**TUGAS PENGOLAHAN CITRA**

Nama : Adraianus Wisnu Pratama

NIM : 123190148

Kelas : C

|  |
| --- |
| GAMBAR ASLI |
| C:\Users\Wisnu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\gambar1.jpg |
| C:\Users\Wisnu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\gambar2.jpg |
| C:\Users\Wisnu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\gambar3.jpg |
|  |
|  |

1. CITRA PENCERAHAN

|  |  |
| --- | --- |
| from PIL import Image  def clipping(intensitas):  if intensitas < 0:  return 0  if intensitas > 0:  return 255  return intensitas  def citrapencerahan(gambar, nilaicerah, after\_save):  CITRA\_PENCERAHAN = Image.open(gambar)  PIXEL = CITRA\_PENCERAHAN.load()  ukuran\_horizontal = CITRA\_PENCERAHAN.size[0]  ukuran\_vertika = CITRA\_PENCERAHAN.size[1]  for x in range(ukuran\_horizontal):  for y in range(ukuran\_vertika):  R = clipping(PIXEL[x,y][0] + nilaicerah)  G = clipping(PIXEL[x,y][1] + nilaicerah)  B = clipping(PIXEL[x,y][2] + nilaicerah)  PIXEL[x,y] = (R, G, B)  CITRA\_PENCERAHAN.save(after\_save)  citrapencerahan('gambar1.jpg', 10, 'gambar1\_dicerahkan.jpg')  citrapencerahan('gambar1.jpg', -170, 'gambar1\_digelapkan.jpg')  citrapencerahan('gambar2.jpg', 10, 'gambar2\_dicerahkan.jpg')  citrapencerahan('gambar2.jpg', -170, 'gambar2\_digelapkan.jpg')  citrapencerahan('gambar3.jpg', 10, 'gambar3\_dicerahkan.jpg')  citrapencerahan('gambar3.jpg', -170, 'gambar3\_digelapkan.jpg')  citrapencerahan('gambar4.jpg', 10, 'gambar4\_dicerahkan.jpg')  citrapencerahan('gambar4.jpg', -170, 'gambar4\_digelapkan.jpg')  citrapencerahan('gambar5.jpg', 10, 'gambar5\_dicerahkan.jpg')  citrapencerahan('gambar5.jpg', -120, 'gambar5\_digelapkan.jpg') | |
| CITRA PENCERAHAN | CITRA PENGGELAPAN |
|  |  |
|  |  |
| C:\Users\Wisnu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\gambar3_dicerahkan.jpg |  |
|  | C:\Users\Wisnu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\gambar4_digelapkan.jpg |
| C:\Users\Wisnu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\gambar5_dicerahkan.jpg |  |

1. CITRA BINER

|  |  |  |  |
| --- | --- | --- | --- |
| from PIL import Image  #fungsi citra biner  def citrabiner(gambar, nilaiambang):  CITRA\_GRAYSCALE = Image.open(gambar).convert('L')  PIXEL\_GRAYSCALE = CITRA\_GRAYSCALE.load()  ukuran\_horizontal = CITRA\_GRAYSCALE.size[0]  ukuran\_vertikal = CITRA\_GRAYSCALE.size[1]  for x in range(ukuran\_horizontal):  for y in range(ukuran\_vertikal):  if PIXEL\_GRAYSCALE[x,y] < nilaiambang:  PIXEL\_GRAYSCALE[x,y] = 0  else:  PIXEL\_GRAYSCALE[x,y] = 255  after\_save = gambar + str(nilaiambang) + '.jpg'  CITRA\_GRAYSCALE.save(after\_save)  citrabiner('gambar1.jpg', 50)  citrabiner('gambar1.jpg',128)  citrabiner('gambar1.jpg',200)  citrabiner('gambar1.jpg',230)  citrabiner('gambar2.jpg', 50)  citrabiner('gambar2.jpg',128)  citrabiner('gambar2.jpg',200)  citrabiner('gambar2.jpg',230)  citrabiner('gambar3.jpg', 50)  citrabiner('gambar3.jpg',128)  citrabiner('gambar3.jpg',200)  citrabiner('gambar3.jpg',230)  citrabiner('gambar4.jpg', 50)  citrabiner('gambar4.jpg',128)  citrabiner('gambar4.jpg',200)  citrabiner('gambar4.jpg',230)  citrabiner('gambar5.jpg', 50)  citrabiner('gambar5.jpg',128)  citrabiner('gambar5.jpg',200)  citrabiner('gambar5.jpg',230) | | | |
| Nilai Ambang 50 | Nilai Ambang 128 | Nilai Ambang 200 | Nilai Ambang 230 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. CITRA DILATASI

|  |  |
| --- | --- |
| from PIL import Image  from math import floor  def citraskala(gambar, s):  CITRA\_SKALA = Image.open(gambar)  PIXEL = CITRA\_SKALA.load()  ukuran\_horizontal = CITRA\_SKALA.size[0]  ukuran\_vertikal = CITRA\_SKALA.size[1]  ukuran\_horizontal\_baru = floor(ukuran\_horizontal \* s)  ukuran\_vertikal\_baru = floor(ukuran\_vertikal \* s)    CITRA\_BARU = Image.new("RGB", (ukuran\_horizontal\_baru, ukuran\_vertikal\_baru))  PIXEL\_BARU = CITRA\_BARU.load()  for x in range(ukuran\_horizontal\_baru):  for y in range(ukuran\_vertikal\_baru):  x\_lama = ukuran\_horizontal \* x / ukuran\_horizontal\_baru  y\_lama = ukuran\_vertikal \* y / ukuran\_vertikal\_baru  PIXEL\_BARU[x, y] = PIXEL[x\_lama, y\_lama]  after\_save = gambar + ' dilatasi skala ' + str(s).replace('.', ',') + '.jpg'  CITRA\_BARU.save(after\_save)  citraskala('gambar1.jpg', 0.7)  citraskala('gambar1.jpg', 1.6)  citraskala('gambar2.jpg', 0.7)  citraskala('gambar2.jpg', 1.6)  citraskala('gambar3.jpg', 0.7)  citraskala('gambar3.jpg', 1.6)  citraskala('gambar4.jpg', 0.7)  citraskala('gambar4.jpg', 1.6)  citraskala('gambar5.jpg', 0.7) | |
| SKALA 0.7 | SKALA 1.6 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

1. CITRA FLIPPING

|  |  |  |
| --- | --- | --- |
| from PIL import Image  def flippingvertikal(gambar, after\_save):  CITRA\_FLIPPING = Image.open(gambar)  PIXEL = CITRA\_FLIPPING.load()  ukuran\_horizontal = CITRA\_FLIPPING.size[0]  ukuran\_vertikal = CITRA\_FLIPPING.size[1]  CITRA\_BARU = Image.new("RGB", (ukuran\_horizontal, ukuran\_vertikal))  PIXEL\_BARU = CITRA\_BARU.load()  for x in range(ukuran\_horizontal):  for y in range(ukuran\_vertikal):  PIXEL\_BARU[x, y] = PIXEL[x, ukuran\_vertikal - 1 - y]  CITRA\_BARU.save(after\_save)  def flippinghorizontal(gambar, after\_save):  CITRA\_FLIPPING = Image.open(gambar)  PIXEL = CITRA\_FLIPPING.load()  ukuran\_horizontal = CITRA\_FLIPPING.size[0]  ukuran\_vertikal = CITRA\_FLIPPING.size[1]  CITRA\_BARU = Image.new("RGB", (ukuran\_horizontal, ukuran\_vertikal))  PIXEL\_BARU = CITRA\_BARU.load()  for x in range(ukuran\_horizontal):  for y in range(ukuran\_vertikal):  PIXEL\_BARU[x, y] = PIXEL[ukuran\_horizontal - 1 - x, y]  CITRA\_BARU.save(after\_save)  def flippingtitikasal(gambar, after\_save):  CITRA\_FLIPPING = Image.open(gambar)  PIXEL = CITRA\_FLIPPING.load()  ukuran\_horizontal = CITRA\_FLIPPING.size[0]  ukuran\_vertikal = CITRA\_FLIPPING.size[1]  CITRA\_BARU = Image.new("RGB", (ukuran\_horizontal, ukuran\_vertikal))  PIXEL\_BARU = CITRA\_BARU.load()  for x in range(ukuran\_horizontal):  for y in range(ukuran\_vertikal):  PIXEL\_BARU[x, y] = PIXEL[ukuran\_horizontal - 1 - x, ukuran\_vertikal -1 -y]  CITRA\_BARU.save(after\_save)  flippingvertikal('gambar1.jpg', 'gambar1\_flip\_vertikal.jpg')  flippinghorizontal('gambar1.jpg', 'gambar1\_flip\_horizontal.jpg')  flippingtitikasal('gambar1.jpg', 'gambar1\_flip\_titik\_asal.jpg')  flippingvertikal('gambar2.jpg', 'gambar2\_flip\_vertikal.jpg')  flippinghorizontal('gambar2.jpg', 'gambar2\_flip\_horizontal.jpg')  flippingtitikasal('gambar2.jpg', 'gambar2\_flip\_titik\_asal.jpg')  flippingvertikal('gambar3.jpg', 'gambar3\_flip\_vertikal.jpg')  flippinghorizontal('gambar3.jpg', 'gambar3\_flip\_horizontal.jpg')  flippingtitikasal('gambar3.jpg', 'gambar3\_flip\_titik\_asal.jpg')  flippingvertikal('gambar4.jpg', 'gambar4\_flip\_vertikal.jpg')  flippinghorizontal('gambar4.jpg', 'gambar4\_flip\_horizontal.jpg')  flippingtitikasal('gambar4.jpg', 'gambar4\_flip\_titik\_asal.jpg')  flippingvertikal('gambar5.jpg', 'gambar5\_flip\_vertikal.jpg')  flippinghorizontal('gambar5.jpg', 'gambar5\_flip\_horizontal.jpg')  flippingtitikasal('gambar5.jpg', 'gambar5\_flip\_titik\_asal.jpg') | | |
| FLIP VERTIKAL | FLIP HORIZONTAL | FLIP TITIK ASAL |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

1. CITRA GRAYSCALE

|  |
| --- |
| from PIL import Image  def citragrayscale(gambar):  CITRA\_GRAYSCALE = Image.open(gambar)  NEWGRAY = CITRA\_GRAYSCALE.convert('L')    after\_save = gambar + 'negatif.jpg'  NEWGRAY.save(after\_save)    citragrayscale('gambar1.jpg')  citragrayscale('gambar2.jpg')  citragrayscale('gambar3.jpg')  citragrayscale('gambar4.jpg')  citragrayscale('gambar5.jpg') |
| CITRA GRAYSCALE |
|  |
|  |
|  |
|  |
|  |

1. CITRA NEGATIF

|  |
| --- |
| from PIL import Image  def citranegatif(gambar):  CITRA\_NEGATIF = Image.open(gambar)  ukuran\_horizontal = CITRA\_NEGATIF.size[0]  ukuran\_vertikal = CITRA\_NEGATIF.size[1]  PIXEL = CITRA\_NEGATIF.load()  for x in range(ukuran\_horizontal):  for y in range(ukuran\_vertikal):  R = 255 - PIXEL[x, y][0]  G = 255 - PIXEL[x, y][1]  B = 255 - PIXEL[x, y][2]  PIXEL[x, y] = (R, G, B)  after\_save = gambar + 'negatif.jpg'  CITRA\_NEGATIF.save(after\_save)    citranegatif('gambar1.jpg')  citranegatif('gambar2.jpg')  citranegatif('gambar3.jpg')  citranegatif('gambar4.jpg')  citranegatif('gambar5.jpg') |
| CITRA NEGATIF |
|  |
|  |
|  |
|  |
| C:\Users\Wisnu\AppData\Local\Microsoft\Windows\INetCache\Content.Word\gambar5.jpgnegatif.jpg |

1. CITRA ROTASI

|  |  |  |  |
| --- | --- | --- | --- |
| from PIL import Image  from math import sin, cos  def citrarotasi(gambar, derajat):  CITRA\_ROTASI = Image.open(gambar)  PIXEL = CITRA\_ROTASI.load()  ukuran\_horizontal = CITRA\_ROTASI.size[0]  ukuran\_vertikal = CITRA\_ROTASI.size[1]  CITRA\_BARU = Image.new("RGB", (ukuran\_horizontal, ukuran\_vertikal))  PIXEL\_BARU = CITRA\_BARU.load()  x\_tengah = ukuran\_horizontal // 2  y\_tengah = ukuran\_vertikal // 2  for x in range(ukuran\_horizontal):  for y in range(ukuran\_vertikal):    theta = derajat \* 22/7 / 180  x\_baru = (cos(theta) \* (x - x\_tengah) - sin(theta)  \* (y - y\_tengah) + x\_tengah)  y\_baru = (sin(theta) \* (x - x\_tengah) + cos(theta)  \* (y - y\_tengah) + y\_tengah)  if (x\_baru >= ukuran\_horizontal or y\_baru >= ukuran\_vertikal  or x\_baru < 0 or y\_baru < 0):  PIXEL\_BARU[x, y] = (0, 0, 0)  else:  PIXEL\_BARU[x, y] = PIXEL[x\_baru, y\_baru]  after\_save = gambar + ' rotasi ' + str(derajat) + '.jpg'  CITRA\_BARU.save(after\_save)  citrarotasi('gambar1.jpg', 45)  citrarotasi('gambar1.jpg', 90)  citrarotasi('gambar1.jpg', 180)  citrarotasi('gambar1.jpg', 270)  citrarotasi('gambar2.jpg', 45)  citrarotasi('gambar2.jpg', 90)  citrarotasi('gambar2.jpg', 180)  citrarotasi('gambar2.jpg', 270)  citrarotasi('gambar3.jpg', 45)  citrarotasi('gambar3.jpg', 90)  citrarotasi('gambar3.jpg', 180)  citrarotasi('gambar3.jpg', 270)  citrarotasi('gambar4.jpg', 45)  citrarotasi('gambar4.jpg', 90)  citrarotasi('gambar4.jpg', 180)  citrarotasi('gambar4.jpg', 270)  citrarotasi('gambar5.jpg', 45)  citrarotasi('gambar5.jpg', 90)  citrarotasi('gambar5.jpg', 180)  citrarotasi('gambar5.jpg', 270) | | | |
| 45 derajat | 90 derajat | 180 derajat | 270 derajat |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. HISTOGRAM WARNA

|  |  |
| --- | --- |
| from matplotlib import pyplot  from os.path import basename  from os.path import splitext  from PIL import Image  def get\_file\_name(file\_path):  return splitext(basename(file\_path))[0]  def tampilkan\_histogram(r, g, b, gambar):  intensitas = list(range(256))  lebar\_bar = 0.3  intensitas = [i-lebar\_bar for i in intensitas]  pyplot.bar(intensitas, r, width=lebar\_bar, color='r')  intensitas = [i+lebar\_bar for i in intensitas]  pyplot.bar(intensitas, g, width=lebar\_bar, color='g')  intensitas = [i+lebar\_bar for i in intensitas]  pyplot.bar(intensitas, b, width=lebar\_bar, color='b')  pyplot.title('Histogram ' + gambar)  pyplot.xlabel('Intensitas')  pyplot.ylabel('Kemunculan')  pyplot.legend(['R', 'G', 'B'])  pyplot.savefig('histogram ' + get\_file\_name(gambar) + '.jpg')  pyplot.show()    def histogram(gambar):  GAMBAR = Image.open(gambar)  PIXEL = GAMBAR.load()  ukuran\_horizontal = GAMBAR.size[0]  ukuran\_vertikal = GAMBAR.size[1]  gambar\_r = Image.new('RGB', (ukuran\_horizontal, ukuran\_vertikal))  pixel\_r = gambar\_r.load()  gambar\_g = Image.new('RGB', (ukuran\_horizontal, ukuran\_vertikal))  pixel\_g = gambar\_g.load()  gambar\_b = Image.new('RGB', (ukuran\_horizontal, ukuran\_vertikal))  pixel\_b = gambar\_b.load()  r = [0] \* 256  g = [0] \* 256  b = [0] \* 256  for x in range(ukuran\_horizontal):  for y in range(ukuran\_vertikal):  intensitas\_r = PIXEL[x, y][0]  intensitas\_g = PIXEL[x, y][1]  intensitas\_b = PIXEL[x, y][2]  r[intensitas\_r] += 1  g[intensitas\_g] += 1  b[intensitas\_b] += 1  pixel\_r[x, y] = (intensitas\_r, 0, 0)  pixel\_g[x, y] = (0, intensitas\_g, 0)  pixel\_b[x, y] = (0, 0, intensitas\_b)  tampilkan\_histogram(r, g, b, gambar)  histogram('gambar1.jpg')  histogram('gambar2.jpg')  histogram('gambar3.jpg')  histogram('gambar4.jpg')  histogram('gambar5.jpg') | |
| IMAGE | HISTOGRAM WARNA |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

1. HISTOGRAM BINER

|  |  |
| --- | --- |
| # -\*- coding: utf-8 -\*-  """  Created on Wed Sep 22 15:39:03 2021  @author: Wisnu  """  from os.path import basename  from os.path import splitext  from matplotlib import pyplot as plt  def get\_file\_name(file\_path):  return splitext(basename(file\_path))[0]  def histogrambiner(gambar):  img = plt.imread(gambar)  plt.hist(img.flatten(), bins=[-.5,.5,1.5], ec="k")  plt.xticks((0,1))  plt.title('Histogram ' + gambar)  plt.xlabel('Binary')  plt.ylabel('Number of Pixel')  plt.savefig('histogram biner ' + get\_file\_name(gambar) + '.jpg')  plt.show()  histogrambiner('gambar1.jpg')  histogrambiner('gambar2.jpg')  histogrambiner('gambar3.jpg')  histogrambiner('gambar4.jpg')  histogrambiner('gambar5.jpg') | |
| IMAGE | HISTOGRAM BINER |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

1. HISTOGRAM GRAYSCALE

|  |  |
| --- | --- |
| import cv2  from matplotlib import pyplot as plt  img = cv2.imread('gambar5.jpg',0) #file diisi disini  histr = cv2.calcHist([img],[0],None,[256],[0,256])  plt.plot(histr)  plt.show() | |
| IMAGE | HISTOGRAM GRAYSCALE |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |