

RLE Uygulanarak Veri Sıkıştırılması

Recep Furkan Koçyiğit

22501048

furkan.kocigit@std.yildiz.edu.tr

Bilgisayar Mühendisliği Bölümü

Elektrik Elektronik Fakültesi, Yıldız Teknik Üniversitesi

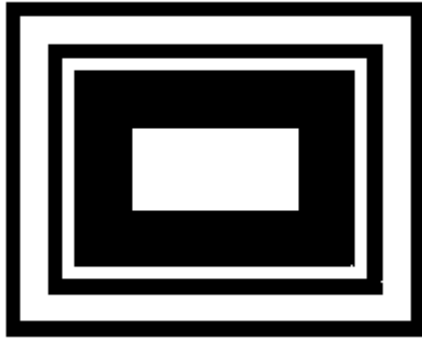
Özet

Bu ödevde, BitMap formatında siyah beyaz, gri ve RGB olmak üzere 3 farklı resim RLE (Run Length Encoding) algoritması ile satır-satır, sütun-sütun ve zigzag olarak okunup sıkıştırılacaktır. Daha sonrasında sıkıştırılan dosya açılarak doğru bir şekilde sıkıştırma olduğu teyit edilecektir. Resimler arasında ve okuma yöntemleri sıkıştırma oranı kullanılarak karşılaştırılacaktır. Bu sistemin gerçekleştirilmesi için programlama dili olarak Python kullanılmıştır. En iyi sonuç siyah beyaz resimde satır-satır okuma ile elde edilmiştir.

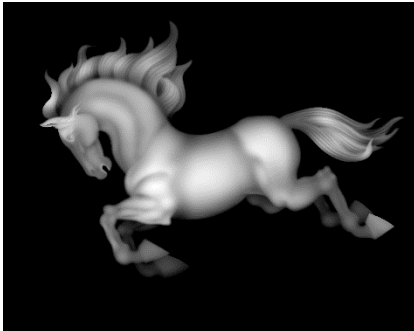
Giriş

Ödev kapsamında aşağıdaki resimler kullanılmıştır:

1. Siyah-Beyaz Resim:



2. Gri Tonlamalı Resim:



3. Renkli Resim:



Yöntem

Bu sistem aşağıdaki ana modüllerden oluşmaktadır:

1. **Resmin Okunması:** Sıkıştırılacak resmin matris olarak okunması.
2. **Matristen Diziye Zigzag Dönüşüm:** Resme ait bilgileri tutan matrisin zigzag bir şekilde okunup diziye doldurulması.
3. **Matristen Diziye Satır Bazlı Dönüşüm:** Resme ait bilgileri tutan matrisin satır-satır okunup diziye doldurulması.
4. **Matristen diziye sütun bazlı dönüşüm:** Resme ait bilgileri tutan matrisin sütun-sütun okunup diziye doldurulması.
5. **Diziden Matrise Zigzag Dönüşüm:** Zigzag olarak okunup sıkıştırılan dosyanın tekrar eski formuna getirilmesi.
6. **Diziden Matrise Satır Bazlı Dönüşüm:** Satır-satır olarak okunup sıkıştırılan dosyanın tekrar eski formuna getirilmesi.
7. **Diziden Matrise Sütun Bazlı Dönüşüm:** Sütun-sütun olarak okunup sıkıştırılan dosyanın tekrar eski formuna getirilmesi.
8. **Siyah-Beyaz Resimlere RLE Uygulanması:** Siyah-beyaz resimlere RLE uygulanırken tek bit yeterli olduğundan özelleştirilmiştir.

9. **RLE Uygulanan Siyah-Beyaz Resimlerden Orijinal Hale Dönüştürülmesi:** RLE uygulanan siyah-beyaz resimlerden orijinal hale dönüştürülürken tek bit yeterli olduğundan özelleştirilmiştir.

10. **Renkli ve Gri Tonlamalı Resimlere RLE Uygulanması:** 255 renk değerine sahip değerler 254 olarak işaretlenmiş ve RLE algoritması uygulanırken 255 değeri bir bayrak olarak kullanılmıştır. Kendinden sonraki değer kaç adet olduğu ve ondan sonraki değer ise hangi renk değeri olduğu bilgisini tutmaktadır.

11. **RLE Uygulanan Renkli ve Gri Tonlamalı Resimlerden Orijinal Hale Dönüştürülmesi:** RLE işleminden orijinal resme dönüştürmek için yazılmıştır.

Uygulama

Siyah-beyaz, gri tonlamalı ve renkli resimlere, satır-satır, sütun-sütun ve zigzag olarak RLE algoritması uygulandığında sonuçlar aşağıdaki gibi olmuştur.

Comprasion Ratio			
	Black&White	Gray Scale	Color Table
Row-Row Rotate	0.533	0.935	3.17
Column-Column Rotate	4.763	1.932	3.047
Zigzag Rotate	4.719	1.933	3.046

Tabloya bakıldığında sıkıştırma oranı olarak birden düşük iki farklı değer gözükmemektedir. Siyah-Beyaz ve gri tonlamalı resimlere satır-satır RLE algoritması uygulandığında başarılı sıkıştırma oranları elde edilmiştir. Bir diğer göze çarpan şey ise sütun-sütun ve zigzag işleme olarak okunan resimlerdeki sıkıştırma oranının başarısız olmasıdır. Yani, sıkıştırma işlemi yapılmadan daha az yer kaplamaktadır. Siyah-beyaz resimde sütun bazlı ve zigzag olarak okumanın satır bazlı olarak okumaya göre 9 kat başarısız olduğu görülmektedir.

Orijinal dosya bilgileri aşağıdaki gibidir:

Ad	Tarih	Tür	Boyut
colortable	21.10.2023 15:14	BMP Dosyası	801 KB
grayscale	22.10.2023 11:41	BMP Dosyası	2.303 KB
monochrome	22.10.2023 16:35	BMP Dosyası	7 KB

Sıkıştırma işlemi yapıldıktan sıkıştırılmış dosyalara ait bilgiler aşağıdaki gibidir:

Ad	Tarih	Tür	Boyut
colortable_col_by_col	22.10.2023 18:04	RLE File	2.440 KB
colortable_row_by_row	22.10.2023 17:42	RLE File	2.539 KB
colortable_zigzag	22.10.2023 18:04	RLE File	2.440 KB
grayscale_col_by_col	22.10.2023 15:14	RLE File	4.448 KB
grayscale_row_by_row	22.10.2023 15:14	RLE File	2.152 KB
grayscale_zigzag	22.10.2023 15:14	RLE File	4.450 KB
monochrome_col_by_col	22.10.2023 14:21	RLE File	32 KB
monochrome_row_by_row	22.10.2023 14:21	RLE File	4 KB
monochrome_zigzag	22.10.2023 13:54	RLE File	31 KB

Sıkıştırılan dosyalar açıldıktan sonra aşağıdaki gibidir:



Sonuç

Siyah-beyaz resim için satır-satır okunarak RLE algoritması uygulanması ile elde edilen sıkıştırma oranının en iyi sonucu almasının nedeni, resmin satır olarak daha fazla sayıda tekrar eden aynı bitleri tutması olarak söylenebilir. Resimdeki ayrıntı arttıkça RLE algoritması ile sıkıştırma işleminin başarısının düştüğü görülmüştür. Aynı zamanda resme RLE uygulanırken okuma yönteminin sıkıştırma başarısındaki etkisi görülmüştür. RGB resimlerde sıkıştırma işleminin orijinal resimden yaklaşık üç kat daha fazla yer kapladığı, siyah-beyaz resimde ise okuma yönteminin ne kadar fazla etki edebileceği görülmüştür. Genel olarak bakıldığında RLE algoritmasının detay içeren ve çok sayıda tekrar eden bilgi içermeyen verilerde başarısız sonuçlar üreteceği söylenebilir.