Computer Vision ELC

Himanshu Sardana 102303244

Contents

INTRODUCTION

PROBLEM STATEMENT

In this project, you can build an application to upload the image on the app. Then by performing different transformations on the image we can make the image look like a cartoon

TOOLS AND TECHNOLOGIES USED

1. **Python:** Chosen for its simplicity, wide community support, and powerful ecosystem for image processing and scripting.
2. **OpenCV:** A powerful open-source computer vision and image processing library. Used for tasks like grayscale conversion, edge detection, and bilateral filtering to produce the cartoon effect.
3. **Argparse:** A standard Python library used to parse command-line arguments. It makes the script more flexible and user-friendly by allowing users to specify input/output paths and optional filter settings.


Development Environment:
1. **OS:** Arch Linux
2. **Editor:** Neovim
3. **Python Environment:** Python 3.11

USAGE INSTRUCTIONS

1. Installing Dependencies
   ```
   pip3 install -r requirements.txt
   ```
2. Running the script
   ```
   python3 main.py --input <path_to_input_image> --output <path_to_output_image>
   ```
3. Example
   ```
   python3 main.py --input test.jpg --output output.jpg
   ```

METHODOLOGY

Original Image

↓

Grayscale Conversion

↓

Noise Reduction

↓

Edge Detection

↓

Adaptive Thresholding

↓

Bilateral Filtering

↓
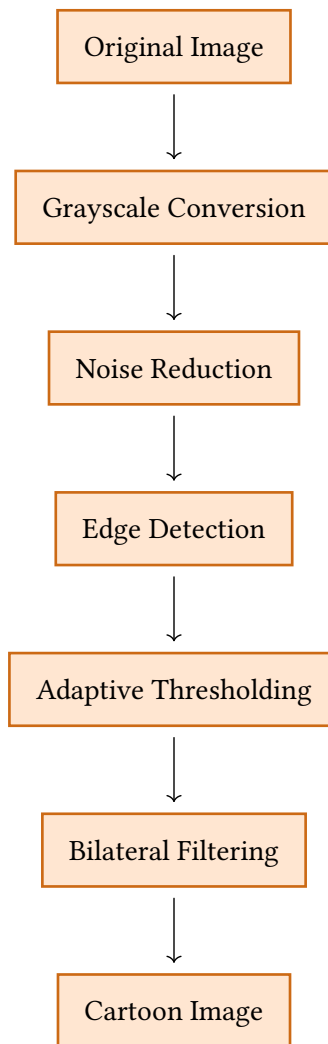
Cartoon Image

EXPLANATION

The methodology involves a series of image processing steps to transform the original image into a cartoon-like effect. Each step is crucial for achieving the final result, and the process is designed to be efficient and effective.

1. **Grayscale Conversion:** The first step converts the original image to grayscale, simplifying the color information and preparing it for further processing.
2. **Noise Reduction:** A bilateral filter is applied to reduce noise while preserving edges, which is essential for maintaining the cartoon effect.
3. **Edge Detection:** The Canny edge detection algorithm is used to identify edges in the image, which are crucial for defining the cartoon's outlines.
4. **Adaptive Thresholding:** This step creates a binary image that highlights the edges, enhancing the cartoon effect.
5. **Bilateral Filtering:** Finally, a bilateral filter is applied to smooth the image while keeping the edges sharp, resulting in a cartoon-like appearance.
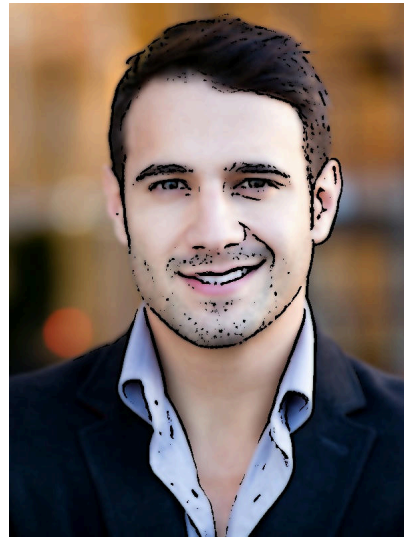
Original Image

Cartoon Image

The result of the project is a cartoon-like image that retains the essential features of the original while applying a stylized effect. The transformation process effectively simplifies the image, making it visually appealing and suitable for various applications, such as social media filters or artistic effects.