

Explanation of Custom Code

In order to test various structuring elements quickly, i created a function called `testStrels()` to create an array of structuring elements. The array contains `n`-elements where `n` is equal to `iterations` - a function parameter. Each element is size `increment` larger than the previous element in the array.

the function is given below:

```
% Function to create structuring lements of increasing sizes
function [test_se_array] = testStrels(strel_type, increment,
iterations)
    test_se_array = cell(1, iterations); % preallocating array for
holding se's

    % test structuring elements with increasing size
    for i = 1:iterations
        se_test = strel(strel_type, i*increment);
        test_se_array{i} = se_test; % save to array
    end
end
```

Image Pre-processing

Before performing the image manipulations, i first created a 1-D version of the iamge by applying `rgb2gray()` and finally `imbinarize()` over the image.

```
% loading in the image
im = imread('~/Downloads/monochrome_bitmap.jpg');
im = rgb2gray(im);
im = imbinarize(im);
```

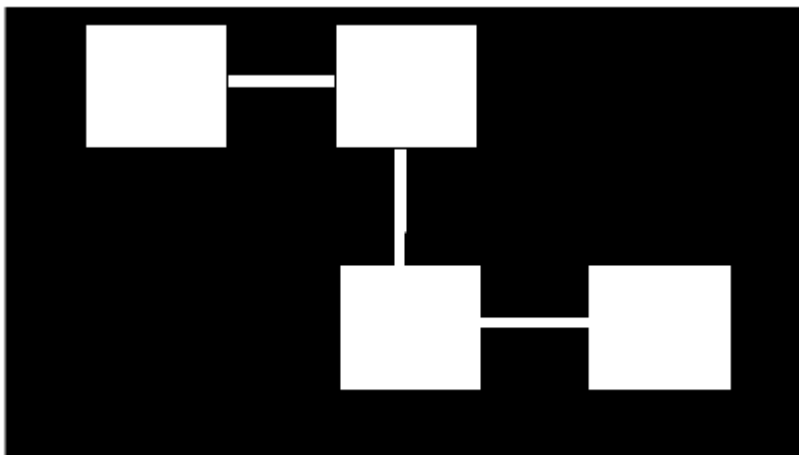
Imopen()

I first created the original image in google slides and opened it in MATLAB:

```
% loading in the image
im = imread('~\Downloads/monochrome_bitmap.jpg');
im = rgb2gray(im);
im = imbinarize(im);

% displaying original
imshow(im)
```

Original Image



Size

```
ans = 1×2
540 960
```

Generating & Applying Structuring Elements

i use my `testStrels()` function to create structuring elements to be tested. In the code below, i define my structuring elements to be 7 disks where each disk is 10 units larger than the last.

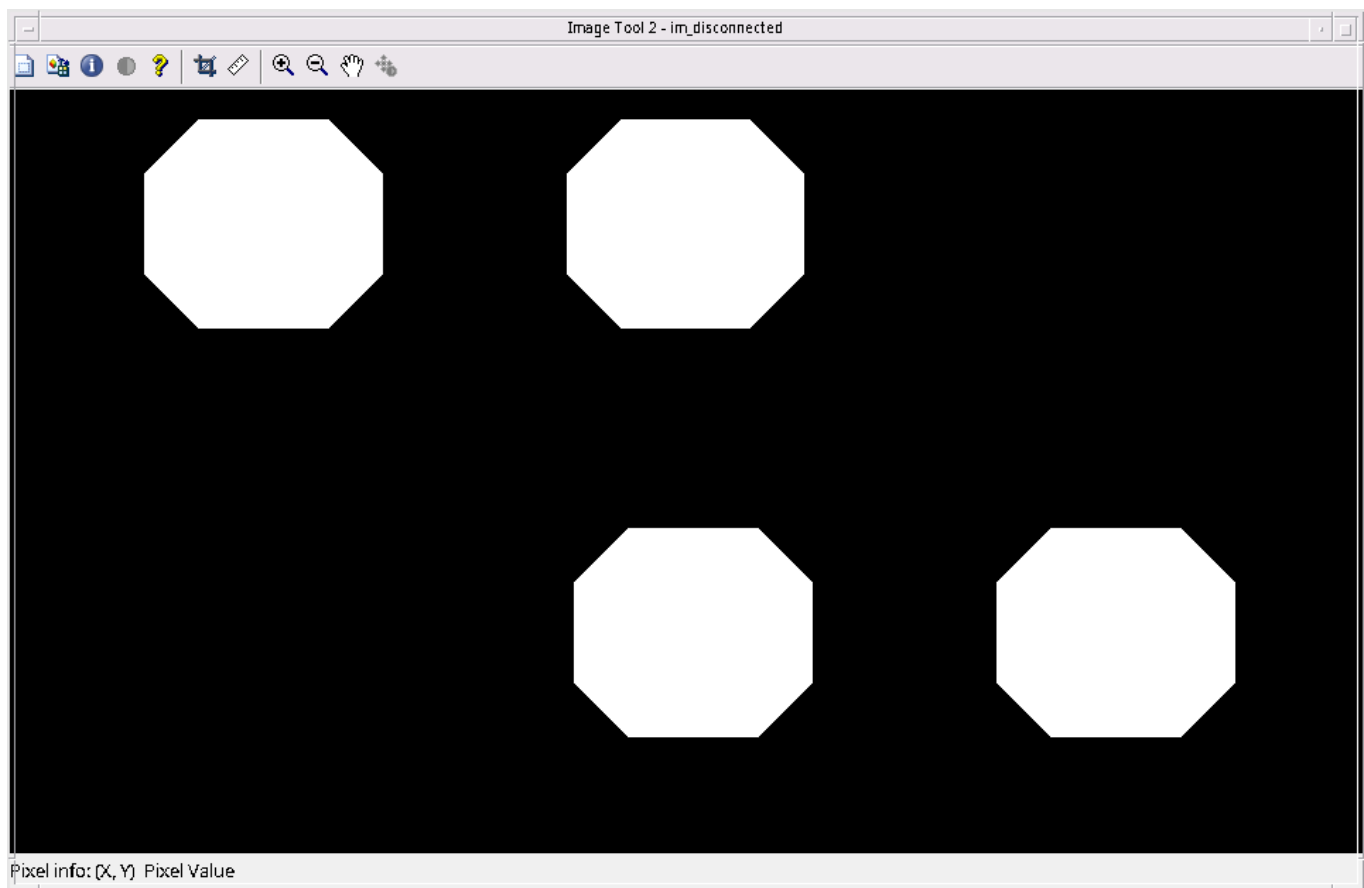
```
% testing strels for open
strel_type = 'disk';
increment = 10;
iterations = 7;

% creating disk based structuring elements
strels = testStrels(strel_type, increment, iterations);

for i=1:iterations
    im_disconnected = imopen(im, strels{i});
    disp('Strel Size: ')
    disp(num2str(i*increment))
end

imtool(im_disconnected)
```

Resulting image



After testing, a structuring element disk with a size of 70 produced this image.

Imclose()

Initial Image

this is the initial image that will have its elements joined into a contiguous object using `imclose()`



Size

ans = 1×2

540 960

Code

```
% loading in the image
im = imread('~/Downloads/monochrome_bitmap_2.jpg');
im = rgb2gray(im);
im = imbinarize(im);

% displaying original
imshow(im)

% rejoining the disconnected image
strel_type = 'square';
increment = 30;
iterations = 5;

strels = testStrels(strel_type, increment, iterations);

for i=1:iterations
    im_disconnected = imclose(im, strels{i});
    disp('Strel Size: ')
    disp(num2str(i*increment))

end

imshow(im_disconnected)
```

```
% Function to create structuring elements of increasing sizes
function [test_se_array] = testStrels(strel_type, increment,
iterations)
    test_se_array = cell(1, iterations); % preallocating array for
holding se's

% test structuring elements with increasing size
for i = 1:iterations
    se_test = strel(strel_type, i*increment);
    test_se_array{i} = se_test; % save to array
end
end
```

Resulting Image

After the final structuring element of size 150 has been applied to the image, we end up with the following result

