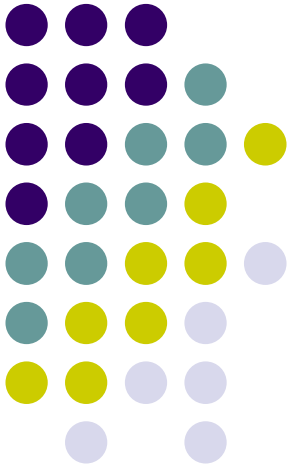
Lecture 1: Introduction & Image and Video Coding Techniques (I)

A/Prof. Jian Zhang

NICTA & CSE UNSW
COMP9519 Multimedia Systems
S2 2009
jzhang@cse.unsw.edu.au



NICTA

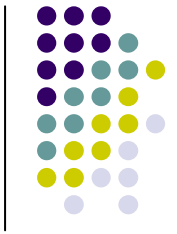
1.1 Introduction

- Our team's profile – Multimedia and Visual Communication Research Group (MMVC); NICTA
  - A/Prof. Jian Zhang http://www.cse.unsw.edu.au/~jzhang
- This course covers three parts:
  - Image/video coding technology,
  - Streaming multimedia and
  - Multimedia content description, analysis and retrieval

1.2 Course Scope & Arrangement

- The Scope of this Course:
  - Provide fundamentals of state-of-art multimedia technologies
    - Concepts
    - Principles of these technologies and,
    - Their applications
  - Provide a base of introduction to multimedia system
    - Digital audio and image/video signal coding and compression;
    - Multimedia streaming and multimedia presentation
    - Multimedia content description
    - Video structure analysis; video summarization and representation
    - Multimedia database indexing, browsing and retrieval;.

1.2 Course Scope & Arrangement – Subject Outline



- Objectives:
  - On successful completion of this subject, students will:
    - understand fundamental concepts, theory and techniques of
      - digital audio and image/video signal coding and compression;
      - multimedia streaming and multimedia presentation
      - multimedia content description
      - video structure analysis; video summarization and representation
      - multimedia database indexing, browsing and retrieval;.
    - be familiar with applications of multimedia systems and their implementations;
    - be able to apply the techniques in real applications
    - gain skills and knowledge beneficial to future work and post-graduate study in multimedia area

1.2 Course Scope & Arrangement

- Lecture 1 -- Introduction & Image and Video Coding Techniques (I)
- Lecture 2 -- Image and Video Coding Techniques (II)
- Lecture 3 -- Video Compression Standards (part 1)
  Assignment 1
- Lecture 4 -- Video Compression Standards (part 2)
- Lecture 5 -- Internet Streaming Media
- Lecture 6 -- Multimedia Presentation

1.2 Course Scope & Arrangement (Cont.)

- Lecture 7 -- Multimedia Content Description
- Lecture 8 -- Multimedia Information Retrieval (part 1)
- Lecture 9 -- Multimedia Information Retrieval (part 2)
  Assignment 2
- Tutorial – Multimedia information Retrieval
- Lecture 10 -- Multimedia Information Retrieval (part 3)
- Lecture 11 –Multimedia Signal Processing + Tutorials
- Lecture 12 -- Course Overview

1.2 Course Scope & Arrangement

- http://www.cse.unsw.edu.au/~cs9519/
  - 13 teaching weeks (12 lectures + one tutorial class)
- Consultation time:
  - Level 4 NICTA L5 Building . Every Wednesday 2-5 PM from Week 2.
- Tutors:
  - Evan Tan – lectures 1-7
  - Sakrapee (Paul) Paisitkriangkrai – lectures 8-12
- Tutorials
  - Embed the tutorials to each lecture. Only one tutoring classes in week 10.

1.2 Assessment

- Assignment 1 (30%)
- Assignment 2 (30%)
- Final Exam (40%)
  - Understand basic concepts
  - Describe concepts for problem solving

References

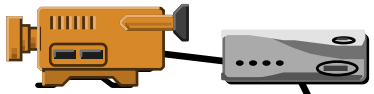
- Reference Books:
  - M. Ghanbari, Video coding: an introduction to standard codecs, 1999.
  - Barry G. Haskell, Digital video: an introduction to MPEG - 2, 1997.
  - Yun Q. Shi, Image and video compression for multimedia engineering, 2000.
  - F. Pereira, The MPEG-4 book, 2002
  - Feng D, Siu W C and Zhang H J (editor), Multimedia Information Retrieval and Management, Springer, 2003
- International Standards:
  - RTSP www.ietf.org/rfc/rfc2326.txt
  - SDP www.ietf.org/rfc/rfc2327.txt
  - RTP www.ietf.org/rfc/rfc3550.txt
  - RTP for MPEG-4 www.ietf.org/rfc/rfc3016.txt
  - XML www.w3.org/TR/REC-xml/
  - SMIL www.w3.org/TR/smil20/

1.3 Multimedia Applications -- Digital Video

- Video conference and telephony
- Multimedia communications
- Digital video Camera
- DVD/VCD
- HDTV and SDTV
- Video surveillance and security
- Video/image database
- Interactive multimedia
- Multimedia data storage and management
- Digital terrestrial and satellite TV

1.3 Multimedia Applications -- Streaming Video (MPEG-4 standard)

Content Creation → Delivery → Consumption



Live MPEG-4
Wireline Multicast
E.g. daycare, security

MPEG-4 Server
Internet Content

Wired Networks

PC

GPRS

M'Phone

IP Network

Professional MPEG-4 Encoder
Content Provider

MPEG-4
Local Servers/
Caches

EDGE

M'Phone

Live MPEG-4
Wireless Camera
Error resilient encoder

3G Network

M'Phone

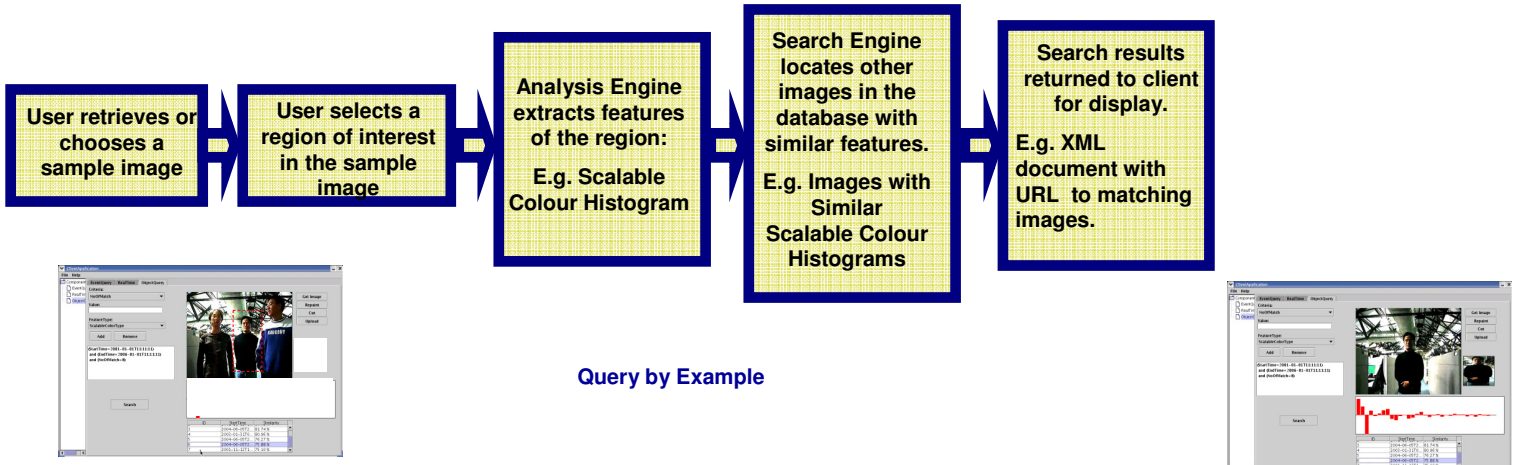
Home RF Network

Home entertainment unit

Embedded
MPEG-4
Players
Error resilient decoder

1.3 Multimedia Applications-- Content Management Demonstration Platform

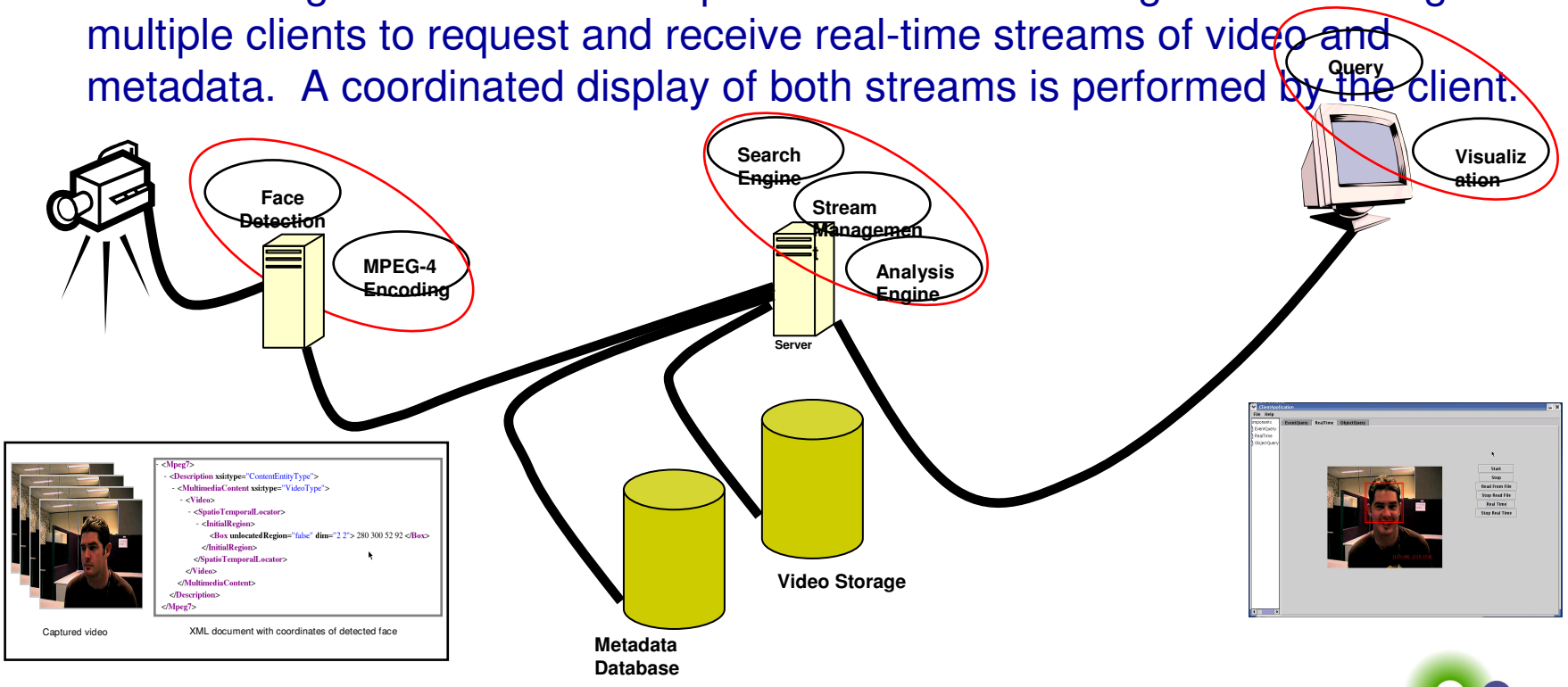
- Client / Server platform demonstrating content based search using MPEG-7 visual descriptors
  - Content can be searched using methods “query by specification” or “query by example”.
  - For “query by example”, the Analysis Engine at the server extracts visual features from images
  - the Search Engine searches for archived images that have similar features to those of the example region.



1.3 Multimedia Applications-- Content Management Demonstration Platform

- Real Time MPEG-7 metadata and coordinated display at the client

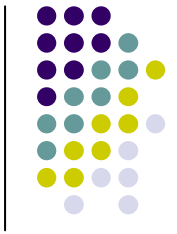
  the metadata stream refers to locations of faces identified by a face detection algorithm. The server performs Stream Management allowing multiple clients to request and receive real-time streams of video and metadata. A coordinated display of both streams is performed by the client.



1.4 Introduction to Multimedia Research

- Why we need video compression ?
  - There is a big gap of digital bandwidth demand between users expectation for multimedia application.
    - To handle huge uncompressed digital video
    - The state-of-the-art of computer and telecommunication technologies
      - storage capacities including access speed
      - transmission capacities
  - Money saving to pay the cost for video data in transmission and storage applications

1.4 Introduction to Multimedia Research



- Comparison figures:

Ref: H. Wu

Uncompressed ITU-R Rec. 601:

- 720 x 576 pixels at 25 frames per second (4:2:2);
- **8-bit** per component, leading to

Total bit rate = 25 frames/s x (720+360+360) x576 component pixels/frame x 8 bits per component pixel

= **166 Mbps**,

and a 90 minute movie requires over 100 GBytes.

If one DVD can save 4.6 GB, How many DVDs?

If the ADSL-2@512 bits/second (bps), what's the Compression ratio required for video transmission?

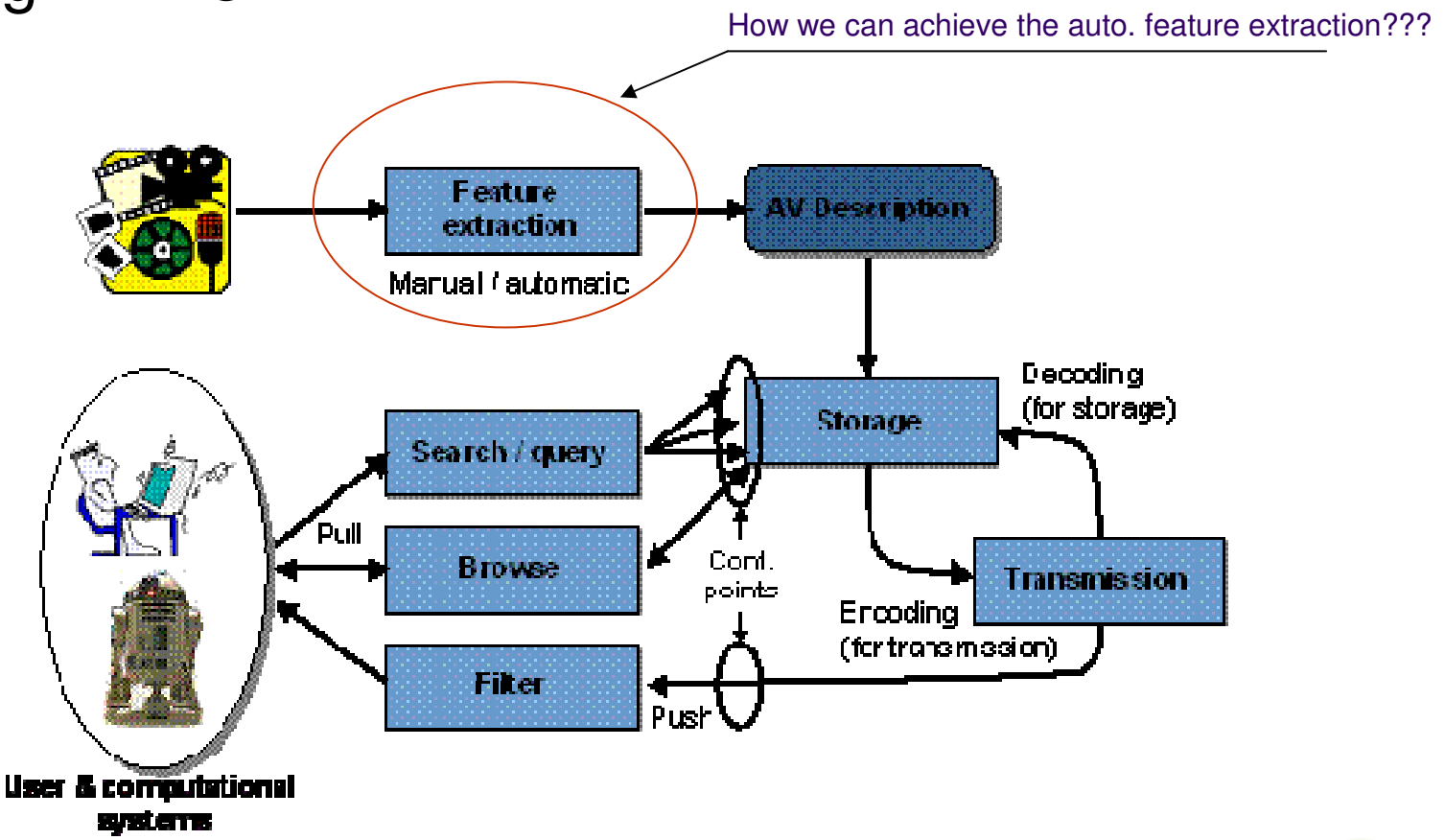
1.4 Multimedia Content Management

- Why we need multimedia content management ?
  - It is necessary to develop a platform to handle the following cases
    - where audiovisual information is created, exchanged, retrieved, and re-used by computational systems
    - where information retrieval is required for quickly and efficiently searching for various types of multimedia documents of interests of the users
    - where a stream of audiovisual content description for users to receive only those multimedia data items which satisfy their preference

1.4 Introduction to Multimedia Research

Ref: J. Martinez

- Abstract representation of possible applications using MPEG-7



1.4 Multimedia Content Management

- Given that the strong need for multimedia content management, there are some key challenges:
  - Majority of existing techniques for content management are based on low-level features
  - There is a significant gap between:
    - low-level feature extraction and users expectation on high level understanding (semantic level)
  - Video analysis and understanding technologies (tools) serves the key enabling technology towards semantic content description and representation
  - **This field of research presents significant challenges and enormous opportunities !!!**

1.4 Introduction to Multimedia Research

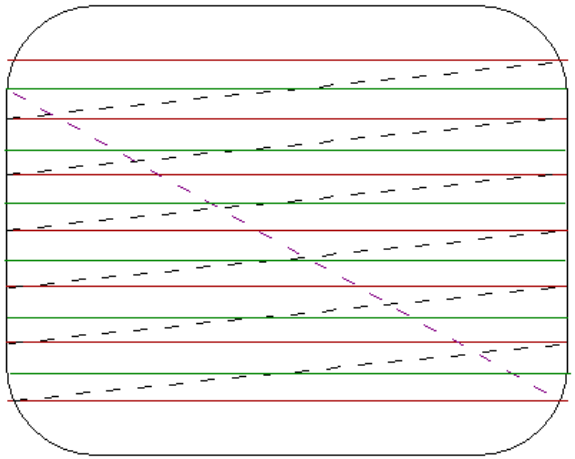
- Demos:
  - Video Content Understanding
    - Search by event
    - Search by example
  - Video Content Analysis
    - Object tracking under heavy collusions
    - Object classification
    - Tracking trajectory
  - Video coding and Communication
    - MPEG-4 coding and transmission
    - Scalable coding
    - Error resilience coding
    - Two-way video communication on PDA or 3G Mobile phone

1.5 Basic Concepts of Image and Video Processing

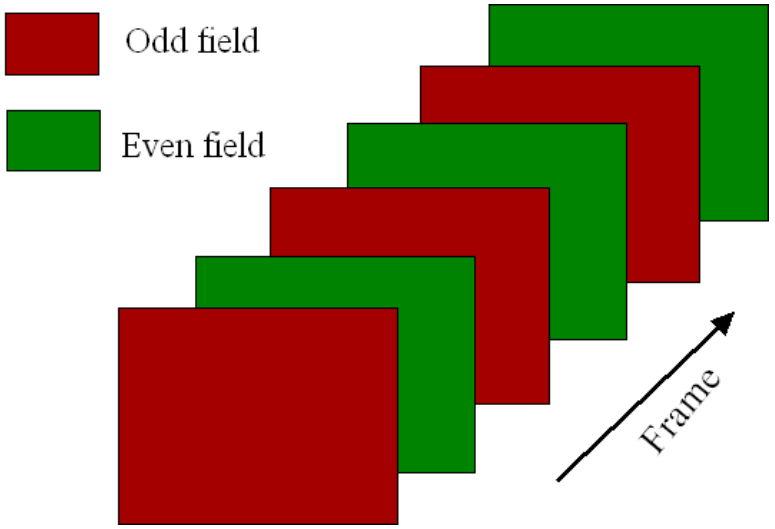
- 1.5.1 Image and Video Sequence
- 1.5.2 Pixel Representation
- 1.5.3 Chrominance sub-sampling
- 1.5.4 Digital Video Formats
- 1.5.5 Information Measure – a Review
- 1.5.6 Introduction to Entropy Coding

1.5.1 Image and Video Sequence

- Digital Image & Video
  - The basic unit to build a image is called pixel (pel)
  - Image resolution is calculated by pixels in horizontal and vertical coordinates
  - Video sequences consist of a number of motion pictures
  - One of the video sequence formats is well known as Interlaced structure that follows the analogue PAL format



Scanning pattern of TV



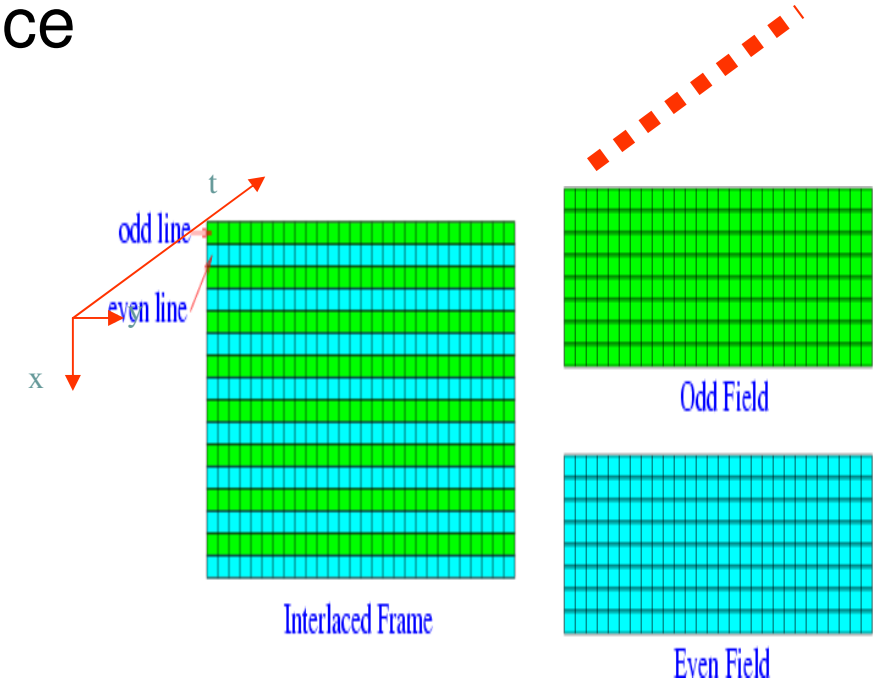
1.5.1 Video Image and Sequence

- Interlaced Video Format
  - Standard frame rates (25 or 30 frames per second) are high enough to provide smooth motion, but are not high enough to prevent flickering.
  - To prevent perceptible flicker in a bright image, the refresh rate must be at least 50 frames per second. With 2-to-1 interlacing, the odd numbered lines of a frame are displayed first (field 1), followed by the even lines (field 2).
  - A 25 frame per second sequence is displayed at 50 fields per second.
  - The eye does not readily perceive flickering objects that are small, the 25 per second repetition rate of any one scan line is not seen as flickering, but the entire picture appears to be refreshed 50 times per second.

1.5.1 Image and Video Sequence

- Image & Video Sequence

(a) Frame or picture



(b) Video sequence

1.5.2 Pixel Representation

- Y,U,V Colour Space
  - The Human Visual System (HVS) is sensitive to three colour components. Colour can be represented by Red, Green and Blue components (*RGB*).
  - Transform to YUV or *YCbCr* with less correlated representation:.

$$Y = 0.299R + 0.587G + 0.114B$$

$$U_t = \frac{B - Y}{2.03}$$

$$V_t = \frac{R - Y}{1.14}$$

$$\begin{bmatrix} Y \\ U_t \\ V_t \end{bmatrix} = \underbrace{\begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.100 \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Note:

The two chrominance components (U,V) contain considerably less information than the luminance component. For this reason, chrominance is often sub-sampled

1.5.2 Pixel Representation

- $Y_d C_b C_r$ Colour Space

For digital component signal (CCIR Rec 601), 8-bit digital variables are used, however:

1. Full digital range is not used to give working margins for coding and filtering.
2. RGB to $Y_d C_b C_r$ conversion is given by

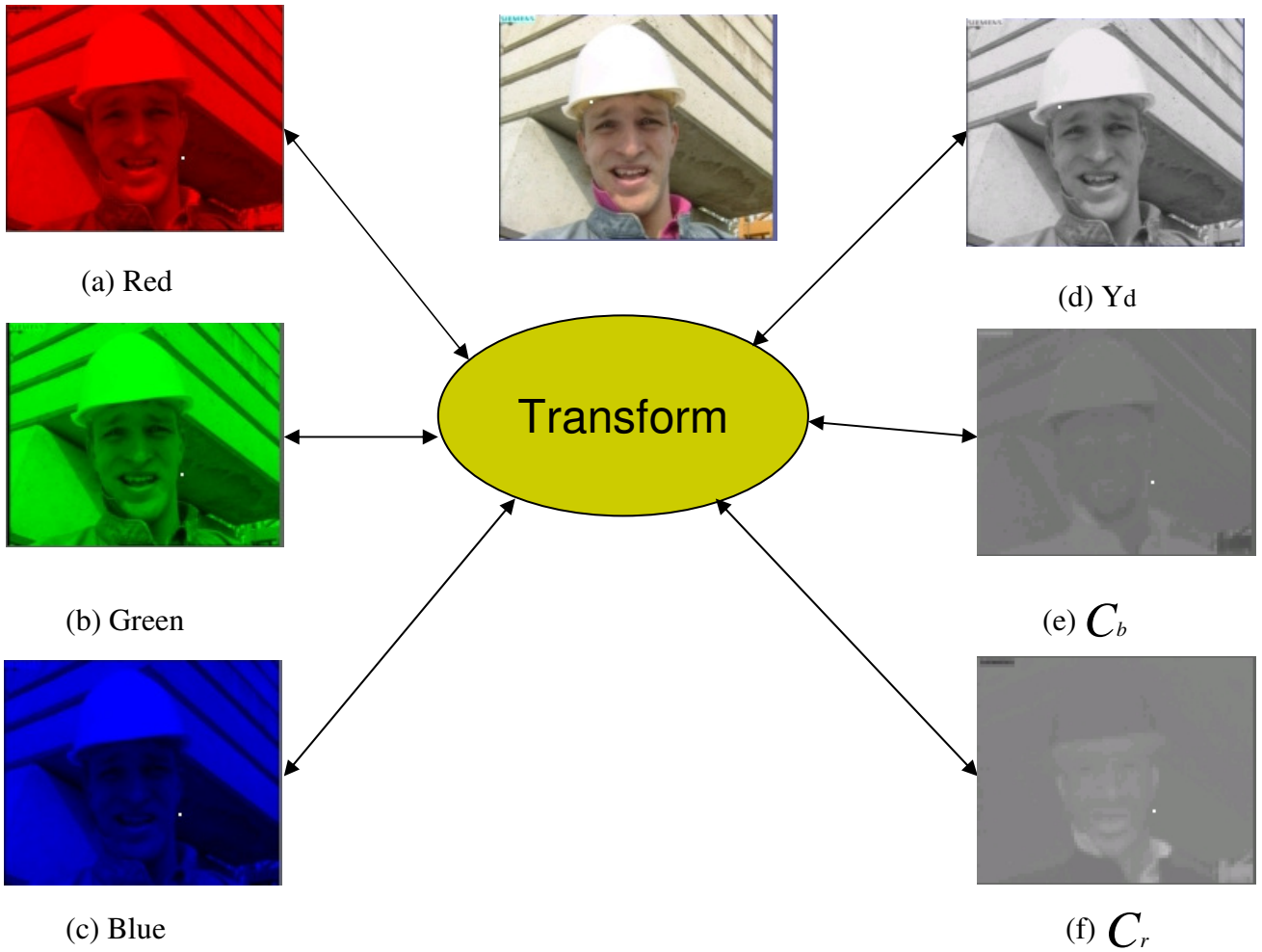
$$\begin{bmatrix} Y_d \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0.257 & 0.504 & 0.098 \\ -0.148 & -0.291 & 0.439 \\ 0.439 & -0.368 & -0.071 \end{bmatrix} \begin{bmatrix} R_d \\ G_d \\ B_d \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} \qquad \begin{bmatrix} R_d \\ G_d \\ B_d \end{bmatrix} = \begin{bmatrix} 1.164 & 0.000 & 1.596 \\ 1.164 & -0.392 & -0.813 \\ 1.164 & 2.017 & 0.000 \end{bmatrix} \begin{bmatrix} Y_d - 16 \\ C_b - 128 \\ C_r - 128 \end{bmatrix}$$

The positive/negative values of U and V are scaled and zero shifted in a transformation to the Cb and Cr coordinates.

where digital luminance, Yd, has a rang of (16-235) with 220 levels starting at 16, and digital chrominance difference signals, Cb and Cr, have a range of (16-240) with 225 levels centered at 128.

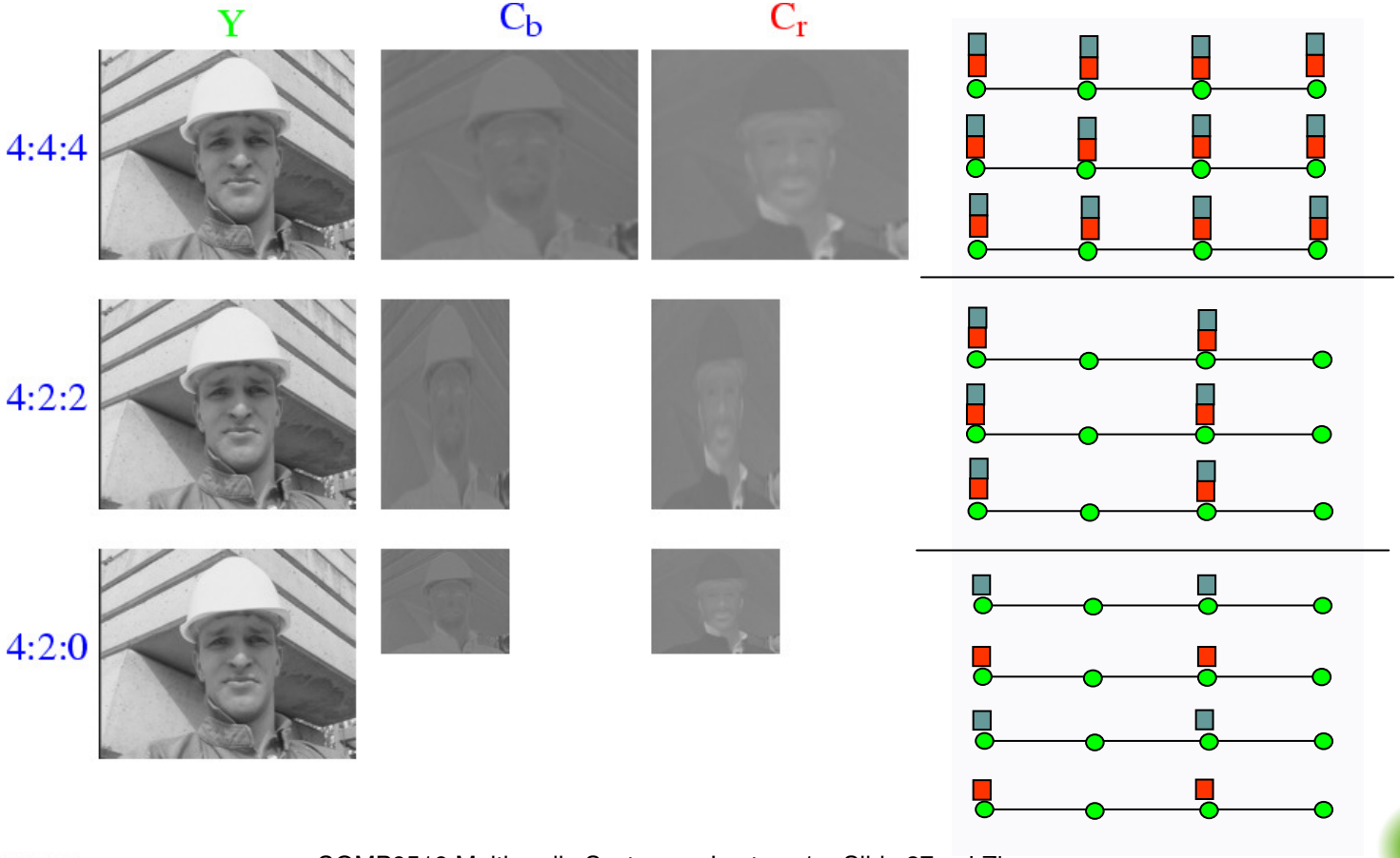
1.5.2 Pixel Representation

- Yd,Cb,Cr Colour Space



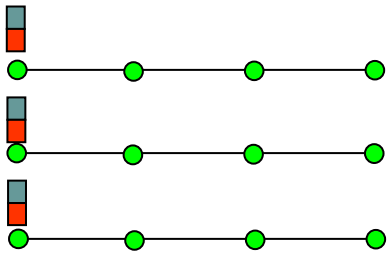
1.5.3 Chrominance sub-sampling

- Human vision is relatively insensitive to chrominance. For this reason, chrominance is often sub-sampled.
- Chrominance sub-sampling is specified as a three-element ratio.



1.5.3 Chrominance sub-sampling

- In 4:4:4 format: Y, Cr & Cb – 720 x 576 pixels per frame
- In 4:2:2 format: Y – 720 x 576 and Cr & Cb – 360 x 576 pixels per frame
- In 4:2:0 format: Y – 720 x 576 and Cr & Cb – 360 x 288 pixels per frame
  - A commonly used format is **4:2:0** which is obtained by sub-sampling each colour component of 4:2:2 source vertically to reduce the number of lines to 288;
- 4:1:1 format is obtained by sub-sampling 4:4:4 source by 4:1 horizontally only. Y – 720x576 and Cr&Cb – 144x576

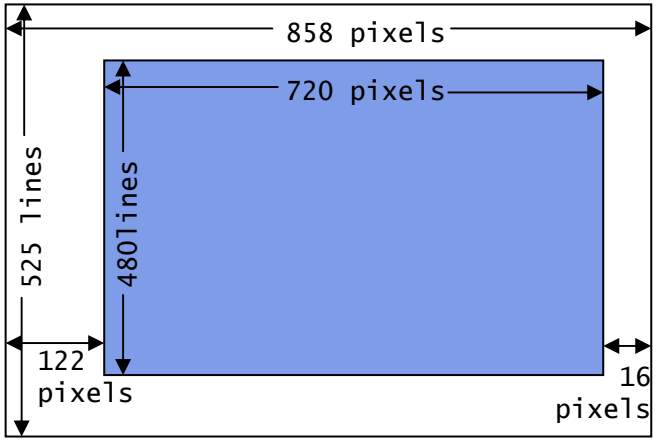


4:1:1 format

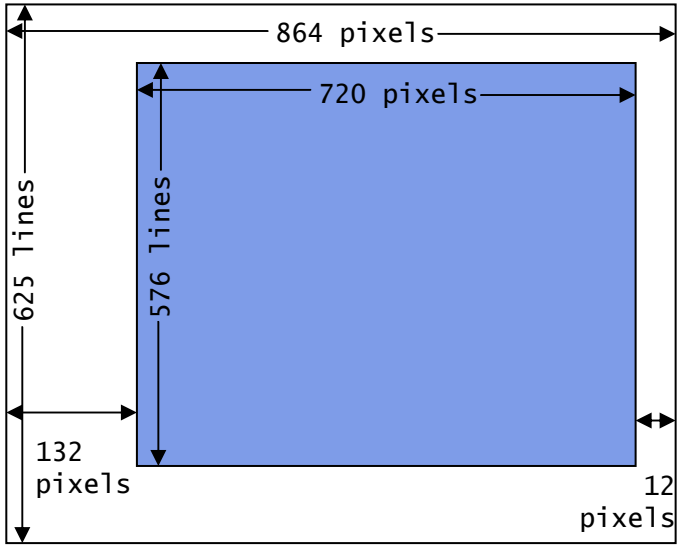
1.5.4 Digital Video Formats

- International Consultative Committee for Radio (CCIR) Rec. 601:
  - – Two display rates:

    50Hz: 720x576 pixels at 50 fields per second.

    60Hz: 720x480 pixels at 60 fields per second.
  - – Both rates are 2:1 interlaced and 4:2:2 chrominance sampling (with optional 4:4:4).
- 

525/60: 60 fields/s



625/50: 50 fields/s

NOTE: Figures referred from Y. Wang et al, 2002.

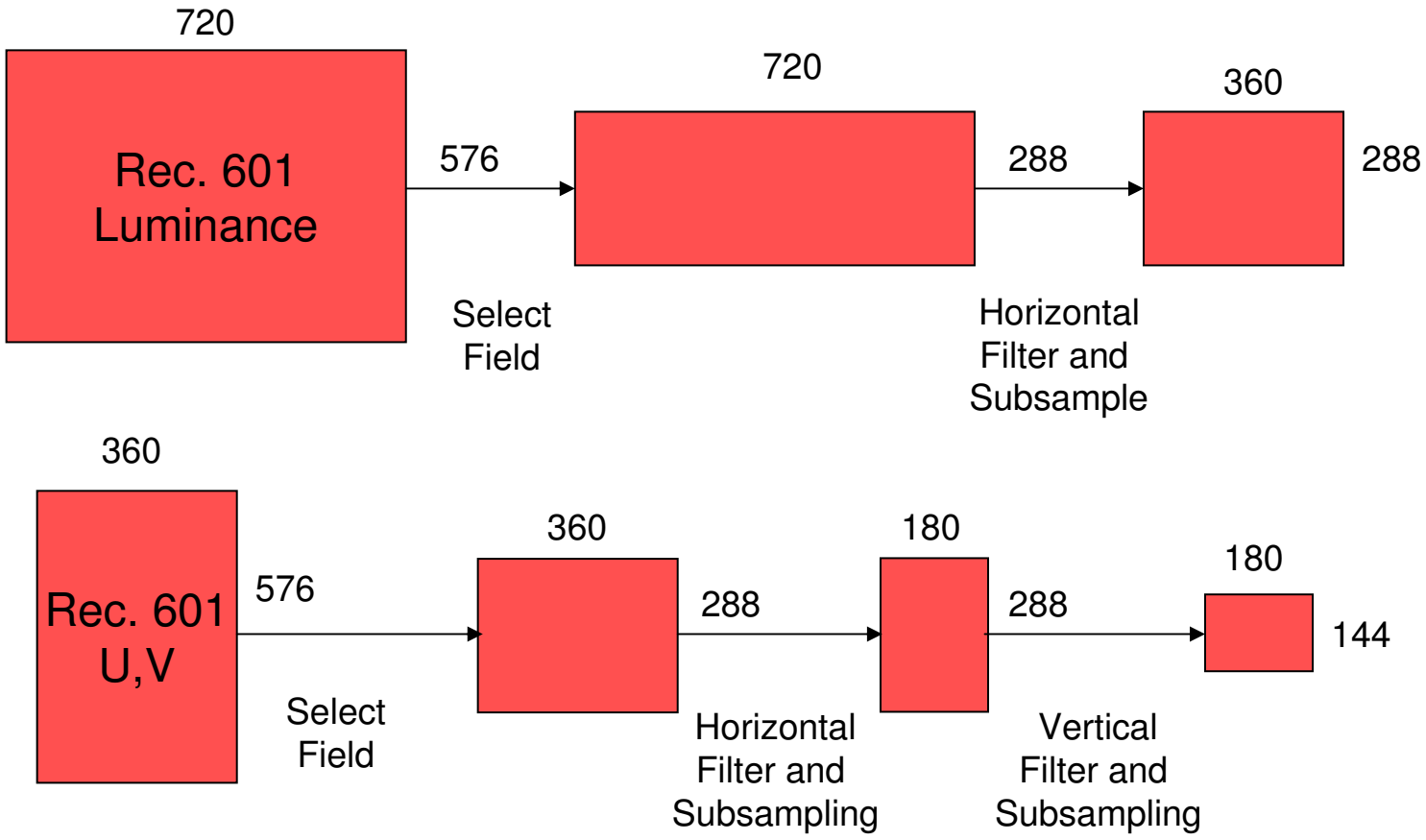
1.5.4 Digital Video Formats

- Common Intermediate Format (CIF):
  - This format was defined by CCITT (TSS) for H.261 coding standard (teleconferencing and videophone).
  - – Several size formats:
    - **SQCIF:** 88x72 pixels.
    - **QCIF:** 176x144 pixels.
    - **CIF:** 352x288 pixels.
    - **4CIF:** 704x576 pixels.
- – Non-interlaced (progressive), and chrominance sub-sampling using 4:2:0.
- – Frame rates up to 25 frames/sec

1.5.4 Digital Video Format

- Source Input Format (SIF):
  - – Utilized in MPEG as a compromise with Rec. 601.
  - – Two size formats (similar to CIF):
    - **QSIF:** 180x120 or 176x144 pixels at 30 or 25 fps
    - **SIF:** 360x240 or 352x288 pixels at 30 or 25 fps
  - – Non-interlaced (progressive), and chrominance sub-sampling using 4:2:0.
  - – It is assumed that SIF is derived from a Rec.601.
- High Definition Television (HDTV):
  - – 1080x720 pixels.
  - – 1920x1080 pixels.
- JPEG format
  - ITIF
- Audio Format

1.5.4 Digital Video Format



Interlaced format conversion

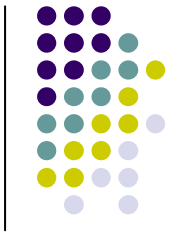
1.5.5 Information Measurement -- Review

- Information Measure
  - Consider a symbol *x* with an occurrence probability *p*, its info. content $i(x)$(i.e. the amount of info contained in the symbol)

$$i(x) = \log_2 \left\lfloor \frac{1}{p(x)} \right\rfloor = -\log_2 p(x) \qquad \text{bits} \qquad 2\text{-}1$$

  - The smaller the probability, the more info. the symbol contains
  - The occurrence probability somewhat related to the uncertainty of the symbol
    - A small occurrence probability means large uncertainty or the info. Content of a symbol is about the uncertainty of the symbol.
- Average Information per Symbol
  - Consider a discrete memoriless information source
    - By discreteness, the source is a countable set of symbols
    - By memoriless, the occurrence of a symbol in the set is independent of that of its preceding symbol.

1.5.5 Information Measurement -- Review



- Look at a source that contains *m possible* symbols: {***si**, i=1,2..m*}
- The occurrence probabilities: {***Pi**, i=1,2..m*}
- The info. content of a symbol ***si;*** $I_i = i(s)_i = -\log_2 p_i$ bits

- Information Entropy
  - The Entropy is defined as the average information content per symbol of the source. The Entropy, *H, can be expressed as follows:*

$$H = -\sum_{i=1}^{m} p_i \log_2 p_i \quad \text{bits}$$

  - From this definition, the entropy of an information source is a function of occurrence probabilities.
  - The entropy reaches the Max. when all symbols in the set are equally probable.

1.5.5 Information Measurement -- Review

- Information Content
  - Consider the two blocks of binary data shown below which contains the most information?

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

$$P_0 = \frac{63}{64}$$

$$P_1 = \frac{1}{64}$$

$$H = -\frac{63}{64}\log_2\frac{63}{64} - \frac{1}{64}\log_2\frac{1}{64}$$

$$= 0.116 \text{ bits/pixel}$$

0	1	1	0	1	0	1	0
1	0	1	0	1	0	0	1
1	1	0	1	0	1	1	0
0	1	0	0	1	1	0	0
1	0	0	0	1	0	1	1
0	0	1	0	1	1	1	1
0	1	0	1	1	1	0	1
0	1	0	0	0	1	0	0

$$P_0 = \frac{32}{64} = \frac{1}{2}$$

$$P_1 = \frac{32}{64} = \frac{1}{2}$$

$$H = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$$

$$= 1.0 \text{ bits/pixel}$$

1.5.5 Information Measurement -- Review

- **Definition-- Image mean**
  - Given a two-dimensional (2-D) image field with pixel value, x[n.m], n=1,2,…,N and m=1,2,…,M, the mean of the image is defined as the spatial average of the luminance values of all pixel, i.e.,

$$\bar{x} = \frac{1}{N \times M} \sum_{n=1}^{N} \sum_{m=1}^{M} x[n,m] \qquad (2\text{-}5)$$

- **Definition--Image variance**
  - Given a two-dimensional (2-D) image field with pixel value, x[n.m], n=1,2,…,N and m=1,2,…,M, the variance of the image is defined as the average value of the squared difference between the value of an arbitrary pixel and the image mean, i.e.,

$$\sigma^2 = \frac{1}{N \times M} \sum_{n=1}^{N} \sum_{m=1}^{M} (x[n,m] - \bar{x})^2 \qquad (2\text{-}6)$$

1.5.5 Information Measurement -- Review

- **Image quality measurement (MSE,MAE,SNR and PSNR)**

  Assume symbol x represents the original image and $\hat{x}$ the reconstructed image, M and N the width and the height of respectively

  - Mean squared error (MSE):

$$MSE = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}[x(m,n)-\hat{x}(m,n)]^2 \qquad (2\text{-}7)$$

  - Mean absolute error (MAE):

$$MAE = \frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\left|x(m,n)-\hat{x}(m,n)\right| \qquad (2\text{-}8)$$

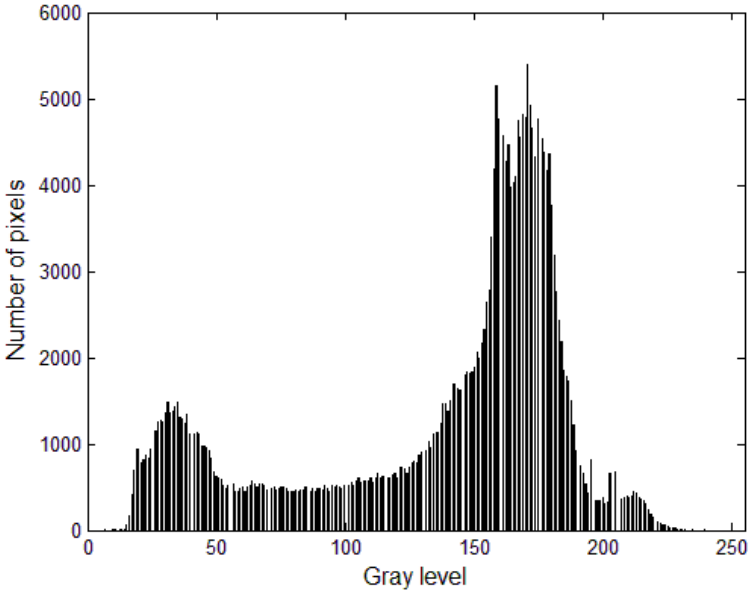
  - Peak signal to noise ration (PSNR):

$$PSNR = 10\log_{10}\frac{255^2}{MSE}dB \qquad (2\text{-}9)$$

Peak pixel value is assumed 255

1.5.6 Introduction to Entropy Coding

- Image Histogram



Entropy = 7.63 bits/pixel

1.5.6 Introduction to Entropy Coding

- The number of bits required to represent an image can be made based on the information content using an entropy (variable length coding) approach such as a Huffman code

- Highly probable symbols are represented by short code-words while less probable symbols are represented by longer code-words

- The result is a reduction in the average number of bits per symbol

1.5.6 Introduction to Entropy Coding

- Example
  - Fixed length coding

Symbol	Probability	Codeword	Codeword Length
A	0.75	00	2
B	0.125	01	2
C	0.0625	10	2
D	0.0625	11	2

  - Average bits/symbol = 0.75*2 + 0.125*2 + 0.0625*2 + 0.0625*2 = 2.0 bits/pixel

1.5.6 Introduction to Entropy Coding

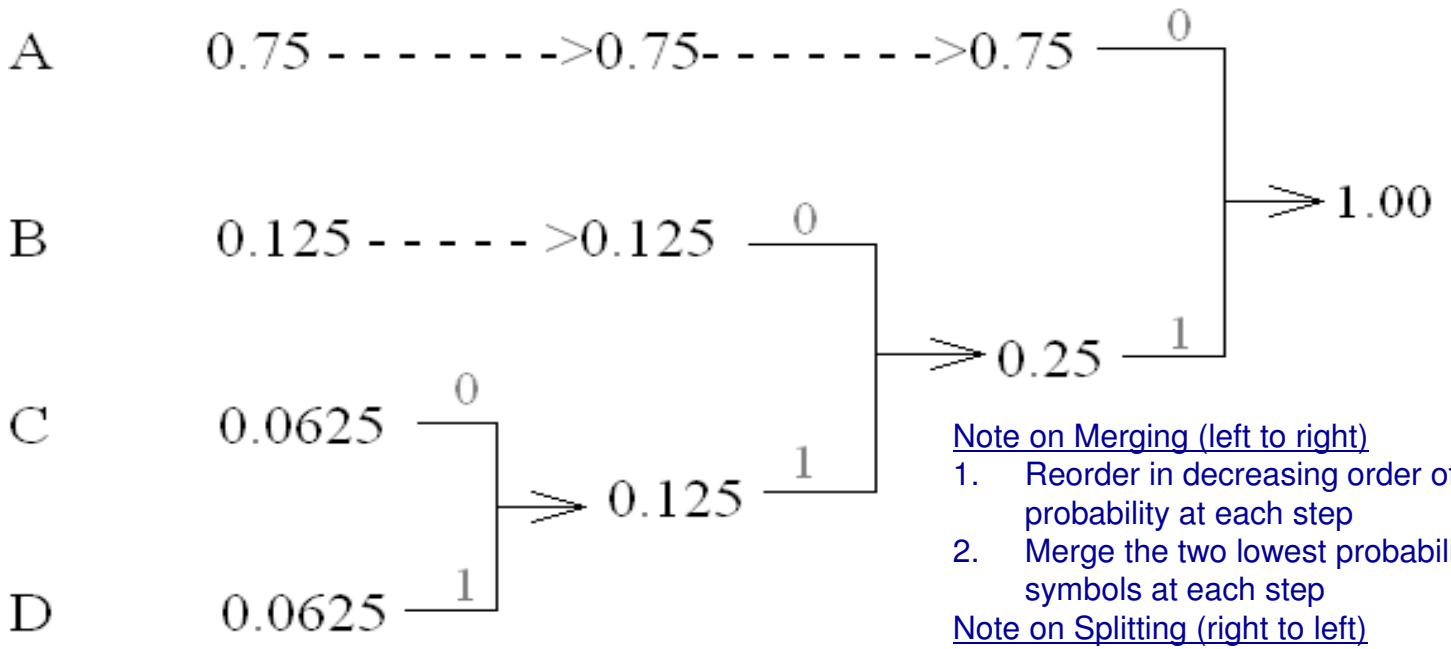
- Example
  - Entropy (Variable Length) Coding

Symbol	Probability	Codeword	Codeword Length
A	0.75	0	1
B	0.125	10	2
C	0.0625	110	3
D	0.0625	111	3

- Average bits/symbol = 0.75*1 + 0.125*2 + 0.0625*3 + 0.0625*3 = 1.375 bits/pixel (A 30% saving with no loss)

1.5.6 Introduction to Entropy Coding

- Generation of Huffman Codewords
  - If the symbol probabilities are known, Huffman codewords can be automatically generated



Note on Merging (left to right)

1. Reorder in decreasing order of probability at each step
2. Merge the two lowest probability symbols at each step

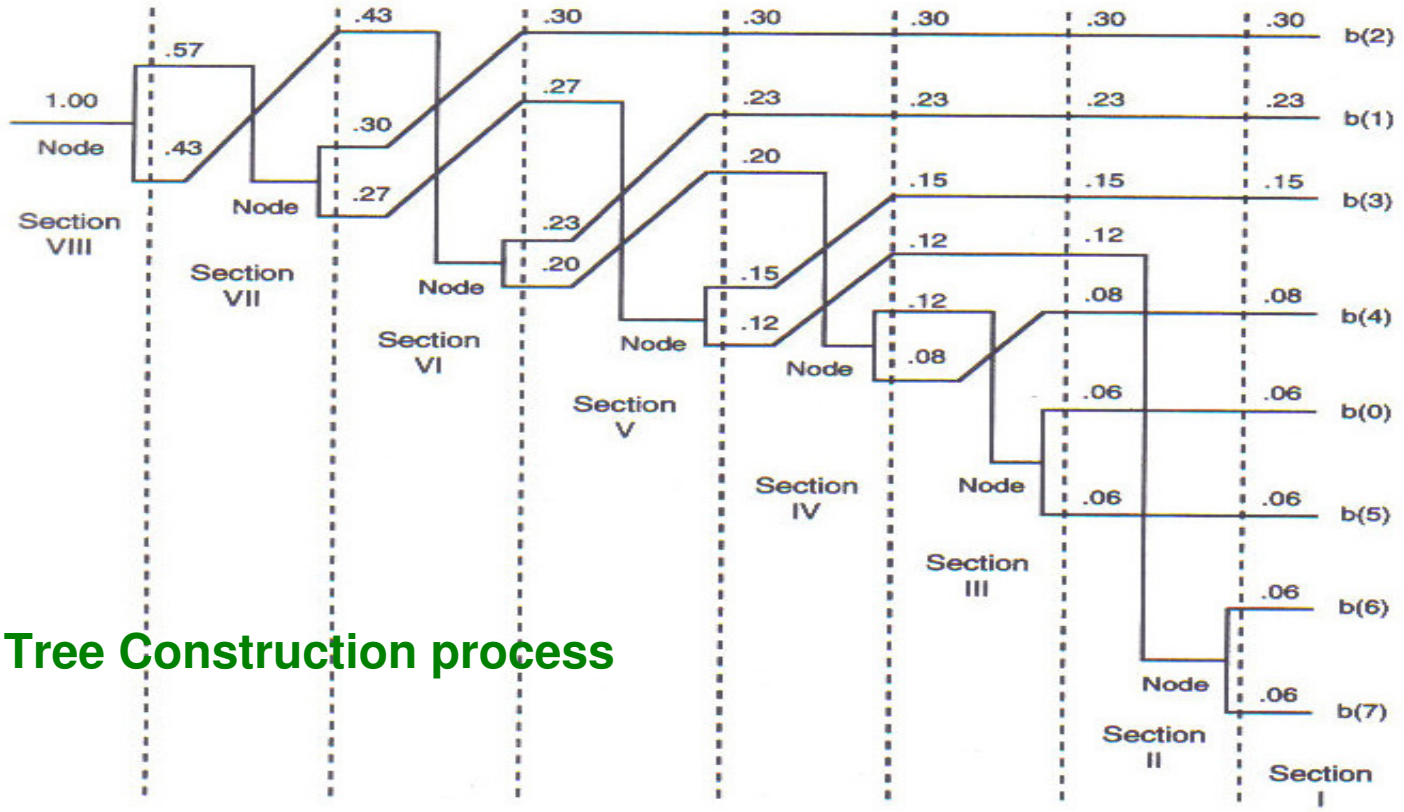
Note on Splitting (right to left)

1. Split the symbol merged at that step into two symbols

Details are introduced in next two slides

1.5.6 Introduction to Entropy Coding

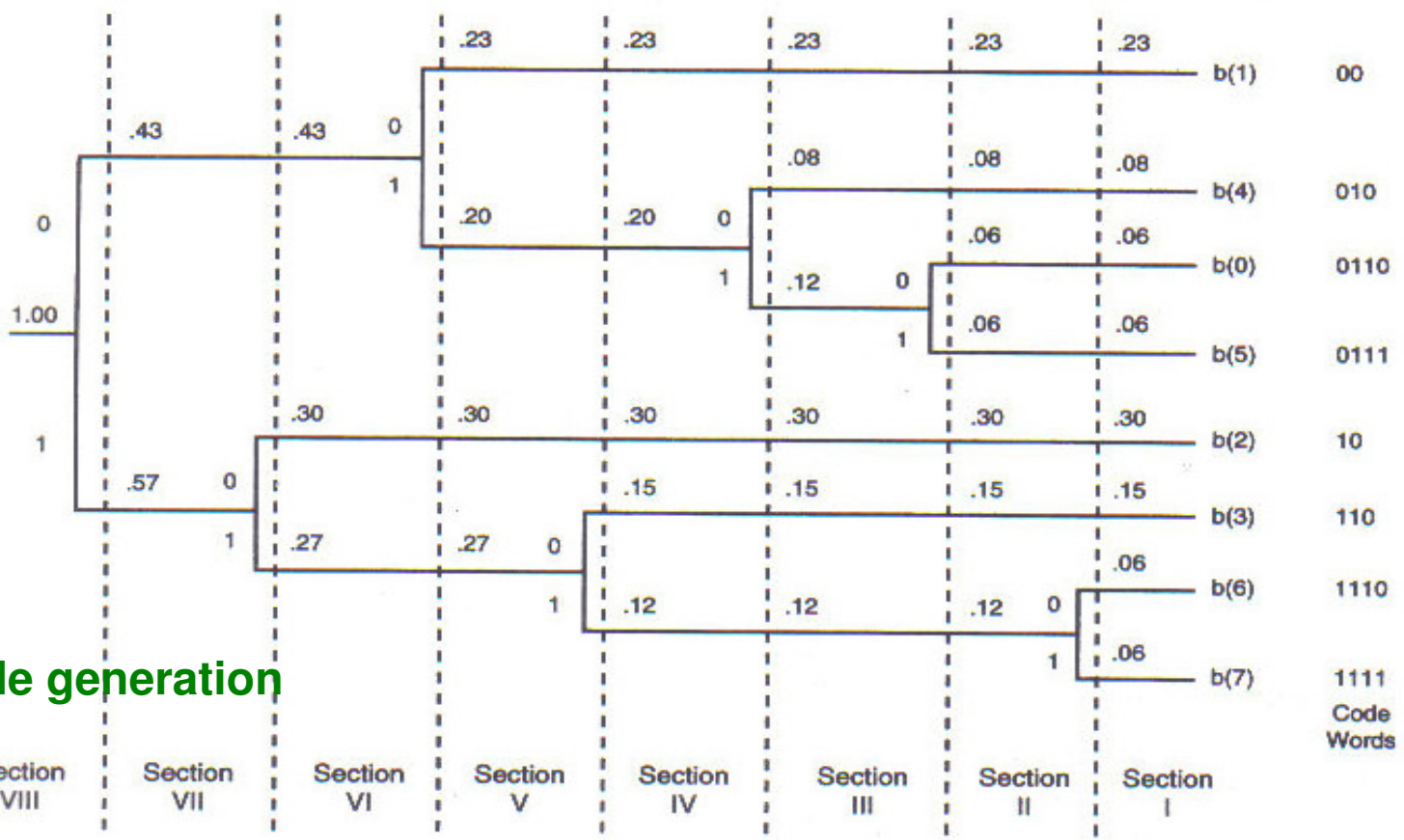
1. From Right to Left, 2. Two bottom-most branches are formed a node
3. Reorder probabilities into descending order



Tree Construction process

1.5.6 Introduction to Entropy Coding

1) Re-arrange the tree to eliminate crossovers, 2) The coding proceeds from left to right, 3) 0– step up and 1– step down.



Code generation

1.5.6 Introduction to Entropy Coding

- Truncated and Modified Huffman Coding
  - Given the size of the code book is L, the longest codeword will reach L bits
  - For a large quantities of symbols, the size of the code book will be restricted
  - Truncated Huffman coding
    - For a suitable selected L1<L, the first L1 symbols are Huffman coded and the remaining symbols are coded by a prefix code, following by a suitable fixed-length code

- Second Order Entropy
  - Instead of find the entropy of individual symbols, they can be grouped in pairs and the entropy of the symbol pairs calculated. This is called the ***SECOND ORDER ENTROPY.*** For correlated data, this will lead to an entropy closer to the source entropy.

1.5.6 Introduction to Entropy Coding

- Limitations of Huffman Coding
  - Huffman codewords have to be an integer number of bits long.
    - If the probability of a symbol is 1/3, the optimum number of bits to encode that symbol is -log2 (1/3) =1.6. Assigning two bits leads to a longer code message than is the theoretically necessary
  - The symbol probabilities must be known in the decoder size. If not, they must be generated and transmitted to the decoder with the Huffman coded data
  - A larger of number of symbols results in a large codebook
  - Dynamic Huffman coding scheme exists where the code words are adaptively adjusted during encoding and decoding, but it is complex for implementation.