

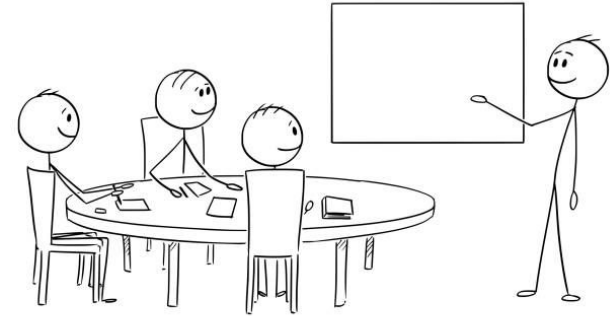


## Lesson 08 - Edge detection Project

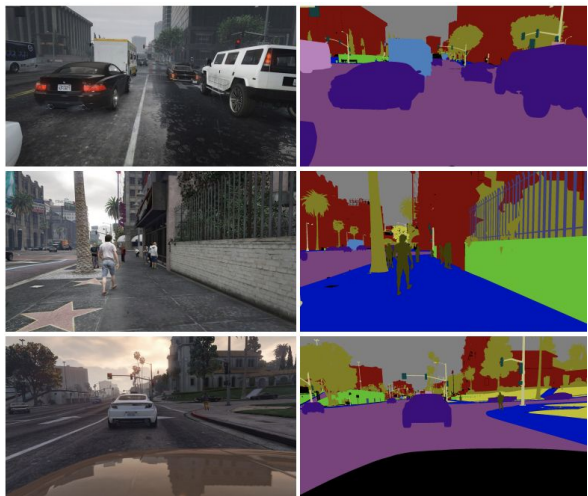
**EQ:** How can we apply edge detection in real life?

### Do Now

**Search and Share:** Using the internet, find a practical application of edge detection or convolution that you find cool or interesting. Summarize how it helps in this situation and come up with a question you might have. **Be prepared to share with the class.**



# Edge detection applications



```

0 2 15 0 0 11 10 0 0 0 0 9 9 0 0 0
0 0 0 4 60 157 236 255 255 177 95 61 32 0 0 29
0 10 16 119 238 255 244 245 243 250 249 255 222 103 10 0
0 14 170 255 255 244 254 255 253 245 255 249 253 251 124 1
2 98 255 228 255 251 254 211 141 116 122 215 251 238 255 49
13 217 243 255 155 33 226 52 2 0 10 13 232 255 255 36
16 229 252 254 49 12 0 0 7 7 0 70 237 252 235 62
6 141 245 255 212 25 11 9 3 0 115 236 243 255 137 0
0 87 252 250 248 215 60 0 1 121 252 255 248 144 6 0
0 13 113 255 255 245 255 182 181 248 252 242 208 36 0 19
1 0 5 117 251 255 241 255 247 255 241 162 17 0 7 0
0 0 0 4 58 251 255 246 254 253 253 120 11 0 1 0
0 0 4 97 255 255 255 248 252 255 244 255 182 10 0 4
0 22 206 252 246 251 241 100 24 113 255 245 255 194 9 0
0 111 255 242 255 158 24 0 0 6 39 255 232 230 56 0
0 218 251 250 137 7 11 0 0 2 62 255 250 125 3
0 173 255 255 101 9 20 0 13 3 13 182 251 245 61 0
0 107 251 241 255 230 98 55 19 118 217 248 253 255 52 4
0 18 146 255 255 247 255 255 255 249 255 240 255 129 0 5
0 0 23 113 215 255 250 248 255 255 248 248 118 14 12 0
0 0 6 1 0 52 153 233 255 252 147 37 0 0 4 1
0 0 5 5 0 0 0 0 0 14 1 0 6 6 0 0
    
```



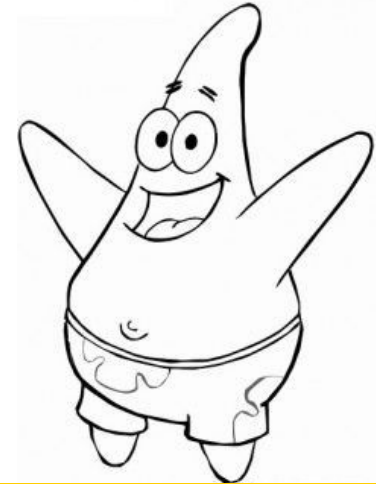
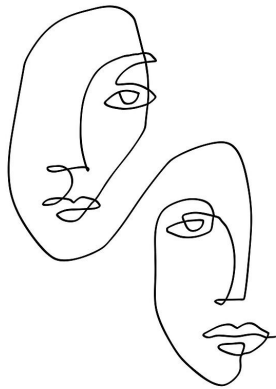
# Seam Carving (content-aware image resizing)





## Line Art Generator - Project

In pairs of your choosing, you will write code that takes an in image path and generates a line art version of that image. The background colors and foreground/line colors should be supplied by the user.

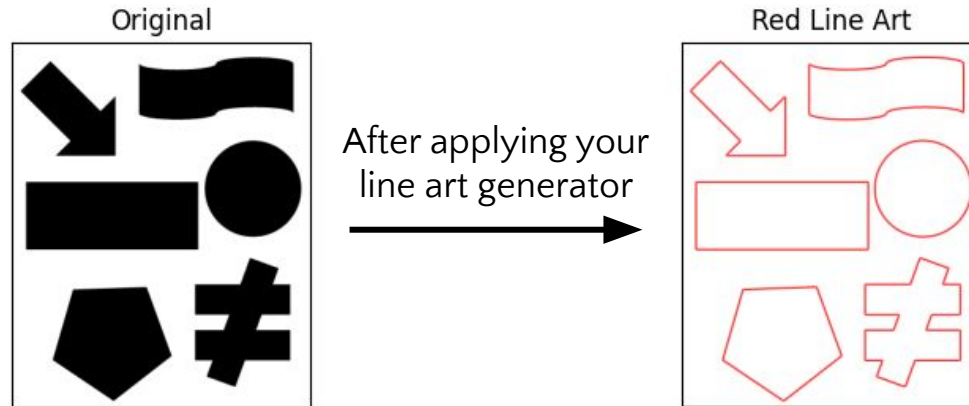




## Line Art Generator - Requirements

Your code **must**:

- ❑ Accept file path, background/foreground color (rgb) as user input
- ❑ Extract the important lines from an image using different kernels
- ❑ Recolor the line image to be the user's colors
- ❑ Formal requirements on Google Classroom





## Line Art Generator - Notes

Your code **should**:

- Account for noisy images
- Normalize images to be in the proper range (0-255)
- Make the color intensity correspond to the strength of the edge
  - [Alpha-blending](#) would be helpful for this
- Use helper functions to reuse code and debug it in sections

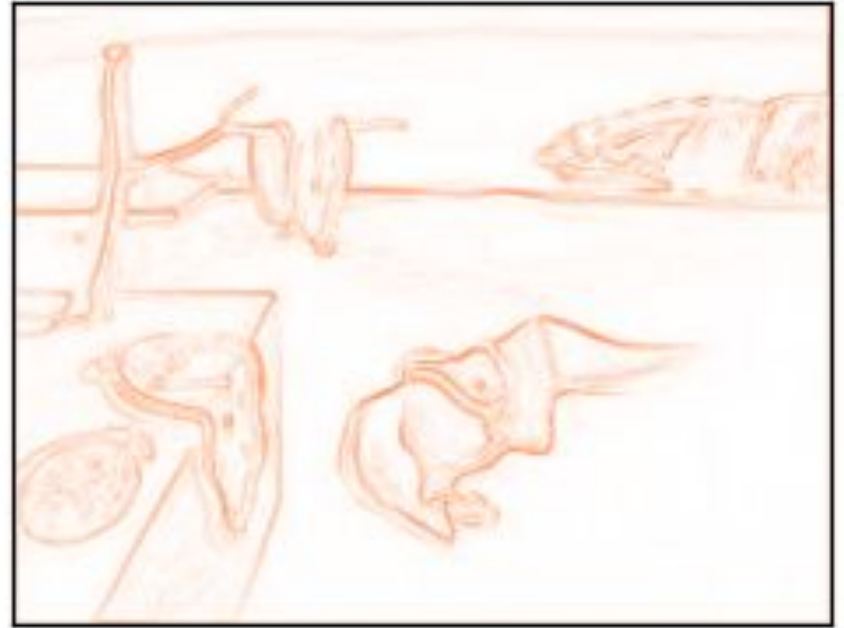
**Protip:** You should definitely refer to class slides, your homework, and class code!

# Edge strength to line intensity

Original



Line Art





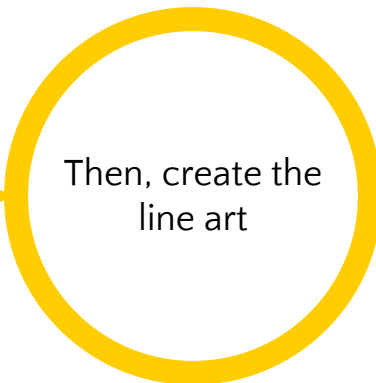
## Line Art Generator - Timeline

You should tackle this project in three stages

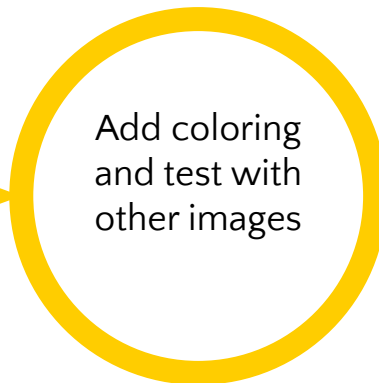
1.



2.



3.







# Let's get it **started!**



## Any **questions** ?

Project description and starter code on Google Classroom.

Project due Monday.