Report : Image Processing Toolkit

Student Name: *M. Rohan*

Roll No: *22671A7334* ***(****AIML – A)*

Subject: *Computer Vision*

Task: 3 – *Image Processing Toolkit Submission*

---

1. Introduction

This project demonstrates basic and advanced image processing operations using **Python, OpenCV, and Streamlit**.
The toolkit provides an interactive GUI where users can upload an image, apply transformations, filters, and enhancements, and visualize results side-by-side.

---

2. Notes on Image Fundamentals

2.1 CMOS vs CCD Sensors

- **CCD (Charge-Coupled Device):**
    - Captures images with low noise.
    - Better for scientific and medical imaging.
    - Expensive and consumes more power.
- **CMOS (Complementary Metal-Oxide Semiconductor):**
    - Common in smartphones and digital cameras.
    - Lower cost and power consumption.
    - Slightly noisier compared to CCD.

Comparison Table:

Feature	**CCD**	**CMOS**
Cost	Expensive	Cheaper
Power Use	High	Low
Noise	Low	Higher
Applications	Scientific, Med	Phones, DSLRs

2.2 Sampling & Quantization

- **Sampling**: Dividing the image into a grid of pixels.
- **Quantization**: Assigning discrete intensity values to each pixel.

Example:
A grayscale image of 256×256 pixels with 8-bit quantization → each pixel stores intensity between 0–255.

- Higher sampling → better resolution.
- Higher quantization → smoother intensity levels.

2.3 PSF (Point Spread Function)

- PSF describes how a single point of light spreads in an imaging system.
- Ideal system: point remains sharp.
- Real systems: due to diffraction, noise, lens effects → point spreads (blurs).
- Important in **restoration and deblurring algorithms**.

---

3. Toolkit Design

- **GUI Framework:** Streamlit
- **Backend Library:** OpenCV (image operations), NumPy (matrix representation)
- **Features Implemented:**
    - Color conversions (RGB, HSV, YCbCr, Gray)
    - Transformations (rotate, scale, translate, affine, perspective)
    - Filtering & Morphology (Gaussian, Median, Mean, Sobel, Laplacian, erosion, dilation, opening, closing)
    - Enhancement (Histogram Equalization, Contrast Stretching, Sharpening)
    - Edge Detection (Sobel, Canny, Laplacian)
    - Compression (JPG, PNG, BMP saving & size comparison)

---

4. Screenshots of Toolkit Results

Deploy

PIXEL PLAYGROUND

Upload an image

Drag and drop file here
Limit 200MB per file • JPG, JPEG, PNG
Browse files

obito.jpg 0.5MB

Original Image

Operations Menu

Choose Category

Image Info

- Image Info
- Color Conversions
- Transformations
- Filtering & Morphology
- Enhancement
- Edge Detection
- Compression

Deploy

obito.jpg 0.5MB

Original Image

Processed Image

Operations Menu

Choose Category

Color Conversions

Select Conversion

RGB → BGR

- RGB → BGR
- RGB → HSV
- RGB → YCbCr
- RGB → Grayscale

- JPEG
- JPG
- PNG

Save Image

Deploy

obito.jpg 0.5MB

Original Image

Processed Image

Operations Menu

Choose Category

Transformations

Select Transformation

Rotation

Rotation Angle

-27

Save Processed Image

Select Format

- JPEG
- JPG
- PNG

Save Image

Deploy

obito.jpg 0.5MB

Operations Menu

Choose Category

Filtering & Morphology

Select Operation

Mean Blur

- Gaussian Blur
- Mean Blur
- Median Blur
- Sobel
- Laplacian
- Dilation
- Erosion
- Opening

PNG

Save Image

Original Image

Processed Image

Deploy

obito.jpg 0.5MB

Operations Menu

Choose Category

Enhancement

Select Enhancement

Histogram Equalization

- Histogram Equalization
- Contrast Stretching
- Sharpen

Select Format

JPEG
JPG
PNG

Save Image

Original Image

Processed Image

Deploy

obito.jpg 0.5MB

Operations Menu

Choose Category

Edge Detection

Select Method

Canny

- Sobel
- Canny
- Laplacian

40

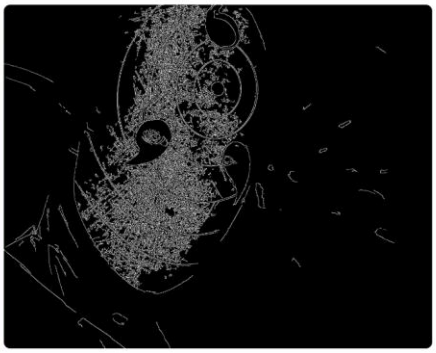
Save Processed Image

Select Format

JPEG
JPG
PNG

Original Image

Processed Image



5. Explanation of Algorithms Used

- **Color Conversions:**
    - HSV → separates color from intensity.
    - YCbCr → used in video compression.
    - Grayscale → average or weighted conversion of RGB channels.
- **Transformations:**
    - Rotation (by angle θ) uses affine matrix.
    - Scaling multiplies pixel coordinates.
    - Translation shifts image by (x, y).
- **Filtering:**
    - Gaussian filter → smooths image by weighted average.
    - Median filter → removes salt & pepper noise.
    - Sobel/Laplacian → edge enhancement by gradient detection.
- **Enhancement:**
    - Histogram Equalization → improves contrast.
    - Sharpening → highlights edges.
- **Edge Detection:**
    - Canny → multi-stage algorithm (gradient + non-max suppression + hysteresis).
- **Compression:**
    - Saving in JPG reduces size (lossy).
    - PNG/BMP preserve details (lossless, but larger).

---

6. Conclusion

This project successfully combines **theory and practice** of image processing.
The toolkit helps visualize concepts such as sampling, quantization, color models, transformations, filtering, and compression in a hands-on way.