CARDIFF UNIVERSITY
EXAMINATION PAPER

Academic Year:	2011/2012
Examination Period:	Autumn
Examination Paper Number:	CM0340 Solutions
Examination Paper Title:	Multimedia
Duration:	2 hours

Do not turn this page over until instructed to do so by the Senior Invigilator.

Structure of Examination Paper:

There are 12 pages.
There are 4 questions in total.
There are no appendices.
The maximum mark for the examination paper is 81 and the mark obtainable for a question or part of a question is shown in brackets alongside the question.

Students to be provided with:

The following items of stationery are to be provided:
ONE answer book.

Instructions to Students:

Answer 3 questions.
The use of calculators is permitted in this examination.
The use of translation dictionaries between English or Welsh and a foreign language bearing an appropriate departmental stamp is permitted in this examination.

Q1. (a) *What is* MIDI?

Definition:

A protocol that enables computers, synthesizers, keyboards, and other musical devices to communicate with each other. [1]

1 Mark — Bookwork

(b) *What features of MIDI make it* suitable *for use in the* MPEG-4 audio compression standard*?*

- MIDI is very low bandwidth when compared to audio. Sounds synthesised at client only control data transmitted [1]
- MIDI can control many performance aspects [1]

2 Marks — Bookwork

(c) *Briefly outline the MPEG-4* structured audio *standard.*

MPEG-4 Structured Audio Tools

MPEG-4 comprises 6 Structured Audio tools:

SAOL the Structured Audio Orchestra Language: SAOL is the central part of the Structured Audio toolset. It is a software-synthesis language — defines how to make the sounds [1]

SASL the Structured Audio Score Language: MIDI controls how SAOL makes the sounds [1]

SASBF the Structured Audio Sample Bank Format: SASBF is a format for efficiently transmitting banks of sound samples to be used in wavetable, or sampling, synthesis. [1]

Set of MIDI semantics which describes how to control SAOL and SASL scripts (MIDI based). [1]

Scheduler which describes how to take the above parts and create sound: It is a set of carefully defined and somewhat complicated instructions that specify how SAOL is used to create sound when it is driven by MIDI or SASL. [1]

AudioBIFS part of BIFS, which lets you make audio soundtracks in MPEG-4 using a variety of tools and effects-processing techniques [1]

6 Marks — Bookwork

(d) *What* features *of MIDI make it suitable for* controlling *software or hardware devices*?

- Basic syntax of midi is all about control playing notes, setting up sounds etc. [1]
- A wide number of specialised control messages — some set like sustain, modulation, pitch bend others freely assignable. [1]
- Wide range of controllers available — e.g. built in to keyboards, specialist hardware, software reassignable. [1]
- Midi System Real-time Messages for control of timing/syncing e.g. SMPTE, Midi Time Code. [1]
- System exclusive command set makes MIDI completely ”extensible” to control any device. [1]
- External hardware to convert MIDI to/from other formats e.g. Pitch to Midi converters, Midi to Control Voltage (analogue synths), Motion Capture to MIDI!. [1]

6 Marks — unseen. Assimilation of various aspects of bookwork

(e) *Outline how you could utilise MIDI in the following situations:*

- *A MIDI controller that has to be able to accommodate a range of values in excess of 255.*
  Use **two** MIDI controllers in tandem. [1]
  One sets MSB and other LSB for the number - This allows 16-bit address space of values. [1]

  2 Marks — unseen.

- *A MIDI sampler has limited memory but needs to be configured to be play trumpet sounds at one instant but also violin sounds at other instants.*

  Use MIDI Sample Dump Standard (SDS) to transfer trumpet or violin samples as required. [2]

  2 Marks — unseen.

- *A MIDI controlled Avatar where the avatars limbs and facial features are required to be controlled via MIDI.*
  Assign a MIDI **continuous controller** for each action, e.g. Controller 1 for lower left leg, Controller 2 upper left leg, ..., Controller n mouth open/close ... [1]
  Send MIDI control data accordingly [1]

  2 Marks — unseen.

- *A MIDI controlled toy car where MIDI needs to control the starting and stopping of the car and also its speed.*

  Use MIDI Midi System Real-time Messages (Sync) messages to start/stop car — usually used to start/stop/continue recording sequences/devices. [2]
  Use MIDI tempo message to control car speed — usually used to set tempo of a piece of MIDI music (in Beats per minute format) [1]

  3 Marks — unseen.

- *A robot is to be controlled via MIDI. The robot has custom configurable circuitry that control its movement and sensors. The circuitry has to be initialised via MIDI.*

  Set up a MIDI system exclusive (Sysex) message – System dependent creation of messages. [2]
  Assume/configure circuitry to map parameter values from Sysex to 'drivers' for circuits [1]

  3 Marks — unseen.

  12 Marks total for part (e).

27 Marks Question Total

Q2. (a) *How does the* human eye *sense colour? What characteristics of the human visual system can be exploited for the compression of colour images and video?*

The eye is basically sensitive to colour and intensity

- Retina of the eye has 'neurons' on which light is focus. Each neuron is either a rod or a cone. [1]
- Rods are not sensitive to colour - sense intensity (monochrome). [1]
- Cones come in 3 types: The first responds most to light of long wavelengths, red/yellowish colours. The second type responds most to light of medium-wavelength, peaking at a green colour, The third type responds most to short-wavelength light, of a bluish colour. [1]
- Each responds differently — Non linearly and not equally for RGB differently to various frequencies of light. [1]
- Compression in image video uses the fact that intensity (monochrome) can be modelled in high resolution and colour modelled in lower resolution and non-linearly w.r.t colour sensitivity. [1]

5 Marks - Bookwork

(b) *What is the YIQ color model ? How is compression achieved with YIQ in Analog NTSC Video and Digital MPEG Video?*

YIQ Colour Model

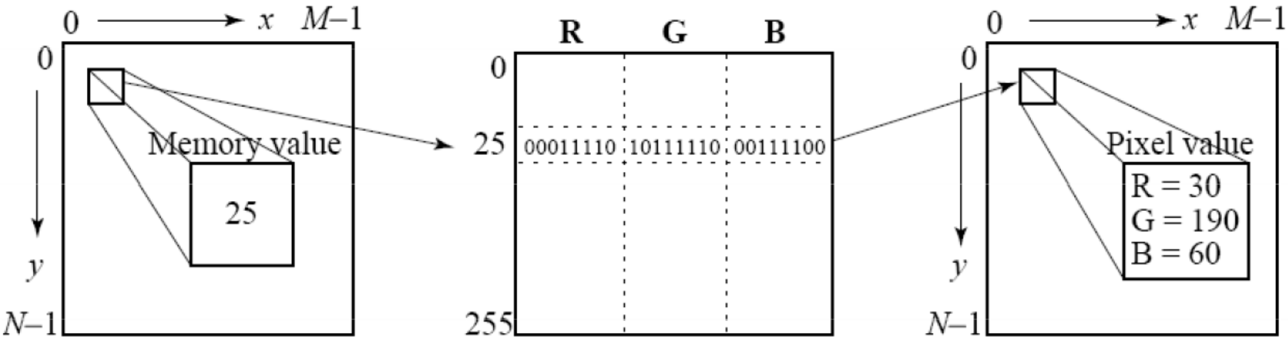
- YIQ is used primarily in colour TV broadcasting (although it is downward compatible with B/W TV.):
  - Y (luminance) is the CIE Y primary. [1]
  - I is red-orange axis, Q is roughly orthogonal to I. [1]
- Eye is most sensitive to Y, next to I, next to Q. [1]
- In NTSC analog 4 MHz bandwidth is allocated to Y, 1.5 MHz to I, 0.6 MHz to Q. [1]
- In digital video, Chroma subsampling ratio: 4:1:1 [1]

5 Marks - Bookwork

(c) *What is a* colour look-up table *and how is it used to represent colour?*

Colour Look-Up Tables (LUTs)

- Store only the index of the colour LUT for each pixel. [1]
- Look up the table to find the colour (RGB) for the index [1]



[3]

5 Marks - Bookwork

Give an advantage and a disadvantage of this representation with respect to true colour (24-bit) colour.

Advantage : Use up significantly less memory than full 24-bit colour. [1]
Disdvantage : Restricted number of colours available. [1]

2 Marks - Bookwork

How do you convert from 24-bit colour to an 8-bit colour look up table representation?

- LUT needs to be built when converting 24-bit colour images to 8-bit: grouping similar colours (each group assigned a colour entry) [1]

1 Mark - Bookwork

(d) *Describe how colour look-up tables can be used to implement simple computer animations. Illustrate you answer with the following example: In an 7x7 image you have to animate a 3x3 red square moving from left to right at a rate of 2 pixels per frame. The square is centred vertically within the image and the image background is black.*

Solution Sketch:

- Need to construct a image indices to LUT and Manipulate LUT values rather than image values. [1]
- Need to set up the 8x8 image as follows:
  - Rows 1,2,6,7 always black so LUT value 0 set to (0,0,0) in LUT [1]
  - Rows 3–5 Need to have indices that take on black (0,0,0) or red (255,0,0) as animation proceeds. [1]
  - there is a 2 pixel step at each from so there is an *overlap* of 1 'column' of the block at each iteration. Need another LUT entry to accommodate this in the animation. [1]

0	0	0	0	0	0	0
0	0	0	0	0	0	0
1	1	2	3	4	5	6
1	1	2	3	4	5	6
1	1	2	3	4	5	6
0	0	0	0	0	0	0
0	0	0	0	0	0	0

- Animation proceeds as follows:
  - Step 1: Set 1 and 2 to red 3- black [1]
  - Step 2: Set 1 to black and 2, 3 and 4 to red (rest remain black) [1]
  - Step 3: Set 2 and 3 to black and 5 and 6 to red (4 stays red, rest remain black) [1]
  - similarly for rest of animation. [1]

8 Marks - Unseen

Give a limitation of colour look-up table animation
Limitation:

- Even greater limitation of colour values than normal LUTs as several values need to reserved to display the animated object/background colours [1]
  or
- Need fast LUT memory [1]

1 Mark - Unseen

27 Marks Question Total

Q3. (a) *GIF and JPEG are two commonly used image representations. What images are suitable to be represented as GIF and JPEG? Do they usually use lossless or lossy compression?State the major compression algorithm (for lossless) or the lossy steps of the algorithm (for lossy) for each.*

Lossless or lossy:

GIF : Lossless. [1]

JPEG : Lossy. [1]

Key algorithms:

GIF : Key algorithm is LZW (lossless) [1]

JPEG : Lossy steps involve quantisation and chroma subsampling [1]

4 Marks — Bookwork

(b) *Briefly describe the four basic types of data redundancy that data compression algorithms can apply to audio, image and video signals.*

4 Types of Compression:

- Temporal – in 1D data, 1D signals (Audio), 3D temporal frames in Video. [2]
- Spatial – correlation between neighbouring pixels or data items. [2]
- Spectral – correlation between colour or luminescence components. This uses the frequency domain to exploit relationships between frequency of change in data. [2]
- Psycho-visual, psycho-acoustic – exploit perceptual properties of the human visual system or aural system to compress data. [2]

8 Marks Bookwork

(c) *Encode the following steam of characters using decimal arithmetic coding compression:*

`MULTI`

You may assume that characters occur with probabilities of `M` *= 0.1,* `U` *= 0.3,* `L` *= 0.3,* `T` *= 0.2 and* `I` *= 0.1.*

- Sort Data into largest probabilities first and make cumulative probabilities
  0 — `U` — 0.3 — `L` — 0.6 — `T` — 0.8 — `M` - 0.9 — `I` — 1.0
  There are only 5 Characters so there are 5 segments of width determined by the probability of the related character. [3]
- The first character to encoded is `M` which is in the range 0.8 — 0.9, therefore the range of the final codeword is in the range 0.8 to 0.89999.. [1]
- Each subsequent character subdivides the range 0.8 0.9 [1]
- SO after coding `M` we get
  0.8 — `U` — 0.83 — `L` — 0.86 — `T` — 0.88 — `M` — 0.89 — `I` — 0.9 [1]
- So to code `U` we get range 0.8 — 0.83 So we subdivide this range
  0 — `U` — 0.809 — `L` — 0.818 — `T` — 0.824 — `M` — 0.827 — `I` — 0.83 [1]
- Next range is for `L` so we split in the range 0.809 — 0.818
  0.809 — `U` — 0.8117 — `L` — 0.8144 — `T` — 0.8162 — `M` — 0.8171 — `I` — 0.818 [1]
- Next Character is `T` so range is from 0.8144 — 0.8162 so we get
  0.8144 — `U` — 0.81494 — `L` — 0.81548 — `T` — 0.81584 — `M` — 0.81602 — `I` — 0.8162 [1]
- Final Char is `I` which is in the range 0.81602 — 0.8162 [1]
- So the completed *codeword* is any number in the range
  0.81602 $<=$ *codeword* $<$ 0.8162 [2]

12 Marks Unseen

(d) *Show how your solution to (c) would be decoded.*

Assume Codeword is 0.8161

Sort Data into largest probabilities first and make cumulative probabilities [1]

Code can determine first character is `M` since it is in the Range 0.8 — 0.9 [1]

By expanding interval we can see that next char must be an E as it is in the range 0.8 — 0.83 and so on for all other intervals. [1]

3 Marks Unseen

27 Marks Question Total

Q4. (a) *List* two *psychological phenomena that are exploited in MPEG audio compression.*

These involve *frequency masking* and *temporal masking*. [2]

Briefly explain their meanings.

Frequency Masking : When an audio signal consists of multiple frequencies the sensitivity of the ear changes with the relative amplitude of the signals. If the frequencies are close and the amplitude of one is less than the other close frequency then the second frequency may not be heard. [1]

Temporal Masking : After the ear hears a loud sound, consisting of multiple frequencies, it takes a further short while before it can hear a quieter sound close in frequency. [1]

4 Marks: Bookwork

(b) *How does MPEG audio compression* implement *methods which use the above psychological phenomena?*

Both use the following basic process

- (MPEG employs Frequency Masking and may employ temporal masking (MP3 Level 3).)
- Uses Fourier Transform (FFT) to perform analysis. [1]
- Spilt signal into frequency sub-bands [1]

[2]

Frequency Masking :

- Determine amount of masking for each band caused by nearby bands with a lookup table of scaling ratios:
  - Input: set hearing thresholds and sub-band masking properties (model dependent) and scaling factors (above).

[3]

Temporal Masking :

- Same sub-band coding technique as *Frequency Masking* used.
- Two *windows* of sample length 36 and 12 with 50% overlap used to encode temporal aspects
- Three short 12 sample blocks concatenated to make a 36 sample length block

[3]

8 Marks: Rationalisation of Bookwork

(c) *What are the* fundamental differences *between* MPEG *and* Dolby audio compression *algorithms?*

MPEG/Dolby algorithm difference:

- MPEG perceptual coders control quantisation accuracy of each sub-band by computing bit numbers for each sample.
- MPEG needs to store each quantise value with each sample.
- MPEG Decoder uses this information to dequantise:
  forward adaptive bit allocation
- **DOLBY**: Use **fixed bit rate allocation** for each subband based on characteristics of the ear.

[4]

Give an advantage and a disadvantage of Dolby with respect to MPEG audio compression.

Dolby Advantage:

- Uses *fixed bit rate allocation* so no need to send such info with each frame.

[1]

Dolby Disdvantage:

- **DOLBY** encoders and decoder **both** need psychoacoustic
  model information *or* Decoder not independent of encoder method.

[1]

6 Marks: Bookwork

(d) *Given two stereo channels of audio:*

Left Channel:	*112*	*102*	*113*	*114*	*115*	*127*	*136*	*144*
Right Channel:	*112*	*114*	*116*	*104*	*124*	*120*	*122*	*133*

i. *Apply Middle/Side (MS) stereo coding to the sequence.*

Basic Idea:

- **Middle** — sum of left and right channels
- **Side** — difference of left and right channels.

Middle:	224	216	229	218	239	247	258	277
Side:	0	-12	-3	10	-9	7	14	11

[3]

3 Marks Applied Unseen Problem

ii. *How may this result be employed to achieve compression?*

Encode side in less bits as its essentially Differential Pulse Code Modulation. [1]
Use specially tuned threshold values to compress the side channel signal further. [1]
Code Middle in normal (for audio) 16 bits [1]
Code Side in reduced number of bits. Needs to be ***signed*** so in the above 5 bits (minimum) needs (usually have an 8 bit range) [1]

4 Marks Unseen (Briefly discussed part in notes needs to applied to)

iii. *Give two potential problems of this coding method.*

- Two Audio signals are 16-bit so possible overflow in Middle (e.g. above if numbers we 8-bit last would two value overflow) Solution: Could be to average signals rather than sum. [1]
- Need signed 8-bit value for Side so only 7 bit range for difference - so either quantise or overflow error possible without care (special tuned threshold in MPEG). [1]

2 Marks Unseen

27 Marks Question Total