



Görüntü İşleme

Vize Ödevi

Şevval Sultan Çelik

201505030

April 1, 2024

Akademik dürüstlük beyan formu

2023-2024 güz döneminde YMM4242 kodlu Görüntü İşleme dersi için sunmuş olduğum bu raporun vize sınavına %10 olarak etki etmesini kabul ediyorum

Histogram Eşitleme

İlk çözüm Histogram eşitleme tekniği kullanılarak gerçekleştirildi. Histogram işlemi, belirli bir gruptaki verileri belirli özelliklere göre sınıflandırılarak oluşturulur. Görüntü gri formatta okunarak piksel sayıları ortaya çıkarılır. Histogram eşitleme tekniği, belirli noktalarda kümelenen renk değerlerinin ortaya çıkardığı renk bozukluğunu gidermek için kullanılan bir teknik olarak tanımlanabilir.

Pythonda histogram elde etmek için bazı fonksiyonlar mevcuttur:

-plt.hist komutuyla görüntünün histogramı alınır.

-plt.show komutuyla histogram gösterilir. -equalizeHist() komutu ile histogram eşitlemesi yapılır.



Figure 1: Histogram eşitleme ile işlenmiş resim

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
img = cv2.imread("resim.jpg", cv2.IMREAD_GRAYSCALE)
```

```

img = cv2.resize(img, (1024, 768))
# Histogram
histogramx, bin = np.histogram(img.flatten(), 256, [0, 256])
cdf = histogramx.cumsum()
cdf_normal = cdf * histogramx.max() / cdf.max()
plt.plot(cdf_normal, color="b")
plt.hist(img.flatten(), 256, [0, 256], color="r")
plt.xlim([0, 256])
plt.legend(("cdf", "histogram"), loc="upper left")
# histogram eşitleme uygula
esit = cv2.equalizeHist(img)
cv2.imshow("histogram resmi", esit)
cv2.waitKey(0)
cv2.destroyAllWindows()

```

Contrast Limited Adaptive Histogram Equalization (CLAHE)

İkinci çözümde Contrast Limited Adaptive Histogram Equalization (CLAHE) tekniği kullanılarak gerçekleştirildi. CLAHE kontrast sınırlı adaptif histogram eşitleme olarak tanımlanır. Bu teknikte görüntü bölgelere ayrılır, her bir görüntü için ayrı histogram eşitleme işlemi gerçekleştirilir. Bir bölgedeki parlaklık değişiminden diğer bölgeler etkilenmeksizin düzeltme uygulanır. Bölgesel bir düzeltme gerçekleştirildiğinden görüntü daha doğal gözükür.



Figure 2: CLAHE ile işlenmiş resim

```

# CLAHE
import cv2
import matplotlib.pyplot as plt
img = cv2.imread("resim.jpg", 0)
claheTek = cv2.createCLAHE(clipLimit=2.0, tileGridSize=(8, 8))
# CLAHE işlemi
clahe_image = claheTek.apply(img)

```

```
plt.figure(figsize=(12, 12))
plt.subplot(221)
plt.title("Orjinal resim")
plt.imshow(img, cmap="gray")
plt.subplot(222)
plt.title("Histogram - Orjinal resim")
plt.hist(img.ravel(), bins=256, range=[0, 256], color="black", alpha=0.7)
plt.subplot(223)
plt.title("CLAHE işlenmiş Image")
plt.imshow(clahe_image, cmap="gray")
plt.subplot(224)
plt.title("Histogram - CLAHE Uygulaması")
plt.hist(clahe_image.ravel(), bins=256, range=[0, 256], color="black", al
plt.show()
```

Sonuç olarak CLAHE tekniği ile elde ettiğim görüntü istenilen niteliklere bakıldığında histogram eşitleme tekniğine göre daha iyi bir çıktı vermiştir.