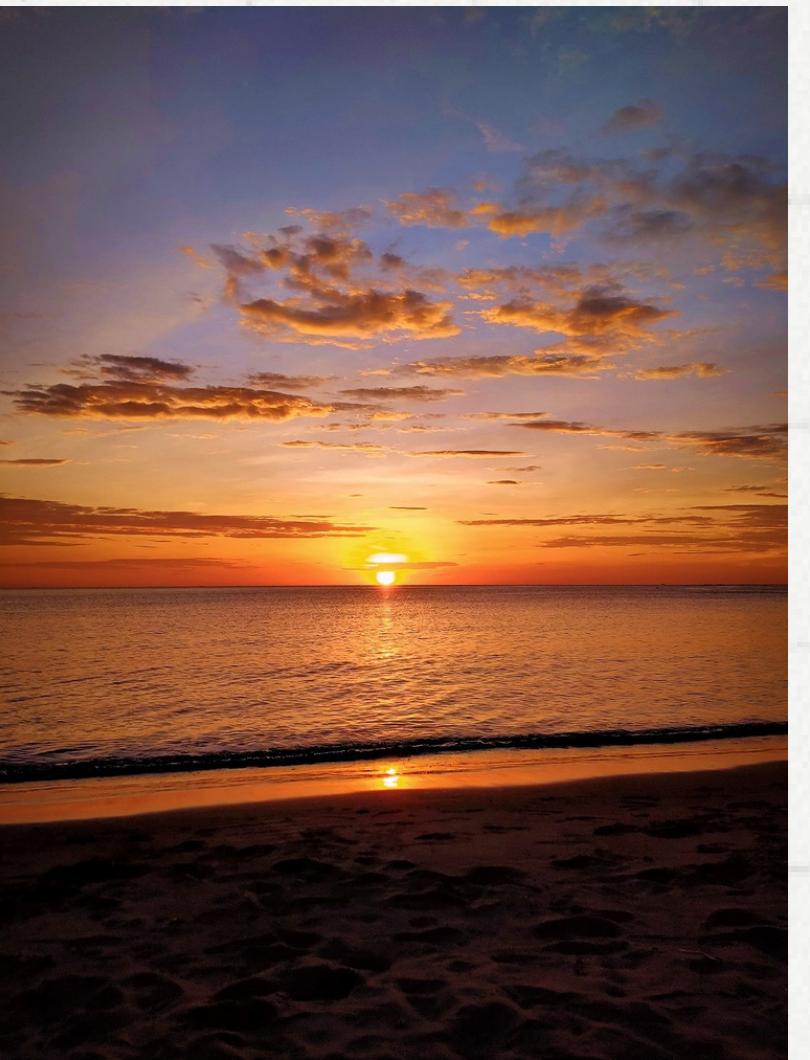


# **APLIKASI PADA ALJABAR LINEAR: KONVERSI CITRA RGB KE GRayscale**

Muhammad Aldi Alfatih - 221011044

Nur Aini Sri Oktaviani Bahar - 221011049

Nur Jamaliyah - 221011109



AldiAlfatih.jpg



Aini.jpg

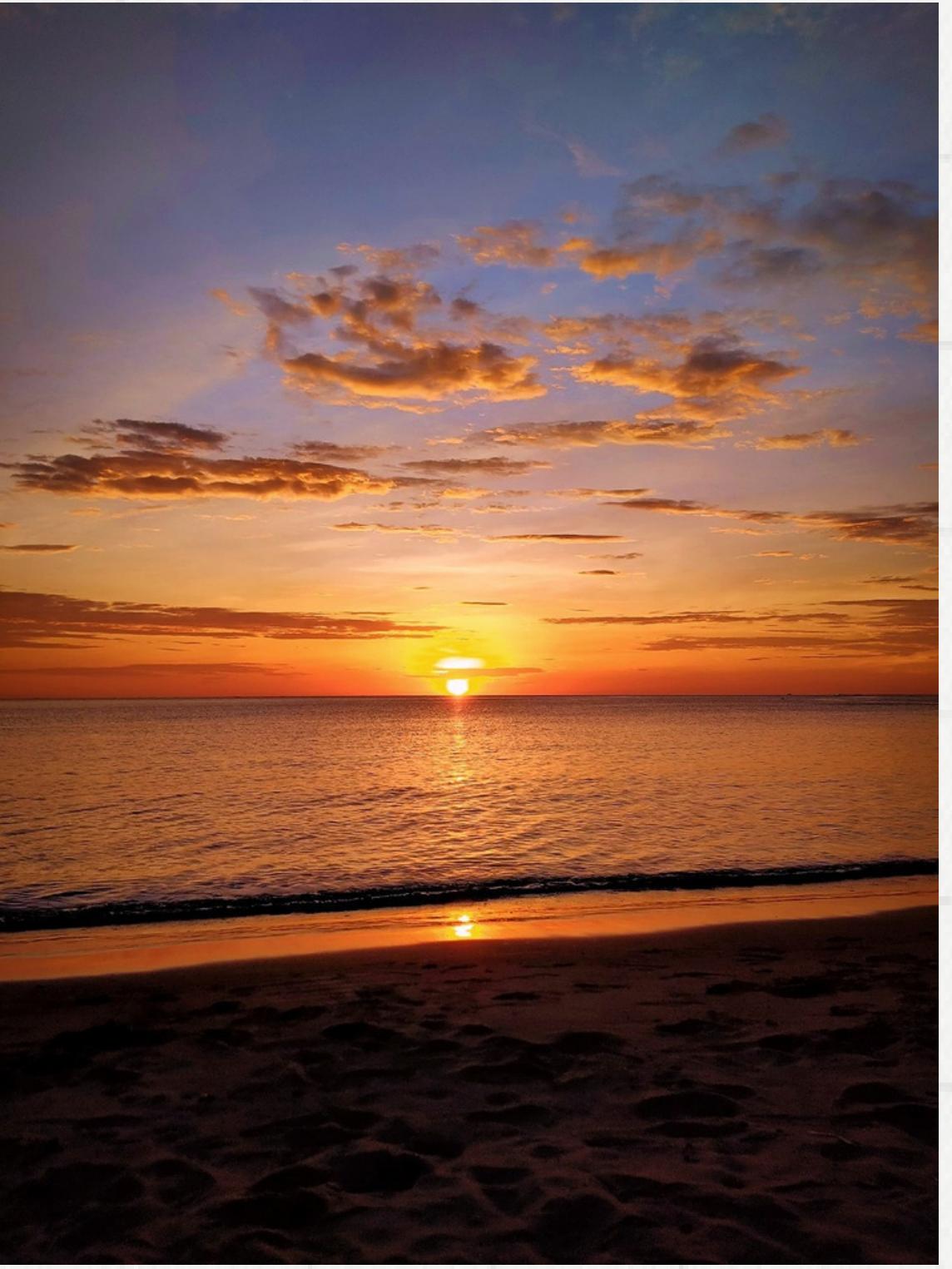


Nurjamaliyah.jpeg

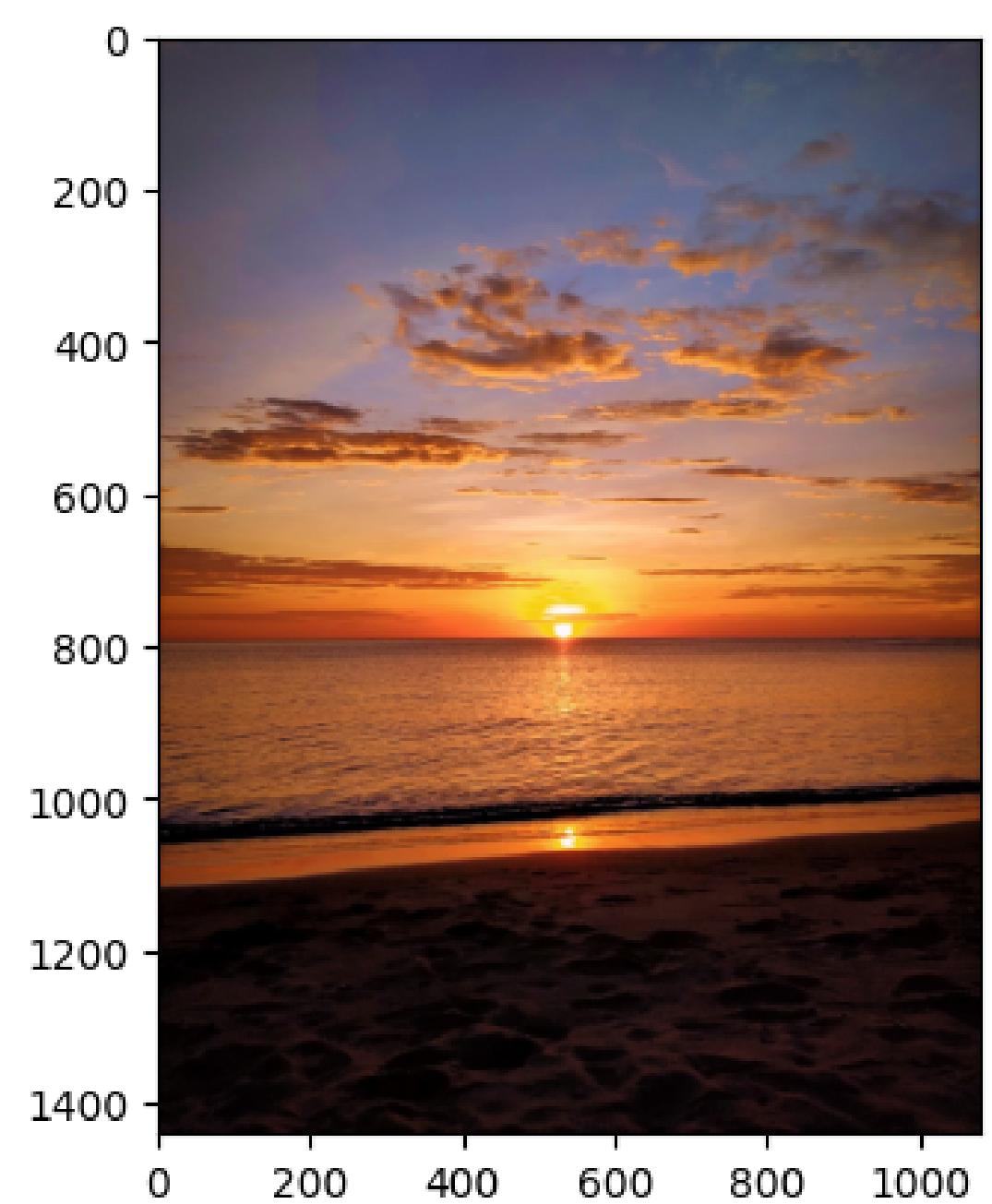
```
import cv2
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

img_path = 'AldiAlfatih.jpg'
img = cv2.imread(img_path)
print(img.shape)
fix_img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
plt.imshow(fix_img)

R, G, B = fix_img[:, :, 0], fix_img[:, :, 1], fix_img[:, :, 2]
print(np.array(fix_img))
```



```
→ (1440, 1080, 3)  
[[[61 54 72]  
[57 50 66]  
[54 50 65]  
...  
[55 55 65]  
[57 55 68]  
[60 58 71]]  
  
[[62 55 73]  
[60 53 69]  
[56 52 66]  
...  
[59 59 69]  
[59 57 70]  
[59 57 70]]
```



# METODE LIGHTNESS

```
fix_img[:] = np.max(fix_img, axis = -1, keepdims = 1) + np.min(fix_img, axis = -1, keepdims = 1) / 2

print(np.array(fix_img[:]))
plt.axis('off')
plt.imshow(fix_img[:])
plt.savefig('metode lightness.jpg', bbox_inches = 'tight')
```

ⓧ [[[148 148 148]  
[136 136 136]  
[135 135 135]  
...  
[138 138 138]  
[142 142 142]  
[150 150 150]]  
  
[[[150 150 150]  
[142 142 142]  
[138 138 138]  
...  
[147 147 147]  
[147 147 147]  
[147 147 147]]]



# METODE AVERAGE

```
gray_img = np.mean(fix_img, axis = 2)
print(np.array(gray_img))

plt.axis('off')
plt.imshow(gray_img, cmap = 'gray')
plt.savefig('Metode Average (box).jpg', bbox_inches = 'tight')
```

```
[[148. 136. 135. ... 138. 142. 150.]
 [150. 142. 138. ... 147. 147. 147.]
 [147. 144. 144. ... 156. 147. 142.]
 ...
 [ 33.  24.  31. ...  25.  31.  34.]
 [ 33.  18.  31. ...  30.  28.  27.]
 [ 36.  19.  31. ...  37.  31.  25.]]
```



# METODE LUMINOSITY

```
lumi_img = (0.2126*R) + (0.7152*G) + (0.0722*B)
print(lumi_img)
plt.axis('off')
plt.imshow(lumi_img, cmap = 'gray')
plt.savefig('Metode Luminosity.png', bbox_inches = 'tight')
```

```
[[148. 136. 135. ... 138. 142. 150.]
 [150. 142. 138. ... 147. 147. 147.]
 [147. 144. 144. ... 156. 147. 142.]
 ...
 [ 33.  24.  31. ...  25.  31.  34.]
 [ 33.  18.  31. ...  30.  28.  27.]
 [ 36.  19.  31. ...  37.  31.  25.]]
```



# METODE WEIGHTED AVERAGE

```
wavr_img = (0.299*R) + (0.587*G) + (0.114*B)  
print(wavr_img)  
plt.axis('off')  
plt.imshow(wavr_img, cmap = 'gray')  
plt.savefig('Metode Weighted Average.png', bbox_inches = 'tight')
```

```
[[148. 136. 135. ... 138. 142. 150.]  
 [150. 142. 138. ... 147. 147. 147.]  
 [147. 144. 144. ... 156. 147. 142.]  
 ...  
 [ 33.  24.  31. ...  25.  31.  34.]  
 [ 33.  18.  31. ...  30.  28.  27.]  
 [ 36.  19.  31. ...  37.  31.  25.]]
```

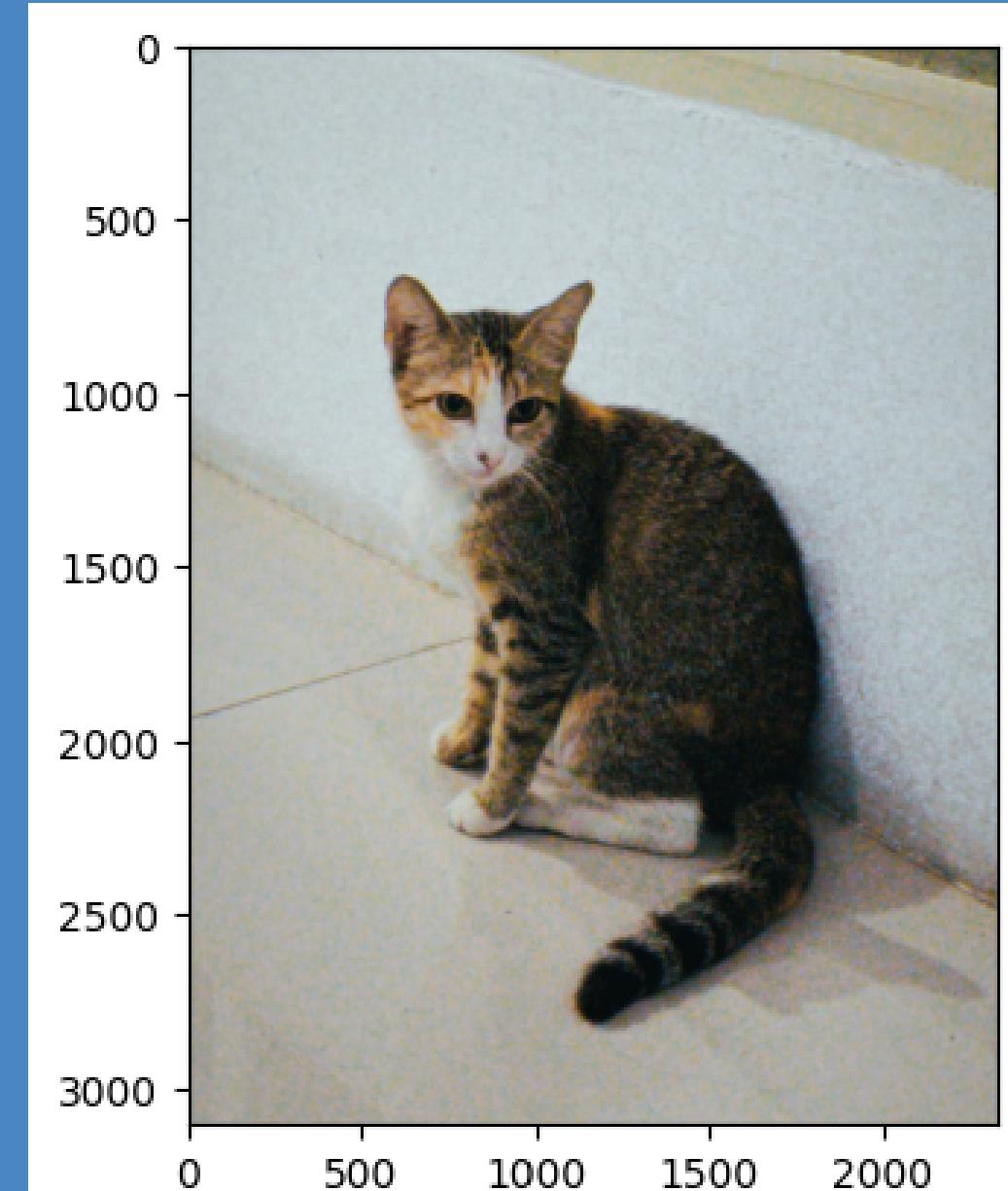


```
▶ import cv2  
import numpy as np  
import matplotlib.pyplot as plt  
%matplotlib inline  
  
img_path = 'Aini.jpg'  
img = cv2.imread(img_path)  
print(img.shape)  
  
plt.imshow(img)  
  
fix_img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)  
plt.imshow(fix_img)  
  
R, G, B = fix_img[:, :, 0], fix_img[:, :, 1], fix_img[:, :, 2]  
print(np.array(fix_img))
```





```
(3106, 2330, 3)  
[[[180 199 197]  
 [180 199 197]  
 [182 198 197]  
 ...  
 [ 87 103 76]  
 [ 91 107 78]  
 [ 91 107 78]]  
  
[[180 199 197]  
 [180 199 197]  
 [182 198 197]]
```



# METODE LIGHTNESS

```
▶ fix_img[:] = np.max(fix_img, axis = -1, keepdims = 1)/2 + np.min(fix_img, axis = -1, keepdims = 1)/2  
print(np.array(fix_img[:]))  
  
plt.axis('off')  
plt.imshow(fix_img[:])  
plt.savefig('Mode Lightness.jpg', bbox_inches='tight')
```

→ [[[189 189 189]  
[189 189 189]  
[190 190 190]  
...  
[ 89 89 89]  
[ 92 92 92]  
[ 92 92 92]]  
  
[[189 189 189]  
[189 189 189]  
[190 190 190]]



# METODE AVERAGE

```
▶ gray_img = np.mean(fix_img, axis = -1)  
print(np.array(gray_img))  
  
plt.axis('off')  
plt.imshow(gray_img, cmap='gray')  
plt.savefig('Metode Average.jpg' , bbox_inches='tight')
```

```
→ [[189. 189. 190. ... 89. 92. 92.]  
 [189. 189. 190. ... 87. 87. 87.]  
 [190. 190. 190. ... 85. 85. 85.]  
 ...  
 [153. 152. 154. ... 156. 156. 156.]  
 [155. 153. 154. ... 158. 158. 158.]  
 [154. 153. 154. ... 159. 158. 158.]]
```



# METODE LUMINOSITY

```
▶ lumi_img = (0.2126*R) + (0.7152*G) + (0.0722*B)  
print(lumi_img)  
  
plt.axis('off')  
plt.imshow(lumi_img, cmap = 'gray')  
plt.savefig('Metode Luminosity.jpg' , bbox_inches='tight')
```

```
→ [[189. 189. 190. ... 89. 92. 92.]  
 [189. 189. 190. ... 87. 87. 87.]  
 [190. 190. 190. ... 85. 85. 85.]  
 ...  
 [153. 152. 154. ... 156. 156. 156.]  
 [155. 153. 154. ... 158. 158. 158.]  
 [154. 153. 154. ... 159. 158. 158.]]
```



# METODE WEIGHTED AVERAGE

```
▶ warv_img = (0.299*R) + (0.587*G) + (0.114*B)  
print(warv_img)  
  
plt.axis('off')  
plt.imshow(warv_img, cmap = 'gray')  
plt.savefig('Metode Weighted Average.jpg' , bbox_inches='tight')
```

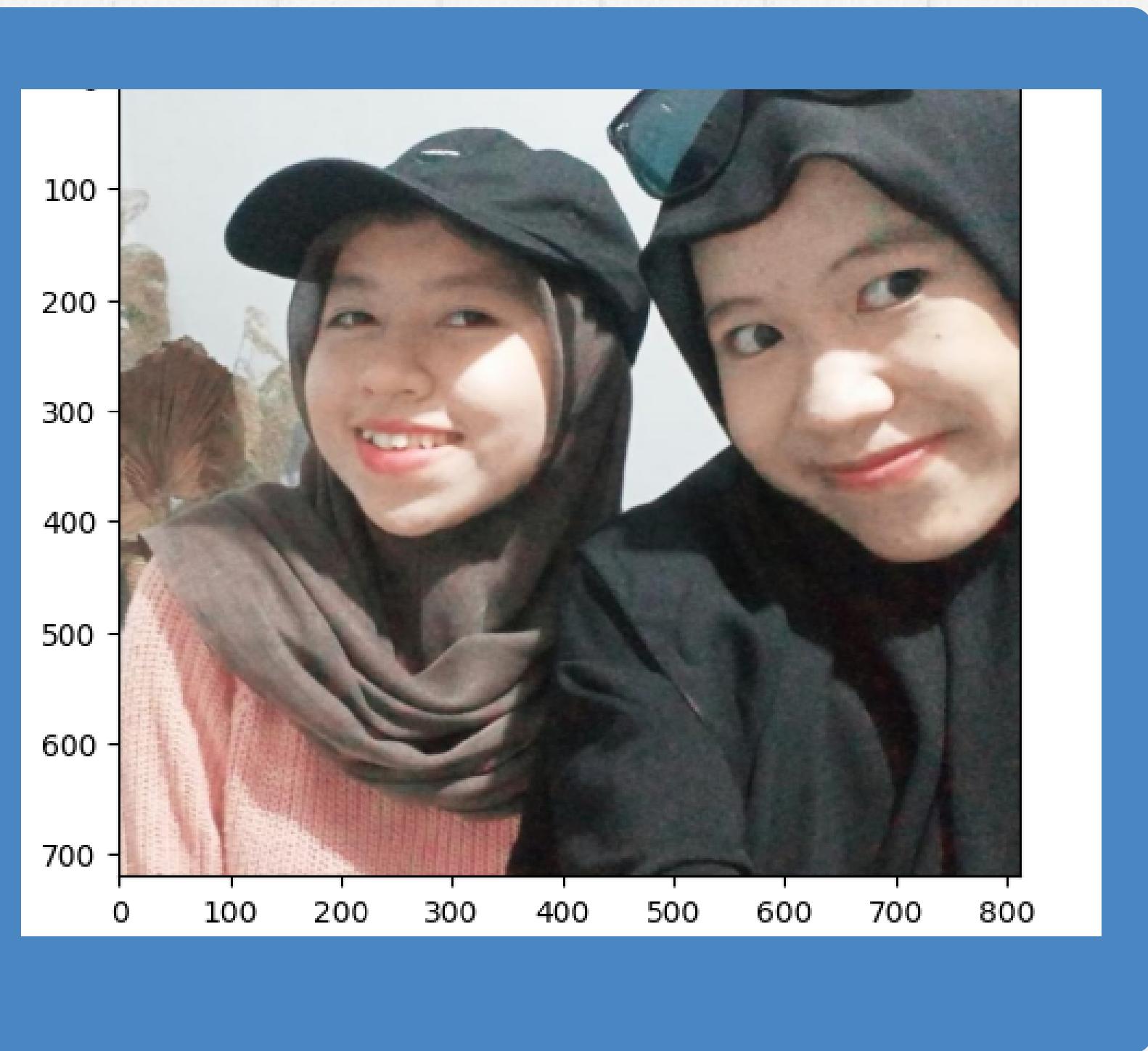
```
⇨ [[189. 189. 190. ... 89. 92. 92.]  
 [189. 189. 190. ... 87. 87. 87.]  
 [190. 190. 190. ... 85. 85. 85.]  
 ...  
 [153. 152. 154. ... 156. 156. 156.]  
 [155. 153. 154. ... 158. 158. 158.]  
 [154. 153. 154. ... 159. 158. 158.]]
```



```
▶ import cv2  
import numpy as np  
import matplotlib.pyplot as plt  
  
img_path = 'Nurjamaliyah.jpeg'  
img = cv2.imread(img_path)  
print(img.shape)  
  
plt.imshow(img)  
  
fix_img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)  
plt.imshow(fix_img)  
  
R, G, B = fix_img[:, :, 0], fix_img[:, :, 1], fix_img[:, :, 2]  
print(np.array(fix_img))
```



```
...  
[ 45  45  47]  
[ 45  45  47]  
[ 45  45  47]]  
  
[[223 195 183]  
 [226 198 186]  
 [236 208 196]  
 ...  
 [ 45  45  47]  
 [ 45  45  47]  
 [ 45  45  47]]  
  
[[217 189 177]  
 [222 194 182]  
 [236 208 196]  
 ...  
 [ 47  47  49]  
 [ 47  47  49]  
 [ 47  47  49]]]
```



# METODE LIGHTNESS

```
▶ fix_img = np.max(fix_img, axis = -1, keepdims = 1)/2 + np.min(fix_img, axis = -1, keepdims = 1)/2  
  
plt.axis('off')  
plt.imshow(fix_img, cmap= 'gray')  
plt.savefig('metode lightness.jpg', bbox_inches = 'tight')
```

→ [[223 195 183]  
[226 198 186]  
[236 208 196]  
...  
[ 45 45 47]  
[ 45 45 47]  
[ 45 45 47]]  
  
[[217 189 177]  
[222 194 182]  
[236 208 196]  
...  
[ 47 47 49]  
[ 47 47 49]  
[ 47 47 49]]]



# METODE AVERAGE

```
gray_img = np.mean(fix_img, axis = 2)
print(np.array(gray_img))

plt.axis('off')
plt.imshow(gray_img, cmap = 'gray')
plt.savefig('Metode Avarage.jpg', bbox_inches = 'tight')
```

```
[[229.5 229.5 229.5 ... 128. 127. 125. ]
 [229.5 229.5 229.5 ... 126. 125. 123. ]
 [229.5 229.5 229.5 ... 123. 122. 121. ]
 ...
 [214. 211. 214. ... 46. 46. 46. ]
 [203. 206. 216. ... 46. 46. 46. ]
 [197. 202. 216. ... 48. 48. 48. ]]
```



# METODE LUMINOSITY

```
▶ lumi_img = (0.2126*R) + (0.7152*G) + (0.0722*B)  
print(lumi_img)  
  
plt.axis('off')  
plt.imshow(lumi_img, cmap = 'gray')  
plt.savefig('Metode Luminosity.jpg' , bbox_inches='tight')
```

```
→ [[189. 189. 190. ... 89. 92. 92.]  
 [189. 189. 190. ... 87. 87. 87.]  
 [190. 190. 190. ... 85. 85. 85.]  
 ...  
 [153. 152. 154. ... 156. 156. 156.]  
 [155. 153. 154. ... 158. 158. 158.]  
 [154. 153. 154. ... 159. 158. 158.]]
```



# METODE WEIGHTED AVERAGE

```
▶ wavr_img = (0.299*R) + (0.587*G) + (0.114*B)  
print(wavr_img)  
  
plt.axis('off')  
plt.imshow(wavr_img, cmap = 'gray')  
plt.savefig('Metode weighted Average.jpg', bbox_inches = 'tight')
```

```
➡ [[229.918 229.918 229.918 ... 128.313 127.313 125.313]  
 [229.918 229.918 229.918 ... 126.313 125.313 123.313]  
 [229.918 229.918 229.918 ... 123.313 122.313 121.313]  
 ...  
 [213.004 210.004 213.004 ... 45.228 45.228 45.228]  
 [202.004 205.004 215.004 ... 45.228 45.228 45.228]  
 [196.004 201.004 215.004 ... 47.228 47.228 47.228]]
```



## **HASIL DISKUSI KELOMPOK**

Metode lightness merupakan metode yang paling baik dan mudah dimengerti, serta menciptakan citra grayscale yang konsisten dalam mencerminkan warna asli.

**Terima  
Kasih**