# Homework 4: Video Processing and Motion Magnification

Total points: 70pts

## Instructions

Due date: April 7th, 2021. No late work acceptable.

We provide a python notebook with the code to be completed. You can run it locally

or in Google CoLab (upload it to Google Drive and select ‘open in colab’ ) to avoid setting up your

own environment. Once you finish, run the cells and download the notebook to be submitted. Please do include links to the media file if they are uploaded online or include in your folder.

As what we have done in Homework 3, please also complete the .html in the html folder that summarizes your results and comment on your success and failures. You can replace the images (videos) and texts with the default.

**Please submit a link of CoLab code and also a .zip file named <your first\_lastname>.zip containing**

**1) report named report.pdf including your answers to all required questions with**

**images and/or plots showing your results, and 2) the python notebook provided, with the**

**cells run and the relevant source code. 3) All of your source code as .py if you prefer me to run it locally.**

The best resources of this homework are Lecture 11&12 and cvbookTemporal.pdf

You can learn how to play video from file using Python from:

<https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_gui/py_video_display/py_video_display.html>

The homework instructions and starter code can be found here:

<https://colab.research.google.com/drive/1GF3vRh1DmPy3_DU8Qa4GZM6X2EN0NU9-#scrollTo=IazqT6pfFSMp>

## Question 1 (20pts) Follow the instructions in Google CoLab and finish the exercises described. Some tutorials on video processing.

<https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_gui/py_video_display/py_video_display.html>

## Question 2 (50pts) Problem Statement

In this problem we will investigate motion magnification in videos. Recall that **position shifts**

**in image space** correspond to **phase shifts in the frequency domain of the Fourier transform.**

This means that for two images, we can compare the Fourier transform of the two images

to find the phase shift between the images. Amplifying the phase shift by a fixed factor in

the Fourier transform frequency domain will amplify the position shift by the same factor in

the image domain after we perform the inverse Fourier transform. We will use this idea to

exaggerate the motions in videos.

(a) For a purely horizontal offset of an impulse signal, magnifying the phase shift will result in

a magnified horizontal offset after the inverse transform. Please fill in lines 6 and 9 in function

magnifyChange. You should find the phase shift between the two input images and magnify it by the

specified *magnificationFactor*. When complete, the function magnifyChange should return

an image showing what image 2 would look like with the magnified offset. Please run **Problem (a)** and submit the generated plot.

(b) If there is motion in more than one direction between two images, we will see that naively

magnifying the phase shift of the whole images will not work. In **Problem (b)**, we have set

up a vertical offset of an impulse signal as well as the horizontal one from part a. Please

run **Problem (b)** and submit the generated plot, then explain why the two offsets were not

properly magnified.

(c) One strategy we can use if there are multiple motions between two images is to do a

**localized Fourier transform** by independently magnifying the offsets on small windows of the

images and aggregating the results across the windows. When we restrict our window of

consideration, it is more likely for everything in the window to be moving the same way. We

will use Gaussian filters to mask small windows of the image and perform magnification on

each window independently. In **Problem c**, please fill in the Gaussian filter in line 27 and

the appropriately windowed input images in line 28. Since we are working with images, we

will use the discrete Gaussian filter rather than the continuous one. Run **Problem c** to

confirm that the two motions were properly magnified and submit the generated plot.

(d) We are now ready to apply motion magnification to videos. We will use the same approach

as in part c of magnifying Gaussian windowed regions of the video frames. Rather than

directly finding the phase shifts between consecutive video frames, we will keep a moving

average of the Fourier-transformed phases and compare each new frame’s DFT phase with the

current moving average of phase. The moving average is an IIR low-pass filter, averaging 0.5

times the previous average with 0.5 times the current phase. For simplicity, each of the RGB

channels are processed independently and identically. **In Problem d**, you will need to fill

in the Gaussian filter in line 28, the DFT phase of the magnified window in line 44, and the

DFT of the magnified window in line 47. Please run **Problem d** and submit the generated

video. Note that the code may take some time to run - you can temporarily modify sigma to

decrease the number of windowed regions to process.

f) Motion magnifies videos other than bill.avi (e.g.baby breathing, or any facial expression videos). Please comment on how your code works. You can use the guitar video I attached as well.

More resources:

<https://www.youtube.com/watch?v=MYp298fhlzk>