Laporan Praktikum Pengolahan Citra

Histogram Ekualisasi

Dimas Rizky H.P. – 2110141011 – 3 D4 IT A

1. Kode program

* Button Load

Button load berfungsi untuk membuka file citra dan menhyimpan datanya pada sebuah variabel untuk nanti diproses lebih lanjut menggunakan fitur citra lainnya.

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

{

DialogResult d = openFileDialog1.ShowDialog();

if (d == DialogResult.OK)

{

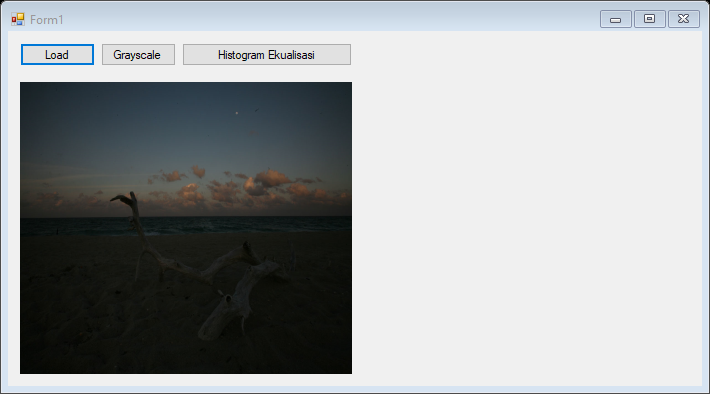
objBitmap = new Bitmap(openFileDialog1.FileName);

pictureBox1.Image = objBitmap;

}

}

*Box Code 1 – Button Load*

**

*Gambar 1.0, load file citra menggunakan button load*

* Button Grayscale

Button grayscale digunakan untuk mengubah file citra yang sudah dibuka menjadi citra grayscale/derajat keabuan

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

{

for (int x = 0; x < objBitmap.Width; x++)

for (int y = 0; y < objBitmap.Height; y++)

{

Color w = objBitmap.GetPixel(x, y);

int r = w.R;

int g = w.G;

int b = w.B;

int xg = (int)((r + g + b) / 3);

Color wb = Color.FromArgb(xg, xg, xg);

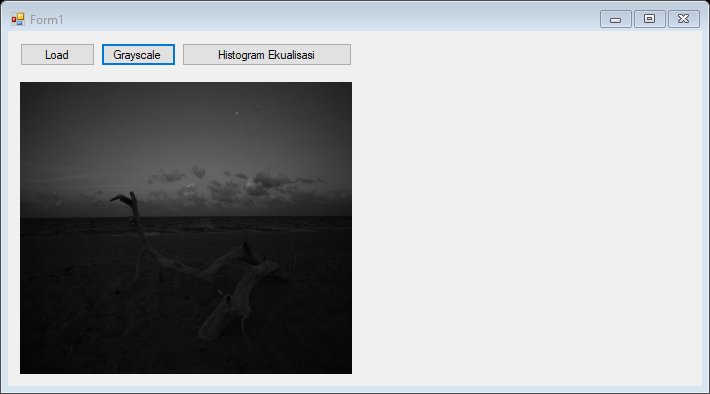
objBitmap.SetPixel(x, y, wb);

}

pictureBox1.Image = objBitmap;

}

*Box Code 2 – Button Grayscale*

**

*Gambar 1.1, konversi gambar citra original ke grayscale*

* Button Histogram Ekualisasi

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

{

float[] h = new float[256];

float[] c = new float[256];

int i;

for (i = 0; i < 256; i++) h[i] = 0;

for (int x = 0; x < objBitmap.Width; x++)

for (int y = 0; y < objBitmap.Height; y++)

{

Color w = objBitmap.GetPixel(x, y);

int xg = w.R;

h[xg] = h[xg] + 1;

}

c[0] = h[0];

for (i = 1; i < 256; i++) c[i] = c[i - 1] + h[i];

int nx = objBitmap.Width;

int ny = objBitmap.Height;

for (int x = 0; x < objBitmap.Width; x++)

for (int y = 0; y < objBitmap.Height; y++)

{

Color w = objBitmap.GetPixel(x, y);

int xg = w.R;

int xb = (int)(255 \* c[xg] / nx / ny);

Color wb = Color.FromArgb(xb, xb, xb);

objBitmap.SetPixel(x, y, wb);

}

pictureBox2.Image = objBitmap;

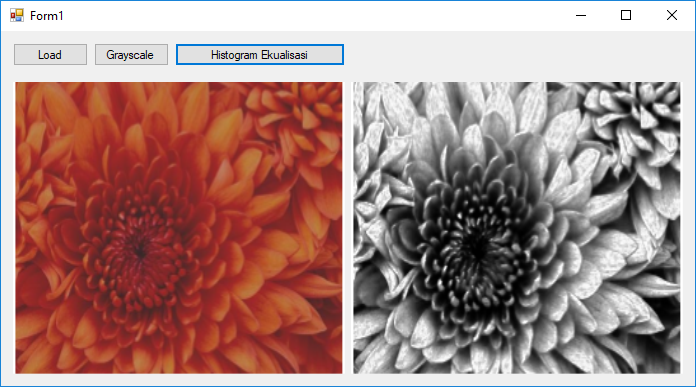
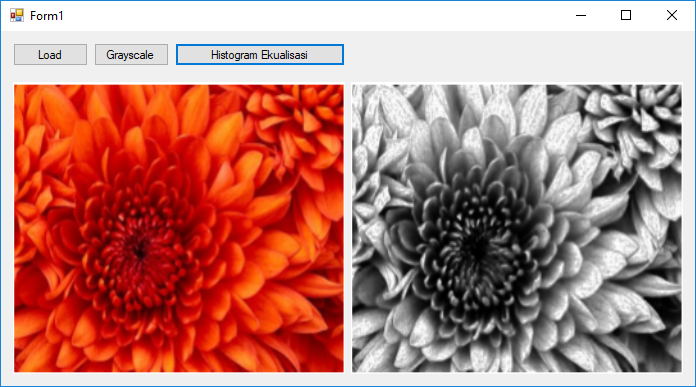
}

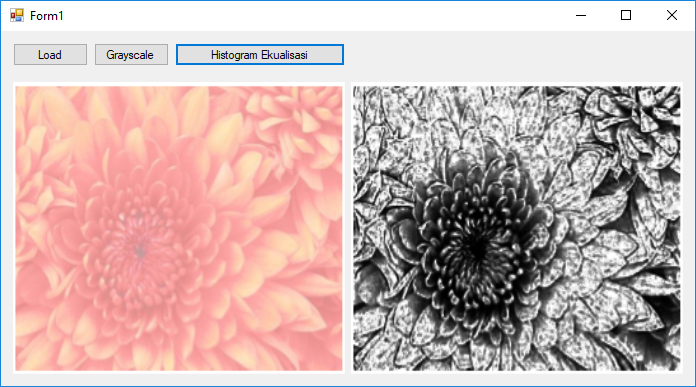
*Box Code 3 – Button Histogram Ekualisasi*

Histogram ekualisasi adalah sebuah proses untuk memperbaiki terang/gelap nya gambar pada derajat keabuan dengan memangaatkan distribusi nilai-nilai derajat keabuan dari gambar tersebut.

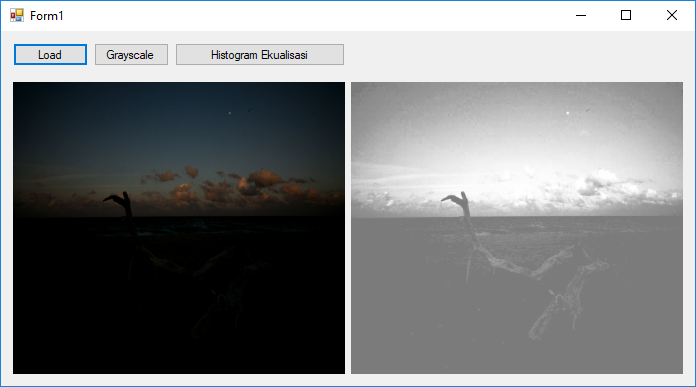
Histogram ekualisasi mengubah distribusi kemunculuna setiap nilai derajat keabaun ke arah seragam, artinya setiap nilai derajat keabuan memiliki kemungkinan yang sama baik dari gelap, sedang maupun terang

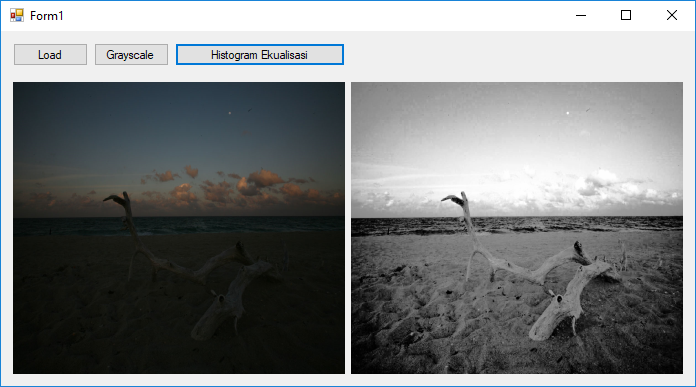
Hasil dari histogram ekualisasi adalah membuat distribusi derajat kebauan gambar menjadi seragam dan gambar menjadi tampak lebih jelas.

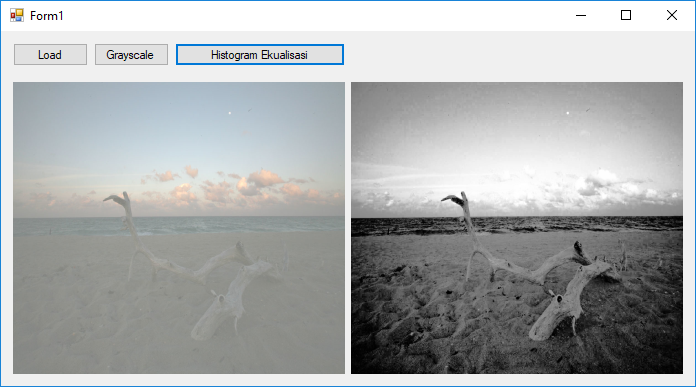




*Gambar 1.2, Contoh penerapan histogram ekualisasi pada gambar tulip dengan 3 tingkat brightness yang berbeda*







*Gambar 1.3, Histogram ekualisasi pada citra gelap hingga terang*

1. Apa perbedaan hasil dari histogram ekualisasi dan auto level ?

Auto-level bekerja dengan menyama ratakan tingkat keabuan per pixelnya, untuk pixel terang diubah menjadi 255, sehingga pixel gelap diubah menjadi 0, untuk seluruh pixel yang ada. Sedangkan untuk histogram ekualisasi, yang dilakukan adalah meratakan distribusi derajat keabuan yang ada, sehingga tidak hanya bekerja dengan per pixel, namun bekerja dengan seluruh pixel yang ada. Maksudnya adalah derajat keabuan tidak semata mata di konversikan ke sebuah nilai, namun dihitung berdasarkan pixel pixel yang lainnya.