

LEMBAR JAWAB LATIHAN SOAL

NIM : 71220917

NAMA : anggrayni layuk

NILAI :

Latihan Pengamatan

1. Tuliskan kode untuk dapat melakukan Flip Horizontal secara manual tanpa bantuan fungsi built-in dari tipe data Image!

Proses flip horizontal dimulai dengan mengambil warna tiap piksel dari gambar asli menggunakan `imageOri.GetPixel(x, y)` untuk membuat salinannya. Kemudian, setiap piksel pada kolom (x) dipindahkan ke posisi yang berlawanan secara horizontal, yaitu pada posisi `imageOri.Width - x - 1` di baris yang sama (y), sehingga gambar tampak terbalik dari kiri ke kanan. Setelah proses pembalikan selesai, gambar hasilnya disimpan dalam variabel `imageFlip` dan ditampilkan ke layar melalui `PictureBox1.Image = imageFlip`.

```
0 references
Private Sub FlipHorizontalToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles FlipHorizontalToolStripMenuItem.Click
    Dim imageOri As Bitmap = CType(PictureBox1.Image, Bitmap)
    Dim imageFlip As New Bitmap(imageOri.Width, imageOri.Height)

    For y As Integer = 0 To imageOri.Height - 1
        For x As Integer = 0 To imageOri.Width - 1
            Dim originalPixel As Color = imageOri.GetPixel(x, y)
            imageFlip.SetPixel(imageOri.Width - x - 1, y, originalPixel)
        Next
    Next

    PictureBox1.Image = imageFlip
End Sub
```

2. Cobalah untuk menambahkan error handling agar pengolahan tidak dapat dijalankan jika belum ada citra yang dibuka. Tuliskan contoh kodenya yang Anda taruh di salah satu fungsi saja!

```

0 references
Private Sub GreyscaleToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles GreyscaleToolStripMenuItem.Click
    If PictureBox1.Image IsNot Nothing Then
        Dim r, g, b, gray As Integer
        Dim bmp = New Bitmap(PictureBox1.Image)
        For bar As Integer = 0 To PictureBox1.Image.Height - 1
            For kol As Integer = 0 To PictureBox1.Image.Width - 1
                r = bmp.GetPixel(kol, bar).R
                g = bmp.GetPixel(kol, bar).G
                b = bmp.GetPixel(kol, bar).B
                gray = Math.Round(0.2126 * r + 0.7152 * g + 0.0722 * b)
                bmp.SetPixel(kol, bar, Color.FromArgb(gray, gray, gray))
            Next
        Next
        'Dim img As Image
        'img = CType(bmp, Image)
        'PictureBox1.Image = img
        PictureBox1.Image = bmp
    Else
        MessageBox.Show("Belum ada image", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error)
    End If
End Sub

```

3. Cobalah menerapkan matriks kernel lain yang ukurannya sama (3x3). Amati perubahan yang terjadi terhadap citra hasil pengolahannya!

- a. Tajamkan Ringan

```

0 references
Private Sub TajamkanRinganToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles TajamkanRinganToolStripMenuItem.Click
    Dim r, g, b As Integer
    Dim bmp = New Bitmap(PictureBox1.Image)
    Dim kernel As Integer() = {-1, -1, -1, -1, 9, -1, -1, -1, -1}
    For bar As Integer = 1 To PictureBox1.Image.Height - 2
        For kol As Integer = 1 To PictureBox1.Image.Width - 2
            r = 0
            g = 0
            b = 0
            For i As Integer = 0 To 8
                r = r + (kernel(i) * bmp.GetPixel(kol - 1 + (i Mod 3), bar - 1 + (i \ 3)).R)
                g = g + (kernel(i) * bmp.GetPixel(kol - 1 + (i Mod 3), bar - 1 + (i \ 3)).G)
                b = b + (kernel(i) * bmp.GetPixel(kol - 1 + (i Mod 3), bar - 1 + (i \ 3)).B)
            Next
            r = Math.Floor(r / 3)
            g = Math.Floor(g / 3)
            b = Math.Floor(b / 3)
            If r < 0 Then r = 0
            If g < 0 Then g = 0
            If b < 0 Then b = 0
            If r > 255 Then r = 255
            If g > 255 Then g = 255
            If b > 255 Then b = 255
            bmp.SetPixel(kol, bar, Color.FromArgb(r, g, b))
        Next
    Next
    PictureBox1.Image = bmp
End Sub

```

Pada kode di atas menggunakan kernel {-1, -1, -1, -1, 9, -1, -1, -1, -1} menghasilkan efek penajaman yang lebih ringan dibandingkan penajaman yang biasa.

- b. Gaussian Blur

```

0 references
Private Sub GaussianBlurToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles GaussianBlurToolStripMenuItem.Click
    Dim r, g, b As Integer
    Dim bmp = New Bitmap(PictureBox1.Image)
    Dim kernel As Integer() = {1, 2, 1, 2, 4, 2, 1, 2, 1}
    For bar As Integer = 1 To PictureBox1.Image.Height - 2
        For kol As Integer = 1 To PictureBox1.Image.Width - 2
            r = 0
            g = 0
            b = 0
            For i As Integer = 0 To 8
                r = r + (kernel(i) * bmp.GetPixel(kol - 1 + (i Mod 3), bar - 1 + (i \ 3)).R)
                g = g + (kernel(i) * bmp.GetPixel(kol - 1 + (i Mod 3), bar - 1 + (i \ 3)).G)
                b = b + (kernel(i) * bmp.GetPixel(kol - 1 + (i Mod 3), bar - 1 + (i \ 3)).B)
            Next
            r = Math.Floor(r / 16)
            g = Math.Floor(g / 16)
            b = Math.Floor(b / 16)
            If r < 0 Then r = 0
            If g < 0 Then g = 0
            If b < 0 Then b = 0
            If r > 255 Then r = 255
            If g > 255 Then g = 255
            If b > 255 Then b = 255
            bmp.SetPixel(kol, bar, Color.FromArgb(r, g, b))
        Next
    Next
    PictureBox1.Image = bmp
End Sub

```

Pada kode di atas menggunakan kernel {1, 2, 1, 2, 4, 2, 1, 2, 1} dan dibagi 16 pada setiap R, G, B sehingga menghasilkan efek gaussian blur.

4. Buka sebuah citra, kemudian lihat histogramnya. Lakukan pengurangan kontras sebanyak 8 kali pada citra tersebut, dan amati kembali histogramnya. Apa perbedaannya?

Citra hasil menjadi lebih buram dan warnanya pudar dibandingkan citra awal.

Histogram menjadi lebih sempit dan rentang nilai intensitas piksel dalam gambar lebih kecil dibandingkan dengan yang sebelumnya. Pada nilai piksel tertentu terdapat peningkatan frekuensi karena rentang nilai piksel menjadi lebih sempit.