Math Teachers' Circle Lesson: Fun with Filtering

Justin Crum, Andrew Gillette, Josh Levine
University of Arizona

June 22, 2018

Abstract. Photo applications on smart phones have all sorts of filters, but how do they work? In this session, we will learn about the mathematics behind filtering and compute some filters by hand through some simple examples. We will also learn about the "Game of Life" in which pixels appear and disappear according to simple rules; the game can easily be adapted to classroom activities. We'll conclude with a discussion of the relationship between filtering procedures and the "Game of Life."

Materials.
- Board space for class and scratch paper for participants.
- Computer connected to internet for Game of Life demo at end of lesson.

Introduction - Overview and Discussion

- Take a photo on your phone and 'edit' it - what's happening?
- Think of this as input image →(apply some rules)→ output image
- Think of an image as a grid of numbers - the number indicates the color of that pixel
- Example: 3x3 image with 1's and 0's; 1=black, 0=white; rule = swap 1 to 0 and vice versa; output will be color reversal black/white of input
- Explain how a single box - called a pixel - has 8 neighbors (including diagonals)

Activity 1 - Image morphology

We'll start our study of image manipulation on images that are only black and white (1's and 0's). Consider two possible rules:

Rule D: At each pixel of the image that has value 1, make it and all 8 of its neighbors equal to 1.

Rule E: At each pixel of the image that has value 1, do all 8 neighbors have value 1? If yes, leave it as 1. If not, change it to 0.

Discuss:

- What do you think will happen when we apply rule D to a black/white image?

- What about rule E?

Some notes about potential points of confusion:

- Draw a grid for the output image separate from the input image.
- Note that Rule D affects a pixel and its neighbors; Rule E only affects one pixel at a time
- Leave space for multiple iterations to be drawn
- What should happen on the boundary of the image? Treat neighbors outside the image as 0's.

Matrix A (11x11, for Rule D):

$$A = \begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Matrix B (11x11, for Rule E):

$$B = \begin{matrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{matrix}$$

(10 more rows of 1's)

For participants:

- Apply Rule D to image A. What happens? Apply Rule D again. Repeat until nothing will change further.
- What effect does Rule D have on the 'shape' of the image?
- Apply Rule E to image B. What happens? Apply Rule E again. Repeat until nothing will change further.
- What effect does Rule E have on the 'shape' of the image?
- — leave time for participants to work here; possible skip next part —
- Apply Rule E then Rule D to image B. What happens? (This is called 'opening' because it opens a gap at the top of the image)
- Apply Rule D then Rule E to image A (This is called 'closing' because it closes the gap at the top of the image)

These examples are taken from Wikipedia pages on Dilation and Erosion (Morphology)

Activity 2 - Image filtering

The last activity modified the image by expanding or shrinking an object in the image. Now we look a different kind of modification called filtering. For this, we change the value in a cell by doing some addition or subtraction of the neighboring values.

Example:

$$F = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 0 & 1 \\ \hline 0 & 0 & 0 \\ \hline \end{array} \qquad A = \begin{array}{|c|c|c|c|c|c|}\hline 0&0&0&1&1&1\\ \hline 0&0&0&1&1&1\\ \hline 0&0&0&1&1&1\\ \hline 0&0&0&1&1&1\\ \hline 0&0&0&1&1&1\\ \hline 0&0&0&1&1&1\\ \hline \end{array}$$

Explain what it means to "apply" a filter F to an "image" A: imagine that you place the center of the filter (F) over a particular cell of A. Numbers that sit on top of each other are multiplied together, giving nine new numbers. These nine numbers are added together and stored in the center of where the filter was placed. If the filter goes over the edge of A (as happens when you place the filter on a boundary cell of A) then any missing values should be treated as 0. Another way to say this: imagine there is a circle of 0's all around A.

For instance, the above example shifts every pixel the left and leaves a column of zeros on the right.

For participants: consider image B below.

$$B = \begin{array}{|c|c|c|c|c|c|}\hline 0&0&0&1&1&1\\ \hline 0&0&0&1&1&1\\ \hline 0&0&0&1&1&1\\ \hline 1&1&1&0&0&0\\ \hline 1&1&1&0&0&0\\ \hline 1&1&1&0&0&0\\ \hline \end{array}$$

What happens when each of the following filters is applied to B?

$$\begin{array}{|c|c|c|}\hline 0&0&0\\ \hline 1&0&0\\ \hline 0&0&0\\ \hline \end{array} \qquad \begin{array}{|c|c|c|}\hline 0&1&0\\ \hline 0&0&0\\ \hline 0&0&0\\ \hline \end{array} \qquad \begin{array}{|c|c|c|}\hline 0&0&0\\ \hline 0&0&0\\ \hline 0&1&0\\ \hline \end{array}$$

Is the following "equation" true?

$$\begin{array}{|c|c|c|}\hline 0&1&0\\ \hline 0&0&0\\ \hline 0&0&0\\ \hline \end{array} + \begin{array}{|c|c|c|}\hline 0&0&0\\ \hline 1&0&0\\ \hline 0&0&0\\ \hline \end{array} = \begin{array}{|c|c|c|}\hline 0&1&0\\ \hline 1&0&0\\ \hline 0&0&0\\ \hline \end{array} ?$$

Answer: No, because filters don't add that way; try an example on B.

What do these filters do?

$$\begin{array}{|c|c|c|}\hline 0&0&0\\ \hline 1&-2&1\\ \hline 0&0&0\\ \hline \end{array} \qquad \begin{array}{|c|c|c|}\hline 0&1&0\\ \hline 0&-2&0\\ \hline 0&1&0\\ \hline \end{array}$$

Try them on A and B.

Discuss how these are effectively horizontal and vertical "edge detectors" since they pick out (with numbers) where the edges are between "1" regions and "0" regions of the image.

Also discuss the application of this filter:

0	1	0
1	-4	1
0	1	0

which detects edges in both directions. This filter is called the "discrete Laplacian" operator.

Additional background and related topics can be found on the Wikipedia page for "Kernel (image processing)."

Activity 3 - The "Game of Life"

Played on an infinite grid of squares (cells).
Each cell is either "alive" or "dead".
At each step in time, cells transition as follows:

For each cell that is alive:

- If it has 0 or 1 neighbors it dies, as if by solitude.
- If it has 4 or more neighbors it dies, as if by overpopulation
- If it has exactly 2 or 3 neighbors, it survives.

For each cell that is dead:

- If it has exactly 3 neighbors, it becomes alive, as if by reproduction.

Examples:

	1	1	1	

$\Longleftrightarrow$

		1		
		1		
		1		

This example is called "Blinker" because it goes back and forth between two states.

Participants: investigate behavior of these patterns:

		1	1		
	1			1	
		1	1		

Beehive

		1	1	1	
	1	1	1		

Toad

		1			
			1		
	1	1	1		

Glider

(Beehive is stationary; Toad has period 2; Glider has period 4 but moves southeast.

Participants attempt these questions:

- Can you find another pattern that remains stationary?
- Can you find a pattern that dies out?
- How close can two blinkers be to each other without interfering?
- Can you find a pattern that seems to grow for a long time?

Discussion:

- English mathematician John Conway invented this game in 1970 and it remains a nice example of complicated behavior spawned by simple rules.
- Show website for Conway's game of life: https://bitstorm.org/gameoflife/

Contest: each group of participants picks one pattern (max 5x5 grid). Then they input this pattern into the game of life simulator and see how long it generates life before entering into a stable pattern. Longest pattern wins!