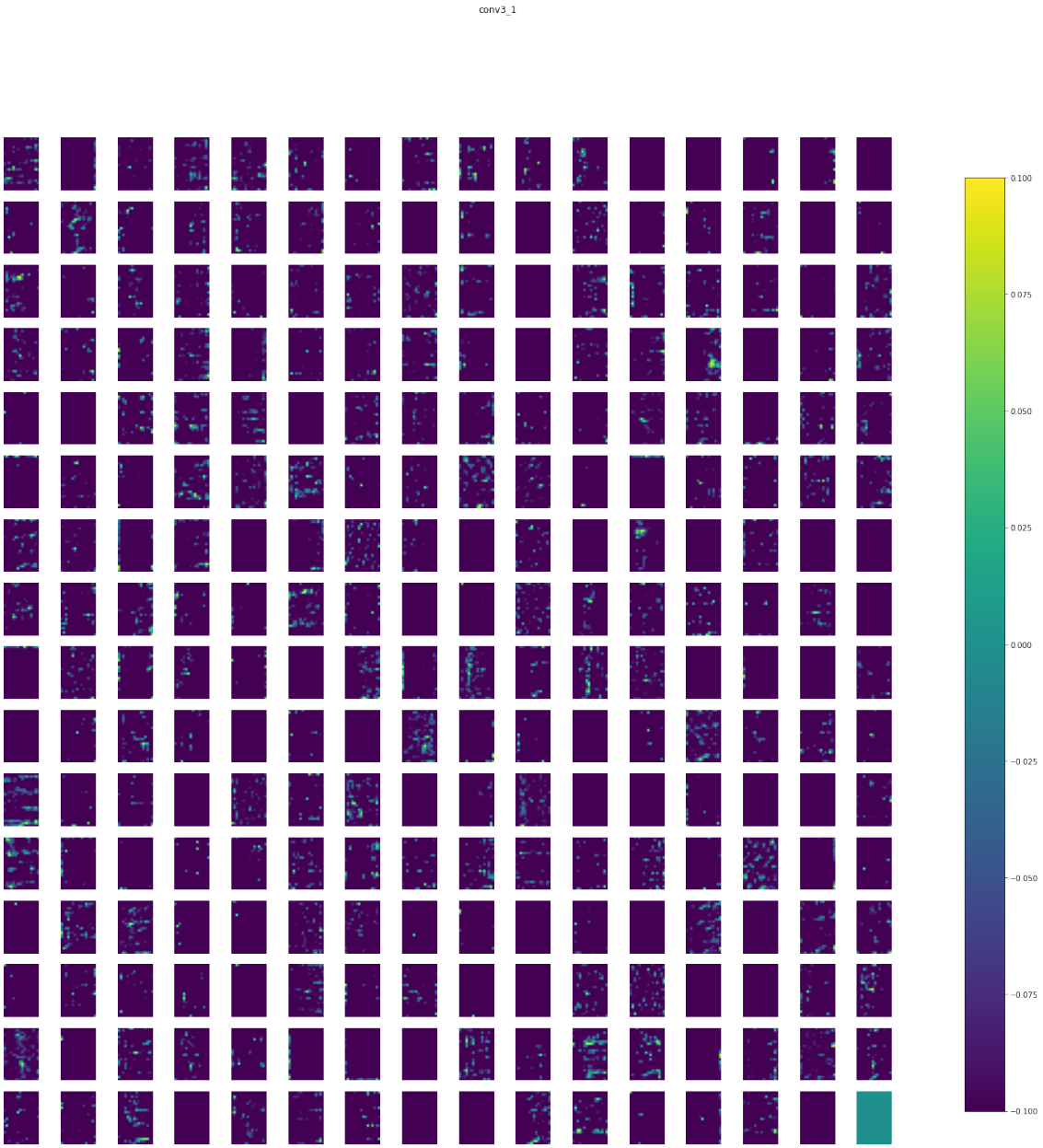


**Figure B.6:** Activations of VGGish layer: Conv3-1

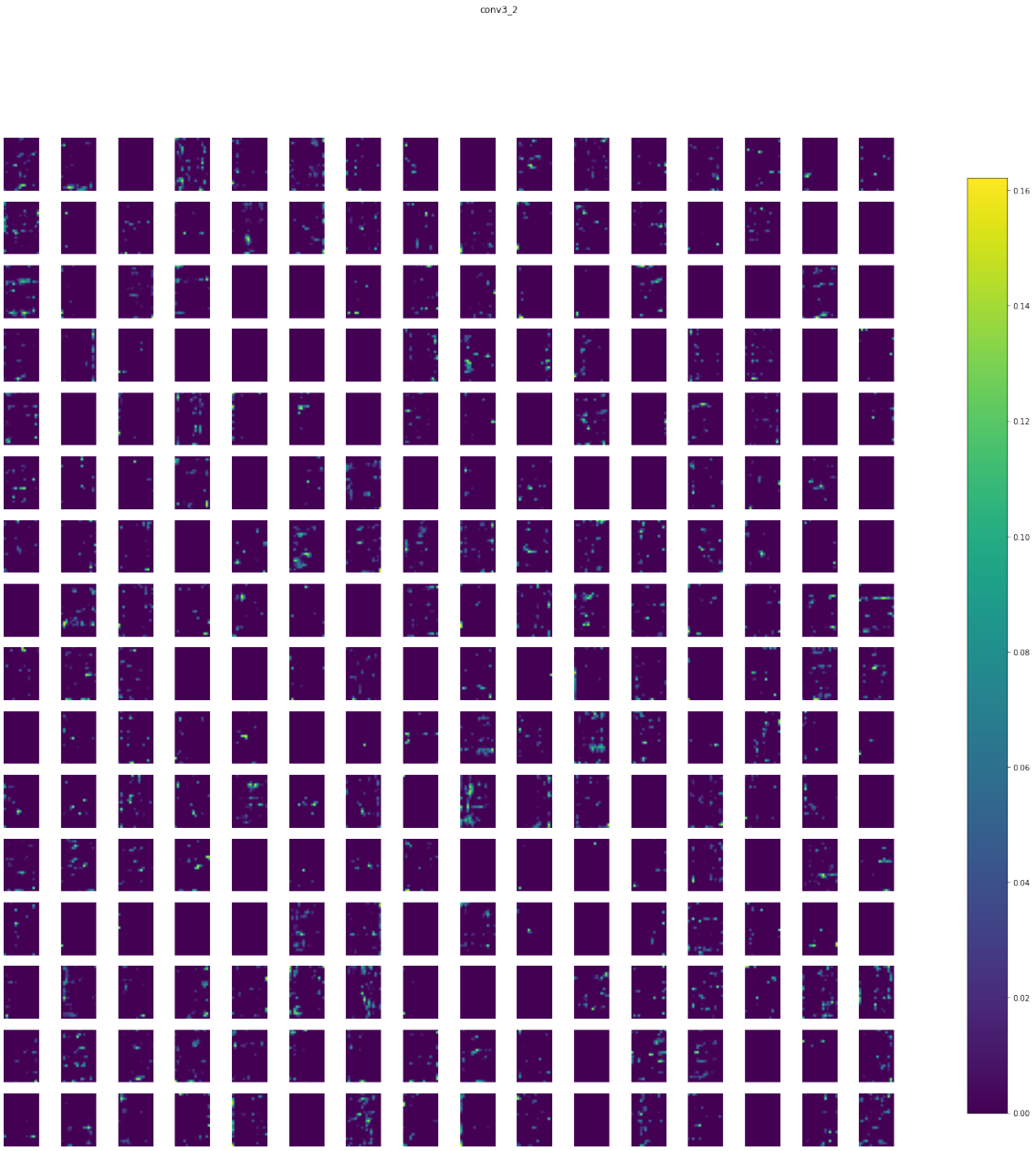


**Figure B.7:** Activations of VGGish layer: Conv3-2

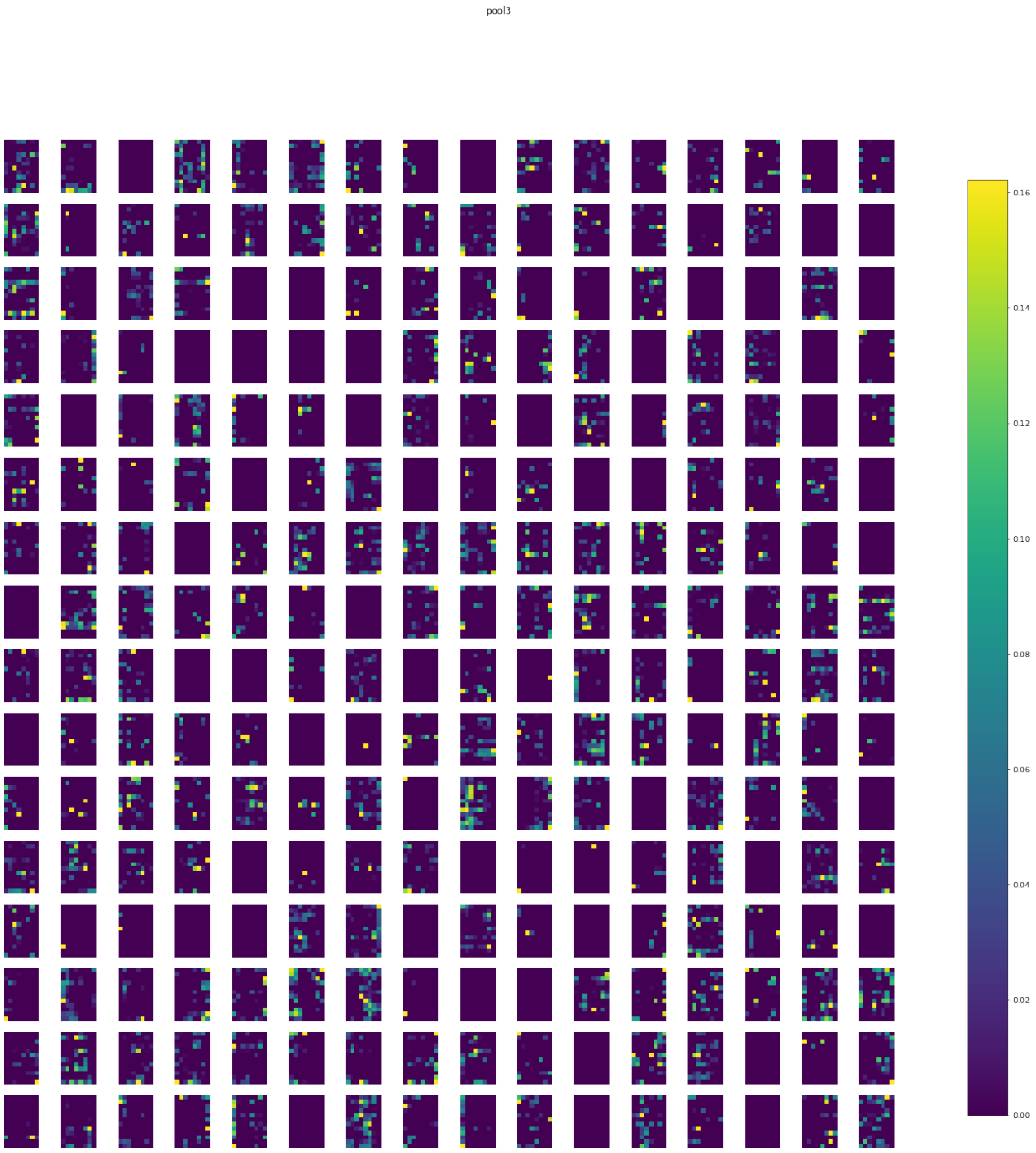


**Figure B.8:** Activations of VGGish layer: Pool3

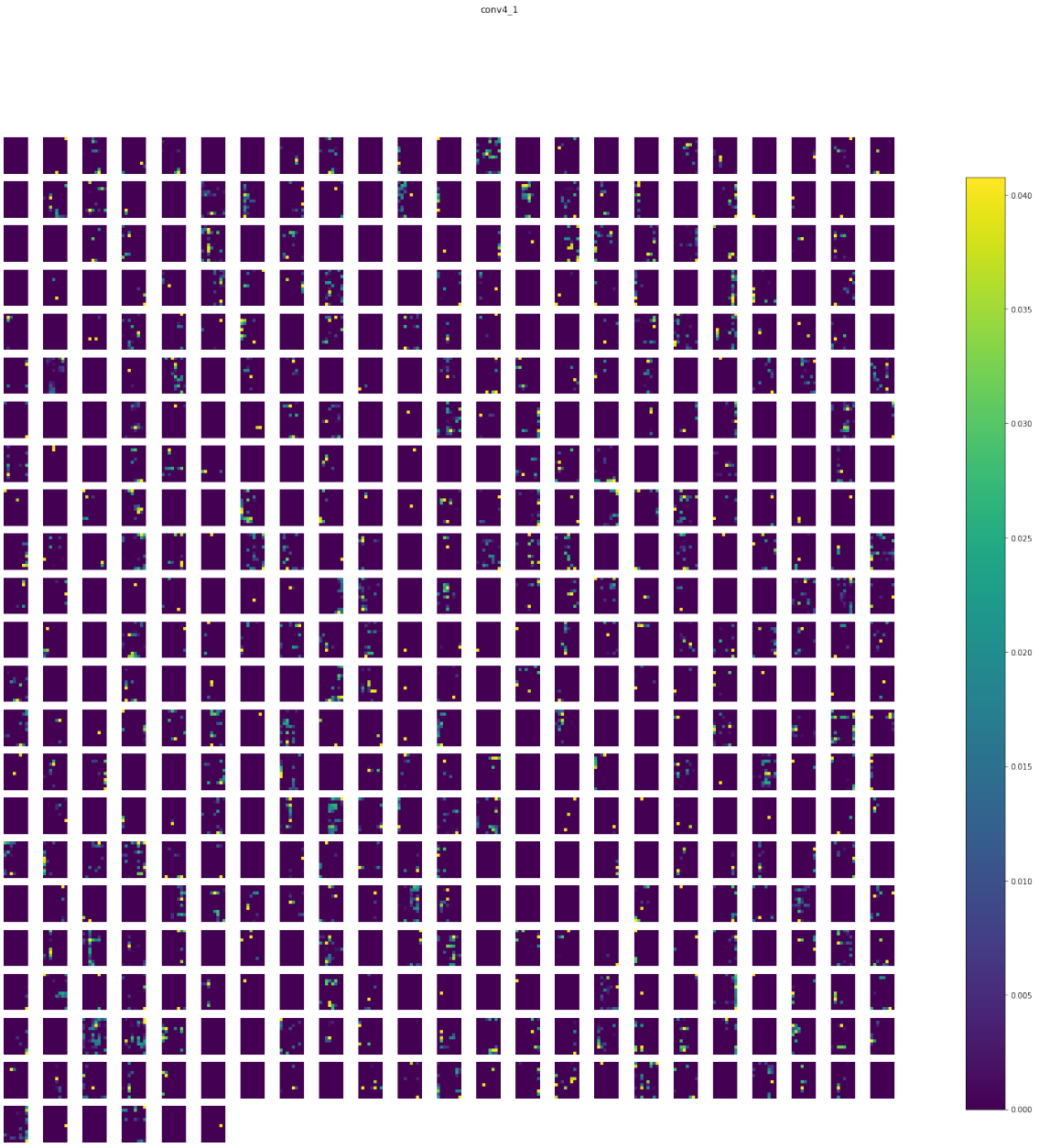


**Figure B.9:** Activations of VGGish layer: Conv4-1

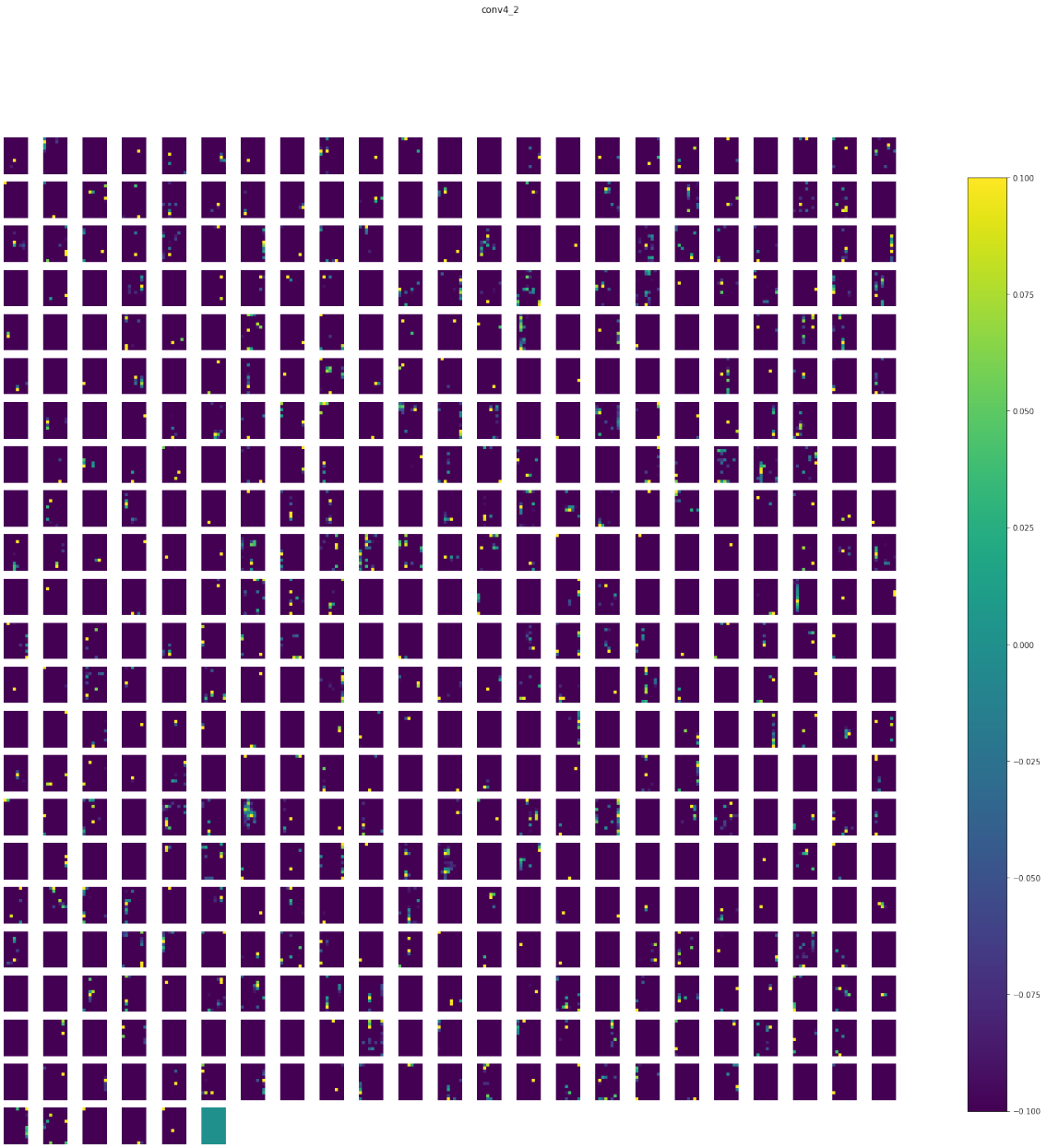


**Figure B.10:** Activations of VGGish layer: Conv4-2

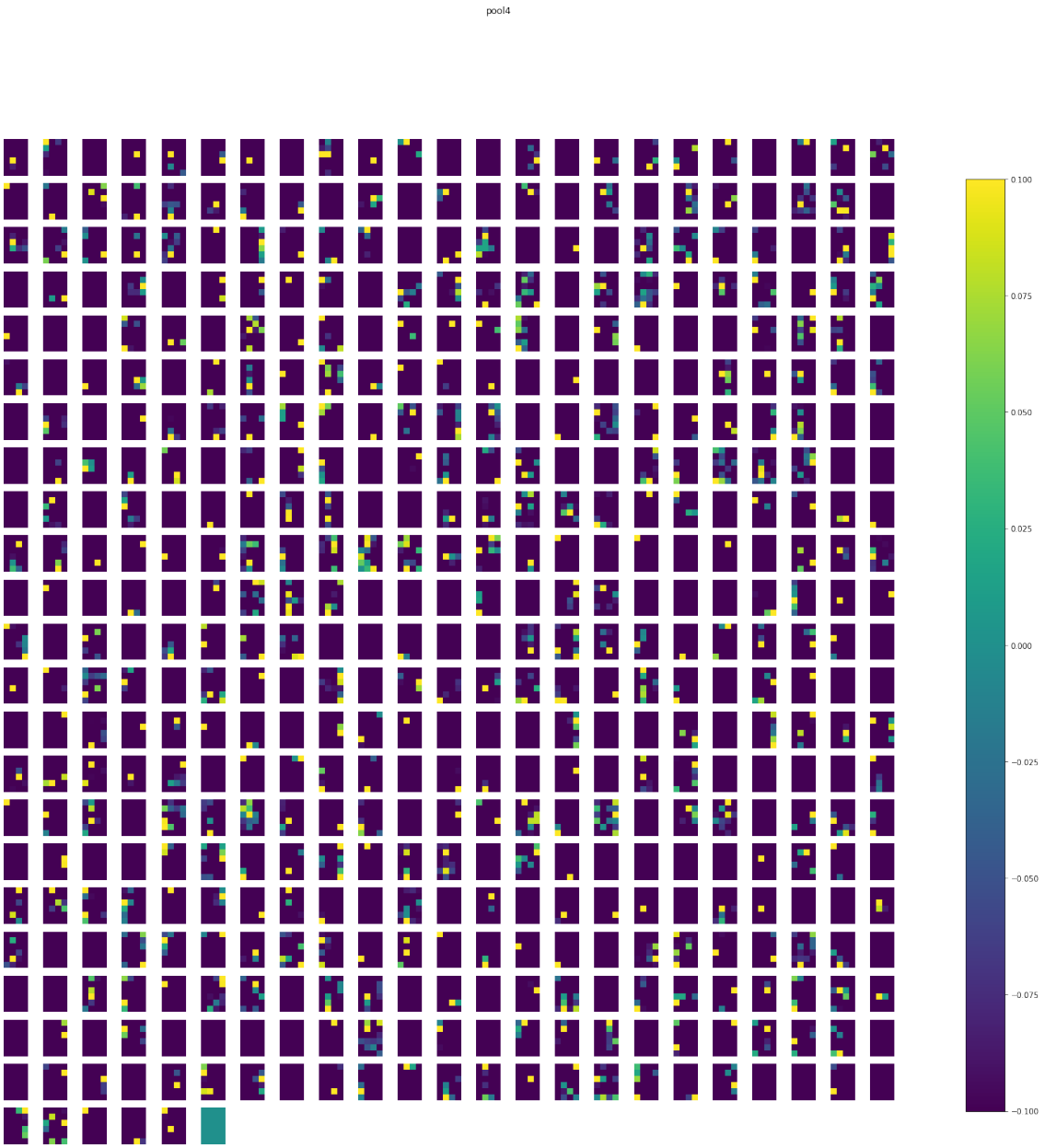


**Figure B.11:** Activations of VGGish layer: Pool4

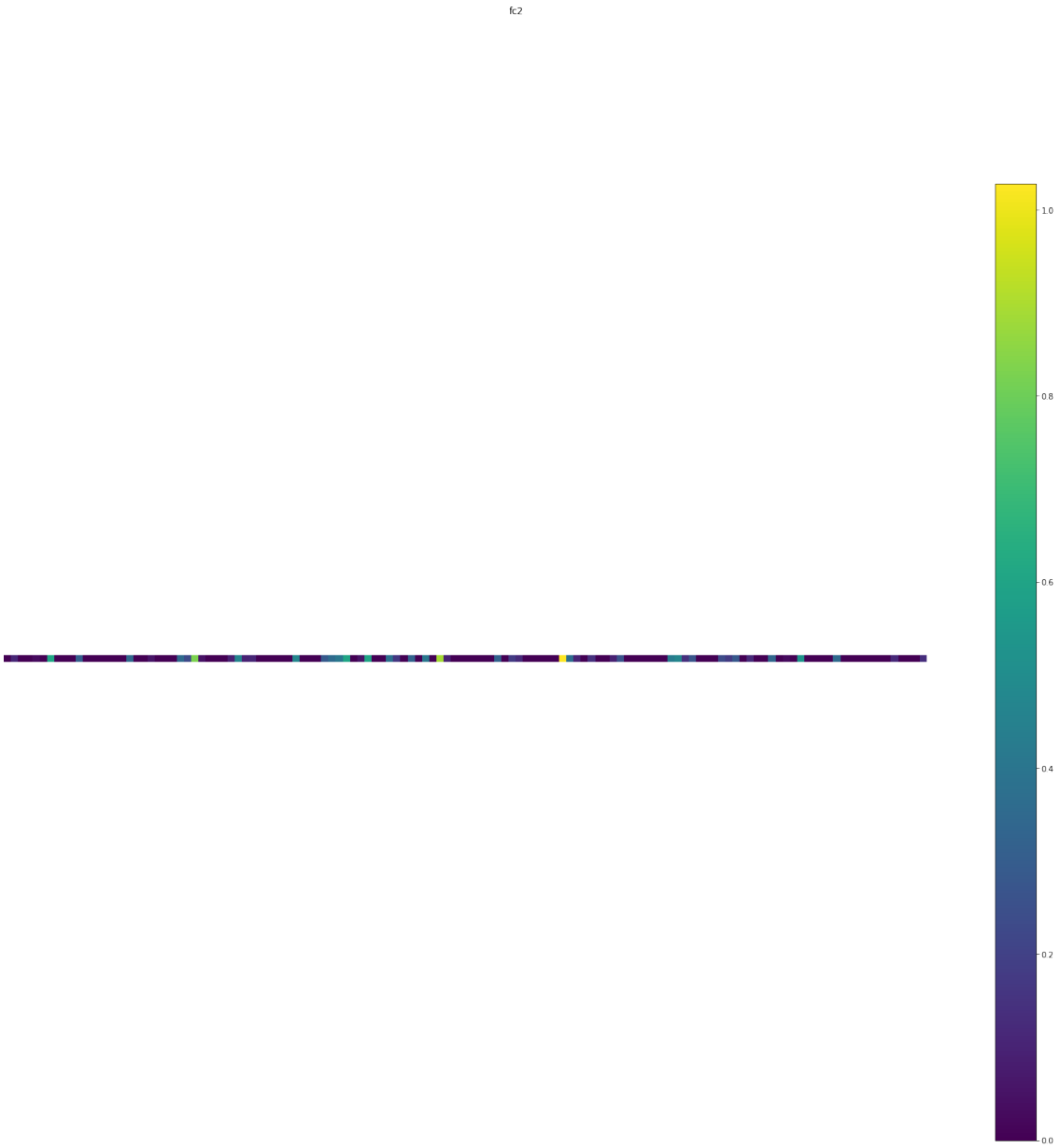


**Figure B.12:** Activations of VGGish layer: Embedding