GROUPE ESEO

Angers
Paris
Shanghai
Dijon

www.eseo.fr

2016 / 2017

NOM GONTIER……...

PRENOM FELIX……...

☐ Stage Sc. et Technique I2 (S7)

☒ Stage de Fin d'études I3 (S10)

Rapport de stage / emploi

>ENTREPRISE : **Laboratoire des Sciences du Numérique de Nantes**

>DATES : *Du 1er mars au 8 septembre 2017.*

>SUJET MISSIONS : *Privacy enforcement in large networks of acoustic sensors*

>TUTEUR ENTREPRISE : *LAGRANGE Mathieu – CR CNRS*

A COMPLETER UNIQUEMENT POUR LES STAGES I3 :

>Option : ☐ SE ☐ BIO ☐ OC-EOC ☐ NRJ ☐ CC-CR ☐ LD ☒ DSMT-SIAT ☐ IIT

>Niv. de confidentialité: ☐ **mon rapport est confidentiel de** Niveau : ☐ 1 (moy.) ☐ 2 (élévé)
Tel qu'indiqué sur convention signée. Non réponse = l'ESEO retient Niveau 0 (aucune conf.)

>Domaine de l'entrepr.: ☐ Auto. ☐ Aéron.. ☐ Banque-fi.-assur. ☐ Biomédical-santé
☐ Energie ☒ NTIC ☐ Télécoms ☐ ……..……

>autres points :
☐ stage à dominante '**management**' ou ☒ à dominante '**recherche**'
☐ année I3 effectuée sous 'contrat Pro' entreprise
☐ **mon tuteur sera présent à ma soutenance orale**
☐ **mon tuteur sera présent au déjeuner le jour de ma soutenance**

Acknowledgements

I am particularly grateful to Mathieu Lagrange who directed this internship. His close monitoring helped guide me throughout the many aspects of this experience.

I would like to thank Pierre Aumond at IFSTTAR for his broad contribution to the development of the coder, as well as to the written paper which he took the time to fully review several times. The fruitful discussions on fractional-octave band measurement error constituted an invaluable help.

I wish to thank Arnaud Can at IFSTTAR for his implication in the CENSE project and his interest in the present work. His insight and comments on the paper were particularly appreciated.

I express my gratitude to Jean-Rmy Gloaguen, PhD student at IFSTTAR/LS2N, for his sincere will to support my work with his extensive knowledge of acoustics.

I would like to thank Ludovic Charpentier, engineer at Wi6labs, for his patience regarding the embedded implementation of the proposed scheme.

I thank Catherine Lavandier of the Université de Cergy for her continued interest in my work, and along with Jean-Franois Petiot and Mathieu Lagrange for accepting to supervise my upcoming PhD thesis.

Finally, I would like to thank all the members of the Laboratoire des Sciences du Numérique de Nantes and of the CENSE project for their warm welcome into this internship and the research world in general.

Introduction

This report presents the masters internship proposed as part of the final year at ESEO with a two-fold objective. First, it was an opportunity to put to test both the technical knowledge and working skills learned during the five years of studies through an experience of long duration within a full company environment. Second, it aimed at finalizing the professional project developed at ESEO.

With respect to this last point, I had considered pursuing a PhD thesis in audio signal processing and machine-learning for a long time. The academic environment, the correlation with my expectations and the overall quality of ongoing works motivated my decision to apply at the Laboratoire des Sciences du Numérique de Nantes. I had the possibility to join the CENSE project and contribute to research on an ever-rising problem: noise management in large cities and more generally the understanding of urban soundscapes.

The internship's purpose was to enable continuous transmission of measured audio data through a city-scale sensor network. An audio coding scheme was necessary to satisfy several constraints linked to the size of the data, its information content for further processing concerns and the embedded hardware capabilities.

The first part of this report will provide contextual elements to the internship. A detailed explanation of the work contributing to the subject will be presented in the second part. Lastly, a third part will briefly reflect my view of the experience, as well as the many issues and opportunities faced during these past six months.

Internship summary

Subject: Privacy enforcement in large networks of acoustic sensors

This internship aimed at developing an audio processing and encoding scheme to enable continuous data recording in a large-scale embedded sensor network, as part of project CENSE. Among the technical constraints were low computational complexity, low output bitrate, and the ability to perform acoustic monitoring and sound event recognition tasks with the transmitted data. Additionally, decoded speech utterances were to be unintelligible to ensure the privacy of citizens. The involved themes are audio signal processing, acoustics, data coding and speech intelligibility.

Company

Laboratoire des Sciences du Numérique de Nantes - École Centrale de Nantes - Nantes, France.

Dates

The internship took place between 1rst March 2017 and 8th September 2017, for a total duration of 28 weeks.

Team and contacts

This internship was mostly composed of individual work. Regular exchanges were planned with my referent professor, M. Lagrange, to discuss about progress and planning concerns. I also worked with other members of the project from June to the end, including engineers at Wi6labs regarding an embedded implementation of the studied scheme and A. Can/P. Aumond at IFSTTAR about acoustic indicators and more generally the writing of the scientific paper.

Summarized planning

- First two months: bibliographical research on acoustics and audio signal processing, study of datasets and classification methods, first version of the coder without computing acoustic indicators.
- To early May: study of the coder's parameters effects on intelligibility indicators, classification tasks and bitrate.
- Mid May to early June: computation of third-octave bands and paper writing beginning.

- To the end of the internship: paper writing, intelligibility, measurement errors and sharing of this work with the project's teams.

Results

The coder is completed, its parametrization as well as embedded implementation is being discussed with other members of the project. A study of the multiple involved factors is conducted with regard to the technical constraints. A scientific paper is also redacted and will be proposed for peer review and publication in the near future.

Contents

List of Figures

List of Tables

Part I
Context of the internship

1 LS2N: Laboratoire des Sciences du Numérique de Nantes

The Laboratoire des Sciences du Numérique de Nantes (LS2N) which hosted this internship is a joint research unit (UMR6004) created in january 2017 as a result of the fusion of two research laboratories: the Institut de Recherche en Communication et Cybernétique de Nantes (IRCCyN) and the Laboratoire d'Informatique de Nantes Atlantique (LINA). It is under the supervision of the CNRS, the University of Nantes, the IMT, the École Centrale de Nantes (ECN) and holds a partnership with the INRIA. The laboratory's activities are currently distributed over 5 sites in the city of Nantes, with the most important being located within the ECN's campus.

The LS2N's creation aimed at associating the local expertises on computer science and cybernetics. For this reason it contributes on a wide range of research areas:

- System Design and Operation (CCS) which focuses on embedded and industrial systems as well as general automation.
- Robotics, Processes and Calculation (RPC) which studies the interactions of robots with their environment,
- Data Science and Decision-making (SDD), with topics such as data modelling and classification,
- Signals, Images, Ergonomics and Languages (SIEL) that takes interest in particular types of data related to human interactions and perception,
- Software and Distributed Systems Science (SLS) which aims at solving software engineering challenges.

The domains of application are: industry of the future, management of energy and environmental impact, life sciences, vehicle and mobility, design, culture and digital society are all topics on which the LS2N has an expertise and influence.

The LS2N regroups 450 staff including researchers, engineers and PhD students. A total of 22 teams exist, each with its own members, research themes and projects. Lastly, the laboratory is involved in several academic programs in Nantes.

2 The CENSE project

The need to monitor and reduce noise pollution is nowadays a major concern, especially in urban environments where it is the most intense. To this aim, a few methods are commonly used to gather data relative to noise. First, noise maps are an efficient way to assess environmental boise pollution, and are regularly produced for large cities as required by the European Commission [19]. Current noise maps are mainly simulated with high-precision geographical data. They however rely on simplified sound source and propagation models that reduce the accuracy of the data. Additionnaly, the models do not consistently correlate with the human perception of a given sound event. Alternatively, observatories sample real sound signals. While these measurements are more accurate, they are very scarcely distributed in space and are thus insufficient to fully characterize soundscapes. Moreover, the existing observatories are often expensive to install and maintain.

Figure 1: The project's logo.

The CENSE project, funded by the ANR (French national research funding agency), aims at proposing a viable solution to both noise maps construction and urban sound environment assessment. The main idea being developed is the deployment of a city-scale, dense low-cost sensor network to allow both measurement accuracy and geographical precision for noise maps. The recorded audio and physical indicators are to be used together with predicted data to combine these sources' advantages: accuracy and highly precise spatial information. The studied sensors are embedded low-power devices with wireless communication systems that are placed on public street lights. The city of Lorient was selected to develop this project.

The project is divided in six parts in its organization. Besides coordination, five interconnected units fragment the research objectives into development axes:

- The **Data and Modelling** intends to evaluate and improve uncertainties linked to simulated data by accessing exact measured values.
- The **Measurement network** of which the present report is a part of aims at implementing a sensor network to match the above-mentionned needs and requirements.
- The **Data assimilation** and network optimisation consists in associating the two previous units' outputs, respectively simulated and measured data, to best estimate noise levels in the considered environment.
- The **Characterisation of sound environments** must use available data to better assess urban soundscape contents through indicators such as enhanced noise maps.

Finally, to promote accessibility and reproductibility of this work, another unit aspires to propose a software platform with all the project's data and a map application.

The project is coordinated by the Laboratoire d'Acoustique Environnementale (LAE) at IFSTTAR, although several other entities take part in its development. Their roles and actions are the following:

- The LAE and IFSTTAR also work on sensor placement and acoustical measurements processing,
- The CNRS through the LS2N on data encoding with this internship and soundscape characterization,
- Bouygues Energies and Services provide the energy and locations of embedded systems,
- Wi6Labs implement the sensors and their hardware and software contents,
- The Agence Nationale de la Recherche (ANR) funds the project.

Multiple disciplines and parties are involved with which this internship collaborates closely.

Part II
Detailed activities

1 Objectives

This section describes the objectives pursued in the scope of this project.

1.1 Audio coding in large scale sensor networks

In the context of sensor network at a city scale, an important problem is to process the large amounts of audio signals that are continuously recorded. As the data is to be transmitted through wireless communication, the maximum bitrate allowed to a single sensor is severely insufficient for raw sound measurements. The scope of this work is to develop an audio coder so that all the data can be transferred to a remote server from the sensors.

The project primarily aims at processing data through two tasks: acoustic monitoring and audio event recognition. As will be developed later in this report, these two applications are different in the physical and mathematical concepts and quantities they involve. However, a common point is the use of relevant information extracted from audio signals instead of the raw data itself. This enables the use of transform functions to reduce bitrate. The proposed scheme must therefore employ a representation of the data in a way that efficiently makes both tasks possible.

Due to the coder being targeted to be implemented on embedded hardware other subsidiary constraints emerge, namely computational and memory costs. The target is a STM32L4 microcontroller with Cortex-M4 processor powered by solar energy. Operations are thus limited as well as data storage.

Finally, as sensors initially sample sound pressure, the privacy of citizen is an important concern. A study will be conducted on the intelligibility of encoded speech to ensure that both speakers and utterances content cannot be recognized.

Concerning implementation constraints during the project, they are as follow: the coder prototype is to be developed in Matlab, and evaluated using the expLanes [32] framework. expLanes principle is to decompose projects into steps with a number of variable factors as inputs. A well defined project architecture is proposed which allows simple runs on different machines and operating systems.

It also provides tools for results observation, report generation and execution parameters such as parallel processing. A GitHub repository must be maintained for both the project and a scientific paper presenting its characteristics.

1.2 Acoustic monitoring

The considered sensor network primarily aims at monitoring urban sound environment, that is, continuously assessing their content and impact on the population. In the literature, this is typically performed by measuring energetic acoustic indicators such as the equivalent sound pressure level L_{eq} in dB SPL or its A-weighted equivalent L_{Aeq} in dBA. However, while these indicators have proven to correlate well with perceptual evaluations for negative impact sound environments [38], they are not sufficient to fully describe urban sound environments [40]. Many other variables can be derived to better account for previously implicit properties [15] including percentile values or time dynamics. Studies have been conducted to select relevant subsets of descriptors in sound environment characterization [16, 13, 34]. The mentionned L_{eq} features are measured at periods of 0.125 ms or 1 s, respectively fast and slow as a convention.

Most measured or derived indicators describe the global urban soundscape and lack information on its composition. Sound sources recognition enables a more efficient understanding of soundscape properties. For instance, a bird song and traffic noise with the same energy can be perceived very differently. The analysis of sources within a sound environment is generally linked with spectral content [26] and high temporal measurement resolutions. Other indicators can correlate with specific sound environment content [7] but do not achieve the same performances as a source detection scheme in characterizing soundscapes.

A solution is the calculation of spectral energetic indicators such as the 31 third-octave bands within the human audition range 20 Hz - 20 kHz. Slow or fast third-octave bands measurement appears suitable for this work's purpose: in addition to being a relevant descriptor [46] for urban sound environment assessment, it allows for the computation of most cited indicators while representing reasonably small, fixed amounts of data to be transmitted. It additionnaly contains spectral information that can be used for source recognition tasks. However, as a tradeoff other representations such as Mel or Bark spectrograms, as well as psychoacoustical indicators cannot be precisely estimated.

1.3 Audio event recognition

As discussed in the previous section, the recognition of sources of interest is of importance to fully model the studied soundscapes. As far as the encoding scheme is concerned, it is thus primordial that the encoded data allows us to compute the above cited indicators but also to detect the different sound sources with state of the art methods.

The recognition of sources of interest from audio streams has been subject of extensive research in the past on speech [5], music [47], and lately more complex scenes in which the current work falls. Studied classification methods are diverse, ranging from time-dependant modeling with hidden markov models [35] to "bag-of-frames" approaches [6, 22]. Common architectures include learning-based classifiers such as support vector machines [30], Gaussian mixture models [37] or neural networks [43, 36]. However, the selection of relevant features is still an open debate. The most used are certainly spectral [29] or cepstral [20] representations of the signal. Among them, Mel spectrograms and their cepstral-domain derivation, the Mel-frequency cepstrum coefficients (MFCC), are the most recurrent. These representations effectively model the human cochlear response to sounds by grouping frequency components around critical bands in a logarithmic scale. While parameters are empirical and vary in the literature, a reference implementation includes 23.2 ms analysis and a 40 Mel bands resolution. The resulting representations are generally more precise in time and frequency than in acoustic monitoring applications because it is a primordial factor for classification whereas acoustic indicators must be computed more accurately.

They may also be exploited together with features computed in other domains to better model signal properties. For instance, [14] adds spectral features related to harmonicity and salient frequencies, and [18] uses a matching pursuit (MP) algorithm to deduce time-domain features. Another promising solution is feature engineering via unsupervised learning, which [42] implements with a k-means clustering technique. Alternative data representations such as the scattering transform [9] show good results in environmental sound classification tasks [41].
A more detailed review of used methods is available in [17].

1.4 Intelligibility

Intelligibility in decoded and reconstructed audio is also an important concern of this study. It is indeed important that the proposed scheme ensures a high level of recognition of acoustic events but, as the data is transmitted over the network and potentially stored, the level of intelligibility of the decoded stream should be

as low as possible.

The use of an audio recording as an evidence is considered by the forensic phonetics [8] domain. However, even with a good recording quality audio remains a weak biometrical indicator [12]. Still, even if the identity of the speaker cannot be asserted, one can attempt to transcribe the spoken utterances. As such, it is important to demonstrate that the proposed encoding scheme effectively degrades intelligibility.

In order to ensure this property, several approaches can be considered. First, a speech versus non speech detector can be implemented on the sensor. If speech is detected, the data is not transmitted. This concept is however prone to failures of the detector and leads to the unavailability of data during speech periods. This is undesirable as it might compromise the computation of indicators which require time continuity.

Second, as the speech is mostly concentrated at frequencies ranging from 1 kHz to 4-5 kHz, a band-rejection filter located in this region could be considered in order to remove the formant information [28]. However, as some frequency bands are no longer available, the computation of standard acoustic indicators is no longer possible.

A third approach is to wisely choose the frame rate. Since a phoneme has a duration of about 100ms to 200ms [31] [39] a frame rate lower than 4 Hz should dramatically reduce intelligibility.

2 Coder scheme

This section details the composition of the proposed coder scheme, as well as an introduction on the employed concepts and methods.

2.1 Data representation

2.1.1 Short-Term Fourier Transform

To this day, the most commonly used tool for audio signal processing is the Fourier transform. This operation provides a mathematically simple, linear frequency analysis, with low computational complexity while remaining invertible. Despite the advent of more advanced methods such as the wavelet transform [10], most audio analysis, classification and synthesis schemes are thus still based on this transform.

The two descriptors studied in this report can be obtained from the Fourier transform of a given input signal. As such, the discrete variant of this operation will be used as the first step of the coding scheme.

The Discrete Fourier Transform (DFT) of a signal $x[n]$ of arbitrary length $N > 0$ is defined as

$$X[k] = \mathcal{F}\{x\} = \sum_{n=0}^{N-1} x[n] e^{-i2\pi k \frac{n}{N}}, k \in [0, N-1] \tag{1}$$

where k represents the frequency points. The first $N/2+1$ points correspond to the $[0, Fs/2]$ frequencies linearly spaced with a factor of Fs/N for a given sampling rate Fs. Because the studied signals are real-valued, the remaining $N/2-1$ points are the complex conjugates of the first half of the representation. The *resolution* of an analysis is defined as Fs/N and provides information regarding the capacity to differenciate two close frequency components in the signal. In some cases the signal will be modified to increase the *precision* of the analysis, *i.e.* the interval between two adjacent frequency points. This can be obtained by zero-padding the signal, but is generally done for practical purposes as it does not increase resolution.

However, in practice the processed signals are rarely stationnary, meaning that the information content of the signal varies with time. The DFT does not account for such variations, and the analysis of a long signal using this transform is often inefficient. The Short-Term Fourier Transform (STFT) is introduced to solve this concern by applying a DFT on fixed-length portions of the signal. Its expression is

$$X[m, k] = \sum_{n=0}^{N-1} x[n] w[mR - n] e^{-j2\pi k \frac{n}{N}} \tag{2}$$

where w is a windowing function used to isolate analysis frames and R represents the hop size in samples, that is the time interval between two consecutive frames. Unlike the DFT where an implicit unit rectangular window spanning over the whole signal is applied, the STFT explicitely requires a window function as a separate parameter.

When applying a Fourier transform on short signals, framing effects can appear due to the windowing function. The convolution theorem states that an element-wise multiplication in the time-domain is equivalent to a convolution in the Fourier space. For this reason, windowing effectively convolutes the pure frequency representation of the signal by the Fourier transform of the used function. This produces an effect known as spectral leakage. Figure 2 shows the Fourier transform of common windows. The rectangular window has a narrow main lobe,

so that a FFT bin value is unaffected by its neighbours. However, its sidelobes gains are high (the first peak at -13 dB compared to the main lobe) and even far frequency components have an impact on a given bin. Conversely, the Hann window has a larger main lobe but the sidelobes rapidly decrease in amplitude. Finally, the gaussian window compromises between the two by displaying a better attenuation on the first sidelobes.

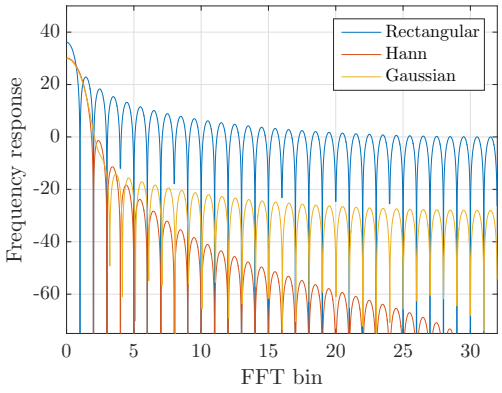


Figure 2: Comparison of common windowing functions and their effect on the Fourier transform.

Another property of the Fourier transform is the conservation of energy. Parseval's theorem states that the energy of the Fourier transform is equal to that of the time-domain signal up to a multiplicative constant:

$$\sum_{n=0}^{N-1} |x[n]|^2 = \frac{1}{N} \sum_{n=0}^{N-1} |X[k]|^2 \tag{3}$$

where N is the length of the signal x. This is a necessary condition to the computation of third-octave bands for acoustic monitoring where the sound level is the main studied measurement.

In the coder application, the STFT of a continuously sampled signal is computed. As the studied descriptors depend only on the magnitude of the resulting spectrogram, the phase is discarded as well as the conjugate half. The choice of framing parameters as well as the windowing function will be discussed in the results section.

2.1.2 Third-octave analysis

The third-octave bands analysis consists in the application of bandpass filters with precise gains and cutoff frequencies. The measurement is then computed as the

energy of each filtered result. The specifications for such filters are given by the ANSI S1.1-1986 [1] and IEC 61260-1:2014 [3] standards. Additionally the IEC 61672-1:2013 [2] standard specifies band-wise tolerances determining the class of sound level meter developed: class 1 requires highly accurate measurements while class 2 has softer constraints. Globally the error should be low around the reference frequency of 1 kHz and is allowed to be larger at the edge bands, particularly at very low frequencies of 20-50 Hz.

Time-filtering

The usual method [21] to perform fractional-octave analysis involves the design of time-domain FIR filters. The filter coefficients are computed for the highest desired octave and applied on progressively time-decimated versions of the input signal.

This operation is closely related to wavelet analysis in both design and complexity. However, its main limitation is given by the constraints of time-domain filter design such as ripples, slopes and causality conditions.

FFT-based filtering

An alternative method is proposed in [4] using the Fourier representation of the signal. The filters can then be defined as constant frequency weights, and the filtering operation as a matrix multiplication.

The cutoff frequencies are shared by adjacent bands. The frequency weights matrix is thus designed around the corresponding points k_i as follows:

$$G_i(k_i + p_i) = sin\left(\frac{\pi}{2}\varphi_l(p_i)\right) \tag{4}$$

$$G_i(k_{i+1} + p_{i+1}) = cos\left(\frac{\pi}{2}\varphi_l(p_{i+1})\right) \tag{5}$$

$$\varphi_l(p_i) = sin\left(\frac{\pi}{2}\varphi_{l-1}(p_i)\right) \tag{6}$$

$$\varphi_0(p_i) = \frac{1}{2}\left(\frac{p_i}{P_i} + 1\right), p_i \in [-P_i, P_i] \tag{7}$$

where P_i determines the frequency range where the weights are between 0 and 1 around each cutoff frequency, and the l parameter controls the slope of the resulting filters. All other frequencies are assigned a weight of 1 between the increasing and decreasing part of a band, and 0 otherwise. Figure 3 shows an example of bandpass filter computed with this process for different values of l and compared

to Couvreur's [20] time domain implementation.

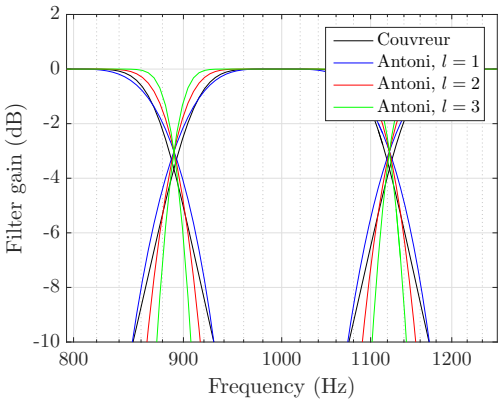


Figure 3: Comparison of Couvreur's and Antoni's implementations of third-octave filters. Frequency-weighting allows for arbitrary transfer functions and thus more accurate gains as standards impose.

This design ensures the following properties $\forall\ l \geq 1$:

- The cutoff frequencies weights are fixed to -3dB
- The energy of the signal is conserved over all bands, as the sum of the square of filter weights along bands equals 1 for all frequency points.

For $l = 2$ or $l = 3$, the filters are shown to be compliant with both the ANSI S1.1-1986 and IEC 61260-1:2014 standards. The calculation of third-octave bands is then:

$$X_{L_{eq}} = aG^{\circ 2}|X|^{\circ 2} \tag{8}$$

where a is a normalization factor applied for energy conservation, X is given by equation 2 and $\circ$ denotes an element-wise operation. The log_{10} function can be applied to obtain third-octave bands in dB SPL.

Although this method allows for arbitrary filter design, it is limited by the framing effects inherent to the STFT, particularly visible in low frequencies. It is also apparently of lower computational complexity than time-filtering, and is thus preferred in the proposed coder scheme.

2.2 Data encoding

At this level in the coder process, the data is constituted of third-octave bands representations of the audio input signal. The data is accumulated over arbitrary

durations to form texture frames to be transmitted. Depending on the time integration of 125 ms or 1 s and with 31 bands within the range of [20 Hz, 20 kHz], the representation is a matrix of size $8T \times 31$ or $T \times 31$ respectively for T seconds texture frames. The data size can however be further reduced using source coding techniques.

2.2.1 Notions on source and entropy coding

Every signal follows a distribution which can be associated to a probability density function (PDF). Digital signals have limited possibility as to the different values they can take, depending on the type of data they are coded on. These values are referred to as symbols, and result in discrete probability distributions. The PDF then contain the precise probability of appearance of each symbol, and can be exploited to pair each symbol or group of symbols to a code of different size so that the output data size is minimized.

The Shannon entropy H defines the least possible average code size per input symbol given a signal's distribution. It is expressed as:

$$H = \sum_i p_i log_n(p_i), i = 0, .., i \tag{9}$$

where p_i is the probability of appearance of the i^{th} symbol and n is the output representational base, usually 2 for binary outputs.

The entropy is therefore closely related to the PDF and the underlying information content of the data. It is minimum at $H = 0$ when the signal is deterministic, *i.e.* when a single symbol with a probability of 1 appears. Conversely, it reaches a maximum for uniform distributions at $H = log_n(1/N)$ where N is the number of symbols. A low entropy is thus achieved for signals with few values appearing with high probabilities and a low range of possible symbols.

Entropy coding algorithms use symbol-probability pairs to approach entropy. Arithmetic [48] and Huffman [25] coding are two examples, and the latter is studied in this work.

2.2.2 Huffman coding

The Huffman algorithm is as follow. For $n = 2$, symbols are sorted by probability of appearance, then a binary Huffman tree is constructed using the following principle: at each step, the two lowest probabilities are drawn from the list and associated in the tree. The sum of the two probabilities is then added back to the list. When the root of the tree attains the cumulated probability of 1, the symbols

are assigned a variable-length code. Each layer away from the root adds a bit so that the highest probability symbol has a one-bit code while the rarest has the longest code. The resulting symbol-code pairs form a Huffman dictionnary which is then used to transcribe inputs. The size of separately encoded symbols using this method is optimal, that is the closest achievable to the data entropy. Huffman coding is a straightforward algorithm with few situational variants.

2.2.3 Proposed algorithm

To allow for minimal errors, the audio analysis step is computed using data types with sizes ranging from 16 to 64 bits per stored value. Directly applying a coding algorithm is thus inefficient due to large dictionnaries inducing high computational costs and output code dimensions. A quantization step is used to reduce the number of symbols to an acceptable amount of 2^q values for q-bit values.

However, an example of data distribution computed for environmental audio recordings is shown in Figure 4a. Third-octave measurements are concentrated in low values and linearly quantifying would result in most of the information being lost. The use of a logarithmic function solves this problem by "flattening" the PDF. Measurements are therefore expressed in dB, then quantized as visible in Figure 4b.

In order to enable better Huffman coding performances, the PDF must follow the principles discussed in section 2.2.1. As measurements are made continuously, third-octave bands values usually vary slowly. Especially when using overlap in the STFT, consecutive frames contain redundant information that can be eliminated. To this aim Δ-compression is applied along the time dimension. It consists in subtracting each frame to the previous one to encode only variations in the signal. This has the desired effect of concentrating the probability of appearance around zero as shown in Figure 4c. Δ-compression inherently doubles the maximum number of symbols: for a signal $x \in [0, N]$, the result is $x_\Delta \in [-N, N]$. This is easily outweighted by the effects on the PDF and the compression yields a significant efficiency increase of Huffman coding.

One last choice regarding data encoding concerns the construction of the Huffman dictionnary. One solution is to generate a tree and dictionnary for each texture frame, which is then optimal but must be transmitted along with the code. Alternatively, a global dictionnary can be computed from large databases of environmental sounds. In that case, the coding process does not take into account the exact data distribution and must pair every possible symbol with a code. The added data of the first method and the sub-optimality of the second are found to

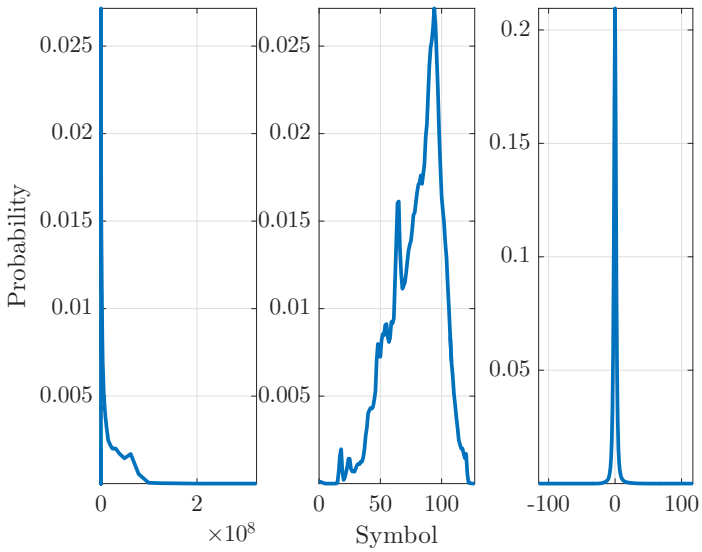


Figure 4: Example of estimated probability density functions of the data throughout the encoding step. Unchanged output of the representation step (left), concentrated towards very low values. PDF "flattening" effect induced by logarithm application (middle). Here values are mapped to the range $[0, 2^7 - 1]$ and rounded to perform quantization. Output of the Δ compression (right) -, with desirable probabilities as the input to a Huffman algorithm.

compensate each other as they yield equivalent performances in this application. The computational complexity is found to be lower with dynamic dictionnary generation for a Matlab implementation, due to the number of symbols being generally between 2 and 5 time less than in the full dictionnary.

2.3 Overview of the proposed scheme

The coder scheme is summarized in Figure 5.

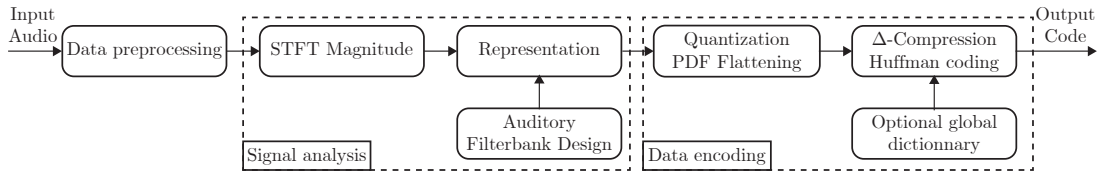


Figure 5: Overview of the coder processing scheme.

3 Experimental validation setup

3.1 Datasets

To evaluate the performances of the proposed coder, two datasets are used. The first consists of urban environmental audio recordings, and is used for measurement error, event recognition and bitrate concerns as its contents match the aim of the application. Furthermore, as unintelligibility is to be ensured in decoded and reconstructed audio extracts, a second dataset of clean speech recordings is studied.

3.1.1 UrbanSound8k

UrbanSound8k [44] is a dataset of urban audio recordings proposed by the Sounds Of New-York City (SONYC) project researchers. It is composed of 8732 audio extracts with varying durations of less than 4 s, amounting to about 9 hours total. Each excerpt corresponds to one class among ten: *street music*, *child playing*, *dog bark*, *air conditionning*, *drilling*, *jackhammer*, *car engine*, *siren*, *car horn*, *gunshot*. The proportions are quite balanced with 1000 extracts per class except for the three last with about 900, 400 and 400 files respectively. The audio is provided at a sampling rate of 44.1 kHz in wave format which allows a degree of freedom as to the quality of the input data for testing purposes.

The UrbanSound8k dataset is a subset of a larger one, UrbanSound, made with the aim of providing a small but reliable resource for environmental audio applications such as classification. For this reason the data is distributed over 10 independant folds containing multiple examples of each class of sound, enabling cross-validation schemes. Several classification schemes and results are proposed for this dataset that can be used as a baseline, with techniques ranging from support vector machines and decision trees [44] to deep learning [43] with unsupervised feature learning [42].

3.1.2 Speech recordings

For intelligibility concerns, a small dataset of 23 clean speech recordings is used. This is to ensure the validity of the proposed test's results in the final implementation: if clean speech is made unintelligible, speech occurences in an environmental context where noise levels are important will be even harder to comprehend.

The dataset contains recordings of 9 french sentences enunciated by 6 different speakers, 3 of which are male and 3 female. This represent a total of 216 extracts of about 3 seconds each. The recordings were made in studio conditions and provided in 44.1 kHz stereo wave files.

3.2 Measurement error

The measurement error of third-octave bands is one of the main concerns of the coder validation, as this indicator is subject to strict tolerances. In the studied application, the error is two-fold: third-octave bands computation from raw audio is biased by framing the signal and the data encoding chain adds quantization error to the data.

3.2.1 Analysis error

To study the analysis error of third-octave bands, they can be compared to a trusted reference implementation. To this aim, third-octave bands measured in both "fast" computation with 125 ms windows and "slow" 1 s analysis mode are averaged over time to compute the acoustic indicator on extracts of several seconds. The chosen reference considers the Matlab *ita_ toolbox* implementation [11]. In both cases the sound dataset is the same and is composed of 2 s extracts. The experiment is run for white noise as well as urban environmental recordings. As the analysis frames are of short duration, poor frequency resolution and spectral leakage are expected, particularly in the lower frequencies bands. For this reason we evaluate the effect of the use of windowing functions on each band measurement

error in order to mitigate this loss of precision. To provide relevant statistics, these quantities are estimated for about 4500 extracts obtained from the UrbanSound8k dataset [44] and 4500 *iid.* noise sequences.

3.2.2 Additional encoding error

The data encoding process is composed of lossless operations with the exception of quantization. The representation before this step is given in dB. Let us define ε as the absolute error between the data x and its quantized equivalent x_q, such as

$$\varepsilon = |x_q - x| \tag{10}$$

The entire encoding process is defined by the desired Huffman dictionnary size N_H which is linked to a word size q with the relation

$$q = log_2(N_H) \tag{11}$$

q is obtained at the output of Δ-compression. The values after quantization are thus coded on $q-1$ bits. x_q is then given by:

$$x_q(n) = \frac{\Delta_x}{2^{q-1}-1}\text{round}\left(\frac{(2^{q-1}-1)x(n)}{\Delta_x}\right), x \in [0, \Delta_x] \tag{12}$$

where Δ_x is the range of values taken by x. The error ε can then be theoretically estimated by modelling the PDF of x with a uniform distribution such as $x \sim U\{0, \Delta_x\}$. While in reality x will never be uniform, this approximation matches the aim of a decibel representation as discussed in section 2.2.3. ε then also follows a uniform distribution $\varepsilon \sim U\{0, \frac{\Delta_x}{2\times(2^{q-1}-1)}\}$. Its mean μ_ε and standard deviation σ_ε are:

$$\begin{cases} \mu_\epsilon = \frac{\Delta_x}{4\times(2^{q-1}-1)} \\ \sigma_\epsilon = \frac{1}{12}\frac{\Delta_x}{2\times(2^{q-1}-1)} \end{cases} \tag{13}$$

In practice, the error is therefore assumed to be $\varepsilon = f(\Delta_x, \frac{1}{2^{q-1}-1})$ with the heterogeneity of the data's PDF inducing small variations to the equations in (12). Most mathematical operations such as base 10 logarithm and matrix multiplication can induce small additional rounding errors, especially on embedded implementations. Furthermore, the error should be the same across all third-octave bands as quantization operation is applied regardless of data dimensions.

The experimental evaluation consists in computing the absolute error for a set of word sizes q. The UrbanSound8k dataset is used to provide meaningful results both in audio data nature and number of examples.

3.3 Output bitrate

Another metric of the efficiency of the proposed scheme is the measurement of the output data bitrate. Similarly to classification, the parameters influencing the bitrate are the representation's time resolution and encoded word size q.

3.4 Event recognition

Descriptors based on third-octave bands analysis are relatively untested on classification tasks. In order to ensure that the proposed scheme allows the recognition of events using state of the art methods, it can be compared to the very similar and commonly used features derivated from mel bands analysis. Baseline performances [44] are available for the UrbanSound8k dataset using Mel-Frequency Cepstrum Coefficients (MFCC), which are obtained by applying a Discrete Cosine Transform (DCT) to mel bands. The same operation is performed on third-octave bands to match the physical meaning of the MFCC descriptor.

The event recognition task performance is evaluated on four different models to ensure the validity of the results:

- a Support Vector Machine (SVM), which isolates data distributions of different classes by finding the maximum-margin separation between them. In this application, the C-SVM variant is used as a nonlinear classifier with a radial-basis function (RBF) kernel of variance σ^2. C and σ^2 are found via grid search.
- a Decision Tree (DT), that learns to take consecutive decisions based on the input features.
- a Random Forest (RF) classifier, which is an ensemble method based on the decision tree. It uses the bagging concept of training several models and classifying by majority voting. The number of trees is here set to 500.
- a k-Nearest Neighbors (KNN) classifier, which simply outputs the class of a test sample as that of the k closest training examples. If N features characterize each data point, a N-dimensional mathematical distance is computed. In this evaluation the metric is the Euclidean distance and $k = 5$.

The features are computed from the 25 first mel or third-octave cepstrum coefficients, which as summarized along time with the mean, variance, skewness, kurtosis, minimum, maximum, median, derivative mean and variance, second order derivative mean and variance operators. The feature vector are thus comprised

of 275 values. Each classifier is trained using 10-fold cross-validation which consists in training the models with 9 of the 10 folds and testing with the last, that for each of the 10 possible combinations.

For this process, the studied coder factors are the quantization word size q and analysis frame duration. Mel spectrogram analysis is computed with the baseline 23 ms, 50% overlap windows, then averaged over time to match the 125 ms or 1 s integration time used for third-octave bands measurements.

3.5 Intelligibility assessment

Encoding a raw audio recording should render it unintelligible at reconstruction. To guarantee this property the clean speech dataset presented in section 3.1.2 is passed through the encoding process, then decoded and recovered using methods detailed in Appendix A. Intelligibility is then assessed using both objective and subjective indicators. Results will be shown for varying time resolutions during analysis only as the quantization was found to have very little impact on the perceived intelligibility.

3.5.1 Objective indicators

Intelligibility is known as an important factor in psychoacoustics and noisy environment studies. As a results, many objective indicators have been developed to model intelligibility based on either raw audio or spectral features. A review of these metrics is available in [33]. Various concepts and physical properties are exploited, although almost every method compares the noisy signal to a clean version. While several of the indicators are proved to correlate well with subjective estimations of intelligibility, results are only available for small degradations such as clipping and addition of white or colored noise. The studied signals experience much harsher conditions with band analysis and discarded phase spectrum. The accuracy of objective metrics is therefore not guaranteed and need to be compared to subjective evaluations.

Two indicators with good apparent results are computed: the Coherence-Speech Intelligibility Index (CSII) [27] and the Frequency-Weighted Segmental SNR (fwSNRseg) [24].

3.5.2 Perceptual test

A perceptual intelligibility test is also conducted as both a reference for comparison with the objective indicators and an accurate evaluation to validate that the

coder ensures privacy for the studied parameters.

The test is realised under the following conditions: a Matlab interface is displayed on a desktop computer. Each extract is played through *Beyerdynamics DT 770* headphones in a random order. The output level is the same for all subjects. For each example, the participant is asked to type the words he understands and to rate the global intelligibility between 1 and 5. The Intelligibility Ratio (IR) *i.e.* the percentage of correctly transcribed words constitutes the first subjective metric while the Average Intelligibility Score (AIS), that is the note scaled to 0 and 1, is a second indicator. 12 subjects of age ranging from 17 to 60 which reported normal hearing participated to the listening test.

4 Results

4.1 Measurement error

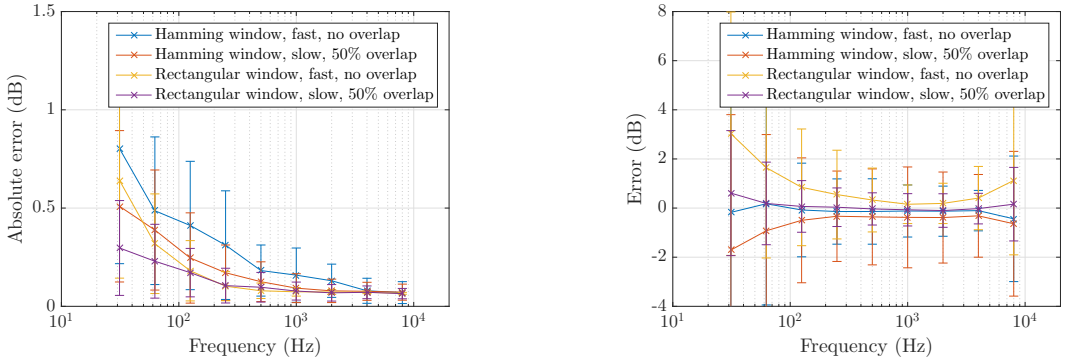
Analysis error

The measurement error is computed for third-octave bands. Results are displayed in Figure 6. Only full-octave bands are shown for visibility. When computing the slow analysis, it is found that applying overlap attenuates the error. This is not observed for fast analysis where the error is similar or higher with the use of overlapping frames. As all experiments on noise show a mean error close to zero, the absolute error metric is preferred.

As can be expected when performing a short time analysis, high errors appear in low frequency bands where precision is poor. The slow analysis yields better estimations as it provides a globally better resolution and thus also reduces the impact of spectral leakage effects in this region. When considering white noise (left) the error is low on the whole spectrum. This effectiveness does not translate to environmental sounds analysis. A factor of this phenomenon is the sound level disparities absent in the first case but omnipresent in the second. In fact, spectral leakage induces a correlation between close frequency components as briefly explained in section 2.1.1. In lower frequencies, large differences between adjacent frequency bins can cause important errors on third-octave bands computations. This issue is however not specific to Fourier transform based schemes as it is also encountered with a time domain filtering approach.

To mitigate this phenomenon, the choice of the window function shall be studied further. For a fast analysis with no overlap, the rectangular window seems a

reasonable choice at first with its energy conservation qualities. While it yields the lowest error in high frequency bands, its high spectral leakage effect makes it unprecise for lower frequency regions. Non-flat windows better account for this issue but require assuming that the signal is stationary in a given frame and are thus biased. For slow measurements, the rectangular window impact is less harmful to band analysis due to an increased frequency resolution.



(a) Noise extracts

(b) Urban sounds recordings

Figure 6: Measurement error of third-octave bands over two-seconds recordings. The analysis of short frames has an effect on energy estimation at low frequencies. Framing is more impactful on signals where the energy is localized in frequency.

To demonstrate that the FFT-based method is not solely responsible for these errors, third-octave bands are computed via the *ita_toolbox* reference for the same parameters (fast and slow, windowing, overlap). It is found that for all parameters and all bands, except for the five lower frequency bands in fast analysis, the resulting errors are not statistically different to those obtained with the proposed method. For the remaining bands, the toolbox exhibit slightly higher or lower errors.

Encoding error

The additional error caused by the encoding steps after obtaining the desired data representation, with the word size q varying. Figure 7 shows an estimation of the error $\hat{\varepsilon}$ as a function of q for third-octave and Mel bands. In both cases, the mean and standard deviation seem to decrease by a factor of 2 as q increases. This is in accordance with the approximate theoretical expressions obtained in equation 13. This error is to be added to the measurement error discussed in the precedent section.

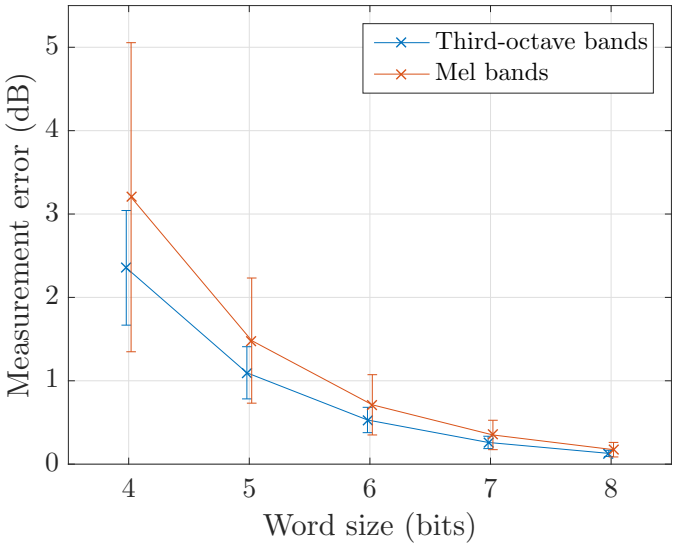


Figure 7: Measurement error induced by encoding for different quantization resolutions.

4.2 Output bitrate

The main indicator of performance is the bitrate obtained at the output of the coder. The three varying factors studied are the data word size q, the number of bands and the effect of reducing time-resolution by averaging analysis frames. Figure 8 shows estimations of the output bitrate for different values of q. To match third-octave bands computation principle where a 125 ms analysis is mandatory, Mel frames were averaged over time. Because we use the most common parameters, namely 23.2 ms window with 50% overlap, the closest achievable rate considering simple averaging is 7.74 frames per second. Third-octave representation on 31 bands yielded an overall higher size than their Mel equivalent, here estimated for 30 bands. It however compares with the 40-Mel bands representation which is the most used features in literature. This higher bitrate is likely due to the distribution observed by the data prior to Huffman encoding.

A second set of parameters influences directly the time-frequency resolution of the analysis. By choosing a frame rate and number of bands, one can effectively control the size of periodically transmitted data. The bitrate is evaluated for 10 to 40 Mel bands and a frame rate from 2 to 10 per second, with fixed $q = 8$. Results are shown in Figure 9. As expected, the bitrate for a given word size q can be modeled as a linear function of the representation dimensions for one second of analysis. Small variations are induced by data distributions on a per-frame level

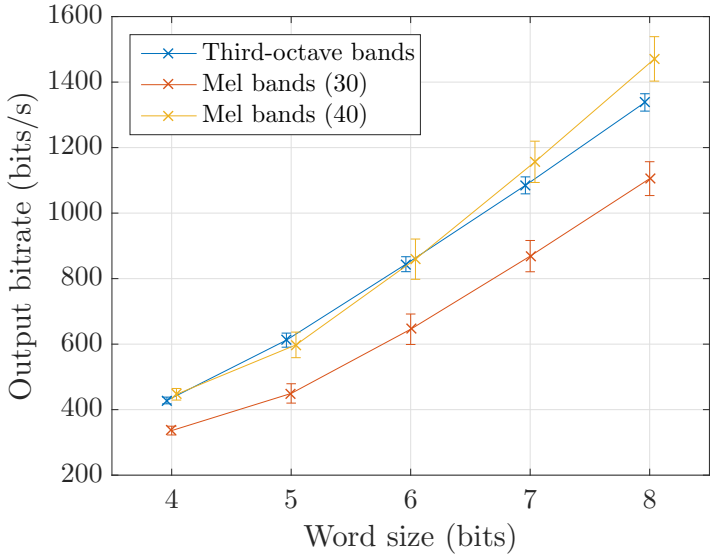


Figure 8: Coder output bitrate as a function of quantization for third-octave and Mel bands with 8 frames per second.

and their impact on the Huffman algorithm.

4.3 Event recognition

The same set of parameters as in bitrate evaluation is used to study the impact of their tuning to the sound event recognition performance. The experiment is aimed at finding ways to further reduce the encoded data size without strongly affecting recognition performance. Results of the four presented classification methods are provided for the sake of completeness.

First, the impact of quantization on classification accuracy can be observed. The models are trained on the most complete implemented representation, ie. 40 Mel bands, 23.2 ms, 50% overlap and no time averaging (85 frames per second). Figure 10 shows that this process yields equivalent accuracy for higher resolutions.

The effect of changing the time-frequency resolution of the representation is presented in Table 1. Similarly to baseline results, the random forest and SVM classifiers are the higher performing systems at 0.69 ± 0.06 and 0.68 ± 0.04 respectively. However, it is found that given our setup, reducing the number of analysis bands up to a given level most often does not induce a strong drop of performance.

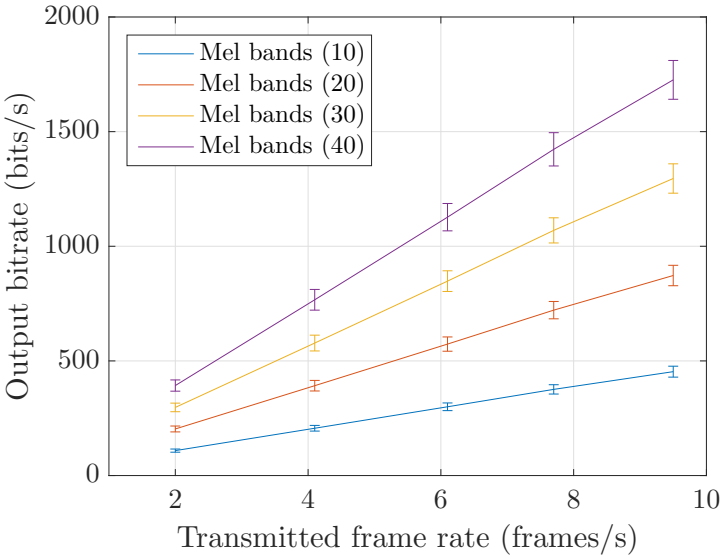


Figure 9: Impact of representation resolution on the encoded data bitrate.

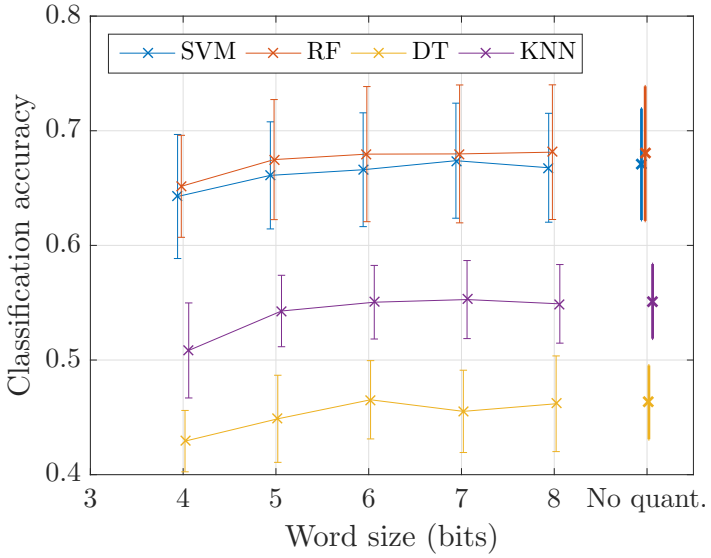


Figure 10: Classification accuracy as a function of word size before encoding. The baseline on the right is computed without quantizing the representation.

The performance of the decision tree classifier is a good example of this behavior, with best performances for 10 Mel bands only and a 4% lower accuracy for 40. The latter result may be due to the low ability of the decision tree classifier to handle hugh dimensional features. Even if the difference is small, all other models perform best with 30 or 40 Mel bands as a representation. Time decimation cn also be considered, as the original representation can be averaged and consistently yield a good classification accuracy. The loss of information can be considered negligible for FPS values as low as 20 or 10 depending on the method. Even below, classification accuracy only drops by one or a few percents. This means that the data size can be effectively divided by a factor of at most 10 without strongly affecting sound event recognition performance. It also provides us with a preliminary confirmation of the possibility for "fast"-sampled third-octave bands cepstra to match MFCC performances.

The next test compares the efficiency of third-octave bands at characterizing urban soundscapes to that of Mel spectrograms. Following the previous discussions, the classification task is run on corresponding cepstra with a fixed 8 frames per second and 31 bands. Figure 11 displays similar results to the 40 bands, 8 fps Mel spectrograms seen in Table 1 for all classification schemes. In this table, equivalence of performance with respect to the best performing setting (in red) is depicted with a bold font evaluated using the following procedure. The null hypothesis that the subtraction of the two compared distribution (the distribution of the consider setting minus the distribution of the best performing one) comes from a normal distribution with mean equal to zero and unknown variance is evaluated using the paired-sample t-test at the 0.05 significance level. If the null hypothesis is not rejected, the setting is considered as equivalent in terms of performance to the best performing one.

To further analyze both representations advantages, the confusion matrix is a reliable tool. It aims at providing information regarding class-by-class accuracy and misclassification rates. Confusion matrices are thus computed for the SVM classifier, with predictions accumulated over the ten folds in test configuration. The analogous natures of the two descriptors is highlighted by close one-versus-one differentiation performances, with a slightly lower accuracy for third-octave bands on average. Both representations yield best results for the *Gun shot* class with 88.2% for third-octave cepstra and 86.6% for Mel cepstra. Their poorest accuracy is on the *Air conditioning* class with 32.0% and 38.1% respectively. However, a noticeable difference between them is that on the log-frequency scale the bandwidth of Mel filters narrows as frequency increases, while third-octave are evenly distributed. The effect of this can be seen between classes *Drilling* and *Jackhammer*.

Table 1: Classification accuracy in percentage for different classifiers: SVMs (a), Random Forests (b), Decision Trees (c), and Nearest Neigbors (d) with respect to varying representation resolutions. Numbers in red indicate best performance and numbers in bold indicate statistical equivalent results compared to the best performing setting.

(a)

SVM		Frames per second						
		2	4 (4.1)	6 (6.1)	8 (7.7)	10 (9.5)	20 (21)	85
Mel bands	10	55±3	60±3	61±4	62±3	62±4	**63±6**	**65±6**
	20	58±4	62±4	63±4	64±4	63±4	**65±5**	**67±6**
	30	60±3	64±4	64±4	**65±4**	**65±3**	**67±4**	**68±4**
	40	60±3	63±4	**64±4**	64±4	**64±4**	**66±4**	**68±5**

(b)

RF-500		Frames per second						
		2	4 (4.1)	6 (6.1)	8 (7.7)	10 (9.5)	20 (21)	85
Mel bands	10	60±3	62±3	62±3	63±3	63±3	**65±4**	**67±5**
	20	61±4	63±3	64±3	64±3	64±4	**66±6**	**69±6**
	30	62±3	63±3	64±3	64±3	64±4	**67±5**	**69±6**
	40	62±4	63±4	63±4	64±4	63±4	**67±6**	**68±6**

(c)

DT		Frames per second						
		2	4 (4.1)	6 (6.1)	8 (7.7)	10 (9.5)	20 (21)	85
Mel bands	10	42±3	**46±4**	46±2	44±3	45±3	**46±5**	**49±3**
	20	43±5	43±3	43±2	44±3	45±3	45±5	**47±5**
	30	42±4	43±2	44±3	45±5	43±3	43±4	**45±5**
	40	42±6	43±3	43±3	42±3	44±3	46±3	**46±4**

(d)

KNN-5		Frames per second						
		2	4 (4.1)	6 (6.1)	8 (7.7)	10 (9.5)	20 (21)	85
Mel bands	10	43±2	51±4	53±4	53±5	53±4	54±4	**56±3**
	20	44±3	52±4	53±3	54±4	**54±4**	**55±4**	**58±4**
	30	45±4	54±5	**55±5**	**55±4**	**55±4**	**56±4**	**56±4**
	40	46±3	53±5	**55±5**	**55±4**	**55±5**	**57±4**	**57±3**

These sounds involve important low-frequency information which can generally differentiate them, leading third-octave descriptors to perform better. Conversely, using Mel-based cepstra improves globally *Air conditioning* recognition as most of its defining components are situated in higher frequencies.

Both descriptor are thus found to have similar representational capabilities

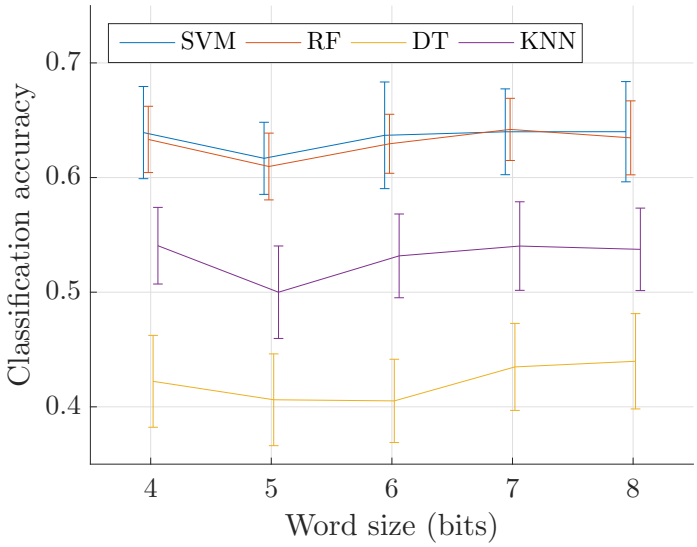


Figure 11: Classification accuracy with third-octave bands and varying word size.

despite minor differences due to their respective designs.

4.4 Intelligibility

Objective intelligibility metrics are then computed for the encoder's settings under evaluation and compared to the results of the perceptual test. The results of the perceptual test is evaluated using the average intelligibility score (AIS) given by the subjects and the intelligibility ratio (IR) computed as the number of correctly transcribed words versus the number of words in the spoken utterance. Figure 12 presents the results of the perceptual test for the same encoder's settings, Figure 13 and Figure 14 show the CSII and fwSNRseg estimations respectively.

Perceptual test results confirm the qualitative evaluation of speech properties discussed in Section 1.4. Phonemes separation is indeed mandatory in order to ensure good intelligibility of speech. This implies an analysis frame rate significantly larger than the phonemes rate the considered language. A limit of about 10 Hz corresponding to a frame duration of 100 ms is observed, as the studied 8 Hz setting is found to be almost completely unintelligible. Very few words are understood correctly. It can be assumed that correctly identified words are a result of coincidentally adequate frame timings, *i.e.* analysis frames matching the location of phoneme utterances for a short duration. Similarly, framing effects induced by the proposed processing scheme yielded errors for higher frame rates transcriptions and a globally lower perceptual intelligibility.

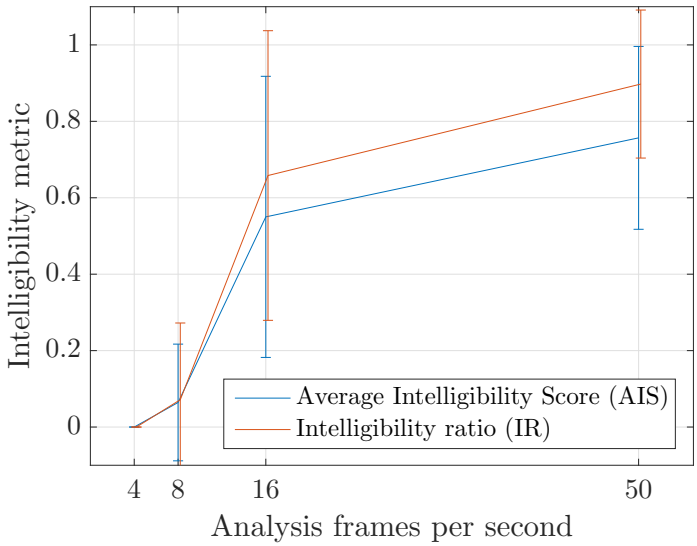


Figure 12: Average intellibility score (AIS) given by the subjects and the inteligibility ratio (IR) as a function of the frame rate. Unintelligibility appears at around 10 Hz, corresponding to the average duration of a phoneme.

The CSII accounts for the distance between clean and distorted audio spectrograms at each point in time and frequency. This includes the phase, which we discard early in the representation step. The error induced by forward and inverse third-octave bands transformation as well as operations reducing data bitrate further contribute to very high element-wise disparities. In fact, the signal is too distorted for the proposed SNR normalization to ensure CSII values in the 0-1 range. When compared to perceptual test results, the CSII shows no clear separation in the 10 frames per second neighbourhood. It is therefore not a good indicator in the studied conditions.

Alternatively, the frequency-weighted segmental SNR considers only spectrogram magnitudes. As a result, distance is generally lower and more representative of the induced perceptual distortion. Nevertheless, the fwSNRseg still uses a pointwise comparison which is certainly too precise for the studied degradations. Its shape seems more accurate at higher frame rates where intelligibility is almost constant. The previously observed threshold is however absent as the curve present a smoother increase trend, and the difference between 8 Hz and 16 Hz is not significant enough to correlate with subjective measurements.

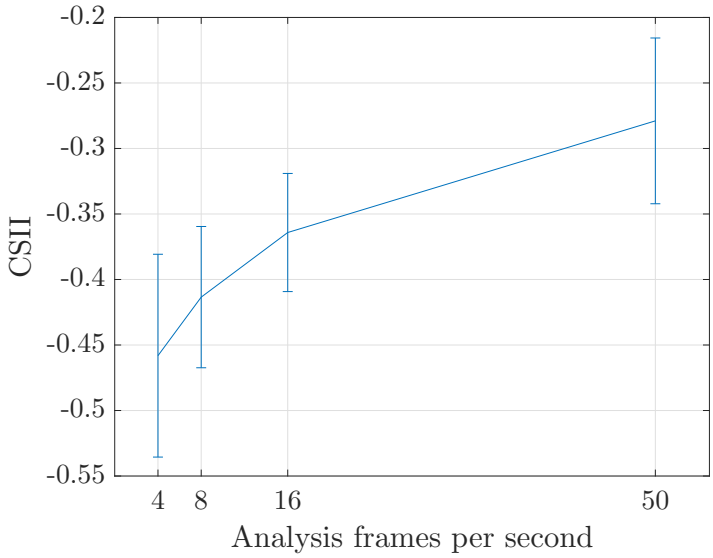


Figure 13: The Coherence SII objective indicator computed for intelligibility assessment. Negative CSII values indicate a signal-to-noise ratio too low for the recommended normalization, thus severe disparities between the original and reconstructed signal.

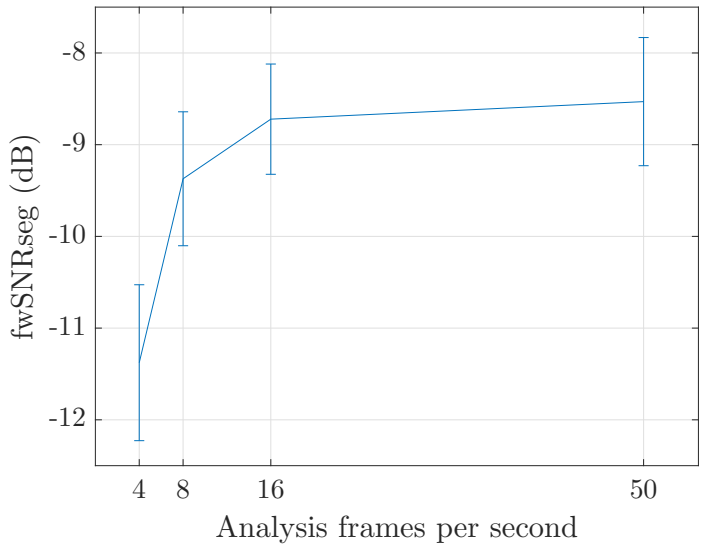


Figure 14: Frequency-weighted segmental SNR with varying analysis frame rates around the considered 8 Hz value. This indicator shows a better correlation with subjective tests but lacks an inintelligibility threshold.

Part III
Personal report

This part provides personal views and perspectives regarding this internship, both retrospectively and about its implication on my future development. To this aim, the recurring technical and human challenges, opportunities and discoveries will be detailed.

The main technical issue encountered still follows my work. I had indeed poor initial knowledge on acoustics and only basic audio signal processing. Either to correctly design the system or to interpret results, I often had to read entire book chapters about new notions. Fortunately, my referent professor and other people I interacted with were always willing to help by providing me with either written material, oral explanations or even a course. I continue to view this concern as an opportunity to enrich the yet small extent of my knowledge.

A second major technical problem was the computation of third-octave bands measurement error. The issue had multiple factors. First, I did not have access to the full text of standards about this indicator and thus only had approximate tolerances to work with. Second, there was a lack of reference implementation to compare with. The ita_toolbox supposedly was one, but that was never really demonstrated in the studied conditions of short-time analysis. Third, there was also a lack of clear experimental setup. The error was computed as an absolute difference, which is only sensible if the unit of measurements is disregarded. Uncertainties on the nature of the quantities to compare were also regularly discussed. A significant amount of time was spent in trying to solve this problem with acousticians at IFSTTAR, that while important was subsidiary to the internship's objective.

Organization-wise, I often misjudged the duration of certain tasks leading to previsional plannings not being fully met. Similarly, some of my priorities were suboptimal and I had difficulties stopping on more delicate, less secondary problems. Every mistake is however an opportunity to learn and I retrospectively do not view these issues as failures.

Working alone on a daily basis, though evidently expected in an academic environment, was arduous at times. However, this issue did not have as important an impact as I first imagined. In fact, apart from a few moments towards three months after the beginning of the internship, I quickly started adapting to this

concept and it never really slowed my progress. In this respect, this six month long working experience was formative. It introduced me to a different approach of production as opposed to the teamwork-oriented approach I was familiar with, effectively adjusting my agenda and working methods to those that can be expected for a PhD candidate.

Overall this masters internship was an accurate preview of the academic and research world in which I intend to develop. I was familiarized with the specific technical knowledge on experimental rules, results interpretation, paper writing and bibliographical search among others. New work organization elements were also addressed, as well as project management tools and practices. Lastly, I was introduced to a passionate and open community which corresponds in every respect to my expectations.

Conclusion

This report presented the work accomplished during this six-months masters internship. An audio coder scheme has been developed which allows the computation of acoustic indicators and sound event recognition while degrading the intelligibility of speech. It also fullfills the requested constraints in terms of bitrate and computational complexity.

This internship was rich in multiple aspects. My working activity involved diverse domains of technical skills, previously unknown for some, and tested a significant part of my training. The CENSE project is the first truly large project that I am a part of, with more than ten member entities with strong expertise in their own domain working together towards solving a single problem. My exchanges with researchers and engineers in other areas allowed me to understand the views, aims and concerns of each and provided me with an accurate insight about large research projects.

I also acquired a global vision on the research and academic environment. This includes time and resources management, project development, technical practices and several other points ranging from information on scientific journals organization to complete experiment management methods.

The internship comforted me in my decision to pursue a PhD thesis. I look forward to the three upcoming years during which part of my work will benefit the CENSE project. This will also allow me to follow the implications of the scheme I helped develop for its embedded implementation and operation. A journal paper will be submitted to a special issue on smart cities of the MDPI Sensors journal (impact factor 2.6) in the next few months.

A Signal reconstruction techniques

While reversing the data encoding step is straightforward, reconstructing audio from a frequency band representation is not trivial. The forward transform discards part of the information that the original recording contained: first when omitting spectrogram phase data, then when applying the linear transform that defines band analysis.

A.1 Spectrogram computation from band representations

The operation applied to obtain a band representation T from a linearly-scaled spectrogram X of size $m \times n$ can in the general case be written as:

$$T = MX \tag{14}$$

with M the transform matrix of size $k \times m, k < m$. The operation is thus essentially an averaging, and by definition there exist no single solution N such as $NT = X$, that is $NM = I_m$. The matrix used to recover the linearly-scaled spectrogram must therefore be estimated. This is possible by scaling the transpose of M:

$$\hat{N} = \frac{1}{a} M^T \tag{15}$$

where a is a normalization scalar usually expressed as $a = c \cdot diag(M^T M)$ with c a numerical constant.

This estimation does not yield a perfect reconstruction of the spectrogram, as can be seen in Figure 15. The resulting effect can be interpreted as a heterogeneous loss of resolution that increases with frequency bands. Each frequency bin of the linear spectrogram is averaged with other bins that belong to the same band.

A.2 Phase recovery

A large number of methods exist to reconstruct the phase from spectrogram magnitude. An extensive review of such methods is available in [45]. In the scope of this application, the Griffin and Lim algorithm [23] is used. It consists in computing successive estimations $\tilde{x}$ of the signal x by repeating the process:

$$\tilde{X}_i(k, f) = STFT(\tilde{x}_i[n]) \tag{16}$$

$$\tilde{X}_{i+1}(k, f) = |X(k, f)| \frac{\tilde{X}_i(k, f)}{|\tilde{X}_i(k, f)|} \tag{17}$$

$$\tilde{x}_{i+1}[n] = STFT^{-1}(\tilde{X}_{i+1}(k, f)) \tag{18}$$

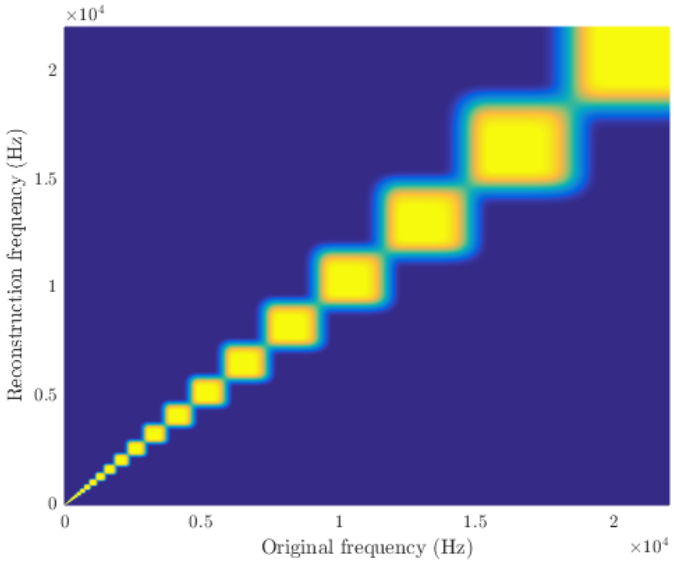


Figure 15: Third-octave bands analysis and approximate inverse transformation effects on energy location. This process yields an important and heterogeneous loss in resolution, particularly at higher frequency points.

for a fixed number of iterations i or until convergence. The estimate $\tilde{x}_0$ is usually initialized with random numbers or zeros.

Félix **Gontier**

Étudiant en 5ème année ingénieur

7, Avenue Jeanne d'Arc
49100 Angers, FRANCE
☎ +33 6 42 08 22 81
✉ felix.gontier@reseau.eseo.fr

Objectif : Préparer une thèse de doctorat en machine-learning et analyse de données dans le domaine d'applications de l'acoustique et de l'audio.

Formation

Université de Rennes 1
2016-2017 - Master SISEA

Université de Sherbrooke
2016 - Ecole d'été
Représentation Énergétique Macroscopique (RÉM)

Université de Glyndwr
2013 - Ecole d'été
Informatique et robotique

ESEO Shanghaï
2014 - Ecole d'été
Culture, économie et langue chinoises

Baccalauréat général
2012 - Série scientifique

ESEO Angers
2012-2017 - Formation d'ingénieur
2016-2017 - Option SIAT
(Signal, Image, Audio, Télécoms)

Expérience Professionnelle

Airbus Group
2016-2017 - Projet industriel ESEO
Reconnaissance de signaux sonores par réseaux de neurones

LS2N - Nantes
2017 - Stage - 6 mois
Projet CENSE (Poursuivi en thèse)
Codage - Intelligibilité - Classification

Université de Sherbrooke
2016 - Stage - 3 mois
Système d'émulation d'une chaîne de traction de véhicule électrique

Outils de travail

Langages informatiques

C/C++/Cuda-C
Java/Android API
PHP/MySQL
HTML/CSS
VHDL

Logiciels

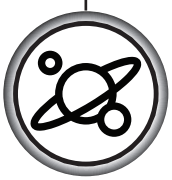
Matlab/Simulink
LabView (NI CLAD)
Altium
Latex/Outils Adobe CC

Langues

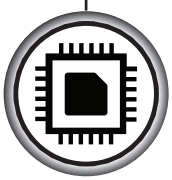
Français - Natif (Voltaire 824)
Anglais - Courant (TOEIC 990)
Mandarin - Débutant
Espagnol - Scolaire

Centres d'intérêt

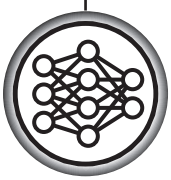
Musique, guitare basse



Astronomie



Électronique numérique



Machine-Learning

Planning

Task	March	April	May	June	July	August
Bibliography - General						
Coder design - Mel bands						
Bibliography - Classification						
Bibliography - Intelligibility						
Presentations						
expLanes Implementation						
Tests and results - Mel bands						
Bibliography - Ac. Indicators						
Coder design - Third-octave						
Tests and results - Third-octave						
Paper writing						
ESEO internship report						

References

[1] ANSI S1.1-1986, (ASA 65-1986)—Specifications for Octave-Band and Fractional-Octave-Band Analog and Digital Filters, 1993.

[2] IEC 61672-1:2013 —Electroacoustics - Sound level meters - Part 1: Specifications, 2013.

[3] IEC 61260-1:2014 —Electroacoustics - Octave-band and fractional-octave-band filters - Part 1: Specifications, 2014.

[4] J. Antoni. Orthogonal-like fractional-octave-band filters. *J. Ac. Soc. Am.*, 127(2):884895, 2010.

[5] M. Anusuya and S. Katty. Speech recognition by machine, a review. *International Journal of Computer Science and Information Security*, 6(3):181–205, 2009.

[6] J. Aucouturier, B. Defreville, and F. Pachet. The bag-of-frames approach to audio pattern recognition: A sufficient model for urban soundscapes but not for polyphonic music. *J. Ac. Soc. Am.*, 122(2):881–891, 2007.

[7] P. Aumond, A. Can, B. De Coensel, D. Botteldooren, C. Ribeiro, and C. Lavandier. Modeling soundscape pleasantness using perceptual assessments and acoustic measurements along paths in urban context. 103:430–443, 2017.

[8] John R Baldwin and Peter French. *Forensic phonetics*. Pinter Publishers, 1990.

[9] C. Baug, M. Lagrange, J. Andén, and S. Mallat. Representing environmental sounds using the separable scattering transform. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013.

[10] P. Bentley and J. McDonnell. Wavelet transforms: an introduction. *Electronics Communication Engineering Journal*, 6(4):175–186, Aug 1994.

[11] Marco Berzborn, Ramona Bomhardt, Johannes Klein, Jan-Gerrit Richter, and Michael Vorlnder. The ITA-Toolbox: An Open Source MATLAB Toolbox for Acoustic Measurements and Signal Processing. 43th Annual German Congress on Acoustics, Kiel (Germany), 6 Mar 2017 - 9 Mar 2017, Mar 2017.

[12] Louis-Jean Boë. Forensic voice identification in france. *Speech Communication*, 31(2):205–224, 2000.

[13] L. Brocolini, C. Lavandier, M. Quoy, and C. Ribeiro. Measurements of acoustic environments for urban soundscapes: Choice of homogeneous periods, optimization of durations, and selection of indicators. *J. Ac. Soc. Am.*, 134(1):813–821, 2013.

[14] R. Cai, L. Lu, A. Hanjalic, H. Zhang, and L. Cai. A flexible framewok for key audio effects detection and auditory context inference. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(3):1026–1039, 2006.

[15] A. Can, P. Aumond, S. Michel, B. De Coensel, C. Ribeiro, D. Botteldooren, and C. Lavandier. Comparison of noise indicators in an urban context. In *45th International Congress and Exposition on Noise Control Engineering*, pages 5678–5686, 2016.

[16] A. Can and B. Gauvreau. Describing and classifying urban sound environments with a relevant set of physical indicators. *J. Ac. Soc. Am.*, 137(1):208–218, 2015.

[17] S. Chachada and C. Kuo. Environmental sound recognition: A survey. In *2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, 2013.

[18] S. Chu, S. Narayanan, and C. Kuo. Environmental sound recognition with time-frequency audio features. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(6):1142–1158, 2009.

[19] European Commission. Directive 2002/49/ec. http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32002L0049, 2002.

[20] L. Couvreur and M. Laniray. Automatic noise recognition in urban environments based on artificial neural networks and hidden markov models. In *The 33rd International Congress and Exposition on Noise Control Engineering*, 2004.

[21] S. Davis. Octave and fractional-octave band digital filtering based on the proposed ansi standard. In *1986 IEEE International Conference on Acoustics, Speech and Signal Processing*, 1986.

[22] P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio, and M. Vento. Reliable detection of audio events in highly noisy environments. *Pattern Recognition Letters*, 65:22–28, 2015.

[23] D. Griffin and J. Lim. Signal estimation from modified short-time fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(2):236–243, 1984.

[24] Y. Hu and P. Loizou. Evaluation of objective quality measures for speech enhancement. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(1):229238, 2008.

[25] D. Huffman. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, 40(9):1098–1101, 1952.

[26] T. Ishiyama and T. Hashimoto. The impact of sound quality on annoyance caused by road traffc noise: an influence of frequency spectra on annoyance. *JSAE Review*, 21(2):225–230, 2000.

[27] J. Kates and K. Arehart. Coherence and the speech intelligibility index. *J. Ac. Soc. Am.*, 115(5):22242237, 2005.

[28] Raymond D Kent, Charles Read, and Ray D Kent. *The acoustic analysis of speech*, volume 58. Singular Publishing Group San Diego, 1992.

[29] P. Khunarsal, C. Lursinsap, and T. Raicharoen. Very short time environmental sound classification based on spectrogram pattern matching. *Information Sciences*, 243:57–74, 2013.

[30] A. Kumar and B. Raj. Features and kernels for audio event recognition. `https://arxiv.org/abs/1607.05765`, 2016.

[31] Hisao Kuwabara. Acoustic properties of phonemes in continuous speech for different speaking rate. In *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, volume 4, pages 2435–2438. IEEE, 1996.

[32] M. Lagrange. Explanes - beautiful computational experiments. `http://mathieulagrange.github.io/expLanes/`.

[33] J. Ma, Y. Hu, and P. Loizou. Objective measures for predicting speech intelligibility in noisy conditions based on new band-importance functions. *J. Ac. Soc. Am.*, 125(5):33873405, 2009.

[34] M. Nilsson, D. Botteldooren, and B. De Coensel. Acoustic indicators of soundscape quality and noise annoyance in outdoor urban areas. In *19th International Congress on Acoustics*, 2007.

[35] S. Ntalampiras. Universal background modeling for acoustic surveillance of urban traffic. *Digital Signal Processing*, 31:69–78, 2014.

[36] K. Piczak. Environmental sound classification with convolutional neural networks. In *IEEE 25th International Workshop on Machine Learning for Signal Processing*, 2015.

[37] R. Radhakrishnan, A. Divakaran, and P. Smaragdis. Audio analysis for surveillance applications. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2005.

[38] G. Rey Gozalo, J. Trujillo Carmona, J.M. Barrigon Morillas, and V Gomez Escobar. Relationship between objective acoustic indices and subjective assessments for the quality of soundscapes. *Applied Acoustics*, 97:1–10, 2015.

[39] Stuart Rosen. Temporal information in speech: acoustic, auditory and linguistic aspects. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 336(1278):367–373, 1992.

[40] M. Rychtarikova and G. Vermeir. Soundscape categorization on the basis of objective acoustical parameters. *Applied Acoustics*, 74(2):240–247, 2013.

[41] J. Salamon and J. Bello. Feature learning with deep scattering for urban sound analysis. In *23rd European Signal Processing Conference*, 2015.

[42] J. Salamon and J. Bello. Unsupervised feature learning for urman sound classification. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015.

[43] J. Salamon and J. Bello. Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Processing Letters*, 24(3):279–283, 2017.

[44] J. Salamon, C. Jacoby, and J. Bello. A dataset and taxonomy for urban sound research. In *22nd ACM international conference on Multimedia*, 2014.

[45] N. Sturmel and L. Daudet. Signal reconstruction from stft magnitude: A state of the art. In *14th International Conference on Digital Audio Effects (DAFx-11)*, 2011.

[46] A. Torija, D. Ruiz, and A. Ramos-Ridao. Application of a methodology for categorizing and differentiating urban soundscapes using acoustical descriptors and semantic-differential attributes. *J. Ac. Soc. Am.*, 134(1):791–802, 2013.

[47] G. Tzanztakis, G. Essl, and P. Cook. Automatic musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, 2002.

[48] I. Witten, R. Neal, and J. Cleary. Arithmetic coding for data compression. *Commun. ACM*, 30(6):520–540, June 1987.