ENSC 180: Introduction to Engineering Analysis

Craig Scratchley, Spring 2016

Assignment #3 – Mandelbrot Panning and Zooming

Read Chapter 13 of the Moler book. You can also read Chapter 6 to get an introduction to a couple of the MATLAB functions that we are using such as tic and toc. The file assignment3.m provides you with template code that makes a medium-resolution video that pans across a Mandelbrot image. File assignment3.m as it is given to you uses function mandelbrot_step to update the z and c matrices. In order to get access to function mandelbrot_step, I suggest copying files assignment3.m and WarningOff.m to the 'exm' folder you should have after expanding the 'exmgui.zip' file that can be downloaded from Canvas. As indicated in the output of the program in the MATLAB Command Window, how long does it take to calculate all the frames when using function mandelbrot_step? _____ Comment out that line and allow the MATLAB code below it to be executed to update the z and c matrices. Now how long does it take to calculate all the frames? _____ Now go back to using function mandelbrot_step. As explained in Chapter 13, the mandelbrot_step function allows for compiled C language to be executed instead of MATLAB code and this can allow for the implementation to be optimized. Look at the mandelbrot_step.c file and compare with the mandelbrot_step.m file. What do you think is the primary optimization that file mandelbrot_step.c leverages?

Through use of the variable DO_IN_PARALLEL, we can use 'parfor' instead of 'for' to iterate through the frames. The code as provided uses 'parfor'. Record again how long it takes the code as provided to calculate all the frames. _____ Now set DO_IN_PARALLEL to false to switch to using 'for'. Now how long does it take to calculate all the frames? _____ At this point, I suggest switching back to having DO_IN_PARALLEL set to true.

The goal of this assignment is to create a beautiful video that does a combination of zooming and panning into (and/or out of) the Mandelbrot image. Instead of panning from 0.5+0i to -1.5+0i, start by modifying the program as provided to make a nice zoom into the image from point -1.5+0i. You can make sensible improvements to the template code to facilitate this and the further requirements below. How far into the image can you zoom before the resulting image frame gets grainy and pixelated? Why does this happen?

How would you repeatedly call the iterate nested function so that the centre point of frames can change (i.e. pan) as well as allowing for the zoom level to change, and making the resulting video smooth? Consider that you might not be panning and zooming at the same rates during different portions of your

video. If you get a chance, try the Frax software available for iOS devices to get an idea of what is possible. Perhaps using the programs ultrafractal available from

http://www.ultrafractal.com/

and XaoS available from

https://sourceforge.net/projects/xaos/

figure out a path that you would like to follow for your final video. Consider starting with the entire Mandelbrot set visible, and then in varying degrees zoom and pan along a path of your choosing. Alternatively, you can also start your video at any point and zoom level of your choosing. Feel free to zoom out during portions of your video in addition to zooming in. For example, while zooming in start panning over, and then reduce your zooming and in fact start zooming out. As you start panning, consider accelerating the pan and then decelerating it.

While we hope your video will have artistic merit, you can also get credit for using interesting math while creating your video. Use your imagination. Make sure you highlight the artistic and mathematical merit of your video/programming before you submit your work.

You can use whatever aspect ratio you want. While you are doing exploratory work, feel free to reduce resolution and frame rate, etc. Once you have a better idea of your starting point and your path of panning and zooming, etc., increase the resolution, and perhaps fame rate and depth parameters in order to bring your video to a quality that you are happy with. Feel free to use university computers, at least while finalizing your video, if your computer is slower. The depth parameter does not need to be constant during calculations for each frame that you calculate.

You will need to upload your file assignment3.m as well as your video file. Depending on any bonus work that you do, you might need to upload mandelbrot_step.? as well.

Bonus marks are available for doing optimizations to the code. You can get bonus marks for optimizing the code with DO_IN_PARALLEL set to false if those optimizations would be difficult or impossible with DO_IN_PARALLEL set to true. Talk to me if you would like some ideas for optimizations. You will need to report execution times with and without the optimizations for examples you provide, and a description of the circumstances under which the optimizations should be useful.

As an alternative bonus question, try to explain exactly why mandelbrot_step.c produces slightly different images than mandelbrot_step.m, at least on 64-bit Microsoft Windows computers where I have tested this. Prove your explanation.