

THE MUSICAL GESTURES TOOLBOX FOR MATLAB

Alexander Refsum Jensenius

RITMO Centre for Interdisciplinary Studies in Rhythm, Time and Motion,
Department of Musicology, University of Oslo

ABSTRACT

The Musical Gestures Toolbox for Matlab (MGT) aims at assisting music researchers with importing, preprocessing, analyzing, and visualizing video, audio, and motion capture data in a coherent manner within Matlab.

1. INTRODUCTION

As the interest in studying music-related body motion has increased in recent years, many researchers are facing challenges of handling their multimodal data sets. Such data sets typically include audio, video, and motion capture (mocap) recordings, and possibly also sensor data, score material, questionnaire results and qualitative annotations. The challenge, then, is to find solutions for visualising and analysing the data in a coherent and consistent manner. For non-programming researchers, there are some tools to assist researchers, including EyesWeb,¹ RepoVizz,² and SonicVisualiser,³ to name but a few. Within the MIR community there are also numerous libraries and frameworks for different programming languages.

Rather than taking on all the challenges involved in making *the* unified tool for everything, we have focused on bridging the gap between two widely used Matlab toolboxes, the MIR Toolbox [5], and the MoCap Toolbox [1]. These allow for analysing and visualising audio and motion capture (mocap) data, respectively. There is, however, no similar toolbox for working with video recordings, and there are no easy ways to handle these three types of data (audio, video, mocap) together. The Musical Gestures Toolbox for Matlab (MGT) aims at solving these two problems, by (1) providing a complete set of functions for working with video visualisation (and some analysis), and (2) providing a structure and some useful functions for working with these video visualisations together with the tools available in the MoCap and MIR Toolboxes, respectively.⁴

¹ http://www.infomus.org/eyesweb_eng.php

² <https://repovizz.upf.edu>

³ <http://sonicvisualiser.org/>

⁴ Toolbox available at <https://github.com/fourMs/MGT>.

Many of the video tools available in MGT are ‘ported’ from similar functions available in the Musical Gestures Toolbox for Cycling ’74’s Max [4], which is currently available as part of the Jamoma environment.⁵ These Max tools work well for real-time processes, and have been developed to also partly work for non-real-time processing. For more scientific applications, however, Matlab is a better environment for creating visualizations and animations. In the rest of the paper we will explain the structure of the toolbox and its functions.

2. OVERVIEW OF THE MGT

Figure 1 shows a sketch of how MGT can be integrated with the MoCap Toolbox and MIR Toolbox, thereby allowing for working with audio, video and mocap data in parallel.

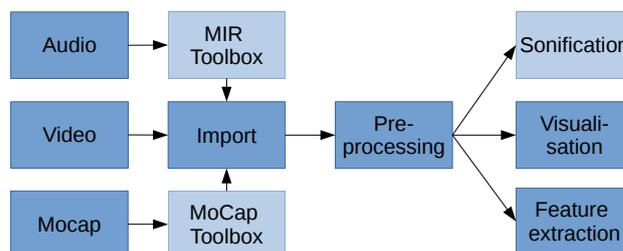


Figure 1. A sketch of MGT, the lightly coloured boxes indicate the integration with other toolboxes.

2.1 Importing

The MGT provides several functions for reading media/data files. Similar to the MoCap and MIR Toolboxes, it lets the user either start by importing a file before doing the analysis, or it can import data files directly with some of the analysis functions. This provides multiple entry points for the user, dependent on whether one just want to do a simple analysis of one file, or create a longer pipeline for the analysis of a whole folder of files. The import functions have also been built to switch between loading short files into memory, while loading longer files from disk. To work with combined audio, video and mocap data, it is necessary to use a dedicated MGT Matlab structure, which correctly links up the files.

⁵ <http://www.jamoma.org>



2.2 Pre-processing

There are numerous functions for pre-processing the data before moving on with the analysis. For video files this includes functions for cropping, rotating, flipping, and trimming, as well as basic colour adjustments. It also allows for pixel reduction and downsampling, to improve the speed of later analyses. All of these functions can be applied to both independent video files or complete data structures with audio and mocap data present. In the latter case, the toolbox will also trim the audio and mocap data to match the selected time segment of the video file, and to ensure that the files are properly time-aligned.

2.3 Visualizing

The MGT provides three techniques for estimating motion in the video files: (1) regular frame differencing, (2) optical flow [2], (3) eulerian video magnification [6]. The outputs of these three functions can be further visualised as motion history images, average images, videograms, motiongrams, or flowgrams [3].

2.4 Analysing

Most of the current functions in MGT are focused on handling the general workflow and creating different types of visualisation. There are also some basic tools for feature extraction, including quantity of motion (QoM), centroid of motion (CoM) and area of motion (AoM) from video files. The results of these functions can be plotted alongside the visualizations or used together with features from the other toolboxes for further statistical analysis and/or machine learning. The MGT provides also functions for calculating various statistical descriptors, including estimating motion periodicities. The latter is building on functions from the MoCap Toolbox, and is an example of how creating ‘glue’ between toolboxes can open for new and interesting, combined analyses. Then we can investigate relationships between for example QoM and various audio features, as illustrated in Figure 2.

3. CONCLUSION

The MGT is primarily aimed at providing music researchers and students with a complete and easy-to-use solution for performing analysis of music-related video recordings, as well as integrating with the MoCap Toolbox and MIR Toolbox. The MGT has already proven useful in both research and education settings at the University of Oslo.

The aim now is to officially launch the toolbox and thereby get a larger user group to get feedback from. There are still numerous functions that we want to implement, not least some more computer vision methods. We also want to continue exploring visualization techniques, and particularly the creation of combined displays of audio, video and mocap data. Sonification is another thread we have started, but which is still not fully functional. Finally, we are interested in exploring using different types of machine learning techniques on the extracted features.

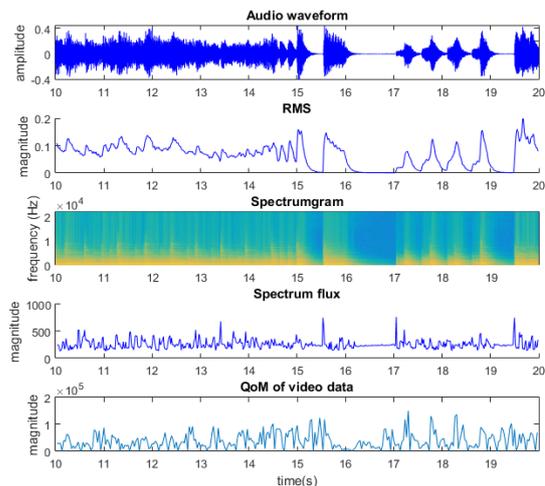


Figure 2. Plot of QoM from the video file against various audio features calculated by the MIR Toolbox.

4. ACKNOWLEDGEMENTS

Thanks to Bo Zhou for his work on the Matlab code during his master’s thesis [7]. The work is partially supported by the Research Council of Norway through its Centres of Excellence scheme, project numbers 262762 and 250698.

5. REFERENCES

- [1] Birgitta Burger and Petri Toiviainen. MoCap Toolbox - A Matlab toolbox for computational analysis of movement data. In *Proceedings of the Sound and Music Computing Conference*, pages 172–178, 2013.
- [2] Berthold K.P. Horn and Brian G. Schunck. Determining optical flow. *Artificial Intelligence*, 17(1-3):185–203, August 1981.
- [3] Alexander Refsum Jensenius. Some video abstraction techniques for displaying body movement in analysis and performance. *Leonardo*, 46(1):53–60, 2013.
- [4] Alexander Refsum Jensenius, Rolf Inge Gody, and Marcelo M. Wanderley. Developing tools for studying musical gestures within the Max/MSP/Jitter environment. In *Proceedings of the International Computer Music Conference*, pages 282–285, 2005. 00035.
- [5] Olivier Lartillot and Petri Toiviainen. A Matlab toolbox for musical feature extraction from audio. In *International Conference on Digital Audio Effects*, pages 237–244, 2007.
- [6] Hao-Yu Wu, Michael Rubinstein, Eugene Shih, John V. Guttag, Frdo Durand, and William T. Freeman. Eulerian video magnification for revealing subtle changes in the world. *ACM Trans. Graph.*, 31(4):65, 2012.
- [7] Bo Zhou. Video Analysis of Music Related Body Motion in Matlab. Master’s thesis, University of Oslo, 2016.