

Nama : Hidayatullah Wildan Ghaly B.

NIM : 13521015

Fotosop

Fotosop is a Python application with a graphical user interface (GUI) that allows users to apply various filters to images captured from the camera or loaded from files. The application provides options to adjust contrast, saturation, brightness, grayscale, Gaussian blur, and edge detection filters.

A. Overview

Fotosop is a user-friendly image processing application that provides real-time camera preview and image file processing capabilities. Users can select filters and adjust parameters to modify the appearance of images captured from the camera or loaded from files. The application offers the flexibility to enhance images and experiment with different filters to achieve desired effects.

B. Concept

Advanced Features: NumPy Arrays and Image Processing with OpenCV

In addition to the core functionality of the Interpolation Polynomial script, several advanced features are worth highlighting: the integration of NumPy arrays for efficient parallel processing and the incorporation of OpenCV (cv2) for streamlined image input/output and manipulation.

- NumPy Arrays for Parallel Processing

The script takes full advantage of Numpy, a powerful library for numerical computing in Python. NumPy introduces the concept of arrays, enabling efficient mathematical operations on entire datasets. In the context of the Interpolation Polynomial script, NumPy arrays are employed to perform parallel processing on data, significantly boosting computational performance.

By leveraging the inherent parallelism offered by NumPy arrays, the script can efficiently handle large datasets without compromising on execution speed. This is particularly advantageous when working with extensive data points for polynomial interpolation.

- Image Processing with OpenCV (cv2)

Incorporating [OpenCV](<https://opencv.org/>) (Open Source Computer Vision Library) into the Interpolation Polynomial script adds a layer of versatility, allowing the script to seamlessly process image data. OpenCV is renowned for its comprehensive set of tools designed for image and video analysis, making it an ideal choice for projects involving visual data.

Here's how OpenCV is utilized within the script:

Image Input/Output: OpenCV simplifies reading and writing image files in various formats. It streamlines the process of loading input images and saving output images, enhancing the script's usability.

Image Manipulation: OpenCV provides a plethora of functions for image manipulation, such as resizing, cropping, filtering, and more. These functionalities can be integrated into the script to preprocess input images before interpolation or to post-process the interpolated results.

By harnessing OpenCV's capabilities, the script gains the ability to not only work with numerical data but also process visual information, expanding its potential applications to tasks involving image interpolation and enhancement.

C. Installation

1. **Install Python:** Install Python 3.7 or higher from the official python website. Make sure to add Python to your PATH environment variable during installation.
2. **Clone the Repository:** Clone or download this repository to your local machine
3. **Install Dependencies:** Install the required Python libraries using pip:

```
pip install opencv-python-headless pyqt5
```

4. **Run the Application:** Execute the application by running the following command:

```
cd src  
python main.py
```

D. Usage

1. **Camera Preview:** Click the "Camera" button to activate the camera preview in the left image frame (IMAGE_INPUT_CAMERA). Adjust filters in real-time using the provided options.
2. **Image Input:** Click the "Input Image" button to select an image file from your file system. The selected image will be displayed in the left image frame (IMAGE_INPUT).
3. **Filter Adjustment:** Adjust the filter options on the right side to modify the appearance of the input image or camera preview.
4. **Recalculate:** Click the "Recalculate" button to reapply filters to the input image (non-camera mode) and display the result in the right image frame (IMAGE_OUTPUT).

E. Input Output Example









