The Large Time Frequency Analysis Toolbox

Clara Hollomey, Nicki Holighaus
Institut für Schallforschung, Österreichische Akademie der Wissenschaften, 1040 Wien,
E-Mail: clara.hollomey@oeaw.ac.at
nicki.holighaus@oeaw.ac.at

Introduction

The Large Time Frequency Analysis Toolbox (LTFAT) [1] is a toolbox for audio signal processing written in Matlab/Octave and C/C++. It is licensed under the GNU Public License Version 3 and is freely available from github and as an official Octave-Forge package.

LTFAT aims at providing a common ground for research on time-frequency analysis. While there certainly is no shortage of signal processing toolboxes, LTFAT has a combination of features we believe is not commonly found elsewhere:

- Comprehensiveness: LTFAT covers all major time-frequency representations (Short-Time Fourier Transform, Discrete Gabor Transform, Discrete Wavelet Transform, filterbanks, frames)
- Flexibility: LTFAT allows for a high degree of freedom in the parametrization of those representations, enabling the design of representations like e.g. Constant-Q transforms and perceptually inspired filter banks
- Speed: Many LTFAT functions are implemented in C/C++, allowing for the efficient calculation of the above
- Link to current research: Many algorithms in LTFAT are linked to publications that can be accessed via the LTFAT homepage http://ltfat.org
- Extensive Documentation: In-code generated online documentation including code examples and demos
- Unrestricted availability: LTFAT is freely available from https://github.com/ltfat/ltfat

In the following, we give an overview on LTFAT's core functionality and resources for getting started.

Core functionality

Discrete Gabor Transform

For a signal $x \in C^L$ of length L, the Short-Time Fourier Transform (STFT), is defined as

$$X_m(\omega_k) = \sum_{n=0}^{L} \overline{g[n-m]}\, x[n] e^{-j\frac{2\pi kn}{M}} \tag{1}$$

with g the window and M the number of frequency channels used for the transform, equalling the signal length L.

The STFT yields a highly redundant time-frequency representation, with a signal of length L requiring L^2 coefficients for analysis. Not all of these are needed to arrive at a comprehensive and invertible time-frequency representation of a signal.

In LTFAT, the STFT as defined in (1) is considered a special case of its sampled version, the Discrete Gabor Transform (DGT)

$$X_m(\omega_k) = \sum_{n=0}^{L} \overline{g[n-am]}\, x[n] e^{-j\frac{2\pi kn}{M}} \tag{2}$$

where a and M can be chosen arbitrarily. By introducing a temporal subsampling factor a and choosing $M < L$, the redundancy of the STFT can be reduced as illustrated in Figure 1.

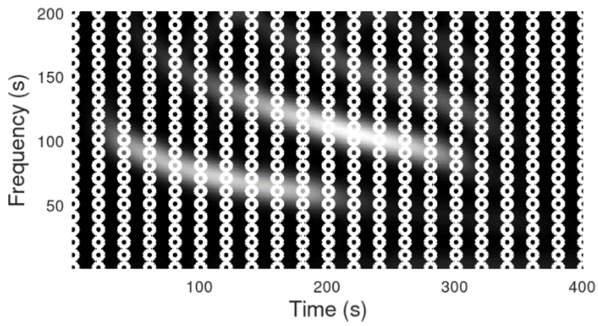

Figure 1: DGT coefficients contrasted with STFT coefficients for a bat signal. The white circles indicate the placement of the DGT coefficients on the absolute valued STFT.

LTFAT provides algorithms for the computation of the DGT for arbitrary time-frequency sampling factors, window types and window lengths [2], whereby the most efficient algorithms are automatically selected.

Non-stationary discrete Gabor Transform

The classical DGT provides a signal representation on a regular time-frequency grid where the window needs to be fixed for the whole time-frequency plane. Neither is desirable for processing non-stationary acoustic signals, such as music or speech.

LTFAT's non-stationary Gabor transform functions [3] allow for the creation of invertible time-frequency representations using arbitrary windows at the cost of increased redundancy and an irregular sampling grid over time as exemplarily depicted in Figure 2.

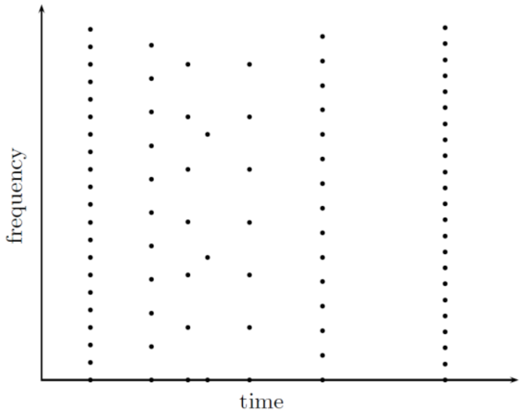

Figure 2: Example of a sampling grid of the time-frequency plane when building a decomposition with time-frequency resolution evolving over time. Figure taken from [3].

Similarly, irregular sampling over frequency can be used for arriving, for example, at quasi-logarithmically spaced representations, such as the Constant-Q transform depicted in Figure 3.

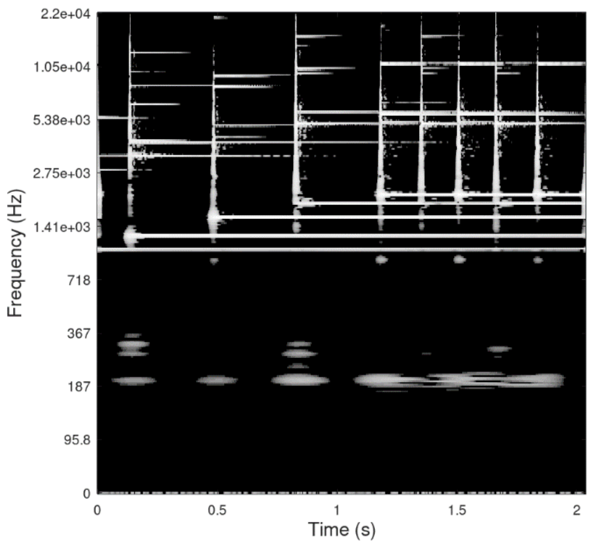

Figure 3: Example of the constant-Q transform of a glockenspiel signal as derived from LTFAT's non-stationary discrete Gabor framework.

Discrete Wavelet Transform

Depending on the application, it can be computationally more efficient and conceptually more straight-forward to approach time-frequency analysis from a filterbank point of view.

The non-stationary DGT, for example, can be used for deriving the discrete wavelet transform, but filter bank approaches are usually computationally more efficient.

Besides some commonly known algorithms such as Mallat's [4], LTFAT's wavelet module [5] provides functionality for computing arbitrary wavelet filter banks and packets, as well as functionality for converting one to the other.

Filterbanks

In addition, LTFAT also provides a general framework for constructing filterbanks. It comprises generator functions for auditory-inspired, constant-Q, Gabor, warped, and wavelet filters g_m. along with their associated subsampling factors a_m.

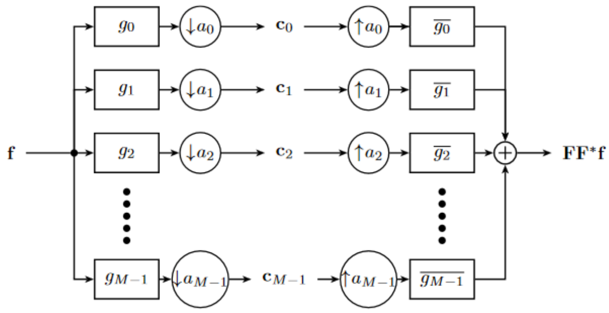

Figure 4: Analysis and synthesis stage of a filterbank given as a set of filters g_m and subsampling factors a_m. Figure taken from [5].

These filters and subsampling factors interface with LTFAT's filterbank base routines and signal processing module, allowing for the creation of uniform and non-uniform filterbanks.

Frames

Being able to achieve perfect reconstruction, i.e. the invertibility of time-frequency representations, is a common requirement in digital audio signal processing. It is ensured whenever a time-frequency representation forms a frame [6].

A frame is an extension of the mathematical concept of a vector basis. While bases are always linearly independent, a frame might be overcomplete, i.e. contain more vectors than are strictly necessary to span the time-frequency space of interest. Consequently, frames are not as straightforward to invert as systems forming a basis, but inversion can often be done with reasonable efficiency.

All of the above-mentioned approaches to time-frequency analysis can be parametrized such that they yield a frame.

With its frames framework, LTFAT offers an object-oriented means for interacting with various time-frequency representations –usually, but not necessarily forming a frame– in a unified way.

This can be useful for quick prototyping on the one hand, as well as for in-depth investigations of mathematical properties of time-frequency representations on the other.

The phase retrieval toolbox

The phase retrieval toolbox is an external module built with and depending on LTFAT. It can be downloaded from https://github.com/ltfat/phaseret and provides implementations of various algorithms for phase reconstruction from the STFT magnitude, some of them real-time capable, ranging from the well-known Griffin-Lim algorithm [7] to the more recent phase gradient heap integration [8].

Resources for getting started

The current release and installation instructions can be found under https://github.com/ltfat/ltfat/releases, along with the bug tracker and the development repository.

LTFAT's documentation is generated directly from the source code via mat2doc [9] and can be found under http://ltfat.org/doc/. A collection of LTFAT-associated notes and publications can directly be accessed from http://ltfat.org/notes/.

Summary and Outlook

LTFAT is a versatile tool for supporting new scientific developments in digital signal processing. It provides a tested and documented environment for the implementation of algorithms and can serve as a means for researchers to make their work available to a larger audience.

Besides ongoing additions complementing the existing code, such as a wavelet filterbank and the according prototypes, future work aims at enhancing the accessibility and usability of LTFAT. An interface to Python is currently in development, with the goal to provide increased accessibility to LTFAT's functionality for the growing community of Python users by allowing LTFAT functions being called directly from Python.

Resources

[1] Prusa et al..: The Large Time-Frequency Analysis Toolbox 2.0 in Sound, Music, and Motion, Lecture Notes in Computer Science, pp. 419–442. Springer International Publishing, 2014.

[2] Sondergaard, P..: Efficient Algorithms for the Discrete Gabor Transform with a long FIR window. Journal of Fourier Analysis and Applications 18(3) (2012)

[3] Balazs et al.: Theory, implementation and applications of non-stationary Gabor frames. Journal of Computational and Applied Mathematics (2014)

[4] S. Mallat.: A wavelet tour of signal processing. Academic Press, San Diego, CA, 1998.

[5] Prusa et al..: Discrete Wavelet Transforms in the Large Time-Frequency Analysis Toolbox for Matlab/GNU Octave. ACM Transactions on Mathematical Software 9(4) (2010)

[6] Christensen, O..: An Introduction to Frames and Riesz Bases, Springer International Publishing Switzerland, Basel (2016)

[7] D. Griffin and J.S. Lim.: Signal estimation from modified short-time Fourier transform, IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 32, no. 2, pp. 236–243 (1984)

[8] Prusa et al..: A non-iterative method for reconstruction of phase from STFT magnitude, IEEE/ACM Transactions on Audio, Speech, and Language Processing (2016)

[9] Documentation.: The mat2doc documentation system, http://ltfat.org/notes/ltfatnote023.pdf (2014)