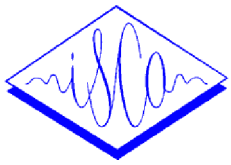
ISCA Archive
http://www.isca-speech.org/archive



Auditory-Visual Speech Processing
2007 (AVSP2007)
Hilvarenbeek, The Netherlands
August 31 - September 3, 2007

MATLAB Toolbox for Audiovisual Speech Processing

Adriano V Barbosa[1], *Hani C Yehia*[2], *Eric Vatikiotis-Bateson*[1]

[1]Department of Linguistics, University of British Columbia, Vancouver, Canada
[2]Department of Electronics, Federal University of Minas Gerais, Belo Horizonte, Brazil

adriano.vilela@gmail.com, hani@cefala.org, evb@interchange.ubc.ca

Abstract

Audiovisual speech processing has reached a stage of maturity where there are now numerous computational procedures needed to measure and assess multimodal signals. However, as is often the case, the results of these procedures are better known than the procedures themselves. This paper presents a MATLAB toolbox consisting of an extensive collection of tools we have developed over the past 10 years. These tools are not intended to be the final answer for multimodal speech analysis; rather they are presented as an easy-to-use and well-documented library whose scope is sufficiently broad to be useful to both experts and novices.

The toolbox includes procedures for measuring, organizing, modeling, and validating multiple streams of time-varying data, including acoustics, two- and three-dimensional motions of the speaker. In addition to physical and derived (from video) marker data, new functions have been implemented that incorporate optical flow techniques based on the OpenCV library. When complete the toolbox will allow us to track human body gestures during speech from video noninvasively and to quantify the correspondences between different performance modalities within and across speakers.

Index Terms: audiovisual speech, multimodal speech, face motion, optical flow, system identification, Matlab toolbox.

1. Introduction

In speech science, research on the area of audiovisual speech is something relatively recent. Although much progress has been made over the last decade or so, computational procedures commonly required in audiovisual speech processing are still not widely available to the speech community.

This paper presents a Matlab toolbox containing a set of functions for measuring, organizing, processing and assessing multiple streams of multimodal speech data. These include the speaker's acoustics and head/face motions (in 2D and 3D) as well as other signals derived from those. Two-dimensional motion is measured from video using either a marker tracking technique or optical flow analysis. More general functions for measuring human body gestures during speech are being developed. System identification techniques are used to assess how different performance modalities within and across speakers are related. An aim of this development is to quantitatively assess the extent to which the speech and body gestures of two talkers are spatiotemporally coordinated (entrained).

The functional organization of the toolbox is described in Section 2. The section starts by describing how 2D and 3D motion is measured and then how the measured auditory and visual data are organized into audiovisual objects. It proceeds with a description of the main audiovisual processing capabilities of the toolbox, and concludes by looking at the tools used to assess the relation between multimodal data streams. The summary is presented in Section 3.

2. Toolbox organization

The toolbox comprises a set of Matlab functions for

- measuring a speaker's motion;
- organizing audiovisual data;
- processing audiovisual data;
- assessing the relation between different channels of audiovisual data.

Figure 1 shows a schematics diagram detailing the steps involved in recording, conditioning and processing multimodal speech data with the toolbox. The first step shown in the figure, the data recording, is performed during speech production experiments and, naturally, does not involve the toolbox. The toolbox provides functions for all operations in the next two steps of the figure (signal conditioning and signal processing), but one. The exception is the head and face motion decomposition of 3D marker trajectories; this operation is performed by a piece of software accompanying the 3D sensing system (see Section 2.3 for details).

Besides the operations shown in Figure 1, the toolbox also provides various tools for the analysis of audiovisual data, including a set of functions for assessing the relations between different channels of the data. In the following, the individual functional parts of the toolbox are detailed.

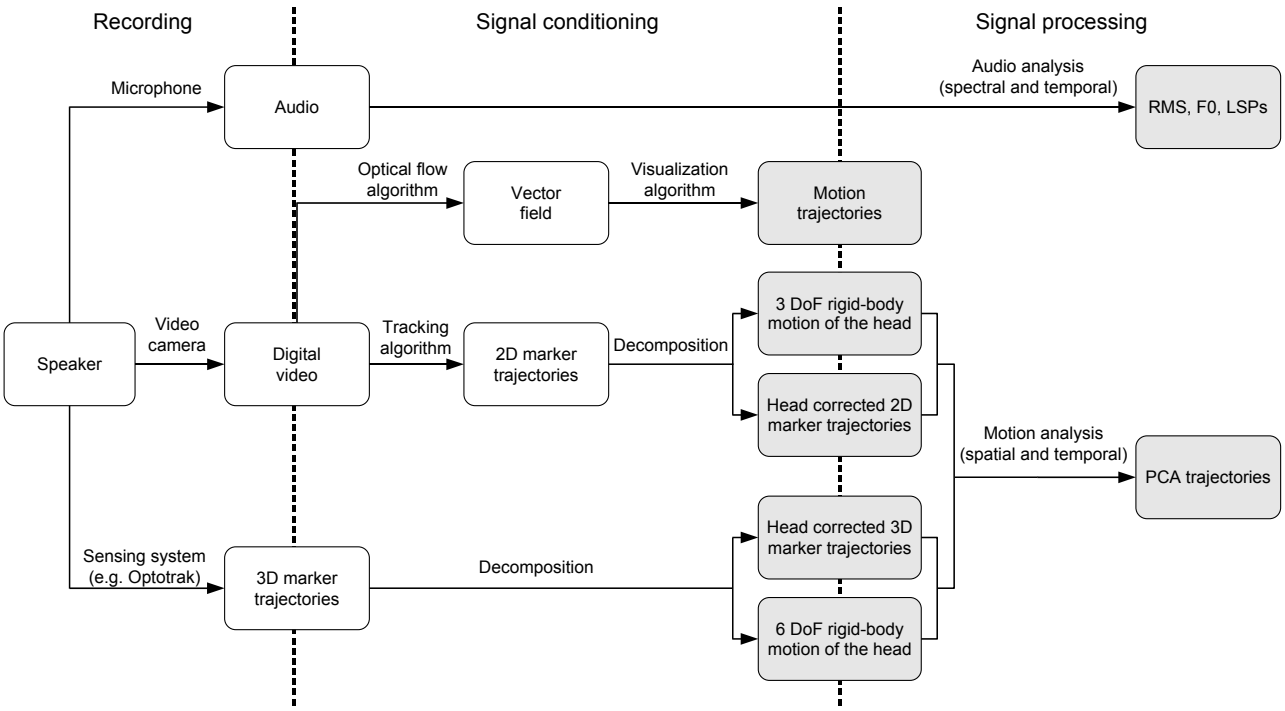


Figure 1: *Recording, conditioning and processing of audiovisual speech data. All operations in the second (signal conditioning) and third (signal processing) steps are performed by the toolbox functions, with the exception of the decomposition of 3D marker trajectories, which is done separately. The signals in the shaded boxes are stored in avo.signals (see Section 2.2) and therefore are available to all the analysis functions of the toolbox.*

2.1. Measuring motion

Currently, there are three approaches to motion measurement:

- 3D marker positions. Motion is measured by tracking the 3D positions of active markers (infra-red light emitting diodes) on the speaker's face. The tracking is performed in real time by an OPTOTRAK [1].

- 2D marker positions. Motion is measured by tracking the 2D positions of passive markers (colored stickers) on the speaker's face. The tracking, done in an offline stage, is performed by a toolbox function on the video sequence filmed during the experiment. This function finds the marker positions for every frame of the video sequence. A detailed description of the tracking algorithm can be found in [2].

- Optical flow. Motion is measured by computing the pixel displacement between every pair of consecutive frames of the video sequence filmed during the experiment. Again, this is done in an offline stage by one of the toolbox functions. This function has been written in C using the OpenCV library [3] and can use either Horn & Schunck [4] or Lucas & Kanade [5] method. Figure 2 shows optical flow results for one frame of a hand motion video sequence.

Regardless of the approach used for motion measurement, the resulting motion data has to be stored in a format that can be handled by the toolbox functions. The toolbox functions implementing the second and third approaches above do that automatically. In the first approach, however, the actual tracking is done by the OPTOTRAK and then the toolbox needs to convert the binary files with the motion data from OPTOTRAK format to the toolbox format.

2.2. Organizing audiovisual data

The experimental data need to be organized somehow before it can be manipulated by the functions of the toolbox. The toolbox uses Matlab data structures called *avos* (audio visual objects) as containers for all experimental data.

Experiments are organized into sentences. A sentence is simply a data segment taken for analysis and does not necessarily have to match an actual utterance. It could be an entire paragraph, for example, or even an entire conversation, in the case of spontaneous speech experiments.

The toolbox provides a means of organizing the experiment data into sentences. In order to do that, a matrix of sentence boundaries needs to be defined first. This ma-

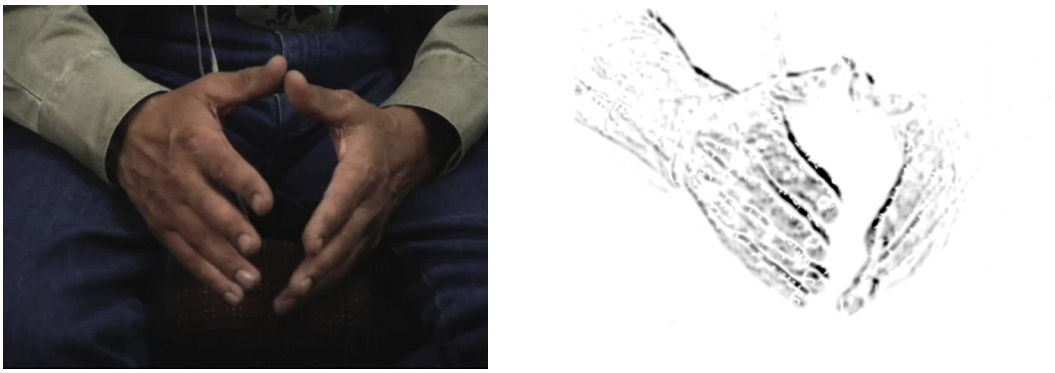


Figure 2: *Optical flow results (right) for one frame (left) of a hand motion video sequence.*

trix contains the initial and final points, in time, of all data segments of interest (the sentences). These segments are then retrieved from the files with the measured audiovisual data and stored in the experiment's *avo*. An *avo* can contain any number of sentences.

Figure 3 shows an *avo* structure. Currently, the structure contains the following fields (which are structures themselves):

- *signals* – contains signals with measured data, like audio waveforms and face marker trajectories, as well as other data signals derived from measured data (for example, the fundamental frequency F0 obtained from the audio waveform). All signals in the shaded boxes in Figure 1 are stored in this field;
- *rates* – contains the sampling rates for the signals in *avo.signals*;
- *config* – contains configuration data needed by the toolbox functions;
- *info* – contains meta-data information (currently, the experiment name).

```
>> avo
avo =
    signals: [98x1 struct]
      rates: [1x1 struct]
     config: [1x1 struct]
       info: [1x1 struct]
```

Figure 3: *An avo structure with 98 sentences as reported in Matlab prompt.*

Figure 4 shows the contents of *avo.signals*, which is a structure array where each element corresponds to one sentence of the experiment. Initially, when organizing the experiment data into sentences, only the fields corresponding to measured data are filled. In the example, these are *waveform* (the audio waveform) and *markers* (the marker trajectories). The remaining fields contain data signals obtained from these primary signals by other

```
>> avo.signals
ans =
98x1 struct array with fields:
    waveform
    markers
    markers_face
    markers_head
    pca
    rms
    lsp
    F0
```

Figure 4: *The field signals of the avo in Figure 3.*

functions of the toolbox, which will be discussed in Section 2.3. For example, the fields *rms*, *lsp* and *F0* contain, respectively, the RMS values, the LSP frequencies [6], and the fundamental frequency of the speech signal. The fields *markers_face* and *markers_head* contain the decomposed motion of face and head, respectively (see Section 2.3). Finally, the field *pca* contains the *principal components* of the face motion, obtained by applying Principal Component Analysis [7] to the decomposed face motion.

2.3. Processing audiovisual data

The toolbox offers a variety of functions for processing audiovisual data once they have been organized into *avo*s. Common tasks include:

- Resampling signals. This allows to express audiovisual data, whose sampling rates can vary over a wide range of values depending on the recording equipment, at suitable rates for further processing. For example, it is common for speech signals to be recorded at 48 kHz, but we usually want to downsample them to a more appropriate rate around 10 or 8 kHz. The toolbox can also handle non-integer data rates. This is useful when working with NTSC video sequences, which give rise to motion signals at 29.97 (frame rate) or 59.94 (field rate) Hz.

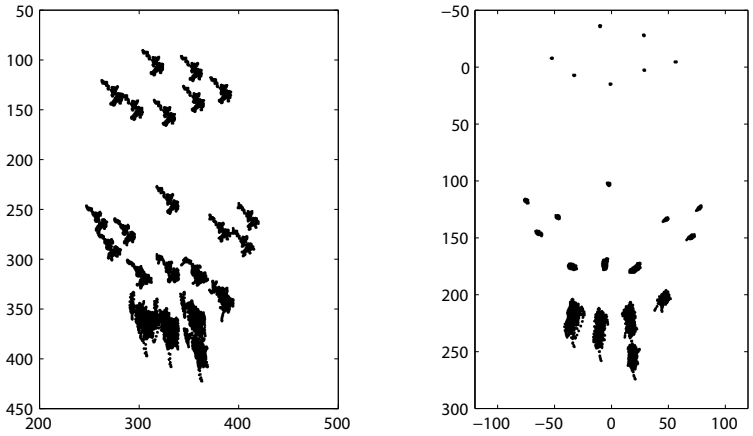


Figure 5: *Face motion before (left) and after (right) decomposition.*

- Analyzing/parameterizing audio signals. Different analyses for audio signals are available. For example, there are functions for computing the energy (RMS values) and the fundamental frequency F0 of the speech signal. It is also possible to perform LPC (Linear Predictive Coding) [8, 9] and LSP (Line Spectrum Pairs) [6] analyses. LSPs are particularly useful as a parameterization of the speech acoustics spectrum and have typically been used as such in our studies of the relation between speech acoustics and face motion [10, 11, 12, 13].

- Decomposing face/head motion. Subjects move their heads during speech and, therefore, head motion is intrinsically embodied in the measured trajectories of the face markers. In the case of 2D experiments, the toolbox provides a function for breaking the measured motion down into its head and face components. The rigid-body motion of the head is recovered in the form of two translation (along the x and y directions) and one rotation (about the z axis) signals. The recovered head position is used as a new coordinate system in which the marker positions are expressed. In the case of 3D experiments, the decomposition is performed by a piece of OPTOTRAK software. Figure 5 shows an example of face/head motion decomposition.

- Parameterizing motion. Motion is parameterized by means of Principal Component Analysis [7]. The main motivation for using PCA is to reduce the dimensionality of the face motion by exploiting the high redundancy among face marker trajectories. Typically, seven principal components are enough to account for about 99% of the motion variance.

The toolbox offers functions for various other tasks besides those listed above. For example, there are functions for enframing, windowing and pre-emphasizing speech signals; functions for filtering and visualizing audiovisual signals, and for merging sentences, among others.

2.4. Assessing the relation between channels of audiovisual data

A typical way of assessing the relation between two sets of signals is by looking at how good a mathematical model is at explaining (or predicting) one set from the other. The toolbox provides a number of functions for building, training and validating mathematical models that relate sets of audiovisual signals. These functions use system identification techniques and can handle linear and nonlinear, static and dynamical models. Model representations such as ARMAX [14] and NN-ARMAX [15] are available. Model validation can be done according to different criteria; for example, by looking at the correlation coefficient between measured and estimated signals.

Correlation coefficients have been used in our previous work [10, 12, 11] to assess the correspondence between audible and visible speech events over data segments of varying size (typically sentences). In this case, they serve as a measure of the global match between the signals, but without providing any temporal information about how the signals are related. As a result, the role of temporal organization in multi-modal speech cannot be easily assessed.

To address this deficit, we have developed an algorithm, based on recurrent correlation [16], that computes the instantaneous correlation coefficient between measurement domains. This allows rapid changes in correspondence for auditory-visual events to be evaluated as a time-varying function. The analysis of the time-varying coupling of multi-modal events has implications for speech organization and communicative coordination between speaker and listener. Figure 6 shows an example of instantaneous correlation coefficient between the speech energy (RMS amplitude) and the motion of the subject's hands measured by optical flow.

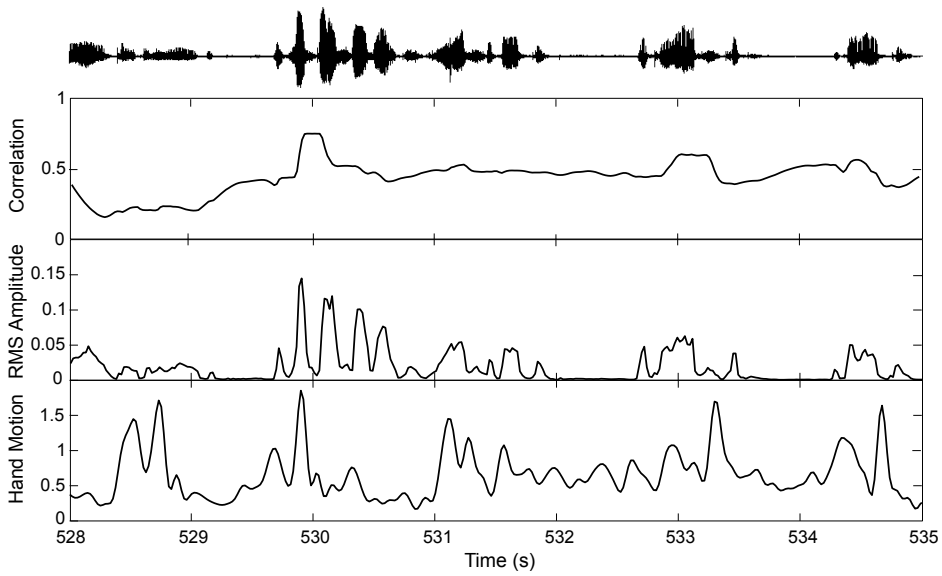


Figure 6: *Instantaneous correlation coefficient between hand motion (measured with optical flow) and the RMS amplitude of the auditory speech signal.*

3. Summary

This paper has presented a Matlab toolbox for audiovisual speech processing. The toolbox comprises a set of computational routines for measuring, organizing and processing streams of audiovisual data, as well as for assessing how the multimodal data streams may be related. The toolbox can handle both 2D and 3D motion, and provides functions for measuring 2D motion from video based on two different approaches: marker tracking and optical flow. All data are stored in audiovisual objects which are used throughout the toolbox. A wide range of analysis tools is available for processing both the auditory and the visual data. These tools can extract various types of information from the signals and parameterize them in suitable ways. It is possible to build, train and validate mathematical models relating data channels of interest in order to assess the relations between them. Tools for visualizing the data, as well as the results of the various analyses are also available.

4. Future work

Although we believe the toolbox presented here provides a collection of valuable tools for processing audiovisual speech, there are still many features that could be added or improved. For example, the algorithms currently used for recovering motion from optical flow vector fields are very simple and can be further developed. New mathematical mappings used to model the relation between channels of the audiovisual data are currently being developed. Some CPU intensive functions could be ported to a compiled language, like C, in order to reduce computation times. Furthermore, the development of a graphical user interface may result in a more intuitive experience for the toolbox user.

5. Acknowledgements

Support for this work was provided by NSERC and CFI (Canada) to Eric Vatikiotis-Bateson.

6. References

[1] "NDI: Optotrak Technical Specifications," http://www.ndigital.com/optotrak-techspecs.php, accessed in June, 2007.

[2] A. V. Barbosa and E. Vatikiotis-Bateson, "Video tracking of 2D face motion during speech," in *Proceedings of the IEEE International Symposium on Signal Processing and Information Technology – ISSPIT'2006*, Vancouver, Canada, August 2006, pp. 791–796.

[3] "Open Source Computer Vision Library," http://www.intel.com/technology/computing/opencv/, accessed in June, 2007.

[4] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[5] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI)*, 1981, pp. 674–679.

[6] N. Sugamura and F. Itakura, "Speech analysis and synthesis methods developed at ECL in NTT - from LPC to LSP," *Speech Communication*, vol. 5, pp. 199–215, 1986.

[7] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 1985.

[8] J. D. Markel and A. H. G. Jr., *Linear Prediction of Speech*. New York: Springer-Verlag, 1976.

[9] J. Makhoul, "Linear prediction: A tutorial review," *Proceedings of the IEEE*, vol. 63, no. 4, pp. 561–580, April 1975.

[10] H. C. Yehia, P. Rubin, and E. Vatikiotis-Bateson, "Quantitative association of vocal-tract and facial behavior," *Speech Communication*, vol. 26, no. 1–2, pp. 23–43, October 1998.

[11] H. C. Yehia, T. Kuratate, and E. Vatikiotis-Bateson, "Linking facial animation, head motion and speech acoustics," *Journal of Phonetics*, vol. 30, pp. 555–568, 2002.

[12] E. Vatikiotis-Bateson and H. C. Yehia, "Speaking mode variability in multimodal speech production," *IEEE Transactions on Neural Networks*, vol. 13, no. 4, pp. 894–899, July 2002.

[13] A. V. Barbosa, "A study on the relations between audible and visible speech," Ph.D. dissertation, Federal University of Minas Gerais, Belo Horizonte, Brazil, November 2004.

[14] L. Ljung, *System Identification Toolbox: For Use with Matlab*. The Mathworks, 2005.

[15] M. Nørgaard, "Neural network based system identification toolbox, version 2," Department of Automation, Technical University of Denmark, Tech. Rep. 00-E-891, 2000.

[16] R. M. Aarts, R. Irwan, and A. J. E. M. Janssen, "Efficient tracking of the cross-correlation coefficient," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 6, pp. 391 – 402, September 2002.