

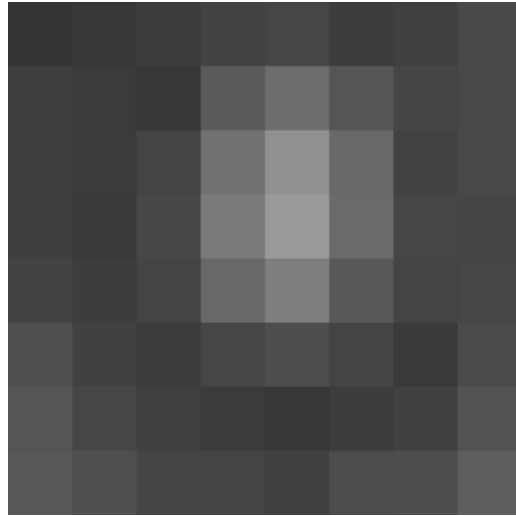
# ELE 474 Projesi

Güz 2023

Bu proje 2 kısımdan oluşmaktadır. İlk kısımda klasik JPEG encoder (kodlayıcı) ve decoder (kod çözücü) geliştireceksiniz. Aşağıdaki açıklamalarda basit düzeyde JPEG encoder ve decoder'ın basit manada nasıl yapılacağı anlatılacaktır. Ayrıntılı bilgi için konuyu anlatan <https://en.wikipedia.org/wiki/JPEG> web sitesindeki JPEG codec example'daki bilgileri kullanabilirsiniz. Bunun dışındaki bilgileri sizin araştırmanız beklenmektedir. Bunun yanı sıra burada 512x512 çözünürlüğündeki Lena.bmp grayscale resim kullanılacaktır. Bu resmi internetten bulabilirsiniz

- 1) (60 puan) JPEG, Joint Pictures Experts Group, resim sıkıştırma için kullanılan bir algoritmadır. Bu algoritmanın sıkıştırma kısmı encoder, sıkıştırılan resimden tekrar resim oluşturma kısmı decoder olarak bilinir. Encoder de sırasıyla şu işlemler gerçekleşir.

Öncelikle resim 8x8lik bloklara bölünür.



Bunun içindeki bir blok aşağıdaki gibidir.

52	55	61	66	70	61	64	73
63	59	55	90	109	85	69	72
62	59	68	113	144	104	66	73
63	58	71	122	154	106	70	69
67	61	68	104	126	88	68	70
79	65	60	70	77	68	58	75
85	71	64	59	55	61	65	83
87	79	69	68	65	76	78	94

Daha sonra DCT kullanılacaktır. Bunun için önce wiki de anlatıldığı gibi ortalama çıkartılır. Daha sonra aşağıdaki formül kullanılıp DCT alınır. Bu formül uygulandıktan sonra ilgili 8x8 bloğunu DCT si aşağıdaki resim gibi olur.

$$G_{u,v} = \frac{1}{4} \alpha(u) \alpha(v) \sum_{x=0}^7 \sum_{y=0}^7 g_{x,y} \cos \left[ \frac{(2x+1)u\pi}{16} \right] \cos \left[ \frac{(2y+1)v\pi}{16} \right]$$

$$G = \begin{matrix} & \begin{matrix} u \\ \longrightarrow \end{matrix} & \begin{bmatrix} -415.38 & -30.19 & -61.20 & 27.24 & 56.12 & -20.10 & -2.39 & 0.46 \\ 4.47 & -21.86 & -60.76 & 10.25 & 13.15 & -7.09 & -8.54 & 4.88 \\ -46.83 & 7.37 & 77.13 & -24.56 & -28.91 & 9.93 & 5.42 & -5.65 \\ -48.53 & 12.07 & 34.10 & -14.76 & -10.24 & 6.30 & 1.83 & 1.95 \\ 12.12 & -6.55 & -13.20 & -3.95 & -1.87 & 1.75 & -2.79 & 3.14 \\ -7.73 & 2.91 & 2.38 & -5.94 & -2.38 & 0.94 & 4.30 & 1.85 \\ -1.03 & 0.18 & 0.42 & -2.42 & -0.88 & -3.02 & 4.12 & -0.66 \\ -0.17 & 0.14 & -1.07 & -4.19 & -1.17 & -0.10 & 0.50 & 1.68 \end{bmatrix} \\ \begin{matrix} \downarrow \\ v. \end{matrix} \end{matrix}$$

Bu aşamadan sonra quatization (nicemleme) gerçekleşecek. Bu da 8x8 lik DCT matrisini aşağıdaki 8x8 lik Quantization matrisini böleceksiniz. Wiki de nasıl yapılacağı anlatılmaktadır. Buradaki bilgi kayıplıdır. Kullanılan matristeki quantization miktarına göre resim kalitesindeki bozulma artar. Diğer taraftan sıkıştırma miktarı da artar. 2. Kısım bu matrisin ayarlanması ile ilgili olacaktır.

$$Q = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}.$$

Son aşamada wiki de anlatıldığı gibi bir zig zag scanning yapılacak ve ardından Huffman coding uygulanacaktır. Bunun nasıl yapıldığı sınıfta anlatılacaktır. İlgili Huffman tabloları internetten araştırılacaktır. Bu aşamadan sıkıştırılmış resmi elde etmiş olacaksınız. Daha sonra buradaki aşamaları tersten gerçekleştireceksiniz. Yani Huffman decoding yapacaksınız. Sonra bölmek için kullandığınız quantization matrisleri ile dequantization yapacaksınız. En sonunda Inverse DCT ile resmi elde edeceksiniz ve PSNR ile kaliteyi ölçeceksiniz.

PSNR'ın nasıl hesaplandığını internetten araştırarak öğrenebilirsiniz.

- 2) (40 Puan) Bu kısımda yukarıdaki JPEG encoderini biraz deęiřtireceksiniz. Yukarıdaki iřlemde kullandığımız quantization matris tek bir matris olmayacak. Bu matrisin sabit bir katsayı ile arpılmasıyla farklı miktarda quantization yapabilirsiniz. arpma miktarı arttıka sıkıřtırma miktarı artar. Buradaki amacımız adaptive quantization kullanarak aynı miktar sıkıřtırmayla olabildięince iyi kaliteli resim elde etmektir. Bu kavram rate-distortion optimization olarak gemektedir. Bunu internetten arařtırabilirsiniz.

Bu kısımda elde edilen sonular sıralanacak ve 40 puan buna gre verilecektir. Projeyi tek yapanlar %10 ek puan alacaktır. Projeyi Python da yapanlar da %10 ek puan alacaktır. Bu iki durum beraber olduęunda sadece %10 bonus olur. Python ile yapıp matlab kullanılırsa bonus olmaz.

Bařarılar