

# VERİ SIKIŞTIRMA

## Giriş-1

Prof.Dr. Banu DİRİ

# Kaynaklar

- Introduction to Data Compression, Khalid Sayood
- Digital Compression for Multimedia, J. Gibson, T. Berger, T. Lookabaugh
- Data Compression, D.Salomon

## Değerlendirme

Vize	%20
Seminer	%20
Ödev	%20
Final Projesi	%40

### Ödev

Bir veya iki adet ödev verilecektir. Ödev teslim süresi verildiği tarihten iki hafta sonra teslim edilecektir.

### Seminer

Veri Sıkıştırma alanı ile ilgili bir makale seçilecek  
Aralık ayında power point sunum eşliğinde anlatılacak  
Hazırlanılan konu iee formatında bir rapor halinde yazılarak sunum günü teslim edilecek.

### Proje

Veri Sıkıştırma alanı ile ilgili bir konu seçilecek  
Final haftasında kod üzerinden anlatımı yapılarak, projenin çıktısı gösterilecek  
Final sınav evrakı olarak proje raporu hazırlanarak [online.yildiz.edu.tr](http://online.yildiz.edu.tr) üzerinden sisteme yüklenecek

# Konular

- Veri Sıkıştırma Nedir ?
- Hata Bulma ve Düzeltme
- Temel Teknikler  
(Sezgisel Yöntemler, Run Length Coding, Move-To-Front)
- Statik Kodlar  
(Prefix, Colomb)
- İstatistiksel Yöntemler  
(Shannon Fano, Huffman ve Türevleri, MNP ve Türevleri, Aritmetik, PPM Kodlama)

- Sözlüksel Yöntemler  
(LZ ve Türevleri)
- Görüntü Sıkıştırma

## Veri Sıkıştırma Nedir ?

Bir input data stream (source stream, original raw data) yerden ve zamandan kazanmak amacı ile verinin boyutunu küçülterek bir başka data stream (output, compressed) çevirme işlemidir.

- Fixed Size Code
- Redundancy
- Variable Size Code

**Fixed Size Code** : Tüm sembollerin aynı bit uzunluğunda kodlanması

**Variable Size Code** : Sembollerin farklı bit uzunlukları ile kodlanması. Kullanım sayısı fazla olan semboller daha az bit ile kodlanırken, kullanım sayısı az olan semboller daha çok bit ile kodlanır.

**Redundancy (Artıklık)** : Sıkıştırılmış veri içerisindeki fazlalık

## Veri Sıkıştırırmada Kullanılan Kavramlar

- **Compressor / Encoder / Kodlayıcı**  
Input stream ham veriyi sıkıştıran ve sıkıştırılmış veri ile output stream oluşturan algoritma
- **Decompressor / Decoder / Kod Çözücü**  
Kodlayıcının yaptığı işi ters yönde yapar
- **Codec**  
Encoder ve decoder'ı birlikte tanımlar
- **Unencoded / Raw Data / Original Data**  
Orijinal giriş verisi
- **Encoded / Compressed**  
Sıkıştırılmış veri

- **Non-adaptive Compression Method**

Parametre ve işlemler belli tip veriler için geçerlidir (facsimile compression)

- **Adaptive Compression**

İşlenmemiş her türlü veri üzerinde işlem yapar. Parametre ve işlemler değiştirilebilir

- **Semi-Adaptive Compression**

İki adımda işlemler gerçekleştirilir. Önce veri üzerinden geçilerek tüm semboller için kullanım sıklıkları çıkarılır, sonrasında sıkıştırma işlemi gerçekleştirilir

- **Lossy / Lossless Compression**

Lossy (kayıplı) sıkıştırmada sıkıştırılmış veri açıldığında orijinal veriye dönülemez. Ses ve görüntü sıkıştırmada kullanılır

Lossless (kayıpsız) sıkıştırmada sıkıştırılmış olan veri açıldığında orijinal halindedir. Doküman sıkıştırmada kullanılır

- **Symmetrical Compression**

Compresor ve decompresor aynı algoritmayı ters yönde kullanırlar

- **Asymmetric Compression**

Compresor ve decompresor farklı algoritma kullanır



## Soru

ARŞİV uygulamaları

Kodlayıcının karışık ve uzun sürdüğü buna karşılık kod çözücünün basit ve hızlı çalıştığı sıkıştırma algoritmalarının tercih edildiği bir uygulama alanı ne olabilir?

## Soru

Backup uygulamaları

Kodlayıcının basit ve kısa sürdüğü buna karşılık kod çözücünün karışık ve yavaş çalıştığı sıkıştırma algoritmalarının tercih edildiği bir uygulama alanı ne olabilir?

## Soru

İyi bir sıkıştırma performansına sahip bir algoritmanın önemli olduğu fakat kodlama ve kod çözme hızının önemli olmadığı bir duruma örnek veriniz.

- **Stream Mode / Block Mode**

Stream mode : Veri byte byte çekilir ve üzerinde işlem yapılır

Block mode : Veri bloklar halinde çekilir ve her blok üzerinde farklı algoritma çalıştırılabilir

- **Bits per Chars**

Sıkıştırılan her bir sembolün ortalama bit uzunluğu

- **Bits per Pixels**

Sıkıştırılan her bir pixel in ortalama bit uzunluğu

- **Sıkıştırma Oranı**

$\text{compression\_ratio} = \text{size of the output stream} / \text{size of the input stream}$

- **Sıkıştırma Performansı (yüzdesel)**

$100 * (1 - \text{compression\_ratio})$

- **Corpus**

Calgary, Canterbury, UCI

## ■ Root Mean Square Error ( $e_{RMS}$ )

$I(r,c)$  orijinal resim,  $I'(r,c)$  decompress edilmiş resim

$$\text{error}(r,c) = I'(r,c) - I(r,c)$$

Image size  $M \times N$

Sıkıştırıldıktan sonraki toplam hata

$$\text{total error} = \sum_{r=0}^{M-1} \sum_{c=0}^{N-1} [I'(r,c) - I(r,c)]$$

$$\text{Root Mean Square Error } e_{RMS} = \sqrt{\frac{1}{N^2} \sum_{r=0}^{M-1} \sum_{c=0}^{N-1} [I'(r,c) - I(r,c)]^2}$$

## ■ Signal Noise Ratio (SNR)

$$SNR_{RMS} = \sqrt{\frac{\sum_{r=0}^{M-1} \sum_{c=0}^{N-1} [I'(r,c)]^2}{\sum_{r=0}^{M-1} \sum_{c=0}^{N-1} [I'(r,c) - I(r,c)]^2}}$$

$$SNR_{PEAK} = \text{PSNR} = 10 \log_{10} \frac{(L-1)^2}{\frac{1}{N^2} \sum_{r=0}^{M-1} \sum_{c=0}^{N-1} [I'(r,c) - I(r,c)]^2}$$

L : renk sayısı