

CMPT 365 Midterm Examination Summer 2011

(Please answer all questions for a total of 100 points)

1. (30 points) [Short Answer Questions]

- (a) [6 points] Except *cut*, name three other Video Transitions. *slide, wipe, dissolve, fade in/out*
- (b) [6 points] A CD-quality digital sound is usually sampled at 44.1 KHz. Why do we not use a lower rate, say 8 KHz for it? *> Nyquist rate 2×20 KHz; otherwise, alias*
- (c) [8 points] If the paint on the surface of an object is Yellow, (i) What color would it appear to be under a Blue light? Why? (ii) What color would it appear to be under a Cyan light? Why? *(i) Black, blue subtracted/absorbed. (ii) Green,*
- (d) [10 points] Briefly describe two examples where *Differential Coding* is used. *Same reason Lossless JPEG, DC in lossy JPEG, MV coding in video, P-frame coding, ...*

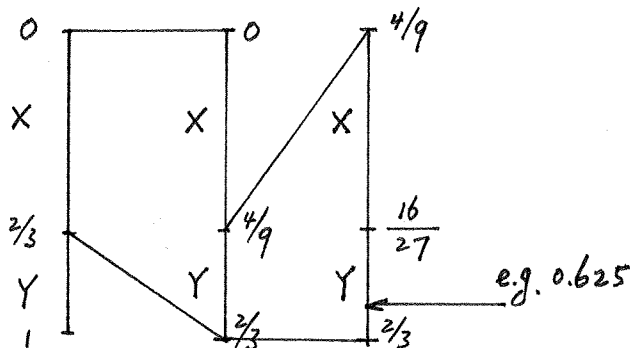
2. (15 points) [Video Compression]

- (a) [7 points] Briefly describe how Hierarchical Search of motion vectors works.
- (b) [8 points] Give three reasons why Hierarchical Search of motion vectors is more efficient than Sequential Search. *(i) smaller search area (p) in low-res (ii) smaller MB in low-res (iii) only 3×3 search areas in higher-res.*

3. (25 points) [Arithmetic Coding]

- (a) [5 points] Arithmetic coding and Huffman coding are two of the most used lossless compression methods in multimedia. What is the advantage of Arithmetic coding over Huffman coding? *Can have less than one bit per symbol. $R = \lceil \log_2 \frac{1}{\prod_i P_i} \rceil$*
- (b) Suppose we have a vocabulary of two symbols: X and Y. The probabilities of them are $P_X = 2/3$ and $P_Y = 1/3$.
- [12 points] (i) Provide details (draw a diagram and briefly explain) how the arithmetic encoder works for encoding the string *XY*.
- [8 points] (ii) Provide details how the decoder works in this case.
- (Note: You may assume that the encoder and decoder have agreed to handle only 3 characters at a time, so no EOF is needed.)

(i)



(ii) Decoder

Generate the same graph. start with the code for 0.625, step by step gets *XY*.

4. (30 points) [Discrete Cosine Transform]

The 2D DCT can be *separated* into a sequence of two, 1D DCT steps:

$$G(i, v) = \frac{1}{2}C(v) \sum_{j=0}^7 \left[\left(\cos \frac{(2j+1)v\pi}{16} \right) \cdot f(i, j) \right] \quad (1)$$

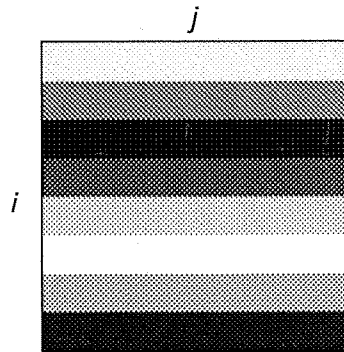
$$F(u, v) = \frac{1}{2}C(u) \sum_{i=0}^7 \left[\left(\cos \frac{(2i+1)u\pi}{16} \right) \cdot G(i, v) \right] \quad (2)$$

The constants $C(u)$ and $C(v)$ are $\frac{\sqrt{2}}{2}$ when u or v is 0; otherwise, they are 1.

- (a) [5 points] What does the 2D DCT do in JPEG image compression? *Decompose f into DC & AC components*
- (b) [5 points] How does the above implementation of the 2D DCT reduce the computational complexity as compared to its straightforward implementation? *8x8 → 8+8 for each F(u,v)*
- (c) Below is an 8 × 8 black and white image.

Its gray-level intensity is $128 \times [1 + \cos \frac{(2i+1) \cdot 3\pi}{16}]$, where $i = 0, 1, \dots, 7$. In other word, its change is synchronized with the cosine function that completes 1.5 cycles in the i -dimension. Apply the above transform to this image.

- (i) [5 points] What is the DC value? *8 × 128 = 1024*
- (ii) [5 points] Which AC has the largest value? *F(3, 0)*
- (iii) [10 points] Show all values of $G(i, v)$. [Hint: You do not need to work out the actual real numbers. Simply use various cosines for the non-zero values.]



		v							
		0	1	2	3	4	5	6	7
i	0	$2\sqrt{2} \cdot 128 \cdot (1 + \cos \frac{3\pi}{16})$	0						0
	1	$2\sqrt{2} \cdot 128 \cdot (1 + \cos \frac{9\pi}{16})$	0						0
	2	$2\sqrt{2} \cdot 128 \cdot (1 + \cos \frac{15\pi}{16})$.						.
	3	$2\sqrt{2} \cdot 128 \cdot (1 + \cos \frac{21\pi}{16})$.						.
	4	$2\sqrt{2} \cdot 128 \cdot (1 + \cos \frac{27\pi}{16})$							
	5	$2\sqrt{2} \cdot 128 \cdot (1 + \cos \frac{33\pi}{16})$							
	6	$2\sqrt{2} \cdot 128 \cdot (1 + \cos \frac{39\pi}{16})$							
	7	$2\sqrt{2} \cdot 128 \cdot (1 + \cos \frac{45\pi}{16})$	0						0