NAMA : Sonnya Ghandi

KELAS : Informatika Malam A

NIM : 201351131

**Source code :**

import cv2

import numpy as np

*# buka citra*

img = cv2.imread('Lenna.png')

*# menambahkan variable row, col, ch*

*# variable tersebut diisi dengan nilai ukuran pixel dari citra yang dibaca*

*# beserta banyaknya chanel pada citra*

row, col, ch = img.shape

*# membuat variable kanvas\_hsv*

*# variable ini dipergunakan untuk membuat canvas kosongan yang nanti akan diisi dengan citra berformat hsv*

*# yang berupa array 3 dimensi yang isinya deretan angka 0*

*# np.uint8 sendiri berupa fungsi untuk membatasi besaran nilai elemen-elemen pada array*

*# dengan maksimal nilainya adalah 8 bit*

kanvas\_hsv = np.zeros((row, col, 3), np.uint8)

*# ubah citra tersebut menjadi HSV Color space.*

for i in range (0, row):

  for j in range (0, col):

    b = img[i,j,0]/255

    g = img[i,j,1]/255

    r = img[i,j,2]/255

    vmax = max(r, g, b)

    vmin = min(r, g, b)

*# value*

*# variable v ini diisi dengan nilai terbesar diantara variable r,g,b*

    v = vmax

*# variable ini diperuntukan menyimpan nilai dari rumus v - min(R,G,B)*

*# supaya tidak mengulangi menulis rumus yang sama berulang kali*

    valv\_min = v - vmin

*# saturation*

*# variable s ini jika v tidak sama dengan 0 maka diisi dengan variable valv\_min dibagi v,*

*# jika v sama dengan 0 maka diisi dengan 0*

    s = valv\_min/v if v != 0 else 0

*# hue*

*# mengisi variable h dengan beberapa kemungkinan*

*# jika nilai v sama dengan r maka h diisi dengan (60\*(g-b))/valv\_min*

    if v == r :

      h = (60\*(g-b))/valv\_min

*# jika nilai v sama dengan g maka h diisi dengan ((120+60)\*(b-r))/valv\_min*

    elif v == g :

      h = ((120+60)\*(b-r))/valv\_min

*# jika nilai v sama dengan b maka h diisi dengan ((240+60)\*(b-r))/valv\_min*

    elif v == b :

      h = ((240+60)\*(b-r))/valv\_min

*# jika nilai r sama dengan g dan juga sama dengan b maka h diisi dengan 0*

    elif r == g and g == b :

      h = 0

*# untuk menghindari nilai h itu minus maka jika h kurang dari 0 nilai h diisi dengan nilai h + 360,*

*# jika tidak minus maka nilai h adalah h itu sendiri*

    h = h + 360 if h < 0 else h

    kanvas\_hsv[i,j,0] = h/2

    kanvas\_hsv[i,j,1] = s\*255

    kanvas\_hsv[i,j,2] = v\*255

*# split citra menggunakan fungsi cv.split*

h,s,v = cv2.split(kanvas\_hsv)

*# variable penampung untuk total setiap chanel*

total\_h = 0

total\_s = 0

total\_v = 0

*# jumlahkan setiap channel hsv menggunakan looping*

for i in range (0, row):

  for j in range (0, col):

*# hue*

*# variable total\_h ditambah dengan nilai elemen h pada baris ke i dan kolom ke j*

    total\_h += h[i][j]

*# saturation*

*# variable total\_s ditambah dengan nilai elemen s pada baris ke i dan kolom ke j*

    total\_s += s[i][j]

*# value*

*# variable total\_v ditambah dengan nilai elemen v pada baris ke i dan kolom ke j*

    total\_v += v[i][j]

*# menghitung jumlah n pixel pada citra*

n = col \* row

*# rata-rata channel h*

*# menghitung rata - rata chanel h dengan membagi nilai total pada chanel h dengan total jumlah n pixel*

rata\_h = total\_h / n

*# rata-rata channel s*

*# menghitung rata - rata chanel s dengan membagi nilai total pada chanel s dengan total jumlah n pixel*

rata\_s = total\_s / n

*# rata-rata channel v*

*# menghitung rata - rata chanel v dengan membagi nilai total pada chanel v dengan total jumlah n pixel*

rata\_v = total\_v / n

*# lakukan thresholding manual menggunakan looping*

for i in range (0, row):

  for j in range (0, col):

*# hue*

*# variable kanvas\_hsv chanel hue pada baris ke i dan kolom ke j*

*# diisi dengan 0 jika nilai kanvas\_hsv chanel hue pada baris ke i dan kolom ke j < dari rata\_h*

*# jika tidak diisi dengan 255*

    kanvas\_hsv[i,j,0] = 0 if kanvas\_hsv[i,j,0] < rata\_h else 255

*# saturation*

*# variable kanvas\_hsv chanel saturation pada baris ke i dan kolom ke j*

*# diisi dengan 0 jika nilai kanvas\_hsv chanel saturation pada baris ke i dan kolom ke j < dari rata\_s*

*# jika tidak diisi dengan 255*

    kanvas\_hsv[i,j,1] = 0 if kanvas\_hsv[i,j,1] < rata\_s else 255

*# value*

*# variable kanvas\_hsv chanel value pada baris ke i dan kolom ke j*

*# diisi dengan 0 jika nilai kanvas\_hsv chanel value pada baris ke i dan kolom ke j < dari rata\_v*

*# jika tidak diisi dengan 255*

    kanvas\_hsv[i,j,2] = 0 if kanvas\_hsv[i,j,2] < rata\_v else 255

*# menyimpan citra chanel hue kedalam file dengan nama gambar-hue.jpg*

cv2.imwrite("gambar-hue.jpg",kanvas\_hsv[:,:,0])

*# menyimpan citra chanel saturation kedalam file dengan nama gambar-saturation.jpg*

cv2.imwrite("gambar-saturation.jpg",kanvas\_hsv[:,:,1])

*# menyimpan citra chanel value kedalam file dengan nama gambar-value.jpg*

cv2.imwrite("gambar-value.jpg",kanvas\_hsv[:,:,2])

Output :

1. Hue Chanel



1. Saturation Chanel



1. Value Chanel

