INTRODUCTION TO MUSICAL GESTURES TOOLBOX

Bo Zhou

Department of Informatics
University of Oslo

6.2016

1. What is MG Toolbox?

The MG Toolbox[1] is a Matlab toolbox and developed for video analysis of music-related body motion, which integrated some existing toolboxes: Mir-toolbox for music information retrievel, Mocap toolbox for visualization and analysis of motion capture data. In addition to these two toolboxes, the MG Toolbox requires matlabPyrtools and Matlab pre-built toolboxes: Image Processing Toolbox and Signal Processing Toolbox. As such, it could work on the multi-data (video, audio, mocap data). Three techniques have been implemented in the current vision of the toolbox, including motion-gram, optical flow, eulerian video magnification. A simple overview of the toolbox is shown in figure 1.

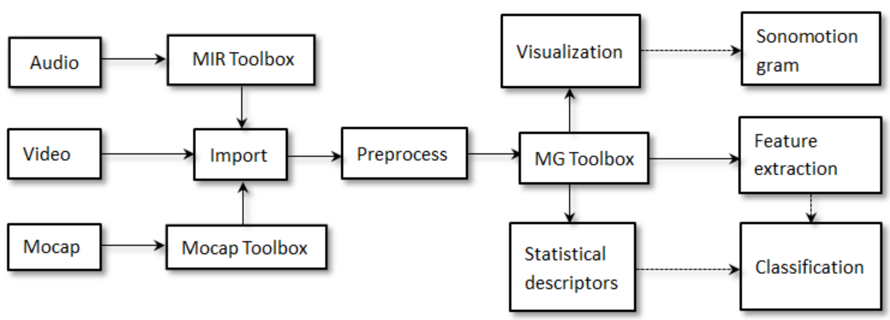

Figure 1: Framework of MG toolbox, dotted line means possible extension in the future work.

2. Four basic parts

As the figure 1 showed, the MG Toolbox consists of four basic parts. They are:

Importing. The MG toolbox provides several functions for reading different raw data (video, audio/sound, mocap data) into Matlab workspace. These

[1]See https://github.com/benlyyan/MGT

functions can operate on any one of them or any combination of them, and import specified *temporal segment* of the raw data. These functions are extremely useful when we deal with large video files. For one thing, importing the whole data would need lots of memory, and it would be time-consuming. For another thing, the signal of interest is often hidden in the temporal domain. Extracting the temporal segment of the input data is important during importing.
Importing functions: mgread, mgvideoreader, mgreadsegment

Preprocessing. After importing, preprocessing of the data is important when the raw data is of poor quality. For instance, the video has low contrast or improper vision angle. The MG toolbox contains various functions for video preprocessing. Such preprocessing operations include adjusting the contrast of a video, rotating the view angle of a video, cropping the region of interest of a video, down sampling a video with high resolution, magnifying micro motion of a video, etc.
Preprocessing functions: mgmap, mgvideocrop, mgvideoadjust, mgvideosample, mgvideocat, mgvideomagnify

Analysis. The MG toolbox provides several techniques for video analysis, such as motiongram, optical flow, eulerian video magnification. The features computed by these techniques can reveal the relationship of body motion and music. Motiongram and optical flow make it possible to see spatial temporal information of the movements of an object. From these features, we can find the correspondence with music. Eulerian video magnification reveals the subtle movements in a video which are difficult or impossible to see with naked human eyes.
Analysis functions: mgmotion, mgstatistics, mgpca, mgautocor, mgsimilarity

Visualization. The MG toolbox has two visualization tools. One is to show the motiongram, optical flow field and bounding box over time. It mainly shows two types of motiongrams: the vertical motiongram and horizontal motiongram. Another is to show waveform and spectrum of music or sound, together with quantity of motion.
Visualization functions: mgvideoplot, mgvideoplot1, mgwaveplot

Using MG Toolbox

In this section, we show an example of how to use MG Toolbox. The dataset includes pianist.mp4, pianist.wav, and pianist.tsv.

mg = mgvideoreader('pianist.mp4','Extract',10,30);

where "Extract" indicates the temporal segement extraction. If the video has very high resolution, downsampling the video in order to reduce the size of the video could be useful.

mgsam = mgvideosample(mg,'Sampling',[2,2]);

mgcro = mgvideocrop(mgsam);

which will plot the first frame of the video and allow user to select region of interest. If we want to analyze pianist's upper body, we can just select the corresponding regions as shown in figure 3.

Figure 2: The original video

Figure 3: The cropped video only shows the upper body of the pianist

Next, we can import the audio, mocap data into our data structure like this:

mgcro = mgmap(mgcro,'Both','pianist.wav','pianist.tsv');

mgmo = mgmotion(mgcro,'Diff');
or
mgop = mgmotion(mgcro,'OpticalFlow')

where 'Diff' indicates using difference method for motion estimation, 'OpticalFlow' indicates using optical flow method for motion estimation. Figure 4 shows the qom and com of the video and figure 6 shows the motiongram of the video.

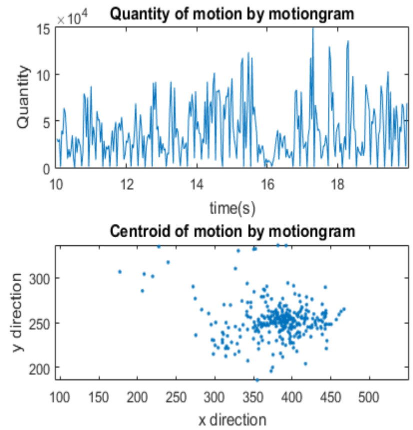

Figure 4: The QoM and CoM of the video computed by motiongram

The futher estimation can be made to find the periodicity of motion based on the QoM, i.e.

[per,ac,eac,lag] = mgautocor(mgmo,3);

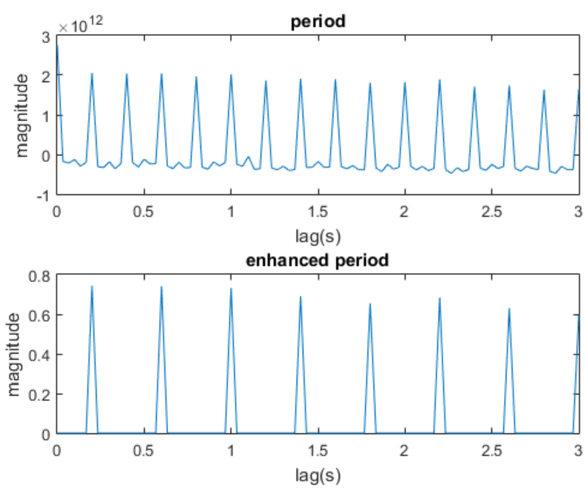

Figure 5: The periodicity estimation of the movements based on QoM of the video data. The first non-zero lag is around 0.2 sec. It is computed by the mgautocor in the MG toolbox.

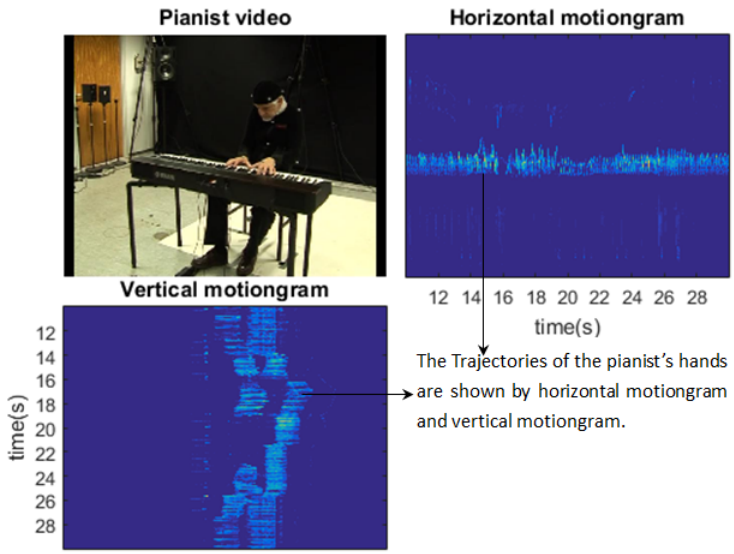

Figure 6: Two types of motiongrams are computed with the function *mgmotion* with "Diff" option in the MG toolbox. From the vertical motiongram, a break can be noted from 16 seconds to 17 seconds, and from 19 seconds to 23 seconds, there is only the pianist's left hand playing the piano. From the horizontal motiongram, a series of repeatable actions could be observed during performance.

We could show the bounding box or optical flow field in the video using original data structure mgcro before applying the function mgmotion, i.e.

mgvideoplot1(mgcro,'Boundingbox');
mgvideoplot1(mgcro,'Opticalflow');

The statistics descriptors can be computed based on the motion image using mgstatistics, i.e.

feat1 = mgstatistics(mgmo,'Video','FirstOrder');
feat2 = mgstatistics(mgmo,'Video','SecondOrder');

The video can be magnified by the function mgvideomagfnify,i.e.

mg = mgvideomagnify(mg,'IIR',0.2,0.01,5,8,0.1);

Figure 7: mgvideoplot1 with “Boundingbox” option.The red bounding box in the video frame shows the largest motion area.

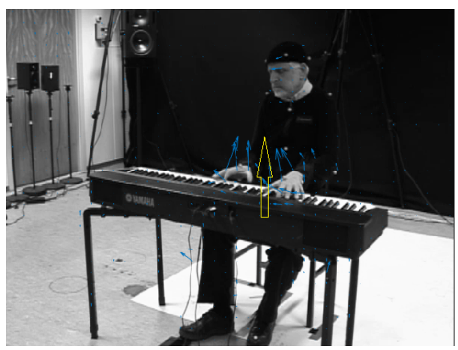

Figure 8: mgvideoplot1 with “Opticalflow” option. The background image with optical flow field.Large arrow indicates the moving orientation of the pianist’s hand.

Figure 9: The original video

Figure 10: Magnified video,the local objects,hands and head are magnified.

The similarities could be calculated by the function mgsimilarity, i.e.

$$simix = mgsimilarity(mgmo.video.gram.gramx');$$
$$simiy = mgsimilarity(mgmo.video.gram.gramy);$$

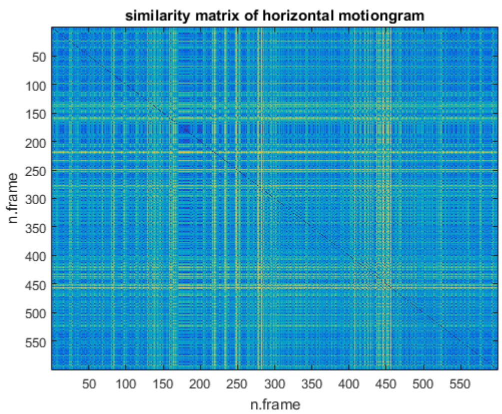

Figure 11: similarity matrix of horizontal motiongram

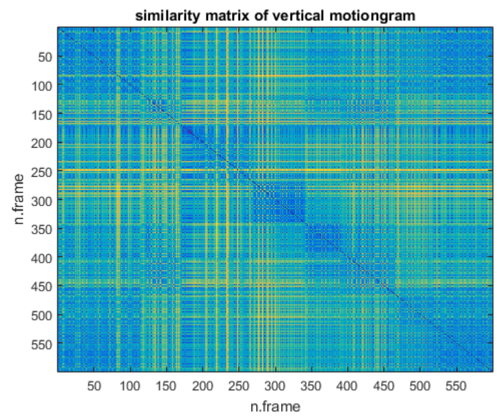

Figure 12: similarity matrix of vertical motiongram

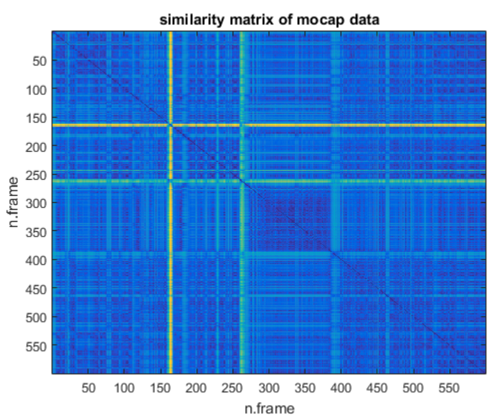

Figure 13: similarity matrix of Qom of mocap data

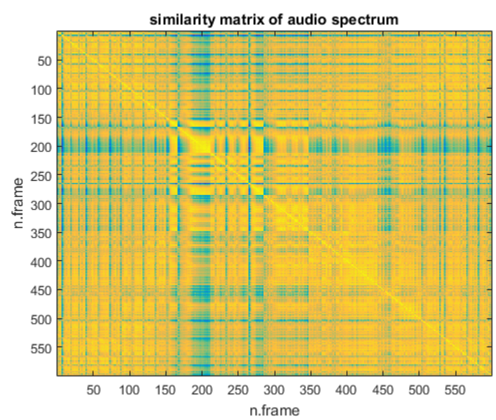

Figure 14: similarity matrix of spectrum of audio