📑 **Report: Image Processing & Analysis Toolkit (GUI in Python + OpenCV + Streamlit)**

1. Introduction

This project implements a **Graphical User Interface (GUI) application** for fundamental **Image Processing and Computer Vision operations**. The toolkit provides both theoretical and practical insights into image processing by allowing users to **upload an image, apply transformations, filters, enhancements, and compression**, and compare results interactively.

The application is built using **Python, OpenCV, NumPy, Pillow, Matplotlib, and Streamlit**. It supports both **basic** (color conversions, filtering) and **advanced operations** (affine/perspective transforms, CLAHE, edge detection). Users can visualize results side-by-side and dynamically adjust parameters using interactive controls.

---

2. System Design

2.1 Tools & Libraries

- **Streamlit** → GUI framework
- **OpenCV (cv2)** → Image processing functions
- **NumPy** → Array & matrix operations
- **Pillow** → Image loading and format handling
- **Matplotlib** → Histogram plotting

2.2 GUI Layout

1. **Menu Bar**
    - Open: Upload an image
    - Save: Save processed image in PNG/JPEG/BMP
    - Exit: Close application
2. **Left Panel (Sidebar)**
    - Categories: Image Info, Color Conversions, Transformations, Bitwise Ops, Filtering & Morphology, Enhancement, Edge Detection, Compression, Extras
    - Parameter sliders: kernel size, rotation angle, scaling factor, thresholds
3. **Right Panel (Main Area)**
    - Display Original vs Processed image

    - Split screen comparison (optional)
    - Histograms (optional)
4. **Status Bar**
    - Shows dynamic image properties: dimensions, DPI, file format, file size

---

3. Implemented Modules

3.1 Image Info

- Resolution, shape, channels, format, file size
- Displays metadata in JSON format

3.2 Color Conversions

- RGB ↔ BGR
- RGB ↔ HSV
- RGB ↔ YCbCr (implemented mathematically)
- RGB ↔ Grayscale
- HSV ↔ RGB

3.3 Transformations

- Rotation (with adjustable angle & scaling factor)
- Scaling (up/down)
- Translation (shift)
- Affine Transform (3-point mapping)
- Perspective Transform (4-point warp)

3.4 Bitwise Operations

- NOT, AND, OR, XOR (requires second image)
- Supports automatic resizing of the second image

3.5 Filtering & Morphology

- **Filters**: Gaussian, Mean, Median, Sobel, Laplacian
- **Morphology**: Dilation, Erosion, Opening, Closing

3.6 Enhancement

- Histogram Equalization (grayscale)

- CLAHE (Contrast Limited Adaptive Histogram Equalization)
- Contrast Stretching
- Sharpening filter (custom kernel)

3.7 Edge Detection

- Canny (with threshold sliders)
- Sobel (combined magnitude)
- Laplacian

3.8 Compression & File Handling

- Save image in **JPG, PNG, BMP**
- File size comparison for different formats
- Download processed image

3.9 Bonus Features

- Split-screen comparison (original vs processed)
- Webcam snapshot mode (with live processing)
- Interactive histograms

---

4. Algorithms & Techniques Used

1. **Sampling & Quantization**
    - Images represented as discrete pixels (sampling)
    - Intensity values stored with limited precision (quantization)
2. **Point Spread Function (PSF)**
    - Describes how a point source of light is spread in imaging
    - Filters like Gaussian approximate real-world PSFs
3. **Filters**
    - **Gaussian Blur**: Low-pass filter, smooths image
    - **Median Filter**: Removes salt-and-pepper noise
    - **Sobel & Laplacian**: Gradient-based edge detection
4. **Histogram Equalization**
    - Redistributes intensity values to improve contrast

    - CLAHE prevents over-amplification of noise
5. **Transformations**
    - **Affine**: Preserves collinearity & parallelism
    - **Perspective**: Allows 3D-like transformations
6. **Compression**
    - **JPEG**: Lossy compression, smaller size
    - **PNG**: Lossless, larger size but high quality
    - **BMP**: Uncompressed, very large files
7. **CMOS vs CCD (Image Sensors)**
    - **CCD**: Better light sensitivity, less noise, but costly & power-hungry
    - **CMOS**: Cheaper, lower power, faster readout (dominant in modern devices)

---

5. Results & Screenshots

(📷 Add screenshots of each operation: filtering, edge detection, enhancement, compression comparison)

- Side-by-side display of original vs processed images
- Split-screen comparison for intuitive visualization
- Histograms for contrast/enhancement analysis

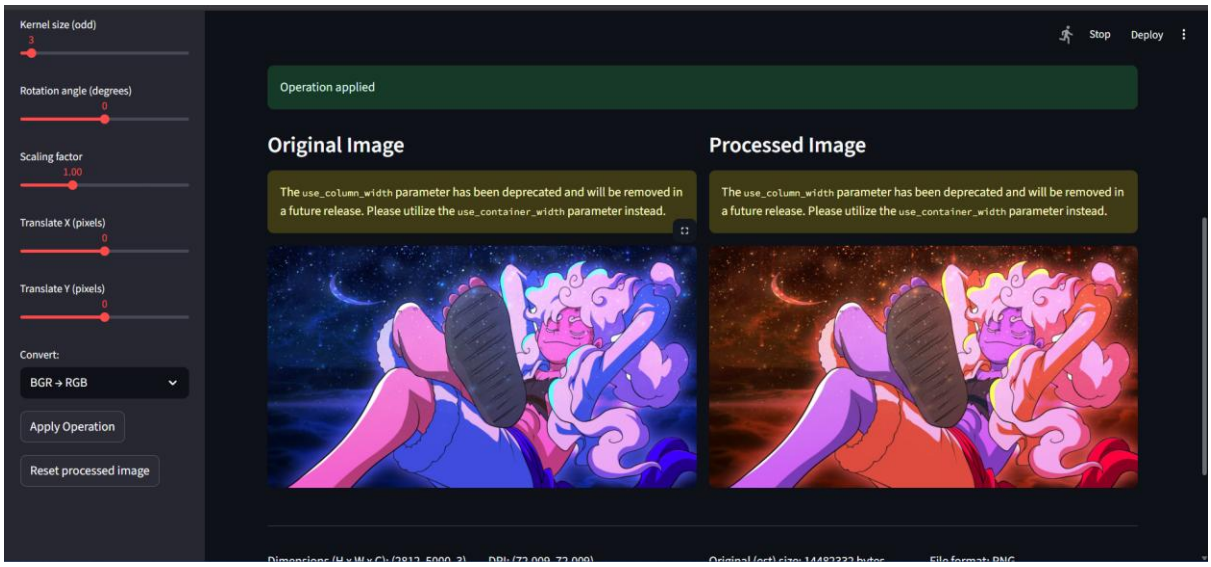
---

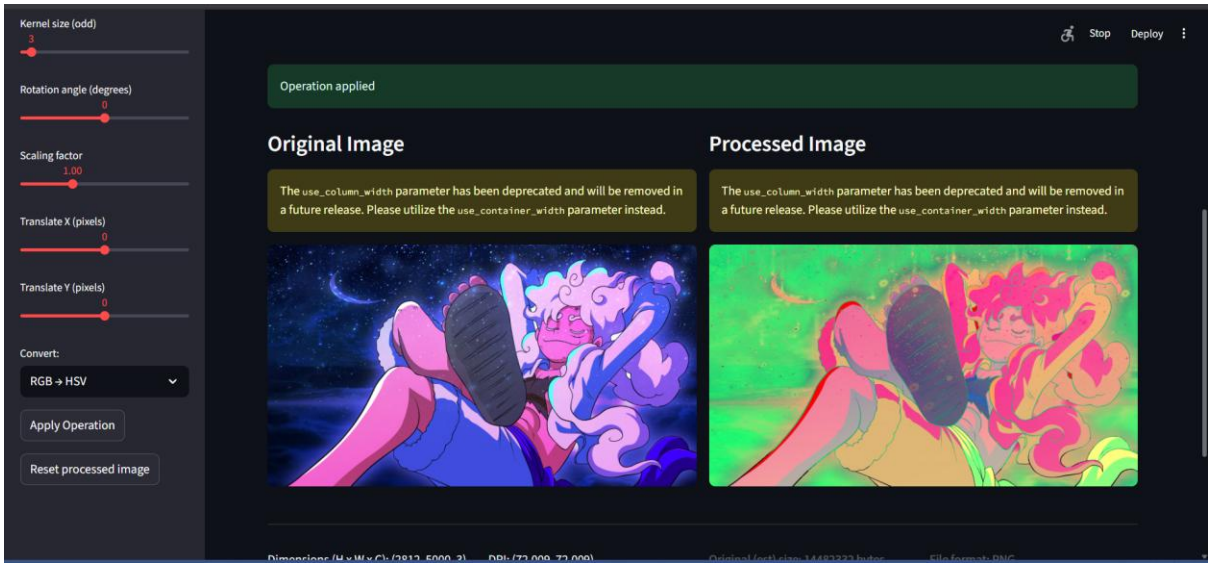
6. Conclusion

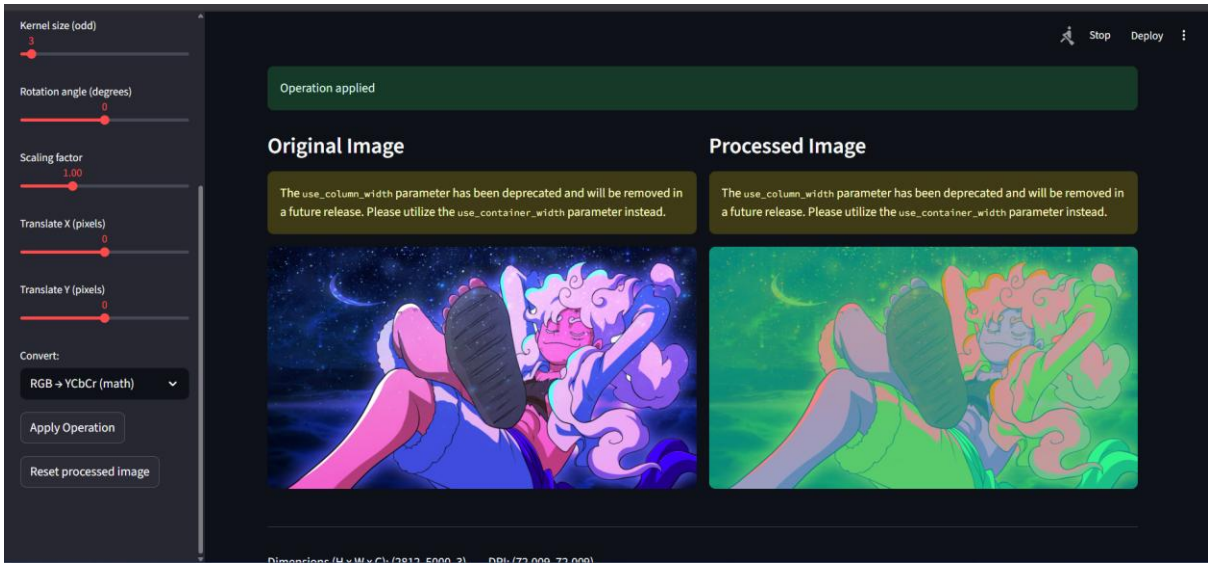
The **Image Processing & Analysis Toolkit** successfully integrates **theory with practice** by providing an interactive environment to test fundamental computer vision operations. It demonstrates:

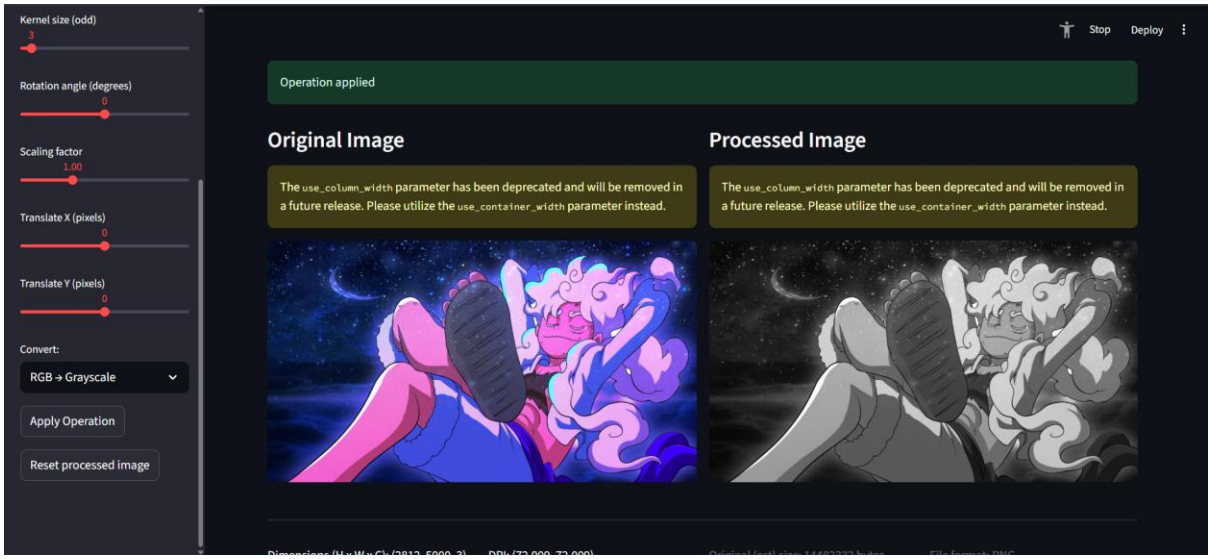
- Core image processing transformations
- Filtering and enhancement techniques
- Compression trade-offs
- GUI-based visualization of algorithms

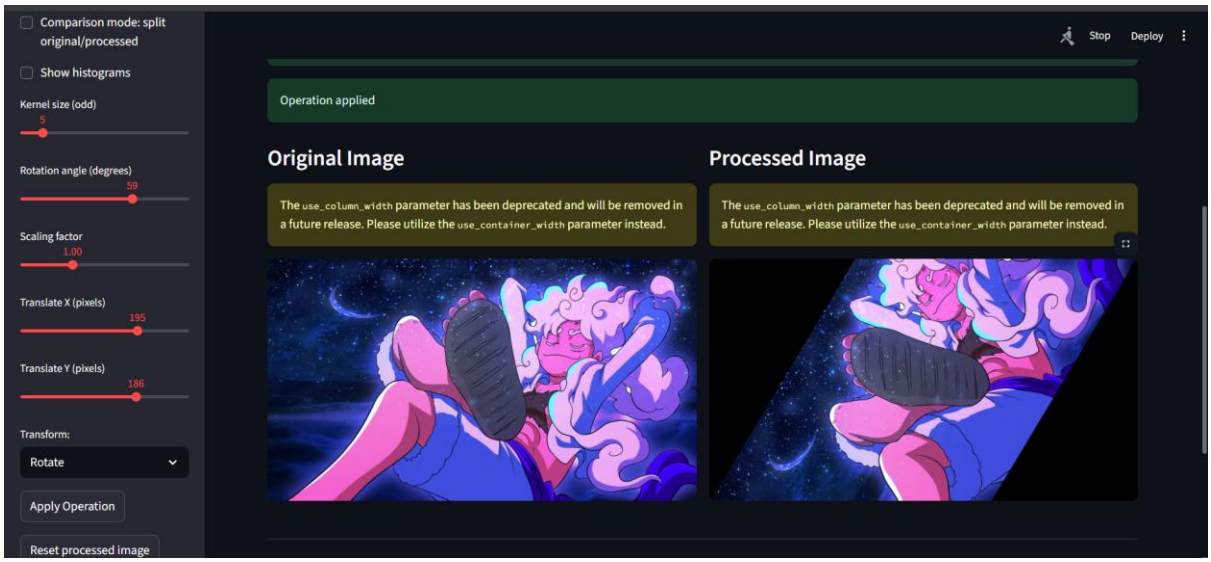
This project bridges the gap between **image processing fundamentals** and **real-world applications**, making it suitable for both beginners and intermediate learners.

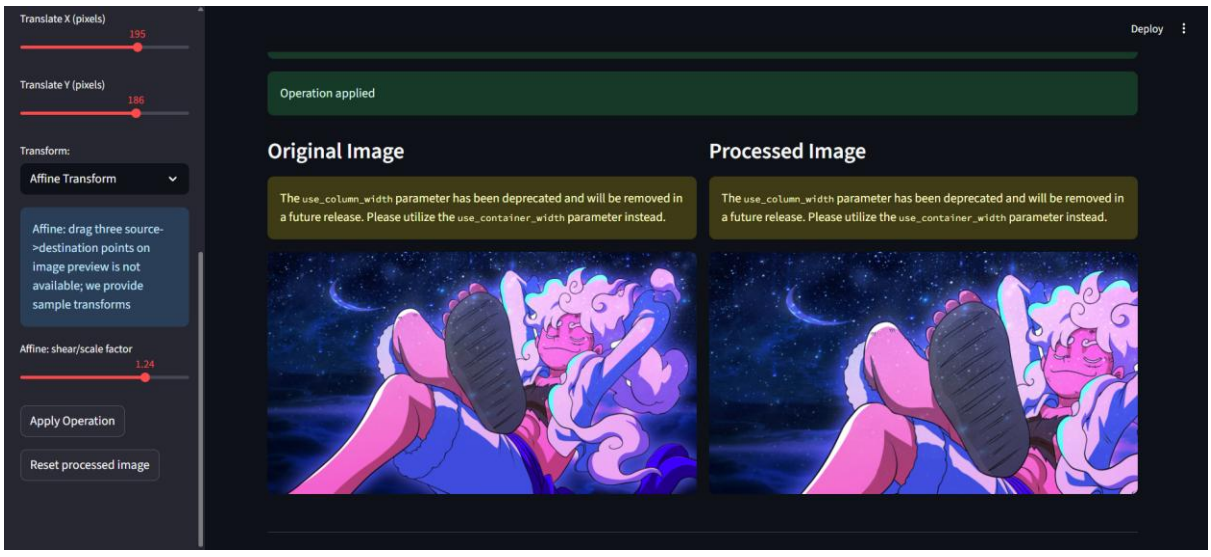

Kernel size (odd)
3
Rotation angle (degrees)
0
Scaling factor
1.00
Translate X (pixels)
0
Translate Y (pixels)
0
Convert:
BGR → RGB
Apply Operation
Reset processed image
Stop
Deploy
Operation applied
Original Image
Processed Image
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.

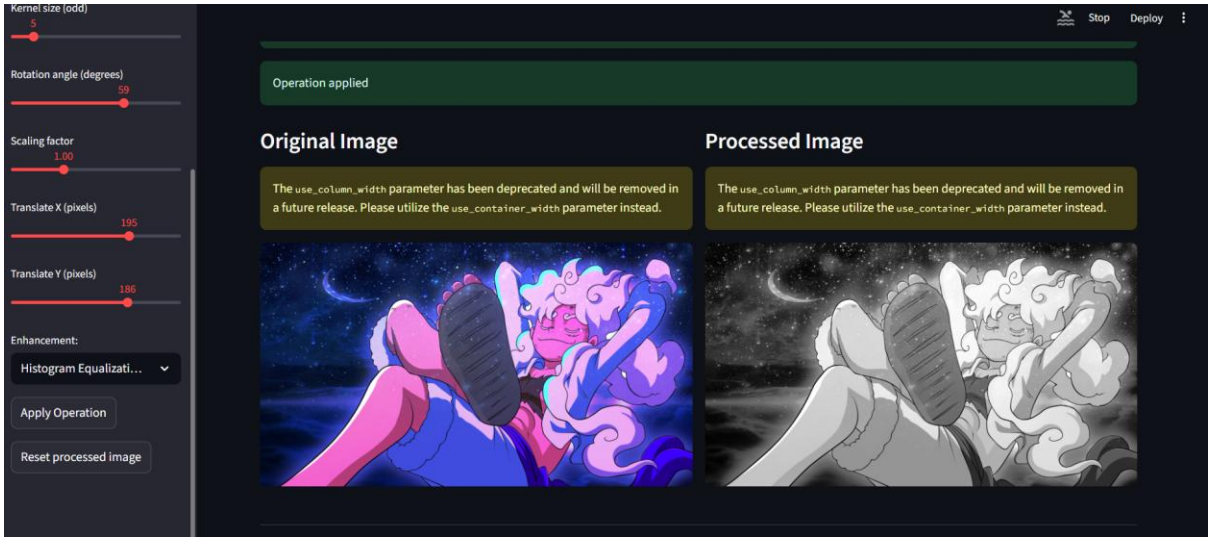

Kernel size (odd)
3
Rotation angle (degrees)
0
Scaling factor
1.00
Translate X (pixels)
0
Translate Y (pixels)
0
Convert:
RGB → HSV
Apply Operation
Reset processed image
Stop
Deploy
Operation applied
Original Image
Processed Image
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.

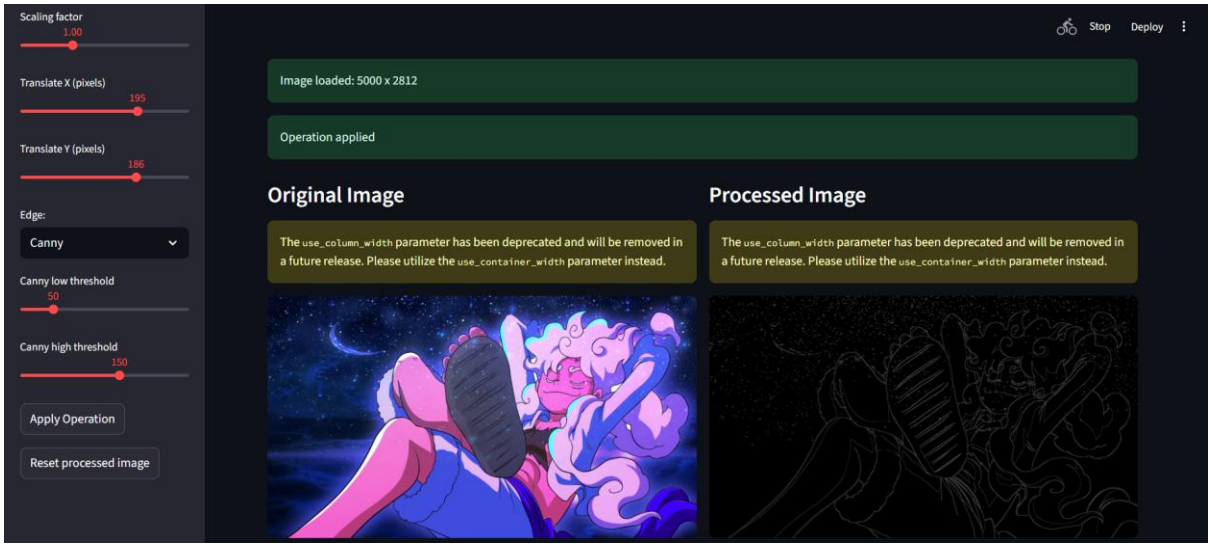

Kernel size (odd)
3
Rotation angle (degrees)
0
Scaling factor
1.00
Translate X (pixels)
0
Translate Y (pixels)
0
Convert:
RGB → YCbCr (math)
Apply Operation
Reset processed image
Stop
Deploy
Operation applied
Original Image
Processed Image
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.

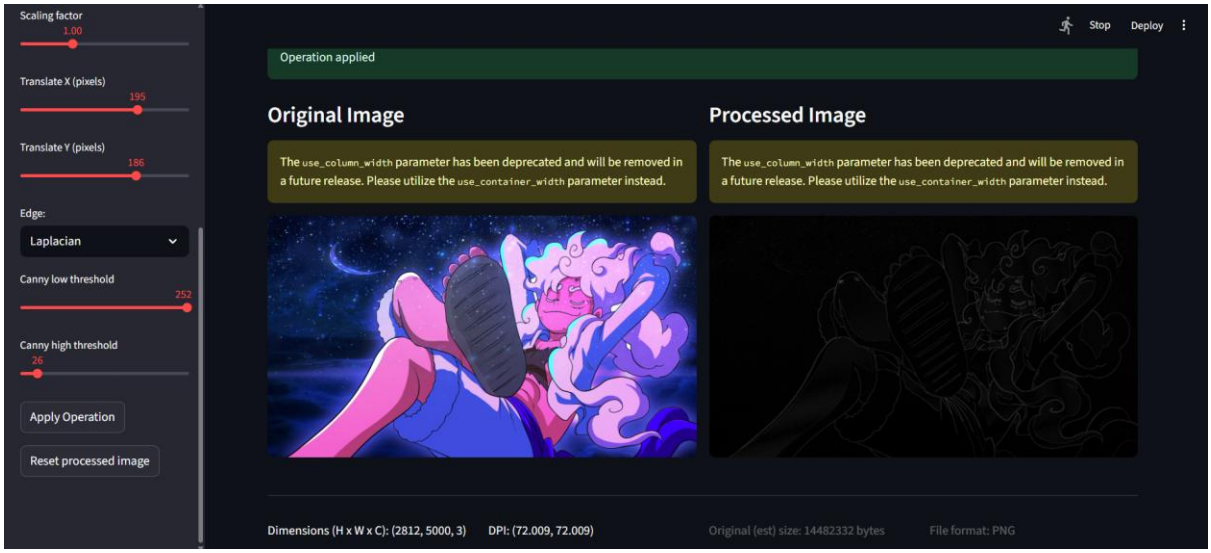

Kernel size (odd)
3
Rotation angle (degrees)
0
Scaling factor
1.00
Translate X (pixels)
0
Translate Y (pixels)
0
Convert:
RGB → Grayscale
Apply Operation
Reset processed image
Stop
Deploy
Operation applied
Original Image
Processed Image
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.


Comparison mode: split original/processed
Show histograms
Kernel size (odd)
5
Rotation angle (degrees)
59
Scaling factor
1.00
Translate X (pixels)
195
Translate Y (pixels)
186
Transform:
Rotate
Apply Operation
Reset processed image
Stop
Deploy
Operation applied
Original Image
Processed Image
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.


Translate X (pixels)
195
Translate Y (pixels)
186
Transform:
Affine Transform
Affine: drag three source->destination points on image preview is not available; we provide sample transforms
Affine: shear/scale factor
1.24
Apply Operation
Reset processed image
Deploy
Operation applied
Original Image
Processed Image
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.


Kernel size (odd)
5
Rotation angle (degrees)
59
Scaling factor
1.00
Translate X (pixels)
195
Translate Y (pixels)
186
Enhancement:
Histogram Equalizati...
Apply Operation
Reset processed image
Stop
Deploy
Operation applied
Original Image
Processed Image
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.


Scaling factor
1.00
Translate X (pixels)
195
Translate Y (pixels)
186
Edge:
Canny
Canny low threshold
50
Canny high threshold
150
Apply Operation
Reset processed image
Stop
Deploy
Image loaded: 5000 x 2812
Operation applied
Original Image
Processed Image
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.


Scaling factor
1.00
Translate X (pixels)
195
Translate Y (pixels)
186
Edge:
Laplacian
Canny low threshold
252
Canny high threshold
26
Apply Operation
Reset processed image
Stop
Deploy
Operation applied
Original Image
Processed Image
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.
The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.
Dimensions (H x W x C): (2812, 5000, 3)
DPI: (72.009, 72.009)
Original (est) size: 14482332 bytes
File format: PNG