SYNESTHESIA

Can we hear pictures?

ISC 3222 Final Project Presentation
Dr. Lemmon, Fall 2019

Abelardo Riojas

What is Synesthesia?

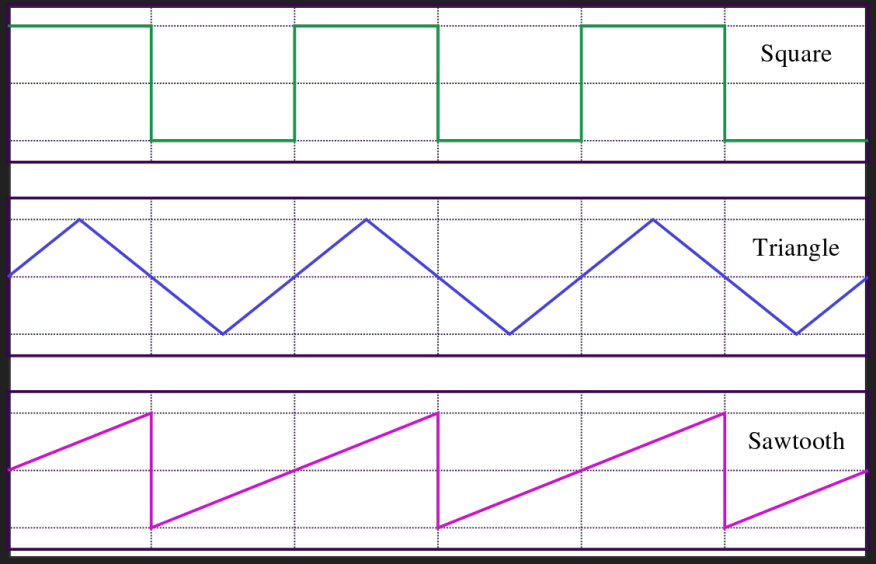
Dictionary.com:

A sensation produced in one modality when a stimulus is applied to another modality, as when the hearing of a certain sound induces the visualization of a certain color.

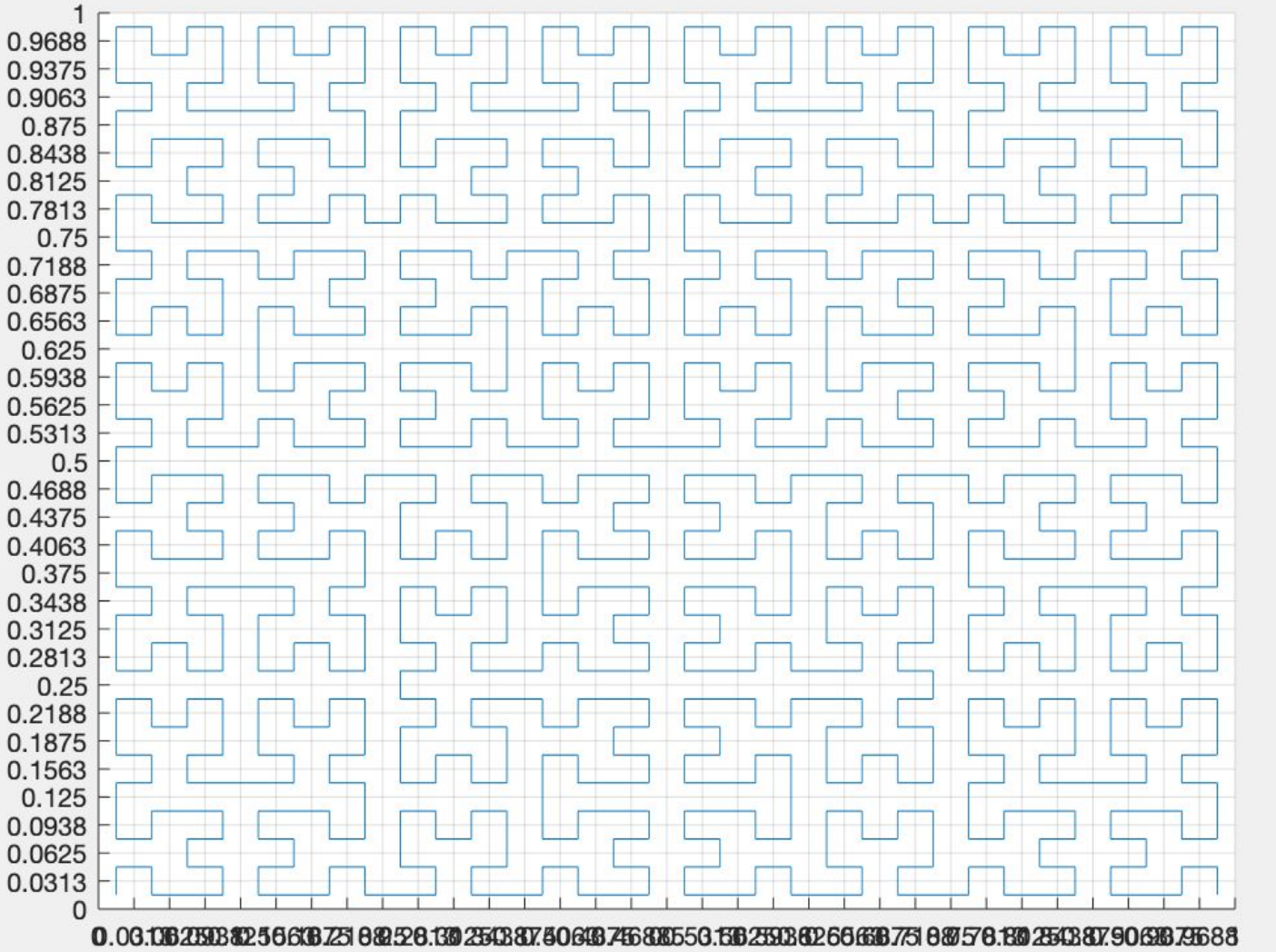
What we’re doing

- Using the hilbert curve to read the pixels of an image
- Using the color values (red, green blue) to generate a sound wave
  - Red = Sawtooth wave
  - Green = Triangle wave
  - Blue = Square wave
- Combining the three waves together to make a sound for an image
- Try to see if people can match pictures to sounds. (Using a visual aid).

What we’re doing



Why


1
0.9688
0.9375
0.9063
0.875
0.8438
0.8125
0.7813
0.75
0.7188
0.6875
0.6563
0.625
0.5938
0.5625
0.5313
0.5
0.4688
0.4375
0.4063
0.375
0.3438
0.3125
0.2813
0.25
0.2188
0.1875
0.1563
0.125
0.0938
0.0625
0.0313
0

THE CODE 😎

```
function [x,y] = hilbert(n)
    %uses a recursive funciton to plot the hilbert curve
    if n<=0
      x=0;
      y=0;
    else
      [xo,yo]=hilbert(n-1);
      %all points are scaled and then shifted to make the curve
      x=.5*[-.5+yo -.5+xo .5+xo  .5-yo];
      y=.5*[-.5+xo  .5+yo .5+yo -.5-xo];
    end
end
```

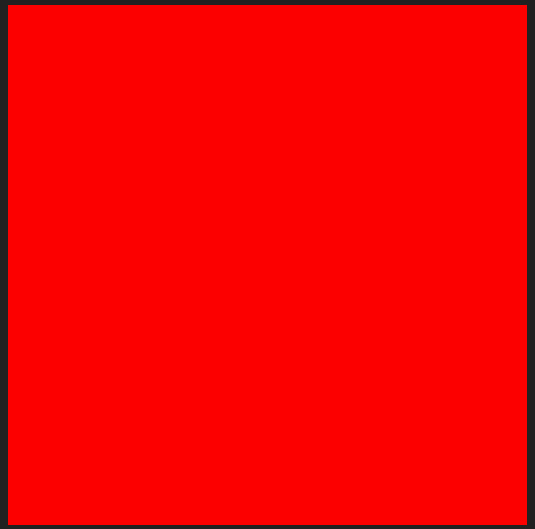
THE CODE 😎

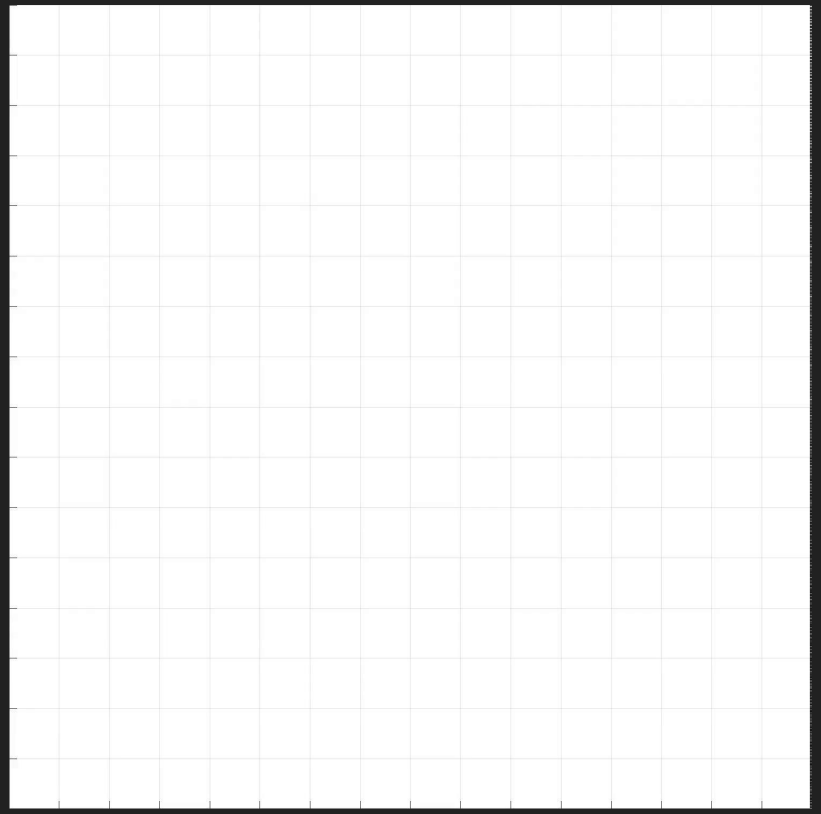
```
function [red, green, blue] = hilbertRead(image)
%this function reads the image's pixels in the order of the hilbert curve
[ m, ~ ] = size ( image );
image = flipud(image); %gotta flip it upsidedown to read it correctly. 1,1 is at the top
order = log2(m);
dim = 2^order;
[x,y] = hilbert(order);
x = x + .5;
y = y + .5;
xy = 2^(order+1)*[x' y']; %scale it so 1,1 is the first pixel
xy = (xy+1)/2;
red = zeros(1,dim*dim);
green = zeros(1,dim*dim);
blue = zeros(1,dim*dim);
for i = 1:dim*dim %generates red green and blue arrays to transform into audio
    col = xy(i,1);
    row = xy(i,2);
    red(1,i) = image(row,col,1);
    green(1,i) = image(row,col,2);
    blue(1,i) = image(row,col,3);
end
end
```

THE CODE 😎

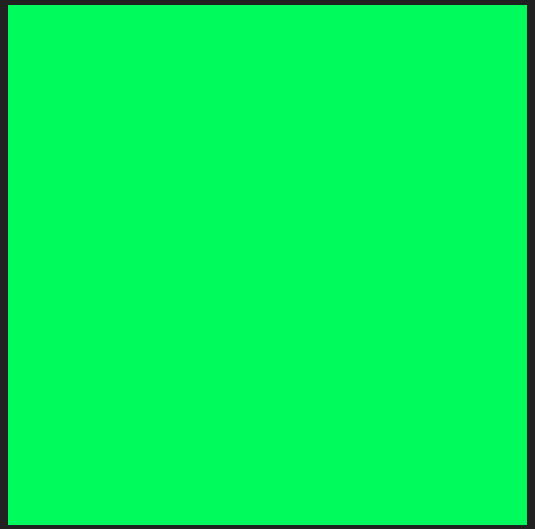
```
%creating a sawtooth wave
T = (dim*dim)*(1/50);
fs = 1000;
t = 0:1/fs:T-1/fs;

saw = sawtooth(2*pi*50*t);
%using the red color values as scalars for each sawtoon wave
%each period is sampled 20 times.
for i = 1:dim^2
    saw(1,1+(20*(i-1)):20*i)=saw(1,1+(20*(i-1)):20*i)*r(1,i);
end

%same as sawtooth, but with green color values.
T = (dim*dim)*(1/50);
fs = 1000;
t = 0:1/fs:T-1/fs;
tri = sawtooth(2*pi*50*t,1/2);
for i = 1:dim^2
    tri(1,1+(20*(i-1)):20*i)=tri(1,1+(20*(i-1)):20*i)*g(1,i);
end

%generating square waves and using blue values as scalars.
t = 0:1/1e3:size(saw,2)/1000;
square = square(2*pi*30*t,50);
for i = 1:dim^2
    square(1,1+(20*(i-1)):20*i)=square(1,1+(20*(i-1)):20*i)*b(1,i);
end
```

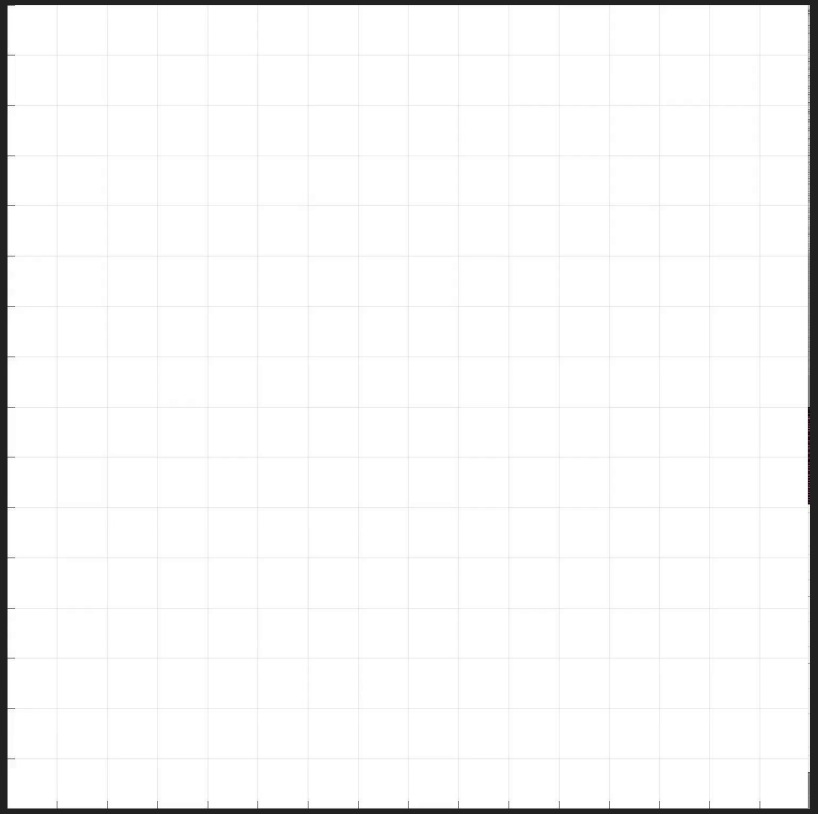
Demonstration: red.



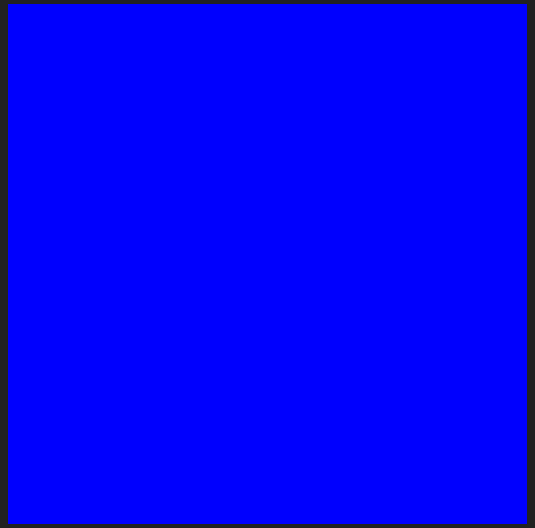


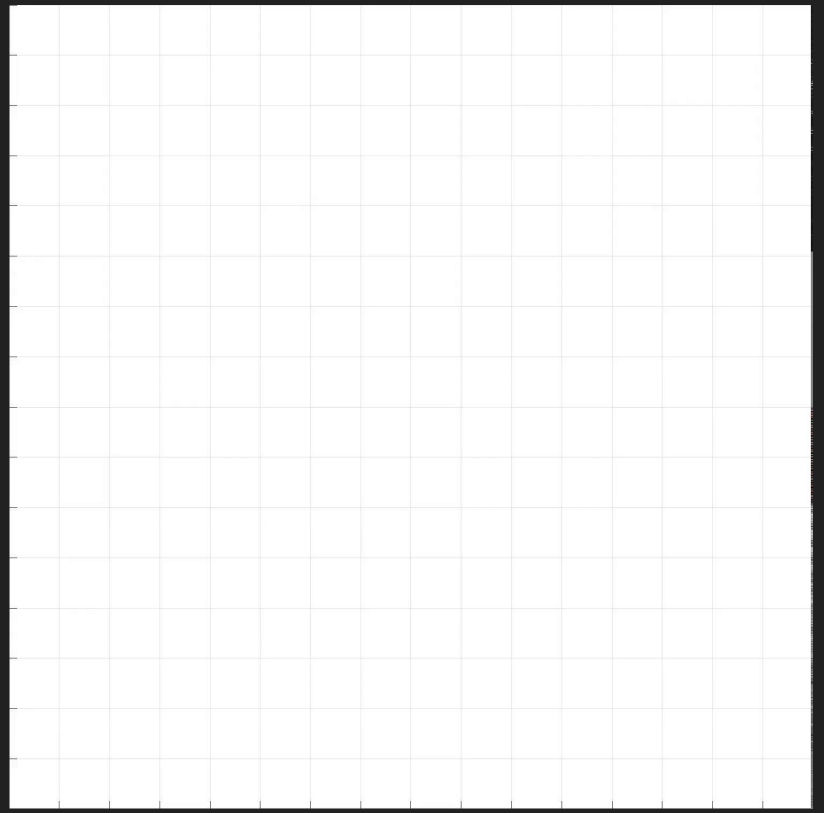
Demonstration: green.



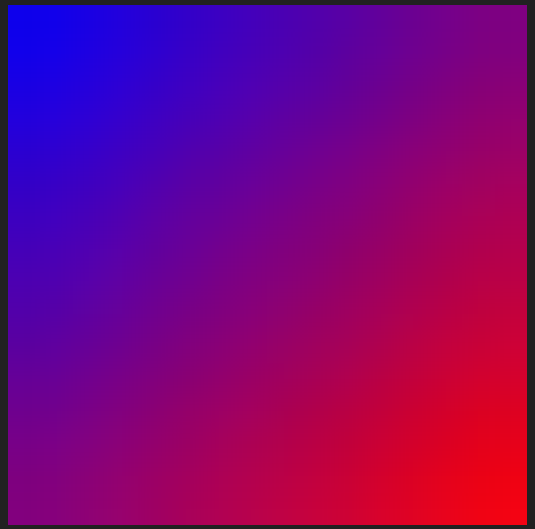


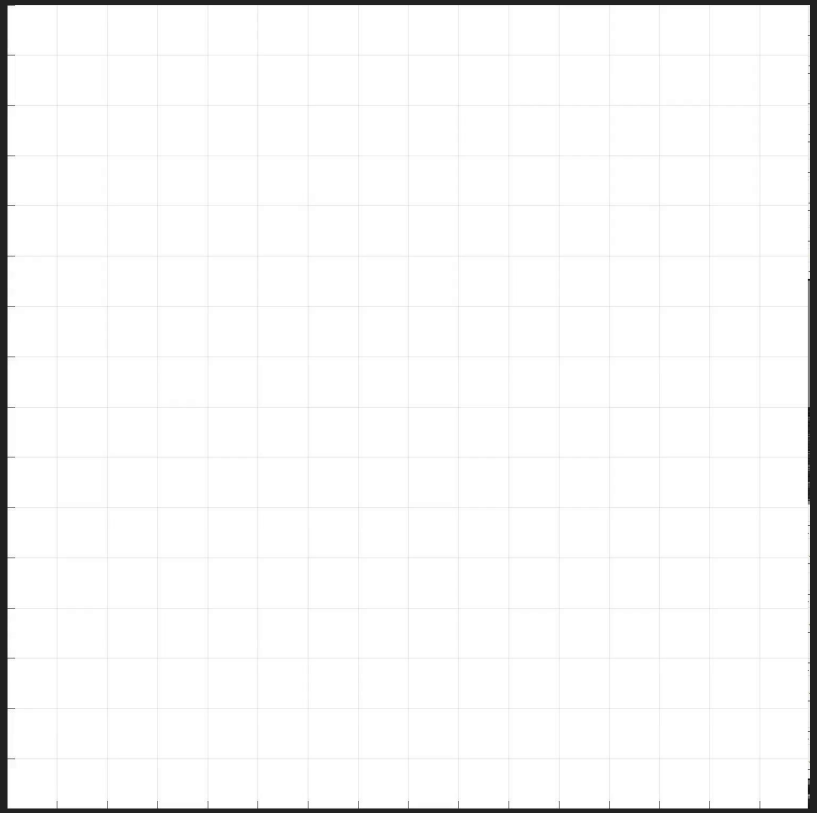
Demonstration: blue.

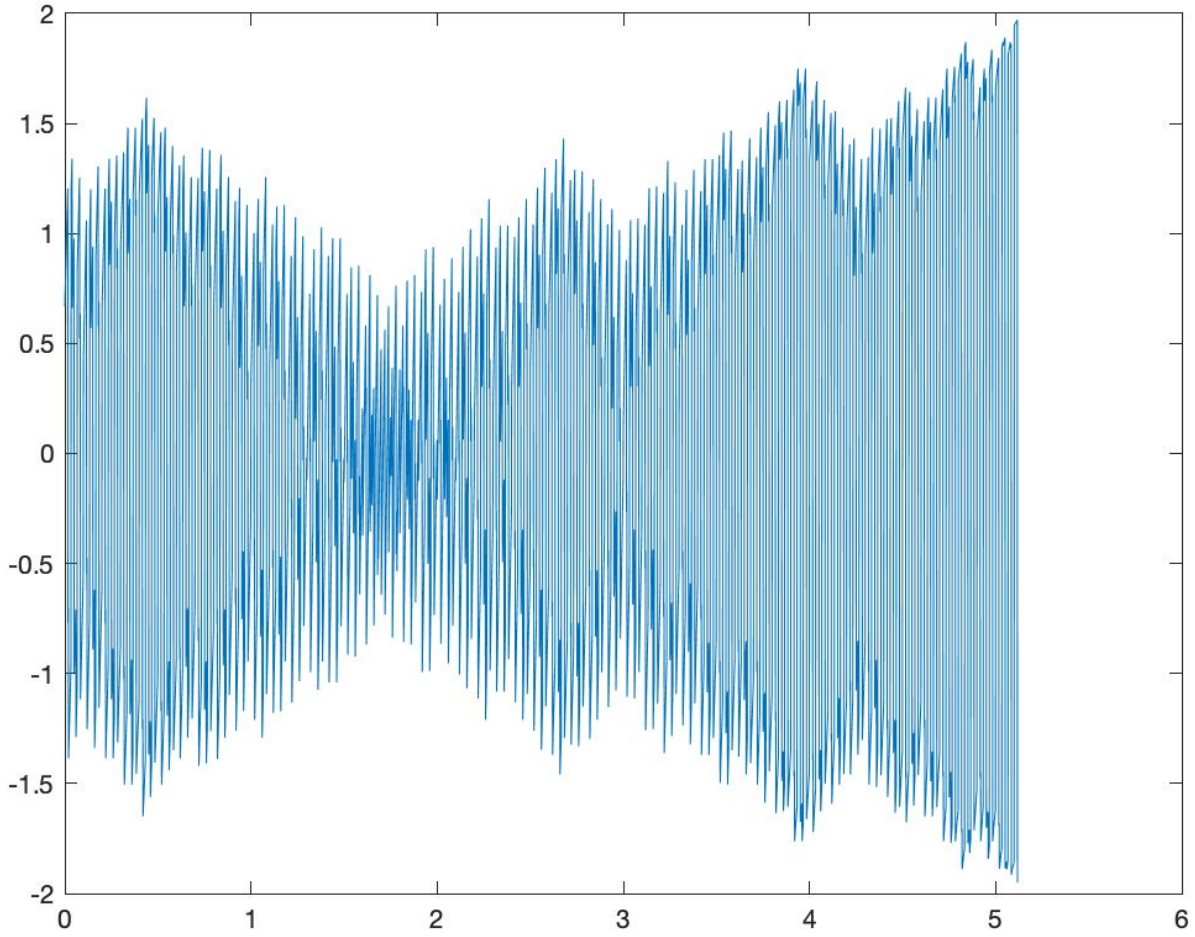




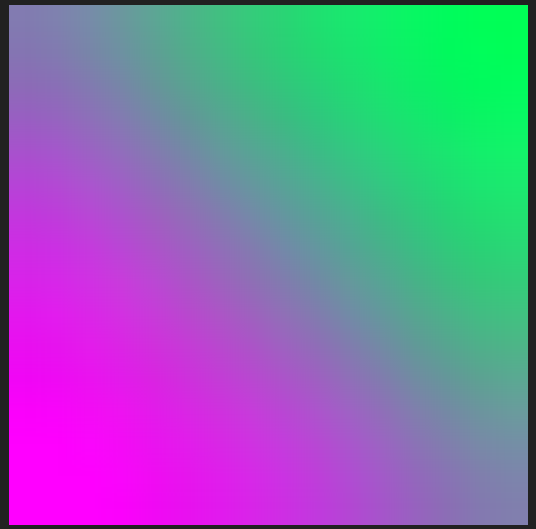
Demonstration: gradient.

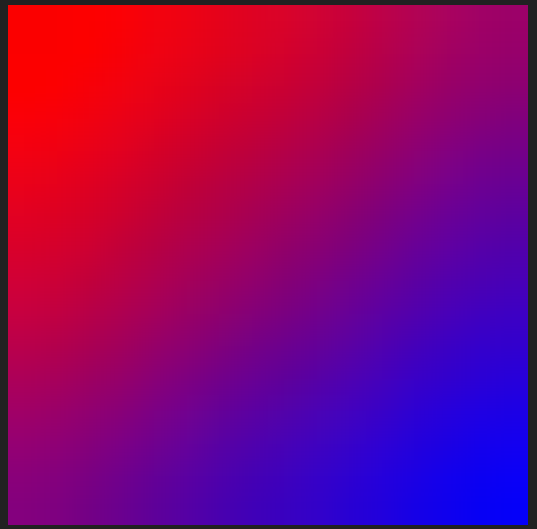


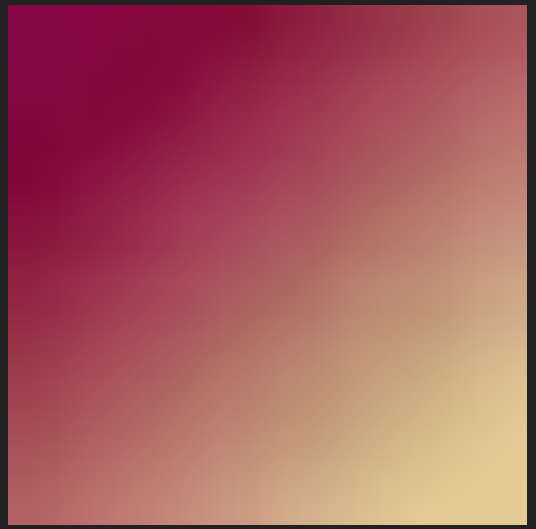



2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
1
2
3
4
5
6

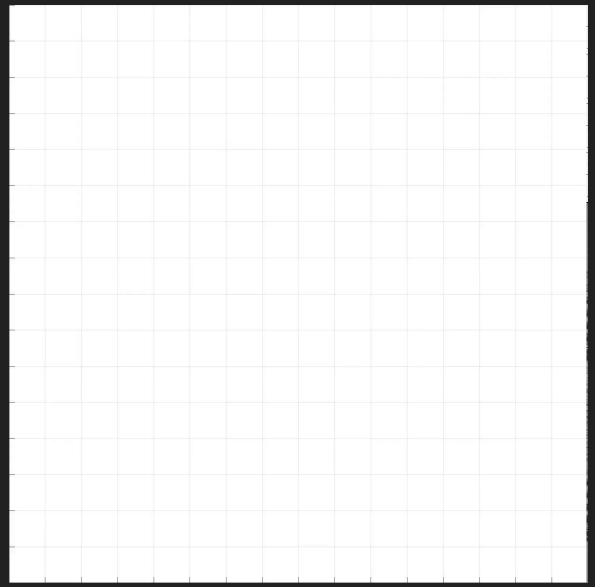
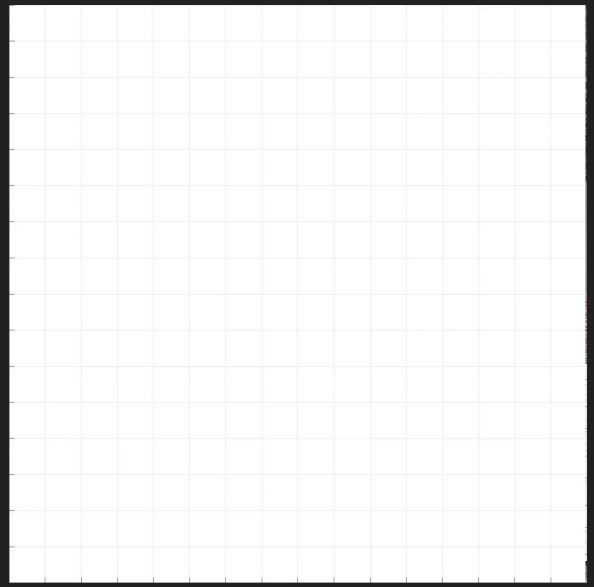
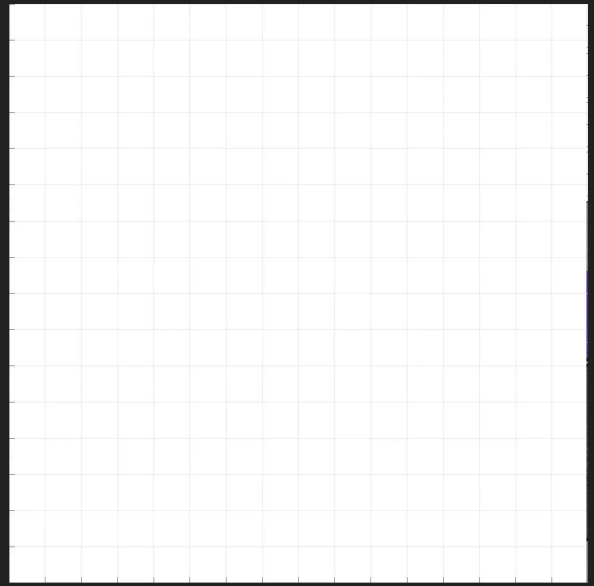
Guessing game!







Choice A	Choice B	Choice C

Answer:

B, C, A!

Conclusion

Can we hear pictures?

Kinda?

Limitations

- The sounds generated are very long, even with a high sample rate
    - A 256x256 size image would produce a sound that would take 21 minutes to listen to!
- One can only really ‘hear’ really simple images like gradients
- Image dimensions have to be square and 2^n

...Any questions?