

Assessment Problem Statement:

Image Transformer Tool

Apply OpenCV Functions via GUI

Objective:

Design and develop a Tkinter-based GUI application that allows a user to:

- Upload a static image
- Apply **different image processing transformations** using OpenCV
- Preview the results within the GUI
- Save the transformed image

Functional Requirements:

1. Image Upload

- Allow the user to upload a static image (.jpg, .png, etc.).
- Display the uploaded image within the GUI.

2. Transformation Options

- Provide buttons or dropdowns to apply **at least 10 different image processing functions** using OpenCV.

Suggested Transformations:

- Convert to Grayscale
- Gaussian Blur
- Median Blur
- Sobel Edge Detection
- Canny Edge Detection
- Thresholding (Binary or Adaptive)
- Image Rotation
- Image Resizing
- Erosion

- Dilation
- Brightness Adjustment
- Contrast Adjustment
- Flip (Horizontal / Vertical)

3. Transformation Controls

- If applicable, provide sliders or inputs to tweak parameters (e.g., kernel size for blur, rotation angle, etc.).

4. Live Preview

- The user should be able to see the result of each transformation immediately on the GUI.

5. Reset & Save

- Button to reset to the original image.
- Button to save the transformed image to disk.



Technical Constraints:

- GUI must be built using Tkinter.
- Image transformations must be done using **OpenCV** (cv2).
- Logic and GUI code must reside in **separate Python files**.
- Application should handle invalid uploads gracefully (e.g., non-image files).