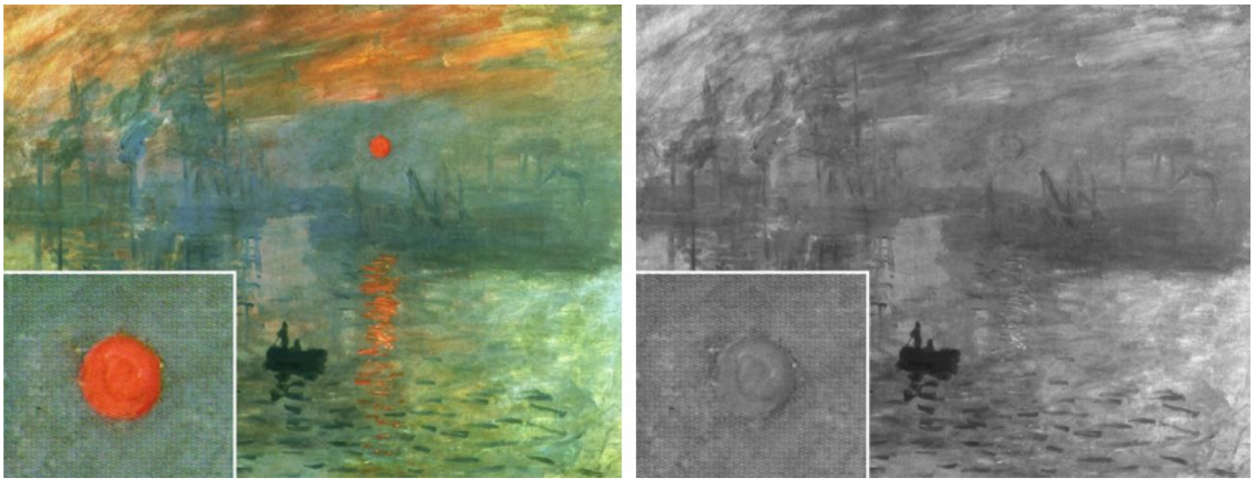
CS/ECE 8690 Computer Vision

Homework 1B – Experiments with Color Edge Detection [60 pts]

Out: Tu Feb 4 **Spring 2025**
Due: Th Feb 13



Edges carry important semantic information about images and are essential to many computer vision pipelines including image segmentation and object detection or recognition. Most edge detection techniques have been developed for grayscale images. These techniques may fail when used on color images that provide additional information.

<u>Edge Detection Strategies on Color Images:</u>

- **Approach-1:**
    1. Convert color image to grayscale.
    2. Compute gradients and perform edge detection on grayscale image (as described in class).

    Problem: Noticeable edges between iso-luminant colors (colors that have the same luminance) are useful cues but fail to be detected by grayscale edge operators.
- **Approach-2:**
    1. Compute gradients on each color band separately.
    2. Combine the outputs.

    Problem: if the gradients in each of the color bands are summed up, the signed gradients may cancel each other!
- **Approach-3:**
    1. Detect edges independently in each band
    2. Take the union of these.

    Better, but might lead to thickened or doubled edges that are hard to link.
- **Approach-4:**
    1. Instead of using the raw RGB space, image can be converted to a more perceptually uniform color space such as L*a*b*
    2. Select one of the color channels to detect edges.

    Problem: fails to detect luminance edges.

Edge Detection using Color Structure Tensor:

As part of this assignment, you will perform edge detection on color images using 2D color structure tensor.

You will analyze the image edges at two scales:

(1) a small scale to detect fine details in the input image;

(2) a larger scale to detect prominent structures in the image.

You can use built-in functions in Python, numpy, OpenCV for image filtering and convolution.

Step-1: Compute 2D Color Structure Tensor:

$$Jc(x,y) = \begin{bmatrix} \sum_{i=R,G,B} G(\sigma) \otimes (I_i(x,y)_x^2 & \sum_{i=R,G,B} G(\sigma) \otimes (I_i(x,y)_x I_i(x,y)_y) \\ \sum_{i=R,G,B} G(\sigma) \otimes (I_i(x,y)_x I_i(x,y)_y) & \sum_{i=R,G,B} G(\sigma) \otimes I_i(x,y)_y^2 \end{bmatrix}$$

Where

- $G(\sigma)\otimes$ denotes convolution with a Gaussian filter of scale σ
- $I_i(x,y)_x$ denotes first derivative of image channel i in x at image pixel (x,y).

Note: You can compute the derivatives using Sobel filters.

Step-2: Compute a scalar edge indicator from 2D Color Structure Tensor.
Trace(Jc) is a good candidate (trace of a square matrix is sum of its diagonal elements).

Homework Tasks:

1) Compute and display Ix, image gradient in x for R,G,B channels. Note that the image gradients can be negative.
2) Compute and display Iy, image gradient in y R,G,B channels
3) Compute and display the elements of the 2D color structure tensor as described above
4) Compute and display trace of the 2D color structure tensor
5) Convert original color image to grayscale. Compute Ix and Iy for the grayscale image. Display gradient magnitude.

Report: Generate the above outputs for two scales (for two $G(\sigma)$) for the given test image.

Submission instructions: Your submission should include a report (including output images & your interpretation of the outputs) and associated programs (as separate files).

References:

[1] Slides for Lecture 4 and 5 on Canvas

[2] Szeliski, Computer Vision Book, Chapter 7.2 (https://szeliski.org/Book/)

[3] Van De Weijer, Joost, Theo Gevers, and Arnold WM Smeulders. "Robust photometric invariant features from the color tensor." *IEEE Transactions on Image Processing* 15, no. 1 (2005): 118-127.

Sample iso-luminant image:

https://commons.wikimedia.org/wiki/Category:Impression,_soleil_levant_by_Claude_Monet