Experiment Project Documentation

Introduction

This document captures the technical details related to the experiment development.

Project

Domain Name : Computer Science & Engineering

Lab Name : Image Processing

Experiment Name : Image Arithmetic

The Image Arithmetic virtual lab is the JavaScript version of the experiment written originally in PHP. It is used to demonstrate how various arithmetic operations between two images produce different results according to the operations performed.

Purpose of the project

The purpose of the project is to convert the **Image Arithmetic** experiment simulation from **PHP** to **JavaScript**.

Project Developers Details

S.NO	Names	Year of Study	Role	Email-ID	github handles
1.	Shoaib Areeb	2019	Developer	Sareeb50@gmail.com	@sareeb50

Technologies and Libraries

Technologies :

1. HTML

2. CSS
3. Javascript

Libraries :

1. JQuery.js

Development Environment

OS : Windows
Bandwidth: 100Mbps

Documents :

S.NO	Link to Document	Role
1.	Procedure	This document captures the instructions to run the simulations
2.	Test Cases	This document captures the functional test cases of the experiment simulation
3.	Code Documentation	This document captures the details related to code

Process Followed to convert the experiment

1. Understand the assigned experiment Java simulation
2. Understanding the experiment concept
3. Re-implement the same in javascript

Value Added by our Project

1. It would be beneficial for engineering students
2. Highly beneficial for tier 2 and tier 3 college students who can use this to learn and understand the concept of perception learning.

Risks and Challenges

The main Challenge in this experiment was to access the pixel data of the images and that was successfully done using the HTML canvas property.

Digital images are stored as b bit images. Hence, the range of values a pixel can take is restricted to the range [0, 1,... (2b-1)]. With b= 8 this range is [0,1,...255]. The closed interval poses a problem when performing arithmetic operations in practice, as the results are not guaranteed to be within this interval. For an 8-bit image the intervals for the output pixel for each operation are:

Addition: $r \in [0,(2x255=510)]$

Subtraction: $r \in [-255,255]$

Difference: $r \in [0,255]$

Multiplication: $r \in [0,(2552 =65025)]$

Division: $r \in [0,\infty]$

Since we need r to be in [0,255], we will have an underflow or overflow. A final processing step is generally required to handle this problem.

Issues :

The experiment runs best in a desktop environment.

Mobile responsiveness has not been considered.