Copy 1

Solutions: 4F8 2013

ENGINEERING TRIPOS PART IIB

Wednesday 24 April 2013 9.30 to 11

Module 4F8

IMAGE PROCESSING AND IMAGE CODING

Version: JL02

1 (a) (i) We can write s as a Fourier series:

$$s(u_1,u_2) = \sum_{p_1=-\infty}^{\infty} \sum_{p_2=-\infty}^{\infty} c(p_1,p_2)\, e^{j(p_1\Omega_1 u_1 + p_2\Omega_2 u_2)}$$

where $\Omega_1 = \frac{2\pi}{\Delta_1}$ and $\Omega_2 = \frac{2\pi}{\Delta_2}$.
We can then find the Fourier coefficients c in the usual way:

$$c(p_1,p_2) = \frac{1}{\Delta_1 \Delta_2} \int_{-\frac{\Delta_2}{2}}^{\frac{\Delta_2}{2}} \int_{-\frac{\Delta_1}{2}}^{\frac{\Delta_1}{2}} s(u_1,u_2)\, e^{-j(p_1\Omega_1 u_1 + p_2\Omega_2 u_2)}\, du_1\, du_2$$

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

$$\times e^{-j(p_1\Omega_1 u_1 + p_2\Omega_2 u_2)}\, du_1\, du_2$$

$$\Longrightarrow c(p_1,p_2) = \frac{1}{\Delta_1 \Delta_2} \quad \text{for all} \quad p_1, p_2$$

The sampled image may then be expressed as:

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

Using the frequency shift or spatial modulation theorem to take the Fourier transform

$$g(u_1,u_2) e^{j(p_1\Omega_1 u_1 + p_2\Omega_2 u_2)} \Leftrightarrow G(\omega_1 - \Omega_1 p_1, \omega_2 - \Omega_2 p_2)$$

gives:

$$\boxed{G_s(\omega_1,\omega_2) = \frac{1}{\Delta_1 \Delta_2} \sum_{p_1=-\infty}^{\infty} \sum_{p_2=-\infty}^{\infty} G(\omega_1 - p_1\Omega_1, \omega_2 - p_2\Omega_2)}$$

It can therefore be seen that the Fourier transform or spectrum of the sampled 2d signal is the periodic repetition of the spectrum of the unsampled 2d signal – precisely analogous to the 1d case. It is therefore clear that for a bandlimited 2d signal, we must sample at more than twice the largest frequencies in the signal to keep these copies of the FT separate. Hence

$$\frac{2\pi}{\Delta_1} > 2\Omega_{B1} \quad \frac{2\pi}{\Delta_2} > 2\Omega_{B2}$$

These are the Nyquist frequencies, and if we sample below these we observe artefacts which we call aliasing.

[30%]

(ii) If an image does not contain high frequencies simultaneously in both dimensions, then it is more efficient to sample on a diamond grid (for obvious reasons). For functions which are circularly symmetric or bandlimited over a circular region, it can be shown that it is more efficient to sample on a hexagonal grid. When we talk about 'efficiency' here we generally mean 'needing fewer samples'. However, computational load is rarely a problem with current computational power, so the simplicity of rectangular sampling means that we would rarely use anything else.

[10%]

(iii) we know that to avoid aliasing we need to sample at twice the largest frequencies in the signal. Therefore

$$\Omega_1 = \frac{2\pi}{\Delta_1} \geq 2\Omega$$

and

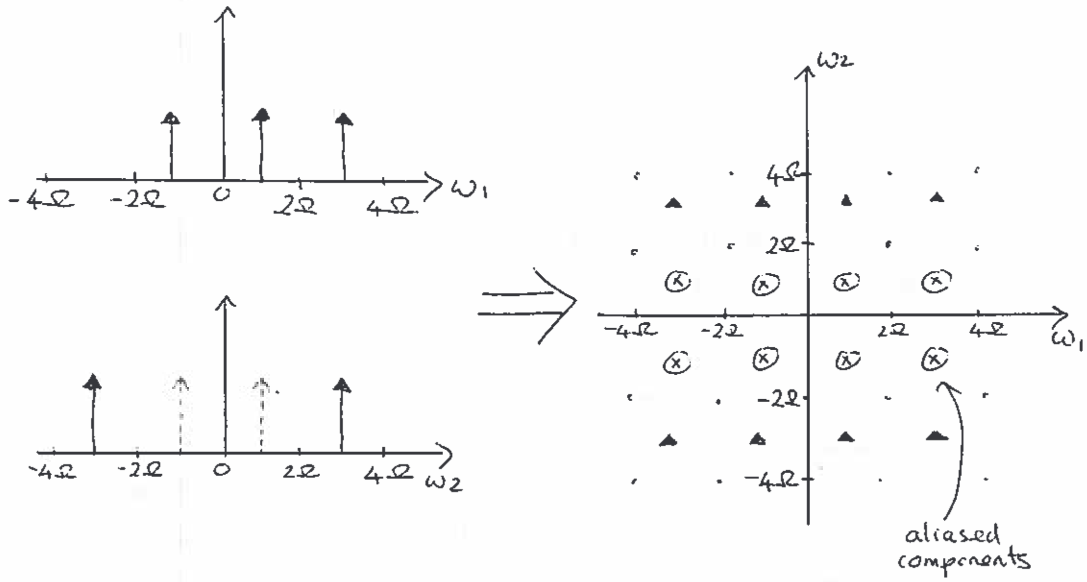
$$\Omega_2 = \frac{2\pi}{\Delta_2} \geq 6\Omega$$

Therefore 2Ω and 6Ω are the minimum sampling frequencies (in u_1 and u_2 directions) required to avoid aliasing.
So if $\Delta_1 = \Delta_2 = \pi/\Omega$, we see that

$$\Omega_1 = \Omega_2 = \frac{2\pi}{\Delta_1} = 2\Omega$$

and we will avoid aliasing in the ω_1 direction but not in the ω_2 direction. The spectrum is therefore a set of aliased delta functions as sketched below:

[20%]



(b) Consider the bandpass filter given in the figure – 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 bandpass filter given by $H_{1a} - H_{1b}$

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

$$H_{1b}(\omega_1, \omega_2) = \begin{cases} 1 & \text{if } |\omega_1| < \Omega_{L1} \text{ and } |\omega_2| < \Omega_{L2} \\ 0 & \text{otherwise} \end{cases}$$

and H_2 is the separable ideal bandpass filter given by

$$H_2(\omega_1, \omega_2) = \begin{cases} 1 & \text{if } \Omega_{L1} < |\omega_1| < \Omega_{U1} \text{ and } \Omega_{L2} < |\omega_2| < \Omega_{U2} \\ 0 & \text{otherwise} \end{cases}$$

We know that since H_2 is separable, we can write it as the product of two 1d filters, i.e.

$$H_2(\omega_1, \omega_2) = H_a(\omega_1) H_b(\omega_2)$$

where $H_a(\omega_1)$ is an ideal 1-D bandpass filter with a lower cut-off frequency of Ω_{L1} and an upper cut-off frequency of Ω_{U1}. Similarly $H_b(\omega_2)$ is an ideal 1-D bandpass filter with cut-off frequencies Ω_{L2} and Ω_{U2}. More explicitly we have

$$H_a(\omega_1) = \begin{cases} 1 & \text{if } \Omega_{L1} < |\omega_1| < \Omega_{U1} \\ 0 & \text{otherwise} \end{cases}$$

$$H_b(\omega_2) = \begin{cases} 1 & \text{if } \Omega_{L2} < |\omega_2| < \Omega_{U2} \\ 0 & \text{otherwise} \end{cases}$$

Thus, we can now work out the ideal impulse response of the filter from the impulse responses of H_1 and H_2. We have (where $h(n_1,n_2) \equiv h(n_1\Delta_1, n_2\Delta_2)$)

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

so that the impulse response for H_{1a} is

$$h_{1a}(n_1,n_2) = \frac{\Delta_1\Delta_2}{(2\pi)^2}\int_{-\Omega_{U2}}^{\Omega_{U2}}\int_{-\Omega_{U1}}^{\Omega_{U1}} e^{j(\omega_1 n_1\Delta_1+\omega_2 n_2\Delta_2)}d\omega_1 d\omega_2$$

$$= \frac{\Delta_1\Delta_2\Omega_{U1}\Omega_{U2}}{(\pi)^2}\operatorname{sinc}(\Omega_{U2}n_2\Delta_2)\operatorname{sinc}(\Omega_{U1}n_1\Delta_1)$$

Similarly, for H_{1b} we have

$$h_{1b}(n_1,n_2) = \frac{\Delta_1\Delta_2\Omega_{L1}\Omega_{L2}}{(\pi)^2}\operatorname{sinc}(\Omega_{L2}n_2\Delta_2)\operatorname{sinc}(\Omega_{L1}n_1\Delta_1)$$

The impulse response for H_2 is similarly given by

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

$$= \frac{\Delta_1\Delta_2}{(2\pi)}\left[\int_{-\Omega_{U1}}^{-\Omega_{L1}} e^{j\omega_1 n_1\Delta_1}\,d\omega_1 + \int_{\Omega_{L1}}^{\Omega_{U1}} e^{j\omega_1 n_1\Delta_1}\,d\omega_1\right]\frac{1}{(2\pi)}\left[\int_{-\Omega_{U2}}^{-\Omega_{L2}} e^{j\omega_2 n_2\Delta_2}\,d\omega_2 + \int_{\Omega_{L2}}^{\Omega_{U2}} e^{j\omega_2 n_2\Delta_2}\,d\omega_2\right]$$

Thus we have

$$h_2(n_1,n_2) = \frac{\Delta_1\Delta_2}{(\pi)^2}\left[\Omega_{U1}\operatorname{sinc}(\Omega_{U1}n_1\Delta_1) - \Omega_{L1}\operatorname{sinc}(\Omega_{L1}n_1\Delta_1)\right]\left[\Omega_{U2}\operatorname{sinc}(\Omega_{U2}n_2\Delta_2) - \Omega_{L2}\operatorname{sinc}(\Omega_{L2}n_2\Delta_2)\right]$$

Now, forming $h(n_1,n_2)$ from the difference of $h_1(n_1,n_2)$ and $h_2(n_1,n_2)$ we have

$$h(n_1,n_2) = \frac{\Delta_1\Delta_2}{\pi^2}\left[\Omega_{U1}\,\Omega_{L2}\operatorname{sinc}(\Omega_{U1}n_1\Delta_1)\operatorname{sinc}(\Omega_{L2}n_2\Delta_2) + \Omega_{L1}\,\Omega_{U2}\operatorname{sinc}(\Omega_{L1}n_1\Delta_1)\operatorname{sinc}(\Omega_{U2}n_2\Delta_2)\right.$$

$$-2\Omega_{L1}\,\Omega_{L2}\,\mathrm{sinc}(\Omega_{L1}n_1\Delta_1)\,\mathrm{sinc}(\Omega_{L2}n_2\Delta_2)]$$

Thus we have that

$$K_1 = \frac{\Delta_1\Delta_2}{\pi^2}\Omega_{U1}\Omega_{L2}, \quad K_2 = -\frac{\Delta_1\Delta_2}{\pi^2}\Omega_{L1}\Omega_{U2}, \quad K_3 = 2\frac{\Delta_1\Delta_2}{\pi^2}\Omega_{L1}\Omega_{L2}$$

The figure shows that the maximum frequency ranges are within the values of π/Δ_1 (for ω_1) and π/Δ_2 (for ω_2) – thus, the Nyquist criteria are satisfied and there will be no aliasing.

[40%]

2 (a) (i) 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 $w_1(u)$.

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

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

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

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 $W(\omega_1,\omega_2)$.

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. [25%]

(ii) To find the spectrum we need to FT each of the 1-D window functions. Taking w_1;

$$W(\omega_1) = \int_{-U_1}^{+U_1} t(u_1)\mathrm{e}^{-j\omega_1 u_1}\, du_1 \;+\; 0.5\int_{-U_1}^{+U_1} \mathrm{e}^{-j\omega_1 u_1}\, du_1$$

We know that the second term is a pulse so has the standard FT of a sinc

$$0.5 \times 2U_1 \;\mathrm{sinc}(\omega_1 U_1)$$

The triangle pulse $t(u_1)$ is given by

$$t(u_1) = \begin{cases} u_1/U_1 + 1 & \text{for } -U_1 < u_i < 0 \\ -u_1/U_1 + 1 & \text{for } 0 < u_i < U_1 \\ 0 & \text{otherwise} \end{cases}$$

So, we know find the FT of $t(u_i)$ via

$$\int_{-U_1}^{0} (u_1/U_1 + 1)\mathrm{e}^{-j\omega_1 u_1}\, du_1 \;+\; \int_{0}^{U_1} (-u_1/U_1 + 1)\mathrm{e}^{-j\omega_1 u_1}\, du_1$$

$$= 2\int_0^{U_1} (-u_1/U_1 + 1)\cos(\omega_1 u_1)\, du_1$$

Evaluating this gives

$$-\frac{2}{U_1}\int_0^{U_1} u_1 \cos(\omega_1 u_1)\, du_1 + 2\int_0^{U_1} \cos(\omega_1 u_1)\, du_1$$

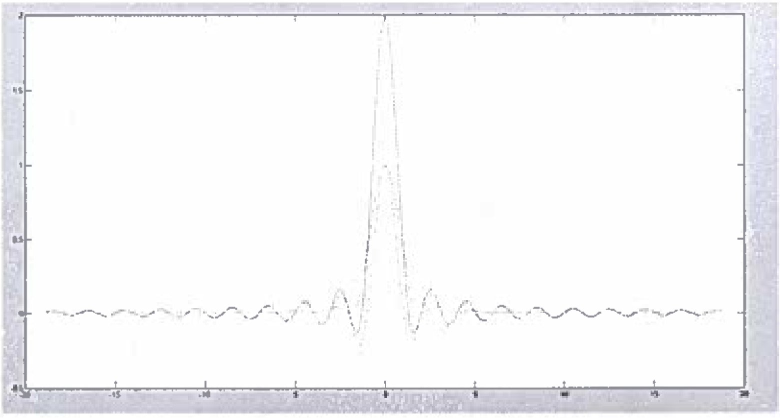
Integrating the first term by parts and straightforwardly integrating the second term gives:

$$\frac{2}{U_1 {\omega_1}^2}[1 - \cos(\omega_1 U_1)] = U_1 \;\mathrm{sinc}^2 \frac{\omega_1 U_1}{2}$$

Thus, the FT of $w(u_1)$ is

$$W(\omega_1) = U_1 \;\mathrm{sinc}(\omega_1 U_1) + U_1 \;\mathrm{sinc}^2 \frac{\omega_1 U_1}{2}$$

A plot of each term, plus the sum is given below (for $U_1 = 1$) (where dashed is the sinc^2 term, dash-dot is the sinc term and solid is the sum):



The 2-D spectrum is simply given by the product of the2 1-D spectra:

$$W(\omega_1,\omega_2) = U_1U_2\left[\text{ sinc}(\omega_1 U_1) + \text{ sinc}^2\frac{\omega_1 U_1}{2}\right]\left[\text{ sinc}(\omega_2 U_2) + \text{ sinc}^2\frac{\omega_2 U_2}{2}\right]$$

[35%]

(iii) From the equations above and figure (a)(ii) we can see that the triangular window produces a sinc^2 spectrum which has its first null at $2\pi/U_1$, while the constant term has a sinc spectrum with first null at π/U_1. From the figure, we can see that the addition of the constant therefore reduces the width of the mainlobe, but increases the amplitude of the sidelobes. [10%]

(b) (i) If we take the convolution as a discrete sum, the expression for y is:

$$y(\mathbf{n}) = \sum_{\mathbf{m}\in\mathbf{Z}^2} h(\mathbf{m})x(\mathbf{n}-\mathbf{m}) + d(\mathbf{n})$$

where $\mathbf{n} = (n_1, n_2)$. [10%]

(ii) 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%]

(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 non-linear methods which can be dealt with via alternative *priors*. One such prior which has been widely andsuccessfully 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 $\mathbf{x} = \mathbf{m}$.)

[A very much shorter answer will suffice in the exam!]

[10%]

3 a) The rows of T are samples from cosine functions with 0, ½, 1, 3/2 cycles of a cosine wave. Hence the first row is constant as shown and the 3rd row is $\propto \cos\pi/4, \cos 3\pi/4, \cos 5\pi/4, \cos 7\pi/4$.

To be normalised: $\sum_{j=1}^{4} t_{ij}^2 = 4a^2 = 1 \quad \therefore a = 1/2$

Row 2 must be $\propto \cos\pi/8, \cos 3\pi/8, \cos 5\pi/8, \cos 7\pi/8$

Hence $\quad b \propto \cos\pi/8 = -\cos 7\pi/8$

$\quad c \propto \cos 3\pi/8 = -\cos 5\pi/8$

$\therefore b/c = \dfrac{\cos\pi/8}{\cos 3\pi/8} = \dfrac{\sin 3\pi/8}{\cos 3\pi/8} = \tan 3\pi/8 = 2{\cdot}4142$

and $\quad b^2 + c^2 + c^2 + b^2 = 1 \qquad \therefore b^2 + c^2 = 1/2$

and $\quad c^2(\tan^2 3\pi/8 + 1) = 1/2 \qquad \therefore c = 0{\cdot}2706$

$b = c\tan 3\pi/8$

$= 0{\cdot}6533$

The final row will be $\propto \cos 3\pi/8, \cos 9\pi/8, \cos 15\pi/8, \cos\dfrac{21\pi}{8}$

$= \cos 3\pi/8, -\cos\pi/8, \cos\pi/8, -\cos 3\pi/8$

Hence $c, -b, b, -c$ is correct

3b) In 1-D) $\underline{y} = T\underline{x}$

In 2-D TX will transform columns of X and $(TX^T)^T$ will transform rows of X

But $(TX^T)^T = XT^T$ $\therefore$ $Y = TXT^T$ will transform rows and columns of X and produce a 2D DCT.

Since T is orthonormal, $T^{-1} = T^T$

$\therefore$ $T^T T = T T^T = I$

$\therefore$ $T^T Y T = T^T T X T^T T = X$ giving the inverse transfo

c) Since $X = T^T Y T$, the coeffs in Y tell us how X is built up from basis functions of the form:

$$T^T \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & & & \\ 0 & & 0 & \\ 0 & & & \end{bmatrix} T, \quad T^T \begin{bmatrix} 0 & 1 & 0 & 0 \\ \vdots & 0 & \vdots & \vdots \\ 0 & 0 & 0 & 0 \end{bmatrix} T \quad \text{etc}$$

ie the basis functions are obtained by inverse transforming: $T^T Y(i,j) T$, where $Y(i,j)$ is a 4x4 matrix containing only a single non-zero element at location (i,j) and of amplitude unity.

So, the 1st basis function will be $\begin{bmatrix} a \\ a \\ a \\ a \end{bmatrix} [a \; a \; a \; a] = \begin{bmatrix} a^2 & a^2 & \cdots & a \\ \vdots & & & \vdots \\ a^2 & \cdots & & a \end{bmatrix}$

2nd will be $\begin{bmatrix} b \\ c \\ -c \\ -b \end{bmatrix} [a \; a \; a \; a] = \begin{bmatrix} ab & \cdots & ab \\ ac & \cdots & ac \\ -ac & \cdots & -ac \\ -ab & \cdots & -ab \end{bmatrix}$

3rd will be $\begin{bmatrix} a \\ a \\ a \end{bmatrix} [b \; c \; -c \; -b] = \begin{bmatrix} ab & ac & -ac & -ab \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}$

4th basis fn: $\begin{bmatrix} a \\ -a \\ -a \\ a \end{bmatrix} [a \ a \ a \ a] = \begin{bmatrix} a^2 & \cdots & a^2 \\ -a^2 & \cdots & -a^2 \\ -a^2 & \cdots & -a^2 \\ a^2 & \cdots & a^2 \end{bmatrix}$

5th basis fn: $\begin{bmatrix} b \\ c \\ -c \\ b \end{bmatrix} [b \ c \ -c \ -b] = \begin{bmatrix} b^2 & bc & -bc & -b^2 \\ bc & c^2 & -c^2 & -bc \\ -bc & -c^2 & c^2 & bc \\ -b^2 & -bc & bc & b^2 \end{bmatrix}$

6th basis fn: $\begin{bmatrix} a \\ a \\ a \\ a \end{bmatrix} [a \ -a \ -a \ a] = \begin{bmatrix} a^2 & -a^2 & -a^2 & a^2 \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ a^2 & -a^2 & -a^2 & a^2 \end{bmatrix}$

d) The 2-level DCT of JPXR takes the lowpass subimage from the 1st level (top left corner coeff of 4x4 DCT) and performs another 4x4 DCT on just these coeffs, leaving the other 15 coeffs from each block unchanged [this is ∵ only the Lo-Lo band from level 1 has similar statistics to a normal image and benefits from a further level of transform].
These level 2 coeffs are similar to the lowest 16 coeffs of a 16x16 single-level DCT and give similar coding performance. The advantage of leaving the other level 1 coeffs as they are is that the basis frames from these coeffs are only 4x4 in size, making their artefacts much less visible than if they were 16x16 (or 8x8) in size. Hence we get a better tradeoff of coding performance vs artefact visibility than with the standard 8x8 DCT of JPEG.

(4) a) See 4F8 Lecture notes pp 59 & 60

b) Given the expressions for H_0 and H_1:

$$H_{00}(z) = H_0(z).H_0(z^2)$$

$$= \frac{1}{16}(-z^2 + 2z + 6 + 2z^{-1} - z^{-2})(-z^4 + 2z^2 + 6 + 2z^{-2} - z^{-4})$$

$$= \frac{1}{16}\big(z^6 - 2z^5 - z^4(6+2) - z^3(2-4) + z^2(1+12-6) + z(4+12)$$
$$+ (-2+36-2) + z^{-1}(12+4) + \ldots\ldots\big)$$

$$= \frac{1}{16}(z^6 - 2z^5 - 8z^4 + 2z^3 + 7z^2 + 16z + 32 + 16z^{-1}$$
$$+ 7z^{-2} + 2z^{-3} - 8z^{-4} - 2z^{-5} + z^{-6})$$ ✓

$$H_{01}(z) = H_0(z).H_1(z^2)$$

$$= \frac{1}{16}(-z^2 + 2z + 6 + 2z^{-1} - z^{-2})(-z^2 + 2 - z^{-2})$$

$$= \frac{1}{16}(z^4 - 2z^3 - 8z^2 + 2z + 14 + 2z^{-1} - 8z^{-2} - 2z^{-3} + z^{-4})z^{-2}$$ ✓

whereas

$$G_{00}(z) = \frac{1}{16}\underbrace{(z + 2 + z^{-1})}_{G_0(z)}\underbrace{(z^2 + 2 + z^{-2})}_{G_0(z^2)}$$

$$= \frac{1}{16}(z^3 + 2z^2 + 3z + 4 + 3z^{-1} + 2z^{-2} + z^{-3})$$

$$G_{01}(z) = G_0(z).G_1(z^2) = \frac{1}{16}(z + 2 + z^{-1})(-z^4 - 2z^2 + 6 - 2z^{-2} - z^{-4})z$$

$$= \frac{1}{16}(-z^5 - 2z^4 - 3z^3 - 4z^2 + 4z + 12 + 4z^{-1} - 4z^{-2}$$
$$- 3z^{-3} - 2z^{-4} - z^{-5})z^2$$

c) In 2D the filters would be combined as Lo-Lo, Hi-Lo, Lo-Hi, Hi-Hi

The 2D impulse responses would be the product of a column vector for the vertical filter with a row vector for the horizontal filter, using the above sets of coefficients to define the column / row vectors.

Hence $h_{ab} = h_a h_b^T$ where h_{ab} is the 2D impulse response (point spread function) and h_a, h_b are column vectors representing the column and row filter coefficients.

d) To obtain best performance from the wavelet filters the basis functions of the reconstruction lowpass filters should be as smooth as possible.

By inspection of the H_{00}, G_{00} filters, we see that G_{00} gives a simple triangular response, proportional to (1, 2, 3, 4, 3, 2, 1), whereas H_{00} is a longer and less smooth set of coeffs: (1, −2, −8, 2, 7, 16, 32, 16, 7, 2, −8, −2, 1) in which the local gradient changes.

Hence we choose the G_0 and G_1 filters for the reconstruction (inverse) wavelet transform and the H_0, H_1 filters for the analysis (forward) wavelet transform

END OF SOLUTIONS