Large Time-Frequency Analysis Toolbox 2.0

Jordy van Velthoven, Zdeněk Průša and Peter L. Søndergaard

Introduction

The Large Time-Frequency Analysis Toolbox (LTFAT) is a software toolbox for time-frequency analysis and synthesis. It is intended both as an educational and a computational tool. It consists of a large number of linear transforms for Fourier analysis, Gabor analysis and wavelet analysis along with associated operators and plotting routines. The frames framework functions here as a unifying layer for accessing the different linear transforms and their associated operators.

The LTFAT is started in 2003 by Peter L. Søndergaard. The current version of the toolbox is 2.0 and is freely available under terms of GPLv3 at `http://ltfat.sourceforge.net`. The toolbox works for the scripting languages Matlab and Octave and contains MEX/OCT interfaces written in C/C++ as a backend library.

The binary version of the LTFAT consists of the subdirectories that are listed below.

Description	Directory name
Fourier analysis	`fourier`
Gabor analysis	`gabor`
Wavelet analysis	`wavelet`
Filterbank routines	`filterbank`
Non-stationary Gabor analysis	`nonstatgab`
Frames framework	`frames`
Operator framework	`operator`
Supporting computational routines	`comp`
Signal processing	`sigproc`
Collection of signals	`signals`
Auditory related functions	`auditory`
Demo scripts	`demos`
Standalone source files written in C	`src`
Matlab MEX functions	`mex`
Octave C++ functions	`oct`

In the following sections the rationale behind the main categories of the LTFAT will be shortly explained and a selection of their files will be listed. Most of the categories of the LTFAT are extensively discussed in [1] and [2].

Frames

The frames framework consists of several types of frames and methods associated with frames. The frames are presented in an object-oriented framework. This object-oriented framework provides an operator-like interface for working with frames rather than explicitly creating frame matrices. Therefore the properties of a frame are related to the attributes of an object and the methods associated with a frame with the methods of an object.

Frame methods

Description	Function name
Creation	
Create a frame	`frame`
Create a frame pair	`framepair`
Create the canonical dual frame	`framedual`
Create the canonical tight frame	`frametight`
Accelerate computation	`frameaccel`
Linear operators	
Frame analysis	`frana`
Frame synthesis	`frsyn`
Iterative analysis	`franaiter`
Iterative synthesis	`frsyniter`
Plotting	
Plot frame coefficients	`plotframe`
Plot squared modulus of frame coefficients	`framegram`
Quering a frame	
Frame bounds	`framebounds`
Redundancy	`framered`
Admissible length	`framelength`
Non-linear analysis and synthesis	
Basis pursuit using the SALSA algorithm	`franabp`
Frame synthesis from magnitude of coefficients	`frsynabs`
LASSO threshholding	`franalasso`
Group LASSO thresholding	`franagrouplasso`

Types of frames

Description	Frame name
General frames	
Frame specified by matrix	`gen`
Canonical orthonormal basis	`identity`
Linear frequency scale	
Gabor frame	`dgt`
Wilson basis	`dwilt`
Windowed modified cosine basis	`wmdct`
Logarithmic frequency scale	
Fast wavelet transform	`fwt`
Undecimated fast wavelet transform	`ufwt`
Constant Q-transform	`cqt`
Adaptable frequency scale	
Filterbank	`filterbank`
Uniform filterbank	`ufilterbank`
Wavelet tree	`wfbt`
Wavelet packet tree	`wpfbt`
Adaptable time scale	
Non-stationary Gabor frame	`nsdgt`
Uniform non-stationary Gabor frame	`undsgt`
Pure freqency bases	
Unitary discrete Fourier transform	`dft`
Discrete cosine transforms	`dcti, dctii, dctiii, dctiv`
Discrete sine transforms	`dsti, dstii, dstiii, dstiv`
Container frames	
Fusion frame	`fusion`
Tensor frame	`tensor`

Fourier analysis

The Fourier analysis category consists of a collection of basic transforms and operators associated with Fourier analysis. It mainly consists of Fourier transforms with associated operators and generalizations of these such as the generalized Goertzel algorithm, chirped z-transform and fractional Fourier transform.

Description	Function name
Basic Fourier analysis	
Unitary discrete Fourier transform	`dft`
Inverse unitary discrete Fourier transform	`idft`
Generalized Goertzel algorithm	`gga`
Chirped z-transform	`chirpzt`
Cosine and sine transforms	
Discrete cosine transforms	`dcti,dctii, dctiii,dctiv`
Discrete sine transforms	`dsti,dstii, dstiii,dstiv`
Fractional Fourier transforms	
Discrete fractional Fourier transform	`dfracft`
Fast fractional Fourier transform	`ffracft`
Operations on periodic functions	
Involution	`involute`
Periodic convolution	`pconv`
Periodic cross-correlation	`pxcorr`
Periodic functions	
Periodic Hermite function	`pherm`
Periodic Gaussian function	`pgauss`

Gabor analysis

The Gabor analysis category consists of a collection of time-frequency transforms on both separable and non-seperable lattices, and includes several associated operators. The main time-frequency transforms are the Gabor, Wilson and modified discrete cosine transform.

Description	Function name
Basic time-frequency analysis	
Twisted convolution	`tconv`
Discrete symplectic Fourier transform	`dsft`
Zak transform	`zak`
Inverse Zak transform	`izak`
s0-norm	`s0norm`
Gabor systems	
Discrete Gabor transform	`dgt`
Inverse discrete Gabor transform	`idgt`
Iterative reconstruction from spectrogram	`isgram`
Evaluate Gabor window	`gabwin`
Wilson bases and modified cosine transforms	
Discrete Wilson transform	`dwilt`
Inverse discrete Wilson transform	`idwilt`
Modified discrete cosine transform	`wmdct`
Inverse modified discrete cosine transform	`iwmdct`
Evaluate Wilson window	`wilwin`
Reconstruction windows	
Canonical dual window	`gabdual`
Canonical tight window	`gabtight`
Window of Wilson orthonormal basis	`wilorth`
Riesz dual window of Wilson basis	`wildual`

Non-stationary Gabor analysis

The non-stationary Gabor analysis category includes generalizations of the Gabor systems. The non-stationary Gabor systems generalizes the standard Gabor systems, where the window function, the time step and the number of frequency channels are fixed; to systems with evolving properties over time.

Description	Function name
Non-stationary Gabor transforms	
Non-stationary Gabor transform	`nsdgt`
Inverse non-stationary Gabor transform	`insdgt`
Uniform non-stationary Gabor transform	`unsdgt`
Window construction	
Non-stationary dual windows	`nsgabdual`
Non-starionary tight windows	`nsgabtight`

Wavelet analysis

The wavelet analysis category consists of linear time-frequency transforms with an adaptable or logarithmic frequency scale in contrast to the linear transforms with an adaptable time scale or linear frequency scale of the sections on Gabor and non-stationary Gabor analysis, respectively.

Description	Function name
Wavelet transforms	
Fast wavelet transform	`fwt`
Inverse fast wavelet transform	`ifwt`
Undecimated fast wavelet transform	`ufwt`
Inverse undecimated fast wavelet transform	`ufwt`
Wavelet filterbanks	
Wavelet filterbank tree	`wfbt`
Inverse wavelet filterbank tree	`iwfbt`
Undecimated wavelet filterbank tree	`uwfbt`
Inverse undecimated wavelet filterbank tree	`iuwfbt`
Wavelet packet filterbank tree	`wpfbt`
Inverse wavelet packet filterbank tree	`iwpfbt`
Wavelet filters in time-domain	
Coiflet filters	`wfilt_coif`
Daubechies filters	`wfilt_db`
Biorthogonal spline wavelet filters	`wfilt_spline`
Dense grid framelets	`wfilt_dgrid`

Operators

The operator framework works as the frame framework with an operator-like interface rather than explicitly creating matrices. The main operators are the frame multipliers and the spreading operators.

Description	Function name
General operator	
Construct an operator	`operatornew`
Apply an operator	`operator`
Apply the inverse of an operator	`ioperator`
Apply the adjoint of an operator	`operatoradj`
Best approximation by operator	`operatorappr`
Eigenpairs of an operator	`operatoreigs`
Matrix representation of an operator	`operatormatrix`
Frame multipliers	
Apply frame multiplier	`framemul`
Apply inverse frame multiplier	`iframemul`
Apply the adjoint of a frame multiplier	`framemuladj`
Best approximation by frame multiplier	`framemulappr`
Eigenpairs of a frame multiplier	`framemuleigs`
Spreading operators	
Spreading operator	`spreadop`
Apply inverse spreading operator	`spreadinv`
Symbol of adjoint spreading operator	`spreadadj`
Symbol of operator expressed as a matrix	`spreadfun`
Eigenpairs of spreading operator	`spreadeigs`

Filterbanks

The filterbank category consists of a collection of computational routines for finite impulse response (FIR), frequency defined and band-limited filters. The filterbanks can, as the other linear transforms, be represented as a frame and are therefore strongly related to the other linear transforms in the LTFAT. The linear transforms that belong exclusively to the filterbank section are the constant Q-transform and ERBlet transform.

Description	Function name
General filterbanks	
Filterbank	`filterbank`
Uniform filterbank	`ufilterbank`
Inverse of normal or uniform filterbank	`ifilterbank`
Auditory inspired filterbanks	
Constant Q-transform	`cqt`
Inverse constant Q-transform	`icqt`
ERBlet transform	`erblett`
Inverse ERBlet transform	`ierblett`
Filter generators	
Logarithmically spaced filters	`cqtfilters`
ERB-spaced filters	`erbfilters`
Window construction	
Canonical dual filters	`filterbankdual`
Canonical tight filters	`filterbanktight`
Total frequency response	`filterbankresponse`

Block processing

The block processing framework is a self-contained framework within the LTFAT for real-time audio processing in Matlab and Octave. Together with the time-frequency analysis and synthesis capabilities of the LTFAT, it allows audio processing in the transform domain.

Description	Function name
Basic methods	
Construct a block-stream	`block`
Read samples into block	`blockread`
Play samples from a block	`blockplay`
Append block to a file	`blockwrite`
Control panel	`blockpanel`
Block-adapted transforms	
Block analysis	`blockana`
Block synthesis	`blocksyn`

LTFAT related

There are several sources related or connected to the LTFAT. The most important are the documentation, LTFAT note series and mat2doc documentation system. All these sources can be found on the homepage of the LTFAT.

- Documentation: There are several forms in which the LTFAT is documented. The most complete and up-to-date description of the toolbox is the auto-generated documentation on the LTFAT homepage. The online documentation is generated from the headers of the M-files of the functions and models that are included in the LTFAT. The documentation of the M-files is also available as a reference manual.
- The LTFAT note series: The LTFAT note series is a collection of texts that have a relation to the LTFAT. A LTFAT note can be anything from a preprint of a journal paper, a technical report or a thesis. The source of all the LTFAT notes are publically available.
- The mat2doc documentation system: Mat2doc is a system, written in Python as a wrapper around reStructuredText, for publishing documentation extracted from Matlab or Octave function headers. The documentation system creates documentation in several formats. The various formats that are currently supported are php, HTML, LaTeX and plain text. For example, all the online documentation and the reference manual of the LTFAT are generated trough the mat2doc documentation system.

References

[1] P. L. Søndergaard, B. Torrésani, P. Balazs. *The Linear Time-Frequency Analysis Toolbox.* International Journal of Wavelets, Multiresolution Analysis and Information, 10(4), 2012.

[2] Z. Průša, P. L. Søndergaard, N. Holighaus, C. Wiesmeyr, P. Balazs. *The Large Time-Frequency Analysis Toolbox 2.0.* (preprint), 2014. http://ltfat.sourceforge.net/notes/ltfatnote030.pdf