Version JL/2

EGT3
ENGINEERING TRIPOS PART IIB: SOLUTIONS

Module 4F8

IMAGE PROCESSING AND IMAGE CODING

1 (a) (i) Taking the inverse FT of the ideal frequency response will give an impulse response which does not have finite support – to remedy this we multiply by a *window function* which forces the impulse response coefficients to zero for (n_1, n_2) outside R_h, the desired support region. The actual filter frequency response $H(\omega_1, \omega_2)$ is then 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.

The two most popular methods of forming 2d windows from 1d windows are

A. Taking the product of 1d windows:

$$w(u_1, u_2) = w_1(u_1)\, w_2(u_2)$$

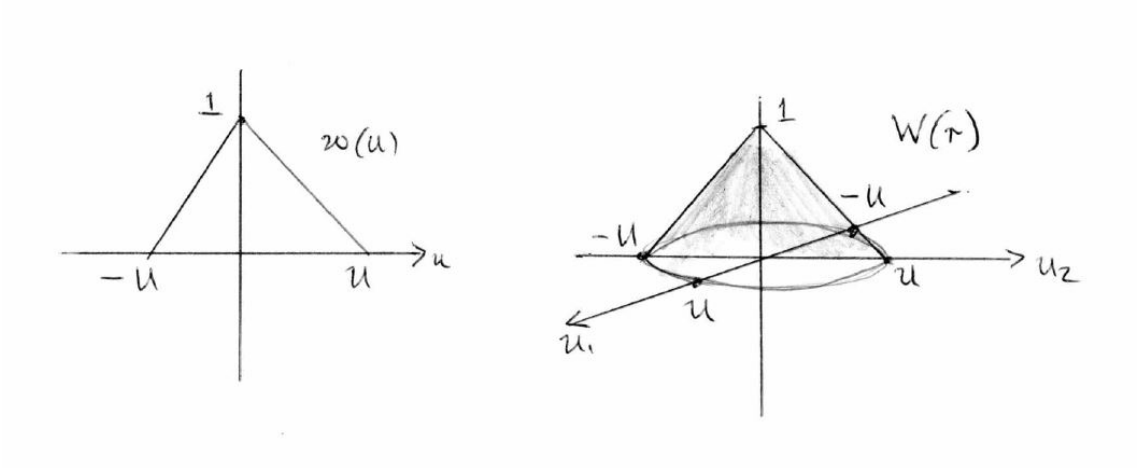
B. Rotating a 1d window:

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

This first part was mainly bookwork and almost all candidates gave reasonable answers.

(ii) The 1D window function w given is a triangular window of height (at $(0,0)$) 1 and width $2U$.

Thus the 2D window function, W, formed by the rotation method from w is a 'cone' of height 1 and maximum radius U – sketch below:



While many candidates produced the required sketch, some produced poor 2d versions.

(iii) Now we want to directly integrate to form the Fourier transform of W.

$$\mathscr{W}(\omega_1,\omega_2)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}W(u_1,u_2)\mathrm{e}^{-j(\omega_1u_1+\omega_2u_2)}du_1\,du_2$$

given the form of W, we make a change of variables from (u_1,u_2) to (r,θ) to give (noting that $du_1du_2=rdrd\theta$)

$$\mathscr{W}(\omega_1,\omega_2)=\int_{r=0}^{U}\int_{\theta=-\pi}^{\pi}r\left(1-\frac{r}{U}\right)\mathrm{e}^{-jr\sqrt{\omega_1^2+\omega_2^2}\sin(\theta+\phi)}drd\theta$$

with $\tan\phi=\omega_1/\omega_2$. Next, expand the complex exponential in terms of sin and cos and argue that the periodicity of sin makes the imaginary part of the integral zero. Using the fact that we can change the limits on the θ integral and the definition of the Bessel function, as given in the lectures, the integral reduces to

$$\mathscr{W}(\omega_1,\omega_2)=2\pi\int_{r=0}^{U}r\left(1-\frac{r}{U}\right)J_0(r\sqrt{\omega_1^2+\omega_2^2})dr$$

So that $f(\omega_1,\omega_2)=\sqrt{\omega_1^2+\omega_2^2}$.

This was the part which caused most problems – while it is essentially bookwork, you do need to remember how to handle the integration to produce a Bessel function. A good number of candidates just left this part of the question out.

(iv) In Fig.1 we see that as $\omega=\sqrt{\omega_1^2+\omega_2^2}$ increases the two terms in the integral tend to the same value, so when subtracted they give zero. This tells us that the spectrum of this rotated window will have very low sidelobes and will thus behave well under convolution. The window will therefore have good behaviour as regards artefacts. The mainlobe is reasonably wide, which is a less good feature.

The answers to this last part of (a) were largely correct, though some candidates also came to the conclusion that the width of the mainlobe was good.

(b) (i) Consider the ideal filter given in fig.2 – one way to construct this is to say that the ideal frequency response of this filter, $H(\omega_1, \omega_2)$, can be written as

$$H(\omega_1, \omega_2) = H_1(\omega_1, \omega_2) + H_2(\omega_1, \omega_2)$$

where H_1 is a rectangular lowpass filter given by

$$H_1(\omega_1, \omega_2) = \begin{cases} 1 & \text{if } |\omega_1| < \Omega \text{ and } |\omega_2| < \Omega \\ 0 & \text{otherwise} \end{cases}$$

and H_2 is an ideal bandpass filter given by $H_2 = H_u - H_l$:

$$H_u(\omega_1, \omega_2) = \begin{cases} 1 & \text{if } |\omega_1| < 5\Omega \text{ and } |\omega_2| < 5\Omega \\ 0 & \text{otherwise} \end{cases}$$

$$H_l(\omega_1, \omega_2) = \begin{cases} 1 & \text{if } |\omega_1| < 3\Omega \text{ and } |\omega_2| < 3\Omega \\ 0 & \text{otherwise} \end{cases}$$

We can therefore use standard results (or derive them) to write our impulse response of H as the sum of the impulse responses of H_1 and H_2.

$$h(n_1\Delta_1, n_2\Delta_2) = \frac{\Delta_1\Delta_2}{\pi^2}[\Omega^2 \operatorname{sinc}(\Omega n_2\Delta_2)\operatorname{sinc}(\Omega n_1\Delta_1)]$$

$$+\frac{\Delta_1\Delta_2}{\pi^2}[5\Omega\operatorname{sinc}(5\Omega n_1\Delta_1)5\Omega\operatorname{sinc}(5\Omega n_2\Delta_2)) - 3\Omega\operatorname{sinc}(3\Omega n_1\Delta_1)3\Omega\operatorname{sinc}(3\Omega n_2\Delta_2)]$$

if we expand this we have that the values required are $\gamma_1 = 1$, $\gamma_2 = -9$, $\gamma_3 = 25$, $\alpha_1 = \beta_1 = 1$, $\alpha_2 = \beta_2 = 3$, $\alpha_3 = \beta_3 = 5$.

Most candidates used standard results, though some integrated, and most did get the correct answers. The most common mistake was to use 2,6,and 10 in the sincs, rather than the half widths, 1,3,5.

(ii) Clearly the given filter is a combination of a lowpass and a bandpass filter – the resulting image will have most high frequencies removed, but will display areas of medium frequency corresponding to the alternative bandpass region.

An easy end to the question which most people answered correctly.

2 (a) (i) The likelihood is obtained by using the fact that the noise is Gaussian:

$$P(\mathbf{y}|\mathbf{x}) \propto \mathrm{e}^{-\frac{1}{2}\mathbf{d}^T N^{-1}\mathbf{d}} = \mathrm{e}^{-\frac{1}{2}(\mathbf{y}-L\mathbf{x})^T N^{-1}(\mathbf{y}-L\mathbf{x})}$$

[10%]

This part was mostly well done, though a good number left the likelihood just in terms of the noise.

(ii) Again, using the fact that we can regard $\mathbf{x}$ as a Gaussian random variable:

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

From Bayes' theorem,

$$P(\mathbf{x}|\mathbf{y}) \propto P(\mathbf{y}|\mathbf{x})P(\mathbf{x})$$

which is then given by

$$P(\mathbf{x}|\mathbf{y}) \propto \mathrm{e}^{-\frac{1}{2}\left[(\mathbf{y}-L\mathbf{x})^T N^{-1}(\mathbf{y}-L\mathbf{x})+\mathbf{x}^T C^{-1}\mathbf{x}\right]}$$

We obtain the Wiener filter result by looking for a W such that $\hat{\mathbf{x}} = W\mathbf{y}$ is the value of $\mathbf{x}$ that minimizes the exponent above. [20%]

Most candidates were able to obtain the correct expression of the posterior, but many did not seem to understand what was being asked for in the last part (i.e. qualitatively explain how the Wiener filter is obtained).

(b) (i) We write down the posterior as proportional to the likelihood and the prior:

$$P(\mathbf{x}|\mathbf{y}) \propto \exp\left(-\frac{1}{2}[\frac{1}{\sigma^2}\sum_i (y_i - x_i)^2 + \lambda \sum_i x_i^2]\right)$$

Differentiate the RHS wrt x_i to give

$$\left[\frac{1}{-\sigma^2}(y_i - x_i) + \lambda x_i\right] \exp\left(-\frac{1}{2}[\frac{1}{\sigma^2}\sum_i (y_i - x_i)^2 + \lambda \sum_i x_i^2]\right)$$

Setting this to zero gives

$$\hat{x}_i = \frac{y_i}{\lambda\sigma^2 + 1}$$

which is our best estimate of the underlying image. [25%]

(ii) In this case our posterior (note that the x_i can only take positive or zero values) is given by

$$P(\mathbf{x}|\mathbf{y}) \propto \exp\left(-\frac{1}{2}\left[\frac{1}{\sigma^2}\sum_i (y_i - x_i)^2 + 2\mu\sum_i x_i\right]\right)$$

Again, differentiate wrt x_i and set to zero, to give

$$-\frac{1}{\sigma_i}(y_i - x_i) + \mu = 0$$

So that $x_i = y_i - \mu\sigma^2$ or zero if RHS is negative (x_i cannot be negative). [35%]

(iii) It is clear that the Wiener result simply scales all image pixels by a constant amount. However, the CS result succeeds in setting a number of pixels to zero, and hence sparsity has certainly been encouraged. [10%]

This was not a popular question, as it looked different from standard questions in previous years. However, it was easy if you knew what you were doing, and many people got full marks on part b).

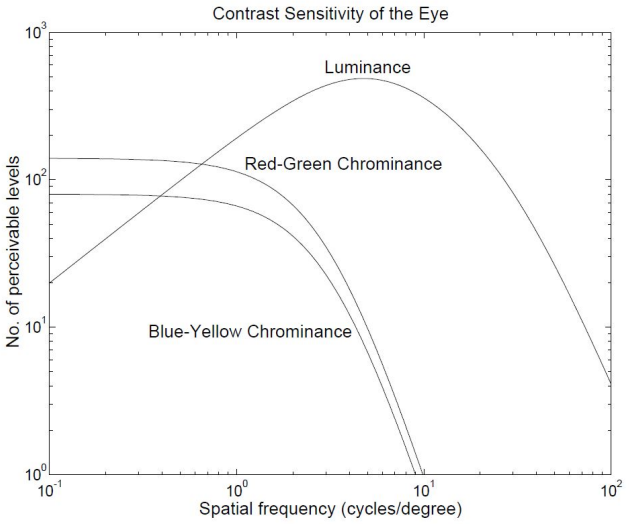
3 (a) (i) YUV has been the colour encoding system used for analogue television worldwide (PAL, NTSC, SECAM standards). There is a very simple conversion from RGB to YUV (values are approximate):

$$\begin{aligned} Y &= 0.3R + 0.6G + 0.1B \\ U &= 0.5(B - Y) \\ V &= 0.625(R - Y) \end{aligned}$$

Y is a luminance component and U,V are chrominance components. This can then be written in matrix form.

A surprising number did not know this transform, but most did get the general ideas, particularly how Y is expressed in terms of R, G, B.

(ii) This sketch shows the sensitivity of the eye to luminance and chrominance intensity changes (V = R-G, U = B-Y).

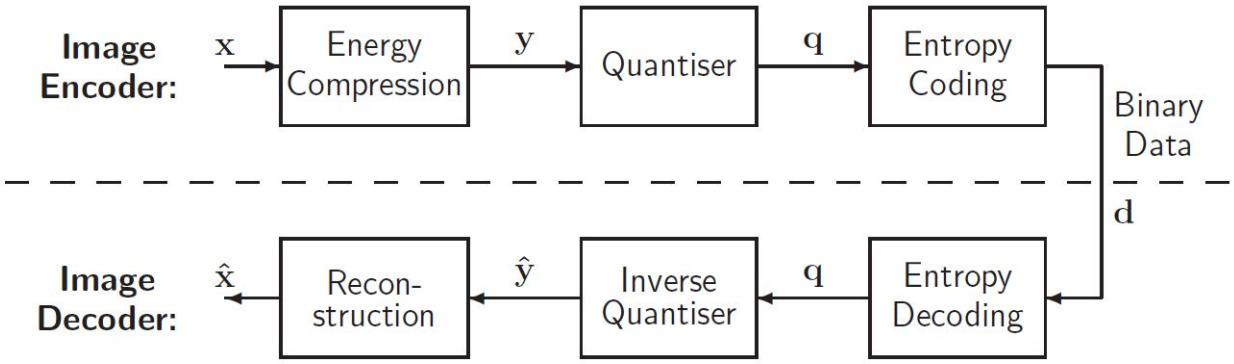


Note: the eye has very little response above 100 cycles/degree; luminance sensitivity drops off at low spatial frequencies; maximum chrominance sensitivity is much lower than maximum luminance sensitivity; chrominance sensitivitites fall off above 1 cycle/degree. This means that U and V components can be sampled at a lower rate than Y (due to the narrower bandwidth), and may be quantised more coarsely (due to the lower contrast sensitivity).

This question was not well done. There were lots of variants of this graph. The implications of this graph were also poorly explained on the whole.

(b) The figure below shows the main blocks in any *image coding* system. **x** is a

monochrome Y) image. Elements of $\mathbf{x}$ are $x(\mathbf{n}) \equiv x(n_1, n_2)$, where n_1 runs over rows and n_2 runs over columns. The *Binary Data* is the transmitted data.



Almost everyone got full or close to full marks on this part.

(c) (i) The 2×2 Haar transformation matrix T acts on a length 2 vector $\mathbf{x}$ to produce a length 2 vector $\mathbf{y}$:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = T \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad \text{where} \quad T = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Note that $T = T^T$, so that

$$TT^T = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Thus T is **orthonormal**. Since

$$(\text{Energy of } \mathbf{y}) = \mathbf{y}^T \mathbf{y} = \mathbf{x}^T T^T T \mathbf{x} = \mathbf{x}^T \mathbf{I} \mathbf{x} = \mathbf{x}^T \mathbf{x} = (\text{Energy of } \mathbf{x})$$

we see that T is an **energy preserving** transform.

Mostly well done. Though more than a handful wrote down an incorrect Haar matrix.

(ii) To apply T to a 2×2 matrix, M, we first act on columns then on rows, via TMT^T. Now take the TL, TR, BL, BR 2×2 blocks of X and transform:

 (cont.

$$Y_{TL} = T \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} T^T = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}$$

$$Y_{TR} = T \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} T^T = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$Y_{BL} = T \begin{bmatrix} 3 & 0 \\ 1 & 2 \end{bmatrix} T^T = \begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix}$$

$$Y_{BR} = T \begin{bmatrix} 1 & 2 \\ 3 & 0 \end{bmatrix} T^T = \begin{bmatrix} 3 & 1 \\ 0 & -2 \end{bmatrix}$$

Surprisingly, there were lots of mistakes and careless errors in this part. The question on the whole was the most popular question, but also the question with the lowest average.

(iii) From these Y_is we then take all top LH entries and form a 2×2 matrix which will form the top LH corner of the 4×4 image Y, and similarly for other entries. This gives:

$$Y = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 3 & 3 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 \end{bmatrix}$$

The top LH corner is the Lo-Lo image (lowpass in both directions). The top RH corner is Hi-Lo image (horizontal highpass, vertical lowpass). The bottom RH corner is Lo-Hi image (horizontal lowpass, vertical highpass). The bottom RH corner is Hi-Hi image (horizontal highpass, vertical Highpass).

Most candidates understood the process of forming the subband images, but suprisingly many simply explained how to do it without actually giving the new 4×4 *block* Y.

(iv) After the first (level-1) Haar transform, we take the resulting Lo-Lo image (which will have statistics similar to the original image) and apply the Haar transform again to this, producing a further 4 (level-2) subimages. The level 2 Lo-Lo subimage is then transformed yet again to produce 4 level-3 subimages etc. We can do this as the Lo-Lo subimages contain most of the image energy – the performance when we quantise these subimages and reconstruct is much better

than simply quantising the original image. Typically there is no improvement in compression performance after 4 levels of the Haar transform.

This last part was well done by almost all who attempted it.

4 (a) (i) If $\mathbf{t}_i$ is the ith row of the $n \times n$ DCT matrix T,

$$\mathbf{t}_1 \cdot \mathbf{t}_1 = \sum_i t_{1i} t_{1i} = \frac{1}{n} n = 1$$

and

$$\mathbf{t}_1 \cdot \mathbf{t}_1 = \frac{2}{n} \sum_{i=1}^{n} \cos^2 \left(\frac{\pi(i-\frac{1}{2})(k-1)}{n} \right) = \frac{1}{n} \sum_{i=1}^{n} \left[1 + \cos \left(\frac{2\pi(i-\frac{1}{2})(k-1)}{n} \right) \right]$$

$$= 1 + \frac{1}{n} \sum_{i=1}^{n} \cos \left(\frac{2\pi(i-\frac{1}{2})(k-1)}{n} \right)$$

for n even, $\cos \left(\frac{2\pi(i-\frac{1}{2})(k-1)}{n} \right) = -\cos \left(\frac{2\pi([n/2+1-i]-\frac{1}{2})(k-1)}{n} \right)$ so all the terms in the sum cancel, verifying that the rows square to 1.
For $\mathbf{t}_j \cdot \mathbf{t}_k$, $j \neq k$, we have

$$\mathbf{t}_j \cdot \mathbf{t}_k = \frac{2}{n} \sum_{i=1}^{n} \cos \left(\frac{\pi(i-\frac{1}{2})(j-1)}{n} \right) \cos \left(\frac{\pi(i-\frac{1}{2})(k-1)}{n} \right)$$

When we take the transform of an n-point vector using $\mathbf{y} = T\mathbf{x}$, $\mathbf{x}$ is decomposed into a linear combination of the basis functions (rows) of T, whose coefficients are the samples of $\mathbf{y}$, because $\mathbf{x} = T^T \mathbf{y}$ and therefore

$$\mathbf{x} = T^T \mathbf{y} = y_1 \mathbf{t}_1 + y_2 \mathbf{t}_2 + ... + y_n \mathbf{t}_n$$

since $\mathbf{t}_k$ is the kth row of T and therefore the kth column of T^T.

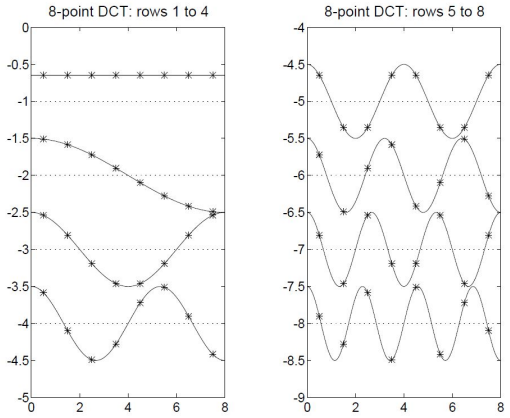
$$= \frac{1}{n} \sum_{i=1}^{n} \cos \left(\frac{\pi(i-\frac{1}{2})(j+k-2)}{n} \right) + \cos \left(\frac{\pi(i-\frac{1}{2})(j-k)}{n} \right) = 0$$

since$\cos A \cos B = (1/2)(\cos(A+B) + \cos(A-B)$ and the sum of each term is zero for the same reasons as stated previously.
The matrix is therefore orthogonal for n even

(ii) When we take the transform of an n-point vector using $\mathbf{y} = T\mathbf{x}$, $\mathbf{x}$ is decomposed into a linear combination of the basis functions (rows) of T, whose coefficients are the samples of $\mathbf{y}$, because $\mathbf{x} = T^T \mathbf{y}$ and therefore

$$\mathbf{x} = T^T \mathbf{y} = y_1 \mathbf{t}_1 + y_2 \mathbf{t}_2 + ... + y_n \mathbf{t}_n$$

since $\mathbf{t}_k$ is the kth row of T and therefore the kth column of T^T. The $\mathbf{t}_i$ can therefore be viewed as basis functions as any $\mathbf{x}$ can be written in terms of them.
A sketch of these basis functions is given below:



(iii) An 8×8 DCT in 2-D transforms a subimage, X, of 8×8 pixels into a matrix, Y, of 8×8 DCT coefficients.Y is given by:

$$Y = TXT^T$$

.
Since this means $X = T^T YT$, we see that we can write X as:

$$X = \sum_{i,j} Y_{ij} M_{ij} = Y_{11}M_{11} + Y_{12}M_{12} + ...$$

where $M_{pq} = \mathbf{t}_p^T \mathbf{t}_q$, which are therefore our 2D basis functions.

(b) (i) $Y \implies 1024 \times 2048 = 2^{21}$ pixels.
$U, V \implies (1024/2) \times (2048/2) = 2^{19}$ pixels.
ie we assume that we downsample U, V by a factor of 2 in each dimension (if candidates use another reasonable factor, this is also OK).
Therefore the number of bits required is approximated by the $entropy \times No.pixels$.
For Y: 1.2×2^{21}, for U, V: $2 \times 0.5 \times \frac{1}{4} \times 2^{21}$.
Therefore total number of bits required is : $1.45 \times 2^{21} = 3.04 \times 10^6$.
The proportion of the total bits needed to encode the U, V channels is

$$\frac{0.25}{1.45} \approx 17.2\%$$

.
...a small fraction of the total.

 (cont.

(ii) After DCT compression (usually 1 level) the DCT coefficients are quantised with an optimised quantisation step – these are predetermined, and are tailored for each subband via experiment with many natural images (see luminance and chrominance quantisation matrices in notes). Then the entropy coding scheme combines run-length and amplitude information into a single Huffman code.

END OF PAPER

THIS PAGE IS BLANK