

Project Statement

Converter from Image to Pencil Sketch

1. Introduction and Context

The development of this digital image processing application involves the utilization of Python with the OpenCV library. The main objective of this application is to apply a series of algorithmic transformations to a regular color photograph in order to generate a high-quality, monochrome result that would visually resemble a hand-drawn pencil sketch. This is a basic computer vision exercise; the task should show how to manipulate the color space, filter, and blend pixels.

2. Problem Definition

The problem is defined as programmatically emulating the artistic effect of a pencil sketch. The successful digital sketch should:

- **Be Monochrome:** rely only on intensity (luminosity) variations.
- **Edge Highlight and Details:** Outline the contours of the subject clearly.
- **Shading:** Soft gradients give an impression of depth and texture- such as simulating soft pencil lead.

Standard photographic images contain rich colour and detailed texture that needs to be systematically stripped away and reinterpreted in order to create this particular artistic style.

3. Technical Approach and Methodology

Conversion relies on an accurate four-step sequence of operations that emulates the Color Dodge blending mode-which is widely employed within digital art programs:

Step 1: Grayscale Conversion

The input image is first converted to a single channel grayscale image from the BGR color space.

Step 2: Inverting and Blurring

First, it inverts the grayscale image to form a 'negative', then applies a Gaussian Blur filter. The blur suppresses high-frequency details like noise and fine texture, leaving smooth gradients representing the broad areas of shading.

Step 3: Second Inversion

The blurred and inverted image is inverted again; this will yield the main layer for the final blending computation.

Step 4: Color Dodge Calculation

The final pencil sketch is obtained by doing a mathematical division between the original grayscale image and the second inverted, blurred image scaled by 256. Following is the Color Dodge formula:

$$\text{Sketch} = \frac{\text{Grayscale}}{\text{Inverted and Blurred Image}} \times 256,$$

Division brightens the areas where the shading layer is dark, creating sharp, defined edges, giving luminous highlights typical for a pencil drawing.

4. Deliverables and Outcome

Deliverable: A working Python script that takes as input an image file path and produces the following output visualizations:

- The Original Input Image.
- The Grayscale Intermediate Image.
- Final Transformed Pencil Sketch Image.

Outcome: The outcome will be a robust and runnable program, demonstrating how the efficacy of various sequential image processing techniques can be combined to produce a complex artistic visual result.