Syed Azhar Hussain Quadri

CS 5330 PRCV

Project 1 Report

Real-Time Filtering

What is Real-time Filtering:

Real-time filtering refers to the process of applying an image or video filter to live, streaming data in order to change or enhance its appearance. This can include a wide variety of filters, such as color filters, edge detection filters, blurring filters, and others. Real-time filtering is used in many applications such as video conferencing, live streaming, security cameras, and more. The goal of real-time filtering is to provide an immediate response, displaying the filtered video or image on the fly, in real time, as the video or image is being captured or streamed. This requires powerful hardware, efficient algorithms, and low latency in order to process the live data quickly enough to keep up with the stream.

About the Project:

This project involved various challenges related to real-time filtering of video and images, such as: displaying an image from a file, displaying live video on the screen with options to quit, save an image, display the video image in greyscale using openCV functions and manually altering color channels, applying filters like 5x5 Gaussian Filter as 1x5 separable filters, 3x3 sobel X and Y filters as 1x3 separable filters, creating gradient magnitude images, blurring and quantizing an image, making a cartoonization function and applying Scharr gradient filter, Laplacian gradient filter, and threshold filter using openCV filter functions. The project involved working with images, using openCV functions, creating filters to manually change color channels, and applying them to images.

Original Image

GreyScale image:

Alternate Grayscale image:

Blur Image

Sobel X:

Sobel Y:

Gradient Magnitude:

Blur Quantize:

Cartoonish:

10) Other effects --- a)Gradient Grainy Filter:

b) Laplacian gradient filter:

c) Threshold:

Ghost Filter:

Video: Ghost filter effect.mp4

Reflections:

The most challenging part for me was the setup of the visual studio. I always felt it was overrated and used VS Code instead. Initially, I started this project on VS code and used Cmake and other libraries to make this work, but it wasn't consistent. I had conducted an OpenCV workshop using Python in my undergrad and that knowledge gave me a head start. It was challenging at times to implement new and learn about live filters. Also, thanks to my cat for participating in this experiment, but he clearly didn't like the process and bit me by the end of the shoot.

References/ Sources

OpenCV documentation

Many YouTube Tutorials for setting up the environment

Online codes to learn about filter and mixing and matching them to create new ones.