Interactive Website to Create Bokeh Effect

NITIN REDDY K - CS21B2028

1. AIM

The aim of this project is to develop an interactive website that allows users to create a bokeh effect on their uploaded images.

II. APPARATUS/SOFTWARE REQUIRED

# Hardware:

* A computer or server to host the website

# Software:

* Python 3.x
* Flask (Python web framework)
* OpenCV (Open Source Computer Vision Library)
* NumPy (Numerical Python Library)
* Matplotlib (Plotting Library for Python)

III. PROCEDURE

1. *Introduction*

The bokeh effect is a popular photographic technique that creates an aesthetic blur around the subject, making it stand out from the background. This project aims to provide a user-friendly web interface where users can upload their images, and the website will automatically apply the bokeh effect to the uploaded image.

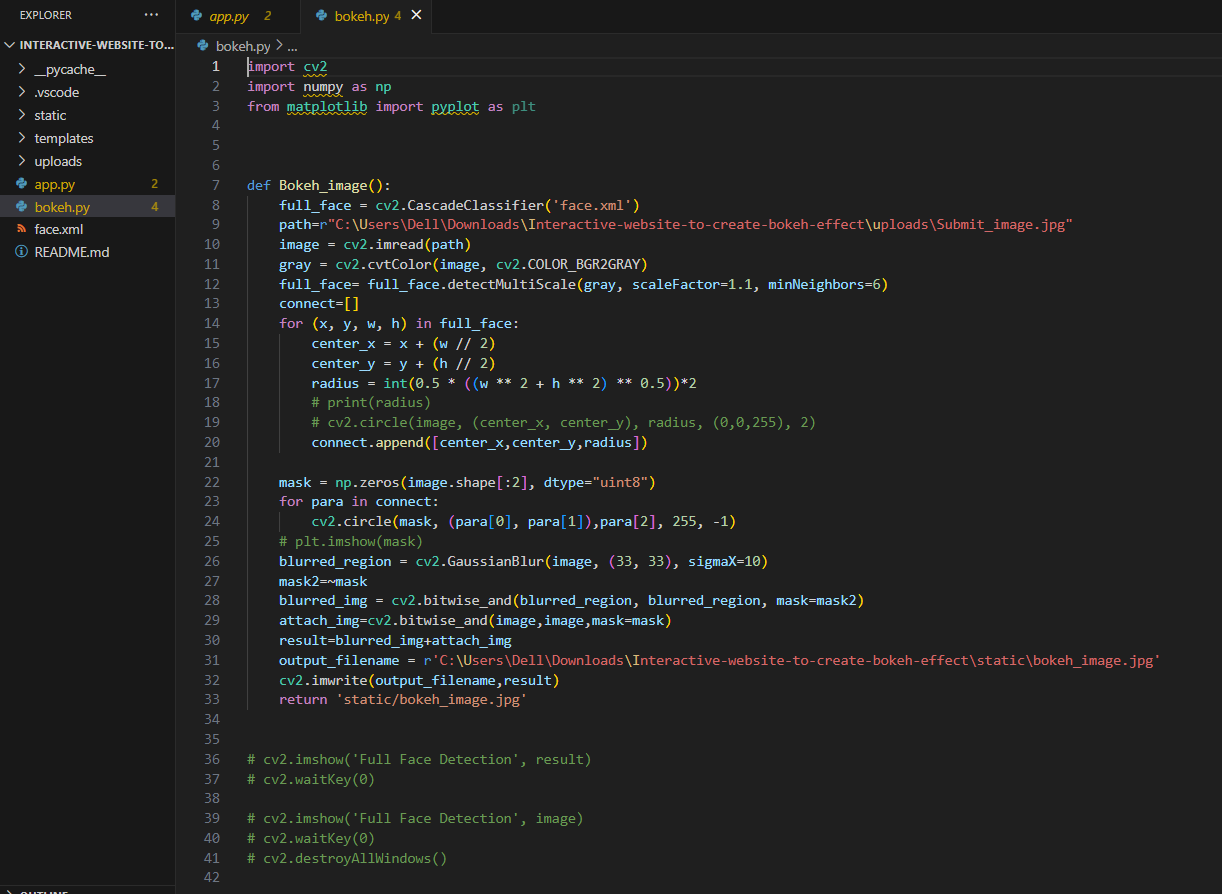
1. *Methodology/Problem Statement*

The project involves the following steps:

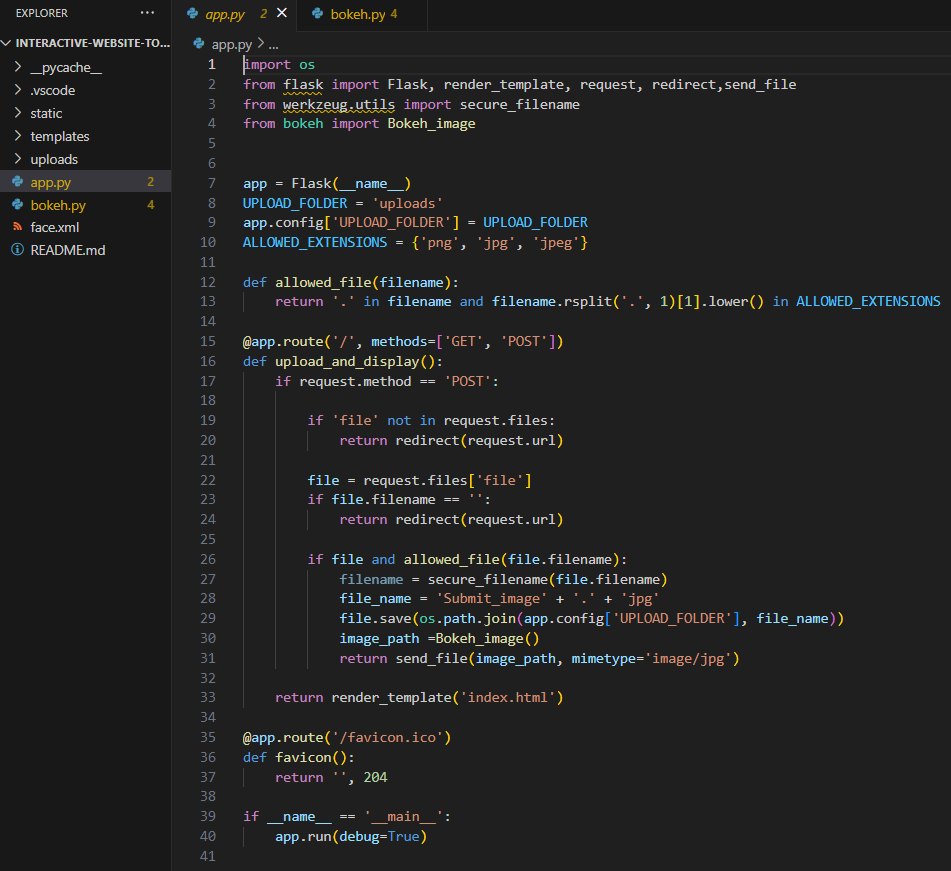
* 1. Develop a web interface using Flask to allow users to upload their images.
  2. Implement an image processing algorithm using OpenCV to detect faces in the uploaded image.
  3. Apply the bokeh effect to the uploaded image by blurring the background while keeping the detected faces sharp.
  4. Display the processed image with the bokeh effect on the website.

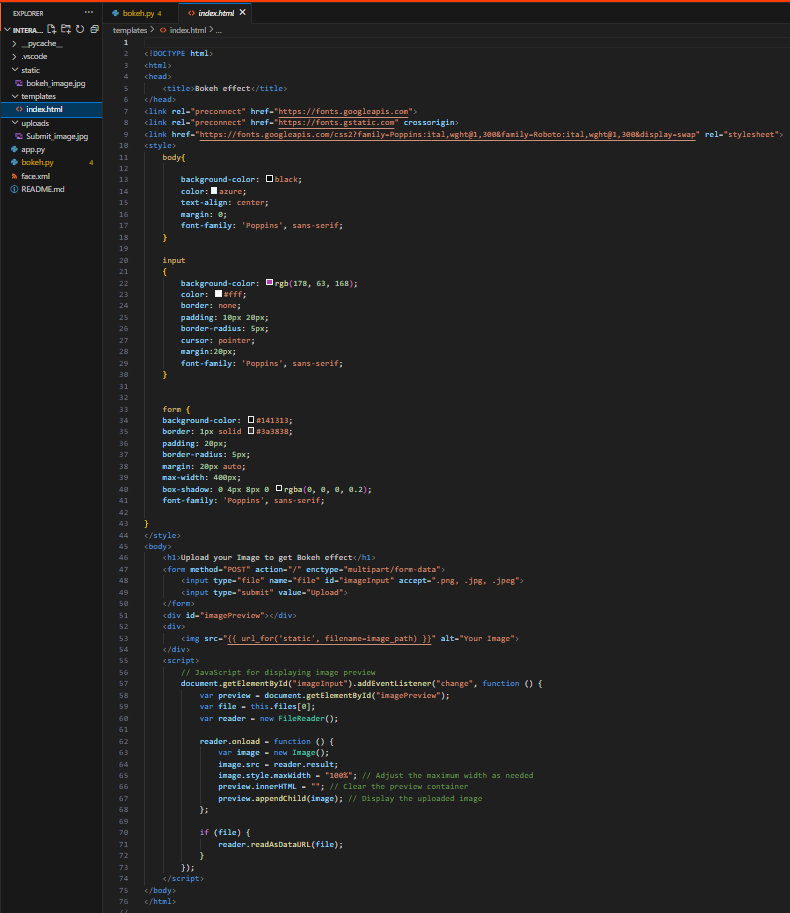
1. *Implementation*
2. *Executable Code:*

* The bokeh.py file contains the code for applying the bokeh effect to the uploaded image. It uses OpenCV’s cascade classifier to detect faces, creates a mask for the detected faces, and blurs the background while keeping the faces sharp.



* The app.py file contains the Flask application code. It handles the user’s image upload, saves the uploaded image, calls the Bokeh\_image() function from bokeh.py, and displays the processed image with the bokeh effect on the website.





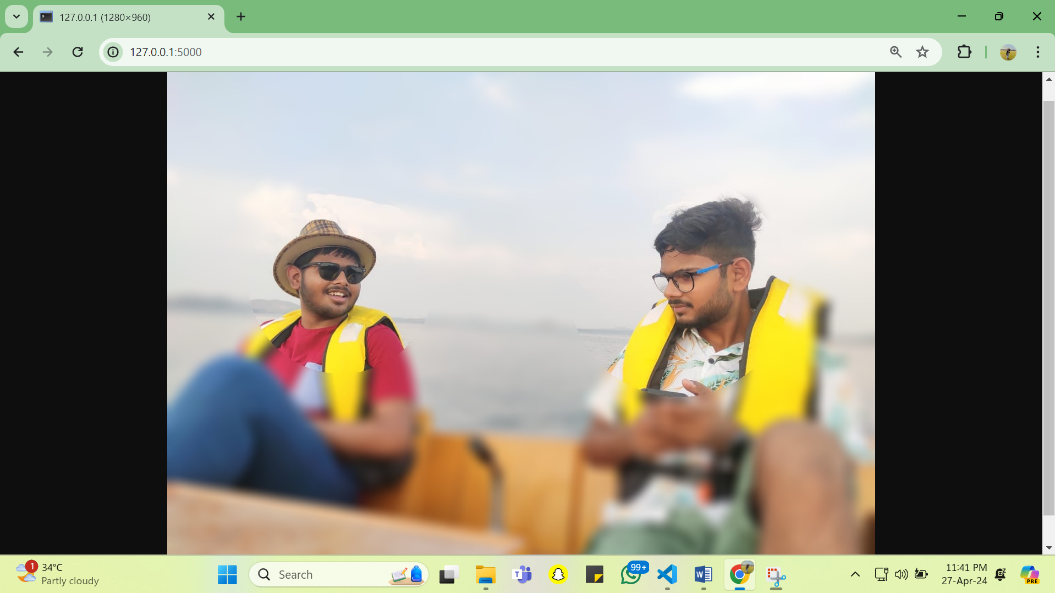
1. *Software Implementation:* The project is implemented using Flask as the web framework and OpenCV for image processing. Flask handles the user interface, file uploads, and rendering the processed image. OpenCV is used for face detection and applying the bokeh effect to the uploaded image.
2. RESULTS/OUTPUT

The project successfully creates an interactive website where users can upload their images, and the website will automatically apply the bokeh effect to the uploaded image. The processed image with the bokeh effect is displayed on the website for the user to view or download.

Before Effect:



After Effect:



1. CONCLUSION AND FUTURE DIRECTIONS

This project demonstrates the integration of web development, computer vision, and image processing techniques to create an interactive website for applying the bokeh effect to user-uploaded images.

*Future Directions:*

* Enhance the face detection algorithm to handle multiple faces or improved accuracy.
* Provide additional options for customizing the bokeh effect, such as adjusting the blur intensity or shape.
* Implement real-time image processing for live video feeds or webcam input.
* Improve the user interface and overall user experience of the website.

1. REFERENCES

* Flask Documentation: https://flask.palletsprojects.com/
* OpenCV Documentation: https://docs.opencv.org/
* NumPy Documentation: https://numpy.org/doc/
* Matplotlib Documentation: https://matplotlib.org/stable/contents.html