Prof. Raghuveer Parthasarathy
University of Oregon; Fall 2024

Physics 410/510 Image Analysis: Homework 0

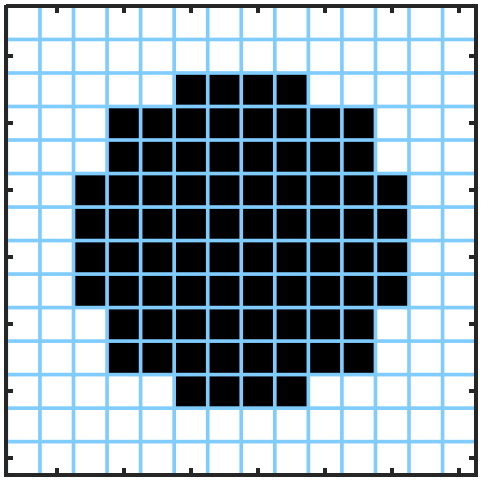
This isn't an assignment, but an exercise to give a flavor of the sort of programming we'll be doing in this course. You should be able to do the following – it will take some time and some thought, but the necessary programming techniques should be familiar *before* starting the course. Looking things up is fine. I strongly prefer that you use Python or MATLAB (see the syllabus).

You should become familiar, if you aren't already, with how to make Python or MATLAB code run quickly and efficiently by vectorizing and avoiding for-loops, pre-allocating arrays, etc.[1] Images can be large, so efficiency is often important.

1 Circles on grids. Consider a disk of radius 1 on a grid of square tiles that are black if the distance from the center of the square to the origin is ≤ 1 and white if > 1. The width of each square is $1/N$ and the origin is at the corner of four squares. See the illustration below, for $N = 5$.

(a) Write a function that calculates the total black area as a function of N. Note that you don't have to create an image of a circle-on-a-grid.

(b) Make a graph of the black area as a function of N for N from 2 to 2,000, with N on a logarithmic axis. Does the area have the expected behavior as $N \rightarrow \infty$?



Suggestions: `logspace` and `meshgrid` are useful function!

[1] Python: See e.g. https://realpython.com/numpy-array-programming/, https://towardsdatascience.com/tips-and-tricks-for-high-performance-computing-adeb78e96a4 . MATLAB: See e.g. http://www.csc.kth.se/utbildning/kth/kurser/DN2255/ndiff13/matopt.pdf , http://web.cecs.pdx.edu/~gerry/MATLAB/programming/performance.html.

Solution:

There are many ways to perform this calculation. My approach was to note that the x- and y-positions of the grid centers are each $\pm(0.5, 1.5, 2.5, ...)/N$. We can therefore make an array of distances-to-the-center $r = (x^2 + y^2)^{1/2}$ for each grid square, and count the number of squares with r are less than 1. (Actually, I counted the number for which $r^2 < 1$ to avoid calculating an unnecessary square root.) Each such square contributes $1/N^2$ to the total area.

See the code:

Python: **`circles_on_grids.py`**

https://pages.uoregon.edu/raghu/TeachingFiles/circles_on_grids.py

MATLAB: **`circles_on_grids.m`**

https://pages.uoregon.edu/raghu/TeachingFiles/circles_on_grids.m

(This also plots deviation from π and illustrates the grid.)

Here's what area vs. N looks like, with a dotted line at π, to which the area asymptotes as expected.