universität wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis

„Aspects of Time-Frequency Scattering and Towards Phase Scattering“

verfasst von / submitted by

Daniel Haider, BSc

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien, 2019 / Vienna, 2019

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	UA 066 821
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Masterstudium Mathematik
Betreut von / Supervisor:	Mag. Dr. Peter Balazs, Privatdoz.

Abstract

The scattering transform computes a layered network structure by cascading time-frequency magnitude decompositions. It was originally designed as translation invariant representation on $L^2(\mathbb{R})$, based on the wavelet transform and in applications it turned out to be a feature extractor for wide scale information. In this master thesis we discuss several aspects of this transform and motivate it from different points of view. We introduce it as natural extension of time-frequency decompositions, inspired by the structure of a convolution neural network and apply it to audio. In particular, we consider its capability of extracting information regarding to amplitude and frequency modulation and rhythmical features. In the second part of the thesis we extend the principle idea of scattering to alternative ways of representing time-frequency information based on the partial derivatives of the phase. For this, we broadly prepare the notion of phase in general to get a feeling of how to handle it in the context of time-frequency representations. Finally, the thesis is completed by showing the novel phase scattering procedure in action, leaving promising results.

Abstract (German)

Die Scatteringtransformation berechnet eine Kaskade von Zeit-Frequenz Magnitudenzerlegungen, sodass eine schichtartige Netzwerkstruktur ensteht. Ursprünglich wurde sie, basierend auf der Wavelet Tranformation als translationsinvariante Darstellung in $L^2(\mathbb{R})$ designt und in den Anwendungen zeigte sie sich als Extraktor von Information bezüglich längerer zeitlicher Abhängigkeiten. In der vorliegenden Masterarbeit werden einige Aspekte bezüglich dieser Transformation besprochen und sie wird von verschiedenen Blickwinkeln motiviert. Dabei wird sie als natürliche Erweiterung von Zeit-Frequenz Zerlegungen eingeführt, inspiriert von der Struktur eines Convolutional Neural Networks und auf Audiosignale angewandt. Insbesondere wird gezeigt, wie sie Informationen bezüglich Amplituden-, und Frequenzmodulation und rhythmischen Eigenschaften extrahiert. Im zweiten Teil der Areit wird die grundlegende Idee der Scatteringprozedur auf alternative Zeit-Frequenz Darstellungen erweitert, basierend auf den partiellen Ableitungen der Phase. Dazu wird zunächst der Begriff der Phase umfassend vorbereitet um ein gutes Gefühl zu bekommen, wie man damit in einem Zeit-Frequenz Zusammenhang umgehen kann. Schließlich wird die Arbeit abgeschlossen mit den neu eingeführten Phasen-Scatteringkoffizienten und ihrer vielversprechenden Anwendung auf Standardbeispiele.

Acknowledgement

Before I started with the actual topic of the thesis, my supervisor Peter encouraged me to submit something for the CMMR19, the conference on computer music multidisciplinary research in Marseille. I had done some experiments with the Gabor scattering transform and he said, »Just apply it on music«, which I did. Many thanks for his support to realize the participation at that conference. Also many thanks to Nicki and Andrés for their feedback on that first try of a research paper. The actual idea for this thesis then came from Nicki, he said, »Try scattering with phase derivatives«, which I did. Although things did not always seem to work out very nicely, the plots the phase derivatives produced were really inspiring from an artistic point of view (see Appendix) and finally, I managed to arrive at an output I am quite pleased with.
It was really great to spend the time during the thesis in the holy halls of the Acoustics Research Institute and even receive financial support by the FLAME project, thank you Peter, ARI-fan for life.

Introduction

Considering audio as any kind of correspondence of an acoustic event (travelling sound waves through the air), one could think of many creative ways to set up a representation for it. For example I repeatedly have this specific melody in my mind, which has been following me for quite a while. Somehow an abstract representation of this melody has to be stored in my brain. To formalize it, I could write down the notes corresponding to the melody on a music paper, such that everyone could decode and reproduce it.

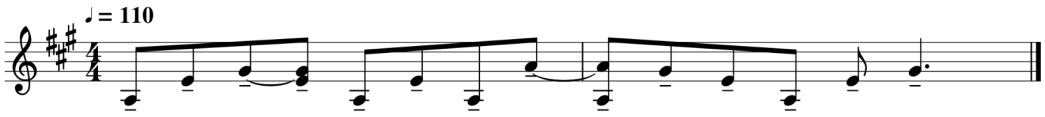

Figure 0.1: Partitur of a melody.

However, as good as I might be able to represent my melody, the actual music event when I humm it under the shower will never be the same as an attempt by anyone else reproducing the same melody from the music paper I wrote. This questions the consistency of this representation. Approaching the problem from the other end, we could capture the actual sound event with a microphone and measure the amplitude of the vibrating membrane. Apart from the analog tape version, these days we would rather go digital and store the measurements as numerical values. This yields a sampled time series representation of the acoustic event which is able to capture all the contained information in a sufficiently precise way. Plotting this representation gives though very little insights about the characteristics of the recorded sound and nobody could even guess how this might sound since beside the amplitudes and a rough time evolution there is nothing to see (Figure 0.2). Indeed, it is impossible to say how this could have sounded when it was recorded. So, it would be nice to have a representation which is able to depict some essential features, such as pitch and tone duration, like on the music sheet *and* still contains all the information of the sound event, like the time series. This can be achieved by so-called *time-frequency representations* which show the time series decomposed into its different frequencies. However, there are lots of other features that describe the information in acoustic events. When we (humans) listen, we identify and interpret abstract patterns on several levels. We follow compositions built on micro-structures, as well as long-term progressions and interpret fundamental ideas of over-all works, generalize and categorize on them, differentiate between different types of sounds and analyze their characteristics, filter semantic information, etc. All this information is encoded in the

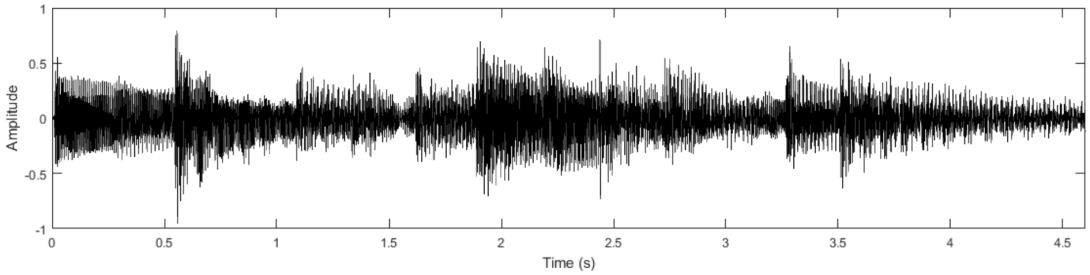

Figure 0.2: Plot of the time series of a digital recording of the melody in Figure 0.1, played on a classical guitar in fingerpicking style.

dependecies of different temporal scales in the envolement of the acoustic events and we learned how to extract it by experience. As we are living in an era, where one great goal in computer science seems to be anthropomorphizing computers, we want to make a computer also "listen" like humans do. This means we need to tell it how to access the different scales and how to extract something meaningful from them. And as we humans *learn* the skills to perform specific tasks, we also want to make the machines *learn*.

Instead of programming a system to perfom a certain task, the idea of *machine learning* is to feed it with data and let it adapt to the structure of the data by minimizing a loss between the output of the system and data. In other words, we *train* the system to fit the data. In the manner of *supervised learning*, we have given data $\mathbf{z}$, consisting of input-output pairs $(x_i, y_i)_{i=1}^m \in Z = X \times Y$, where X is a space of signals and Y a space of labels to them. The goal is to find a function $f : X \to Y$ which approximates a mapping that reproduces these pairs and additionally produces correct labels for new input data. For a proper approximation we need to find a class of functions, from that suitable approximators can be drawn. Such a space of functions is called a hypothesis space (or model space). Mathematically it is modelled to be a compact subset of the Banach space $\mathcal{C}(X, Y) = \{f : X \to Y, f \text{ contiuous}\}$ with norm $\|f\| := \max_{x \in X} |f(x)|$. To measure the performance of a function we can compute a distance between $f(x_i)$ and the corresponding label y_i. This distance is called *loss function* $\mathcal{L}(f, x_i, y_i)$, which we want to minimize during the learning procedure with a hope of a good generalization to unseen data. The balance of the "complexity" of the hypothesis space and the size of the training data plays an important role for generalization. A very flexible model will fit the data with relatively few training samples but will lead to "overfitting", which means that noise and outliers are interpreted as representative values for the data and thus, it will not produce reliable predictions. On the other hand, a stiff model will need a lot of training data to fit somehow, often not very well. To test how good the generalization to unseen data is, the dataset is usually split into a "training" and "test" set.

However, as it is in general impossible for the function to reach the original process of how the data was generated, it will end up aiming for a less complex representation of the

process, based on its main *features*. The term "feature" is not really well-defined. Roughly speaking, features describe structures of the data which are essential for a specific task. Mathematically speaking, we map the data into a lower dimensional (feature)space, where the important information is extracted and represented nicely, i.e. we aim for a mapping $\Phi : V \to Y$ with V being a lower dimensional space providing a representation $f(x) = \Phi(g(x))$, where $g : X \to V$ is "much simpler" than f. Φ is called a "feature extractor". Designing features in advance can be a key for a better performance with less training data. Especially in audio-related learning problems it seems that pre-processing the data is beneficial.

Our goal for the thesis will be to explore different concepts of representing certain features of audio signals, in particular to get a feeling of how to reach wider temporal scales. The *scattering transform* will be our tool for this.
The thesis is setup as following.

1. **Introduction to Time-Frequency Representations and Deep Learning**
   We give an introduction to the basic tools of time-frequency analysis, introducing the Gabor and the wavelet transform, followed by an introduction to convolutional deep neural networks where we point out the connection to our time-frequency tools.

2. **The Scattering Transform**
   Inspired by the ideas from the introductory chapter, we introduce the scattering transform as feature representator for audio signals. We discuss how it extracts information from amplitude and frequency modulated signals and extend the principle to let it also extract rhythmical features. The latter application resulted in an article in the CMMR19 conference proceedings with the title "Extraction of Rhythmical Features with the Gabor Scattering Transform".

3. **Aspects of the Phase**
   In this chapter we discuss the notion of phase and related concepts. In particular we introduce *channelized instantaneous frequency* (CIF) and *local group delay* (LGD) and deduce special representations, which allow novel analytic results on their computational appearence. The insights shall pave the way for the last chapter.

4. **Phase Scattering**
   Finally, we define the novel *phase scattering coefficients* based on CIF and LGD and conduct promising experiments on toy examples.

All codes producing the numeric plots were done with MATLAB. Most of them make use of the LTFAT toolbox https://ltfat.github.io/. The sound examples were written and recorded/produced by the author. All together, including the content around the CMMR19 article can be found under https://github.com/danedane-haider/Phase-Scattering-Masterthesis.

Contents

1 Time-Frequency Representations and Deep Learning

To realize our ideas in a formal way, we start by considering an acoustic or sound event as a mapping, assigning an amplitude to every point in time and call it an *audio signal*. This mapping can be viewed from different perspectives. On one hand, it would seem natural to think it as a continuous function $f : \mathbb{R} \to \mathbb{C}$ with finite energy, i.e. $f \in \mathbf{L}^2(\mathbb{R}) = \{f : \mathbb{R} \to \mathbb{C} : \int_{\mathbb{R}} |f(t)|^2 dt < \infty\}$, the function space of square integrable functions. This space is a Hilbert space, which is a complete normed linear inner product space over $\mathbb{R}$ or $\mathbb{C}$ and has some very nice properties. On the other hand, signal processing deals with discrete series which suggests $\ell^2(\mathbb{Z}) = \{f : \mathbb{Z} \to \mathbb{C} : \sum_{k\in\mathbb{Z}} |f(k)|^2 dt < \infty\}$, the space of square summable sequences. However, in real life a computer can only process signals of finite length, in this way we will consider our signals as members of the finite vector space $\mathbb{C}^L$, respectively $\mathbb{R}^L$ [18]. For audio signals as we set them now, we will introduce some of the basic concepts in time-frequency analysis and signal processing to find ways of representing them in nice ways. Motivated by our thoughts on machine learning in the introductory part, we will also give a brief introduction to deep neural networks, which will care for further inspiration.

1.1 Basics of Time-Frequency Analysis

As fundamental tool we define the **Fourier Transform** on $\mathbf{L}^1(\mathbb{R}) = \{f : \mathbb{R} \to \mathbb{C} : \int_{\mathbb{R}} |f(t)| dt < \infty\}$ as

$$\hat{f}(\omega) = \int_{\mathbb{R}} f(t) e^{-2\pi i \omega t} dt. \tag{1.1}$$

For our audio signals in $\mathbf{L}^2(\mathbb{R})$, it is in general not defined pointwise, but through a limit. Mathematically, the Fourier transform is a linear operator with plenty of nice properties. For functions in $\mathbf{L}^1(\mathbb{R}) \cap \mathbf{L}^2(\mathbb{R})$, by Plancherel's Theorem, $\|f\|_2 = \|\hat{f}\|_2$, i.e. $\mathcal{F}$ is an isometry. Therefore, $\mathcal{F}$ extends to a unitary operator on $\mathbf{L}^2(\mathbb{R})$. If $f \in \mathbf{L}^1(\mathbb{R})$ and $\hat{f} \in \mathbf{L}^1(\mathbb{R})$, it is invertible explicitely,

$$f(t) = \int_{\mathbb{R}} \hat{f}(\omega) e^{2\pi i t \omega} d\omega \quad \forall t \in \mathbb{R}, \tag{1.2}$$

in other words, $\mathcal{F}^{-1} f = (\mathcal{F}\hat{f})(-\bullet)$. This still holds in $\mathbf{L}^2(\mathbb{R})$.
With the eyes of an engineer or someone who wants to do audio signal processing, $\hat{f}(\omega)$ is understood as the spectrum of f, depending on the frequency ω.

Discrete Fourier Transform

With the Fourier transform of an audio signal we have now a tool for obtaining its frequency content, i.e. information about pitch, which is nice. To perform it numerically on a computer, we define the discrete Fourier transform for a finite discrete signal $f \in \mathbb{R}^L$ as

$$\hat{f}[k] = \sum_{n=0}^{L-1} f[n] e^{-2\pi i \frac{kn}{L}}. \tag{1.3}$$

However, as a signal representation it is still not suitable, since indeed we can see which pitches are present in the signal but we loose all the temporal information.

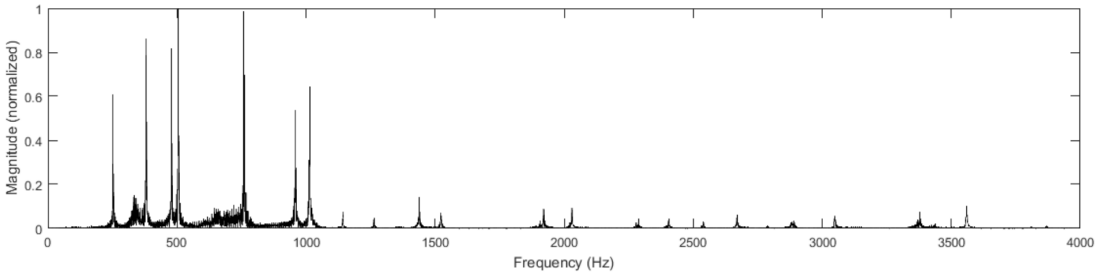

Figure 1.1: Magnitude of the frequency spectrum up to 4000Hz of the recording of the guitar playing the melody. It was computed with MATLABs `fft` routine.

1.1.1 The Short-Time Fourier Transform and the Spectrogram

To keep the temporal alignment of the frequency information, it motivates to consider the frequency content of only one small part of the signal at a time to obtain a time-ordered series of frequency decompositions. To realize this we multiply the signal with a time-localized window function and shift it continuously in time. Usually the window has a bell-shape with a certain width, decaying to zero at the ends to enable a smooth transition. This defines the **Short-Time Fourier Transform** (STFT) [18].

Definition 1 (STFT)**.** *Let $g \in \boldsymbol{L}^2(\mathbb{R})$ be a window function, then the STFT of $f \in \boldsymbol{L}^2(\mathbb{R})$ is defined as*

$$\mathcal{V}_g f(t,\omega) = \mathcal{F}(f \cdot g(\tau - t))(\omega) = \int_{\mathbb{R}} f(\tau)\overline{g(\tau - t)} e^{-2\pi i \omega \tau} d\tau, \quad (t,\omega) \in \mathbb{R}^2. \tag{1.4}$$

From this we obtain complex valued coefficients, which still are not very suitable for the purpose of illustration. The **spectrogram** S_f of f is the classic under all time-frequency representations and is obtained by taking the energy of the STFT coefficients, i.e. the squared magnitude of $\mathcal{V}_g f(t,\omega)$, sampled on a rectangular lattice $a\mathbb{Z} \times b\mathbb{Z}$, i.e.

$$S_f[n,m] = |\mathcal{V}_g f(an, bm)|^2, \quad (a,b) \in \mathbb{R}^2.$$

Discrete STFT

To realize this representation on a computer, we again have to put on our finite discrete glasses and consider a version of the STFT for our signal $f \in \mathbb{R}^L$. Instead of continuously shifting our window along f, we make discrete steps in time and frequency.

Definition 2 (Discrete STFT)**.** *Let $g \in \mathbb{R}^N$, then the discrete STFT of a sequence $f \in \mathbb{R}^L$ w.r.t. g is defined as*

$$\mathcal{V}_g f[m,k] = \sum_{n=0}^{L-1} f[n]g[n-m]e^{-2\pi i \frac{kn}{L}}, \quad m,k \in \{0,\dots,L-1\}. \tag{1.5}$$

Now we can visualize the time-frequency content obtained by the STFT, see Figure 1.2. This representation of our signal depicts both, time and frequency information simultaneously. The horizontal lines correspond to tonal elements of the signal which have sinusoidal characteristics and the vertical lines to transient events, which last only for a short time duration and contain many differerent frequencies at once.

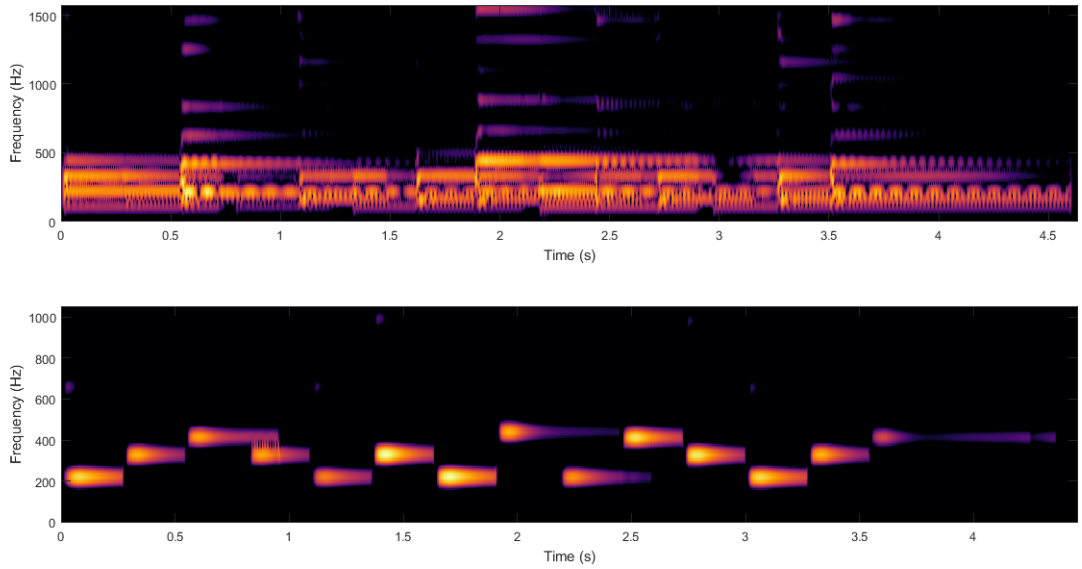

Figure 1.2: (Top) Spectrogram up to 1500Hz of the recording of the guitar playing the melody, sampled from the underlying STFT with $a = 32, b = 24$ and a Hann window with $N = 1024$. One could vaguely identify the melody in the plot, but since a guitar is harmonically a very rich intrument with lots of harmonics and resonances, it is hard to get a clear image of the melody.
(Bottom) To clarify the image above, this plot depicts a spectrogram up to 1000Hz of the same melody in MIDI, played by some digital synthesizer. Here one can see clearly the similarity to the partitur in Figure 0.1.

1.1.2 Frames

We want to change the perspective on time-frequency decompositions and embed it in a more general concept of expanding functions in a Hilbert space $\mathcal{H}$ by **frames** [11].

Definition 3 (Frame)**.** *Let $\mathcal{H}$ be a Hilbert space. A sequence $\{f_k\}_{k\in\mathcal{K}} \subset \mathcal{H}$ with $\mathcal{K}$ being a countable index set is called a frame, if there are positive constants $A \leq B$, such that*

$$A\|f\|^2 \leq \sum_{k\in\mathcal{K}} |\langle f, f_k\rangle|^2 \leq B\|f\|^2 \tag{1.6}$$

holds for all $f \in \mathcal{H}$. The members f_k of the frame are called its atoms.

If $A = B$ the frame is said to be *tight* and if additionally $A = 1$, it is called *Parseval frame* and if we consider only the upper inequality, it is called *Bessel sequence*. The concept of frames provides a powerful tool to represent functions in a more flexible way than bases allow. Unlike with bases, frames can be linearly dependent (redundant), which means that there is not an unique coefficient sequence determining the expansion, but infinitely many. In this way we have certain degrees of freedom on the coefficients and can tweak them to our benefits. The conditions in (1.6) promise stability and invertibility of the representation, i.e. we can construct nice representations with customized properties on the coefficients [11]. To realize a frame expansion we introduce some operators, which are fundamental in frame theory. Let $\{f_k\}_{k\in\mathcal{K}}$ be a sequence in $\mathcal{H}$, then we define

(i) the **analysis operator**

$$\begin{aligned} \mathbf{T}^* :\ & \mathcal{H} \to \ell^2(\mathbb{N}) \\ & f \mapsto \left\{\langle f, f_k\rangle\right\}_{k\in\mathcal{K}} \end{aligned} \tag{1.7}$$

(ii) the **synthesis operator**

$$\begin{aligned} \mathbf{T} :\ & \ell^2(\mathbb{N}) \to \mathcal{H} \\ & \{c_k\}_{k\in\mathcal{K}} \mapsto \sum_{k\in\mathcal{K}} c_k f_k \end{aligned} \tag{1.8}$$

(iii) the **frame operator** as composition of analysis and synthesis operator

$$\begin{aligned} \mathbf{S} := \mathbf{T}\mathbf{T}^* :\ & \mathcal{H} \to \mathcal{H} \\ & f \mapsto \sum_{k\in\mathcal{K}} \langle f, f_k\rangle f_k. \end{aligned} \tag{1.9}$$

$\{f_k\}_{k\in\mathcal{K}}$ being a Bessel sequence provides, that all these operators are well-defined and bounded and that, as the notation already indicates, $\mathbf{T}$ and $\mathbf{T}^*$ are indeed adjoint operators. $\{f_k\}_{k\in\mathcal{K}}$ being a frame makes $\mathbf{S}$ additionally be a positive and invertible operator on $\mathcal{H}$.
For the sake of completeness we also define a frame in a continuous setting, as we will encounter it later in the thesis.

Definition 4 (Continuous Frames)**.** *Let $\mathcal{H}$ be a Hilbert space and (Ω, μ) be a measure space with a positive measure μ. A mapping $F : \Omega \to \mathcal{H}$ is called a continuous frame w.r.t. (Ω, μ), if*

(i) $\omega \mapsto \langle f, F(\omega)\rangle$ is a measurable function on Ω and

(ii) there are positive constants $A \leq B$, such that

$$A\|f\|^2 \leq \int_\Omega |\langle f, F(\omega)\rangle|^2 d\mu(\omega) \leq B\|f\|^2 \tag{1.10}$$

holds for every $f \in \mathcal{H}$.

We shall use this general concept of stably representating functions in the context of time and frequency.

1.1.3 Gabor Transform

Remember, the idea of the STFT was to cut a signal into pieces and apply Fourier transforms on those pieces separately, in other words, we look at the frequency spectrum at every of those pieces. However, in a mathematical manner time and frequency can be treated symmetrically and so, we will aim to expand our signal with respect to certain atomic functions, that carry the time-frequency information. Herefore we introduce two important linear operators.

(i) For $a \in \mathbb{R}$, the **translation operator** T_a is defined by

$$(T_a f)(t) = f(t-a), \quad t \in \mathbb{R}$$

(ii) For $b \in \mathbb{R}$, the **modulation operator** M_b is defined by

$$(M_b f)(t) = f(t)e^{2\pi i b t}, \quad t \in \mathbb{R}$$

On $\mathbf{L}^2(\mathbb{R})$, they are bounded and unitary. In the context of time-frequency analysis, T_a is a time shift by a and due to the fact $\mathcal{F}M_b = T_b\mathcal{F}$, M_b can be thought as a frequency shift by b. Their composition M_bT_a is then a **time-frequency shift** operator with $M_bT_af(t) = e^{2\pi i b t}f(t-a)$ for $(a, b) \in \mathbb{R}^2$ (Figure 1.3).

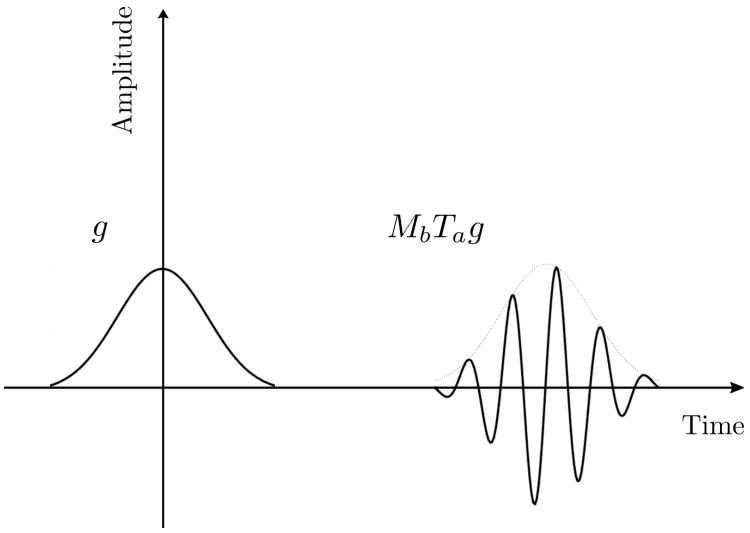

Figure 1.3: Schematic plot of a time-frequency shift of a window g.

Taking a window function g and considering the set of all its time-frequency shifted versions leads to a so-called **Gabor system**.

Definition 5 (Gabor System)**.** *Let $g \neq 0$ be a function in $\mathbb{R}$ and a, b positive real numbers, then*

$$\mathcal{G}(g, a, b) = \{M_{mb}T_{na}g\}_{(m,n)\in\mathbb{Z}^2} \tag{1.11}$$

is called a Gabor System.

A Gabor system is called a *Gabor frame* if it is a frame. To obtain a Gabor expansion of a function $f \in L^2(\mathbb{R})$ we use the analysis operator introduced earlier:

$$\mathbf{T}^*_{\mathcal{G}} f = \left\{\langle f, M_{mb}T_{na}g\rangle\right\}_{(m,n)\in\mathbb{Z}^2} \tag{1.12}$$

This has a special grid structure, in particular it can be viewed as coefficients on a rectangular lattice $\Lambda = a\mathbb{Z} \times b\mathbb{Z}$. Note that one can rewrite the STFT to be

$$\mathcal{V}_g f(t, \omega) = \int_{\mathbb{R}} f(\tau)\overline{g(\tau - t)e^{2\pi i\omega\tau}}d\tau = \langle f, M_\omega T_t g\rangle, \tag{1.13}$$

in other words, a Gabor expansion is a STFT, sampled on a rectangular lattice and thus, the spectrogram can be computed directly from it. With that point of view we can interpret a time-frequency representation as decomposition of the signal with time-frequency-localized atoms. This localization can schematically drawn as in Figure 1.4.

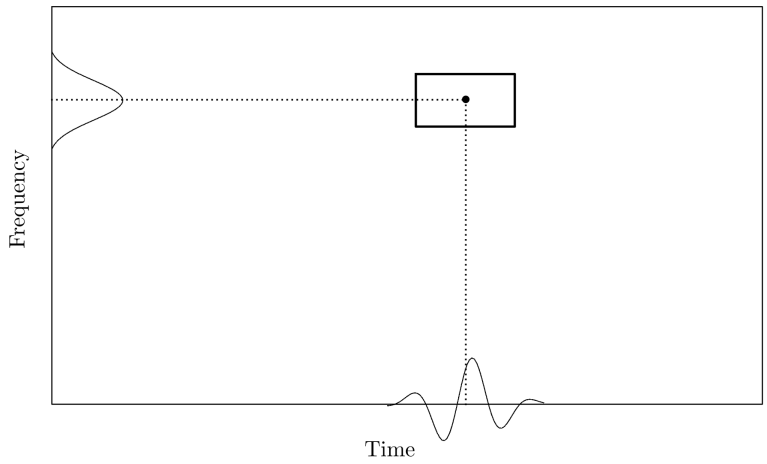

Figure 1.4: Schematic plot of a time-frequency localization.

Discrete Gabor Transform

Again, we discretize this transfom to obtain a routine we can implement on a computer. Our discrete atoms then write as

$$\ell \mapsto g[\ell - na]e^{2\pi i \frac{mb\ell}{M}}, \quad m = 0, \dots, M-1,\ n = 0, \dots, N-1,\ a, b \in \mathbb{N} \tag{1.14}$$

where $a \in \mathbb{N}$ is the time hop size, N the number of temporal bins and M the number of frequency bins. This implicitly assumes that the length of the analyzed signal is $L = aN = bM$, where b is the frequency hop size. This equality is achieved by truncating or zero-padding the signal to the appropriate length. With these conventions, the discrete Gabor transform w.r.t. a window $g \in \mathbb{R}^K$ of a sequence $f \in \mathbb{R}^L$ computes coefficients $c \in \mathbb{C}^{M \times N}$ given by

$$c[m, n] = \sum_{\ell=0}^{L-1} f[\ell] e^{-2\pi i \frac{mb\ell}{M}} g[\ell - na] \tag{1.15}$$

This is very similar to the discrete STFT we defined earlier, only that we have the subsampling already in the sense that $c[m, n] = \mathcal{V}_g f(na, mb)$.
To perform the transform in MATLAB we use the LTFAT routines `dgt` and `dgtreal` [34].

An Excursion to Filter Banks

In signal processing, so-called *filter banks* are used for the analysis of a signal in terms of its time-frequency content [7]. A filter bank is a set of band-pass filters $\{w_k\}_{k \in \mathcal{K}}$ with some indexset $\mathcal{K}$. The idea is to divide the frequency domain in slices which then are excited by the signal via the operation of *convolution*.

Convolution is formally defined for two functions $f, g \in \mathbf{L}^1(\mathbb{R})$ by

$$(f * g)(x) = \int_{\mathbb{R}} f(y)\overline{g}(x-y)dy. \tag{1.16}$$

It is a crucial operation in Fourier analysis, since we have the fundamental relation $\widehat{f * g} = \widehat{f} \cdot \widehat{g}$. For the discrete setting, let $f \in \mathbb{R}^L$ and $w_\ell \in \mathbb{C}^M$ be the ℓ-th filter, then

$$(f * w_\ell)[m] = \sum_{n \in \mathbb{Z}} f[n]\overline{w_\ell}[m-n] \tag{1.17}$$

computes its excitation by f. With that, $\{(f * w_k)\}_{k \in \mathcal{K}}$ is the filter bank decomposition of f w.r.t. $\{w_k\}_{k \in \mathcal{K}}$ (Figure 1.5).

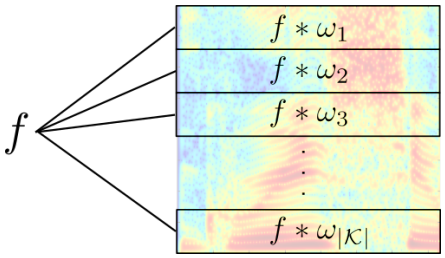

Figure 1.5: Schematic filter bank decomposition of f.

The Gabor transform can also be interpreted as being constructed via a filter bank. Herefore we introduce the **Involution** Operator $\mathcal{I}$ by

$$\mathcal{I}(f(t)) = f(-t) \tag{1.18}$$

and easily see that

$$\begin{aligned} \langle f, M_{mb}T_{na}g \rangle &= \sum_{k \in \mathbb{Z}} f[k]\overline{M_{mb}T_{na}g[k]}dt \\ &= \sum_{k \in \mathbb{Z}} f[k]\overline{M_{mb}g[k-na]} \\ &= \sum_{k \in \mathbb{Z}} f[k]\overline{M_{mb}\mathcal{I}(g[na-k])} = (f * M_{mb}\mathcal{I}(g))[na]. \end{aligned} \tag{1.19}$$

In that manner, the Gabor transform and thus, the spectrogram can be constructed via the filter bank $\{M_m\mathcal{I}(g)\}_{m \in \mathbb{Z}}$. It is also called *Fourier modulated filter bank*.

Stiffness of the Gabor Approach

In Section 1.1.1 we mentioned the specific shape of the window we use for the time-frequency localization. Especially, its width plays a crucial role for the representation.

This refers to the *time-frequency resolution* of the representation. Using a wide window will depict tonal information well, because the signal is considered over a large time interval for the computation of the time-frequency coefficients, which allows to capture well the frequency content around the single time positions. However, this blurres the temporal allocation, i.e. we obtain a high frequency, but a low time resolution. On the other hand, using a narrow window allows a good time resolution, which is desirable for transient events, but fails to capture frequency content adequately. Figure 1.6 shows how the width of the window effects the time-frequency resolution.

The Gabor transform uses a single window to compute all coefficients, which means that the time-frequency resolution is fixed on the whole time-frequency plane. However, this mismatches how the human auditory system resolves time-frequency information. Humans are able to distinguish well between small differences in higher frequency ranges, but struggle to perceive their temporal appearance, i.e. the frequency resolution is high, but the time resolution is poor. In the lower frequency ranges it is contrary. Thus, it is indeed desirable to construct also representations that resolve time-frequency information in a more flexible way. A simple tool to archieve this is the wavelet transform.

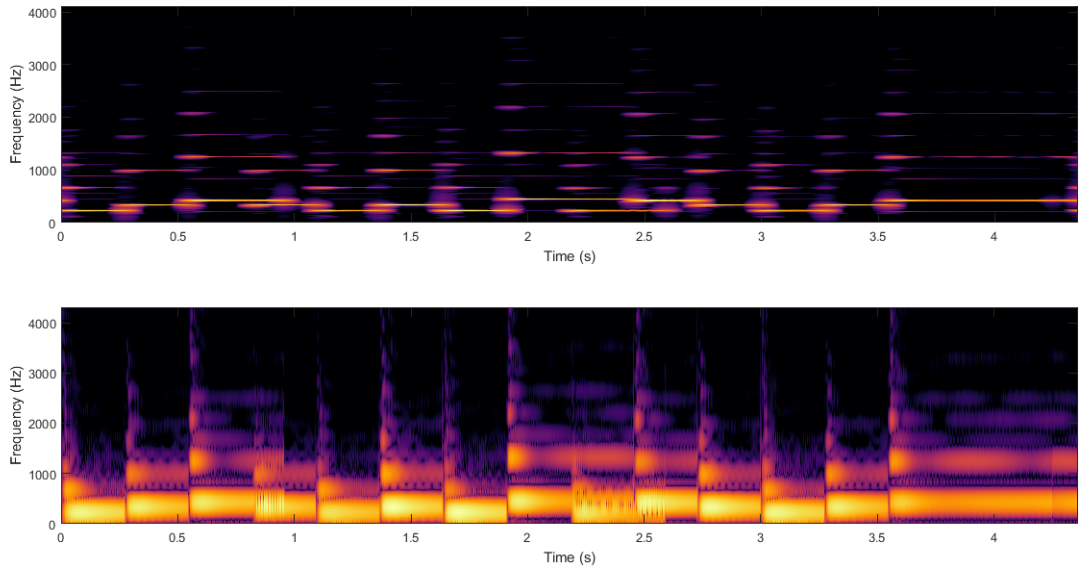

Figure 1.6: Magnitudes of a Gabor transform of a synthesizer playing the melody, computed with `dgtreal` using the parameters $a = 64, M = 4096$ and
(Top) a very wide Hann window with $K = 8192$ and
(Bottom) a very narrow Hann window with $K = 256$.

1.1.4 Wavelet Transform

The wavelet transform computes a time-frequency representation with varying resolution along the frequency scale by the operation of dilation. For $c \in \mathbb{R}$, we define the **dilation operator** D_c by

$$(D_c f)(t) = \frac{1}{\sqrt{c}} f\Big(\frac{t}{c}\Big),\ t \in \mathbb{R}.$$

A wavelet is a function $\psi \in \mathbf{L}^2(\mathbb{R})$ with zero average, i.e.

$$\int \psi(\tau) d\tau = 0. \tag{1.20}$$

For the analysis of a function in terms of time and scale, the wavelet is dilated with a scale parameter $\lambda \in \mathbb{R}$ and translated by $t \in \mathbb{R}$:

$$\psi_{t,\lambda}(\tau) = (D_\lambda T_t \psi)(\tau) = \frac{1}{\sqrt{\lambda}} \psi\Big(\frac{\tau - t}{\lambda}\Big). \tag{1.21}$$

Note, that (1.20) still holds.
The wavelet transform of a $f \in \mathbf{L}^2(\mathbb{R})$ at scale λ and position t is defined by [21]

$$Wf(t,\lambda) = \langle f, \psi_{t,\lambda} \rangle = \int f(\tau) \frac{1}{\sqrt{\lambda}} \psi^*\Big(\frac{\tau - t}{\lambda}\Big) d\tau = f * \psi^*_\lambda(t) \tag{1.22}$$

with

$$\psi^*_\lambda(t) = \frac{1}{\sqrt{\lambda}} \overline{\psi}\Big(\frac{-t}{\lambda}\Big) \tag{1.23}$$

Equation (1.22) induces that the wavelet transform can be constructed via a filter bank, consisting of the filters ψ^*_λ. We may interpret the $\psi_{t,\lambda}$ as window functions that are localized in time and scaled to compute a *time-scale decomposition* of f. However, if we want to analyze the frequency content of f, it is necessary to use complex analytic wavelets. A function $f_a \in L^2(\mathbb{R})$ is called *analytic*, if its Fourier transform is zero for negative frequencies, i.e. $\widehat{f_a}(\omega) = 0$ for $\omega < 0$. Such analytic wavelets can be seen as frequency-modulated window functions. We may interpret the operation of dilation as reaching different frequencies to obtain a time-frequency decomposition. Figure 1.7 shows resolution schemes of the Gabor transform and the wavelet transform with the corresponding window shapes and illustrates how modulation and scaling applies. Moreover, an analytic wavelet transform defines a local time-frequency energy density $P_W f$, called *scalogram* and is computed analog to the spectrogram by applying a squared modulus,

$$P_W f(t,\xi) = |Wf(t,\lambda)|^2, \tag{1.24}$$

where ξ is the center frequency of $\widehat{\psi}_{t,\lambda}$.

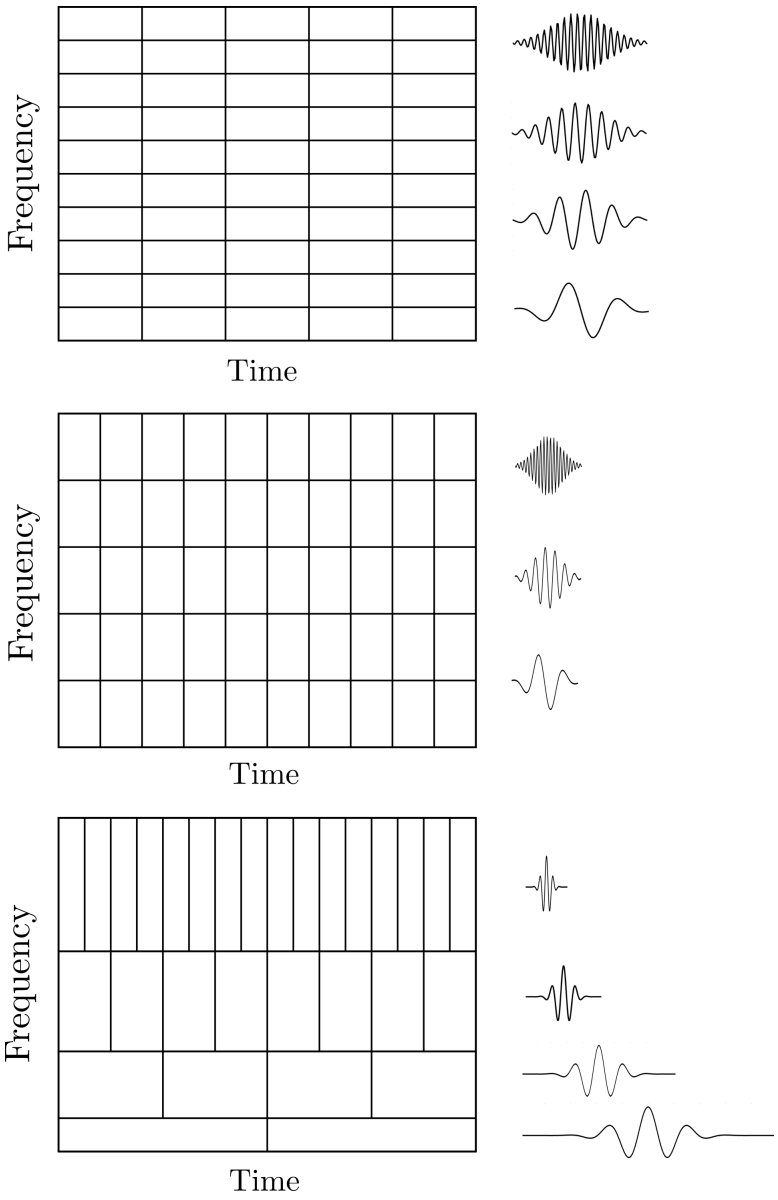

Figure 1.7: Schematic superposition of the time-frequency plane by localized atoms:
(Top) Fixed time-frequency resolution using a wide window.
(Mid) Fixed resolution using a narrow window.
(Bottom) Dyadically increasing time, respectively decreasing frequency resolution along the frequency axis using a scaled wavelet.

Discrete Wavelets

For the discrete case, there are several discretization schemes of the wavelet transform, e.g. sampling the scale along an exponential sequence $\{a^j\}_{j\in\mathbb{Z}}$ and the time translation uniformly at intervals proportionally to a^j, with a sufficiently small dilation step $a > 1$ [21]. In the practical use, however, instead of discretizing a continuous wavelet transformation, one usually uses a filter bank to implement it. A widely used method to construct a wavelet-type filter bank is the *constant-Q transform* (CQT). For a discrete finite signal $f \in \mathbb{R}^L$ and a sequences of windows $g_k \in \mathbb{R}^{N[k]}$, it is defined by

$$C_f[k] = \frac{1}{N[k]} \sum_{n=0}^{N[k]-1} g_k[n] f[n] e^{-2\pi i \frac{nQ}{N[k]}}, \tag{1.25}$$

where $Q = (2^{1/B} - 1)^{-1}$ is the constant ratio of frequency and resolution for B frequency bins per octave. The window widths of g_k depend adaptively on the k-th bin via $N[k] = \frac{Q}{\xi_k}$ in milliseconds. $\xi_k = \xi_{min}(2^{1/B})^k$ is the center frequency in Hz of the k-th frequency bin depending on a startvalue ξ_{min}. Figure 1.8 depicts the scalogram of the melody, computed with a CQT.

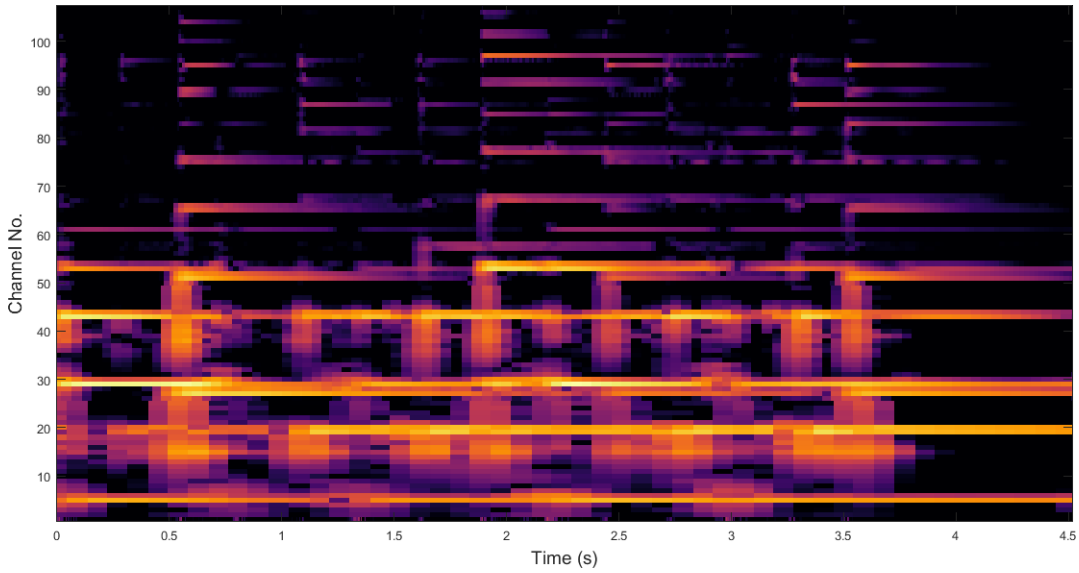

Figure 1.8: Scalogram of the guitar recording of the melody using the LTFAT-routine `cqtfilters` with $B = 24$ bins per octave, $\xi_{min} = 100$ and $\xi_{max} = 2000$.

We have now introduced the basics of the main concepts to represent audio signals in terms of their time and frequency content and will continue with a specific model in machine learning that reveals a deep connection in its construction to the time-frequency transforms we had so far. It will inspire us to an idea of extending our transforms.

1.2 Deep Neural Networks

Deep neural networks (DNNs) build the cornerstone of a relatively new branch of machine learning called *Deep Learning* and have established themselves as a very successful hypothesis space for various learning tasks. The idea is to setup a function, that can theoretically approximate any other function via an optimization procedure (training) [17]. The structure of a DNN is motivated by the way our brain works in a very simplified way. It has a layered structure, each consisting of so-called "neurons" which filter the importance of the information arriving. These neurons can be defined as functions $\nu : \mathbb{C}^s \to \mathbb{C}$ with $x \mapsto \sigma\big(x^* w - b\big)$ where $w \in \mathbb{C}^s$ is a vector containing the weights w.r.t. the neurons x_i. b is a bias value acting as affine offset and $\sigma : \mathbb{C} \to \mathbb{C}$ is a non-linear function, called "activation function" that controls the significance of the neurons to the network (Figure 1.9). There are several activation functions used in practice, among them the Heaviside (step function) as biological motivation, sigmoids, rectified linear units (ReLU), modulus functions, etc., commonly required to be Lipschitz continuous (Figure 1.10).

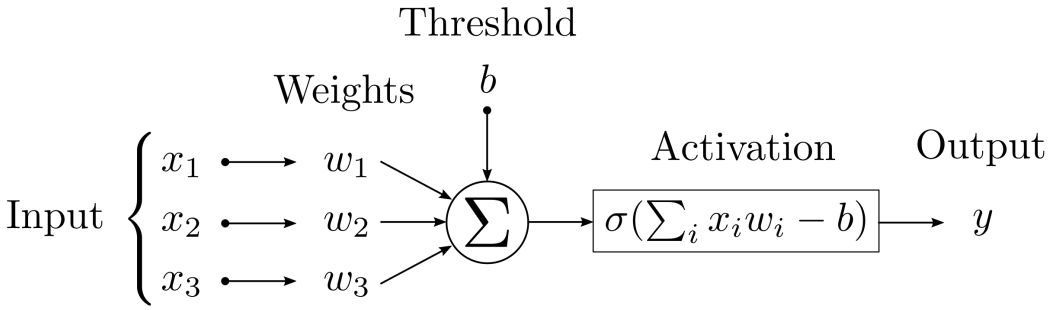

Figure 1.9: The structure of an artificial neuron.

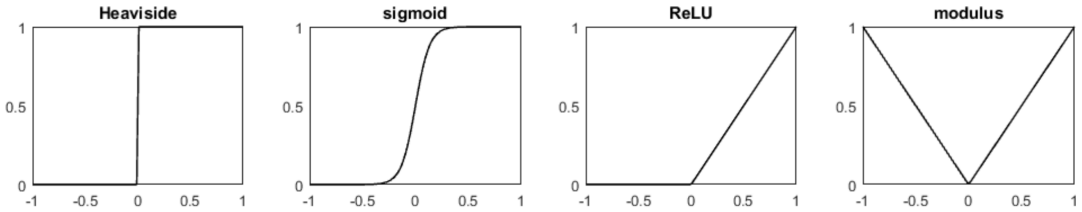

Figure 1.10: Different activation functions in practical use.

With these ingredients we can define a DNN:

Definition 6 (Deep Neural Network)**.** *Let $L, d, N_1, ..., N_L \in \mathbb{N},\ L \geq 2$ and $A_\ell : \mathbb{C}^{N_{\ell-1}} \to \mathbb{C}^{N_\ell},\ 1 \leq \ell \leq L$ be affine linear maps given by $f \mapsto W_\ell f + b_\ell$ with matrices $W_\ell \in \mathbb{C}^{N_{\ell-1} \times N_\ell}$ and bias vectors $b_\ell \in \mathbb{C}^{N_\ell}$. Furtermore let $\sigma : \mathbb{C} \to \mathbb{C}$ be a non-linear function acting component-wise on a matrix. A map $\Phi : \mathbb{C}^d \to \mathbb{C}^{N_L}$ given by*

$$\Phi(f) = A_L(A_{L-1}\sigma(...\sigma(A_1(f)))), \quad x \in \mathbb{C}^d \tag{1.26}$$

is called a deep neural network.

L is the number of layers ($L \geq 2$ refers to the network being considered as "deep"), d the dimension of the input space and N_ℓ the dimensions of the single layers. The setup of a DNN w.r.t. the used parameters is called its *architecture* (Figure 1.11). The classical

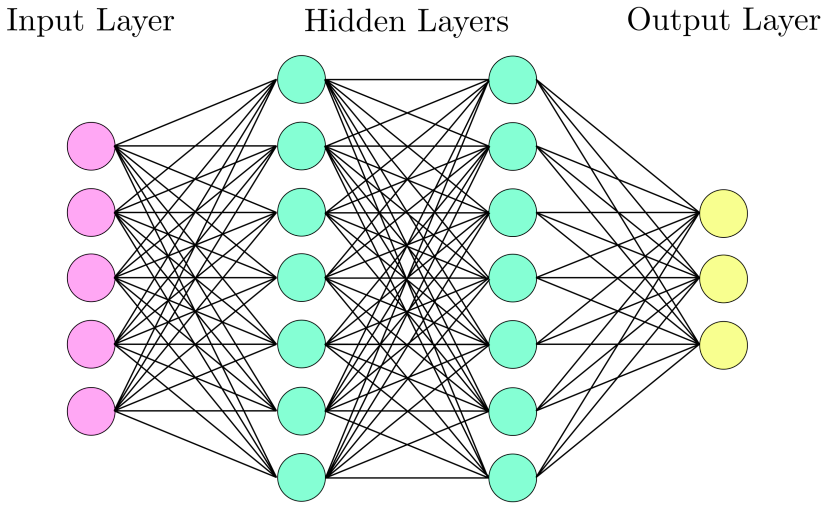

Figure 1.11: The architectures of a deep neural network with $L = 3$, $d = 5$, $N_1 = N_2 = 7$ and $N_3 = 3$.

architecture based on matrix multiplication has fully connected layers, i.e. every output neuron interacts with every input neuron. This is computationally expensive and in some sense unnecessarily much information processed, as it is reasonable to assume that many important connections lie within a certain range in the input. It therefore makes sense to model a network architecture that allows sparse interactions between the neurons by capturing the input part by part (Figure 1.12).

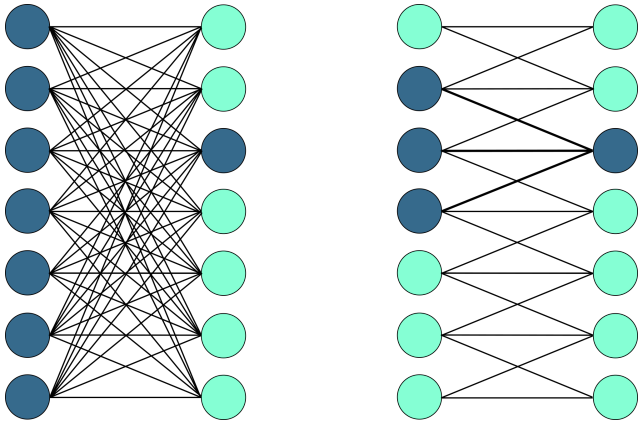

Figure 1.12: Left: dense connections. Right: localized connections of width 3.

1.2.1 Convolutional Neural Networks

Convolutional Neural Networks (CNN) are a specialized form of DNNs to process grid-like data. The idea is to convolve the input with several different filters in parallel that are much smaller than the input dimension. This can be interpreted as localization of certain properties of the data. This architecture connects the layers in a sparser way than the classical approach and allows to capture information more focussed.

Definition 7 (Convolutional Neural Network - 1D)**.** *Let $f \in \mathbb{C}^N$ with K_ℓ filters $w_\ell^k \in \mathbb{C}^{n_\ell}$, $1 \leq k \leq K_\ell$ used for the ℓ-th layer where $n_\ell \ll N$. The basic building block of a CNN is the convolution operation and so, with an activation function σ as before, the ℓ-th layer of the network can be defined, similar to* (1.26)*, by*

$$f_\ell := \sigma\Big(\sum_{k=1}^{K_\ell} f_{\ell-1} * w_\ell^k + b_\ell \otimes \mathbf{1}\Big), \tag{1.27}$$

where $f_{\ell-1}$ is the output of the $(\ell-1)$-th layer ($f_0 = f$), $\otimes$ denotes the Kronecker product and $\mathbf{1}$ is a matrix of the same size as $f_{\ell-1} * w_\ell^k$ (depending on the type of convolution used), consisting of ones.

In fact, (1.27) defines an affine linear map, since the operation of convolution can be realized by a multiplication with a Block-Toeplitz matrix and thus, a CNN is indeed a special form of a DNN.

Finally, in order to make a feature extraction procedure possible, the dimensionality of the input data has to be reduced. This can be achieved (among other techniques [25, 31]) by *pooling*. Pooling computes a "summary-statistic" of nearby elements, so it performs a downsampling, i.e. decreases the dimensionality and generates invariances to specific deformations and variations in the data. Furthermore, the range of the filters

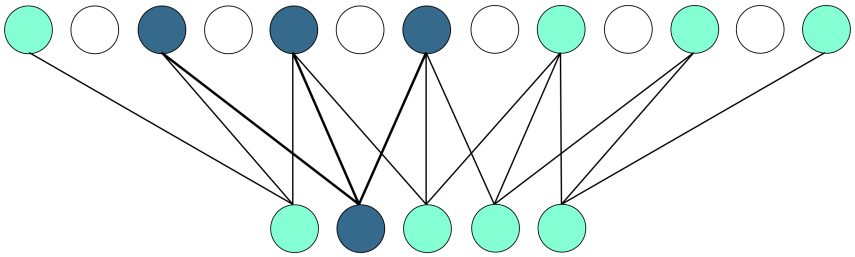

Figure 1.13: Pooling with a scale of 2 indicates the dimensionality reduction and the expansion of the filter range.

in the subsequent layers expand since the filters are applied to the pooled "summary"-elements, that are representative for a whole neighborhood of elements of the previous layer. Thus, the deeper the network gets, the wider dependencies are captured. So, a typical layer has three stages: Filtering by several different filters in parallel, applying a non-linear activation function and then apply a dimensionality reduction. (Figure 1.14) This is by far the most basic setup for a CNN. There is a lot of research going on, vastly exploring various architectures, filtering techniques, etc.

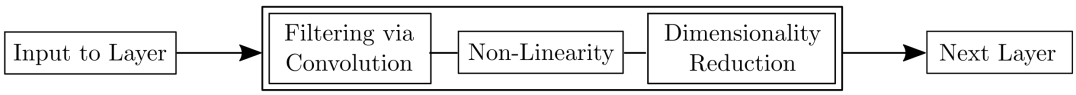

Figure 1.14: The three stages of a basic convolutional layer.

Originally, CNNs were introduced in image processing and have led to an immense progress in image-related learning tasks [16]. In this case, the input is a matrix in $\mathbb{R}^{N\times M}$, filtered by 2D filters $w_\ell^k \in \mathbb{C}^{n_\ell \times m_\ell}$. It seems to make totally sense to apply a CNN also on the images obtained by time-frequency decompositions of audio signals [23]. Those decompositions already provide a representation of basic features of a signal, namely its time-frequency content, i.e. small-scale information. It has been shown that using such a representation as input leads to the possibility of working with less data while achieving the same level of performance, i.e. incorporating known features indeed seems to play an important role in learning [27].

1.2.2 A Link to Frames

Applying a CNN on a spectrogram one can deduce a particularly nice representation in terms of a **Multiplier** [5]. The idea to this comes from [14].

Definition 8 (Multiplier)**.** *Let $\Phi = \{\phi_k\}, \Psi = \{\psi_k\}$ and $\mathbf{m} = \{m_k\}$ be sequences in some Hilbertspace $\mathcal{H}$. The operator $M_{\mathbf{m},\Phi,\Psi}$, given by*

$$M_{\mathbf{m},\Phi,\Psi} f = \sum_k m_k \langle f, \phi_k \rangle \psi_k \tag{1.28}$$

is called a multiplier *for $f \in \mathcal{H}$. The sequence $\mathbf{m}$ is called* symbol *of just of the multiplier.*

If $\Psi = \Phi$, then we write $M_{\mathbf{m},\Psi}$. A multiplier consists of the analysis operator of a sequence Φ, followed by a pointwise multiplicative modification of the obtained coefficients by a symbol $\mathbf{m}$ and finally the synthesis operator of a sequences Ψ. In other words, we modify the analysis coefficients $\langle f, \phi_k \rangle$ by multiplying them with m_k before re-synthesizing with the ψ_k.
With (1.13) we showed that the spectrogram is computed with a Gabor expansion and a pointwise modulus squared, i.e. denoting $g_{m,n} := M_{mb} T_{na} g$, the spectrogram can be written as $\mathcal{S}_f = \{|\langle f, g_{m,n} \rangle|^2\}_{m,n \in \mathbb{Z}}$. Well, taking a discrete filter $w \in \mathbb{C}^{K \times L}$ and convolve it with the transformed values gives

$$(\mathcal{S}_f * w)(\ell, k) = \sum_{m,n} |\langle f, g_{m,n} \rangle|^2 w(\ell - m, k - n). \tag{1.29}$$

With $|\langle f, g_{m,n} \rangle|^2 = \overline{\langle f, g_{m,n} \rangle} \langle g_{m,n}, f \rangle$ we get that the convolution above is equal to

$$\langle \sum_{m,n} w(\ell - m, k - n) \langle f, g_{m,n} \rangle g_{m,n}, f \rangle \tag{1.30}$$

and denoting $\mathcal{I}(T_{\ell,k} w)(m, n) = w(\ell - m, k - n)$, we can see that the left slot in the inner product above is a special kind of multiplier, called *Gabor Multiplier* and we can write,

$$(\mathcal{S}_f * w)(\ell, k) = \langle M_{\mathcal{I}(T_{\ell,k} w), \mathcal{G}} f, f \rangle, \tag{1.31}$$

for every $(\ell, k) \in \mathbb{Z}$ with $\mathcal{G} = \{g_{m,n}\}_{m,n \in \mathbb{Z}}$. This representation is somewhat nice, since Gabor multipliers have many well-known properties which can be exploited [4, 14]. Furthermore, we can generalize this result to Bessel sequences in a Hilbert space, which include frames and Riesz bases. The related multipliers have been well studied [5, 33].

Theorem 1. *Let $\Psi = \{\psi_n\}_{n \in \mathcal{K}}$ be a Bessel sequence in a Hilbert space $\mathcal{H}$ and define $\mathfrak{T}_a^* : \mathcal{H} \to \ell^1(\mathbb{N})$ with $\mathfrak{T}_a^* f = \left\{|\langle f, \psi_n \rangle|^2\right\}_{n \in \mathcal{K}}$ as the magnitude square analysis operator. Let further be $w \in \ell^2(\mathbb{N})$, then the following holds.*

$$(\mathfrak{T}_a^* f * w)(\ell) = \langle M_{\mathcal{I}(T_\ell w), \Psi} f, f \rangle \tag{1.32}$$

for all $f \in \mathcal{H}$.

Proof. Since the analysis operator of a Bessel sequence is bounded, $\sum_{n\in\mathcal{K}} |\langle f, \psi_n\rangle|^2$ converges and so, for a $f \in \mathcal{H}$, using $|\langle f, \psi_n\rangle|^2 = \overline{\langle f, \psi_n\rangle}\langle \psi_n, f\rangle$ we can write

$$\begin{aligned}(\mathfrak{T}_a^* f * w)(\ell) &= \sum_{n\in\mathcal{K}} |\langle f, \psi_n\rangle|^2 w(\ell - n) \\ &= \langle \sum_{n\in\mathcal{K}} w(\ell - n)\langle f, \psi_n\rangle \psi_n, f\rangle = \langle M_{\mathcal{I}(T_\ell w),\Psi} f, f\rangle.\end{aligned} \tag{1.33}$$

□

This result is indeed nice, since we can make use of the theory of multipliers also w.r.t. more general signal expansions, like general shift-invariant systems, non-stationary Gabor frames, filter banks, etc. in this context. In particular, it is related to Section 2.2, where so-called *semi-discrete frames* are use for signal expansion.

1.2.3 A Link to Filter Banks

We already noted that most of the used time-frequency representations can be set up via a filter bank construction, where usually, a modulus $|\,.\,|$ or a modulus squared $|\,.\,|^2$ is applied on the computed coefficients. Also, some transformations use dimensionality reduction in time, e.g. subsampling (spectrogram) or scaling (scalogram). This workflow, i.e. filtering with intrinsic dimensionality reduction and applying a modulus as a non-linear function can be interpreted as computing one layer of a CNN with manually set, instead of learned filters. It points out the fundamental link between filter bank decompositions and the computational structure of a CNN.

Scattering Network - A Preview

At this point, something very obvious lies in the air, namely to combine time-frequency filtering with the structure of a neural network. Let us consider the construction of a filter bank in the framework of a CNN (1.27). We take one dimensional filters $w_k \in \mathbb{C}^{L_k}$ with k in some frequency index set $\mathcal{K}$, set $b = 0$ and keep the array of convolution outputs instead of summing them up. We may write this as operator $\mathrm{U}\, f := \{|f * w_k|\}_{k\in\mathcal{K}}$. A recursive application of U on its outputs defines a so-called *scattering network*. In principle it is a CNN with filters that are manually set to time-frequency filters with a modulus as non-linearity. Such a scattering network arises along the *scattering transform*, which considers paths along frequency-indices throughout the tree-like structure of this network. In the next chapter we introduce the scattering transform the way Mallat originally introduced it in [22].

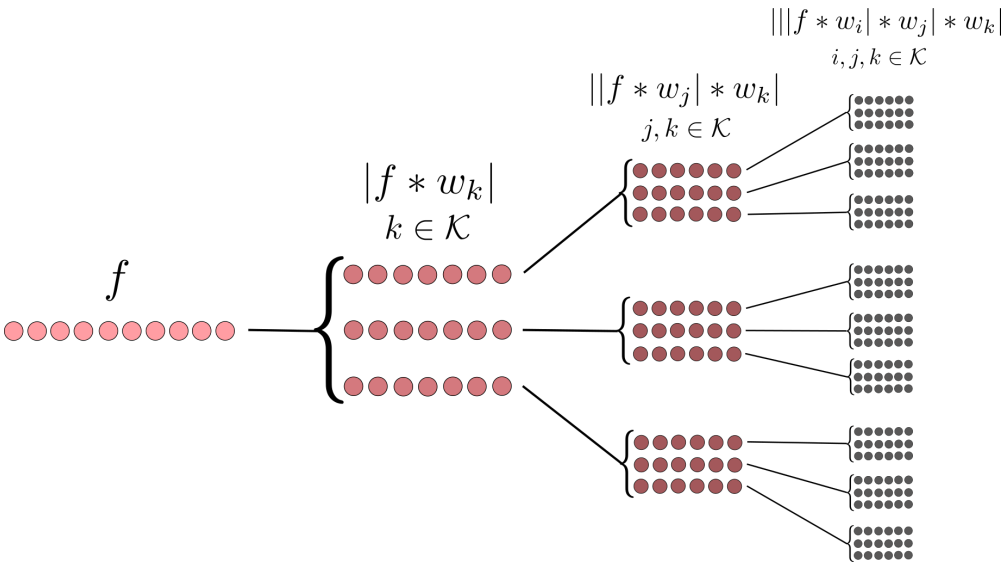

Figure 1.15: A schematic scattering network with 3 layers, based on a filter bank $\{w_k\}_{k\in\mathcal{K}}$, where $|\mathcal{K}| = 3$.

2 The Scattering Transform

The original motivation for the scattering transform was to set up a translation-invariant operator on $L^2(\mathbb{R})$, which is Lipschitz-continuous to the action of diffeomorphisms. In the case of audio this can be seen as a certain stability w.r.t deformations caused by time-warping. Why is this a nice thing for representing a signal?
Translation invariance is a natural property of an audio signal, of course, the information content is preserved when we shift it in time. Also small deformations of a signal do not affect the information content heavily. So, finding a representation of an audio signal, that depicts different features of the signal in such an invariant and stable way is a natural motivation. As we are in $L^2(\mathbb{R})$, we look for an operator with that properties. Such an operator Φ mapping from $L^2(\mathbb{R})$ to some Hilbert space $\mathcal{H}$ is called *translation-invariant* if

$$\Phi(T_a f) = \Phi(f) \quad \text{for all } f \in L^2(\mathbb{R}). \tag{2.1}$$

A time-warping deformation is defined as a translation by a differentiable function $\tau(t)$, i.e. $T_\tau f(t) = f(t - \tau(t))$ with $|\tau'(t)| < 1$. An operator Φ is stable relative to such a deformation if $\|\Phi(f) - \Phi(T_\tau f)\|_{\mathcal{H}}$ is small when the deformation is small, measured by $\sup|\tau'(t)|$. This can be expressed in terms of *Lipschitz-continuity* relative to τ, i.e. for every compact $\Omega \in \mathbb{R}$ there exists a $C > 0$ such that for all $f \in L^2(\mathbb{R})$ supported on Ω,

$$\|\Phi(f) - \Phi(T_\tau f)\|_{\mathcal{H}} \leq C\|f\| \sup_{t\in\Omega} |\tau'(t)| \tag{2.2}$$

holds. This implies that Φ is "almost invariant" to local translations by τ up to the first-order deformation term. So, taking only into account deformations by "small" diffeomorphisms close to translations, it is a nice thing to have for a representation.
An example of a translation-invariant operator is the modulus of the Fourier transform,

$$|\widehat{T_a f}(\omega)| = |e^{-2\pi i a\omega}\widehat{f}(\omega)| = |\widehat{f}(\omega)|. \tag{2.3}$$

The STFT localizes this invariance since $|\mathcal{V}_g T_a f(t,\omega)| \approx |\mathcal{V}_g f(t,\omega)|$ only holds for $|a| \ll \text{supp}(g)$, thus the windowing limits the invariance in time direction. However, both operators are not Lipschitz-continuous to time-warping, especially at high frequencies instabilities occur [2]. Mallat constructed an operator based on wavelet transforms, which provides the desired Lipschitz-continuity being additionally translation-invariant. For the application on audio as stable feature representation in the sense of a time-frequency representation this operator was modified by windowing, similar to the ideas for the STFT, such that the invariance is localized. It turns out that this new representation is able to capture time-frequency features beyond those contained in common time-frequency representations. Events such as note attacks, amplitude and frequency modulation, as well as chord structures can be captured by this transform [1].

2.1 Wavelet Scattering

A wavelet is a regular and localized waveform and thus stable to deformations, i.e. convolution with wavelets define operators that are Lipschitz-continuous to the action of diffeomorphisms [21]. But truly, a wavelet transform is not translation-invariant. However, simply integrating over the wavelet filter outputs forces the translation-invariance, i.e. if $\int Wf(t,\lambda)dt < \infty$, then the integral is translation-invariant. Unfortunately it is zero since the wavelet filters fulfil $\int \psi_{t,\lambda} dt = 0$. Also any linear transformation of Wf, that is translation-invariant will yield 0. To get a nonzero invariant, we have to use a nonlinear function that is Lipschitz-continuous to diffeomorphisms. The modulus function is a good candidate that fulfills the requirements and so, $\int |Wf(t,\lambda)|dt$ is translation-invariant. The modulus makes lower frequencies appear in Wf, caused by interferences [22]. These high frequencies can be recovered by computing another wavelet transform of $Wf(t,\lambda)$ for each λ. To again obtain translation-invariant coefficients we repeat the procedure, i.e. we apply a modulus and integrate: $\int |W(Wf)(t,\lambda,\mu)|dt$. In a general manner we can define a wavelet scattering propagator that extends this decomposition.

Definition 9 (Wavelet Scattering)**.** *Let $p = (\lambda_1, \ldots, \lambda_m) \in \Lambda^m$ be a path of frequency-indices and $\{\psi_{\lambda_k}\}_{\lambda_k \in \Lambda, k=1,\ldots,m}$ wavelet filters. Then we define*

$$\mathcal{U}[p]f = ||...||f * \psi_{\lambda_1}| * \psi_{\lambda_2}| * ...| * \psi_{\lambda_m}| \tag{2.4}$$

as m-th order wavelet scattering coefficients or path-ordered scattering coefficients.

This is well-defined on $L^2(\mathbb{R})$ since $\|\mathcal{U}[p]f\| \leq \Pi_{i=1,..,m}\|\psi_{\lambda_i}\|_1\|f\|_2$ for all $p \in P_\infty$, the set of all finite paths. If we now consider the construction above for every $p \in P_\infty$, a layered network structure, similar to (1.27) emerges, as we prospected already in Figure 1.15. In that sense we call $\mathcal{U}[q,\lambda]f$ with $\lambda \in \Lambda$ and $q = (\lambda_1, \ldots, \lambda_{m-1})$ the m-th *Wavelet Scattering Layer* w.r.t. q and write it as $\mathcal{U}[q,\cdot\,]f$.

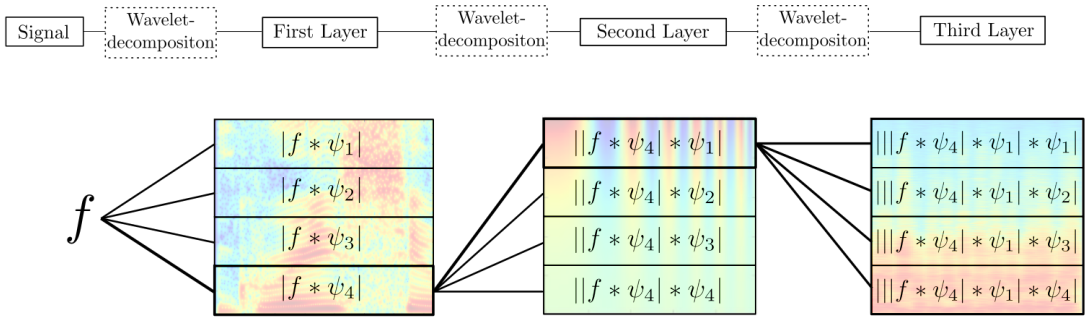

Figure 2.1: Scheme of computing 3rd order wavelet scattering layer w.r.t. the path $[4,1]$, i.e. $\mathcal{U}[4,1,\cdot\,]f$.

The translation-invariant scattering transform is then computed by integration and normalization of the scattering coefficients.

Definition 10 (Wavelet Scattering Transform)**.** *Let $p \in P_\infty$ and $f \in L^1(\mathbb{R})$, then the wavelet scattering transform is defined by*

$$\overline{\mathcal{S}} f[p] = \frac{1}{\mu_p} \int (\mathcal{U}[p]f)(t)dt, \tag{2.5}$$

with a nonvanishing normalization factor $\mu_p = \int (\mathcal{U}[p]\delta)(t)dt$ (response of a Dirac).

$\overline{\mathcal{S}}$ is a translation-invariant operator on $L^1(\mathbb{R})$, that is Lipschitz-continuous to the actions of diffeomophisms. Going more into details on this would go beyond the scope of this thesis, please see [22] for more.

To use it as a representation for audio in practice, we will motivate a time-localized windowed version of $\overline{\mathcal{S}}$ by the construction of the *mel-frequency spectrogram*.

2.1.1 Windowed Scattering Transform

The classic spectrogram has a linear frequency scale which is not in correspondence to the human perception of pitch, especially in higher frequency range. Therefore a scale was invented, along which pitches are perceived to be equally spaced from each other, the so-called *mel-scale* (from *mel*ody). It can be calculated from Hertz to mel according to $\Lambda(\omega) = 2595 * \log_{10}(1 + \omega/700)$ [32]. The mel-frequency spectrogram M_g can be constructed from the spectrogram by taking weighted averages over frequency channels according to this scale. In [2] this is done using CQT filters: Let $\{\widehat{\psi}_\lambda\}_{\lambda \in \Lambda_{mel}}$ denote the mel-scale filters with center frequencies λ respectively, then

$$M_g f(t, \lambda) = \int |\mathcal{V}_g f(t, \omega)|^2 \, |\widehat{\psi}_\lambda(\omega)|^2 dw. \tag{2.6}$$

At high frequencies, $\text{supp}(\widehat{\psi}_\lambda)$ is of the order λ/Q, at low frequencies we keep a linear scale with a bandwidth equal to the one of $\widehat{g}$. This shall smooth the spectrum with an emphasis on perceptually meaningful frequencies. Unlike the spectrogram, the mel-frequency spectrogram satisfies the Lipschitz deformation stability condition (2.2) and thus, provides time-warping stability [2]. We will see that the frequency averaging is in fact equivalent to time averaging of the outputs of the filters ψ_λ. Using this locally stabilizing technique instead of integration as in (2.5) will yield an audio-motivated version of the scattering transform in form of a time-frequency representation. In the following we derive the proposed "equivalence" in a rather heuristic way.
We shall assume the window g to be smooth and note that $\mathcal{V}_g f(t, \omega)$ is the Fourier transform of $f(\tau)g(\tau - t)$ at ω. Using Plancherel's formula, we get

$$\begin{aligned} M_g f(t, \lambda) &= \int |\mathcal{V}_g f(t, \omega)|^2 \, |\widehat{\psi}_\lambda(\omega)|^2 dw = \int |(f(\tau)g(\tau - t)) * \psi_\lambda(v)|^2 dv \\ &= \int \Big| \int f(\tau)g(\tau - t)\psi_\lambda(v - \tau)d\tau \Big|^2 dv. \end{aligned} \tag{2.7}$$

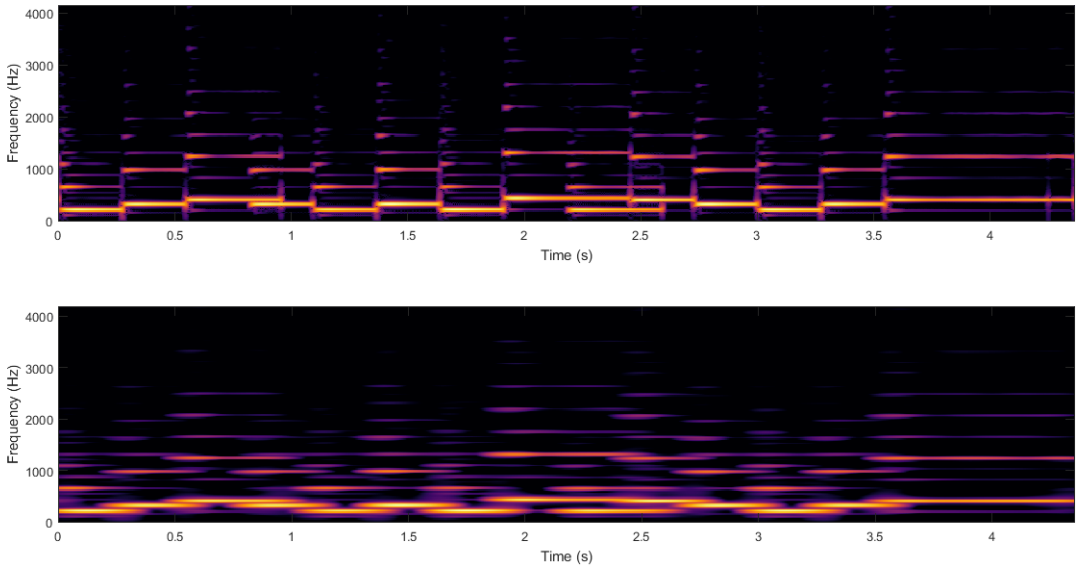

Figure 2.2: (Top) A normal spectrogram of the synthesizer-version of the melody. (Bottom) A time-averaged spectrogram of the same signal, mimicking the mel-spectrogram using a window with a length of 23ms. It is the first layer of the windowed scattering transform.

For λ, where the essential support of $\widehat{\psi_\lambda}$ is much larger as $\widehat{g}$, our smooth g is essentially constant on the support of ψ_λ. Hence, we can estimate $g(\tau - t)\psi_\lambda(v-\tau) \approx g(v-t)\psi_\lambda(v-\tau)$ and so

$$M_g f(t,\lambda) \approx \int \Big| \int f(\tau)\psi_\lambda(v-\tau)d\tau \Big|^2 \, |g(v-t)|^2 dv = |f * \psi_\lambda|^2 * |g|^2(t). \tag{2.8}$$

Thus, frequency averaging of the spectrogram seems to have the same effect as filtering $|f * \psi_\lambda|^2$ by $|g|^2$, which performs an averaging over time, resulting in a low-pass filtering. The filtering removes information, especially in the high frequency regions. To balance this information loss, the mel-spectrogram is therefore often computed using very short windows ($\leq$ 25ms). Thus, wide-scale structures cannot be captured which yet, inherit key features for many application. To use wider windows, but compensate for the removed high frequencies in $|f * \psi_\lambda|^2$, we apply the idea we derived in the last section, namely performing another subsequent wavelet decomposition $|f * \psi_\lambda|^2 * \psi_\mu$ for all λ. To again gain stability of the coefficients, we take the modulus and average in time using g and get $\big||f * \psi_\lambda|^2 * \psi_\mu\big| * |g|$. Cascading this procedure defines the windowed scattering transform. In this particular setting, let us rewrite our wavelet transform W by computing, on one hand, a convolution with a low-pass filter φ corresponding to the lowest band-width (base-bin) and on the other hand convolutions with all higher-frequency wavelet filters $\psi_\lambda, \lambda \in \Lambda$ (The intention behind this is to let φ do the time averaging as atom, associated

to the wavelet transform.):

$$W_\varphi f = \Big(f * \varphi, f * \psi_\lambda\Big)_{\lambda\in\Lambda} \tag{2.9}$$

If we set the filters to fulfill

$$1 - \alpha \leq |\hat{\varphi}(\omega)|^2 + \frac{1}{2}\sum_{\lambda\in\Lambda}\Big(|\hat{\psi}_\lambda(\omega)|^2 + |\hat{\psi}_\lambda(-\omega)|^2\Big) \leq 1 \tag{2.10}$$

for all $\omega \in \mathbb{R}$ and $\alpha < 1$, then this wavelet decomposition provides a covering of the whole time-frequency plane and thus, implies stablity and invertibility of W_φ [22].

Definition 11 (Windowed Scattering Transform)**.** *Let $p = (\lambda_1, ..., \lambda_m) \in \Lambda^m$ be a path of frequency-indices of order m, $\mathcal{U}[p]f$ the m-th order wavelet scattering coefficients with respect to W_φ, where φ is the base-bin filter. Then we can define*

$$\mathcal{S}[p]f = \mathcal{U}[p]f * \varphi \tag{2.11}$$

as the Windowed Scattering Transform w.r.t. $\mathcal{U}[p]f$.

We state the most crucial properties of the windowed scattering transform and refer to [2] for the details.

(i) Defining $Uf := (f * \varphi, |f * \psi_\lambda|)_{\lambda\in\Lambda}$ as the wavelet modulus operator, then with the stability condition (2.10), this operator can be shown to be contractive, i.e. $\|Uf\| \leq \|f\|$ and since $\mathcal{S}$ is a repeated application of U, $\mathcal{S}$ is also contractive. This implies its Lipschitz-continuity relative to the action of small diffeomorphisms close to translations, i.e. $\mathcal{S}$ is stable to time-warping deformations w.r.t. an Euclidian norm defined by $\|\mathcal{S}f\|^2 = \|\{\mathcal{S}[p]f\}_{p\in P_\infty}\|^2 = \sum_{p\in P}\|\mathcal{S}[p]f\|^2$

(ii) f can be reconstructed from $\mathcal{S}f$ with "acceptable" quality by inverting U.

(iii) It can be shown that most of the input signals energy is contained in the output of the first two layers.

The scale decomposition by a wavelet transform accesses different time scales, it captures longer dependencies in high scales and finer structures in the lower scales. Applying a wavelet decomposition on the filtered signal at different scales will analyze the corresponding structure widths and thus, expand the captured dependencies even further, similar to the effect of pooling. This allows to access coarser structures of a signal which are extracted in a stable way by time averaging, pushed towads translation-invariance of these structures. In the following, we derive what the second layer reveals for two special cases of signal properies. To do this we take the dervations found in [1] and adopt them to sinusoidal models.

2.1.2 Amplitude Modulation

Let $f(t) = a(t)e^{2\pi i\xi_1 t}$ be a complex sinusoid with a fundamental frequency of ξ_1. $a(t) \geq 0$ for all $t \in \mathbb{R}$ denotes an amplitude modulation function. We shall assume it to be slowly varying, i.e. $|a'(t)| \ll 1$, in particular we want to assume it to be approximately constant on $\text{supp}(\psi_{\lambda_1})$, where $\lambda_1 = \text{argmin}_{\lambda\in\Lambda} |\xi_1 - \lambda|$. Doing the very rough estimation $|f * \psi_{\lambda_1}|(t) \approx |e^{2\pi i\xi_1 t} * \psi_{\lambda_1}|(t) \cdot a(t)$, we can deduce the following for the first and second order scattering coefficients.

$$\mathcal{S}[\lambda_1]f(t) \approx |\widehat{\psi}_{\lambda_1}(\xi_1)| \cdot (a * \varphi)(t) \tag{2.12}$$

and

$$\mathcal{S}[\lambda_1, \lambda_2]f(t) \approx |\widehat{\psi}_{\lambda_1}(\xi_1)| \cdot \mathcal{S}[\lambda_2]a(t). \tag{2.13}$$

From (2.12) it is obvious that first order coefficients describe the pitch structure of f, where the fine structure of a is averaged out. Considering $\mathcal{S}[\lambda_2]a = |a * \psi_{\lambda_2}| * \varphi$, we see that second order coefficients are proportional to the scalogram of the amplitude modulation, averaged in time, which reveals the shape of a.

2.1.3 Frequency Modulation

A special case of a frequency modulation is a vibrato, which is created by applying a periodic deformation to a source signal. Using the complex sinusoidal this writes as $f(t) = e^{2\pi i\xi_1(t-\epsilon\cos(\eta t))}$. To again perform an estimation of the first order coefficients like in (2.12), we use the *instantaneous frequency* of the modulated signal, which describes the signals frequency as a function of time. We will introduce it in detail later in the thesis. For now this shall serve as demonstration. With $(1 + \epsilon\eta\sin(\eta t))$ as instantaneous frequency of f, we can deduce

$$\mathcal{S}[\lambda_1]f(t) \approx \big(|\widehat{\psi}_{\lambda_1}\big(\xi_1(1 + \epsilon\eta\sin(\eta t))\big)| * \varphi\big)(t), \tag{2.14}$$

Thus, as the modulation frequency η in the sine arises in the first order coefficients, it will be reproduced by second order coefficients $\mathcal{S}[\lambda_1, \lambda_2]f(t) \approx \big(||\widehat{\psi}_{\lambda_1}\big(\xi_1(1 + \epsilon\eta\sin(\eta t))\big)| * \widehat{\psi}_{\lambda_2}| * \varphi\big)(t)$.

These examples show that indeed, as we go deeper into the scattering network, coarser structures of the signal are captured. In the amplitude modulated signal it was the envelope that appeared, i.e. how the sound changes over time. In the vibrato signal we got the frequency of the modulation depicted. Amplitude and frequency modulations have a great influence on the timbre of a sound, its sound color. This means indeed, scattering reveals features of a sound, that a common time-frequency representation is not capable to extract. We will illustrate the proposed on our melody, based on another time-frequency decomposition.

2.2 Scattering Based on Semi-Discrete Frames

In [35] the theory of scattering was extended to more general signal transformations based on *semi-discrete frames*, allowing also to use different transformations in every layer, while keeping translation-invariance and deformation stability.

Definition 12. *Let $\{f_\lambda\}_{\lambda\in\Lambda} \subseteq L^1(\mathbb{R})\cap L^2(\mathbb{R})$ be a set of functions indexed by a countable set Λ. The family of translated and involved functions*

$$\Psi_\Lambda = \{T_b\mathcal{I}f_\lambda\}_{(\lambda,b)\in\Lambda\times\mathbb{R}} \tag{2.15}$$

is called a semi-discrete frame, if there are positive constants $A \leq B$, such that

$$A\|f\|_2^2 \leq \sum_{\lambda\in\Lambda} \|f * f_\lambda\|_2^2 \leq B\|f\|_2^2 \tag{2.16}$$

for all $f \in L^2(\mathbb{R})$. The functions $\{f_\lambda\}_{\lambda\in\Lambda}$ are called atoms of the semi-discrete frame Ψ_Λ.

Note that $\sum_{\lambda\in\Lambda} \|f*f_\lambda\|_2^2 = \sum_{\lambda\in\Lambda} \int |f*f_\lambda|^2 dt$, i.e. the frame condition in (2.16) involves a discrete *and* a continuous setting for t and λ respectively. The considered family can be thought as shift-invariant frames with an index set Λ, that typically labels a collection of scales, directions or frequency shifts and thus, constitutes a certain geometrical structure. Examples for structured semi-discrete frames are Gabor frames [18], curvelets [10], shearlets [20], rigidlets [9] and of course, wavelets [21].
The definition of the corresponding scattering transform is somewhat identical to (9) and (11), only that indexing is more complicated, so we won't state it here explicitely and refer to [35]. There also the proofs for translation-invariance and deformation stability can be found.

2.3 Gabor Scattering

We will have a look at a scattering procedure w.r.t. specific structured semi-discrete frames which we are already familiar with, namely Gabor frames. An extensive analysis has been done for those in [8]. The definition can be formulated in a layer-wise manner:

Definition 13 (Gabor Scattering Coefficients)**.** *Let $\mathcal{G}_\ell := \{M_{mb_\ell}T_{na_\ell}g_\ell\}_{(m,n)\in\mathbb{Z}^2}$ be Gabor frames, then we compute the Gabor scattering operation recursively by*

$$\mathcal{U}_\ell[mb_k]f_{\ell-1}(n) = f_\ell = |\langle f_{\ell-1}, M_{mb_\ell}T_{na_\ell}g_\ell\rangle|,\ n\in\mathbb{Z}. \tag{2.17}$$

With $f_0 = f$. Denoting $q = (q_1, ..., q_\ell) \in b_1\mathbb{Z}\times ...\times b_\ell\mathbb{Z}$ as frequency-index path, then we define

$$\mathcal{U}[q]f = \mathcal{U}_\ell[q_\ell]\ldots\mathcal{U}_1[q_1]f \tag{2.18}$$

as the ℓ-th order Gabor Scattering Coefficients.

Analog to the windowed wavelet scattering transform we use an output generating atom φ from the Gabor system.

Definition 14 (Windowed Gabor Scattering Transform)**.** *Let $q = (q_1, ..., q_\ell) \in b_1\mathbb{Z} \times ... \times b_\ell\mathbb{Z}$ be a path, $\mathcal{U}[q]f$ the ℓ-th order scattering coefficients and $\varphi = g_\ell$, the Gabor atom corresponding to the base-frequency in $\mathcal{G}_\ell$, then*

$$\mathcal{S}[q]f = \mathcal{U}[q]f * \varphi \tag{2.19}$$

is the Windowed Gabor Transform w.r.t. $\mathcal{U}[q]f$.

The principle is exactely the same as for the wavelet scattering transform, only that here the time-sampling parameters a_k play the role of giving access to coarser structures. Also in the same manner we call $\mathcal{S}[p, b_\ell m]f$ considered for all $m \in \mathbb{Z}$ and a frequency-index path of length $\ell - 1$, the *ℓ-th Gabor Scattering Layer* w.r.t. p and write $\mathcal{S}[p, \cdot\,]f$. We turn to applications again.

2.3.1 Modulation of Tones

In [8] the authors explain, how Gabor scattering sepatares structures of signals, modelled by the signal space of "harmonic tones"

$$\mathcal{T} = \Big\{ \sum_{\ell=1}^{N} a_\ell(t) e^{2\pi i \ell \xi_1 t} | a_n \in \mathcal{C}_c^\infty(\mathbb{R}) \Big\},$$

where $\mathcal{C}_c^\infty(\mathbb{R})$ denotes the space of smooth functions (infinitely often differentiable) with compact support over $\mathbb{R}$. A $f \in \mathcal{T}$ consists of a complex sinusoid at fundamental frequency ξ_1 with the corresponding harmonics $\ell\xi_1$ and shaping envelopes $a_\ell(t) \geq 0$ for $t \in \mathbb{R}$ and $1 \leq \ell \leq N$, which model a specific timbre. In fact, the amplitude modulation case we had in 2.1.2 was a special case of this model with $N = 1$. Denoting $\ell_1 = \operatorname{argmin}_\ell |\ell\xi_1 - mb|$ and φ as output generating atom, the following estimates for first and second order can be deduced:

$$\mathcal{S}[mb_1]f(n) = |\widehat{g}(mb_1 - \ell_1\xi_1)| \cdot (a_{\ell_1} * \varphi)(n) + \epsilon_1(n) \tag{2.20}$$

and

$$\mathcal{S}[mb_1, kb_2]f(n) = |\widehat{g}(mb_1 - \ell_1\xi_1)| \cdot \mathcal{S}[kb_2]a_{\ell_1}(n) + \epsilon_2(n) \tag{2.21}$$

with ϵ_1, ϵ_2 being essentially bounded under specific conditions on g_1, g_2 and a_ℓ, the details can be found in [8]. The equations above look very similar to (2.12) and (2.13). In (2.20) we see that the contributions of the frequencies near the fundamental frequency ξ_1 are present, while the fine structure of a_{ℓ_1} is averaged out. Concerning (2.21), $\mathcal{S}[kb_2]a_{\ell_1} = |\langle a_{\ell_1}, M_{kb_2}T_{na_2}g_2\rangle| * \varphi$ captures the Gabor coefficients w.r.t. the envelope a_{ℓ_1}. This information is located in the low frequency range, since we assume it to be varying slowly. If a_ℓ is modulated, it can be discerned in the second layer.

To illustrate the findings, we let a digital synthesizer play the melody, where one of the notes got a tremolo effect. A tremolo is a periodic amplitude modulation of a tone, i.e. in our harmonic tone setting we would write $f(t) = \sum_{\ell=1}^{N} \cos(2\pi\eta_1 t)e^{2\pi i\ell\xi_1 t}$, where we, strictly speaking, have to restict the cosine to a compact subset of $\mathbb{R}$ to make it a member of $\mathcal{T}$. It makes sense to assume $\eta_1 \ll \xi_1$ to have a clear distinction between amplitude and frequency, in our case we used $\eta_1 = 10$Hz and $\xi_1 = 880$Hz. Figure 2.3 shows how the second Gabor scattering layer of our amplitude modulated melody looks like. We also provide a frequency modulated version of the melody, where the digital synthesizer added a vibrato effect in 10Hz on the same note. (Figure 2.4). In both figures, the frequency of the modulation is clearly discernible in the second scattering layer. In the next section we let the scattering transform extract even coarser structures of a signal.

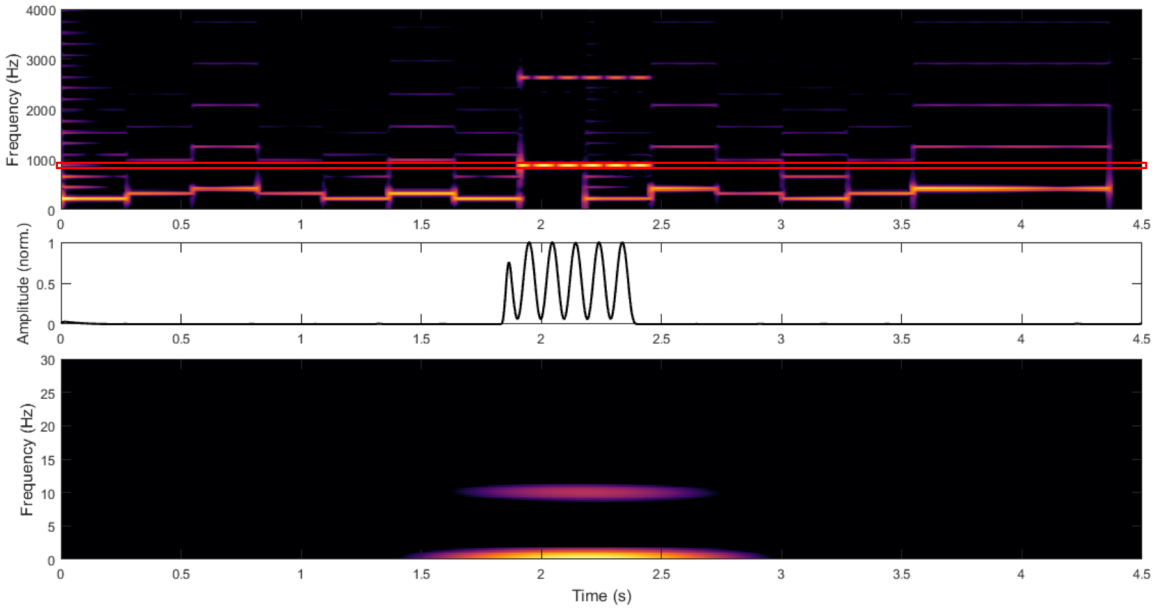

Figure 2.3: (Top) Gabor magnitudes of the tremolo modulated melody, $\mathcal{U}[\,\cdot\,]f$ (`dgtreal` with $a_1 = 100, M_1 = 2048$ and a Hann window of length 2048).
(Mid) $\mathcal{U}[880]f$, the filter output w.r.t. the frequency bin at 880Hz (red frame), which we choose to compute the second layer.
(Bottom) Second order Gabor scattering layer, $\mathcal{S}[880,\cdot\,]f$ (`dgtreal` with $a_2 = 1, M_2 = 1760$ and a Hann window of length 550). One clearly sees how the frequency of the modulation is captured.

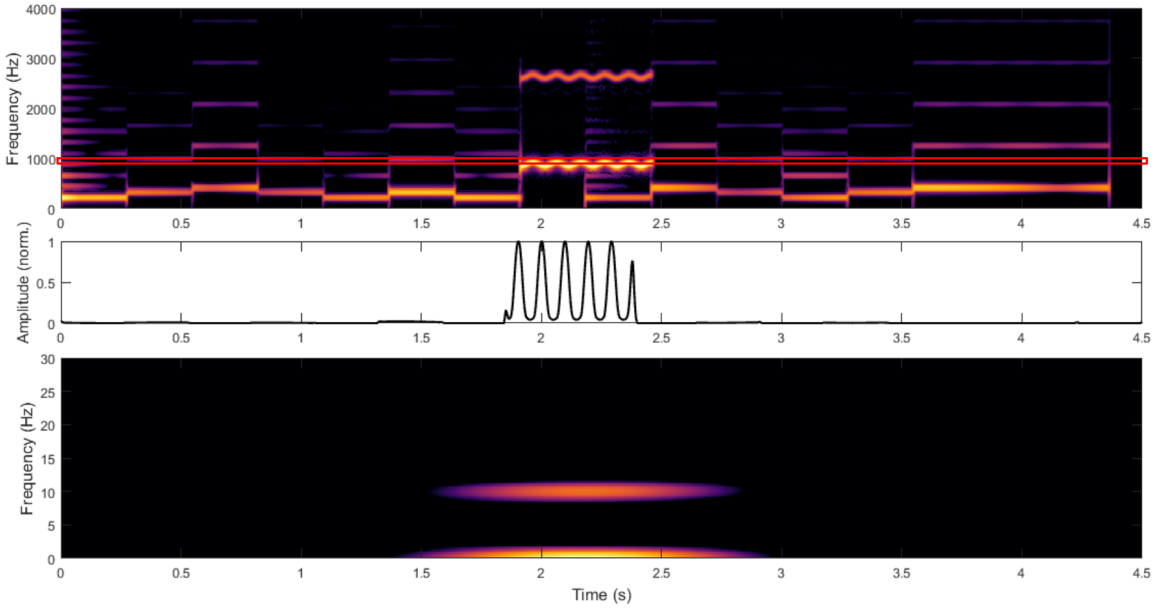

Figure 2.4: (Top) Gabor magnitudes of the vibrato modulated melody, $\mathcal{U}[\,\cdot\,]f$.
(Mid) $\mathcal{U}[880]f$, the filter output w.r.t. the frequency bin at 880Hz (red frame), which we choose to compute the second layer.
(Bottom) Second order Gabor scattering layer, $\mathcal{S}[880, \cdot\,]f$. One clearly sees how the frequency of the modulation is captured.
We used the same `dgtreal`-settings as for Figure 2.3

2.3.2 Rhythmical Features

Also wider stuctures of musical signal can be captured via a stronger dimensionality reduction, i.e. a coarser time-subsampling. This allows to access the scale where rhythmical features live, referring to different tempo levels within the signal. *Rhythm* refers to the timing of events within a musical piece in different levels of periodicity, where the temporal range of these usually lies within seconds. We shall define the *tempo* of a musical piece as the speed of the most prominent rhythm pattern. It is usually measured in bpm (beats per minute) and reaches among different genres and styles of music from 30-300bpm. Embodying the tempo as a periodic pattern of events in time we could also assign a frequency to it, i.e. $0.5 - 5$Hz. As a frequency, this is clearly not audible, but indeed perceivable as a rhythmical pattern. We will see that the scattering transform is also capable of "perceiveing" a rhythmical pattern by depicting its (subsampled) frequency in the second layer. We will demonstrate this on the most simple embodiment of tempo, a *metronome* modelled by an impulse train.

Lemma 1. *Let e be an impulse train with fundamental frequency ξ_0, i.e.*

$$e(t) = \sum_{\ell \in \mathbb{Z}} \delta_{2\pi\ell/\xi_0}(t). \tag{2.22}$$

Let further $\mathcal{G}_1 = \{M_{mb_1} T_{na_1} g_1\}$ be a Gabor frame with a symmetric window fulfilling $\operatorname{supp}(g_1) \leq 2\pi/\xi_0$, *then*

$$\mathcal{U}[mb_1]e(n) = \sum_{\ell \in \mathbb{Z}} |g_1(na_1 - 2\pi\ell/\xi_0)| \tag{2.23}$$

for all $m \in \mathbb{Z}$.

Proof. Note that $e(t) = \begin{cases} 1 & t = 2\pi\ell/\xi_0,\ \ell \in \mathbb{Z} \\ 0 & else \end{cases}$, then

$$\begin{aligned} \langle e, M_{mb_1} T_{na_1} g_1 \rangle &= \int e(t) M_{mb_1} g_1(t - na_1) dt \\ &= \sum_{\ell \in \mathbb{Z}} M_{mb_1} g_1(2\pi\ell/\xi_0 - na_1) \end{aligned} \tag{2.24}$$

Since we chose $\operatorname{supp}(g_1) \leq 2\pi/\xi_0$, the windows that compute nonzero coefficients do not overlap. Taking moduli coefficientwise removes the complex exponential terms induced by M_{mb_1} and we get the required form of $\mathcal{U}[mb_1]e(t)$. □

Equation (2.23) shows a smoothed and subsampled version of the metronome with a frequency of $a_1\xi_0$, instead of ξ_0. If we choose a_1 sufficiently large, we obtain a signal with a frequency that lies in a more accessible frequency region for another subsequent Gabor transform. The second layer $\mathcal{S}[k, mb_2]e$ will thus show a constant frequency of $a_1\xi_0$. (Figure 2.5). Thus, by sampling the time-frequency representation in wide time steps we obtain downscaled signals in the filter bins, which are interpreted as having a higher frequency by subsequent filters. This enables to capture wider scales; the larger a_1 is chosen, the wider the scale in focus.

To consider more complex musical signals, we may see them as consisting of tonal, transient and stochastic components. Truly, the transient parts of a signal indicate its rhytmical structure, e.g. by percussive elements like drums, note-onsets, etc. In that manner, the scattering transform detects periodic patterns among the transient arrangements and extracts the temporal information with respect to those [19]. In the case of the guitar version of our melody, it is the onset transients of the single picked notes, that indicate the tempo (Figure 2.6). Here, instead of picking a single frequency-index, we compute the second layer with an average over the subband signals of the frequency regions, where transients are more dominant.

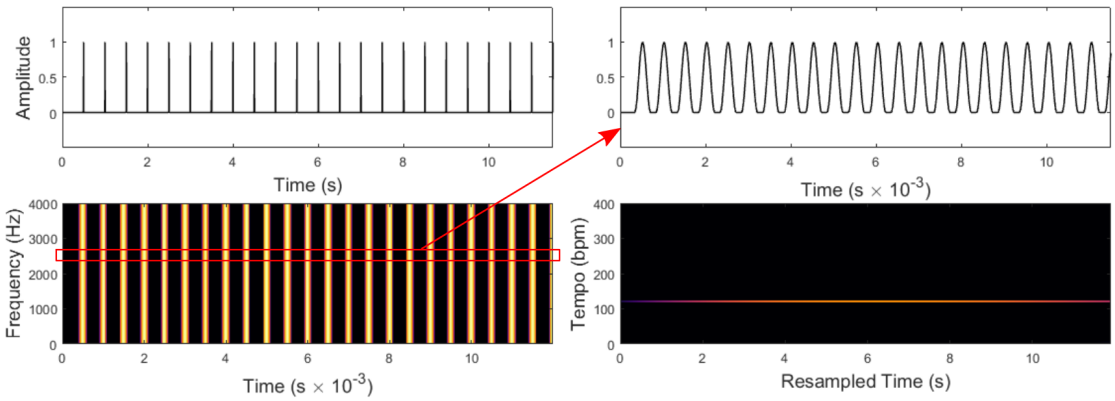

Figure 2.5: (Left) Impulse train in the tempo of 120bpm, i.e. 2Hz and below its Gabor magnitudes, using a time hop size of $a_1 = 10^3$. Note that this is plotted here using a scaled timeaxis in seconds$\times 10^{-3}$!
(Right) The single filter outputs: the impulse train is smoothed by the window and scaled in time by the factor a_1, i.e. absolutely seen it is shorter and has a higher frequency, 2000Hz. Below is then its Gabor magnitudes with $a_2 = 1$ on a resampled timescale and a "tempo"scale measured in bpm. It clearly depicts the tempo of the original signal train, 120bpm.

Towards New Ideas

We have motivated the scattering transform, ilucidated it from different points of views and stated its main properties as mathematical operator. Considering it as feature representator in the context of audio, we have shown some applicational examples of the transform, extracting specific properties of our melody. This should have demonstrated the potential of the ideas behind this transform - expansion of the filter range, translation invariance, stability - which we initially assumed to be fundamental ideas in signal analysis and machine learning. There is much ongoing research and application around the scattering transform, paving it a promising future, see e.g. [3]. Also we want to tiptoe towards new ideas regarding scattering. Our approach is very fundamental and theoretic and will combine the idea of scattering with concepts in signal processing related to the phase information of a signal. We will start with an extensive discussion on the notion of phase to get a feeling on how to work with it.

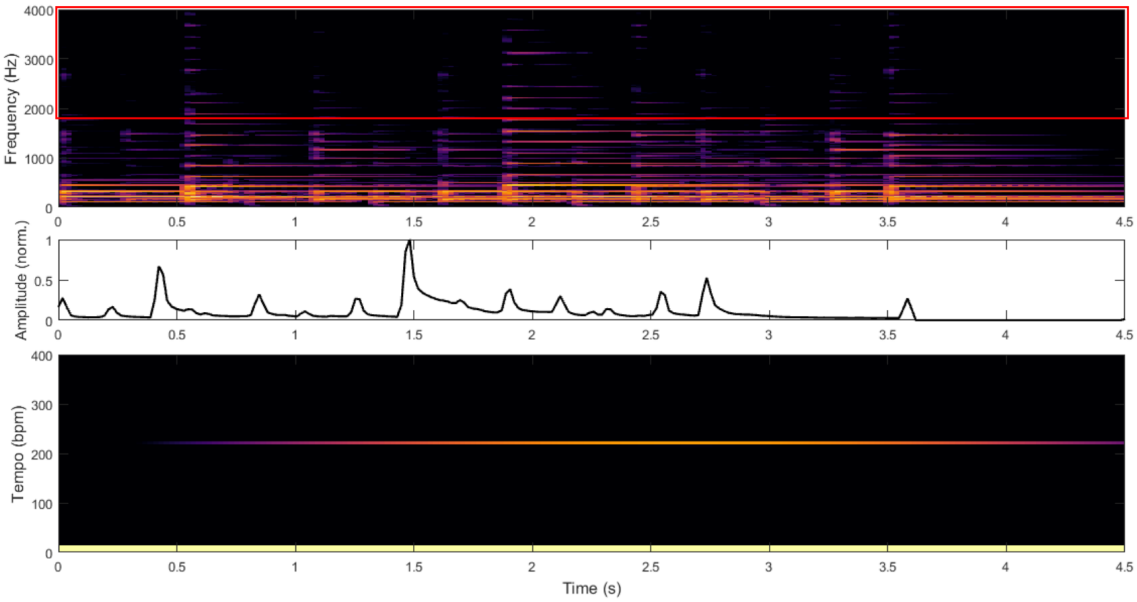

Figure 2.6: (Top) Magnitude of Gabor transform (`dgtreal` with $a_1 = 1000, M_1 = 2048$ and a Hann window of length 16384) of the guitar playing the melody.
(Mid) Signal corresponding to the average of the filter outputs in the red frame.
(Bottom) The second scattering layer w.r.t an average over channels corresponding to frequency regions where transients are most dominant (red frame), here $1800 - 10300$Hz, computed with a Gabor transform (`dgtreal` with $a_2 = 1, M_2 = 440$ and a Hann window of length 880)

3 Aspects of the Phase

So far we have talked about time-frequency representations that are constructed by considering the magnitude of the obtained complex coefficients via applying a modulus. Thinking of the polar representation of a complex number $z = |z|e^{i \arg z}$, applying a modulus on z removes the information contained in $\arg z$, i.e. the argument, or *phase angle* is discarded. However, the phase of a complex number is crucial, it determines the direction of its position in the complex plane and is therefore not less important than the magnitude. It can be shown that under certain conditions on the redundancy of the time-frequency decomposition, a signal can be reconstructed completely (up to a scale factor of the amplitude) only by its phase [29, 26]. Nevertheless, working with phase information is not unproblematic, as we will see. Therefore we start from zero and try to get a feeling for the phase and related concepts and how to work with them.

Since $\arg z$ denotes any real number fulfilling the relation $z = |z|e^{i \arg z}$, which are infinitely many, we consider the principle value of $\arg z$, denoted by $\operatorname{Arg} z = \arg z|_{(-\pi,\pi]}$. This makes the representation very peaky and unsmooth. To solve this problem, one can "unwrap" the phase by adding 2π at every jump form π to $-\pi$, which makes it a monotonically increasing function in time. (Clearly, we can unwrap along both, time and frequency axis.) As demonstration we let MATLAB play our melody using pure sinusoids wich yields clearer plots than for recordings. Figure 3.1 shows the phase angles of the coefficients obtained by a Gabor transform of this version of the melody, in wrapped and unwrapped form. Still, the representation is not quite informative. To derive a useful phase-related concept, that depicts time-frequency information well, we may change our point of view. In general the phase of a signal can be viewed from different angles. As we talked about the phase of a complex number, we can also talk about the phase of a waveform. Let us consider a standard sinusoid,

$$f(t) = a \cdot \cos(2\pi\xi_0 t + \phi_0). \tag{3.1}$$

This signal is characterized by three ingredients: The amplitude $a \geq 0$ determines how strong the signal is, the frequency ξ_0 defines the number of wave cycles within a full circle and the phase-offset ϕ_0 sets the start of the signal. The whole argument of of the cosine is called the *temporal phase*, $\phi(t)$ of the signal. It gives information about the temporal positioning. Deriving how fast this positioning changes, we can get information about how the waveform evolves. With this point of view we will introduce the concepts of *instantaneous frequency* and *group delay*, giving "easier-to-work-with" quantities that overcome some of the issues of the (pure) phase.

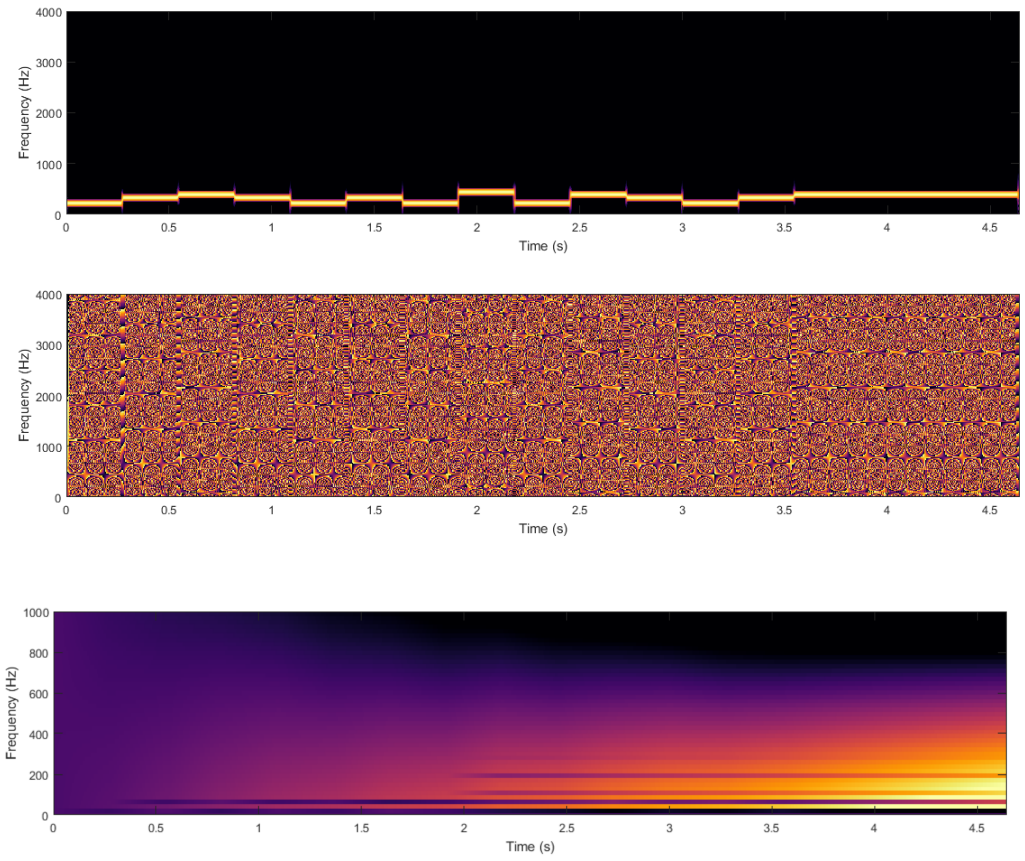

Figure 3.1: (Top) The magnitudes of the coefficients obtained by the Gabor transform using `dgt` of the melody played by MATLAB with pure sines.
(Mid) The principal values of the phase angles w.r.t. complex Gabor coefficients computed with MATLABs `angle`.
(Bottom) Unwrapped phases in time-direction computed with MATLABs `unwrap`.

3.1 Instantaneous Frequency and Group Delay

We will put a different light on the concept of frequency, built on the notion of phase.

Instantaneous Frequency

The notion of frequency loses its effectiveness when the signal is not stationary anymore, i.e. when the frequency varies over time. To describe this in a meaningful way we need to consider a frequency for every time instant. We do this with the signals phase. Having $\phi(t) = 2\pi\xi_0 t + \phi_0$, we find the angular frequency $2\pi\xi_0$ by taking the time-derivative of $\phi(t)$. Generalizing the notion in (3.1) to real signals with modulated amplitude and a differentiable time varying phase function, we can write such signals as

$$f(t) = a(t)\cos\phi(t) \tag{3.2}$$

with $a(t) \geq 0$ and call

$$\omega(t) = \frac{1}{2\pi}\frac{d}{dt}\phi(t) \tag{3.3}$$

its *instantaneous frequency* (IF). (The analytic version for (3.2) would be $f_a = a(t)e^{i\phi(t)}$, where $a(t) = |f_a(t)|$.) Now, amplitude and phase characterize our signal. The phase tells us the time direction appearance, from which we can derive the frequency information. As examples let us consider two signal, a stationary sinusoid, $s_1(t) = \cos(2\pi\xi_1 t)$ and a non-stationary quadratic chirp, which is a sinusoid with quadratically increasing frequency, $s_2(t) = \cos(2\pi(\xi_2 + \frac{k}{3}t^2)t)$. The corresponding IFs $\omega_1(t) = \xi_1$ and $\omega_2(t) = \xi_2 + kt^2$ are depicted in Figure 3.2.

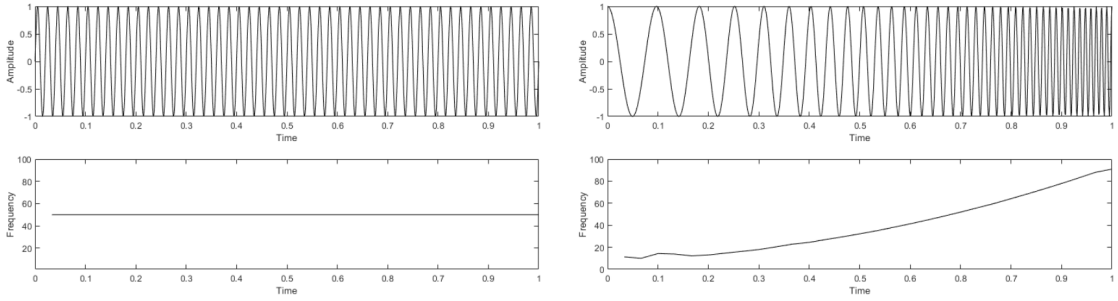

Figure 3.2: (Left) Sine wave with a frequency of 50Hz. Below, the corresponding instantaneous frequency using MATLABs `instfreq`.
(Right) Quardratic chirp, starting at 10Hz, sweeping to 100Hz. Below, again the corresponding instantaneous frequency.

As we have a concept for describing frequency per time instant, we can do the same in the spectral domain to get a possibility to describe the localization of the frequency content in the time domain. This is conceptually analog to the IF.

Group Delay

The spectrum of a signal f, real or complex, is a complex signal

$$\widehat{f}(\omega) = A(\omega)e^{i\theta(\omega)}, \tag{3.4}$$

where $A(\omega) = |\widehat{f}(\omega)|$. $\theta(\omega)$ is called the *spectral phase* of f, depending on the angular frequency ω. Assuming it to be differentiable, then taking its derivative with respect to ω defines the *group-delay* (GD),

$$\tau(\omega) = -\frac{d}{d\omega}\theta(\omega). \tag{3.5}$$

It can be interpreted as indicating the time of arrival of different frequency components. Considering a dirac impulse signal at $t = t_0$, $f_\delta(t) = \delta_{t_0}(t)$, its Fourier transform is given by $\widehat{f_\delta}(\omega) = \int \delta_{t_0}(t)e^{-i\omega t}dt = e^{-i\omega t_0}$ and consequently, the GD is $\tau(\omega) = t_0$. For stationary signals it is zero. In an audio context, computing the GD would correspond to the estimation of transient parts of a signal.

3.2 IF and GD on the Time-Frequency Plane

Starting from a time-frequency representation, as we did at the beginning of the chapter, we can define local versions of IF and GD by the partial derivatives of the phase on the time-frequency plane. Let us consider a general complex-valued time-frequency representation $TF_\Psi f(t,\omega)$ of a signal $f \in L^2(\mathbb{R})$, then we can write it as

$$TF_\Psi f(t,\omega) = M^f_\Psi(t,\omega)e^{i\Phi^f_\Psi(t,\omega)}, \tag{3.6}$$

with $(t,\omega) \in \mathbb{R}^2$, Ψ denoting a family of localization functions and $M^f_\Psi(t,\omega)$ and $\Phi^f_\Psi(t,\omega)$ stand for magnitude and phase respectively. With the phase as a function of two variables, we can define analog to IF and GD

$$\widehat{\omega}(t,\omega) = \frac{1}{2\pi}\frac{\partial}{\partial t}\Phi^f_\Psi(t,\omega) \tag{3.7}$$

as the *channelized instantaneous frequency* (CIF) and

$$\widehat{\tau}(t,\omega) = -\frac{\partial}{\partial \omega}\Phi^f_\Psi(t,\omega) \tag{3.8}$$

as *local group-delay* (LGD).

To actually compute the phase derivatives, we follow a similar approach as introduced by Auger and Flandrin in [15]. It is based on properties of the STFT and computes CIF and LGD directly from it. Before we state the result, we clarify STFT-related conventions of the phase, which model invariances w.r.t. time and frequency [28]. As we defined the STFT in Section 1.1.1, it is *frequency-invariant*, since there is no running variable in the

exponential term. However, we can also define a *time-invariant* version of the STFT by changing the order of modulation and translation, i.e.

$$\mathcal{V}_g^t f(t,\omega) = \langle f, T_t M_\omega g\rangle = \int_{\mathbb{R}} f(\tau)\overline{g(\tau-t)}e^{-2\pi i\omega(\tau-t)}d\tau. \tag{3.9}$$

Now the time variable t runs also in the exponential term. To use the phase derivatives in the context of representation we will consider CIF in a frequency-invariant and LGD in a time-invariant setting. In fact, this refers to wrapping in time and frequency, respectively.

Lemma 2. *Let $f, g \in L^2(\mathbb{R})$ and at least one of them in Schwartz class $\mathcal{S}(\mathbb{R})$ of rapidly decaying smooth functions, then $\mathcal{V}_g f(t,\omega)$ and $\mathcal{V}_g^t f(t,\omega)$ are infinitely partially differentiable in both variables, t and ω. Let further g' denote the differentiated window $g'(t) = \frac{d}{dt}g(t)$ and tg the window multiplied with the running variable $tg(t)$. We can write CIF and LGD as*

$$\widehat{\omega}(t,\omega) = -\frac{1}{2\pi}\operatorname{Im}\left\{\frac{\mathcal{V}_{g'} f(t,\omega)}{\mathcal{V}_g f(t,\omega)}\right\} \tag{3.10}$$

$$\widehat{\tau}(t,\omega) = \operatorname{Re}\left\{\frac{\mathcal{V}_{tg}^t f(t,\omega)}{\mathcal{V}_g f(t,\omega)}\right\}. \tag{3.11}$$

Proof. The first statement is proven in [6]. For the representation statement we apply the logarithm to the magnitude-phase representation of the STFT, which separates magnitude and phase,

$$\log \mathcal{V}_g f(t,\omega) = \log(M_g^f(t,\omega)e^{2\pi i\Phi_g^f(t,\omega)}) = \log M_g^f(t,\omega) + 2\pi i\Phi_g^f(t,\omega), \tag{3.12}$$

Now taking the partial derivative of $\Phi_g^f(t,\omega)$ w.r.t. t gives

$$\frac{\partial}{\partial t}\Phi_g^f(t,\omega) = \frac{1}{2\pi}\operatorname{Im}\Big\{\frac{\partial}{\partial t}\log \mathcal{V}_g f(t,\omega)\Big\} = \frac{1}{2\pi}\operatorname{Im}\left\{\frac{\frac{\partial}{\partial t}\mathcal{V}_g f(t,\omega)}{\mathcal{V}_g f(t,\omega)}\right\} \tag{3.13}$$

and compute $\frac{\partial}{\partial t}\mathcal{V}_g f(t,\omega)$ using Leibnitz's rule

$$\frac{\partial}{\partial t}\mathcal{V}_g f(t,\omega) = \int_{\mathbb{R}} f(\tau)\frac{\partial}{\partial t}g(\tau-t)e^{-2\pi i\omega\tau}d\tau = -\mathcal{V}_{g'} f(t,\omega). \tag{3.14}$$

For the second formula we consider first $\frac{\partial}{\partial \omega}\mathcal{V}_g^t f(t,\omega)$,

$$\frac{\partial}{\partial \omega}\mathcal{V}_g^t f(t,\omega) = \int_{\mathbb{R}} f(\tau)g(\tau-t)\frac{\partial}{\partial \omega}e^{-2\pi i\omega(\tau-t)}d\tau \tag{3.15}$$

$$= -2\pi i\int_{\mathbb{R}} f(\tau)g(\tau-t)(\tau-t)e^{-2\pi i\omega(\tau-t)}d\tau = -2\pi i\mathcal{V}_{tg}^t f(t,\omega). \tag{3.16}$$

Therefore, taking the partial derivatives in (3.12) w.r.t. ω interchanges real and imaginary part and thus gives, analog to (3.13),

$$-\frac{\partial}{\partial\omega}\Phi_g^f(t,\omega) = -\frac{1}{2\pi}\operatorname{Im}\left\{\frac{-2\pi i\mathcal{V}_{tg}f(t,\omega)}{\mathcal{V}_g f(t,\omega)}\right\} = \operatorname{Re}\left\{\frac{\mathcal{V}_{tg}f(t,\omega)}{\mathcal{V}_g f(t,\omega)}\right\}. \tag{3.17}$$

□

These relations are quite general and hold for many types of windows. However, the representations indicate, that the zeros of the STFT is where things might go bad.

3.3 The Pole Behaviour of the Phase

Near the zeros of the STFT the phase derivatives can fly past our ears. In [6], the behaviour of phase information near zeros of the STFT is treated analytically. In the following we summarize the main results in one Lemma.

Lemma 3. *Let $f, g \in L^2(\mathbb{R})$ and assume that $\mathcal{V}_g f(t,\omega) \in C^2(\mathbb{R}^2)$ with $(t_0,\omega_0) \in \mathbb{R}^2$ satisfies*

- $\mathcal{V}_g f(t_0,\omega_0) = 0$
- $\det J_{\mathcal{V}}(t_0,\omega_0) < 0$, *where*

$$J_{\mathcal{V}}(t_0,\omega_0) = \begin{pmatrix} \frac{\partial\mathcal{U}}{\partial t}(t_0,\omega_0) & \frac{\partial\mathcal{U}}{\partial\omega}(t_0,\omega_0) \\ \frac{\partial\mathcal{W}}{\partial t}(t_0,\omega_0) & \frac{\partial\mathcal{W}}{\partial\omega}(t_0,\omega_0) \end{pmatrix} \tag{3.18}$$

  is the Jacobian of $\mathcal{V} = \mathcal{U} + i\mathcal{W}$ at (t_0,ω_0).

Then the phase $\Phi_g^f(t,\omega)$ of $\mathcal{V}_g f(t,\omega)$ satisfies

$$\lim_{\omega\to\omega_0}\frac{\partial}{\partial t}\Phi_g^f(t_0,\omega) = \begin{cases} +\infty, & \text{if } \omega \uparrow \omega_0 \text{ from below} \\ -\infty, & \text{if } \omega \downarrow \omega_0 \text{ from above.} \end{cases} \tag{3.19}$$

and

$$\lim_{t\to t_0}\frac{\partial}{\partial\omega}\Phi_g^f(t,\omega_0) = \begin{cases} +\infty, & \text{if } t \to t_0 \text{ from the left} \\ -\infty, & \text{if } t_0 \leftarrow t \text{ from the right.} \end{cases} \tag{3.20}$$

If furthermore, $\mathcal{V}_g f(t,\omega) \in C^3(\mathbb{R}^2)$, then there exist numbers $c, c' \in \mathbb{R}$, such that

$$\lim_{t\to t_0}\frac{\partial}{\partial t}\Phi_g^f(t,\omega_0) = c \tag{3.21}$$

and

$$\lim_{\omega\to\omega_0}\frac{\partial}{\partial\omega}\Phi_g^f(t_0,\omega) = c'. \tag{3.22}$$

The case for $J_{\mathcal{V}} > 0$ is analogous, only that the signs of the diverging limits changes.

The lemma treats the pole behaviour of the phase derivatives and connects it to the smoothness and continuous differentiability of the STFT. In the proof of Lemma 2 we see that this is linked to the smoothness of the window function.
However, for the purpose we want to use them, namely with a representational emphasis, it pleases our needs to set them to zero at the zeros of the underlying STFT.

Discrete Setting

The method of computing CIF and LGD directly from the STFT is quite beneficial for the implementation since the phase derivatives are computed by pointwise operations of a second STFT with redesigned windows. Instead of using numerical methods for finite differences on a discrete STFT, like in [13]. We can compute the partial derivatives numerically by coefficient-wise operations of discrete STFTs, according to (3.10) and (3.11). Additionally we avoid the problem of unwrapping, since we get our desired adaptive unwrapping (time resp. frequency invariance) already by construction. However, we saw in the latter section that it can get quite inconvenient around the zeros of the underlying STFT, so in a numerical sense, we shall consider a small neighbourhood around these pathetic cases to be zero. This can be justified from the signal processing point of view saying that at regions with zero amplitude, the influence of phase is negligible as well.

We use the LTFAT-routine `gabphasederiv`, which computes the method from Lemma 2, based on the discrete Gabor transform. For these and all following plots we use discrete versions of a dilated Gaussian $g_\lambda(t) = e^{-\pi t^2/\lambda}$ as window functions in the underlying Gabor transform. λ denotes the ratio between the effective time and frequency support of g_λ. This window function enables particularly nice plots for the phase derivatives and we will use it also for theoretical discussions. As demonstration we consider again the MATLAB-version of our melody, see Figure 3.1 and add two harmonics at the high A note around second 2. Figure 3.3 and 3.4 show `gabphasederiv` computing CIF and LGD representations of this version of the melody.

What can we observe from these figures:

CIF: At the onsets and offsets of the notes the CIF values decay to zero in time direction, which seems reasonable since we expect the IFs at transient regions to be small. In frequency direction, we can see locally linear shapes around the IFs of the single notes. So, what the CIF actually seems to compute is something like a local linear frequency distance measure to the frequency bins where the actual IFs lie, positive below and negative above it.

LGD: Here we find the regions of zero around the tonal parts of the melody, almost as if the nonzero CIF values were cut out. In time direction we observe exactly the same phenomenon as in the CIF case, namely locally linear shapes around the transients, induced by the note onsets and offsets. These analogously give local linear time distance measures to the next GD.

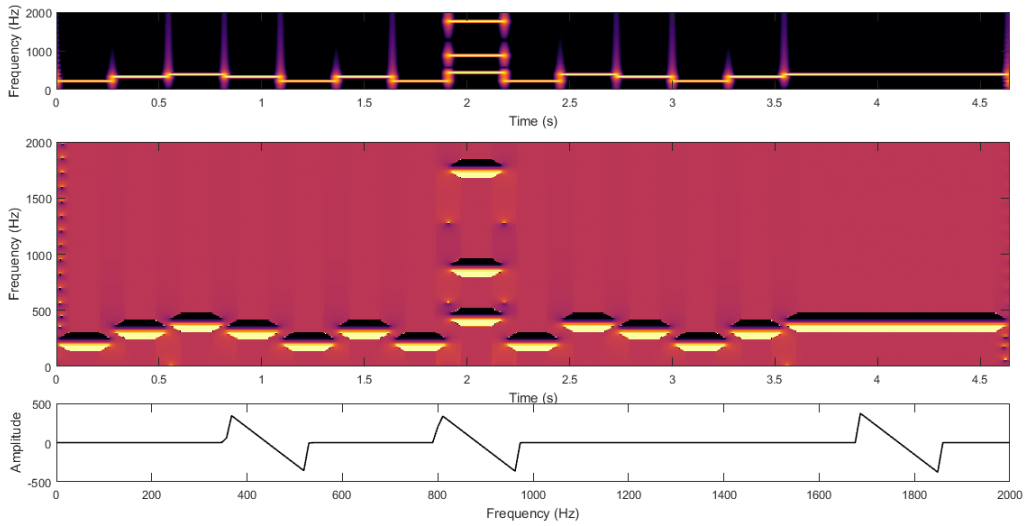

Figure 3.3: (Top) Gabor magnitude representation of the MATLAB melody. (`dgt` settings: $a = 32, M = 4096$ and a Gaussian window with $\lambda = 8$)
(Mid) CIF representation of the MATLAB melody (same `dgt` settings)
(Bottom) Slice along the frequency axis at around second 2.

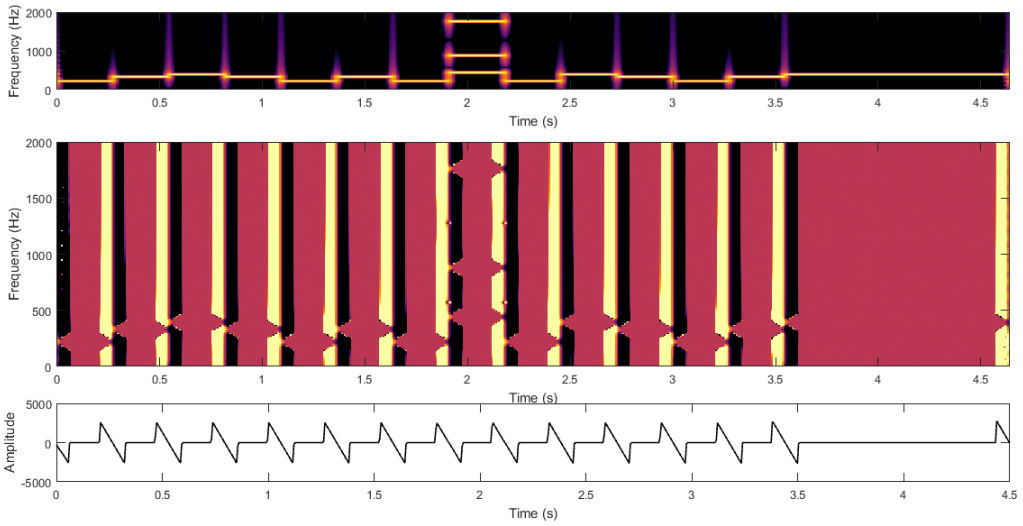

Figure 3.4: (Top) Gabor magnitude representation of the MATLAB melody. (`dgt` settings: $a = 256, M = 4096$ and a Gaussian window with $\lambda = 8$)
(Mid) LGD representation of the MATLAB melody (same `dgt` settings)
(Bottom) Slice along the time axis at around 100Hz.

3.3.1 Analytic Shapes of CIF and LGD

It is possible to derive analytic explanations for the behaviour of CIF and LGD we observed in the latter section.

We start with the CIF representation by considering a stationary signal and show that the CIF along the frequency axis has the locally linear behaviour we have seen in Figure 3.3.

Lemma 4. *Let $f(t) = e^{2\pi i \xi_0 t}$ be a complex sinusoid with frequency ξ_0 and $g(t) = e^{-\pi t^2/\lambda}$ a dilated Gaussian window. Then*

$$\widehat{\omega}(t,\omega) = \xi_0 - \omega \tag{3.23}$$

holds for all $t \in \mathbb{R}$.

Proof. The derivative of the Gaussian is $g'(t) = -\frac{2\pi t}{\lambda} e^{-\pi t^2/\lambda}$. We plug f into our CIF formula (3.10) and make a straightforward computation:

$$\widehat{\omega}(t,\omega) = -\frac{1}{2\pi} \operatorname{Im}\left\{ \frac{\langle f, M_\omega T_t g' \rangle}{\langle f, M_\omega T_t g \rangle} \right\} \tag{3.24}$$

$$= -\frac{1}{2\pi} \operatorname{Im}\left\{ \frac{-2\pi}{\lambda} \frac{\int_{\mathbb{R}} e^{2\pi i \xi_0 \tau} e^{-2\pi i \omega \tau} e^{-\pi(\tau - t)^2/\lambda} (\tau - t)\, d\tau}{\int_{\mathbb{R}} e^{2\pi i \xi_0 \tau} e^{-2\pi i \omega \tau} e^{-\pi(\tau-t)^2/\lambda}\, d\tau} \right\} \tag{3.25}$$

The substitution $s = \frac{\tau - t}{\sqrt{\lambda}}$ further yields

$$= \operatorname{Im}\left\{ \frac{1}{\lambda} \frac{\int_{\mathbb{R}} e^{2\pi i (\xi_0 - \omega)(\sqrt{\lambda} s + t)} e^{-\pi s^2} \sqrt{\lambda} s\ \sqrt{\lambda} ds}{\int_{\mathbb{R}} e^{2\pi i (\xi_0 - \omega)(\sqrt{\lambda} s + t)} e^{-\pi s^2} \sqrt{\lambda} ds} \right\} \tag{3.26}$$

$$= \operatorname{Im}\left\{ \frac{1}{\sqrt{\lambda}} \frac{\int_{\mathbb{R}} e^{2\pi i (\xi_0 - \omega)\sqrt{\lambda} s} e^{-\pi s^2} s\ ds}{\int_{\mathbb{R}} e^{2\pi i (\xi_0 - \omega)\sqrt{\lambda} s} e^{-\pi s^2} ds} \right\} \tag{3.27}$$

Using $e^{-\pi s^2} s = -\frac{1}{2\pi} \frac{\partial}{\partial s} e^{-\pi s^2}$ for partial integration and the fact that $e^{-\pi t^2}$ is invariant under the Fourier transform yields

$$= \operatorname{Im}\left\{ \frac{1}{\sqrt{\lambda}} \frac{e^{-\pi \lambda (\xi_0 - \omega)^2} i \sqrt{\lambda} (\xi_0 - \omega)}{e^{-\pi \lambda (\xi_0 - \omega)^2}} \right\} \tag{3.28}$$

$$= \operatorname{Im}\big\{ i(\xi_0 - \omega) \big\} \tag{3.29}$$

$$= \xi_0 - \omega \tag{3.30}$$

□

This explains the linear shape of the CIF around ξ_0 along the frequency axis. Note that as the denominator of the quotient in (3.10) computes STFTs, the previous computation was only valid, since our dilated Gaussian provides infinite frequency support. In the case of using a window with only finite frequency support, the CIF coefficients need to be set to zero outside. We could see this in Figure 3.3, where we got piecewise linear functions only locally around the harmonics, depending on the width of the frequency support of the window we used. In comparison to the Gabor magnitude, we have

$$|\langle e^{2\pi i\xi_0 t}, M_\omega T_t g\rangle| = |\widehat{g}(\omega - \xi_0)| \tag{3.31}$$

for all $t \in \mathbb{R}$. Thus, from the magnitude to CIF, the magnitude of the shifted frequency response of the window is replaced by an affine linear function (3.30), making it independent of the window.

Now, using a simple Dirac impulse, we show the local linear behaviour of the LGD representation around it.

Lemma 5. *Let δ_{t_0} be an impulse at t_0 and g any differentiable window function with infinite support. Then*

$$\widehat{\tau}(t,\omega) = t_0 - t \tag{3.32}$$

for all $\omega \in \mathbb{R}$.

Proof. We plug δ_{t_0} into how we defined the LGD, (3.11) and get

$$\widehat{\tau}_\delta(t,\omega) = \mathrm{Re}\left\{\frac{\langle \delta_{t_0}, T_t M_\omega t g\rangle}{\langle \delta_{t_0}, T_t M_\omega g\rangle}\right\} \tag{3.33}$$

$$= \mathrm{Re}\left\{\frac{\int \delta_{t_0}(\tau) e^{-2\pi i\omega(\tau-t)}(\tau - t) g(\tau - t) d\tau}{\int \delta_{t_0}(\tau) e^{-2\pi i\omega(\tau-t)} g(\tau - t) d\tau}\right\} \tag{3.34}$$

$$= \mathrm{Re}\left\{\frac{e^{-2\pi i\omega(t_0-t)}(t_0 - t) g(t_0 - t)}{e^{-2\pi i\omega(t_0-t)} g(t_0 - t)}\right\} \tag{3.35}$$

$$= t_0 - t \tag{3.36}$$

for all $\omega \in \mathbb{R}$. □

This explains the shape of the curve we have seen in Figure 3.4, a linearly decreasing function around the actual GD, t_0, measuring linearly the distance to it. Also here, in the case where we use a window with only finite time support, we need to set the values outside to zero. As a comparison, we have a look at the Gabor magnitude,

$$|\langle \delta_{t_0}, M_\omega T_t g\rangle| = |g(t_0 - t)| \tag{3.37}$$

for all $\omega \in \mathbb{R}$. This means that from the magnitude to LGD, the shifted window is replaced by an affine linear function (3.36), making it again independent of the window.

Before we are ready for the grand finale, we briefly discuss some of the applications of CIF and LGD.

3.4 Application of CIF and LGD

In a representational sense, there are methods called *time-frequency reassignment* and *synchrosqueezing*, which are similar concepts to compute time-frequency representations that should overcome the resolution trade-off between time and frequency and allow a sparser and thus, more readable depiction of the time-frequency information (Figure 3.5). This is done by reassigning/squeezing the computed coefficients to positions in the time-frequency plane, according to the information from CIF and LGD [15, 12, 24].
Another application of CIF and LGD is in terms of the phase vocoder [13]. It uses phase information to allow a reasonable resynthesis after modifications in the time-frequency domain and has been further developed, incorporating CIF and LGD for the resynthesis to better handle transients [30].
The next application is based on the same motivation as for the phase vocoder, namely, that synthesizing from the magnitude information only yields a signal with incoherent waveforms, producing terrible soundeffects caused by amplitude fluctuations. In [31] this problem was treated for the generation of single notes with special acoustic characteristics using neural networks. In this case, the correct temporal alignment of the harmonics is important for the perceptual quality of the generated sound. Therefore the authors fed the neural network synthesizer additionally with the CIF of the original sounds, which it learns from, yielding perceptually better results than without.

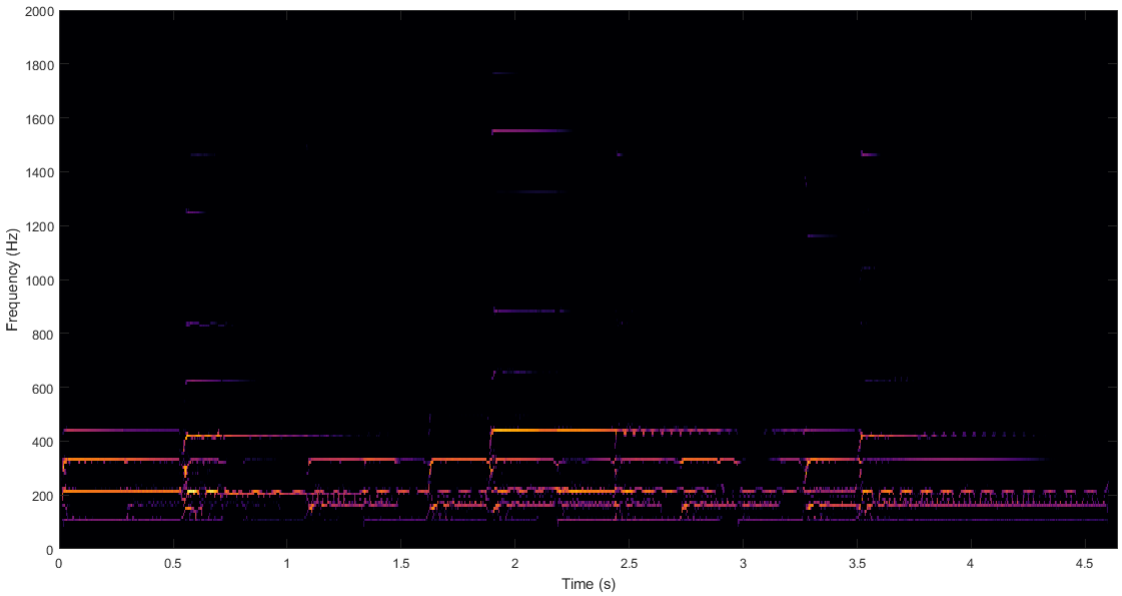

Figure 3.5: Time-frequency reassigned spectrogram of the guitar version of the melody, computed with the LTFAT routine `gabreassign`. Compared to the top plot in Figure 1.2, which depicts the spectrogram of the same signal, this representation is much sparser und thus, clearer to read.

Now we are ready.

4 Phase Scattering

We have seen that CIF and LGD are useful quantities to describe time-frequency information. This makes us very curious to explore the behaviour of them being scattered. Instead of taking the modulus of the complex time-frequency coefficients, we have the mappings to the partial derivatives of the phases as non-linearity involved. Our goal here shall be to set up a scattering procedure with a motivation to exploit the strengths of the phase, i.e. precise time and frequency localization, so we deviate from the original motivation for translation invariance.

Definition 15 (Phase Scattering Coefficients)**.** *Let $\Phi^f_\Psi(t,\omega)$ denote the phase of a $f \in L^2(\mathbb{R})$ w.r.t. a general time-frequency transform, based on a family of localization functions Ψ. Then we compute the CIF, resp. LGD-scattering coefficients recursively by*

$$\widehat{\Omega}_k[\xi_k]f := \frac{1}{2\pi}\frac{\partial}{\partial t}\Phi^{\widehat{\Omega}_{k-1}f}_{\Psi_{k-1}}(t,\xi_{k-1}) \tag{4.1}$$

$$\widehat{T}_k[\xi_k]f := -\frac{\partial}{\partial \omega}\Phi^{\widehat{T}_{k-1}f}_{\Psi_{k-1}}(t,\xi_{k-1}) \tag{4.2}$$

with $\widehat{\Omega}_0 f = \widehat{T}_0 f = f$. Let $p = (\xi_1,\dots,\xi_k) \in \Xi_1 \times \cdots \times \Xi_k$ denote a frequency-index path, then we can define

$$\widehat{\Omega}[p]f = \widehat{\Omega}_k[\xi_k]\dots\widehat{\Omega}_1[\xi_1]f \tag{4.3}$$

as k-th order CIF Scattering Coefficients and

$$\widehat{T}[p]f = \widehat{T}_k[\xi_k]\dots\widehat{T}_1[\xi_1]f \tag{4.4}$$

as k-th order LGD Scattering Coefficients.

We shall again call $\widehat{\Omega}[q,\xi]f$, considered for all $\xi \in \Xi$, the k-th *CIF scattering layer* of f w.r.t. a path q of length $k-1$ and write $\widehat{\Omega}[q,\cdot\,]f$. Analog for the LGD case.

In a mathematical sense, actually, this definition is a bit bold. To make these operators at least well-defined, we have to assume all the $\Phi^{\widehat{\Omega}_{k-1}f}_{\Psi_{k-1}}(t,\xi_{k-1})$ to be in $L^2(\mathbb{R})$, differentiable and finite, which seems to be not very realistic, as our pole discussion suggests. A possibility would be to consider everything in sense of distributions, but this goes beyond the scope of this thesis. Nevertheless, we will put back on our signal processing glasses and run some experiments trying to collect some insights of how these CIF and LGD-scattering coefficients look like and what information there is contained. For that, we set up a phase scattering procedure, based on the Gabor transform.

4.1 Gabor Phase Scattering

To conduct numerical experients, we define the CIF/LGD scattering coefficients based on the approach we introduced in Lemma 2.

Definition 16 (Gabor Phase Scattering Coefficients)**.** *Let* $\mathcal{G}_\ell := \{M_{mb_\ell} T_{na_\ell} g_\ell\}_{(m,n)\in\mathbb{Z}^2}$ *be Gabor frames with differentiable* g_ℓ*. Let* g'_ℓ *denote the differentiated window* $g'_\ell(t) = \frac{d}{dt} g_\ell(t)$ *and* $tg = tg(t)$*. We can define the Gabor CIF/LGD scattering coefficients analogously to* (4.3) *and* (4.4) *by computing the recursions*

$$\widehat{\Omega}_k[mb_k]f(n) = f_k = -\frac{1}{2\pi}\,\mathrm{Im}\left\{\frac{\langle f_{k-1}, M_{mb_k} T_{na_k} g'_k\rangle}{\langle f_{k-1}, M_{mb_k} T_{na_k} g_k\rangle}\right\}, \quad n \in \mathbb{Z} \tag{4.5}$$

$$\widehat{T}_k[mb_k]f(n) = f_k = \mathrm{Re}\left\{\frac{\langle f_{k-1}, T_{na_k} M_{mb_k} t g_k\rangle}{\langle f_{k-1}, T_{na_k} M_{mb_k} g_k\rangle}\right\}, \quad n \in \mathbb{Z} \tag{4.6}$$

along a frequency-index path $q = (q_1, ..., q_k) \in b_1\mathbb{Z} \times ... \times b_k\mathbb{Z}$*.*

We will apply discrete versions of these transforms to toy examples to see our new phase scattering coefficients in action.

4.1.1 Frequency Modulation

Since the supreme discipline of the instantaneous phase is dealing with non-stationary signals, we let our phase scattering procedure apply on a frequency modulated sinusoid $f(t) = \cos(2\pi\xi_0 t + \gamma(t))$. As an illustrative example let us choose $\gamma(t) = \epsilon\cos(2\pi\xi_1 t)$, which yields a vibrato signal at frequency ξ_0 with a modulation at frequency ξ_1, as we had it in 2.1.3. Having a look at the IF of f, we get $\phi'(t) = \xi_0 - \xi_1\epsilon\sin(2\pi\xi_1 t)$, which defines a signal with a IF of ξ_1, exactly the frequency of the vibrato modulation. In the phase scattering case, however, the CIFs are computed with discrete Gabor transforms and thus, are allocated to points on a rectangular grid on the time-frequency plane. Further CIF layers are computed with respect to single frequency bins, i.e. a row of that grid. Since a single row is only responsible for one specific frequency, it seems that it cannot contain all the information of the frequency modulation. However, the linear shapes at every time step are shifted in frequency according to the modulation and thus, its shape can be read along the time axis. So, if we pass the IF of f as time-dependent frequency variable to the expression we computed for the CIF in (3.30) and consider the frequency bin corresponding to the basis frequency ξ_0, then we get

$$\widehat{\Omega}[\xi_0]f(t) = \phi'(t) - \xi_0 = -\epsilon\xi_1\sin(2\pi\xi_1 t). \tag{4.7}$$

This defines a signal with a phase function of $\phi_f(t) = 2\pi\xi_1 t$, having an IF of ξ_1. Thus, $\widehat{\Omega}_2[\xi_0, \cdot\,]f$ will show a linear shape $(\xi_1 - \omega)$ in ω around ξ_1 for all τ. In a theoretical manner we get this result independent of the chosen frequency bin, but in numerical case where we have only finite frequency support windows, we have to care of staying well inside the effective supports.

For a more general frequency modulating γ we find its shape in $\widehat{\Omega}_2$ in the same manner. As numerical examples we set up a sinusoidal signal of 880Hz and modulate it at first by another sinusoid of 10Hz, i.e. $\gamma_1(t) = \cos(2\pi 10t)$ and then by a quadratic chirp, going from 5Hz up to 17Hz in two seconds, i.e. $\gamma_2(t) = \cos(2\pi(5 + t^2)t)$. In Figure 4.1 and 4.3 we show the plots for the flow throughout the different stages of the CIF scattering procedure of the signals f_1 and f_2, w.r.t. the frequency bin of 880Hz. The modulations induced by γ_1 and γ_2 are clearly visible in the 2nd layers respectively and again show a locally linear shape around the present IFs. As a comparison, Figure 4.2 shows the Gabor magnitudes of $\widehat{\Omega}[880]f_1$. Here we can see well the frequency response of the Gaussian at the attracted frequency of 10Hz instead of the locally linear shape the CIF shows.

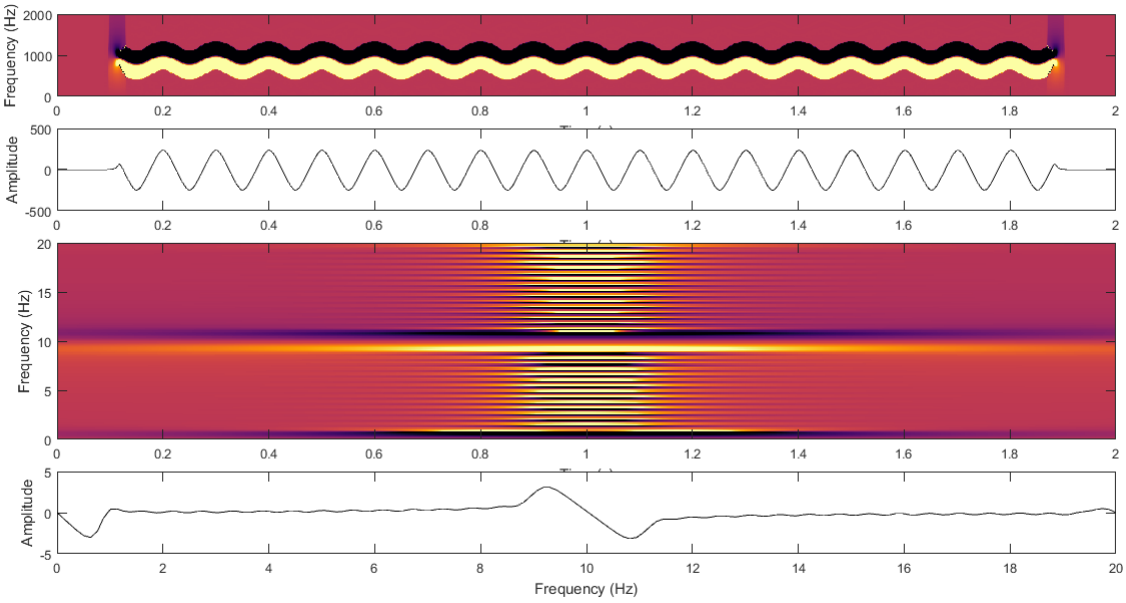

Figure 4.1: (Top) First order CIF scattering layer $\widehat{\Omega}[\,\cdot\,]f_1$ of a vibrato modulated sine wave f_1 with a basis frequency of 880Hz and modulation frequency of 10Hz, i.e. its CIF representation. The underlying Gabor system has parameters $a_1 = 100, M_1 = 2048$ and a Gaussian with $\lambda_1 = 1$.
(2nd Top) Scattering coefficients $\widehat{\Omega}[\xi_0]f_1$ for $\xi_0 = 880$Hz.
(Mid) Second order CIF scattering layer $\widehat{\Omega}[\xi_0, \cdot\,]f_1$. The underlying Gabor parameters are $a_2 = 1, M_2 = 3520$ and a Gaussian with $\lambda_2 = 64$.
(Bottom) $\widehat{\Omega}[\xi_0, \cdot\,]f_1(t_0)$, i.e. a slice along frequency of the plot above. One can see clearly the peak at the modulation frequency of 10Hz.

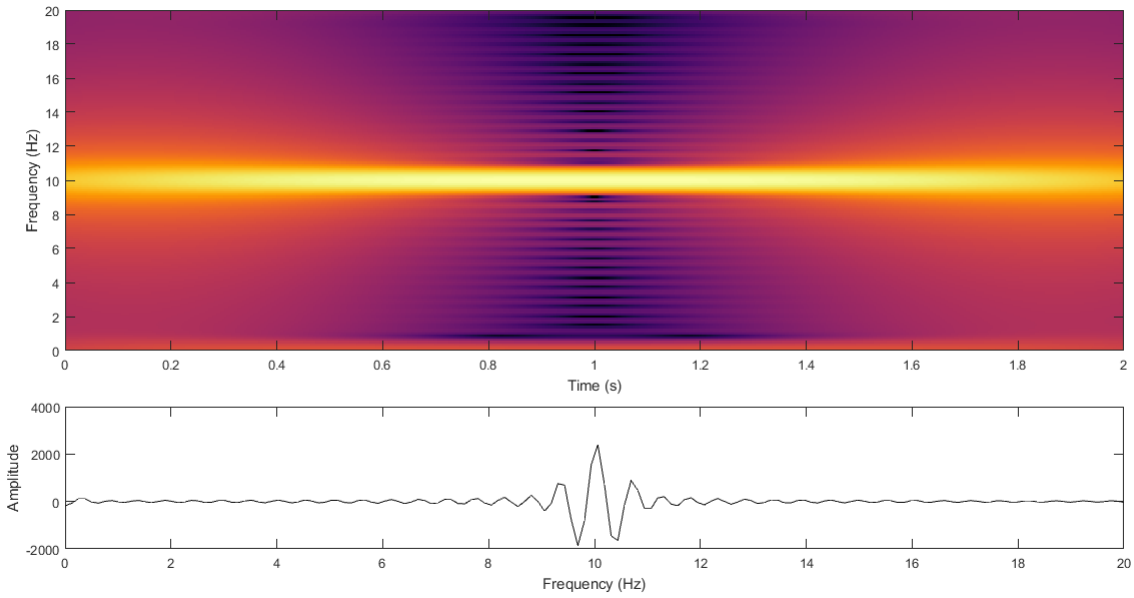

Figure 4.2: (Top) Gabor magnitude representation of $\widehat{\Omega}[\xi_0]f_1$, the first order phase scattering coefficients at ξ_0. The Gabor parameters are the same as in Figure 4.1. (Bottom) $\widehat{\Omega}[\xi_0]f_1(t_0)$, i.e. a slice along frequency of the plot above.

4.2 Mixed Phase Scattering

Let us think about the LGD-scattering procedure for a moment. Roughly speaking, the LGD gives us the distance in time where transients within a signal lie. Figure 3.4 showed how this looks like. We get a locally decreasing linear function centered at the actual GD, peaking at beginning and end of the effective frequency support of the window. Thus, even for very narrow windows, computing another LGD for this type of information does not make much sense, since it will just give again information about the peak arrangement w.r.t. the transients. However, in Section 2.3.2 we have seen how the Gabor magnitude scattering captured the frequency of the temporal patterns among the transients in our signal. In that sense it seems reasonable to use the LGD representation as transient detector with a subsequent CIF transformation to get the frequency information of the LGD-peak arrangement.

Definition 17 (Mixed Phase Scattering)**.** *Let $f \in L^2(\mathbb{R})$ then we can define*

$$\widehat{M}[\xi_0, \xi_1]f = \widehat{\Omega}[\xi_1]\widehat{T}[\xi_0]f \tag{4.8}$$

as second order mixed phase scattering coefficients of f along (ξ_0, ξ_1).

Again, for the sake of having $\widehat{M}$ well-defined we have to assume $\widehat{T}[\xi_0]f$ to be in $L^2(\mathbb{R})$ and differentiable.

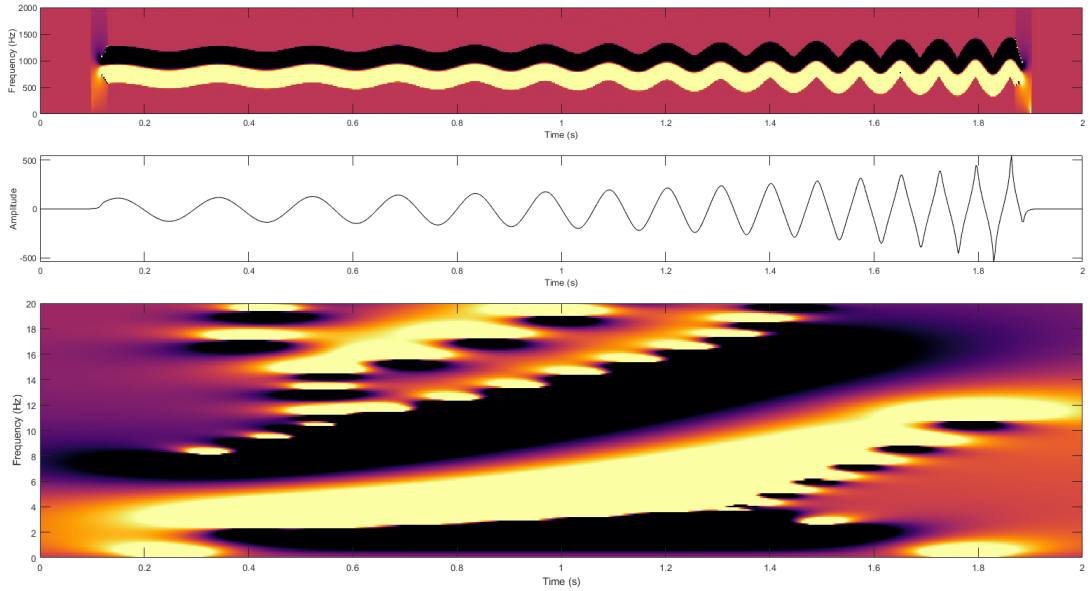

Figure 4.3: (Top) First order CIF scattering layer $\widehat{\Omega}[\,\cdot\,]f_2$ of a vibrato modulated sine wave f_2 with a basis frequency of 880Hz and quadratically increasing modulation frequency from 5Hz to 17Hz in 2 seconds.
(Mid) Scattering coefficients $\widehat{\Omega}[\xi_0]f_2$ for $\xi_0 = 880$Hz, i.e. the CIFs w.r.t. the basis frequency.
(Bottom) Second order CIF scattering layer $\widehat{\Omega}[\xi_0, \cdot\,]f_2$. One can see cleary the quadratic shape of the modulation.
The underlying Gabor parameters are the same as for Figure 4.1, except for $\lambda_2 = 8$.

4.2.1 Impulse Train

Now we apply the Gabor version of the mixed phase scattering to an impulse train to arrive at something similar to what we did in Section 2.3.2.
Let $f_\delta(t) = \sum_{\ell\in\mathbb{Z}} \delta_{2\pi\ell/\xi_0}(t)$ be an impulse train with fundamental frequency ξ_0. In its LGD representation, the regions around each of the single impulses have the locally linear shape we observed in Figure 3.4. Using a window function g with $\operatorname{supp}(g) \geq 2\pi/\xi_0$ we avoid regions of zero amplitude between the impulses. Consequently, we can restrict the linear functions we obtained as LGD for every ℓ symmetrically around the impulses at $2\pi\ell/\xi_0$ and get a piecewise linear function, which we can write as

$$\widehat{\tau}_{f_\delta}(t,\omega) = \sum_{\ell\in\mathbb{Z}} \Delta_\ell(t), \tag{4.9}$$

with

$$\Delta_\ell(t) = \begin{cases} 2\pi\ell/\xi_0 - t & \text{for } t \in (\pi(\ell-1)/\xi_0, \pi(\ell+1)/\xi_0] \\ 0 & \text{else.} \end{cases} \tag{4.10}$$

This signal has a shape in form of a saw. Such a waveform is called *sawtooth wave* and defines a periodic signal, in our case with a fundamental frequency of ξ_0 and the same phase as its sinusoidal sibling, i.e. $\phi_\Delta(t) = 2\pi\xi_0 t$, so its IF of ξ_0 will be captured in $\widehat{M}[\xi_0, \xi_1]f_\delta$ in a locally linear manner, according to $(\xi_0 - \omega)$ in ω.
Figure 4.4 shows the flow throughout the different stages of the mixed phase scattering of an impulse train with $\xi_0 = 5$Hz. We can find the CIF transformation capturing the frequency of the LGD representation of the impulse train. Note that if we would have chosen a window function with a time support smaller than the distance between the impulses, there existed regions where the LGD would have been computed at zero amplitude, i.e. set to zero, yielding a shape like in Figure 3.4. Similar experiments can be conducted using a vibrato signal as input, where the LGD representation marks the inflection points of the modulating sinusoid.

These introductory examples seem to indicate that the scattering of phase information indeed makes sense under some conditions and that phase-related properties can be exploited in a meaningful way. We had to adapt the mathematical issues a bit to our benefit by banning the pole cases to triviality, but did it with a sound justification coming from signal processing. It pleased our motivation of incorporating the phase in finding a representations of audio features in different scales and may motivate to further work with phase information in feature extraction procedures.

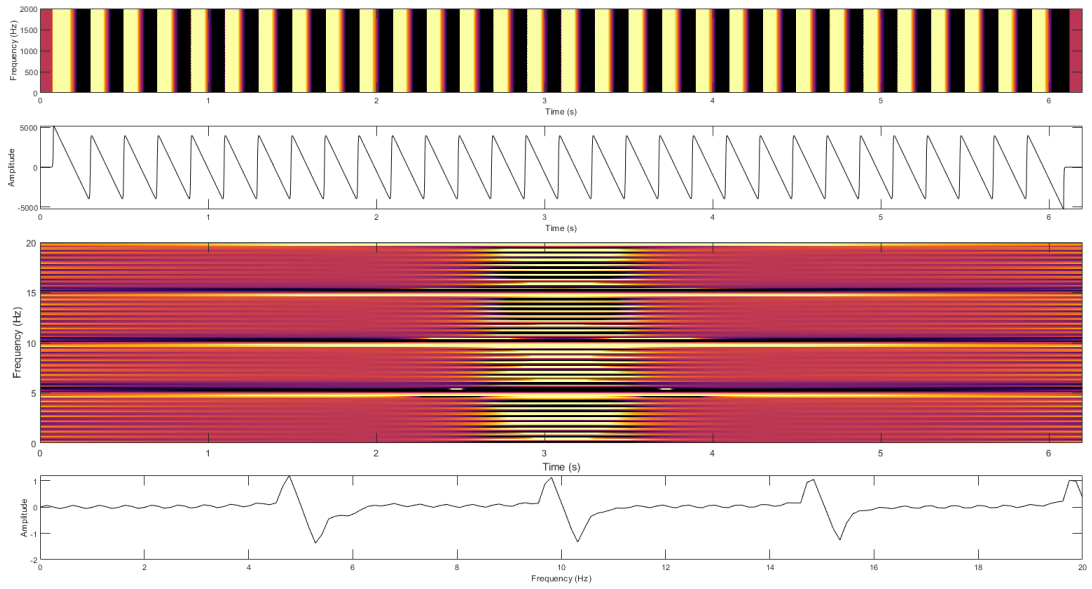

Figure 4.4: (Top) First order mixed phase scattering layer $\widehat{T}[\ \cdot\ ]f_\delta$ of an impulse train with a frequency of 5Hz i.e. its LGD representation. The Gabor parameters used are $a_1 = 100, M_1 = 2048$ and a Gaussian with $\lambda_1 = 16$ (provides time support longer then 200ms).
(2nd Top) Scattering coefficients $\widehat{T}[\xi_0]f_\delta$ for ξ_0, yielding a sawtooth-wave.
(Mid) Second order mixed phase scattering layer $\widehat{M}[\xi_0, \cdot\ ]f_\delta$, i.e. the CIF representation of the sawtooth-wave. The underlying Gabor parameters are $a_2 = 1, M_2 = 3520$ and a Gaussian with $\lambda_2 = 512$.
(Bottom) $\widehat{M}[\xi_0, \cdot\ ]f_\delta(t_0)$, a slice along frequency of the plot above. One can see the local linear shape at 5Hz and at its multiples, which also have been captured.

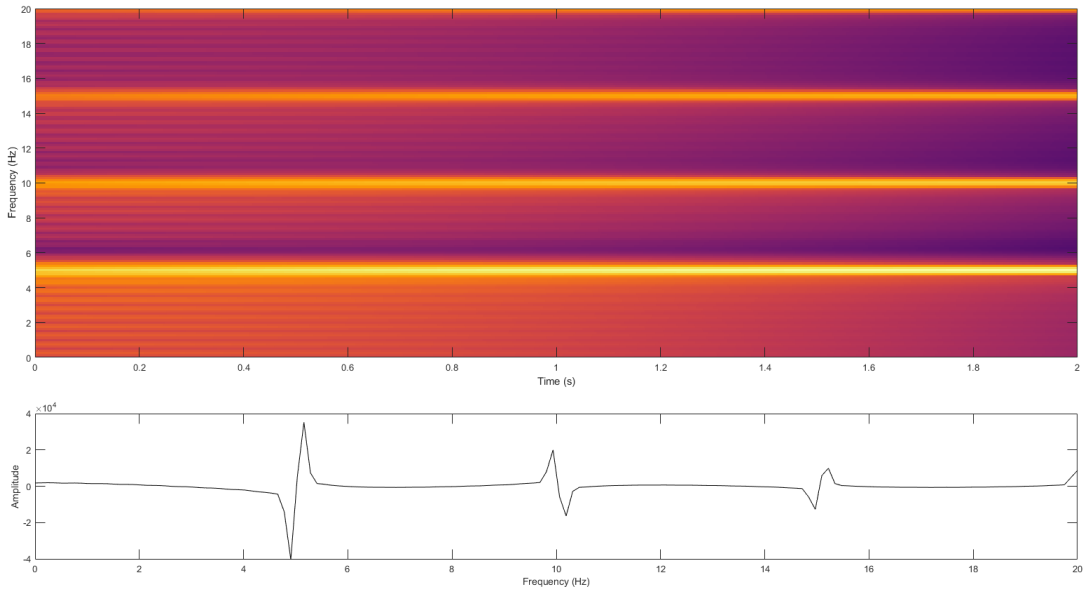

Figure 4.5: (Top) Gabor magnitude representation of $\widehat{T}[\xi_0]f_\delta$ for ξ_0, the sawtooth-wave. The Gabor parameters are the same as for Figure 4.4.
(Bottom) A slice along frequency of the plot above. As the energy in the harmonic response decreases over frequency, also the amplitude of the frequency response of the Gaussians decreases, whereas this doesn't affect the LGD representation.

5 Farewell

We started the thesis with a motivation for representations of audio signals and a glimpse into the basic ideas of machine learning, followed by an overview of some of the most important concepts in time-frequency analysis and signal processing arriving at the intersection of time-frequency representations and the modern queen of machine learning concepts, the convolutional neural network. On the way, we here and there discovered some interesting links between the concepts. All so far should be preparation and motivation for the second chapter, which is dedicated to the scattering transform. We introduced it from a mathematical and from an audio-related point of view, where we refer a lot to Stéphane Mallat, the inventor of the scattering transform. So then, we considered also another scattering version, based on the Gabor transform, originally introduced by Roswitha Bammer. With it we provided numerical examples to show the scattering transform in action and additionally found it as decent extractor of rhythmical features. At this point we have found a concept which brought us closer towards the initial idea of "learning to listen" by extending the temporal scale in focus. Also more in general, to me this concept seems to really be a natural extension of time-frequency decompositions and thus, holds a very fundamental idea.
Then the appealing notion of phase seduced us to undertake the very experimental journey of adapting the idea of time-frequency scattering to a signals phase via its time and frequency derivatives. As a preparation to that we introduced phase-related concepts in a rigorous way to make it handy to use in a scattering framework. In general, this evolved to a try of making the unintuitive concept of phase more accessible to use in the context of time-frequency representations of audio signals. The exploration of phase in a representational context for audio was conceptionally very interesting and spanned links to signal properties, which are not accessible in the common magnitude representations. This paved the way for the daring attempt of defining the novel phase scattering coefficient. The mathematical difficulties we encountered could be tackeled by the theory of distributions, but this would go beyond the scope of the thesis. Nevertheless, toy examples severed as companions to analytically explain its application in numerical experiments. We could conclude that indeed, scattering of phase information works and that properties of the phase can be exploited in a meaningful manner in that way. In the end we have shown that channelized instantaneous frequency and local group delay are suitable concepts for audio representation and scattering propagation and arrived at pleasing results, somewhat comparable to magnitude scattering.
Driven by curiosity and the aim for intuition and recognition about this peaky business, we indeed have been pleased to a certain degree. It was very fruitful learning to see better with the phase-glasses.

Farewell (German)

Die Arbeit wurde begonnen mit einer Motivation für Darstellungen von Audio Signalen und einem kurzen Blick auf deren Verbindung zu maschinellem Lernen. Es folgte ein Überblick von einigen der wichtigsten Konzepte in der Zeit-Frequenz Analyse und Signalverarbeitung, hinleitend zu der Verbindung von Zeit-Frequenz Darstellungen und der Königin des modernen maschinellen Lernens, dem Convolutional Neural Network. Auf dem Weg dahin wurden da und dort einige interessante Verbindungen zwischen den verschiedenen Konzepten herausgearbeitet. Alles soweit war eine Vorbereitung und Motivation für das zweite Kapitel, welches der Scatteringtransformation gewidmet wurde. Diese Transformation wurde von einem mathematischen und von einem audiobezogenen Blickwinkel eingeführt, wobei hier oft auf Stéphane Mallat referiert wurde, dem Urheber der Transformation. Es wurde dann eine alternative Variante der Scatteringtransformation betrachtet, die auf der Gabor Transformation basiert und von Roswitha Bammer eingeführt wurde. Mit dieser Version wurden numerische Bespiele bereitgestellt, bezüglich derer sie sich als Zeit-Frequenz Darstellung bezüglich alternativer Eigenschaften von Audio Signalen zeigte, mitunter Rhythmus und Tempo. Mit diesem Konzept können also weitreichenderen zeitlichen Abhängigkeiten von akustischen Ereignissen auf natürliche Art und Weise erfasst werden, was uns somit ein Stück näher der anfänglichen Idee von "Lernen zu Hören" gebracht hat. Auch allgemeiner scheint dieses Konzept wahrhaftig eine natürliche Erweiterung von Zeit-Frequenz Zerlegungen zu sein und somit eine fundamentale Idee inne zu haben.
In weiterer Folge hat dann der ansprechende Begriff der Phase zu dem Experiment verführt, die Idee des Zeit-Frequenz Scatterings auf Phaseninformation zu adaptieren. Als Vorbereitung darauf wurden phasenbezogene Konzepte eingeführt und verständlich aufbereitet mit einem Fokus auf die Ableitung der Phase in Zeit-, und Frequenzrichtung, was das eher unintuitive Konzept von Phase im Kontext von Zeit-Frequenz Darstellungen von Audio Signalen zugänglicher gemacht hat. Diese Auseinandersetzung war konzeptionell sehr interessant and spannte interessante Verbindungen zu Eigenschaften von Signalen, die für die geläufigen Magnitudendarstellungen nicht zugänglich sind. Es hat somit den Weg geebnet für den gewagten Versuch, die neuen Phasen-Scatteringkoeffizienten zu definieren. Die mathematischen Unebenheiten, die die Definition mit sich brachte konnten mit der Theorie der Distributionen geebnet werden, was jedoch den Rahmen der Arbeit sprengen würde. Standardbeispiele dienten dazu auch analytisch zu erklären was hinter den numerischen Anwendungen passiert.
Schlussfolgernd kann man behaupten, dass es durchaus Sinn ergibt und auch erkenntnisbringend ist, Scattering auf Phaseninformation anzuwenden und dass so Eigenschaften der Phase sinnvoll ausgeschöpft werden können. Im Endeffekt wurde gezeigt, dass die Ableitungen der Phase passende Konzepte für die Darstellung von Audio Signalen sind

und man mit deren Scatteringkoeffizienten zu zufriedenstellenden Resultaten gelangt, die anwendungsbezogen vergleichbar mit denen der herkömmlichen Scatteringtransformation sind.
Getrieben durch die Neugier und dem Bestreben nach Intuition und Erkenntnis bezüglich Phaseninformation, wurden wir in der Tat zu gewissem Grad befriedigt. Insgesamt war es sehr fruchtbar, durch die Phasen-Brillen besser Sehen gelernt zu haben.

Bibliography

[1] J. Andén and S. Mallat. Scattering representation of modulated sounds. In *Proceedings of the 15th International Conference on Digital Audio Effects (DAFx-12)*, York, UK, 2012.

[2] J. Andén and S. Mallat. Deep scattering spectrum. *IEEE Transactions on Signal Processing*, 62(16):4114–4128, 2014.

[3] T. Angles and S. Mallat. Generative networks as inverse problems with scattering transforms. In *International Conference on Learning Representations*, Vancouver, Canada, 2018.

[4] P. Balazs. *Regular and Irregular Gabor Multipliers with Application to Psychoacoustic Masking.* PhD thesis, University of Vienna, 2005.

[5] P. Balazs. Basic definition and properties of bessel multipliers. *Journal of Mathematical Analysis and Applications*, 325(1):571–585, 2007.

[6] P. Balazs, D. Bayer, F. Jaillet, and P. Søndergaard. The phase derivative around zeros of the short-time Fourier transform. *Applied and Computational Harmonic Analysis*, 30(3):610–621, 2016.

[7] P. Balazs, N. Holighaus, T. Necciari, and D. T. Stoeva. Frame theory for signal prcoessing in psychoacoustics. In R. Balan, J. J. Benedetto, W. Czaja, M. Dellatorre, and K. A. Okoudjou, editors, *Excursions in Harmonic Analysis Vol. 5.*, pages 225–268. Springer, 2017.

[8] R. Bammer and M. Dörfler. Invariance and stability of gabor scattering for music signals. In *Proceedings of Sampling Theory and Applications (Sampta)*, 2017.

[9] E. J. Candès. *Ridgelets: Theory and Applications.* PhD thesis, Stanford University, 1998.

[10] E. J. Candès and D. L. Donoho. Continuous curvelet transform ii. discretization and frames. *Applied Computational Harmonic Analysis*, 19(2):198–222, 2005.

[11] O. Christensen. *An Introduction to Frame and Riezs Bases.* Birkhäuser, 2003.

[12] I. Dauberchies and S. Maes. A non-linear squeezing of the continuous wavelet transform based on auditors nerve models. *Wavelets in Medicine and Biology*, pages 527–546, 1996.

[13] M. Dolson. The phase vocoder: A tutorial. *Computer Musical Journal*, 10(4):11–27, 1986.

[14] M. Dörfler and T. Grill. Inside the spectogram: Convolutional neural networks in audio processing. In *Proceedings of Sampling Theory and Applications (Sampta)*, 2017.

[15] P. Flandrin, F. Auger, and E. Chassande-Mottin. Time-frequency reassignment: From principles to algorithms. *Applications in Time-Frequency Signal Processing*, page 179–203, 2003.

[16] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

[17] P. Grohs, D. Perekrestenko, D. Elbrächter, and H. Bölcskei. Deep neural network approximation theory. submitted, 2019.

[18] K. Gröchenig. *Foundations of Time-Frequency Analysis*. Birkhäuser, Boston, 2001.

[19] D. Haider and P. Balazs. Extraction of rhythmical features with the Gabor scattering transform. In *Proceedings of the Conference on Computer Music Mutidisciplinary Research (CMMR)*, Marseille, France, 2019.

[20] G. Kutyniok and D. Labate. *Shearlets: Multiscale Analysis for Multi-Variate Data*. Birkhäuser, 2012.

[21] S. Mallat. *A Wavelet Tour of Signal Processing, 2nd edition*. Academic Press, San Diego, 1999.

[22] S. Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, 2012.

[23] A. Marafioti, N. Holighaus, N. Perraudin, and P. Majdak. Adversarial generation of time-frequency features with application in audio synthesis. *arXiv preprint arXiv:1902.04072*, 2019.

[24] T. Oberlin, S. Meignen, and V. Perrier. The Fourier-based synchrosqueezing transform. In *39th International Conference on Acoustics, Speech, and Signal Processing*, 2014.

[25] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu. Wavenet: A generative model for raw audio. *CoRR abs/1609.03499*, 2016.

[26] A. V. Oppenheim. The importance of phase in signals. In *Proceedings of the IEEE*, volume 69, pages 529–541, 1981.

[27] J. Pons, O. Nieto, M. Prockup, E. Schmidt, A. Ehmann, and X. Serra. End-to-end learning for music audio tagging at scale. In *Proceedings of the 19th ISMIR Conference*, Paris, France, 2018.

[28] Z. Průša. Stft and dgt phase conventions and phase derivatives interpretation. Technical report, Acoustics Research Institute, Austrian Academy of Sciences, 2015.

[29] Z. Průša, P. Balazs, and P. L. Søndergaard. A non-iterative method for (re)construction of phase from stft magnitude. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(5):1154–1164, 2017.

[30] Z. Průša and N. Holighaus. Phase vocoder done right. In *Proceedings of the 25th European Signal Processing Conference (EUSIPCO–2017)*, Kos Island, Greece, 2017.

[31] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[32] S. S. Stevens, J. Volkmann, and E.B. Newman. A scale for the measurement of the psychological magnitude pitch. *Journal of the Acoustic Society of America*, 8(3):208, 1937.

[33] D. T. Stoeva and P. Balazs. Riesz bases multipliers. In *Concrete Operators, Spectral Theory, Operators in Harmonic Analysis and Approximation*, volume 236, pages 475–482. Birkhäuser, Heidelberg, New York, Dordrecht, 2014.

[34] P. Søndergaard, B. Torrésani, and P. Balazs. The linear time frequency analysis toolbox. *International Journal of Wavelets, Multiresolution and Information Processing*, 10(4), 2012.

[35] T. Wiatowski and H. Bölcskei. Deep neural networks based on semi-discrete frames. In *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pages 1212–1216, 2015.

Impressions Through the Phase-Glasses

To get a feeling of how beautiful it is in the world of phase, we want to conclude the thesis with some impressions from my time at the ARI through the phase-glasses.

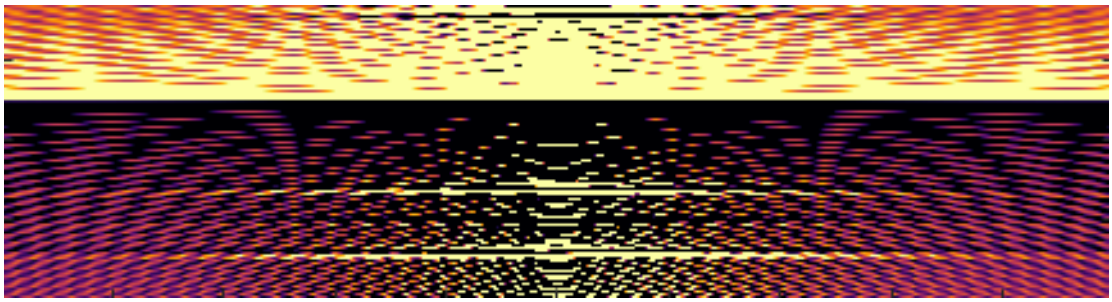

Figure 5.1: This is a panorama picture of the sunrise in styria at the AI summerschool.

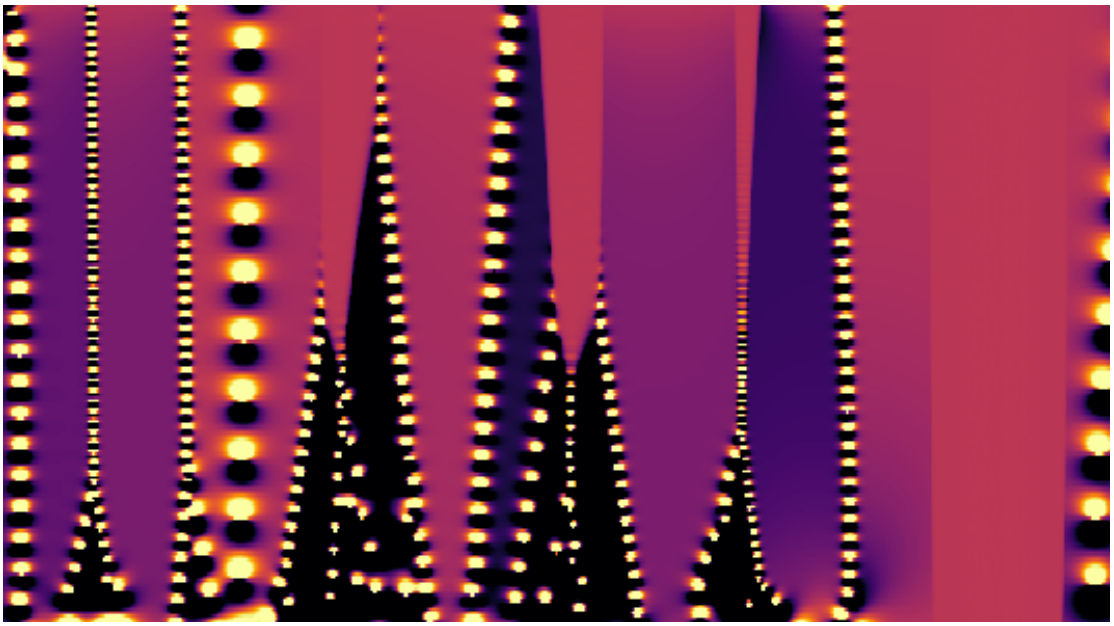

Figure 5.2: This is the view on Vienna from the top of the ARI building.

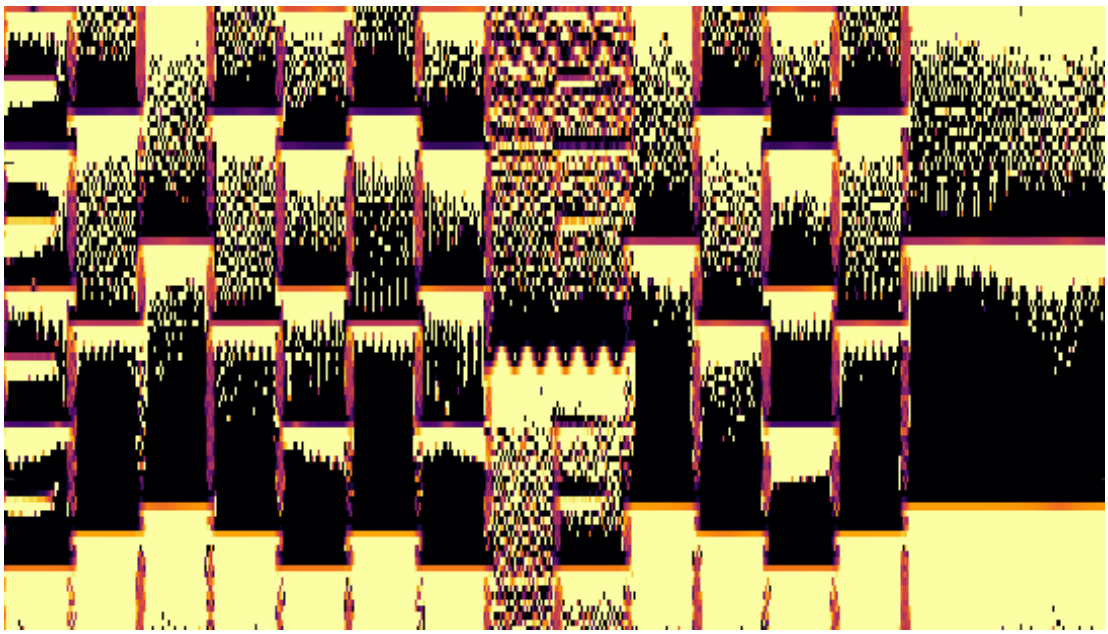

Figure 5.3: This is the front facade of the ARI building.

Figure 5.4: This is the office of Michael and me.

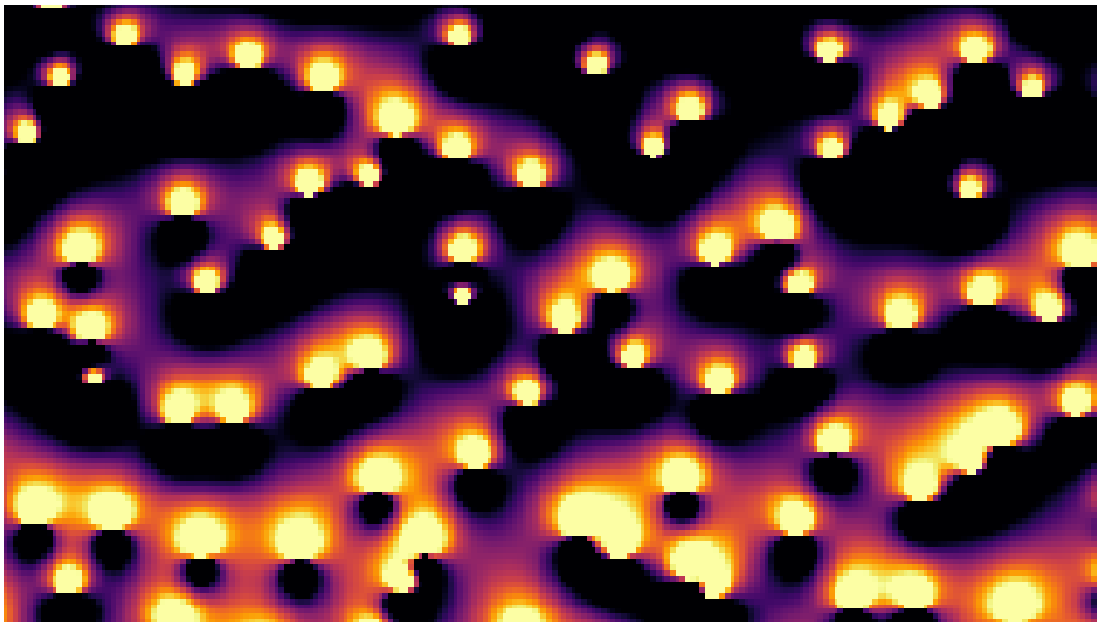

Figure 5.5: This is a group photo of all ARI members, except Konstantin who took the photo.

Figure 5.6: This is a panorama picture of the sunset at our institute's day trip to Wachau.