

Muhammad Irfan Amrullah
TugasLab2 Pengolahan Citra
1706039585 / A

1. A.

```
In [6]: e1 = color.rgb2gray(i1)
plt.imshow(e1, cmap = 'gray')
plt.show()
```



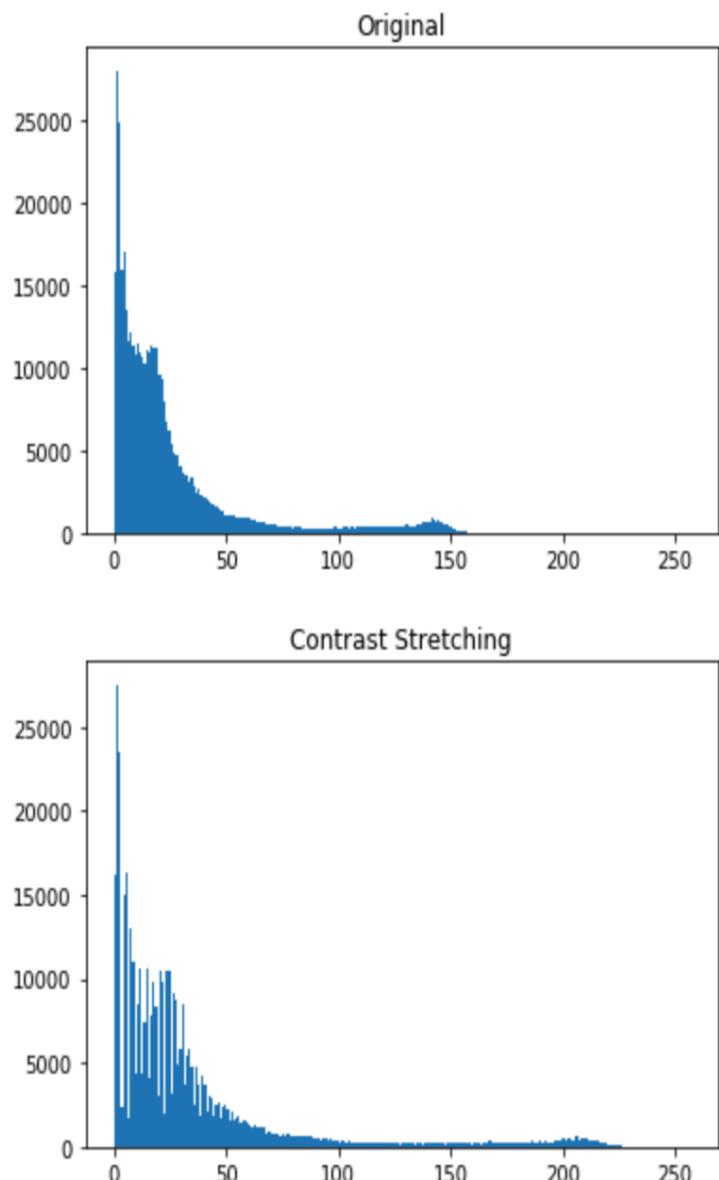
B. Kuantitas tertinggi berada di range 0 – 150. Karena gambar tersebut gelap, sehingga banyak pixel yang nilainya mendekati 0. Range terjauh berada di 150.

C.

```
plt.show();
gray2 = util.img_as_float(e1_cs)
plt.subplot(); plt.hist(gray2.flatten(), 256, range=(0,256))
plt.title('Contrast Stretching');
plt.show();
```



Perbedaan pada gambar sangat sedikit sekali, tapi jika kita cerahkan gambarnya dapat dilihat kontras gambar setelah dilakukan contrast Stretching meningkat.



Pada Histogram bisa diliat range tadinya dipetakan hanya sampai 150 Kembali dipetakan hingga rangenya mencapai 225. Pada bagian range yang gelap bisa diliat adanya jarak antara pixel yang menandakan bahwa pixel2 yang gelap sudah dinaikan hingga berbeda sedikit.

D.

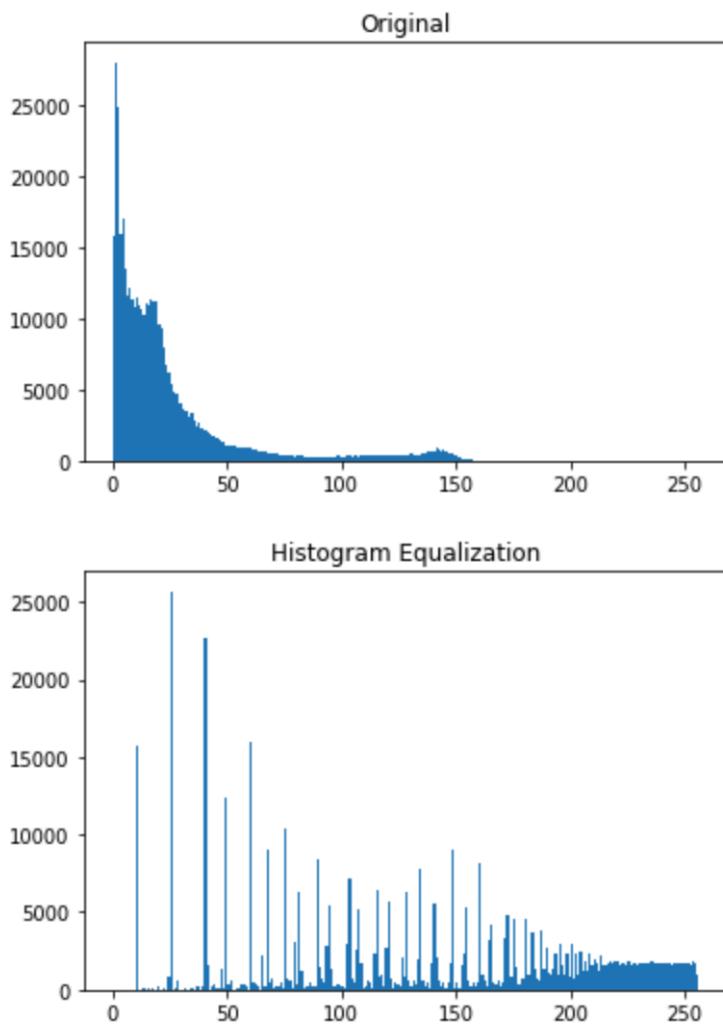
Original



Histogram Equalization



Gambar terlihat sangat terang setelah dilakukan histogram equalization.



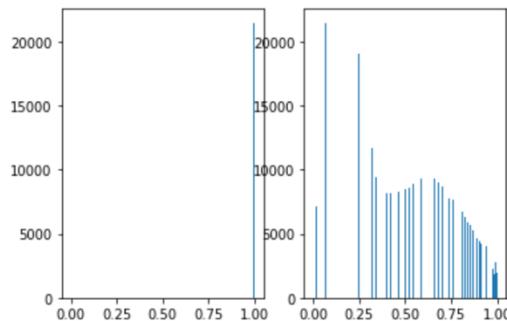
Pada Histogram, histogram equalization menyebarkan range yang tadinya hanya sampai 150, dibuat secara merata dari 0 – 256.

D. Contrast Stretching menaikan contras dengan menarik range intensitas yang ada ke range yang diinginkan. Berbeda dengan histogram equalization yang hanya bisa mengaplikasikan skala linear ke fungsi pixel value gambar. Pada citra yang diberikan, histogram equalization menampilkan gambar yang sangat cerah ketimbang contrast stretching. Yang lebih cocok diberikan adalah Histogram Equalization.

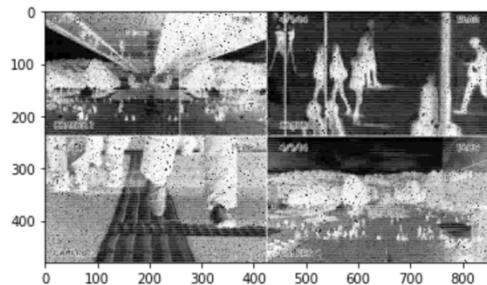
2. A. Pertama saya melakukan histogram equalization agar gambar terlihat lebih terang. Lalu saya lakukan negative, agar background hitam menjadi putih. Kemudian saya lakukan sharpening dengan radius 2 agar detail menjadi lebih detail. Jadilah

gambar yang lebih jelas dari sebelumnya.

```
In [12]: H_eq = exposure.equalize_hist(i2)
plt.subplot(1,2,1); plt.hist(i2.flatten(), 256, range=(0,1))
plt.subplot(1,2,2); plt.hist(H_eq.flatten(), 256, range=(0,1))
plt.show()
plt.imshow(H_eq, cmap='gray')
```

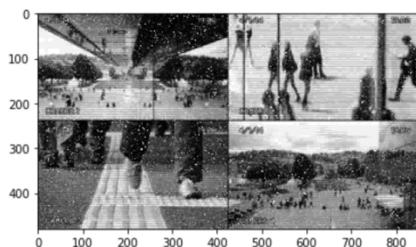


```
Out[12]: <matplotlib.image.AxesImage at 0x109c1b890>
```



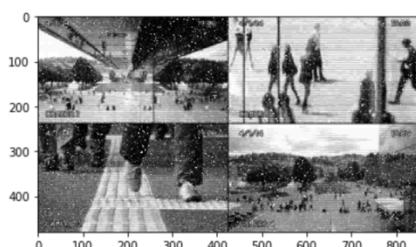
```
In [14]: i3 = 255-H_eq
plt.subplot(); plt.imshow(i3, cmap='gray')
```

```
Out[14]: <matplotlib.image.AxesImage at 0x1274d4dd0>
```



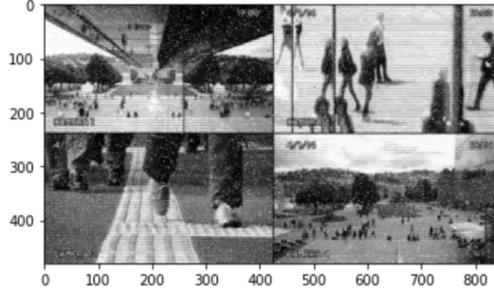
```
In [15]: from skimage import exposure
eq_i3 = exposure.equalize_hist(i3)
plt.subplot(); plt.imshow(eq_i3, cmap='gray')
```

```
Out[15]: <matplotlib.image.AxesImage at 0x1279dbc0>
```



```
In [17]: from skimage import filters, morphology
gray3 = eq_i3
fi = filters.rank.mean(gray3, selem=morphology.square(2))
sh = filters.unsharp_mask(fi, radius=(2))
plt.subplot(); plt.imshow(fi, cmap='gray', vmin=0, vmax=255)
```

Out[17]: <matplotlib.image.AxesImage at 0x1278badd0>



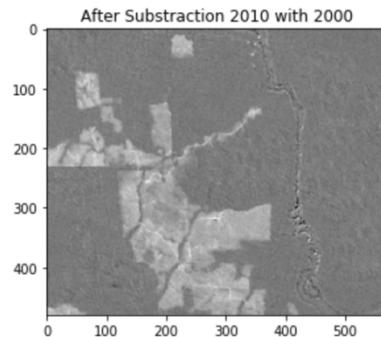
B. Tanggal 4/1/14 jam 19.00



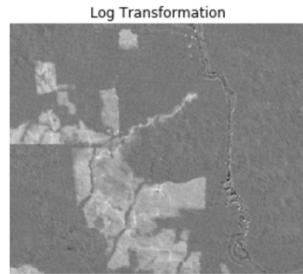
3. A. Kita hanya tinggal mensubtract gambar terbaru dengan gambar lama, kita lakukan log transformation agar lebih cerah. Lakukan Contrast stretching agar kontras lebih meningkat. Terakhir kita naikan threshold dari gambar yang sudah di subtract, maka akan didapatkan gambar sesuai yang diharapkan.

```
In [19]: i4 = io.imread('teles2010.jpg')
i5 = io.imread('teles2000.jpg')
```

```
In [44]: i6 = np.subtract(i4, i5, dtype='int16')
e2 = color.rgb2gray(i6)
plt.subplot(); plt.imshow(e2, cmap='gray');
plt.title('After Subtraction 2010 with 2000')
plt.show()
```

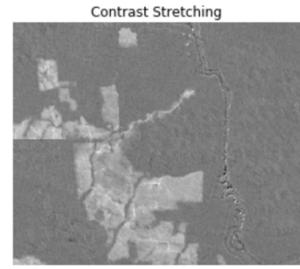


```
In [45]: a = e2/255
c1 = 3
f1 = c1*np.log(1 + (a))
plt.subplot(); plt.imshow(f1, cmap='gray')
plt.title('Log Transformation'); plt.axis("off")
plt.show();
```



```
In [46]: mn = min(f1.flatten())
mx = max(f1.flatten())
b = int(np.floor(255 / (mx - mn)))
f1_cs = (f1 - mn) * b

plt.subplot(); plt.imshow(f1_cs, cmap='gray')
plt.title('Contrast Stretching'); plt.axis("off")
plt.show();
```



```
In [63]: from skimage.filters import threshold_otsu, threshold_minimum
thresh = threshold_minimum(f1_cs)
binary = f1_cs > thresh

plt.subplot(); plt.imshow(binary, cmap='gray')
plt.title("After Threshold"); plt.axis("off")
plt.show();
```



```
In [66]: i6 = io.imread('pantai1.jpg')
i7 = io.imread('pantai2.jpg')

i8 = np.subtract(i6, i7, dtype='int16')
e3 = color.rgb2gray(i8)
plt.subplot(); plt.imshow(e3, cmap='gray');
plt.title('After Subtraction Pantai1 with Pantai2'); plt.axis("off")
plt.show()
```

After Subtraction Pantai1 with Pantai2



```
In [75]: from skimage.filters import threshold_otsu, threshold_yen
thresh = threshold_otsu(e3)
binary = e3 < thresh

plt.subplot(); plt.imshow(binary, cmap='gray')
plt.title("After Threshold"); plt.axis("off")
plt.show();
```

After Threshold



B. Image processing dapat melakukan banyak hal, pada soal ini image processing bahkan dapat mendeteksi perubahan yang terjadi pada gambar. Dengan kemampuan ini kita dapat melihat banyak informasi seperti kutub es yang mencair dan pembakaran hutan. Kemampuan ini tentu sangat dibutuhkan untuk hal yang lebih besar seperti bahan penelitian, bukti ilmiah, yang akan membawa perubahan pada dunia di skala global.