2007 IIB 4F8 Dr J Lasenby
jl@eng.cam.ac.uk

Solutions: 4F8 2007

ENGINEERING TRIPOS PART IIB

Monday 23 April 2007 2.30 to 4

Module 4F8

IMAGE PROCESSING AND IMAGE CODING

Answer not more than ***three*** *questions.*

All questions carry the same number of marks.

The ***approximate*** *percentage of marks allocated to each part of a question is indicated in the right margin.*

There are no attachments.

STATIONERY REQUIREMENTS
Single-sided script paper

SPECIAL REQUIREMENTS
Engineering Data Book
CUED approved calculator allowed.

You may not start to read the questions printed on the subsequent pages of this question paper until instructed that you may do so by the Invigilator

1 (a) (i) We can write the sampling function $s(u_1,u_2)$ as the sum of two rectangular grid sampling functions:

$$s(u_1,u_2) = s_1(u_1,u_2) + s_2(u_1,u_2)$$

where (black)

$$s_1(u_1,u_2) = \sum_{n_1,n_2} \delta(u_1 - n_1\Delta_1, u_2 - 2n_2\Delta_2)$$

and (white)

$$s_2(u_1,u_2) = \sum_{n_1,n_2} \delta(u_1 - \left[n_1 - \frac{1}{4}\right]\Delta_1, u_2 - (2n_2+1)\Delta_2)$$

corresponding to the black and white circles in figure 1

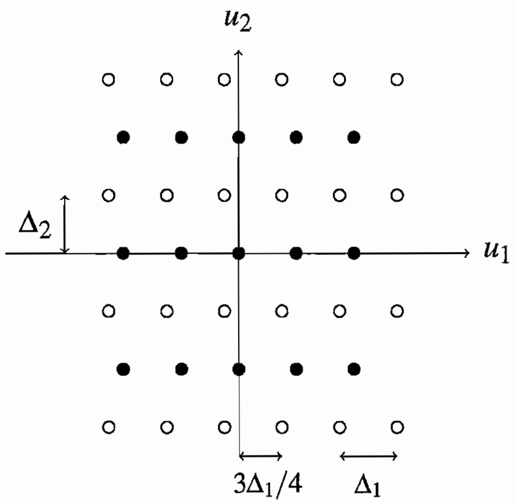

Fig. 1

Therefore

$$s(u_1,u_2) = \sum_{n_1,n_2} \delta(u_1 - n_1\Delta_1, u_2 - 2n_2\Delta_2) + \sum_{n_1,n_2} \delta(u_1 - \left[n_1 - \frac{1}{4}\right]\Delta_1, u_2 - (2n_2+1)\Delta_2)$$

[15%]

(cont.

(ii) $s_1(u_1,u_2)$ and $s_2(u_1,u_2)$ are periodic so we can express them as Fourier series,

$$s_1(u_1,u_2) = \sum_{p_1=-\infty}^{\infty} \sum_{p_2=-\infty}^{\infty} c_1(p_1,p_2)\, e^{j(p_1\Omega_1 u_1 + p_2\frac{\Omega_2}{2}u_2)}$$

$$s_2(u_1,u_2) = \sum_{p_1=-\infty}^{\infty} \sum_{p_2=-\infty}^{\infty} c_2(p_1,p_2)\, e^{j(p_1\Omega_1 u_1 + p_2\frac{\Omega_2}{2}u_2)}$$

where $\Omega_1 = \frac{2\pi}{\Delta_1}$ and $\Omega_2 = \frac{2\pi}{\Delta_2}$ and c_1 and c_2 are given by

$$c_1(p_1,p_2) = \frac{1}{2\Delta_1\Delta_2}\int_{-\Delta_2}^{\Delta_2}\int_{-\frac{\Delta_1}{2}}^{\frac{\Delta_1}{2}} \sum_{p_1=-\infty}^{\infty}\sum_{p_2=-\infty}^{\infty} \delta(u_1 - n_1\Delta_1, u_2 - 2n_2\Delta_2)\, e^{-j(p_1\Omega_1 u_1 + p_2\frac{\Omega_2}{2}u_2)}\, du_1\, du_2$$

$$c_2(p_1,p_2) = \frac{1}{2\Delta_1\Delta_2}\int_{-\Delta_1}^{0}\int_{-\Delta_2/2}^{3\Delta_2/2} \sum_{n_1,n_2} \delta(u_1 - [n_1 - \frac{1}{4}]\Delta_1, u_2 - (2n_2+1)\Delta_2)\, e^{-j(p_1\Omega_1 u_1 + p_2\frac{\Omega_2}{2}u_2)}\, du_2\, du_1$$

Note the limits – choose to integrate over a particular section (range one period) which contains just one grid point.

For c_1 there is only a non-zero contribution when $n_1 = n_2 = 0$, so that

$$c_1(p_1,p_2) = \frac{1}{2\Delta_1\Delta_2} \quad \text{for all } p$$

For c_2 there is only a non-zero contribution when $n_1 = n_2 = 0$, so that

$$c_2(p_1,p_2) = \frac{1}{2\Delta_1\Delta_2} e^{-j(-p_1\Omega_1\frac{1}{4}\Delta_1 + p_2\frac{\Omega_2}{2}\Delta_2)} \text{for all } p$$

giving

$$c_2(p_1,p_2) = \frac{1}{2\Delta_1\Delta_2} e^{j\pi(\frac{p_1}{2} - p_2)}$$

Note that the Fourier coefficients c_2 can also be obtained straightforwardly from the standard rectangular grid given by coefficients c_1 via the shift theorem, i.e. shifts of $-\Delta_1/4$ in the u_1 direction and Δ_2 in the u_2 direction.

Therefore the Fourier coefficients for the sampling grid are

$$c(p_1,p_2) = \frac{1}{2\Delta_1\Delta_2}\left\{1 + e^{j\pi(\frac{p_1}{2} - p_2)}\right\}$$

[20%]

(TURN OVER for continuation of Question 1

(iii) Since the sampled signal can now be written as

$$g_s(u_1,u_2) = \frac{1}{2\Delta_1\Delta_2}\sum_{p_1,p_2} g(u_1,u_2)\left\{1+e^{j\pi(\frac{p_1}{2}-p_2)}\right\}e^{j(p_1\Omega_1 u_1+p_2(\Omega_2/2)u_2)}$$

we can take FTs of this and use the shift theorem to give

$$G_s(\omega_1,\omega_2) = \frac{1}{2\Delta_1\Delta_2}\sum_{p_1,p_2}\left\{1+e^{j\pi(p_2-\frac{p_1}{2})}\right\}G(\omega_1-p_1\Omega_1,\omega_2-p_2(\Omega_2/2))$$

Thus if we represent the modulus of the FT of g by circular contours (larger amplitude implies more contours), the FT of the sampled signal is illustrated in the figure 2

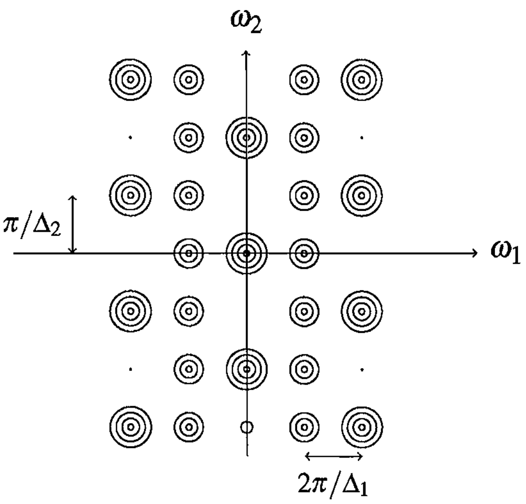

Fig. 2

[25%]

(b) (i)

$$H(\omega_1,\omega_2) = H_u - H_l$$

where

$$H_u(\omega_1,\omega_2) = \begin{cases} 1 & \text{if } |\omega_1| < \Omega_{U1} \text{ and } |\omega_2| < \Omega_{U2} \\ 0 & \text{otherwise} \end{cases}$$

(cont.

$$H_l(\omega_1,\omega_2) = \begin{cases} 1 & \text{if } |\omega_1| < \Omega_{L1} \text{ and } |\omega_2| < \Omega_{U2} \\ 0 & \text{otherwise} \end{cases}$$

[15%]

(ii) Taking the IFT of $H(\omega_1,\omega_2)$ gives us

$$h(n_1,n_2) = \frac{\Delta_1\Delta_2}{(2\pi)^2}\int_{-\pi/\Delta_2}^{\pi/\Delta_2}\int_{-\pi/\Delta_1}^{\pi/\Delta_1} H(\omega_1,\omega_2)e^{j(\omega_1 n_1\Delta_1+\omega_2 n_2\Delta_2)}\,d\omega_1\,d\omega_2$$

$$= \frac{\Delta_1\Delta_2}{(2\pi)^2}\int_{-\Omega_{U2}}^{\Omega_{U2}}\int_{-\Omega_{U1}}^{-\Omega_{L1}} e^{j(\omega_1 n_1\Delta_1+\omega_2 n_2\Delta_2)}\,d\omega_1\,d\omega_2$$

$$+ \frac{\Delta_1\Delta_2}{(2\pi)^2}\int_{-\Omega_{U2}}^{\Omega_{U2}}\int_{\Omega_{L1}}^{\Omega_{U1}} e^{j(\omega_1 n_1\Delta_1+\omega_2 n_2\Delta_2)}\,d\omega_1\,d\omega_2$$

Evaluating these integrals gives

$$\frac{\Delta_1\Delta_2}{(2\pi)^2}\left\{\left[\frac{e^{j\omega_1 n_1\Delta_1}}{jn_1\Delta_1}\right]_{-\Omega_{U1}}^{-\Omega_{L1}}\left[\frac{e^{j\omega_2 n_2\Delta_2}}{jn_2\Delta_2}\right]_{-\Omega_{U2}}^{\Omega_{U2}} + \left[\frac{e^{j\omega_1 n_1\Delta_1}}{jn_1\Delta_1}\right]_{\Omega_{L1}}^{\Omega_{U1}}\left[\frac{e^{j\omega_2 n_2\Delta_2}}{jn_2\Delta_2}\right]_{-\Omega_{U2}}^{\Omega_{U2}}\right\}$$

$$= \frac{\Delta_1\Delta_2}{(2\pi)^2}2\Omega_{U2}\mathrm{sinc}(n_2\Delta_2\Omega_{U2})\left\{\frac{-e^{j\Omega_{L1}n_1\Delta_1}+e^{-j\Omega_{L1}n_1\Delta_1}}{jn_1\Delta_1} + \frac{-e^{j\Omega_{U1}n_1\Delta_1}-e^{-j\Omega_{U1}n_1\Delta_1}}{jn_1\Delta_1}\right\}$$

$$= \frac{\Delta_1\Delta_2}{(\pi)^2}\Omega_{U2}\mathrm{sinc}(n_2\Delta_2\Omega_{U2})\left\{\Omega_{U1}\mathrm{sinc}(n_1\Delta_1\Omega_{U1}) - \Omega_{L1}\mathrm{sinc}(n_1\Delta_1\Omega_{L1})\right\}$$

It is also possible to arrive at the above by using the standard result for a rectangular bandpass filter, ie.

$$h(n_1\Delta_1,n_2\Delta_2) = \frac{\Delta_1\Delta_2}{\pi^2}[\Omega_{U2}\Omega_{U1}\,\mathrm{sinc}(\Omega_{U2}n_2\Delta_2)\,\mathrm{sinc}(\Omega_{U1}n_1\Delta_1)$$

$$-\Omega_{L2}\Omega_{L1}\,\mathrm{sinc}(\Omega_{L2}n_2\Delta_2)\,\mathrm{sinc}(\Omega_{L1}n_1\Delta_1)]$$

and letting $\Omega_{L2} = \Omega_{U2}$. [25%]

(TURN OVER

2 (a) (i) The actual filter frequency response $H(\omega_1, \omega_2)$ is given by the **convolution** of the desired frequency response $H_d(\omega_1, \omega_2)$ with the window function spectrum $W(\omega_1, \omega_2)$.

This is exactly as we should expect since we multiply in the spatial domain and must therefore convolve in the frequency domain.

Thus the effect of the window is to smooth H_d – clearly we would prefer to have the mainlobe width of $W(\omega_1, \omega_2)$ small so that H_d is changed as little as possible. We also want sidebands of small amplitude so that the ripples in the (ω_1, ω_2) plane outside the region of interest are kept small. [10%]

(ii) The Product Method for obtaining a 2-D window from 1-D windows is to simply take the product of two 1-D windows:

$$w(n_1,n_2) = w_1(n_1)\, w_2(n_2)$$

The Rotation Method of forming a 2-D window from 1-D windows is to obtain a 2-D *continuous* window function $w(u_1,u_2)$ by rotating a 1-D continuous window $w_1(u)$.

$$\boxed{w(u_1,u_2) = w_1(u)\big|_{u=\sqrt{(u_1^2+u_2^2)}}}$$

The continuous 2-D window is then sampled to produce a discrete 2-D window $w(n_1,n_2)$:

$$\boxed{w(n_1,n_2) = w(u_1,u_2)|_{u_1=n_1\Delta_1,\; u_2=n_2\Delta_2}}$$

[15%]

(iii) First find the FT of w_1

$$W_1(\omega_1) = \int_{-U_1}^{U_1} \left(0.42 + 0.5\cos(\pi\frac{u_1}{U_1}) + 0.08\cos(2\pi\frac{u_1}{U_1})\right) e^{-j\omega_1 u_1} du_1$$

$$= \int_{-U_1}^{U_1} 0.42 e^{-j\omega_1 u_1} du_1$$

$$+ \int_{-U_1}^{U_1} 0.25\{e^{ju_1(\pi/U_1-\omega_1)} + e^{-ju_1(\pi/U_1+\omega_1)}\} + 0.04\{e^{ju_1(2\pi/U_1-\omega_1)} + e^{-ju_1(2\pi/U_1+\omega_1)}\} du_1$$

(cont.

Evaluating each of these terms will produce sinc functions as follows:

$$W(\omega_1, U_1) = U_1 \{0.84\mathrm{sinc}(\omega_1 U_1) + 0.5 [\mathrm{sinc}(\pi - \omega_1 U_1) + \mathrm{sinc}(\pi + \omega_1 U_1)]$$

$$+0.08 [\mathrm{sinc}(2\pi - \omega_1 U_1) + \mathrm{sinc}(2\pi + \omega_1 U_1)]\}$$

$W(\omega_2, U_2)$ will take precisely the same form so that the required spectrum will be the product of W_1 and W_2. [35%]

(iv) If we sketch the above spectrum:

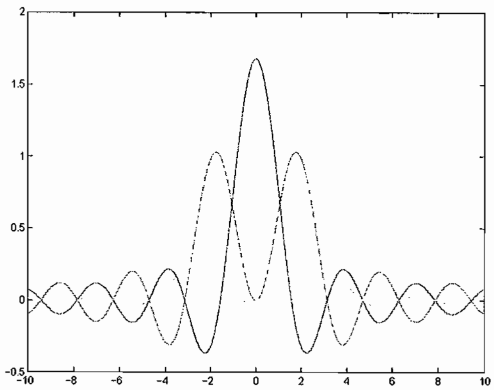

Fig. 3

we see that the effect of the extra sinc terms (Hamming window does not have the second cos term) is to slightly widen the main lobe and to give greater reduction in the sideband amplitudes compared to a Hamming window. This window is actually a *Blackman* window. Since our 2D window is the product of two of these 1D windows, it has the same properties. [10%]

(b) (i) The optimal linear spatially invariant filter called the Wiener Filter is arrived at by estimating the following cost function:

$$Q = E\{[x(\mathbf{n}) - \hat{x}(\mathbf{n})]^2\}$$

[10%]

(TURN OVER for continuation of Question 2

(ii) We assume an ideal world in which $\mathbf{x}$ is a gaussian random variable, described by a *known* covariance matrix $C = E[\mathbf{x}\mathbf{x}^T]$ so that

$$Pr(\mathbf{x}) \propto e^{-\frac{1}{2}\mathbf{x}^T C^{-1}\mathbf{x}}$$

[10%]

(iii) The Wiener solution is 'easy' to calculate and has known reconstruction errors. However, it is certainly by no means the best in real problems. It depends on the *assumption* of gaussianity and knowledge of the covariance structure *a priori*.

Since the world this not so simple, we are forced to consider alternative *priors*. One such prior which has been widely and successfully used is the *entropy prior*.

This produces the Maximum Entropy Method (MEM) which is applied to positive, additive distibutions (PADS). Let $\mathbf{x}$ be the (true) pixel vector we are trying to estimate, $Pr(\mathbf{x})$ is given by

$$Pr(\mathbf{x}) \propto e^{\alpha S}$$

where one version of the *entropy* S (sometimes known as the *cross entropy*) of the image is given by

$$S(\mathbf{x},\mathbf{m}) = \sum_i \left[x_i - m_i - x_i \ln\left(\frac{x_i}{m_i}\right)\right]$$

where $\mathbf{m}$ is the *measure* on an image space (*the model*) to which the image $\mathbf{x}$ defaults in the absence of data. (Can see global maximum of S occurs at $\mathbf{x} = \mathbf{m}$.) [10%]

3 (a) The key characteristics of vision that are exploited in image compression are:

(i) The human visual system (HVS) is much more sensitive to overall intensity (luminance) changes than to colour changes. Usually most of the information about a scene is contained in its luminance rather than its colour (chrominance).

(ii) The bandwidth of the HVS for luminance components is much wider than for chrominance (typically about 5 times as wide).

(iii) The contrast sensitivity of the HVS for luminance is also around 3 times better than red-green sensitivity and around 6 times better than blue-yellow.

(cont.

(iv) The luminance sensitivity also drops off at low spatial frequencies, but only if there is no temporal fluctuation (flicker).

(v) Activity masking occurs, such that in the presence of high image activity (eg a strong texture) it is much more difficult to notice coding distortions than in smooth areas of low activity.

These characteristics are exploited by performing a transformation from RGB to YUV colour space, and by using a lower sampling rate and coarser quantisation for the colour channels. Also the quantisation step size may be designed to adapt to local image activity levels to take advantage of activity masking. [20%]

(b) The 2-D Haar transform of $\mathbf{X}$ is given by

$$\mathbf{Y} = \mathbf{T}\,\mathbf{X}\,\mathbf{T}^T = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$= \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} a+b & a-b \\ c+d & c-d \end{bmatrix} = \frac{1}{2}\begin{bmatrix} a+b+c+d & a-b+c-d \\ a+b-c-d & a-b-c+d \end{bmatrix}$$

[20%]

(c) Consider the top left coefficients from each 2×2 image block. These represent half the sum of the four pixels in each block, which represents an averaging or lowpass filtering of each block. Hence the top left subband is a slightly blurred version of the original image, scaled up in amplitude by a factor 2, but reduced in size by 2:1 horizontally and vertically.

The top right coefficients can be written $\frac{a+c}{2} - \frac{b+d}{2}$, which measures the mean horizontal gradient of each 2×2 block. Hence this subband picks out vertical edges in the image, where pixels a and c are significantly different from pixels b and d.

The lower left coefficients can be written $\frac{a+b}{2} - \frac{c+d}{2}$, which measures the mean vertical gradient of each 2×2 block. Hence this subband picks out horizontal edges in the image, where pixels a and b are significantly different from pixels c and d.

The lower right coefficients can be written $\frac{a+d}{2} - \frac{b+c}{2}$, which measures the diagonal curvature of each 2×2 block. Hence this subband picks out corners and textures in the image, where pixels a and d are significantly different from pixels b and c. [20%]

(d) Since the top left subband is the result of lowpass filtering in both directions, it has similar characteristics to the original image (except that it is smaller).

(TURN OVER for continuation of Question 3

Hence we may apply the Haar transform again to this subband, splitting it into four more bands. The lowpass result of this second stage may be further decomposed by a third Haar transform, and so on for as many levels as required. The energy compaction properties of the transform are improved by each of the first few levels (typically four), but beyond that there is virtually no additional gain.

When subbands are quantised severely, the coefficients from the finer levels are often set to zero, so the reconstructed image is largely made up from basis functions of the lowpass subbands. At level 1 the Haar lowpass basis functions are just 2×2 blocks of equal pixels, and at coarser levels the block size increases to 4×4, 8×8 etc. Hence the image is made up from different sized block patterns, giving it a somewhat 'blocky' appearance, with larger blocks in areas of greater smoothness in the original image. [20%]

(e) The sum and difference operations of the Haar transform may be regarded as simple types of FIR filters, applied to the rows and columns of the image. To avoid the 'blocky' artifacts, we must modify these filters so that they have smoother responses. Wavelet concepts allow us to do this. Using the concept of a two-band filter bank with perfect reconstruction inverse filters, it is possible to design filters which have much better smoothness than the Haar filters by forcing them to have multiple zeros at $z = -1$ in the z-plane. This makes the filters more complex but gives quantisation artifacts which are less visible because the lowpass basis functions have smooth boundaries, rather than the sharp edges of the Haar functions. [20%]

4 (a) Assuming initially that the constants C_k are unity, for $N = 4$ we get:

$$\mathbf{T} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0.9239 & 0.3827 & -0.3827 & -0.9239 \\ 0.7071 & -0.7071 & -0.7071 & 0.7071 \\ 0.3827 & -0.9239 & 0.9239 & -0.3827 \end{bmatrix} = \begin{bmatrix} a & a & a & a \\ b & c & -c & -b \\ d & -d & -d & d \\ c & -b & b & -c \end{bmatrix}$$

where $a = 1,\ b = 0.9239,\ c = 0.3827,\ d = 0.7071$.

For orthogonality the dot product between any pair of rows of $\mathbf{T}$ must be zero. This is clearly the case between row 1 and rows 2, 3 or 4, since the latter 3 rows all sum to zero. Rows 2 and 4 have odd symmetry while row 3 has even symmetry, so the dot products of row 3 with 2 or 4 must be zero. This just leaves the dot product of rows 2 and 4 which is $(bc - bc - bc + bc) = 0$. Hence all dot products between different rows are zero, and therefore the rows are orthogonal.

(cont.

To make **T** orthonormal, the rows must all have unit squared magnitude as well as being orthogonal.

Hence, when the rows are multiplied by the C_k:

$$C_1^2(4a^2)=1 \quad C_2^2(2b^2+2c^2)=1 \quad C_3^2(4d^2)=1 \quad C_4^2(2c^2+2b^2)=1$$

Now $a^2=1$, $b^2+c^2=1$ and $d^2=0.5$. Therefore

$$C_1=0.5 \quad \text{and} \quad C_2=C_3=C_4=0.7071$$

[30%]

(b) To transform the columns of **X**, we pre-multiply it by **T** to get **TX**.

To transform the rows of the result, we post-multiply it by $\mathbf{T}^T$ to get $\mathbf{TXT}^T$. Hence

$$\mathbf{Y}=\mathbf{TXT}^T$$

To invert this expression we pre- and post-multiply both sides by the inverse of **T** and $\mathbf{T}^T$ to get

$$\mathbf{T}^{-1}\mathbf{Y}\mathbf{T}^{-T}=\mathbf{T}^{-1}\mathbf{TXT}^T\mathbf{T}^{-T}=\mathbf{X}$$

Since **T** is orthonormal, its inverse is its transpose, and so

$$\mathbf{X}=\mathbf{T}^{-1}\mathbf{Y}\mathbf{T}^{-T}=\mathbf{T}^T\mathbf{YT}$$

It is important for **T** to be orthonormal both to make it easy to perform the inverse transform as above, and to preserve the total energy of the signal between the image domain **X** and the transform domain **Y**. [20%]

(c) The entropy expression is a function of $p+q$ only. So, as p and q each go from 1 to 4, the value of $p+q$ takes on 7 distinct values from 2 to 8, each corresponding to all the subbands on a given diagonal of the 4×4 array of subbands. The number of bits for a given subband is then

$$B_{p,q}=H_{p,q}\times400\times300=120{,}000\ H_{p,q}$$

assuming ideal entropy coding.

(TURN OVER for continuation of Question 4

Let the number of subbands on a given diagonal of the array of subbands be N_{p+q}. Then the total number of bits needed is

$$B_{tot} = \sum_{p=1}^{4} \sum_{q=1}^{4} B_{p,q} = \sum_{p+q=2}^{8} N_{p+q} \, B_{p,q}$$

We can assemble the following table to calculate the total number of bits:

$p+q$	$H_{p,q}$	$B_{p,q}$	N_{p+q}	Bits
2	4.0	480,000	1	480,000
3	2.8301	339,609	2	679,218
4	2.0	240,000	3	720,000
5	1.3561	162,737	4	650,948
6	0.8301	99,609	3	298,827
7	0.3853	46,235	2	92,470
8	0	0	1	0
		Total:		2,921,463

Hence approximately 2.9 Mbits are needed to code this image (1.52 bit/pixel). [30%]

(d) A colour image would typically be coded in the YUV domain so that the U and V chrominance images could be subsampled 2:1 in each direction. Assuming similar entropies for the colour subbands but only one quarter of the number of pixels in each subband, each chrominance component would require $2.9/4 = 0.725$ Mbits.

The luminance component would still need 2.9 Mbits.

So the total number of bits needed for the colour image would be

$$2.9 + 0.725 + 0.725 = 4.35 \text{ Mbits}$$

In practise, it would probably require less than this because the chrominance components can be quantised a bit more coarsely than the luminance component, especially in the higher frequency subbands. [20%]

END OF PAPER