ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №2

За курсом «Обробка та розпізнавання зображень»

Студента групи ПА-19-2

Ільяшенко Єгора Віталійовича

Кафедра комп’ютерних технологій, ДНУ

**Точкові методи обробки зображень.**

Постановка задачі

Table

Description automatically generated

Diagram

Description automatically generated

Варіант №7

Завдання 1





A picture containing text, water, boat, outdoor

Description automatically generatedГамма-корекція - це процес зміни яскравості зображення шляхом застосування функції гамма до значень яскравості пікселів. Функція гамма визначається як степенева функція, де значення яскравості пікселів підносяться до потужності, що залежить від параметру гамма.

Зазвичай, значення гамма приймає значення в діапазоні від 1.8 до 2.2. Це означає, що яскравості зображення збільшуються в більшій мірі для темних кольорів, ніж для світлих кольорів. Це відповідає сприйняттю кольорів око людини, яке є неоднаковим в залежності від яскравості.

Гамма-корекцію можна застосовувати на різних етапах обробки зображень, наприклад, при збереженні зображень у форматі JPEG, при калібруванні моніторів, при друку зображень тощо. Застосування гамма-корекції дозволяє забезпечити більш точне відтворення кольорів на різних пристроях та знизити ризик спотворення кольорів на пристроях з відмінними характеристиками яскравості.

Загальна формула для гамма-корекції:



Де - коефіцієнт гамма-корекції,

А = 1, якщо діапазон кольорів [0;1]

А = 255, якщо діапазон [0;255]

Chart

Description automatically generated

Вирівнювання висвітленого зображення. Gamma = 5

A picture containing text, outdoor, light, snow

Description automatically generated



Вирівнювання затемненого зображення. Gamma = 0,25

A picture containing text, dark, crowd

Description automatically generated



Graphical user interface

Description automatically generatedGraphical user interface

Description automatically generated

Реалізований метод допомагає вирівняти зображення, зображення стає більш збалансованим, але не використовує весь спектр доступних відтінків.

Завдання 2



Chart, shape, histogram, rectangle

Description automatically generated

Ми ділимо відрізок яскравості на 3 ділянки

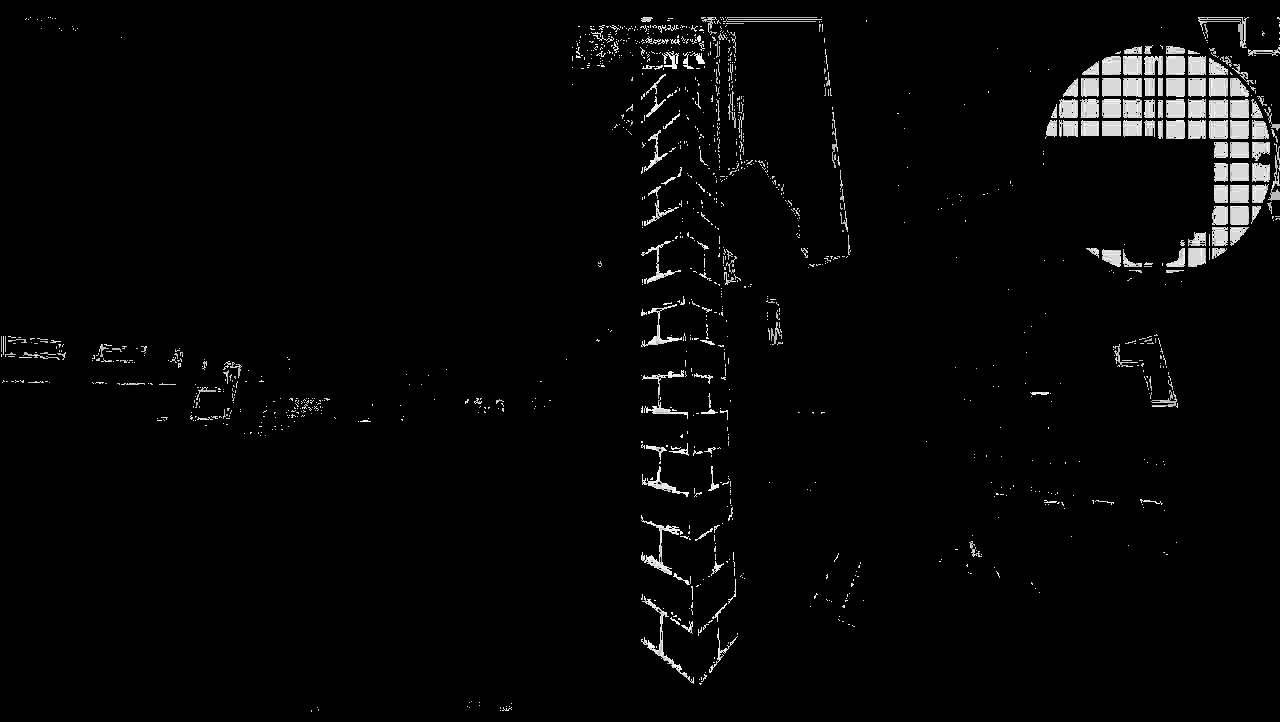
Якщо колір знаходиться у діапазоні [0;a] або [a;1], ми обнуляємо його значення. Якщо ж значення пікселя знаходиться поміж цих відрізків, ми вираховуємо вихідне значення з формули

Graphical user interface

Description automatically generated

Значення параметрів a, b ми задаємо у програмі. Х – вхідне значення, Y – вихідне значення.

Graphical user interface, application

Description automatically generated

Graphical user interface, PowerPoint

Description automatically generated with medium confidence

Висновки

Виконуючи лабораторну роботу №2, я розібрався у темі точкової обробки зображень, розробив програму на мові програмування c#, яка будує гістограму зображення, виконує гамма корекцію зображення, та препарує його згідно з варіанту. Усі типи виводу програми можна завантажити у файл.

Код програми

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Globalization;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Lab2

{

public partial class Form1 : Form

{

string[] args;

Graphics gr;

Bitmap inputBmp;

Bitmap outBmp;

Bitmap task2Bmp;

public Form1()

{

InitializeComponent();

gr = pictureBox1.CreateGraphics();

gr.Clear(Color.Yellow);

pictureBox1.Refresh();

args = Environment.GetCommandLineArgs();

}

void LoadImage(string path)

{

Image inputImage = Bitmap.FromFile(path);

gr.DrawImage(inputImage, pictureBox1.DisplayRectangle);

inputBmp = new Bitmap(inputImage);

int[] gradesCount = new int[256];

for (int i = 0; i < inputBmp.Width; ++i)

for (int j = 0; j < inputBmp.Height; ++j)

++gradesCount[inputBmp.GetPixel(i, j).R];

chart1.Series[0].Points.Clear();

for (int i = 0; i < 256; ++i)

chart1.Series[0].Points.AddXY(i, gradesCount[i]);

}

private void button1\_Click(object sender, EventArgs e)

{

if (openFileDialog1.ShowDialog() != DialogResult.OK)

return;

if (openFileDialog1.FileName.Substring(openFileDialog1.FileName.Length - 4) != ".bmp")

return;

LoadImage(openFileDialog1.FileName);

}

private void button2\_Click(object sender, EventArgs e)

{

if (inputBmp == null)

return;

double trackBarValue = Convert.ToDouble(textBox1.Text);

outBmp = new Bitmap(inputBmp.Width, inputBmp.Height);

double gammaCorrection = trackBarValue;

int chanel;

int[] gradesCount = new int[256];

for (int i = 0; i < outBmp.Width; ++i)

for (int j = 0; j < outBmp.Height; ++j)

{

chanel = (int)(255 \* Math.Pow(inputBmp.GetPixel(i, j).R / 255.0, gammaCorrection));

if (chanel > 255)

chanel = 255;

++gradesCount[chanel];

outBmp.SetPixel(i, j, Color.FromArgb(chanel, chanel, chanel));

}

chart2.Series[0].Points.Clear();

for (int i = 0; i < 256; ++i)

chart2.Series[0].Points.AddXY(i, gradesCount[i]);

Graphics outGr = pictureBox2.CreateGraphics();

outGr.DrawImage(outBmp, pictureBox2.DisplayRectangle);

}

private void button3\_Click(object sender, EventArgs e) //Save

{

if (outBmp == null)

return;

outBmp.Save($"{args[0].Substring(0, args[0].LastIndexOf('\\'))}\\{textBox2.Text}.bmp");

}

int Task2Function(int input, int lowerBound, int upperBound)

{

if (input <= lowerBound || input >= upperBound)

return 0;

return 256\*(input - lowerBound) / (upperBound - lowerBound);

}

private void button4\_Click(object sender, EventArgs e)

{

if (inputBmp == null)

return;

task2Bmp = new Bitmap(inputBmp);

chart3.Series[0].Points.Clear();

int[] gradesCount = new int[256];

int lowerBound = int.Parse(textBox4.Text);

int upperBount = int.Parse(textBox5.Text);

for (int i = 0; i < task2Bmp.Width; ++i)

for (int j = 0; j < task2Bmp.Height; ++j)

{

int val = Task2Function(task2Bmp.GetPixel(i, j).R, lowerBound, upperBount);

task2Bmp.SetPixel(i, j, Color.FromArgb(val, val, val));

++gradesCount[val];

}

for (int i = 1; i < 256; ++i)

chart3.Series[0].Points.AddXY(i, gradesCount[i]);

Graphics task2Gr = pictureBox3.CreateGraphics();

task2Gr.DrawImage(task2Bmp, pictureBox3.DisplayRectangle);

}

private void button5\_Click(object sender, EventArgs e)

{

if (task2Bmp == null)

return;

task2Bmp.Save($"{args[0].Substring(0, args[0].LastIndexOf('\\'))}\\{textBox3.Text}.bmp");

}

}

}