Fundamentals of Multimedia

Authored by Ze-Nian Li Mark S. Drew

Lecturer：Lu Dongming

11. MPEG Video Coding (Ⅱ)

MPEG-4, 7 and Beyond

Content

- MPEG-4
  - Overview of MPEG-4
  - Object-Based Visual Coding In MPEG-4
  - Synthetic Object Coding In MPEG-4
  - Object Types, Profiles and Levels
  - MPEG-4 Part10/H.264
- MPEG-7
- MPEG-21

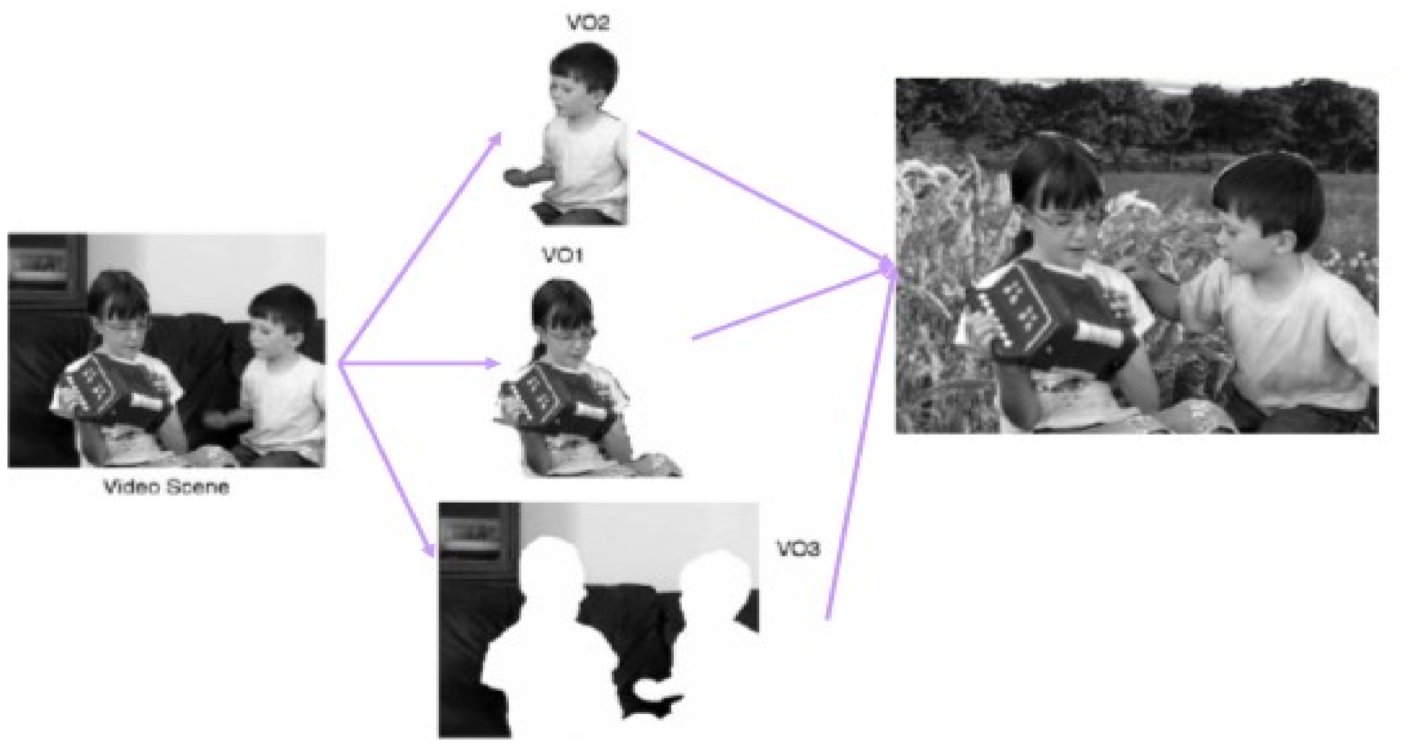
1. MPEG-4

1.1 Overview

- **MPEG-4**
  - **Pays great attention to User Interactivities**
  - **The bitrate covers a large range between 5kbps and 10Mbps.**
- **Some characters**
  - **Object based coding**
  - **Arbitrary Shape Coding**
  - **Static texture coding**
  - **Face object coding and Animation**
  - **Body object coding and Animation**

1.1 Overview

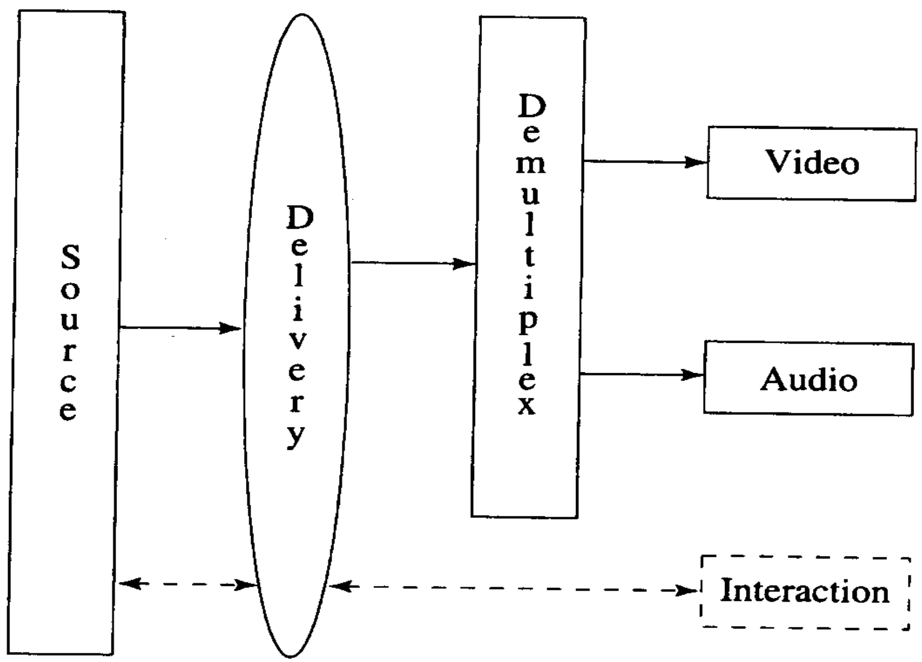
- Object based coding



MPEG-4 Object based processing

1.1 Overview

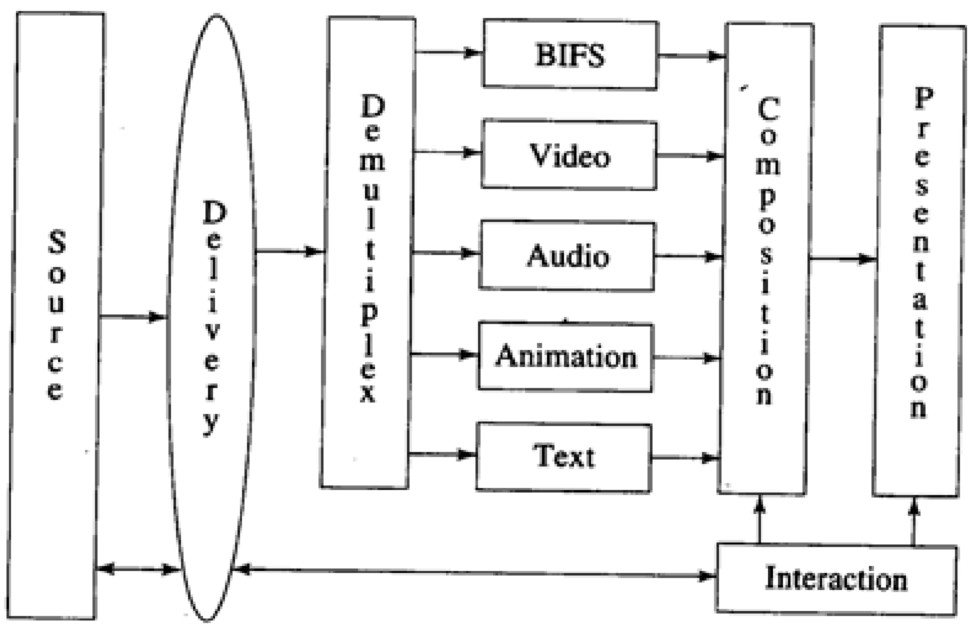
- **Comparison of interactivities in MPEG standards. (a)**
  - **Reference models in MPEG-1 and 2**



Interaction in dashed lines supported only by MPEG-2

1.1 Overview

- Comparison of interactivities in MPEG standards. (b)
  - MPEG-4 reference model



1.1 Overview

- **Hierarchical structure of MPEG-4 bitstreams**
  - Video-object Sequence (VS)
  - Video Object (VO)
  - Video Object Layer (VOL)
    - Scalable coding
  - Group of Video Object Planes (GOV)
    - Optional level
  - Video Object Plane (VOP)
    - Snapshot of a VO at a particular moment

VS
VO
VOL
GOV
VOP

1.2 Object-based visual coding

- VOP-Based Coding vs. Frame-Based Coding
- Motion Compensation

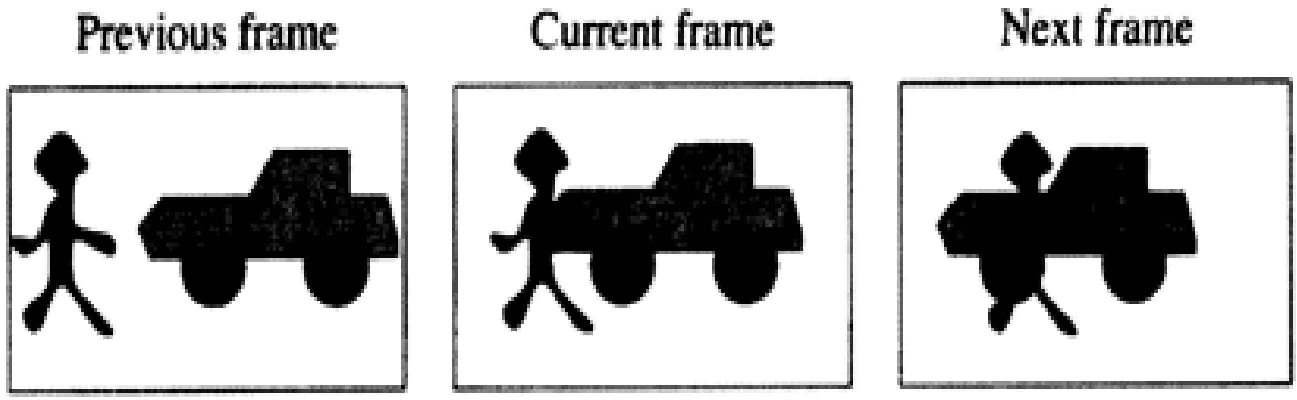
- Texture Coding
- Shape Coding
- Static Texture Coding
- Sprite Coding

- Global Motion Compensation

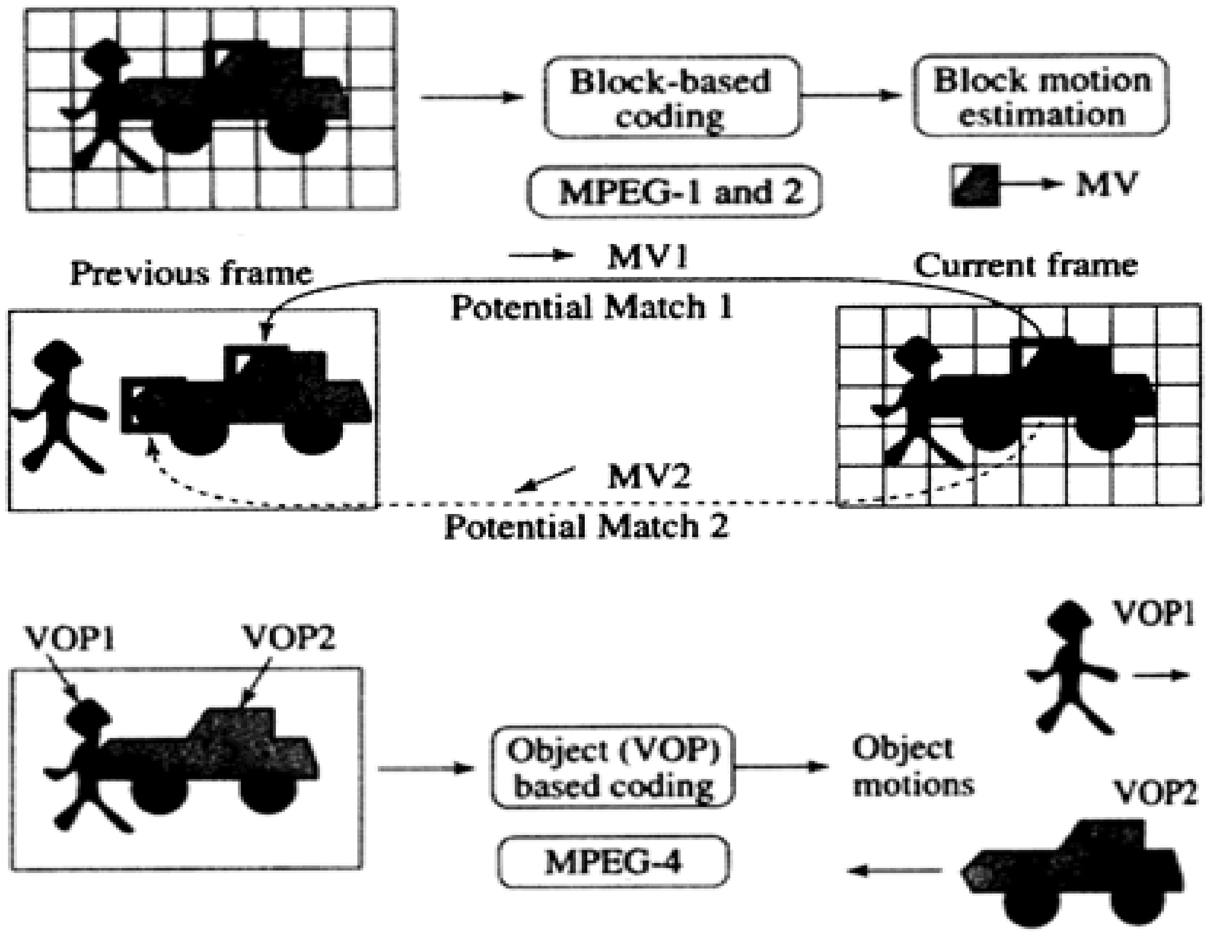
1.2.1 VOP-Based Coding vs. Frame-Based Coding

MPEG-1 and MPEG-2 are Frame-Based Coding

- **Motion vectors generated by frame-based coding may be inconsistent with the object's motion**



1.2.1 VOP-Based Coding vs. Frame-Based Coding



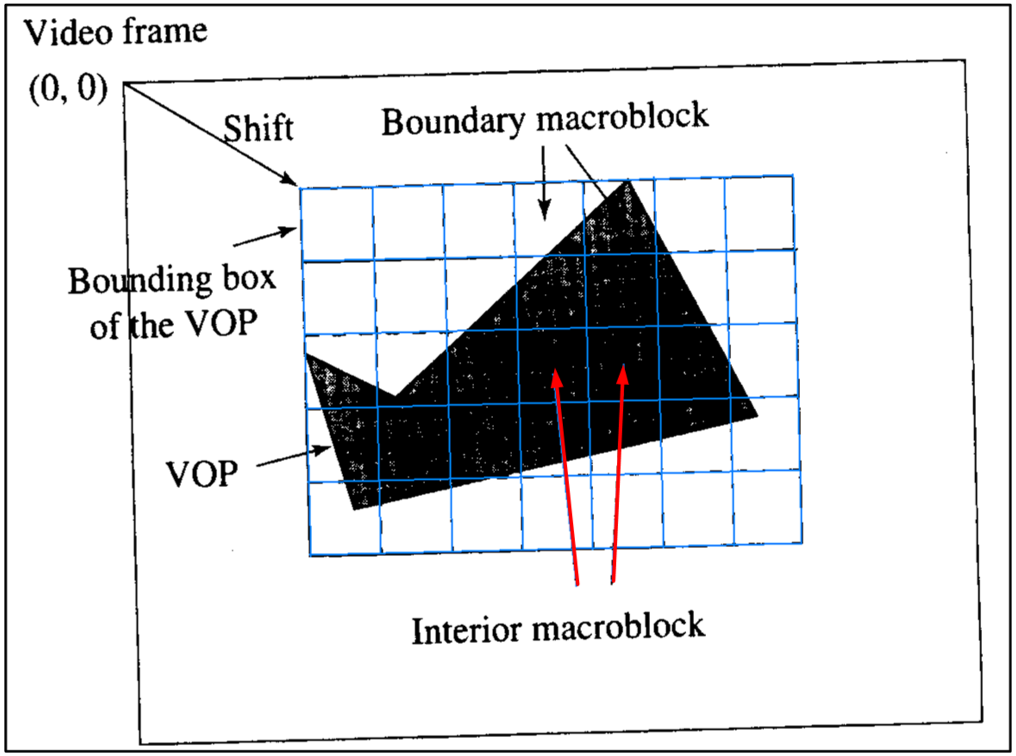
1.2.2 Motion Compensation

- Motion estimation
- Motion-compensation-based prediction

- Coding the prediction error
  - Defines a rectangular bounding box for each VOP
  - Interior macroblocks and boundary macroblocks

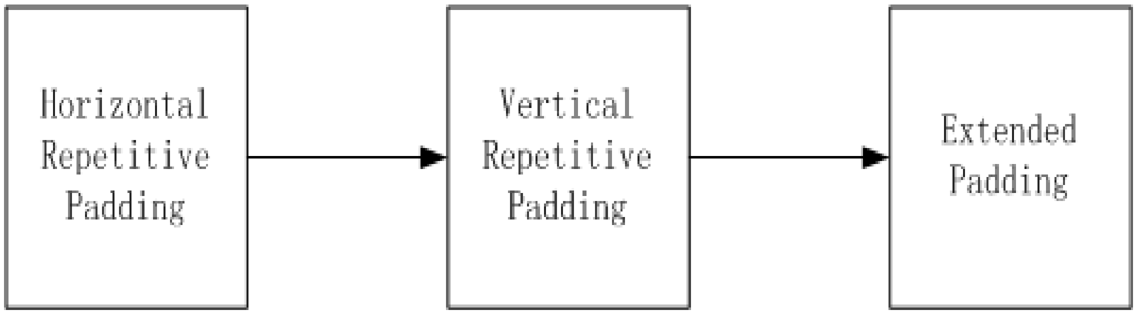
1.2.2 Motion Compensation

- Motion compensation for **interior macroblocks** is carried out in the same manner as in MPEG-1 and 2



1.2.2 Motion Compensation

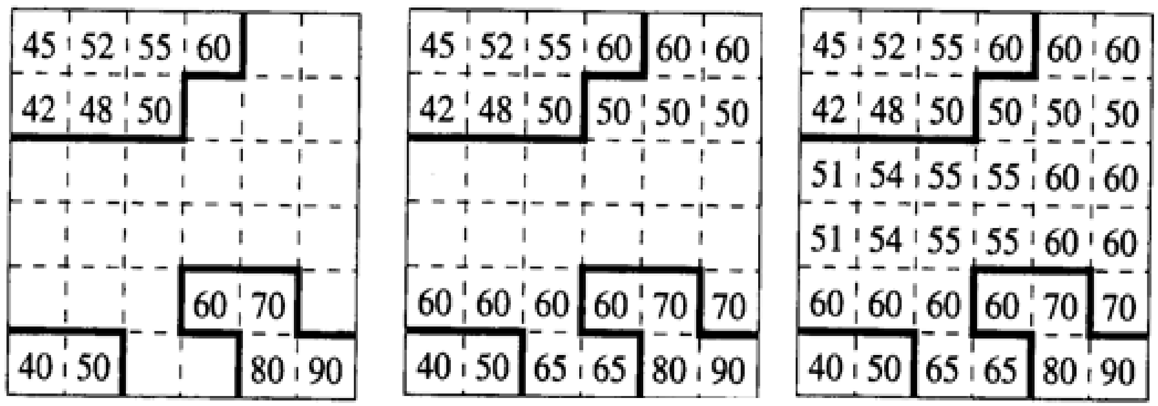
- **Padding is applied prior to motion compensation for boundary macroblocks.**



- **The horizontal padding examines each row and every boundary pixel is replicated to the left and/or right to fill the values out side the VOP.**
- **If the interval is bounded by two boundary pixel, their average is adopted;**
- **The vertical padding works similarly.**

1.2.2 Motion Compensation

- **Padding (an example)**



Original Image Horizontal Padding Vertical Padding

- Extended Padding: exterior macroblocks immediately next to boundary macroblocks are filled by replicating the values of the border pixels of the boundary macroblock.
- The macroblocks to use follows a priority list:

  left, top, right, bottom

1.2.2 Motion Compensation

- **Motion Vector Coding**
  - **Motion estimation**

$$SAD(i,j)=\sum_{k=0}^{N-1}\sum_{l=0}^{N-1}|C(x+k,y+l)-R(x+i+k,y+j+l)|\cdot Map(x+k,y+l)$$

$Map(p,q)=1$ if $C(p,q)$ is pixel in VOP

else $Map(p,q)=0$

motion vector $MV:(u,v)=\{(i,j)\,|\,SAD(i,j)$ is minimum

$i,\ j\in[-p,p]\}$, p is maximum of u and v

  - **Allows quarter-pixel precision in the luminance components.**
  - **MV can point beyond the boundaries of the reference VOP, pixel outside the VOP is defined in padding step.**

1.2.3 Texture Coding

- I-VOP coded like JPEG
- For P-VOP and B-VOP
  - The prediction error is sent to DCT and VLC
- Texture coding based on DCT
  - For portions of the boundary macroblocks outside the VOP, zeros are padded
  - Quantization step_size for the DC component is 8
  - Two methods can be employed for the AC coefficients
    - H.263 method, all coefficients receive the same quantizer
    - MPEG-2 method, DCT coefficients in the same macroblock can have different quantizers.

1.2.3 Texture Coding

- **Shape-Adaptive DCT-based coding for boundary macroblocks.**
  - **DCT-N transform and its inverse and IDCT-N**

$1D \quad (DCT-N)$

$$F(u) = \sqrt{\frac{2}{N}} C(u) \sum_{i=0}^{N-1} \cos \frac{(2i+1)u\pi}{2N} f(i)$$

$1D \quad (IDCT-N)$

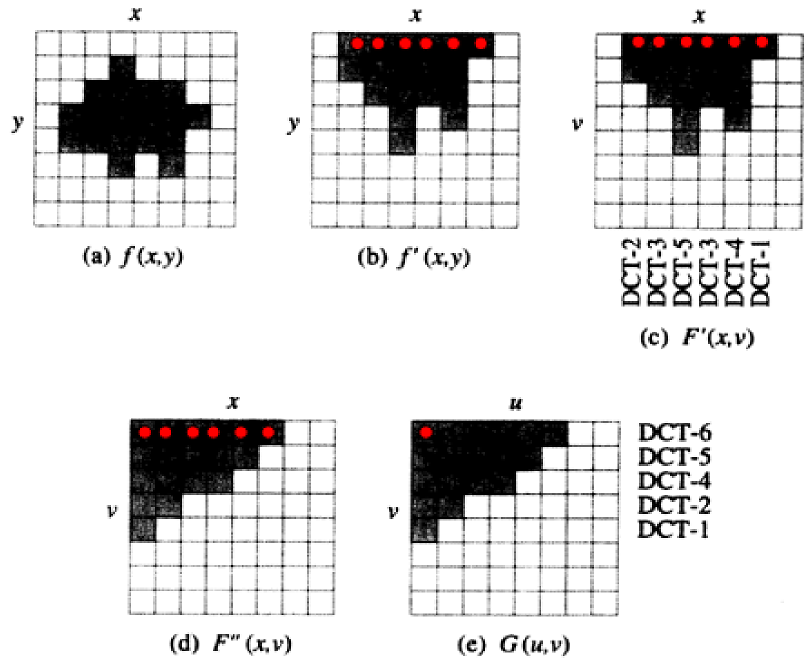
$$\tilde{f}(i) = \sum_{u=0}^{N-1} \sqrt{\frac{2}{N}} C(u) \cos \frac{(2i+1)u\pi}{2N} F(u)$$

where $i = 0, 1, \cdots, N-1; u = 0, 1, \cdots, N-1$

$$C(u) = \begin{cases} \frac{\sqrt{2}}{2} & if\ u = 0 \\ 1 & otherwise \end{cases}$$

1.2.3 Texture Coding

- Texture coding for boundary macroblocks using Shape-Adaptive DCT

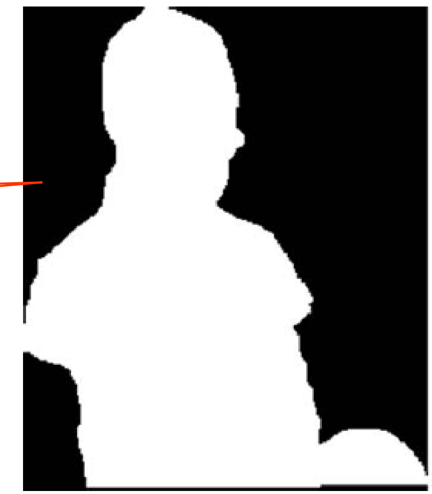


- At decoding time, a binary mask of the original shape is required

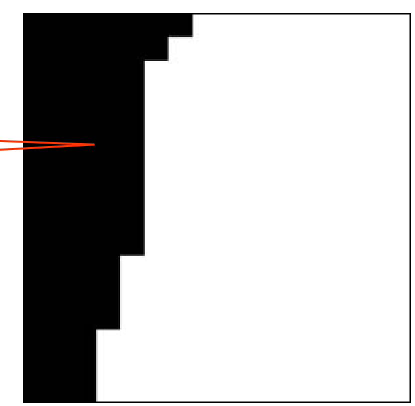
1.2.4 Shape Coding

- **Binary shape coding**

Pixels not in VOP are represented as transparent



For Binary Alpha Block, each pixel must distinguished as transparent or nontransparent.



(Context-based binary Arithmetic Coding)

1.2.4 Shape Coding

- **Binary shape coding**

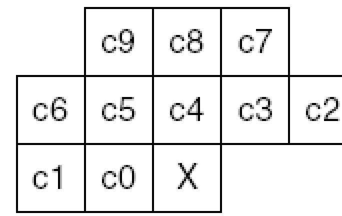
1. Calculate the current context value X

   Each pixel is binary, context X is calculated according to 10 pixels already coded.

   X has 10bits: C9C8C7C6C5C4C3C2C1C0

   X is range in 0~1024

   Lookup tables (MPEG-4 standard) to get the corresponding value.



2. Arithmetic Coding

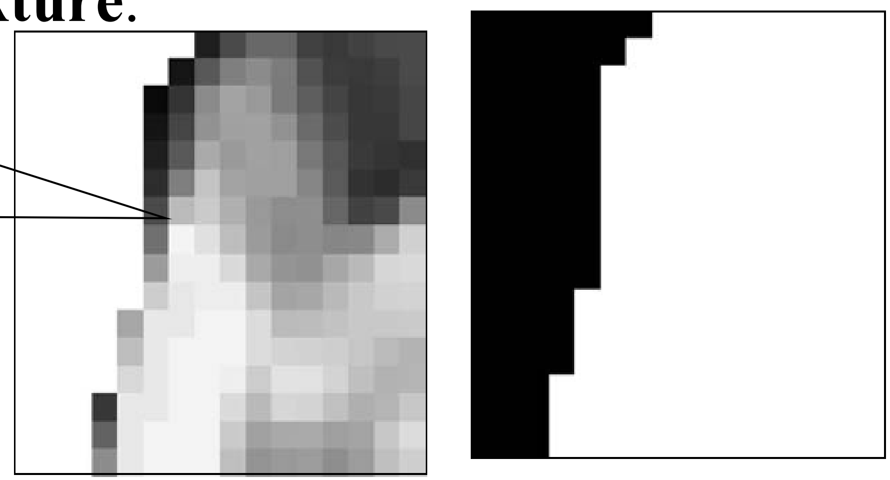
   The value in the lookup table indicate the probability of occurrence for each of the 1024 contexts.

Context (binary)	Context (decimal)	Description	P(0)
0000000000	0	All context pixels are 0	65267/65535 = 0.9959
0000000001	1	c_0 is 1, all others are 0	16468/65535 = 0.2513
1111111111	1023	All context pixels are 1	235/65535 = 0.0036

1.2.4 Shape Coding

- Grayscale shape coding
  - Grayscale is used to describe the transparency of the shape, not the texture.

0~255 degree of transparency, 0 transparent, 255 opaque



  - Raster graphics uses extra bitplanes for an alpha map.
  - Grayscale shape coding is lossy, while binary shape coding is lossless

1.2.5 Static Texture Coding

Wavelet coding for the texture of static objects

- The sub-bands with the lowest frequency are coded using DPCM
  - Prediction of each coefficient is based on three neighbors
- Other sub-bands are based on a multiscale zerotree wavelet coding.

1.2.5 Static Texture Coding

Wavelet coding for the texture of static objects (Cont.)

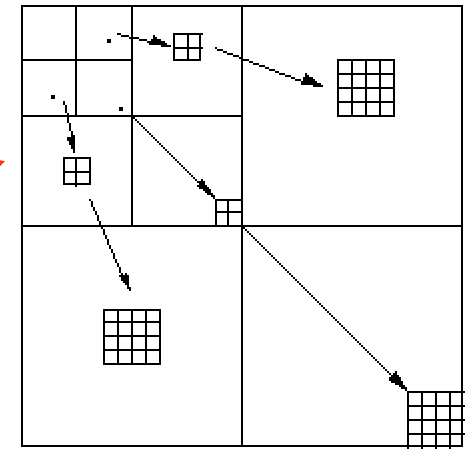
- The multiscale zerotree has a parent-child relation tree (PCR) for each coefficient in the lowest frequency subband
  - The location information of all coefficients is better used.
- A large quantizer is used at first
  - Difference is coded in the next iteration in which a smaller quantizer is employed.
- The most significant coefficients are coded using arithmetic coding.

1.2.5 Static Texture Coding

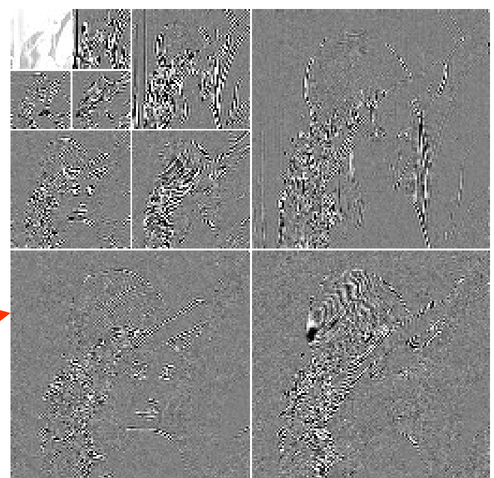
- Zerotree

original image

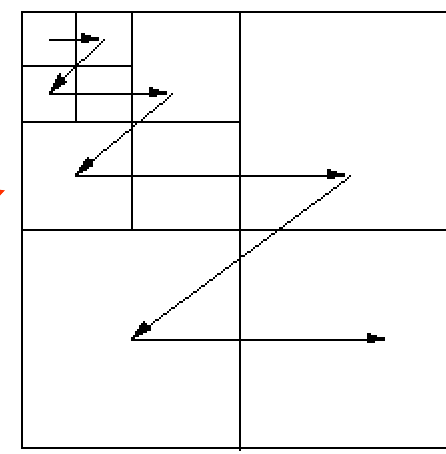
structure



Three level decomposition

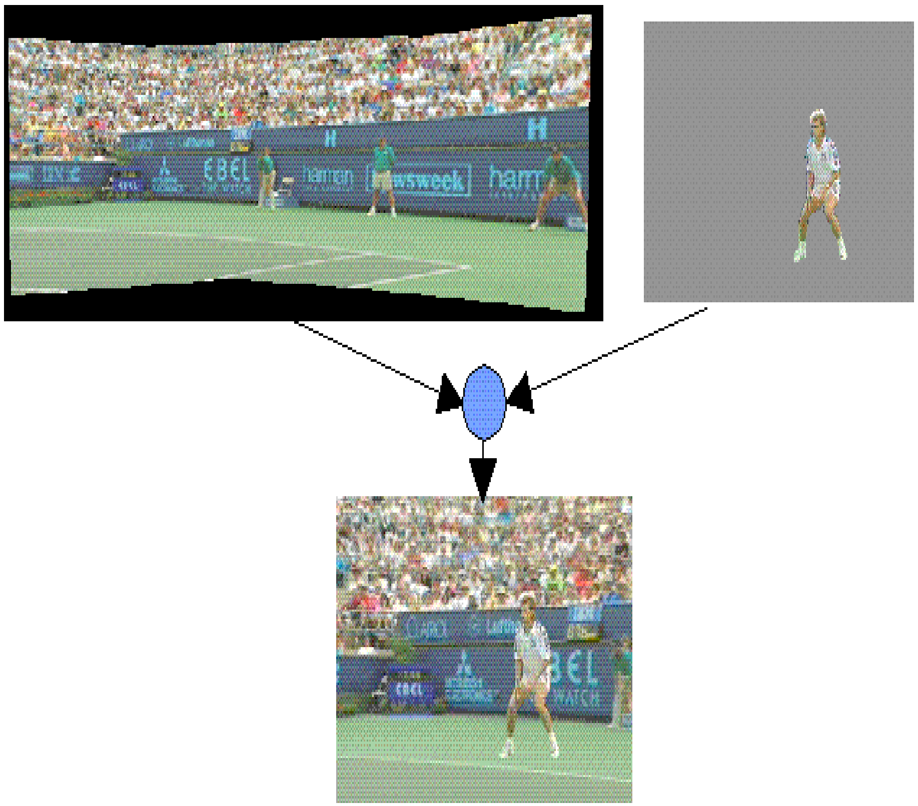


Scan order



1.2.6 Sprite Coding

- Some background can be treated as **static image**
- Foreground is effected by camera movement
- Background can coded separately
- **Foreground objects can be used to create flexible object-based composition of MPEG-4 video**



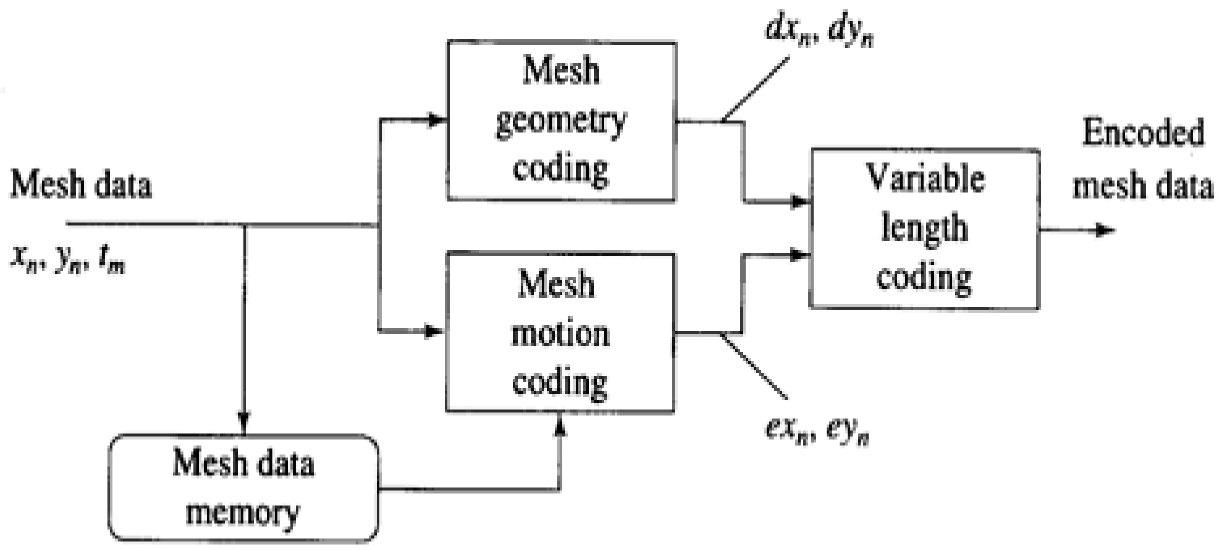
1.2.7 Global Motion Compensation

- Camera motion such as pan, tilt, rotation, and zoom often cause rapid content change between successive frames, block-based motion compensation is not a efficient method for this situation

- GMC is a better choice
- Global Motion Compensation has four major components
  - Global motion estimation
  - Warping and blending
  - Motion trajectory coding
  - Choice of local motion compensation (LMC) or GMC

1.3 Synthetic Object Coding

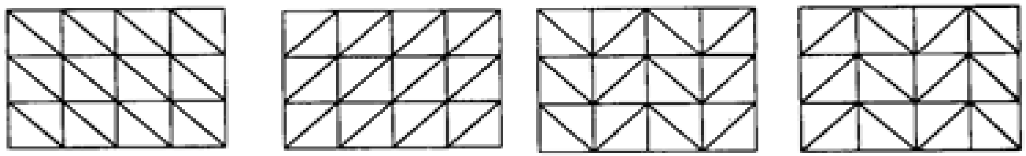
- Synthetic object: objects are created using computer
- 2D Mesh Object Coding



2D Mesh Object Plane (MOP) encoding process.

1.3 Synthetic Object Coding

- 2D Mesh Object Coding
  - 2D Mesh Geometry Coding



(a) Type 0 (b) Type 1 (c) Type 2 (d) Type 3

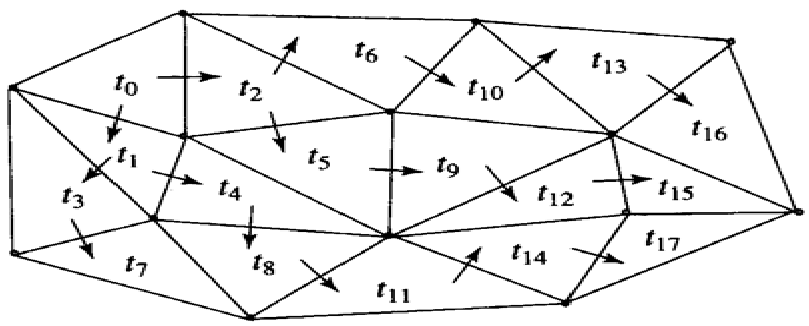
  - Four types of uniform meshes
  - Delaunay mesh is a better object mesh representation
    - Select boundary nodes of the mesh
    - Choose interior nodes
    - Perform Delaunay Triangulation

1.3 Synthetic Object Coding

- 2D Mesh Object Coding
  - 2D Mesh Motion Coding
    - For any MOP triangle i, j ,k . if motion vectors for i, j are known， then motion vector for k can be predicted as

$$Pred_k=0.5(Pred_i+Pred_j)$$

    - When motion vectors of a triangle is coded, uncoded vertex of the neighboring MOP triangle share an edge with the previous triangle is coded, and so on, until all the triangles are coded.



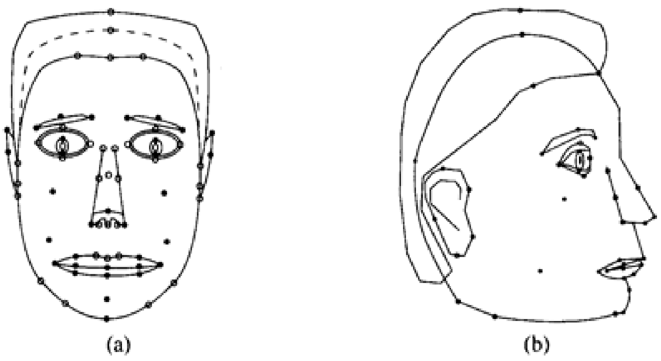
1.3 Synthetic Object Coding

- 2D Mesh Object Coding
  - 2D Object Animation
    - The previous step established a one-to-one mapping between the mesh triangle in the reference MOP and the target MOP
    - Affine transform is used to achieve animated sequence

(a) (b)

1.3 Synthetic Object Coding

- 3D Model-based Coding
  - Face Object Coding and Animation
    - Face models can either be created manually or through computer techniques, the former is cumbersome and inadequate, the latter is not reliable enough.
    - MPEG4 defines 68 Face Animation Parameters (FAPs) to achieve a face model.



1.3 Synthetic Object Coding

- 3D Model-based Coding
  - Body Object Coding and Animation
    - There are 296 body animation parameters (BAPs), The coding of BAPs is similar to that of FAPs.

1.4 MPEG4 Object Types, Profiles and Levels

- Like MPEG2, MPEG4 defines many profiles and levels: Visual profiles, Audio profiles, Graphics profiles, Scene description profiles, Object descriptor profiles

MPEG4 defines the tools needed to create video objects and the ways they can be combined in a scene

Tools for MPEG-4 natural visual object types

Tools	Object types					
	Simple	Core	Main	Simple scalable	N-bit	Scalable Still texture
Basic MC-based Tools	*	*	*	*	*	
B-VOP		*	*	*	*	
Binary shape coding		*	*		*	
Gray-level shape coding			*			
Sprite			*			
Interlace			*			
Temporal scalability(P-VOP)		*	*		*	
Spatial and temporal scalability (rectangular VOP)				*		
N-bit					*	
Scalable still texture						*
Error resilience	*	*	*	*	*	

1.4 MPEG4 Object Types, Profiles and Levels

- Object types and levels in different profiles

Profile	level	Typical Picture size	Bitrate (bits/sec)	Max number of objects
Simple	1	176×144(QCIF)	64K	4
	2	352×288(CIF)	128K	4
	3	352×288(CIF)	384K	4
Core	1	176×144(QCIF)	384K	4
	2	352×288(CIF)	2M	16
Main	1	352×288(CIF)	2M	16
	2	720×576(CCIR601)	15M	32
	3	1920×1080(HDTV)	38.4M	32

Levels in Simple, Core, and Main Visual Profiles

MPEG-4 natural visual object types and Profiles

Object Types	Profiles					
	Simple	Core	Main	Simple Scalable	N-bit	Scalable Texture
Simple	*	*	*	*	*	
Core		*	*		*	
Main			*			
Simple scalable				*		
N-bit					*	
Scalable still texture			*			*

MPEG4 Natural Visual Object Types and Profiles

1.5 MPEG-4 Part10/H.264

- 2001, MPEG and ITU-T VCEG (Video Coding Experts Group) united JVT (Joint Video Team)
- JVT proposed H.264 draft to ISO in 2003
- H.264 offers up to 50% better compression than MPEG-2 and up to 30% better than H.263+ and MPEG-4 advanced simple profile

1.5 MPEG-4 Part10/H.264

- Core Features
  - Entropy decoding
    - Unified-VLC(UVLC) and Context Adaptive VLC (CAVLC)
  - Motion compensation or intra-prediction
    - Variable block size and more accurate motion compensation.
  - Transform, Scan, Quantization
    - Nonlinear quantization and different quantization scales
  - I-Prediction
    - Intra-coded macroblocks are all predicted using neighboring reconstructed pixels
  - In-loop Deblocking Filters
    - Adopts a sophisticated signal-adaptive deblocking filter.

1.5 MPEG-4 Part10/H.264

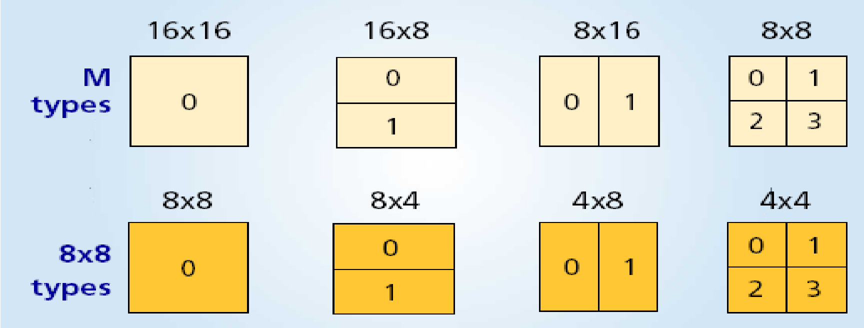
- **Deblocking Filter in H.264 can obtain pleasing results**

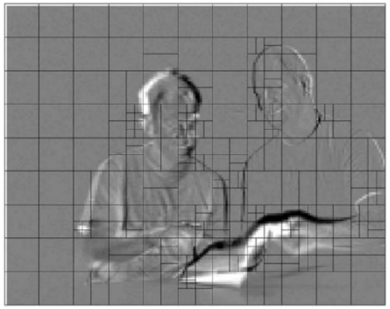
Non Deblocking

After Deblocking

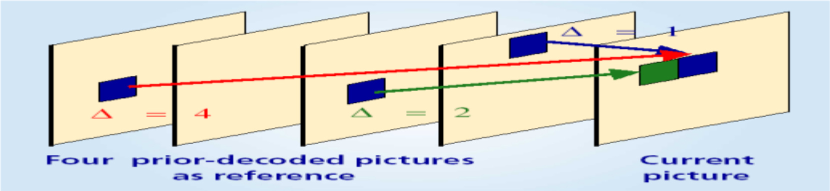
1.5 MPEG-4 Part10/H.264

- **Inter prediction**
  - Tree-structured Motion Compensation
  - H.264 supports different block size, block size can down to 4*4





  - Select the optimal block size, minimize the difference between current an reference frame.
  - P frame can use more than one previous frames as reference frames.



Network & Media Laboratory

2. MPEG-7

Overview of MPEG-7(1)

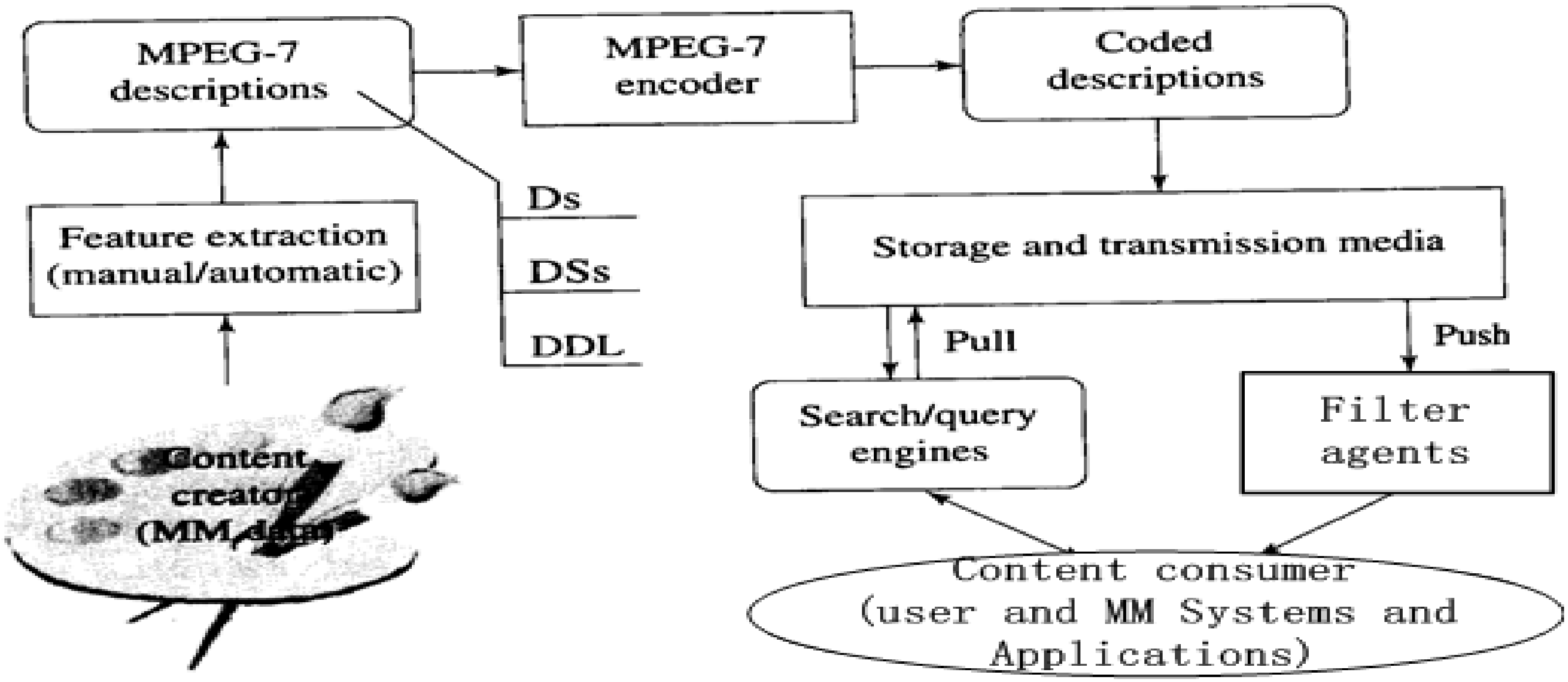
- More and more multimedia content becomes an integral part of various applications, effective and efficient retrieval becomes a primary concern.
- MPEG7 is to satisfy the need of audiovisual content-based retrieval.
- MPEG7 was initialized in 1998, finished in 2001.
- MPEG7 supports a variety of multimedia applications.
- MPEG7 doesn't describe any feature extracting methods. Its formal name is "multimedia content description interface".

Overview of MPEG-7(2)

- MPEG-7
  - Descriptors (D), Description Schemes (DS),
  - Description Definition Language(DDL)
- Descriptor (D)
  - Color, Texture, Shape, motion, localization
- Description Schemes (DS)
  - Basic elements, content management, content description, navigation and access
- XML Schema Language and MPEG7 Extensions

Overview of MPEG-7(2)

- Applications using MPEG7



3. MPEG-21

Overview of MPEG-21

- MPEG-21 is to define a uniform way to define, identify, describe, manage, and protect multimedia data.
- MPEG21 has 7 key parts
  - Digital item declaration
  - Digital item identification and description
  - Content management and usage
  - Intellectual property management and protection
  - Terminal and networks
  - Content representation
  - Event reporting

The End!

Thanks!