ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Multimedia Technology

Section # 10: JPEG Coding

Instructor: George Xylomenos

Department: Informatics

Contents

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Introduction

Class: Multimedia Technology, **Section # 10:** JPEG Coding

Instructor: George Xylomenos, **Department:** Informatics

What is JPEG?

- Joint Photographic Experts Group
  - Joint between ISO and CCITT (now, ITU)
- Result of a competition
  - The committee set the specifications
  - 10 proposals were selected
  - The best one was modified
  - Similar process followed for MPEG standards
    - Industry progressively plays a bigger role

JPEG goals

- Application area
  - Continuous tone still images
    - Color or grayscale
  - Targeting natural images
    - As opposed to drawings
  - Basic format for still cameras
    - Ubiquitous in digital photos
  - Spread via the Internet

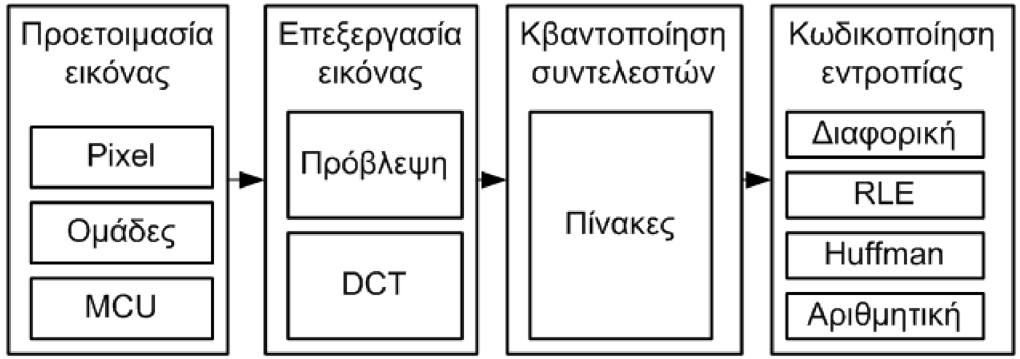
Coding requirements

- Independent of resolution
- Independent of aspect ratio
- Independent of color encoding
- Random content
  - Random pixel statistics
- User tunable parameters
  - Tradeoff between quality and efficiency

JPEG modes

- Lossy sequential
  - Baseline DCT
- Lossy extended
  - Progressive DCT
  - Simple extension of lossy sequential
- Lossless: lower compression
- Hierarchical: multiple resolutions

Coding flow



- JPEG has a set of alternative options
  - Each mode uses specific options
  - Prediction or DCT
  - Optional quantization
  - Differential or RLE
  - Huffman or arithmetic

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Image preparation

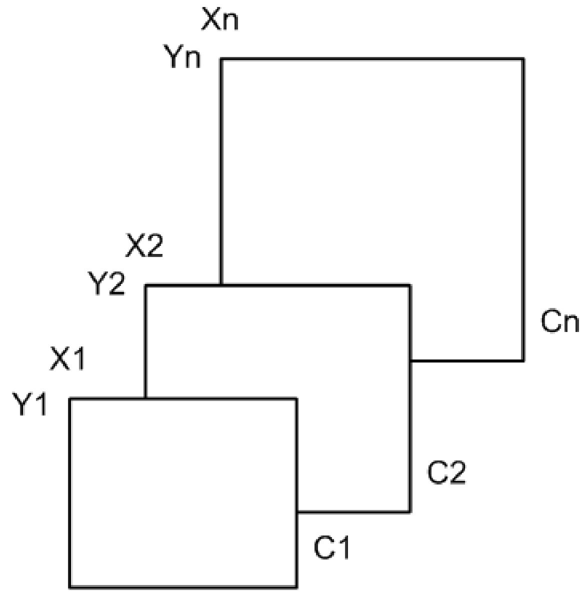
Class: Multimedia Technology, **Section # 10:** JPEG Coding

Instructor: George Xylomenos, **Department:** Informatics

Image model (1 of 5)

- JPEG supports many different models
  - 1 to 255 components or planes (C_i)
    - RGB colors: one plane per color
    - YUV/YIQ signals: one plane per signal
    - Grayscale: one component only
    - Mostly YUV or YIQ
  - Each C_i may have different X_i and Y_i
    - Appropriate for subsampling
    - Not random values!

Image model (2 of 5)



- Fixed bits/pixel per component in each mode
  - 8 for sequential, 8 for 12 extended
    - [0,255] or [0,4095]
  - 2 to 16 for lossless

Image model (3 of 5)

- (X,Y) of image = (max(X_i), max(Y_i))
- Example: image with 4 components
    - Component 1: X_1=48, Y_1=32
    - Component 2: X_2=48, Y_2=16
    - Component 3: X_3=24, Y_3=32
    - Component 4: X_4=24, Y_4=16
  - Max(X_i)=48, Max(Y_i)=32
  - Min(X_i)=24, Min(Y_i)=16

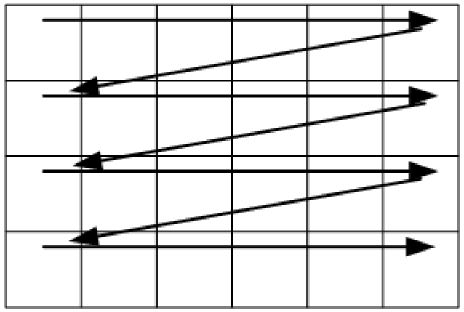
Image model (4 of 5)

- Relative subsampling ratios
  - We start with the lowest resolution
    - X and Y may be from different components
  - Each components has multiples of that
    - The reverse of subsampling
    - Multipliers instead of dividers
- Example: image with 4 components
  - (H_i,V_i) with 1 to 4 for H_i και V_i

Image model (5 of 5)

- Example: image with 4 components
  - $\text{Max}(H_i)=48/24=2$, $\text{Max}(V_i)=32/16=2$
  - $H_1=H_2=2$, $H_3=H_4=1$, $V_1=V_3=2$, $V_2=V_4=1$
    - Component 1: $X_1=2\text{x}24=48$, $Y_1=2\text{x}16=32$
    - Component 2: $X_2=2\text{x}24=48$, $Y_2=1\text{x}16=16$
    - Component 3: $X_3=1\text{x}24=24$, $Y_3=2\text{x}16=32$
    - Component 4: $X_4=1\text{x}24=24$, $Y_4=1\text{x}16=16$

Data ordering (1 of 4)

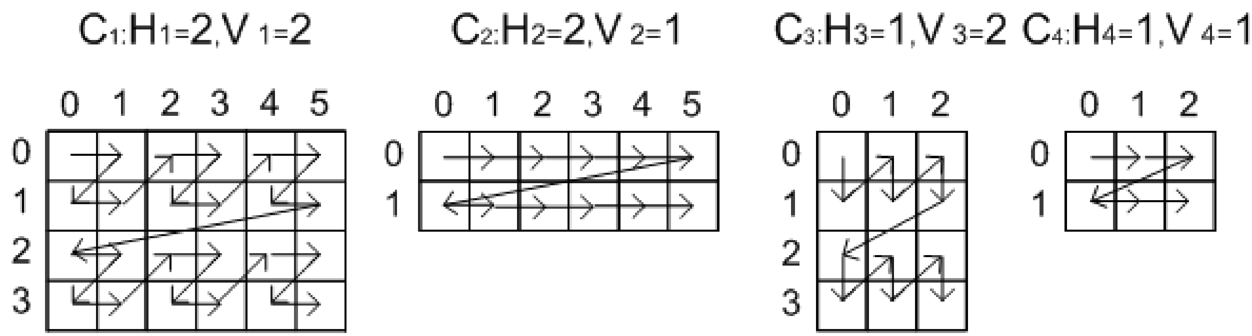


- Coding data units
  - Data which are coded together
  - Lossless mode: one pixel
  - All lossy modes: 8x8 pixels
- Non interleaved ordering
  - Scans units from left to right, top to bottom

Data ordering (2 of 4)

- Interleaved ordering
  - Each component has its own coding units
  - Minimum Coding Unit (MCU)
    - All units corresponding to the same area
    - Group of coding units across components
- Fixed resolution components
  - MCU has one unit per component
  - Processed in component order

Data ordering (3 of 4)



- Variable resolution components
  - Each MCU corresponds to the same area
  - Different number of units per component
  - Ordered per component first
  - Each MCU has H_i x V_i units for C_i

Data ordering (4 of 4)

- MCU limitations
  - Up to four components
  - Up to 10 units per MCU
- Why the complexity of interleaved ordering?
  - Assume an image in YUV
  - The screen however is RGB
  - The YUV->RGB conversion takes by MCU
    - Need to upsample UV and combine with Y

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

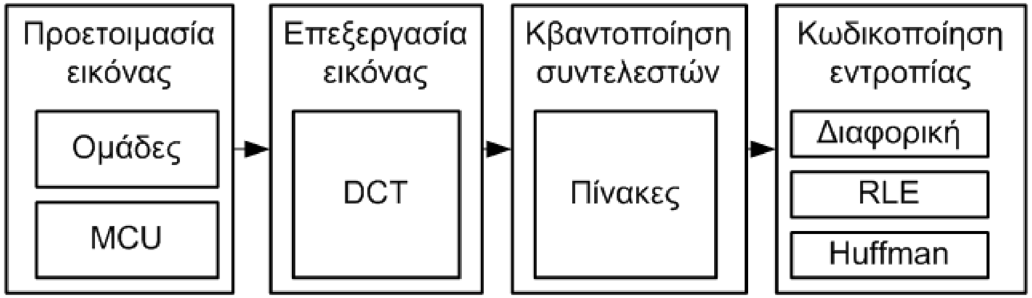
ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Image transform

Class: Multimedia Technology, **Section # 10:** JPEG Coding

Instructor: George Xylomenos, **Department:** Informatics

Image processing (1 of 4)



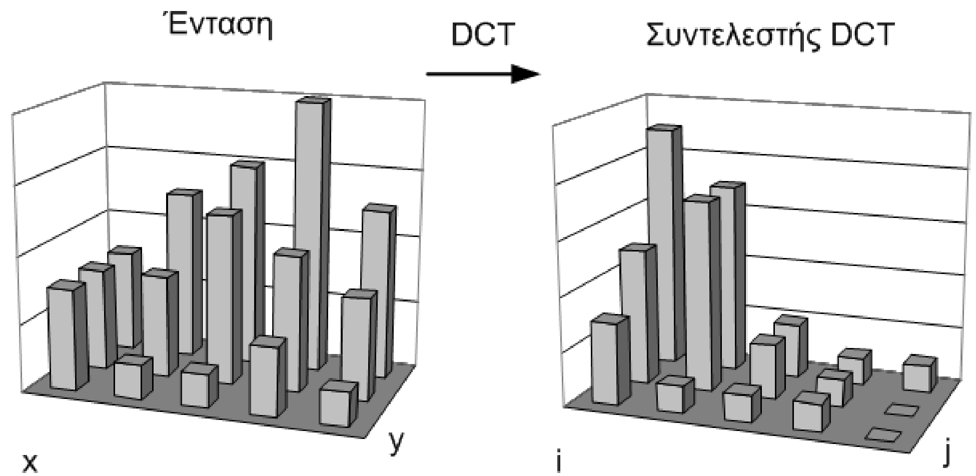
- Input
  - Units of 8x8 pixels (intensities)
    - Each unit transform independently
    - Their order does not matter
- Output
  - Units of 8x8 coefficients

Image processing (2 of 4)

$$F[i,j] = \frac{1}{4} C(i) C(j) \sum_{x=0}^{7} \sum_{y=0}^{7} P[x,y] \cos\frac{(2x+1)i\pi}{16} \cos\frac{(2y+1)j\pi}{16}$$

- Forward Discrete Cosine Transform (FDCT)
  - [0,255] -> [-128,127] via subtraction
  - Initially: P[x,y] with x,y= [0,7] (64 values)
  - Finally: F[i,j] with i,j= [0,7] (64 values)
- The cosines do not depend on P[x,y]
  - Calculated in advanced and stored in table
- C(x)=1/sqrt(2) when x = 0, 1 when x <> 0

Image processing (3 of 4)



- Coefficient F[0,0]: DC coefficient
  - Average value of entire data unit
- Others: AC coefficients
- Highest values concentrated around the DC

Image processing (4 of 4)

$$P[x,y] = \frac{1}{4}\sum_{i=0}^{7}\sum_{j=0}^{7} C(i)C(j)F[i,j]\cos\frac{(2x+1)i\pi}{16}\cos\frac{(2y+1)j\pi}{16}$$

- Inverse DCT (IDCT)
  - Calculates P[x,y] from F[i,j]
  - The cosines again are independent of F[i,j]
  - Calculated in advanced and stored in table
  - In theory, a lossless transform
    - In practice, rounding errors cause some distortion
    - But we have no <u>compression</u>

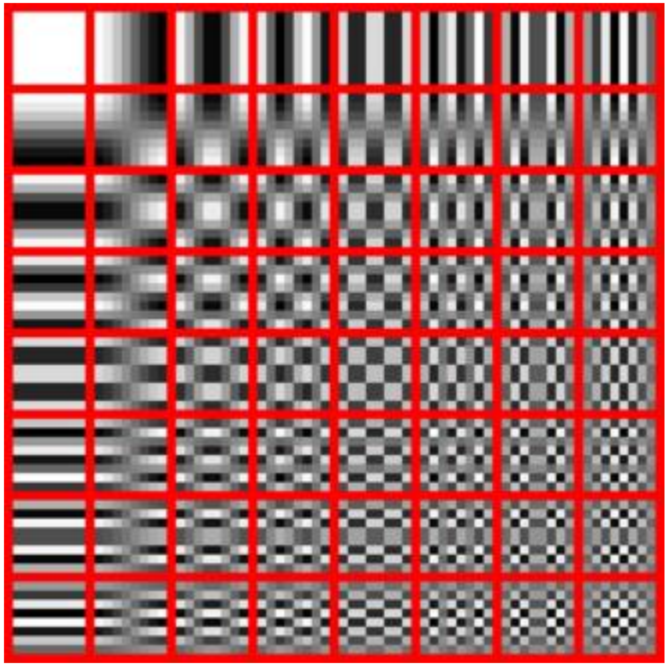
Why DCT in JPEG? (1 of 3)

- Relies on the properties of natural images
  - Large areas with similar colors
    - Each component varies slowly
    - Few drastic changes
  - Very different from computer graphics
    - Fewer colors with drastic changes
- The DC coefficient holds the mean value
  - The mean “color” of the unit

Why DCT in JPEG? (2 of 3)

- The AC coefficients encode variations
  - Slower to faster variations
  - Separately for each dimension
  - Natural images have slow variations
    - The large coefficients are close to the DC
    - Gradually, values drop
    - Many coefficients are close to zero
    - This allows for more efficient coding!

Why DCT in JPEG? (3 of 3)



- Analyzing a unit into coefficients
  - Coefficient (u,v) corresponds to block (u,v)
  - The unit is a linear combination of these blocks

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Quantization

Class: Multimedia Technology, **Section # 10:** JPEG Coding

Instructor: George Xylomenos, **Department:** Informatics

Quantization (1 of 5)

- The FDCT is not lossy
  - The IDCT fully reverses it
  - Some distortion due to rounding
- The FDCT reveals the image structure
  - It shows how much each coefficient
  - Different quantization per coefficient
    - Controlled distortion
    - Ideally, not visible to the human eye

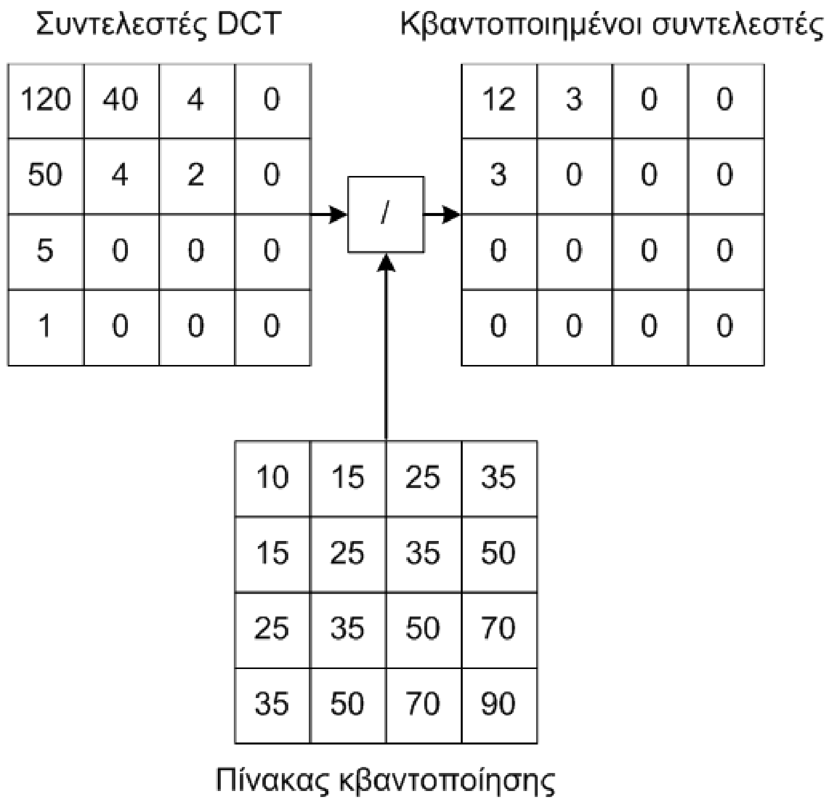
Quantization (2 of 5)

- Coefficient quantization
  - 8x8=64 quantization matrix
    - Integer 8 bit values (quantizers)
    - Same size as the coefficient matrix
  - Each coefficient is divided by its quantizer
    - Rounded to the nearest integer
  - Multiplication at the decoder
    - Using the exact same matrix

Quantization (3 of 5)

- Quantization matrix
  - Quantization is the main lossy stage
  - The JPEG standard offers some matrices
  - Each application can add its own
- Natural images: focus on DC coefficient
  - Smaller quantizers close to DC
  - Progressively larger quantizers further from DC

Quantization (4 of 5)



- Example of quantization

Quantization (5 of 5)

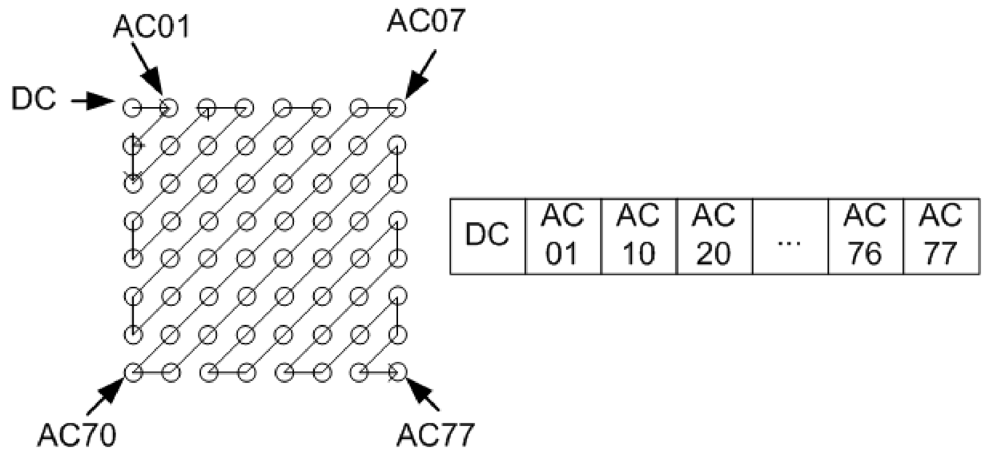
- Variable compression ratio
  - Most applications allow for variable compression
  - Usually x=1-12 (worst-best quality)
  - Multiplies the quantizer matrix
    - Actually, its inverse is used (e.g., 12/x)
  - Controls how much information we drop
  - Same multiplier used at the decoder

Entropy coding

Class: Multimedia Technology, **Section # 10:** JPEG Coding

Instructor: George Xylomenos, **Department:** Informatics

Compression (1 of 4)



- Coefficient ordering
  - We start with the DC coefficient
  - Then we zig-zag across the AC coefficients
  - The values get progressively smaller
    - Due to the higher quantizers!

Compression (2 of 4)

- DC coefficient coding
  - Differential coding from unit to unit
    - We expect small variations in the average value
- AC coefficient coding
  - Independent coding in each unit
  - We have progressively smaller values
  - Many zeroes towards the end of the sequence
- How to compress a sequence of small values?

Compression (3 of 4)

- Entropy coding for coefficients
  - Huffman-like coding is used
  - Predefined trees or application defined
- DC coefficient coding
  - Each DC codes as (bit length, value)
    - Coefficient: difference with previous DC
  - Bit length is Huffman coded
  - Coefficient value in 2's complement

Compression (4 of 4)

- AC coefficient coding
  - Similar to DC, but in triples
  - (number of zeroes, bit length, value)
    - Similar to RLE for zeros
  - Zeroes and bit length: Huffman coded
    - For more than 15 zeroes, need more symbols
  - Coefficient value: 2’s complement

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Image quality

Class: Multimedia Technology, **Section # 10:** JPEG Coding

Instructor: George Xylomenos, **Department:** Informatics

Image quality (1 of 4)

- JPEG elements can be found everywhere
  - All video codecs start with JPEG
    - H.26x, MPEG-x
  - Three basic elements
    - 8x8 units (possibly 4x4 – 16x16)
    - Transform (DCT or others)
    - Quantization (matrix or single value)
  - Hardware support for these elements

Image quality (2 of 4)

- Compression vs quality
  - Controlled by quantization multiplier
  - Causes very specific artefacts (distortion)
  - Muddy text, blocking

Image quality (3 of 4)

- Muddy text
  - Sharp text has sudden intensity changes
  - But these AC coefficients are heavily compressed
  - Shades appear inside each block

Image quality (4 of 4)

- Blocking
  - Each block is compressed around the mean (DC)
  - Edges of blocks are not continuous

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

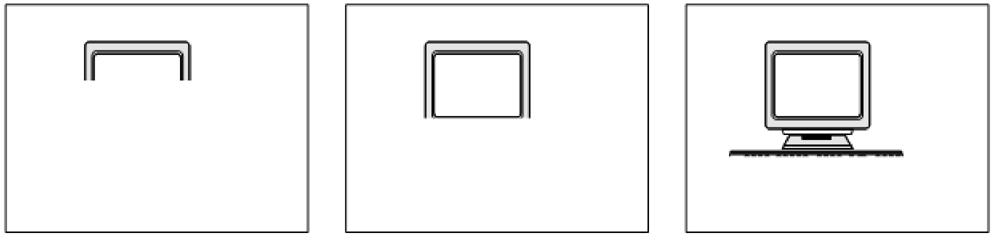
ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Other coding modes

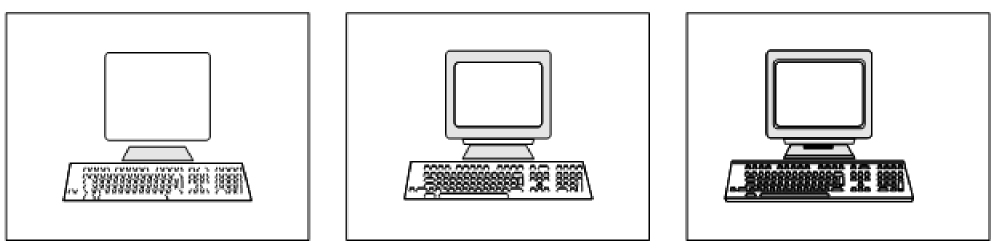
Class: Multimedia Technology, **Section # 10:** JPEG Coding

Instructor: George Xylomenos, **Department:** Informatics

Extended sequential (1 of 3)



- Minor extensions to baseline sequential
- Sequential coding
  – Image appears sequentially
- Progressive coding
  – Image progressively gets more detail



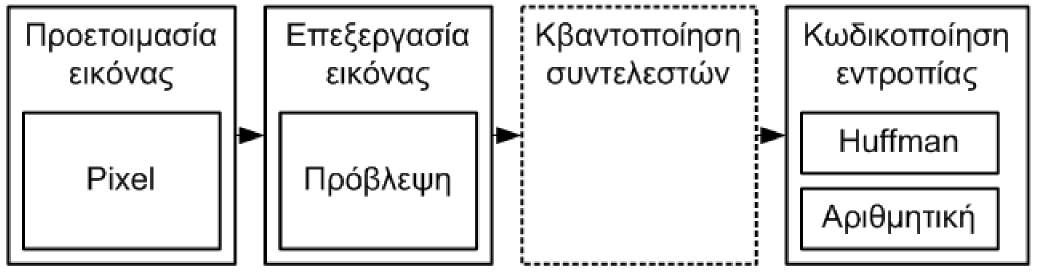
Extended lossy (2 of 3)

- Progressive coding methods
  - Spectral selection
    - First encode lower frequency coefficients
    - Each consecutive pass encodes higher frequencies
  - Progressive approximation
    - All coefficients encoded in each pass
    - First encode most significant bits
    - Each consecutive pass encodes more bits

Extended lossy (3 of 3)

- Color depth extensions
  - Allows either 8 or 12 bits per pixel (per plane)
    - Allows 36 bit color depth
- Entropy coding extensions
  - Allows arithmetic coding
    - 5-10% better compression than Huffman
    - Computationally harder
    - Initially covered by patents

Lossless mode (1 of 2)

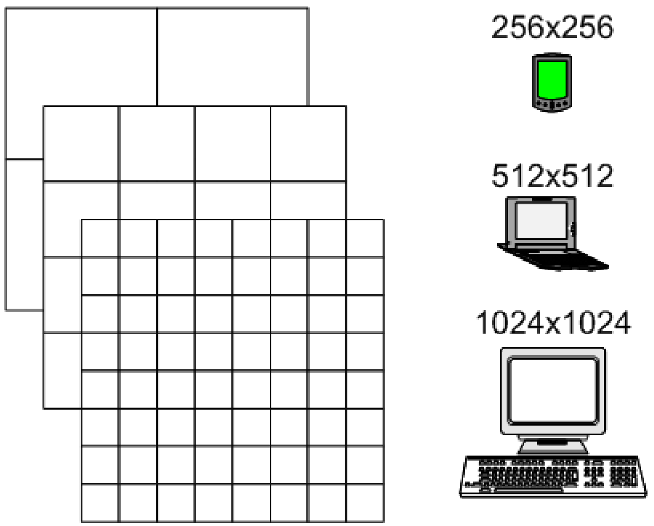


- Prediction instead of transform
  - X: current pixel
  - A: pixel left of X
  - B: pixel above of X
  - C: pixel above and left of X

Lossless mode (2 of 2)

- Coding of pixel X
  - 8 prediction modes
    - X=A, X=B+(A-C)/2, …
  - Difference from prediction
- No quantization at all
  - That is the lossy part!
- Only entropy coding
  - Huffman or arithmetic

Hierarchical mode (1 of 3)



- Multiple resolutions in the same file
  - Progressive (de)coding
  - We can decode only some resolutions
  - Targets heterogeneous devices

Hierarchical mode (2 of 3)

- Progressive: different qualities
- Hierarchical: different resolutions
- Hierarchical coding has some overhead
  - But we can get multiple resolutions
  - We can choose which one to decode
- Works in consecutive levels
  - Each level can use any coding mode
  - But we need all previous levels for decoding

Hierarchical mode (3 of 3)

- Coding process for n levels
  - Reduce the original image resolution by 2^n
  - Encode the image
  - Reduce the original image resolution by 2^{n-1}
  - <u>Decode</u> the previous level
  - Calculate the different between the two
  - Encode only the different
  - Continue up to the original resolution

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

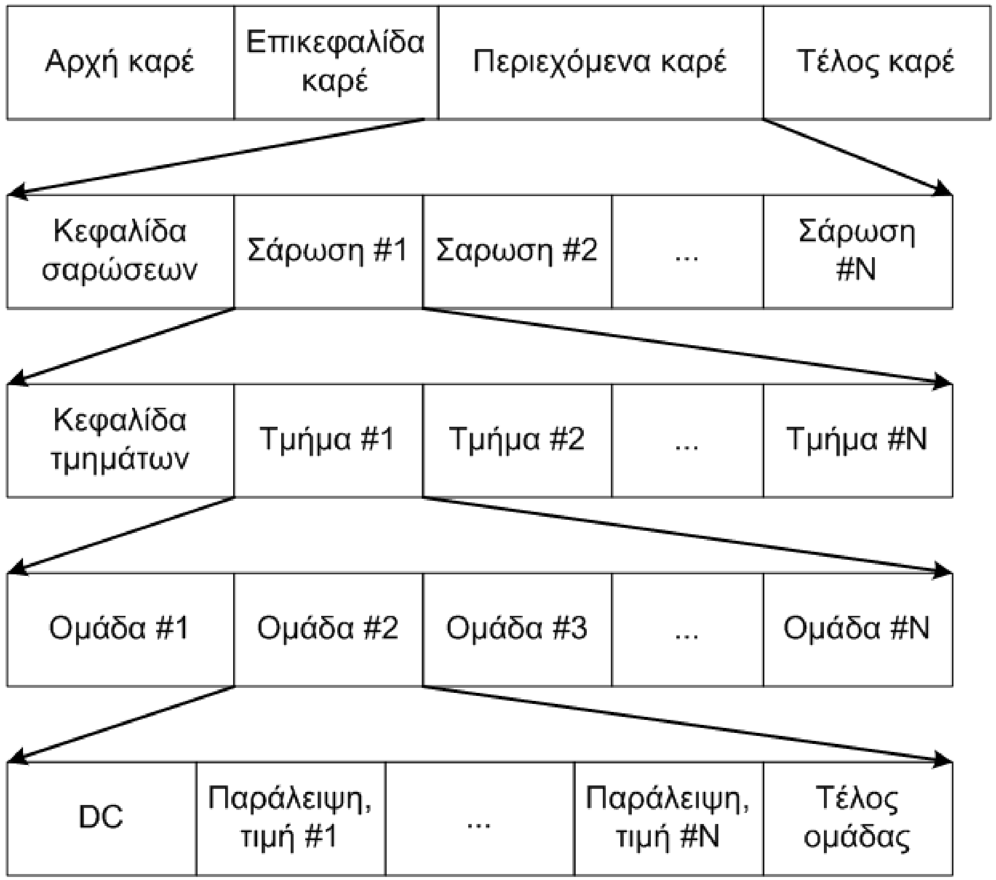
ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Bit stream

Class: Multimedia Technology, **Section # 10:** JPEG Coding

Instructor: George Xylomenos, **Department:** Informatics

JPEG bit stream (1 of 3)



- Bit stream is hierarchically structured
  - Scan = component, segment = MCU

JPEG bit stream (2 of 3)

- Full or abbreviated format
  - Full includes all tables
  - Abbreviated can drop some
    - Makes sense when only a single application is used
- Data are structured hierarchically
  - Each level starts with some parameters
  - Frame: bits / pixel, X and Y
  - Scan: Huffman tree used

JPEG bit stream (3 of 3)

- EXIF (Exchangeable image format)
  - Metadata specification
  - Common in JPEG photos
    - Photo date take (may differ from modification date!)
    - Exposure, speed, focus
    - Resolution per size unit
    - Resolution in pixels
    - Geolocation

End of Section # 10

Class: Multimedia Technology, **Section # 10:** JPEG Coding

Instructor: George Xylomenos, **Department:** Informatics