Requirements for LTFAT

written by Peter L. Søndergaard
## Contents

1 Requirements for LTFAT

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 LTFAT master requirement</td>
<td>4</td>
</tr>
<tr>
<td>1.2 Fundamental properties</td>
<td>4</td>
</tr>
<tr>
<td>1.2.1 Available</td>
<td>4</td>
</tr>
<tr>
<td>1.2.2 Fast</td>
<td>4</td>
</tr>
<tr>
<td>1.2.3 Free as in speech</td>
<td>5</td>
</tr>
<tr>
<td>1.2.4 Gratis / Free as in beer</td>
<td>5</td>
</tr>
<tr>
<td>1.2.5 Purpose</td>
<td>5</td>
</tr>
<tr>
<td>1.2.6 Usable</td>
<td>5</td>
</tr>
<tr>
<td>1.3 User Friendly</td>
<td>5</td>
</tr>
<tr>
<td>1.3.1 Demos</td>
<td>5</td>
</tr>
<tr>
<td>1.3.2 Design Specification</td>
<td>6</td>
</tr>
<tr>
<td>1.3.3 Documented</td>
<td>6</td>
</tr>
<tr>
<td>1.3.4 Graphical User Interface</td>
<td>6</td>
</tr>
<tr>
<td>1.3.5 Parameter handling</td>
<td>6</td>
</tr>
<tr>
<td>1.3.6 Tutorial</td>
<td>6</td>
</tr>
<tr>
<td>1.3.7 User friendly</td>
<td>6</td>
</tr>
<tr>
<td>1.4 Open Source</td>
<td>7</td>
</tr>
<tr>
<td>1.4.1 GNU Public Licence</td>
<td>7</td>
</tr>
<tr>
<td>1.4.2 Homepage</td>
<td>7</td>
</tr>
<tr>
<td>1.4.3 Sourceforge</td>
<td>7</td>
</tr>
<tr>
<td>1.5 Specific functionality</td>
<td>7</td>
</tr>
<tr>
<td>1.5.1 Auditory functions</td>
<td>7</td>
</tr>
<tr>
<td>1.5.2 Filterbanks</td>
<td>7</td>
</tr>
<tr>
<td>1.5.3 Fourier analysis</td>
<td>8</td>
</tr>
<tr>
<td>1.5.4 Frame analysis</td>
<td>8</td>
</tr>
<tr>
<td>1.5.5 Gabor analysis</td>
<td>8</td>
</tr>
<tr>
<td>1.5.6 Non-stationary Gabor frames</td>
<td>8</td>
</tr>
<tr>
<td>1.5.7 Quadratic distributions</td>
<td>8</td>
</tr>
<tr>
<td>1.5.8 Test signals</td>
<td>9</td>
</tr>
<tr>
<td>1.5.9 Signal processing utilitites</td>
<td>9</td>
</tr>
<tr>
<td>1.5.10 Spectral Analysis</td>
<td>9</td>
</tr>
<tr>
<td>1.5.11 Streaming framework</td>
<td>9</td>
</tr>
<tr>
<td>1.5.12 Totally Positive functions</td>
<td>9</td>
</tr>
<tr>
<td>1.5.13 Selected Topics</td>
<td>9</td>
</tr>
<tr>
<td>1.5.14 Wavelet analysis</td>
<td>10</td>
</tr>
<tr>
<td>1.5.15 YAWTB</td>
<td>10</td>
</tr>
</tbody>
</table>
Chapter 1

Requirements for LTFAT

These are the top-level requirements for LTFAT.

1.1 LTFAT master requirement

Description: LTFAT must be usable.
Rationale: The world needs a good time-frequency toolbox.
Solved by: 1.2.1 Available, 1.2.2 Fast, 1.2.3 Free as in speech, 1.2.4 Gratis / Free as in beer, 1.2.5 Purpose, 1.2.6 Usable
Status: finished

1.2 Fundamental properties

These are the fundamental properties and values of LTFAT.

1.2.1 Available

Description: LTFAT must be available
Rationale: LTFAT must be available for the users on the platforms they work on.
Depends on: 1.1 LTFAT master requirement
Solved by: 1.4.2 Homepage, 1.8.8 Platform Support
Status: finished

1.2.2 Fast

Description: LTFAT must be fast.
Rationale: If it is not fast, it will not be used.
Depends on: 1.1 LTFAT master requirement
Solved by: 1.7.1 Backend
Status: finished
1.3. USER FRIENDLY

1.2.3 Free as in speech
Description: LTFAT must free as in speech.
Rationale: LTFAT should be used to benefit other free and open source software projects
Depends on: 1.1 LTFAT master requirement
Solved by: 1.4.1 GNU Public Licence
Status: finished

1.2.4 Gratis / Free as in beer
Description: LTFAT must be available for free.
Rationale: Good mathematics software must be available for people that cannot pay any amount of money for the software.
Depends on: 1.1 LTFAT master requirement
Solved by: 1.4.1 GNU Public Licence, 1.8.1 Octave, 1.8.5 Octave
Status: finished

1.2.5 Purpose
Description: LTFAT must have a relevant purpose.
Rationale: A purpose is the thing to have
Depends on: 1.1 LTFAT master requirement
Solved by: 1.6.1 Academic, 1.5.13 Selected Topics
Status: finished

1.2.6 Usable
Description: The user must be able to use LTFAT
Rationale: The user must be able to use LTFAT
Depends on: 1.1 LTFAT master requirement
Solved by: 1.3.7 User friendly, 1.7.5 Working
Status: finished

1.3 User Friendly
User friendly

1.3.1 Demos
Description: LTFAT must include demos
Rationale: Demos (demonstrationg scripts) gives the user a quick way to get started on a topic
Depends on: 1.3.7 User friendly
Solved by: 1.5.1 Auditory functions, 1.5.8 Test signals, 1.5.9 Signal processing utilitites
Status: finished
1.3. USER FRIENDLY

1.3.2 Design Specification

Description: **LTFAT must** have a design specification
Rationale: A design specification ensures a uniform user interface
Depends on: 1.3.7 User friendly
Status: finished

1.3.3 Documented

Description: The **LTFAT must** be well documented
Rationale: People need good documentation because the toolbox is complicated to understand
Depends on: 1.3.7 User friendly
Solved by: 1.9.5 Mat2doc
Status: finished

1.3.4 Graphical User Interface

Description: **LTFAT must** have a simple GUI
Rationale: There must be an easy way for new users to get started
Depends on: 1.3.7 User friendly
Solved by: 1.9.3 GPC
Status: finished

1.3.5 Parameter handling

Description: **LTFAT must** handle parameters in a user-friendly way
Rationale: Learning and working with the toolbox is much easier if there is a standardized and user-friendly way of working with optional parameters. This is solved by the ltfatarghelper function in Matlab and Octave
Depends on: 1.3.7 User friendly
Status: finished

1.3.6 Tutorial

Description: **LTFAT must** have a tutorial
Rationale: There must be an easy way for new users to get started
Depends on: 1.3.7 User friendly
Status: finished

1.3.7 User friendly

Description: **LTFAT must** be user friendly.
Rationale: If it is not user friendly, it will not be used.
Depends on: 1.2.6 Usable
Solved by: 1.3.1 Demos, 1.3.2 Design Specification, 1.3.3 Documented, 1.3.4 Graphical User Interface, 1.3.5 Parameter handling, 1.3.6 Tutorial
Status: finished
1.4 Open Source

These requirements cover why and how LTFAT is open source

1.4.1 GNU Public Licence

**Description:** LTFAT must be licenced under the GNU Public License

**Rationale:** Good reasons are listed here https://www.gnu.org/licenses/why-not-lgpl.html. Octave and FFTW are also GPL.

**Depends on:** 1.9.2 FFTW, 1.2.3 Free as in speech, 1.2.4 Gratis / Free as in beer, 1.8.5 Octave

**Status:** finished

1.4.2 Homepage

**Description:** The LTFAT must have a homepage

**Rationale:** It must be possible to find everything about LTFAT from one homepage (or at least links from there)

**Depends on:** 1.2.1 Available

**Solved by:** 1.9.5 Mat2doc, 1.4.3 Sourceforge

**Status:** finished

1.4.3 Sourceforge

**Description:** The homepage must use Sourceforge

**Rationale:** The development and download part of the homepage should be available on a publicly accessible server

**Depends on:** 1.4.2 Homepage

**Status:** finished

1.5 Specific functionality

These requirements specify areas of functionality/topics that LTFAT must cover

1.5.1 Auditory functions

**Description:** LTFAT must include basic auditory functions

**Rationale:** LTFAT should cover a limited subset of very basic auditory models to be able to create Erb-scale filterbanks etc.

**Depends on:** 1.3.1 Demos, 1.6.7 Paper about the Erblet-transform

**Status:** finished

1.5.2 Filterbanks

**Description:** LTFAT must include a filterbank section

**Rationale:** Filterbanks are a useful when Gabor or Wavelet analysis are not sufficient, and the creating of good filterbanks is far from trivial. LTFAT must include easy to use functions to create dual and tight filterbanks, and methods for easy creating of filters that are correctly spaced and weighted.

**Depends on:** 1.5.13 Selected Topics

**Status:** finished
1.5.3 Fourier analysis

**Description:** *LTFAT must* include Fourier analysis

**Rationale:** LTFAT should cover a subset of Fourier analysis.

**Depends on:** 1.5.13 Selected Topics

**Solved by:** 1.5.10 Spectral Analysis

**Status:** finished

1.5.4 Frame analysis

**Description:** *LTFAT must* include a frame analysis framework

**Rationale:** An object-oriented framework implementing frames will make it easy for the user to try out different frames in an application, and it will make it easier for LTFAT developers to formulate high-level algorithms that work for all (or just some) frames.

**Depends on:** 1.5.13 Selected Topics

**Status:** finished

1.5.5 Gabor analysis

**Description:** *LTFAT must* include Gabor analysis

**Rationale:** LTFAT should cover a subset of Gabor analysis.

**Depends on:** 1.5.13 Selected Topics

**Solved by:** 1.5.12 Totally Positive functions

**Status:** finished

1.5.6 Non-stationary Gabor frames

**Description:** *LTFAT must* non-stationary Gabor frames

**Rationale:** Non-stationary Gabor frames are the generalization of Gabor frames to non-equidistant time steps. LTFAT must include them to support recent research and to make it possible to mature the field of signal processing in this area, as the details of the construction are far from trivial. LTFAT must include easy to use functions to create dual and tight non-stationary Gabor frames, and methods for easy creating of filters that are correctly spaced and weighted.

**Depends on:** 1.6.4 Paper about nonstationary Gabor systems, 1.5.13 Selected Topics

**Status:** finished

1.5.7 Quadratic distributions

**Description:** *LTFAT must* contain quadratic distributions

**Rationale:** LTFAT should cover a subset of quadratic distributions. Candidates could be:

- The Wigner distribution
- The Choi-Williams distribution

Different versions of the distributions should be created depending on if they are considered in the purely finite, discrete case, or if they are thought of as approximations of the continuous case. Even though quadratic distributions are rarely used in application, they are important instruments for teaching and visualizations.

**Depends on:** 1.5.13 Selected Topics

**Status:** not done
1.5. SPECIFIC FUNCTIONALITY

1.5.8 Test signals
Description: *LTFAT must* include a limited set of test signals
Rationale: LTFAT should include a small subset of test signals to make it possible to write interesting demos and examples
Depends on: 1.3.1 Demos
Status: finished

1.5.9 Signal processing utilities
Description: *LTFAT must* include a limited set of signal processing functions
Rationale: LTFAT should include a small subset of signal manipulation functions to make it possible to write demos and examples without loosing the focus of the demo.
Depends on: 1.3.1 Demos
Status: finished

1.5.10 Spectral Analysis
Description: *LTFAT must* contain function for classical spectral analysis
Rationale: It should be possible to use LTFAT to perform an easy spectral analysis of a signal by looking at its spectrum, power (or energy) spectral density functions etc. Several classes of input signals should be supported, both finite, discrete signals and also sampled signals embedded in the finite, discrete setting.
Depends on: 1.5.3 Fourier analysis
Status: not done

1.5.11 Streaming framework
Description: *LTFAT must* include a streaming framework
Rationale: An simple set of methods using the frames framework and interfacing to the sound processing capabilities of the underlying system in a platform independent way will make it much simpler for students and researchers to experiment with streaming algorithms.
Depends on: 1.5.13 Selected Topics
Status: finished

1.5.12 Totally Positive functions
Description: *LTFAT must* totally positive window functions
Rationale: It should be possible to create suitable totally positive FIR windows and their dual windows to be used for Gabor analysis
Depends on: 1.5.5 Gabor analysis
Status: not done

1.5.13 Selected Topics
Description: *LTFAT must* cover selected topics
Rationale: LTFAT should cover certain topics
Depends on: 1.2.5 Purpose
1.6. ACADEMIC REQUIREMENTS

Solved by: 1.5.2 Filterbanks, 1.5.3 Fourier analysis, 1.5.4 Frame analysis, 1.5.5 Gabor analysis, 1.5.6 Non-stationary Gabor frames, 1.5.7 Quadratic distributions, 1.5.11 Streaming framework, 1.5.14 Wavelet analysis
Status: finished

1.5.14 Wavelet analysis
Description: LTFAT must include Wavelet analysis
Rationale: LTFAT should cover a subset of Wavelet analysis:
Depends on: 1.5.13 Selected Topics
Solved by: 1.5.15 YAWTB
Status: finished

1.5.15 YAWTB
Description: LTFAT must include selected function from YAWTB
Rationale: It has been agreed to merge selected parts of YAWTB into LTFAT
Depends on: 1.5.14 Wavelet analysis
Status: not done

1.6 Academic requirements

These are the academic requirements for LTFAT.

1.6.1 Academic
Description: LTFAT must support selected papers
Rationale: Major parts of the software from these papers should be integrated into LTFAT, or LTFAT should contain functionality to support the software from these papers.
Depends on: 1.2.5 Purpose
Solved by: 1.6.3 LTFAT 1.0 paper, 1.6.4 Paper about nonstationary Gabor systems, 1.6.5 Paper about non-separable lattices, 1.6.6 Ph.d. thesis of Z. Prusa, 1.6.7 Paper about the Erblet-transform, 1.6.8 Paper about the Erblet-transform, 1.6.9 LTFAT 2.0 book chapter, 1.6.10 Paper about the generalized Goertzel transform
Status: finished

1.6.2 Paper about factorization algorithms
Description: LTFAT must contain the algorithms in ltfatnote011
Rationale: LTFAT must contain the factorization algorithm for the DGT from "Efficient Algorithms for the Discrete Gabor Transform with a Long FIR Window" by Soendergaard
Depends on: 1.6.3 LTFAT 1.0 paper
Solved by: 1.7.4 Reliable timings
Status: finished
1.6.3 LTFAT 1.0 paper

Description: LTFAT must contain the functionality from Ltfatnote015
Rationale: LTFAT must contain all the functionality mentioned in the LTFAT 1.0 paper: "The Linear Time Frequency Analysis Toolbox" by Soendergaard, Torresani and Balazs.
Depends on: 1.6.1 Academic
Solved by: 1.6.2 Paper about factorization algorithms
Status: finished

1.6.4 Paper about nonstationary Gabor systems

Description: LTFAT must contain the algorithms in ltfatnote018
Rationale: LTFAT must contain the algorithms about non-stationary Gabor systems from the paper "Nonstationary Gabor frames"
Depends on: 1.6.1 Academic
Solved by: 1.5.6 Non-stationary Gabor frames
Status: finished

1.6.5 Paper about non-separable lattices

Description: LTFAT must contain the algorithms in ltfatnote019
Rationale: LTFAT must contain the algorithms about non-separable lattices from the paper "Efficient algorithms for discrete Gabor transforms on a nonseparable lattice”.
Depends on: 1.6.1 Academic
Solved by: 1.7.4 Reliable timings
Status: finished

1.6.6 Ph.d. thesis of Z. Prusa

Description: LTFAT must contain the algorithms in ltfatnote026
Rationale: LTFAT must contain the segmented Wavelet transform.
Depends on: 1.6.1 Academic
Status: finished

1.6.7 Paper about the Erblet-transform

Description: LTFAT must contain the algorithms in ltfatnote027
Rationale: LTFAT must contain the algorithms about the Erblet-transform from the paper "The ERBlet transform: An auditory-based time-frequency representation with perfect reconstruction”
Depends on: 1.6.1 Academic
Solved by: 1.5.1 Auditory functions
Status: finished

1.6.8 Paper about the Erblet-transform

Description: LTFAT must contain the algorithms in ltfatnote029
Rationale: LTFAT must contain the algorithms about the Gabor dual windows by convex optimization from the paper "Designing Gabor windows using convex optimization”
Depends on: 1.6.1 Academic
1.7 Infrastructure

These requirements cover the infrastructure needed to develop, host and distribute LTFAT

1.7.1 Backend

Description: *LTFAT must* be upon a backend in C or C++.
Rationale: If it is not fast, it will not be used.
Depends on: 1.2.2 Fast, 1.7.4 Reliable timings
Solved by: 1.9.2 FFTW, 1.9.4 Lapack, 1.9.6 Portaudio
Status: finished

1.7.2 Testing Robot

Description: *LTFAT must* have a testing robot
Rationale: A automated system that runs the test suite on all the supported platforms saves a lot of very tedious work.
Depends on: 1.7.5 Working
Status: not done

1.7.3 Testing Suite

Description: *LTFAT must* have a testing suite
Rationale: A testing suite that covers all functions helps find bugs, regressions and makes it easier to port to new architectures.
Depends on: 1.7.5 Working
Status: finished
1.8. PLATFORMS REQUIREMENTS

1.7.4 Reliable timings

**Description:** Selected algorithms must be accurately timeable.

**Rationale:** Certain papers make statements about the speed of algorithms included in LTFAT. There must be a timing framework to accurately time these algorithms.

**Depends on:** 1.6.2 Paper about factorization algorithms, 1.6.5 Paper about non-separable lattices

**Solved by:** 1.7.1 Backend

**Status:** finished

1.7.5 Working

**Description:** LTFAT must work.

**Rationale:** It is useless if it does not work.

**Depends on:** 1.2.6 Usable

**Solved by:** 1.7.2 Testing Robot, 1.7.3 Testing Suite

**Status:** finished

1.8 Platforms requirements

These are the supported platforms that LTFAT must always work on

1.8.1 Octave

**Description:** LTFAT must work on Linux

**Rationale:** LTFAT must work on Linux because it is the most used, free operating system.

**Depends on:** 1.2.4 Gratis / Free as in beer, 1.8.8 Platform Support

**Status:** finished

1.8.2 Mac

**Description:** LTFAT must work on Mac

**Rationale:** LTFAT must work on Mac because this operating system is used by many researchers on their laptops

**Depends on:** 1.8.8 Platform Support

**Status:** finished

1.8.3 Matlab

**Description:** LTFAT must work in Matlab

**Rationale:** Many people use Matlab and it is generally faster than Octave

**Depends on:** 1.8.8 Platform Support

**Solved by:** 1.8.4 Matlab oldest version

**Status:** finished
1.8. PLATFORMS REQUIREMENTS

1.8.4 Matlab oldest version

Description: *LTFAT must* work in Matlab 2009b
Rationale: This is the oldest version we support. Older versions will not work because we use is output parameters
Depends on: 1.8.3 Matlab
Status: finished

1.8.5 Octave

Description: *LTFAT must* work in Octave
Rationale: Many people cannot afford a commercial software solution. We should support the growth of Octave by expanding the software available for Octave.
Depends on: 1.2.4 Gratis / Free as in beer, 1.8.8 Platform Support
Solved by: 1.4.1 GNU Public Licence, 1.8.6 Octave oldest version, 1.8.7 Octave newest version
Status: finished

1.8.6 Octave oldest version

Description: *LTFAT must* work in Octave 3.6
Rationale: This is the oldest Octave version we support. Older versions will not work because we use the tilde symbol in output parameters.
Depends on: 1.8.5 Octave
Status: finished

1.8.7 Octave newest version

Description: *LTFAT must* work in Octave 3.8
Rationale: This is the current stable Octave version
Depends on: 1.8.5 Octave
Status: finished

1.8.8 Platform Support

Description: *LTFAT must* support common computing platforms
Rationale: LTFAT must be available for the users on the platforms they work on.
Depends on: 1.2.1 Available
Solved by: 1.8.1 Octave, 1.8.2 Mac, 1.8.3 Matlab, 1.8.5 Octave, 1.8.9 Python, 1.8.10 Windows
Status: finished

1.8.9 Python

Description: *LTFAT must* work in Python
Rationale: Python is the next big thing in numerical computations
Depends on: 1.8.8 Platform Support
Status: not done
1.9. SOFTWARE USED BY LTFAT

1.8.10 Windows
Description: LTFAT must work on Windows
Rationale: LTFAT must work on Windows because it is the most used operating system.
Depends on: 1.8.8 Platform Support
Status: finished

1.9 Software used by LTFAT

These are the software requirements of LTFAT.

1.9.1 BLAS
Description: LAPACK requires BLAS
Rationale: BLAS is a requirement for LAPACK
Depends on: 1.9.4 Lapack
Status: finished

1.9.2 FFTW
Description: The backend must use FFTW
Rationale: FFTW is the fastest, open source, cross-platform FFT library available.
Depends on: 1.7.1 Backend
Solved by: 1.4.1 GNU Public Licence
Status: finished

1.9.3 GPC
Description: The GUI must use GPC
Rationale: GPC (General Polygon Clipper) is an open-source (but not GPL) that can compute set intersection etc. It is used by the GUI
Depends on: 1.3.4 Graphical User Interface
Status: finished

1.9.4 Lapack
Description: The backend must use LAPACK
Rationale: LAPACK is the fastest, open source, cross-platform linear algebra available.
Depends on: 1.7.1 Backend
Solved by: 1.9.1 BLAS
Status: finished

1.9.5 Mat2doc
Description: The LTFAT must use mat2doc
Rationale: mat2doc creates beautiful documentation
Depends on: 1.3.3 Documented, 1.4.2 Homepage
Status: finished
1.9. SOFTWARE USED BY LTFAT

1.9.6 Portaudio

**Description:** The *backend must* use *Portaudio*

**Rationale:** Portaudio is the best cross-platform, open-source audio library available.

**Depends on:** 1.7.1 Backend

**Status:** finished
Chapter 2

Graphical overview

2.1 Figure 1

The following figure shows a graphical overview of the relationships between the LTFAT requirements.

2.2 Figure 2 - grouped

This following figure shows the LTFAT requirements grouped by topics. The graph is identical to the one on Figure 1, only the layout is different.
2.2. **FIGURE 2 - GROUPED**

![Diagram Image]

---

18