

# ELAN - MATLAB Toolbox

Tommi Himberg

December 31, 2014

# 1 Introduction

This set of MATLAB tools is used for plotting and analysing observation annotations produced in ELAN - Linguistic Annotator<sup>1</sup> (Max Planck Institute for Psycholinguistics). Using ELAN, researchers can annotate audio and/or video data, e.g. based on the types of behaviours or actions observed in participants in the video. While ELAN lets the user to extract some descriptive statistics of the annotations (total times, number of annotations, average durations etc.), it is mostly a tool for annotation, not analysis, and thus researchers have to export the data from ELAN to other environments for visualisation and analysis.

The purpose of this toolbox is to provide MATLAB functions for importing ELAN-data into MATLAB, and to help visualise the structure of the annotations, as well as conduct more in-depth analyses of the annotations. The toolbox utilises functions and ideas implemented in SALEM Toolbox developed in the University of Bielefeld<sup>2</sup>.

In this manual, an example workflow is presented as a tutorial. An ELAN file containing annotations in multiple tiers is imported into MATLAB, some of its tiers plotted, and basic statistics extracted and processed further.

## 1.1 System requirements

This toolbox runs in MATLAB (Mathworks), so you will need it to be installed in your system. These functions were developed in MATLAB R2104b for Mac, and tested in XX (PC). They should work also in Linux, although this was not tested, and also in older MATLAB versions. Plotting and graphics were changed for the R2014b version, and so the results in the older versions might not be satisfactory, especially if the tiers contain lots of very short annotations, as the older graphics engine .

## 1.2 Downloading and installing

The toolbox can be downloaded in GitHub: <http://github.com/tijh/ELAN-MATLAB>

If you download the zip file, it includes the functions, this manual and a sample ELAN data that is used in the tutorial below. Extract the zip into a folder where you keep MATLAB functions, and add the folder into MATLAB's path (Set path).

# 2 List of functions

- `elanPlotColor.m`
- `elanRead.m`
- `elanTotalTimes.m`
- `elanValues.m`
- `elanAudio.m`

---

<sup>1</sup><http://tla.mpi.nl/tools/tla-tools/elan/>

<sup>2</sup><https://github.com/marc-hanheide/SALEM/releases>

## 3 Tutorial

### 3.1 Using the tierplotting script

In MATLAB, processes that contain multiple steps can be saved as scripts. A script is a MATLAB file (file extension .m) that contains the commands needed for the process, along with notes and instructions etc. You can execute all of the script at once, or one step at a time, skipping unnecessary steps if necessary.

Included in the zip-package, is `tierplotting.m`, a script for reading in ELAN data, choosing the tier to plot, assigning colors to value labels and plotting the tier. It contains the workflow typically used for creating plots of annotation data, and it is easy to adapt for individual projects mainly just by changing filenames etc. These, as well as those portions of the script that do the work and should not be edited, are clearly marked in the script.

### 3.2 Importing ELAN files

One of the best ideas in the SALEM toolbox was the ability to read ELAN native data formats (.eaf). This means that the user does not need to export the data in order to be able to bring it to MATLAB. This not only reduces the possibility of errors in changing file formats, it also helps reduce the number of different files needed, and also allows for a more streamlined work-flows, where even unfinished annotation files can be read into MATLAB (e.g. for plotting some tiers), while continuing to work on the same files in ELAN (e.g. annotating further tiers).

### 3.3 Exploring the file contents

### 3.4 Data cleanup

### 3.5 Plotting tiers

### 3.6 Descriptive statistics

### 3.7 Extracting metrics for further analysis

## 4 Function reference

### 4.1 elanRead

`elan=elanRead(fn, prevElan, offset)`

arguments: `fn`: filename of elan file as string (can also be a cell array of filename-strings) `prevElan`: internal, do not use! `offset`: internal argument, do not use!

parses the given ELAN file into a struct of tiers and timeslots. The timeslots are mostly useless, while the tiers is a struct with all annotation according to the tiers, with computed start and end times and values. Additionally two tiers "ValidAnnotation" and "ElanFile" are added to support slicing of files (see help `elanSlice`). If there are linked media files or linked csv files, they will be loaded as well but only if you are working with a single file.

usage: `elan=elanReadFile('example.eaf')`

supports loading of several files at once: `elan=elanReadFile('example.eaf','VP01-face-1.eaf');`

## 4.2 elanTotalTimes

Extracts the total durations per annotation label in a tier of ELAN file.

`function output = elanTotalTimes(elan, tier, abs)`

Inputs:

`elan` = Matlab structure with ELAN data `tier` = name of tier `abs` = results either in absolute times (0) or proportion of total annotated (1) or proportion of total time in tier (2) (default = 1)

Output:

Data structure with `output.labels` = annotation labels and `output.times` = corresponding durations / proportions.

Example:

`mytimes = elanTotalTimes(mydata,'P1_sound',2);`

This produces structure `mytimes`, reads the `P1_sound` tier from `mydata`, which is a MATLAB structure produced e.g. by `elanReadFile`. Option 2 gives the times as proportion of total duration of the tier.

Uses the data structure of SALEM Toolbox, so the structure needs the target tier and `AnnotationValid` and `ElanFile` tiers.

## 4.3 elanValues

`OUT = elanValues(elan, tier, order)`

Extract all unique annotation values from an ELAN file tier.

`OUT = elanValues(elan, tier, order);`

INPUT ARGUMENTS: `elan` = elan file structure `tier` = name of tier `order` = alphabetical (1, default) or order in which they occur in the tier (2).

OUTPUT ARGUMENTS: `OUT` = cell structure with annotation values from the tier.

Example: `vals = elanValue(myelan,'TierName',1);` This produces a N\*1 cell array with all annotation values used in tier 'TierName', in alphabetical order.

Uses data structure of the SALEM 0.1beta toolbox.

## 4.4 elanPlotColor

`elanPlot(elan, codes, colors, optionaltitle)`

Plots annotations of elan files tier-sorted on a time line and colors the annotations according to their annotation value (equal annotations are colored equally) Variable '`codes`' is a container.Map structure with annotation values and their keys. Variable '`colors`' is an array with color codes (n,3), from which the colors are picked to represent annotations.