2008 IIB 4F8 IMAGE PROCESSING AND IMAGE CODING DR J LASENBY

Solutions: 4F8 2008

ENGINEERING TRIPOS PART IIB

???day ?? April 2008 2.30 to 4

Module 4F8

IMAGE PROCESSING AND IMAGE CODING

1 (a) Perception of images is very much concerned with lines and edges. If a filter phase response is non-linear, then the various frequency components which contribute to an edge in an image will be phase-shifted with respect to each other in such a way that they no longer add up to produce a sharp edge – i.e. dispersion takes place. It is often simplest to enforce the *zero-phase* condition, i.e. insisting that the frequency response is purely real, so that

$$H(\omega_1,\omega_2) = H^*(\omega_1,\omega_2)$$

[10%]

(b) Can first of all do this via straightforward FTs. Firstly we can write the frequency response as

$$H(\omega_1,\omega_2) = H_0 - H_1 H_2$$

where

$$H_0(\omega_1,\omega_2) = \begin{cases} 1 & \text{if } |\omega_1| < \Omega_{U1} \text{ and } |\omega_2| < \Omega_{U2} \\ 0 & \text{otherwise} \end{cases}$$

$$H_1(\omega_1,\omega_2) = \begin{cases} 1 & \text{if } \Omega_{L1} < |\omega_1| < \Omega_{U1} \\ 0 & \text{otherwise} \end{cases}$$

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

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

$$h(n_1,n_2) = \frac{\Delta_1\Delta_2}{(2\pi)^2}\int_{-\pi/\Delta_2}^{\pi/\Delta_2}\int_{-\pi/\Delta_1}^{\pi/\Delta_1}[H_0 - H_1H_2]e^{j(\omega_1 n_1\Delta_1+\omega_2 n_2\Delta_2)}\,d\omega_1\,d\omega_2$$

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

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

(cont.

Evaluating these integrals gives

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

$$-\frac{\Delta_1\Delta_2}{(2\pi)^2}\left\{\left[\left[\frac{e^{j\omega_1 n_1\Delta_1}}{jn_1\Delta_1}\right]_{-\Omega_{U1}}^{-\Omega_{L1}}+\left[\frac{e^{j\omega_1 n_1\Delta_1}}{jn_1\Delta_1}\right]_{\Omega_{L1}}^{\Omega_{U1}}\right]\left[\left[\frac{e^{j\omega_2 n_2\Delta_2}}{jn_2\Delta_2}\right]_{-\Omega_{U2}}^{-\Omega_{L2}}+\left[\frac{e^{j\omega_2 n_2\Delta_2}}{jn_2\Delta_2}\right]_{\Omega_{L2}}^{\Omega_{U2}}\right]\right\}$$

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

$$-\frac{\Delta_1\Delta_2}{(2\pi)^2}\{[2\Omega_{U1}\mathrm{sinc}(n_1\Delta_1\Omega_{U1})-2\Omega_{L1}\mathrm{sinc}(n_1\Delta_1\Omega_{L1})][2\Omega_{U2}\mathrm{sinc}(n_2\Delta_2\Omega_{U2})-2\Omega_{L2}\mathrm{sinc}(n_2\Delta_2\Omega_{L2})]\}$$

$$=\frac{\Delta_1\Delta_2}{(\pi)^2}\{\Omega_{U1}\Omega_{L2}\mathrm{sinc}(n_1\Delta_1\Omega_{U1})\mathrm{sinc}(n_2\Delta_2\Omega_{L2})+$$

$$\Omega_{L1}\Omega_{U2}\mathrm{sinc}(n_1\Delta_1\Omega_{L1})\mathrm{sinc}(n_2\Delta_2\Omega_{U2})-\Omega_{L1}\Omega_{L2}\mathrm{sinc}(n_1\Delta_1\Omega_{L1})\mathrm{sinc}(n_2\Delta_2\Omega_{L2})\}$$

It is also possible to arrive at the above by using the standard results for a rectangular lowpass and bandpass filters.

Standard result for a lowpass filter (H_0) is:

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

Standard result for a separable bandpass filter (H_1H_2) is

$$h(n_1\Delta_1,n_2\Delta_2)=$$

$$\frac{\Delta_1\Delta_2}{\pi^2}[\Omega_{U1}\mathrm{sinc}(\Omega_{U1}n_1\Delta_1)-\Omega_{L1}\mathrm{sinc}(\Omega_{L1}n_1\Delta_1)][\Omega_{U2}\mathrm{sinc}(\Omega_{U2}n_2\Delta_2)-\Omega_{L2}\mathrm{sinc}(\Omega_{L2}n_2\Delta_2)]$$

As well as taking $(H_0-H_1H_2)$ we can also treat the shaded region as the sum of lowpass filters ($|\omega_1|<\Omega_{U1}$ and $|\omega_2|<\Omega_{L2}$, $|\omega_1|<\Omega_{L1}$ and $|\omega_2|<\Omega_{U2}$) minus another lowpass filter ($|\omega_1|<\Omega_{L1}$ and $|\omega_2|<\Omega_{L2}$.

[50%]

(c) We can write s as a Fourier series:

(TURN OVER for continuation of SOLUTION 1

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

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

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

(cont.

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

2 (a) For *spatially stationary processes* $x(\mathbf{n})$, $y(\mathbf{n})$, the cross correlation function is defined as follows and is 'translationally invariant', i.e.

$$R_{xy}(\mathbf{0},\mathbf{n}) \equiv R_{xy}(\mathbf{n}) = E[x(\mathbf{k})y^*(\mathbf{k}-\mathbf{n})] \;\; \forall \mathbf{k}$$

i.e. the cross-correlation between the origin in the x image and the point $\mathbf{n}$ in the y image is independent of where the origin is taken.

The *cross-power spectrum* of two jointly stationary processes $x(\mathbf{n})$ and $y(\mathbf{n})$ is written as $P_{xy}(\boldsymbol{\omega})$ and is given by the FT of the cross-correlation function

$$P_{xy}(\boldsymbol{\omega}) = FT(R_{xy}(\mathbf{n}))$$

Note here that the following convention for the cross correlation (denote as R^{+}_{xy}) is also fine

$$R^{+}_{xy}(\mathbf{n}) \equiv R_{xy}(-\mathbf{n}) = E[x(\mathbf{k})y^*(\mathbf{k}+\mathbf{n})] \;\; \forall \mathbf{k}$$

[10%]

Since P_{yy} is the FT of R_{yy}, we first look at this cross-correlation in terms of our deconvolution equation:

$$R_{yy}(\mathbf{p}) = E\{y(\mathbf{n})y(\mathbf{n}-\mathbf{p})\} \;\text{ where }\; y(\mathbf{n}) = \sum_{\mathbf{m}} h(\mathbf{m})x(\mathbf{n}-\mathbf{m}) + d(\mathbf{n})$$

If signal and noise are uncorrelated and noise is zero mean:

$$R_{yy}(\mathbf{p}) = E\left\{\sum_{\mathbf{m}}\sum_{\mathbf{q}} h(\mathbf{m})x(\mathbf{n}-\mathbf{m})h(\mathbf{q})x(\mathbf{n}-\mathbf{p}-\mathbf{q})\right\} + E\{d(\mathbf{n})d(\mathbf{n}-\mathbf{p})\}$$

$$= \sum_{\mathbf{m}}\sum_{\mathbf{q}} h(\mathbf{m})h(\mathbf{q})E\{x(\mathbf{n}-\mathbf{m})x(\mathbf{n}-\mathbf{p}-\mathbf{q})\} + R_{dd}(\mathbf{p})$$

$$\therefore R_{yy}(\mathbf{p}) = \sum_{\mathbf{m}}\sum_{\mathbf{q}} h(\mathbf{m})h(\mathbf{q})R_{xx}(\mathbf{m}-\mathbf{p}-\mathbf{q}) + R_{dd}(\mathbf{p})$$

$$E\{x(\mathbf{n}-\mathbf{m})x(\mathbf{n}-\mathbf{p}-\mathbf{q})\} = E\{x(\mathbf{n}-\mathbf{m})x(\mathbf{n}-\mathbf{m}-(\mathbf{p}+\mathbf{q}-\mathbf{m}))\}$$

(TURN OVER for continuation of SOLUTION 2

Now take the Fourier transform of each side to give:

$$P_{yy}(\omega) = \sum_{\mathbf{p}} \left\{ \sum_{\mathbf{m}} \sum_{\mathbf{q}} h(\mathbf{m})\, h(\mathbf{q})\, R_{xx}(\mathbf{p}+\mathbf{q}-\mathbf{m}) \right\} e^{-j\omega^T \mathbf{p}} + P_{dd}(\omega)$$

where P_{dd} is the FT of the autocorrelation function of the noise. Interchange order;

$$P_{yy}(\omega) = \sum_{\mathbf{m}} \sum_{\mathbf{q}} h(\mathbf{m})\, h(\mathbf{q}) \sum_{\mathbf{p}} R_{xx}(\mathbf{p}+\mathbf{q}-\mathbf{m})\, e^{-j\omega^T \mathbf{p}} + P_{dd}(\omega)$$

Let $\mathbf{k} = (\mathbf{p}+\mathbf{q}-\mathbf{m})$, then:

$$P_{yy}(\omega) = \sum_{\mathbf{m}} \sum_{\mathbf{q}} h(\mathbf{m})\, h(\mathbf{q}) \sum_{\mathbf{k}} R_{xx}(\mathbf{k})\, e^{-j\omega^T (\mathbf{k}-\mathbf{q}+\mathbf{m})} + P_{dd}(\omega)$$

$$\therefore P_{yy}(\omega) = \left\{ \sum_{\mathbf{m}} h(\mathbf{m})\, e^{-j\omega^T \mathbf{m}} \right\} \left\{ \sum_{\mathbf{q}} h(\mathbf{q})\, e^{j\omega^T \mathbf{q}} \right\} \left\{ \sum_{\mathbf{k}} R_{xx}(\mathbf{k})\, e^{-j\omega^T \mathbf{k}} \right\} + P_{dd}(\omega)$$

$$\therefore P_{yy}(\omega) = |H(\omega)|^2\, P_{xx}(\omega) + P_{dd}(\omega) \qquad (1)$$

as h is real.

Now look at R_{xy};

$$R_{xy}(\mathbf{p}) = E\{x(\mathbf{n})\, y(\mathbf{n}-\mathbf{p})\}$$
$$= E\left\{ \left[\sum_{\mathbf{m}} h(\mathbf{m})\, x(\mathbf{n}-\mathbf{p}-\mathbf{m}) + d(\mathbf{n}) \right] x(\mathbf{n}) \right\}$$

The image $x(\mathbf{n})$ and the noise $d(\mathbf{n})$ are uncorrelated and the noise has zero mean (as before):

$$\therefore R_{xy}(\mathbf{p}) = E\left\{ \sum_{\mathbf{m}} h(\mathbf{m})\, x(\mathbf{n}-\mathbf{p}-\mathbf{m})\, x(\mathbf{n}) \right\} = \sum_{\mathbf{m}} h(\mathbf{m}) E\left\{ x(\mathbf{n})\, x(\mathbf{n}-[\mathbf{p}+\mathbf{m}]) \right\}$$
$$= \sum_{\mathbf{m}} h(\mathbf{m})\, R_{xx}(\mathbf{p}+\mathbf{m})$$

Taking the Fourier transform of each side gives:

$$P_{xy}(\omega) = \sum_{\mathbf{p}} \left\{ \sum_{\mathbf{m}} h(\mathbf{m})\, R_{xx}(\mathbf{p}+\mathbf{m}) \right\} e^{-j\omega^T \mathbf{p}} = \sum_{\mathbf{m}} h(\mathbf{m}) \sum_{\mathbf{p}} R_{xx}(\mathbf{p}+\mathbf{m})\, e^{-j\omega^T \mathbf{p}}$$

Let $\mathbf{p}+\mathbf{m} = \mathbf{k}$, then:

(cont.

$$= \sum_{\mathbf{m}} h(\mathbf{m})\, e^{j\omega^T \mathbf{m}} \sum_{\mathbf{k}} R_{xx}(\mathbf{k})\, e^{-j\omega^T \mathbf{k}}$$

$$\therefore P_{xy}(\omega) = H^*(\omega)\, P_{xx}(\omega)$$

Substituting back into our original equation, $G(\boldsymbol{\omega}) = \frac{P_{xy}(\boldsymbol{\omega})}{P_{yy}(\boldsymbol{\omega})}$, gives:

$$\boxed{G(\boldsymbol{\omega}) = \frac{H^*(\boldsymbol{\omega})\, P_{xx}(\boldsymbol{\omega})}{|H(\boldsymbol{\omega})|^2\, P_{xx}(\boldsymbol{\omega}) + P_{dd}(\boldsymbol{\omega})}}$$

This is the most commonly used form of the Wiener Filter. [40%]

(b) It is helpful to draw up a table which lists $H(i)$, the frequency values and $C(i)$, the cumulative frequency values;

i	1	2	3	4	5	6	7	8
$H(i)$	2	6	8	16	16	8	6	2
$C(i)$	2	8	16	32	48	56	62	64

The transformed levels are given by

$$y_k = \sum_{i=1}^{k} L \frac{N_i}{NM}, \quad k = 1 \ldots 8$$

where $N \times M$ are the dimensions of the image, N_i is the number of pixels in grey level i (equivalent to $H(i)$ above) and L is the range in grey level space. Therefore, $L = 8$, $NM = 64$ and

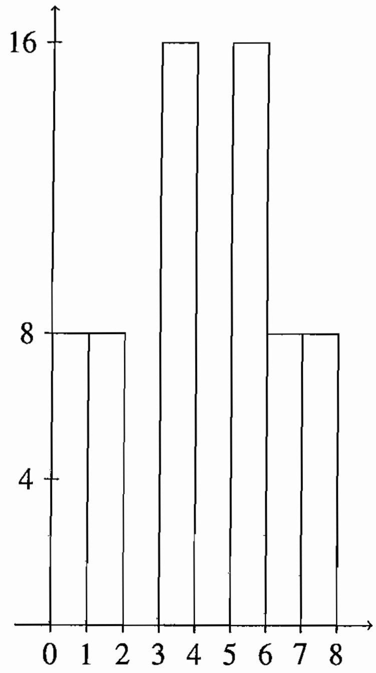
$$y_k = \frac{L}{NM} \sum_{i=1}^{k} N_i = \frac{1}{2} \sum_{i=1}^{k} N_i = \frac{1}{8} C(k), \quad k = 1 \ldots 8$$

We can now add an extra line to our table to show the transformed values:

i	1	2	3	4	5	6	7	8
$H(i)$	2	6	8	16	16	8	6	2
$C(i)$	2	8	16	32	48	56	62	64
$y(i)$	0.25	1	2	4	6	7	7.75	8

From this table it is now easy to sketch the new histogram

(TURN OVER for continuation of SOLUTION 2



We can see from the new histogram that the process has succeeded in spreading out the grey levels more evenly across the scale but that the distribution is far from being uniform. The discreteness of the problem means that the equalisation process tries to do the best job it can according to the rules prescribed. One way to improve on this might be to map to a greater number of levels in the new set – this is precisely what Matlab allows one to do in its `histeq` function and works via interpolation. [50%]

~~END OF SOLUTIONS~~

4F8

3(a) 4x4 DCT matrix:

$n=4$. $\therefore$ Top row $t_{1i} = \sqrt{\frac{1}{4}} = 0.5$ for $i = 1..4$

Rows $2 \to 4$: $\sqrt{\frac{2}{n}} = \sqrt{\frac{2}{4}} = \frac{1}{\sqrt{2}}$

Angles are $\begin{bmatrix} \frac{\pi}{8} & \frac{3\pi}{8} & \frac{5\pi}{8} & \frac{7\pi}{8} \\ \frac{\pi}{4} & \frac{3\pi}{4} & \frac{5\pi}{4} & \frac{7\pi}{4} \\ \frac{3\pi}{8} & \frac{9\pi}{8} & \frac{15\pi}{8} & \frac{21\pi}{8} \end{bmatrix}$

Taking cos of the angles & multiplying by $\frac{1}{\sqrt{2}}$, and then adding in the top row gives:

$$\underline{T} = \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ 0.6533 & 0.2706 & -0.2706 & -0.6533 \\ 0.5 & -0.5 & -0.5 & 0.5 \\ 0.2706 & -0.6533 & 0.6533 & -0.2706 \end{bmatrix}$$

Conditions for orthonormality:

a) rows must be orthogonal (dot product = 0)

b) norm of each row must be unity

Rows 2, 3, 4 each have zero dot product with row 1, since they have symmetric positive & negative terms & sum to zero.

Similarly $\underline{t}_2^T \underline{t}_3 = 0.5[0.6533 - 0.2706 + 0.2706 - 0.6533] = 0$

3 (a) (cont)

$$\underline{t}_3^T \underline{t}_4 = 0.5\left[0.2706 + 0.6533 - 0.6533 - 0.2706\right] = 0$$

$$\underline{t}_2^T \underline{t}_4 = 0.6533 . 0.2706\left[1 - 1 - 1 + 1\right] = 0$$

Norm of rows 1 & 3:

$$\sqrt{0.5^2 + 0.5^2 + 0.5^2 + 0.5^2} = \sqrt{4 . 0.25} = 1$$

Norm of rows 2 & 4:

$$\sqrt{0.6533^2 . 2 + 0.2706^2 . 2} = 1$$

Hence $\underline{T}$ is orthonormal

(b) ~~is 1-D DCT of col~~

$\underline{Y} = \underline{T}\,\underline{X}$ is 1-D DCT of columns of $\underline{X}$

Similarly $\underline{X}\,\underline{T}^T$ is 1-D DCT of rows of $\underline{X}$

$\therefore \underline{Y} = \underline{T}\,\underline{X}\,\underline{T}^T$ is 2-D DCT of ~~rows & cols of~~ $\underline{X}$.

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

$$\therefore \underline{X} = \underline{T}^{-1}\underline{T}\,\underline{X}\,\underline{T}^T\underline{T}^{-T} = \underline{T}^{-1}\underline{Y}\,\underline{T}^{-T}$$

$$= \underline{T}^T\underline{Y}\,\underline{T}$$

Orthonormality allows $\underline{T}$ to be used for both the forward & inverse DCT, since only then is $\underline{T}^{-1} = \underline{T}^T$.

3 (c) The basis functions of the transform ~~as~~ are the separate components that are scaled by each of the transform coefs in turn and are then ~~a~~ summed to give $\underline{X}$ in the inverse transform process. To find them we set each element in $\underline{Y}$ to unity, while the remaining elements are all zero, and calculate ~~$\underline{T}^t$~~ $\underline{T}^T \underline{Y} \underline{T}$.

(d) If

$$\underline{X} = \underline{T}^T \begin{bmatrix} y_{11} & \cdots & y_{14} \\ \vdots & & \\ y_{41} & \cdots & y_{44} \end{bmatrix} \underline{T}$$

~~a~~ if the k^{th} row ~~of~~ of $\underline{T}$ is $\underline{t}_k$,

then component y_{ij} multiplies the i^{th} column of $\underline{T}^T$ and the j^{th} row of $\underline{T}$.

Hence the component of $\underline{X}$ that is proportional to y_{ij} is: $\underline{t}_i^T \, y_{ij} \, \underline{t}_j = y_{ij} \left(\underline{t}_i^T \underline{t}_j \right)$

Note that $\underline{t}_i^T \underline{t}_j$ is a 4x4 matrix – the same size as $\underline{X}$ in each case.

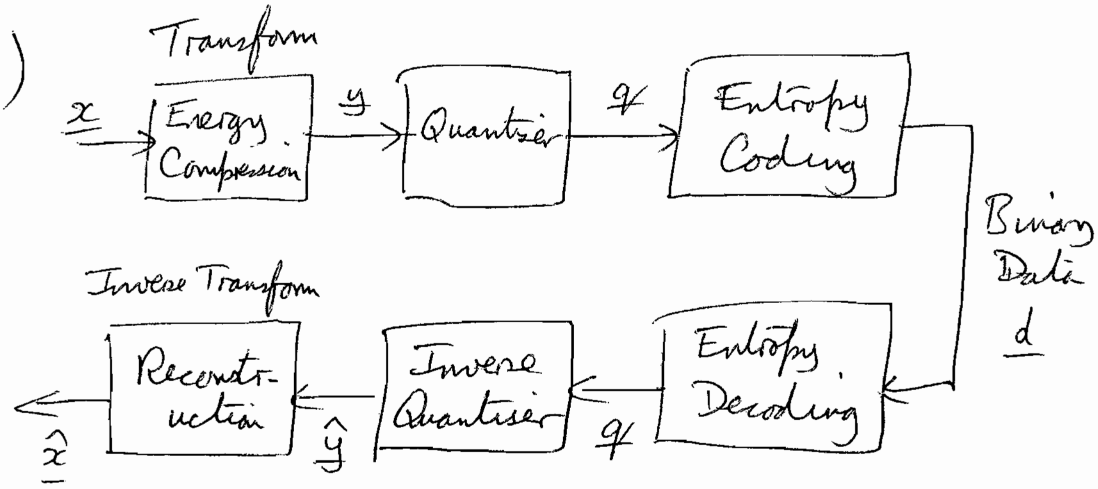
3. (e) If $\underline{X}$ is an image region of uniform intensity, then the only component of $\underline{Y}$ which is non-zero will be y_{11}. This is because the basis function that corresponds to a 4x4 uniform intensity region ~~patch~~ is $\underline{t}_1^T \underline{t}_1 = 0.25 \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$

All of the other components of ~~y~~ $\underline{Y}$ will be zero because of the orthogonality of the rows of $\underline{T}$.

The significance of this result for coding, is that the 4x4 region can be represented by just a single non-zero coefficient, whose value gives ~~the pixel~~ 4 times the pixel intensity of the region, compared with 16 ~~x~~ pixel values being needed if they are coded separately. A second parameter is needed to code the location in $\underline{Y}$ of the non-zero coef, so compression by a factor of approximately $\frac{16}{2} = 8$ has been achieved. Proper entropy coding methods can improve on this a little further.

4F8

4 (a)



If the Transform & Inverse Transform blocks have Perfect Reconstruction (PR) then the only place in the system where information is lost and distortion is introduced is ~~at~~ in the Quantiser. In all other blocks, it is possible to recover the input exactly from the output. Hence the Quantiser is the block where all the coding distortion occurs.

(b) Let y & $\hat{y}$ be the signals before & after

the downsampler shown.

$$\therefore \hat{y}_k = y_k \quad \text{if} \quad k \text{ is even}$$
$$= 0 \quad \text{if} \quad k \text{ is odd.}$$

4(b)(cont)

$$\therefore \quad \hat{Y}(z) = \sum_{k=-\infty}^{\infty} \hat{y}_k \, z^{-k}$$

$$= \sum_{k \text{ even}} y_k \, z^{-k}$$

$$= \frac{1}{2}\left[\sum_{\text{all } k} \left(y_k z^{-k} + y_k (-z)^{-k} \right) \right]$$

$$= \frac{1}{2}\left[Y(z) + Y(-z) \right]$$

$\therefore$ For the two up & down samplers in Fig. 3

$$\hat{Y}_0(z) = \frac{1}{2}\left[Y_0(z) + Y_0(-z) \right]$$

$$\hat{Y}_1(z) = \frac{1}{2}\left[Y_1(z) + Y_1(-z) \right]$$

Now $Y_0(z) = X(z)\, H_0(z)$ & $Y_1(z) = X(z)\, H_1(z)$

& $\hat{X}(z) = \hat{Y}_0(z)\, G_0(z) + \hat{Y}_1(z)\, G_1(z)$

$$= \frac{1}{2}\left[\left(Y_0(z) + Y_0(-z) \right) G_0(z) + \left(Y_1(z) + Y_1(-z) \right) G_1(z) \right]$$

$$= \frac{1}{2}\left[X(z) H_0(z) G_0(z) + X(-z) H_0(-z) G_0(z) + X(z) H_1(z) G_1(z) + X(-z) H_1(-z) G_1(z) \right]$$

$$= \frac{1}{2} X(z) \left[H_0(z) G_0(z) + H_1(z) G_1(z) \right] + \frac{1}{2} X(-z) \left[H_0(-z) G_0(z) + H_1(-z) G_1(z) \right]$$

4(c) The aliased components are those which are proportional to ~~a function of~~ $X(-z)$ in the above expression for $\hat{X}(z)$. Hence to ~~make~~ eliminate them:

$$H_0(-z)\,G_0(z) + H_1(-z)\,G_1(z) = 0$$

If $G_0(z) = z\,H_1(-z)$, then

$$H_0(-z).\,z\,H_1(-z) + H_1(-z)\,G_1(z) = 0$$

$$\therefore H_1(-z)\left[z\,H_0(-z) + G_1(z)\right] = 0$$

Since $H_1(-z) \neq 0$ for a non-trivial solution, we required that $\underline{\underline{G_1(z) = -z\,H_0(-z)}}$

4(d) To obtain PR, we require $\hat{X}(z) = X(z)$.

$\therefore$ As well as the above anti-aliasing condition, we require that

$$\tfrac{1}{2}\left[H_0(z)\,G_0(z) + H_1(z)\,G_1(z)\right] = 1$$

If $H_0(z)\,G_0(z) = P(z)$ —— (1)

then $H_1(z)\,G_1(z) = 2 - P(z)$

But if $G_0(z) = z\,H_1(-z)$ then $H_1(z) = \dfrac{G_0(-z)}{-z}$

$$= -z^{-1}.\,G_0(-z)$$

$$\& \quad G_1(z) = -z\,H_0(-z)$$

$$\therefore H_1(z)\,G_1(z) = \left(-z^{-1}.\,G_0(-z)\right).\left(-z\,H_0(-z)\right) = G_0(-z)\,H_0(-z)$$

$$= P(-z) \quad \text{—— (2)}$$

4 (d)(cont)

$$\therefore \quad P(-z) = 2 - P(z) \quad \left(\begin{matrix}\text{combining} \\ 1 \ \& \ 2\end{matrix}\right)$$

So $$P(z) + P(-z) = 2$$

Let coefs of $P(z)$ be p_k, $k = -\infty \ldots \infty$

For a zero-phase filter, $p_{-k} = p_k$ for all k.

$$P(z) + P(-z) = \sum_k p_k z^{-k} + \sum_k p_k (-z)^{-k}$$

$$= 2 \sum_{k \text{ even}} p_k z^{-k}$$

~~But~~ Hence to make $P(z) + P(-z) = 2$, ~~we~~ we must set $p_0 = 1$ & $p_k = 0$ for k even & $\neq 0$.

The coefs p_k for odd k ~~must~~ can take any values except that p_{-k} must equal p_k to ensure zero phase.

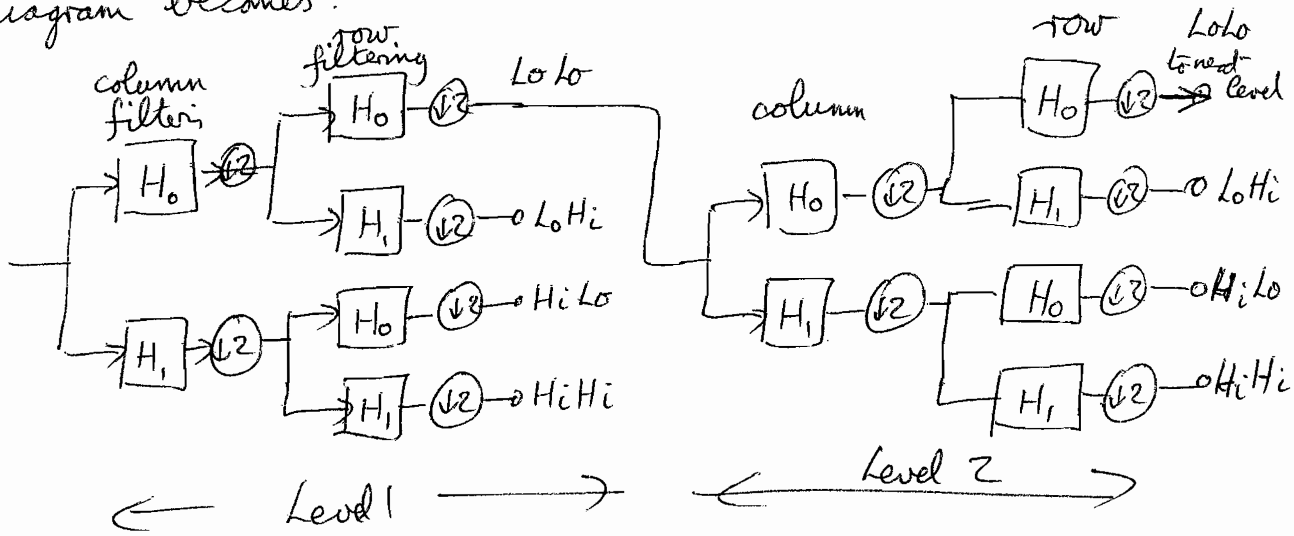
e) To form a wavelet transform, it is necessary to apply the 2-band filter bank iteratively to the lowpass signal at each scale. To ~~apply~~ obtain a 2-D wavelet transform, the 2-band

4 (e) (cont.)

filter bank must be applied to the rows and the columns of the image in turn, resulting in 4 subbands at each scale

Lo-Lo	Lo-Hi
Hi-Lo	Hi Hi

The Lo-Lo subband image is then passed on to the next level of processing. Hence the block diagram becomes:



In order to achieve good energy compression $H_0(z)$ should be a good lowpass filter, and in order to minimise visibility of coding artifacts $G_0(z)$ should also be a good lowpass filter. It is more important for the impulse response of $G_0(z)$ to be smooth, than for $H_0(z)$, because $G_0(z)$ more directly affects the smoothness of the output image.