



Jurusan Teknologi Informasi Politeknik Negeri Malang
Tugas Minggu-14 : Thresholding, Segmentasi Sederhana
Mata Kuliah Pengolahan Citra dan Visi Komputer
Pengampu: Dr. Eng CAHYA RAHMAD., ST.,M.KOM.
Mei 2021

Tujuan

1. Mahasiswa mampu memahami konsep Thresholding.
2. Mahasiswa dapat mengetahui beberapa Teknik Thresholding.
3. Mahasiswa dapat membuat beberapa Teknik Thresholding menggunakan Python pada Google Colab.

Tugas Praktikum

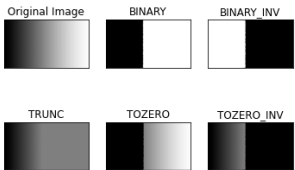
Langkah	Keterangan
1	<p>Buka https://colab.research.google.com/. Setelah dipastikan bahwa google Colab terhubung dengan Github Anda, lanjutkan dengan memilih repository yang telah digunakan pada praktikum minggu lalu, rename file menjadi "Week9.ipynb". Kemudian import folder yang ada di Drive Anda dengan cara sebagai berikut.</p>  <pre>from google.colab import drive drive.mount('/content/drive')</pre> <p>Mounted at /content/drive</p>
2	<p>Import beberapa library berikut yang akan digunakan selama uji coba praktikum minggu ke-6 berikut.</p>  <pre>import cv2 import numpy as np from matplotlib import pyplot as plt</pre>

Buat Global Threshold (BINARY, BINARY_INV, TRUNC, TOZERO, TOZERO_INV) secara manual sesuai dengan deskripsi dari grafik yang ditunjukkan di atas.

3

```
Week14.ipynb
File Edit View Insert Runtime Tools Help Saving...
Comment Share
+ Code + Text
# No 3. Buat Global Threshold (BINARY, BINARY_INV, TRUNC, TOZERO, TOZERO_INV) secara manual sesuai dengan deskripsi dari grafik yang ditunjukkan di atas
img = cv2.imread('/content/drive/MyDrive/PCVK/Images/gradient.jpg')
ret, thresh1 = cv2.threshold(img, 125, 255, cv2.THRESH_BINARY)
ret, thresh2 = cv2.threshold(img, 127, 255, cv2.THRESH_BINARY_INV)
ret, thresh3 = cv2.threshold(img, 127, 255, cv2.THRESH_TRUNC)
ret, thresh4 = cv2.threshold(img, 127, 255, cv2.THRESH_TOZERO)
ret, thresh5 = cv2.threshold(img, 127, 255, cv2.THRESH_TOZERO_INV)

titles = ['Original Image', 'BINARY', 'BINARY_INV', 'TRUNC', 'TOZERO', 'TOZERO_INV']
images = [img, thresh1, thresh2, thresh3, thresh4, thresh5]
for i in range(6):
    plt.subplot(2,3,i+1),plt.imshow(images[i], 'gray', vmin=0, vmax=255)
    plt.title(titles[i])
    plt.xticks([], plt.yticks([]))
plt.show()
```



Buat Otsu Thresholding tanpa menggunakan Library. Tampilkan juga nilai threshold saat anda gunakan Otsu's, seperti terlihat pada gambar hasil berikut. (gunakan image lena_gs_lc2.jpg agar benar2 terlihat beda antara hasil otsu's dengan global threshold biasa).

4

```
Week14.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
+ Code + Text
# No 4. Buat Otsu Thresholding tanpa menggunakan Library. Tampilkan juga nilai threshold saat anda gunakan Otsu's, seperti terlihat pada gambar hasil berikut. (gunakan image lena_gs_lc2.jpg agar benar2 terlihat beda antara hasil otsu's dengan global threshold biasa).
filename = ('/content/drive/MyDrive/PCVK/Images/lena_gs_lc2.jpg')
img = cv2.imread(filename, 0)
blur = cv2.GaussianBlur(img, (5,5), 0)

hist = cv2.calcHist([blur], [0], None, [256], [0, 256])
hist_norm = hist.ravel()/hist.max()
Q = hist_norm.cumsum()
bins = np.arange(256)
wb = 0
sumb = 0
varmax = 0
fn_min = np.inf
thresh = -1
for i in range(0, 255):
    wb += hist[i]
    wf = (img.shape[0]*img.shape[1]) - wb
    sumb += (i*hist[i])
    mb = sumb / wf
    mf = (sumb - sumb)/wf
    betweenvar = wb * wf * (mb - mf) * (mb - mf)
    if (betweenvar > varmax):
        varmax = betweenvar
        thresh = i
x = ("Otsu's Thresholding (v = "+str(thresh)+")")
ret, th1 = cv2.threshold(blur, 127, 255, cv2.THRESH_BINARY)
ret, th2 = cv2.threshold(blur, thresh, 255, cv2.THRESH_BINARY)
```

```
Week14.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
+ Code + Text
x = ("Otsu's Thresholding (v = "+str(thresh)+")")
ret, th1 = cv2.threshold(blur, 127, 255, cv2.THRESH_BINARY)
ret, th2 = cv2.threshold(blur, thresh, 255, cv2.THRESH_BINARY)

titles = ['Citra Asli', 'Global Thresholding (v = 127)', x]
citra3 = [blur, th1, th2]

plt.figure(figsize = (20,15))
for i in range(len(citra3)):
    plt.subplot(1,3,i+1),plt.imshow(citra3[i], 'gray')
    plt.title(titles[i])
    plt.xticks([], plt.yticks([]))
plt.show()
```

/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:19: RuntimeWarning: invalid value encountered in true_divide
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:20: RuntimeWarning: invalid value encountered in true_divide



Buat histogram dari citra tersegmentasi, Histogram hanya pada foreground image saja. Gunakan image Lily atau Leaves yang sudah disediakan di folder images. Petunjuk:

- anda dapat gunakan `cv.calcHist` untuk menampilkan histogram.
- Buka link berikut <https://opencv-tutorial.readthedocs.io/en/latest/histogram/histogram.html>
- Dari link tersebut perhatikan bahwa `cv.calcHist` memiliki salah satu parameter yaitu mask. Jika diset **None**, maka keseluruhan image akan dihitung histogramnya. Jika kita tentukan mask, maka hanya bagian image yang dimasking **warna putih** yang akan dihitung histogramnya (dari contoh dibawah dinamakan dengan **Citra Segment**).

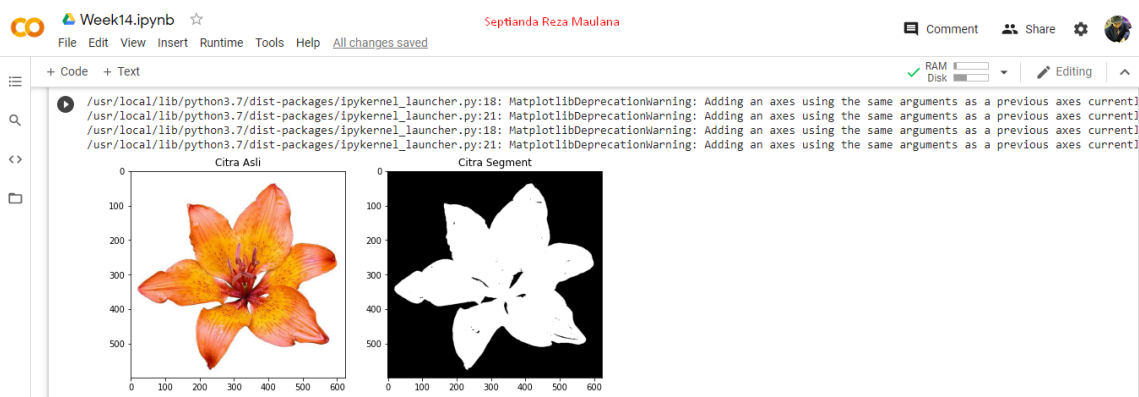
5

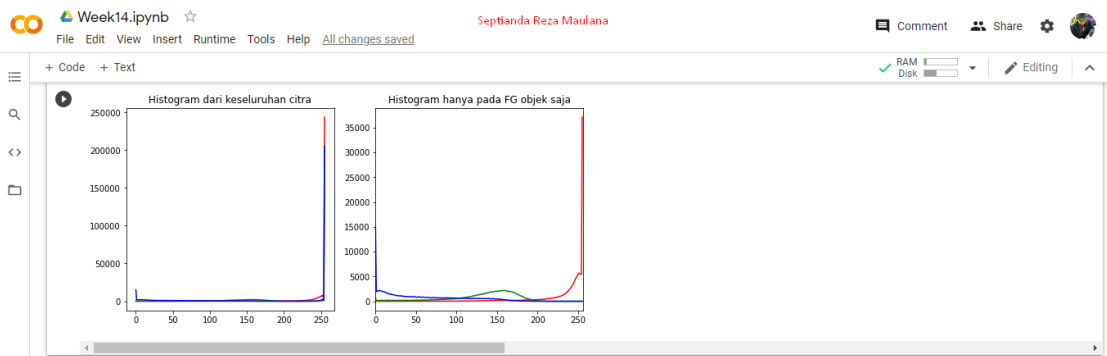
```
# No 5. Buat histogram dari citra tersegmentasi, Histogram hanya pada foreground image saja

filename = ('/content/drive/MyDrive/PCVK/Images/lily.jpg')
img = cv2.imread(filename)
img = cv2.cvtColor(img,cv2.COLOR_BGR2RGB)
img_gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

ret3,mask = cv2.threshold(img_gray,200,255,cv2.THRESH_BINARY_INV)

plt.figure(figsize = (10,10))
plt.subplot(2,2,1),plt.imshow(img)
plt.title('Citra Asli')
plt.subplot(2,2,2),plt.imshow(mask, 'gray')
plt.title('Citra Segment')
color = ('r','g','b')
for i,col in enumerate(color):
    hist0 = cv2.calcHist([img],[i],None,[256],[0,256])
    plt.subplot(2,2,3),plt.plot(hist0,color = col)
    plt.title('Histogram dari keseluruhan citra')
    hist1 = cv2.calcHist([img],[i],mask,[256],[0,256])
    plt.subplot(2,2,4),plt.plot(hist1,color = col)
    plt.title('Histogram hanya pada FG objek saja')
    plt.xlim([0,256])
plt.show()
```





Lakukan segmentasi warna pada image "peppers.jpg", munculkan hanya warna yang kuning saja. (Petunjuk: anda dapat gunakan K-Means untuk menampilkan hanya warna tertentu saja).

```
# No 6. Lakukan segmentasi warna pada image "peppers.jpg", munculkan hanya warna yang kuning saja.
import cv2
import numpy as np
from matplotlib import pyplot as plt

filename = ('/content/drive/MyDrive/PCVK/Images/peppers.jpg')

img = cv2.imread(filename)
img = cv2.cvtColor(img,cv2.COLOR_BGR2RGB)

pixel_values = img.reshape((-1, 3))
pixel_values = np.float32(pixel_values)

#syarat berhenti iterasi dr KMeans adalah jika centroid sudah tidak terlalu banyak pergeseran posisi antara iterasi sekarang dengan iterasi sebelumnya (konverge)
criteria = (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER, 100, 0.2)

#jika diperhatikan pada image asli, terdapat 3 warna utama (hijau, biru, dan putih/orange). untuk percobaan ini kita akan gunakan 3 cluster untuk image ini
k = 3
_, labels, (centers) = cv2.kmeans(pixel_values, k, None, criteria, 10, cv2.KMEANS_RANDOM_CENTERS)

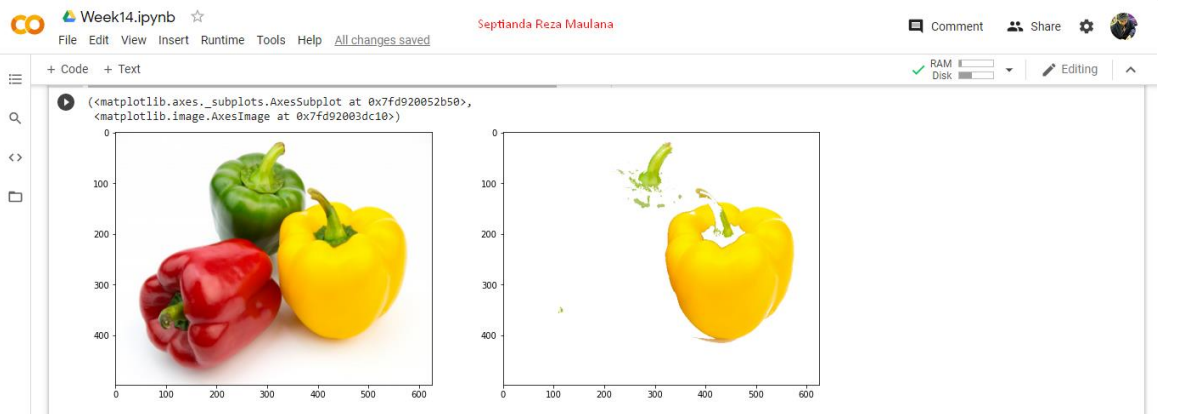
#konversi titik centroid kedalam integer
centers = np.uint8(centers)
```

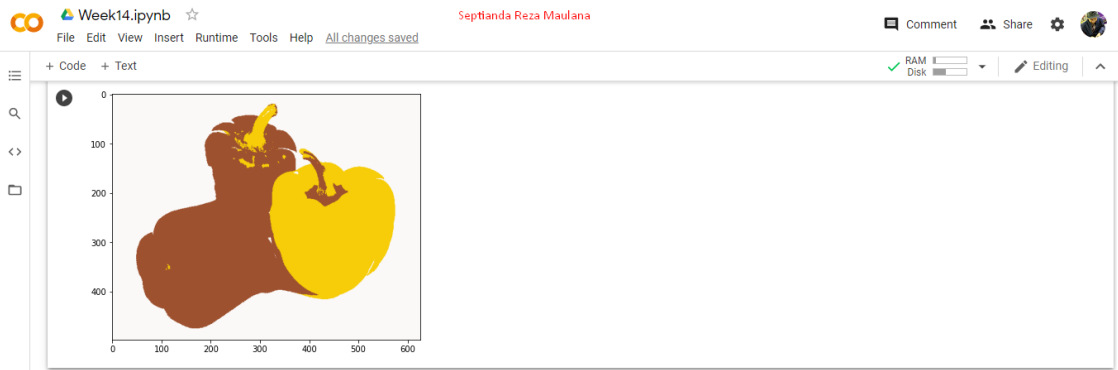
```
#Flattening label array
labels = labels.flatten()

#konversi warna pixel asli ke warna dari tiap centroidnya
segmented_image = centers[labels.flatten()]
# reshape ke bentuk image asli
segmented_image = segmented_image.reshape(img.shape)

masked_image = np.copy(img)
masked_image0 = masked_image.reshape((-1, 3))
masked_image0[labels != 1] = [255, 255, 255]
masked_image0 = masked_image0.reshape(img.shape)

plt.figure(figsize = (15,12))
plt.subplot(2,2,1),plt.imshow(img)
plt.subplot(2,2,2),plt.imshow(masked_image0)
plt.subplot(2,2,3),plt.imshow(segmented_image)
```





Saat anda menampilkan warna tertentu, jelaskan kendala yang di hadapi dan mengapa hal itu terjadi.

Jawab : Kendala yang terjadi warna yang di dapat random dikarenakan berdasarkan cluster terdekat.

Buka File crossword.jpg. Dengan pengetahuan tentang thresholding yang sudah anda pelajari. Lakukan thresholding biner dengan hasil terbaik menurut anda. Copas code dan hasil image di modul ini.

7

```
# No 7. Buka File crossword.jpg. Dengan pengetahuan tentang thresholding yang sudah anda pelajari. Lakukan thresholding biner dengan hasil terbaik menurut anda.

img = cv2.imread('/content/drive/MyDrive/PCVK/Images/crossword.jpg',0)
ret,thresh1 = cv2.threshold(img,127,255,cv2.THRESH_BINARY)
ret,thresh2 = cv2.threshold(img,127,255,cv2.THRESH_BINARY_INV)
ret,thresh3 = cv2.threshold(img,127,255,cv2.THRESH_TRUNC)
ret,thresh4 = cv2.threshold(img,127,255,cv2.THRESH_TOZERO)
ret,thresh5 = cv2.threshold(img,127,255,cv2.THRESH_TOZERO_INV)

titles = ['Original Image','BINARY','BINARY_INV','TRUNC','TOZERO','TOZERO_INV']
images = [img, thresh1, thresh2, thresh3, thresh4, thresh5]

for i in range(6):
    plt.subplot(2,3,i+1),plt.imshow(images[i],'gray',vmin=0,vmax=255)
    plt.title(titles[i])
    plt.xticks([],plt.yticks([]))

plt.show()
```

