

Image Processing Toolkit Report

Student Details

- **Name:** P Harshith Charan
 - **Roll Number:** 23675A7301
 - **Course:** AIML - A
 - **Subject:** Computer Vision
 - **Task - 3:** Image Processing Toolkit Submission
-

Objective

The main objective of this project is to **develop a comprehensive Image Processing Toolkit** using Python and Streamlit, which allows users to upload images and perform a wide range of image processing operations like color conversion, filtering, transformation, enhancement, edge detection, and compression interactively through a web-based GUI.

Technologies and Libraries Used

Library	Purpose
Streamlit	Web app development (GUI)
OpenCV	Image processing and computer vision operations
NumPy	Numerical operations and array handling
PIL (Pillow)	Image conversion, enhancement, and manipulation
Matplotlib	(Optional) Visualization (not used directly)
IO, OS	File handling
Datetime, Time	Timestamps and delay operations

Library	Purpose
Pandas	Not utilized in core processing (can be omitted if unused)

Features Implemented

1. Image Upload & Display

- Supported Formats: JPG, JPEG, PNG, BMP, TIFF
 - Displays **original image** and **processed image** side by side.
 - Uses cv2.imdecode to handle uploaded images as NumPy arrays.
-

2. Color Conversions

- **Supported conversions:**
 - RGB \leftrightarrow BGR
 - RGB \leftrightarrow HSV
 - RGB \leftrightarrow YCbCr
 - RGB \rightarrow Grayscale
 - Uses OpenCV color space conversion functions.
-

3. Geometric Transformations

- **Types of transformations:**
 - **Rotation** (Angle)
 - **Scaling** (X, Y)
 - **Translation** (Shift X, Y)
 - **Affine Transform**
 - **Perspective Transform**
- Uses cv2.getRotationMatrix2D, cv2.resize, cv2.warpAffine, and cv2.warpPerspective.

4. Filtering and Morphological Operations

Filtering:

- Types:

- Gaussian Blur
- Mean Filter
- Median Filter
- Sobel (Edge emphasis)
- Laplacian (Edge emphasis)

Morphological Operations:

- Types:

- Dilation
- Erosion
- Opening
- Closing

Each filter allows **kernel size** selection via a slider.

5. Image Enhancement

- Operations:

- Histogram Equalization
- Contrast Stretching (Autocontrast using percentiles)
- Sharpening (Using ImageEnhance)

Each method improves image clarity or distribution of pixel intensity.

6. Edge Detection

- **Methods:**

- Sobel
- Laplacian
- Canny (Customizable thresholds)

These methods highlight boundaries or sharp transitions in the image.



7. Image Compression & Format Conversion

- Allows saving processed images in:

- JPEG
- PNG
- BMP

Uses PIL for format conversion and saving via BytesIO.



8. Image Information Display

Displays image metadata:

- Width, Height, Channels
 - Image mode (RGB, L, etc.)
 - DPI
 - Estimated file sizes (in KB) for JPEG, PNG, BMP
-



Session Management

Utilizes Streamlit session_state for managing:

- original_image
- processed_image
- image_info

- `image_uploaded`

This ensures the state persists across operations without losing uploaded images.

User Interface Design

- **Responsive layout** with `st.columns`
 - Sidebar for all controls
 - Custom CSS for enhanced look (via `st.markdown`)
 - Sections divided using headers and dividers (`st.markdown("---")`)
-

Output and Export

Users can:

- Preview both original and processed images.
 - Download processed image in multiple formats directly from the interface.
-

Code Structure Summary

Function Name	Purpose
<code>get_image_info()</code>	Extracts metadata from image
<code>update_image_info()</code>	Updates session state with image details
<code>convert_color()</code>	Handles color space conversions
<code>apply_transformation()</code>	Applies geometric transformations
<code>apply_filter()</code>	Applies filtering operations
<code>apply_morphology()</code>	Applies morphological operations
<code>apply_enhancement()</code>	Enhances image quality
<code>apply_edge_detection()</code>	Detects edges using various methods

Function Name	Purpose
save_image()	Prepares image for download
main()	Streamlit app driver function

Conclusion

This **Image Processing Toolkit** is a robust and user-friendly solution for understanding and applying fundamental image processing techniques in real-time through a web interface. It serves as an excellent educational and practical tool for students and professionals learning computer vision concepts.

Suggestions for Improvement (Future Scope)

- Add support for:
 - Batch processing of multiple images
 - Real-time webcam feed processing
 - Implement histogram plotting and pixel intensity analysis
 - Include more advanced operations:
 - CLAHE
 - Noise addition & removal
 - Fourier Transform
 - Use threading or asynchronous processing for large images
-

Screenshots (Optional for Report)

You may include screenshots of:

- App interface
- Sidebar controls
- Before and after image processing exam

DeepSeek Module: Mod Image Process LearnCV.ai/AI Image process ford mustang Download his... Deploy

localhost:8501

Operations

Upload an image

Drag and drop file here
Limit 200MB per file • JPG, JP...

Browse files

cv image 3.jpg 16.6KB

Select Operation Category

Color Conversions

Select Color Conversion

RGB to HSV

Apply Color Conversion

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.



Original Image



Processed Image

Image Information

DeepSeek Module: Mod Image Process LearnCV.ai/AI Image process ford mustang Download his... Deploy

localhost:8501

Operations

Upload an image

Drag and drop file here
Limit 200MB per file • JPG, JP...

Browse files

cv image 3.jpg 16.6KB

Select Operation Category

Transformations

Select Transformation

Affine Transform

Apply Transformation

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.



Original Image



Processed Image

Image Information

localhost:8501

Limit 200MB per file • JPG, JP...

Browse files

cv image 3.jpg 16.6KB

Select Operation Category

Filtering & Morphology

Select Operation Type

Morphology

Select Morphological Operation

Erosion

Kernel Size 5

Apply Morphological Operation

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.

Original Image

Processed Image

Image Information

localhost:8501

Upload an image

Drag and drop file here

Limit 200MB per file • JPG, JP...

Browse files

cv image 3.jpg 16.6KB

Select Operation Category

Enhancement

Select Enhancement

Histogram Equalization

Apply Enhancement

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.

Original Image

Processed Image

Image Information

