1. (a) Line Rate = Framerate × Lines per frame

$= 30 \times 400$

$= 12000$ lines/sec

(b) Max. temporal frequency: $f_{t,freq} = f_{s.t}/2 = 60/2 = 30\,HZ$

(c) Max. Vertical frequency: $f_{v,freq} = f_{s.y}/2 = 200/2 = 100\,HZ$

(d) Horizontal Frequency:

$$f_{h.max} = IAR \cdot K \cdot f'_{v.max} = IAR \cdot f'_{vmax} = \frac{4}{3} \times \frac{400}{2} = \frac{800}{3}\ HZ$$

(e) 1D waveform raster signal

$f_y = f_{max} = IAR \cdot K \cdot f'_{vmax} / 2T_l$ (Horizontal freq × line rate)

$$= \frac{4}{3} \times \frac{400}{2} \times 12000 = 3.2\,MHZ$$

(f) To meet the requirement, two conditions must be satisfied:

① $f_c \le f_y - \frac{1}{4} f_y = \frac{3}{4} f_y$

② f_c sits in the middle of two neighboring multiplication of f_l, that is $f_c = (K + \frac{1}{2}) f_l$

Rounding to the lower bound, yields

$\therefore K = 200$

$\Rightarrow f_c = 200.5 \times 12000 = 2.406\,MHZ$

2. (a) perspective projection gives:

$$\frac{x}{F} = \frac{X}{Z}, \quad \frac{y}{F} = \frac{Y}{Z}$$

$$\Rightarrow x = F \cdot \frac{X}{Z}, \quad y = F \cdot \frac{Y}{Z}$$

$$\text{or } X = \frac{Zx}{F}, \quad Y = \frac{Zy}{F}$$

$\therefore$ For a 3D Scene given by $F(X, Y, Z, t)$

the projected 2D Scene can be represented by $f(x, y, t) = F(\frac{Zx}{F}, \frac{Zy}{F}, Z, t)$

(b) The overall effect can be characterized by a convolution integral:

$$g(x, y, t) = \frac{1}{T \cdot \Delta^2} \int_0^t \int_{-\Delta/2}^{\Delta/2} \int_{-\Delta/2}^{\Delta/2} f(x-\alpha, y-\beta, t-\tau)\, d\alpha\, d\beta\, d\tau$$

This can be treated as three separable rectangle functions in x, y and t axis.

$\therefore$ In frequency domain:

$$G(f_x, f_y, f_t) = F(f_x, f_y, f_t) \cdot H_x(f_x) \cdot H_y(f_y) \cdot H_t(f_t)$$

where $H_x(f_x) = \mathrm{Sinc}(f_x \Delta)$ $H_y(f_y) = \mathrm{Sinc}(f_y \Delta)$ $H_t(f_t) = e^{-\pi f_t T} \cdot \mathrm{sinc}(f_t \cdot T)$

($\mathrm{Sinc}(x)$ is defined as $\sin \pi x / \pi x$)

(c) increase Δ will increase the spatial blurring.
increase T will increase the motion blurring.

3. (a) 2D motion and 3D motion are related using perspective projection

$x = F\cdot\frac{X}{Z}$, $y = F\cdot\frac{Y}{Z}$; $x' = F\cdot\frac{X'}{Z'}$, $y' = F\cdot\frac{Y'}{Z'}$

$$\Rightarrow \begin{cases} X = \frac{Zx}{F} \\ Y = \frac{Zy}{F} \end{cases} ; \quad \begin{cases} X' = \frac{Z'x'}{F} \\ Y' = \frac{Z'y'}{F} \end{cases}$$

Since $X' = X + T_x$, $Y' = Y\cos\theta_x - Z\sin\theta_x$; $Z' = Y\sin\theta_x + Z\cos\theta_x$

$$\frac{Z'x'}{F} = \frac{Zx}{F} + T_x \; ; \quad \frac{Z'y'}{F} = \frac{Zy}{F}\cos\theta_x - Z\sin\theta_x ; \quad Z' = \frac{Zy}{F}\sin\theta_x + Z\cos\theta_x$$

$$\begin{cases} x' = \frac{Zx + FT_x}{Y\sin\theta_x + Z\cos\theta_x} = \frac{Zx + FT_x}{\frac{Zy}{F}\sin\theta_x + Z\cos\theta_x} = \frac{FZx + F^2T_x}{Zy\sin\theta_x + FZ\cos\theta_x} = \frac{Fx + \frac{F^2}{Z}T_x}{y\sin\theta_x + F\cos\theta_x} \\ y' = \frac{Zy\cos\theta_x - ZF\sin\theta_x}{Y\sin\theta_x + Z\cos\theta_x} = \frac{Zy\cos\theta_x - ZF\sin\theta_x}{\frac{Zy}{F}\sin\theta_x + Z\cos\theta_x} = \frac{Fy\cos\theta_x - F^2\sin\theta_x}{y\sin\theta_x + F\cos\theta_x} \end{cases}$$

Motion is defined as $dx = x' - x$, $dy = y' - y$

$$\therefore \begin{cases} dx = x' - x = F\cdot\frac{Zx + FT_x}{Z(y\sin\theta_x + F\cos\theta_x)} - x & (*) \\ dy = y' - y = F\cdot\frac{y\cos\theta_x - F\sin\theta_x}{y\sin\theta_x + F\cos\theta_x} - y & \text{(doesn't contain } Z\text{)} \end{cases}$$

(b) $\because Z = a\cancel{X} + c = a\frac{Zx}{F} + c$ $\qquad \frac{F}{Z} = \frac{F - ax}{c}$

$\therefore (F - ax)Z = Fc$ $\qquad x' = \frac{Fx + F\cdot T_x\cdot\frac{F - ax}{c}}{y\sin\theta_x + F\cos\theta_x}$

$\Rightarrow Z = \frac{Fc}{F - ax}$, substitute into (*) gives

$$dx = \frac{\frac{F^2cx}{F - ax} + F^2T_x}{\left(\frac{Fc}{F - ax}\right)(y\sin\theta_x + F\cos\theta_x)} - x = \frac{F^2cx + F^2T_x(F - ax)}{Fc(y\sin\theta_x + F\cos\theta_x)} - x$$

$$= \frac{[F^2c - Fc(y\sin\theta_x + F\cos\theta_x)]x + F^3T_x - F^2T_x ax}{cF(y\sin\theta_x + F\cos\theta_x)}$$

$$= \frac{(F^2c - CF(y\sin\theta_x + F\cos\theta_x - F^2T_x a)x + F^3T_x}{CF(y\sin\theta_x + F\cos\theta_x)}$$

4. Solution:

(a) A single frame: $WH(2R+1)^2$

framerate: 30 frs/sec

$\therefore$ Total number of operations per second: $30WH(2R+1)^2$

(b) Top level: Search range $R/4$, resolution $\frac{W}{4}\cdot\frac{H}{4}$

$\therefore$ Top level operations: $\frac{WH}{16}(\frac{2R}{4}+1)^2 = \frac{WH}{16}(\frac{R}{2}+1)^2$

Intermediate level: Search range S, resolution $\frac{W}{2}\cdot\frac{H}{2}$

$\therefore$ Intermediate level operations: $\frac{WH}{4}(2S+1)^2$

Bottom level: Search range S, resolution WH

$\therefore$ Bottom level operations: $WH(2S+1)^2$

$\therefore$ Total operations per frame: $\frac{WH}{16}(\frac{R}{2}+1)^2 + \frac{5WH}{4}(2S+1)^2$

Total operation per second: $30\left[\frac{WH}{16}(\frac{R}{2}+1)^2 + \frac{5WH}{4}(2S+1)^2\right]$

(c) Saving factors:

$$\alpha = \frac{OPs(EBMA)}{OPs(HBMA)} = \frac{WH(2R+1)^2}{\frac{WH}{16}(\frac{R}{2}+1)^2+\frac{5WH}{4}(2S+1)^2}$$

Consider $S \ll R$ $\therefore (2S+1)^2 \ll (\frac{R}{2}+1)^2$ $\therefore \frac{WH}{16}(\frac{R}{2}+1)^2+\frac{5WH}{4}(2S+1)^2 \approx \frac{WH}{16}(\frac{R}{2}+1)^2$

$\therefore \alpha = \frac{16(2R+1)^2}{(\frac{R}{2}+1)^2} = 16\cdot(\frac{4R+2}{R+2})$, where $R=16$, $\alpha \doteq 22.67$)

$R\to\infty$ $\alpha = 64$

(d),

Advantage: Less computation, more smooth MV field

Disadvantage: Harder HW/SW implementation,

May given lower PSNR in some cases (can't find global minimal but locale minimal)

5. Solution

Optical flow equation, at (x, y) is

$$g_x(\underbrace{x, y}_{=\bar{x}})\, d_x(x, y) + g_y(x, y)\, d_y(x, y) + \underbrace{g_t(x, y)}_{=\psi_2(x,y) - \psi_1(x,y)} = 0$$

Let $d_x(x, y) = \sum \phi_k(x, y)\, d_{k,x}$, $d_y(x, y) = \sum \phi_k(x, y)\, d_{k,y}$

To determine $d_{k,x}, d_{k,y}$ for $k = 1, 2, 3, 4$

we minimize

$$E_{OF}(\bar{d}_k) = \sum_{\bar{x} \in B} \left[g_x(\bar{x}) d_x(\bar{x}) + g_y(\bar{x}) d_y(\bar{x}) + g_t(\bar{x}) \right]^2$$

$$\frac{\partial E}{\partial d_{k,x}} = 2 \sum_{\bar{x}} \left[g_x(\bar{x}) \underbrace{d_x(\bar{x})}_{=\sum \phi_l(\bar{x}) d_{l,x}} + g_y(\bar{x}) \underbrace{d_y(\bar{x})}_{=\sum \phi_l(\bar{x}) d_{l,y}} + g_t(\bar{x}) \right] g_x(\bar{x}) \phi_k(\bar{x})$$

$$\frac{\partial E}{\partial d_{k,y}} = 2 \sum_{\bar{x}} \left[g_x(\bar{x}) d_x(\bar{x}) + g_y(\bar{x}) d_y(\bar{x}) + g_t(\bar{x}) \right] g_y(\bar{x}) \phi_k(\bar{x})$$

Set $\frac{\partial E}{\partial d_{k,x}} = 0$, $\frac{\partial E}{\partial d_{k,y}} = 0$, we obtain the following equation:

$$\sum_l \left(\sum_{\bar{x}} \phi_k(\bar{x}) \cdot g_x^2(\bar{x})\, \phi_l(\bar{x}) \right) d_{l,x} + \sum_l \left[\sum_{\bar{x}} \phi_k(\bar{x}) \cdot g_y(\bar{x}) g_x(\bar{x}) \phi_l(\bar{x}) \right] d_{l,y}$$

$$= - \sum_{\bar{x}} g_t(\bar{x}) g_x(\bar{x}) \cdot \phi_k(\bar{x})$$

$$\Rightarrow \sum_l \left(\sum \phi_k(\bar{x}) g_y(\bar{x}) g_x(\bar{x}) \right) \phi_l(\bar{x}) d_{l,x} + \sum_l \left(\sum \phi_k(\bar{x}) g_y^2(\bar{x}) \phi_l(\bar{x}) \right) d_{l,y} = -\sum g_t(\bar{x}) g_y(\bar{x}) \cdot \phi_k(\bar{x})$$

(See Next Page)

Rearranging the above equations for $k=1,2,3,4$ yields

$$\overset{[H]}{\begin{bmatrix} A_{4\times4} & B_{4\times4} \\ & \\ B_{4\times4} & C_{4\times4} \end{bmatrix}} \overset{\bar{d}}{\begin{bmatrix} d_{1,x} \\ d_{2,x} \\ d_{3,x} \\ d_{4,x} \\ d_{1,y} \\ d_{2,y} \\ d_{3,y} \\ d_{4,y} \end{bmatrix}} = - \overset{\bar{S}}{\begin{bmatrix} \sum_{\bar{x}} g_t(\bar{x})\, g_x(\bar{x})\, \phi_1(\bar{x}) \\ \sum_{\bar{x}} g_t(\bar{x})\, g_x(\bar{x})\, \phi_2(\bar{x}) \\ \vdots \\ \sum_{\bar{x}} g_t(\bar{x})\, g_x(\bar{x})\, \phi_4(\bar{x}) \\ \sum_{\bar{x}} g_t(\bar{x})\, g_y(\bar{x})\, \phi_1(\bar{x}) \\ \vdots \\ \vdots \\ \sum_{\bar{x}} g_t(\bar{x})\, g_y(\bar{x})\, \phi_4(\bar{x}) \end{bmatrix}}$$

$$A = \begin{bmatrix} \sum_{\bar{x}} \phi_1^2(\bar{x})\, g_x^2(\bar{x}) & \sum_{\bar{x}} \phi_1(\bar{x})\phi_2(\bar{x})\, g_x^2(\bar{x}) & \sum_{\bar{x}} \phi_1(\bar{x})\phi_3(\bar{x})\, g_x^2(\bar{x}) & \sum_{\bar{x}} \phi_1(\bar{x})\phi_4(\bar{x})\, g_x^2(\bar{x}) \\ \sum_{\bar{x}} \phi_2(\bar{x})\phi_1(\bar{x})\, g_x^2(\bar{x}) & \sum_{\bar{x}} \phi_2^2(\bar{x})\, g_x^2(\bar{x}) & \sum_{\bar{x}} \phi_2(\bar{x})\phi_3(\bar{x})\, g_x^2(\bar{x}) & \sum_{\bar{x}} \phi_2(\bar{x})\phi_4(\bar{x})\, g_x^2(\bar{x}) \\ \sum_{\bar{x}} \phi_3(\bar{x})\phi_1(\bar{x})\, g_x^2(\bar{x}) & \sum_{\bar{x}} \phi_3(\bar{x})\phi_2(\bar{x})\, g_x^2(\bar{x}) & \sum_{\bar{x}} \phi_3^2(\bar{x})\, g_x^2(\bar{x}) & \sum_{\bar{x}} \phi_3(\bar{x})\phi_4(\bar{x})\, g_x^2(\bar{x}) \\ \sum_{\bar{x}} \phi_4(\bar{x})\phi_1(\bar{x})\, g_x^2(\bar{x}) & \sum_{\bar{x}} \phi_4(\bar{x})\phi_2(\bar{x})\, g_x^2(\bar{x}) & \sum_{\bar{x}} \phi_4(\bar{x})\phi_3(\bar{x})\, g_x^2(\bar{x}) & \sum_{\bar{x}} \phi_4^2(\bar{x})\, g_x^2(\bar{x}) \end{bmatrix}$$

B is similar to A but with $g_x^2(\bar{x})$ replaced by $g_x(\bar{x})\, g_y(\bar{x})$

C is similar to A but with $g_x^2(\bar{x})$ replaced by $g_y^2(\bar{x})$

$$\bar{d} = [H]^{-1} \cdot \bar{S}$$

(prob. 5 Solution ENDS)

6. (1) Lower Bound should be given by entropy:

$$H_W = -\sum_{k=0}^{K_{max}} P_W(k) \log_2 P_W(k)$$

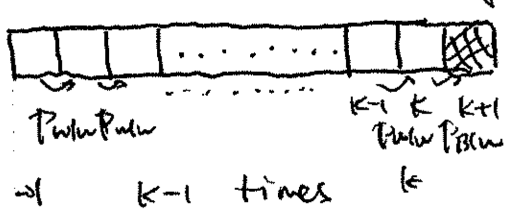
$$H_B = -\sum_{l=1}^{L_{max}} P_B(l) \log_2 P_B(l)$$

(2) Average number of bits per pixel:

$$R = \underbrace{\frac{N_W H_W + N_B H_B}{N}}= \frac{N_W}{N}\left(-\sum_{k=0}^{K_{max}} P_W(k) \log_2 P_W(k)\right) - \frac{N_B}{N}\left(-\sum_{l=1}^{l_{max}} P_B(l) \log_2 P_B(l)\right)$$

(3) $P_W(k) := \{$probability of a white run-length $= k\}$

$:= \{$probability of a white pixel given the previous one is white$\}^{k-1} \cdot \{$probability of a black pixel given the previous one is white$\}$



$\therefore P_W(k) = (P_{W|W})^{k-1} \cdot P_{B|W}, \quad k = 1 \sim K_{max}$

Similarily: $P_B(l) = (P_{B|B})^{l-1} P_{W|B}, \quad l = 1 \sim l_{max}$

$\because P_{W|W} = 1 - P_{B|W}, \quad P_{B|B} = 1 - P_{W|B}$

$\therefore P_W(k) = (1 - P_{B|W})^{k-1} \cdot P_{B|W}$

$P_B(l) = (1 - P_{W|B})^{l-1} \cdot P_{W|B}$

(Cont.)

(d)

$$B_W = \sum_K P_W(K) \log P_W(K)$$

$$= \sum (P_{W|W})^{K-1} P_{B|W} \log [(P_{W|W})^{K-1} P_{B|W}]$$

$$= \Big[\sum \underbrace{P_{B|W} P_{W|W}^{K-1}}_{P_W(K)} \log P_{B|W} + P_{B|W} \sum (K-1) P_{W|W}^{K-1} \log P_{W|W} \Big]$$

$$= \underbrace{\sum P_W(K)}_{\text{"1"}} \log P_{B|W} + P_{B|W} (\log P_{W|W}) \cdot \sum (K-1) P_{W|W}^{K-1}$$

$$= \log P_{B|W} + P_{B|W} (\log P_{W|W}) \cdot P_{W|W} (\sum (K-1) P_{W|W}^{(K-2)})$$

$$= \log P_{B|W} + P_{B|W} (\log P_{W|W}) \cdot P_{W|W} (\underbrace{\sum (P_{W|W})^{K-1}}_{})'$$

$$\left(\frac{1}{1-P_{W|W}}\right)' = \frac{1}{(1-P_{W|W})^2} = \frac{1}{P_{B|W}^2}$$

$$= \log P_{B|W} + \frac{1-P_{B|W}}{P_{B|W}} \log (1-P_{B|W})$$

Similarily: $B_B = \log P_{W|B} + \frac{1-P_{W|B}}{P_{W|B}} \log (1-P_{W|B})$

(e) Average bits per pixel:

$$\frac{L_W}{L_W+L_B} \cdot R_W + \frac{L_B}{L_W+L_B} \cdot R_B = \frac{L_W}{L_W+L_B} \cdot \frac{B_W}{L_W} + \frac{L_B}{L_W+L_B} \cdot \frac{B_b}{L_b}$$

$$= \frac{B_W + B_b}{L_W + L_b}$$

while $L_W = \sum_{K=0}^{\infty} K P_W(K) = \sum K (P_{W|W})^K P_{B|W} = P_{B|W} \cdot P_{W|W} (\sum P_{W|W}^K)'$

$$= P_{B|W} \cdot P_{W|W} \cdot \frac{1}{P_{B|W}^2}$$

$$= \frac{P_{W|W}}{P_{B|W}}$$

Similarily $L_B = \frac{P_{B|B}}{P_{W|B}}$

7\. Code: [mvx, mvy, pimg] = BMA (img1, img2, R. B, width, height)

+10

```
% boundary processing
Y1 = zeros(height + 2*R, width+2*R); Y2 = Y1;
Y1 (R+1 : R+height, R+1 : R+width) = img1;
Y2 (R+1 : R+height, R+1, R+width) = img2;
% EBMA
for i = (1+R): B :( height - B + R+1)
    for j = (1+R). B (width - B + R+1)
        MAD_min = 256* B * B.
        for k = -R : R
            for l = -R : R
                dif = abs ( Y1 (i: i+B-1, j : j+B-1) - Y2 (i+k, i+k+B-1, j+l, j+l+B-1));
                MAD = sum(sum(dif));
                if MAD < MAD_min
                    MAD_min = MAD   dy = k, dx = l; end;
                end;
        end;
        iblk = 1 + floor((i-1-R)/B); Jblk = 1+ floor((j-1-R)/B).
        mvx(iblk, jblk) = dy; mvy (iblk, jblk) = dx;
        pimg (i : i+B-1, j : j+B-1) = Y2 (i+dy : i+dy+B-1, j+dx: j+dx+B-1)
    end;
end;
% half-pel EBMA implementation
% Interpolation
    row = 2* height; col = 2* width;
    for m = 1 : height
```

```
n=1: width
if (m < height) && (n <width)
tmp(2*m-1 ,2*n-1)=Y1(m,n)
tmp (2*m-1, 2*n) = (Y1(m,n)+ Y1(m,n+1))/2);
tmp (2*m, 2*n-1) = (Y1(m,n)+Y1(m+1, n)/2);
tmp (2*m , 2*n ) =(Y1(m,n)+Y1(m,n+1)+Y1(m+1,n)
                   +Y1(m+1, n+1)) /4;
else
tmp ( 2*m-1 : 2*m , 2*n-1 : 2*n) = Y1(m,n);
end;  % tmp2 corresponds to Y2 interpolation, similar to ab
% Search -1 to 1
R2 = 2*R
Y11=zeros( row + 2*1; col + 2*1);
Y11 (1+1: row+1 , 1+1 : col+1) = tmp(:,:);
Y22=zeros ( row + 2*1 , col + 2*1 );
Y22 ( 2 : row+1 , 2 : col+1 ) = tmp2;
for i= 2 : B : row-B+2
  for j=2: B: col-B+2
    MAD_min = 256 * B*B;
     for R= -1 : 2 : 1  % Skip integer candidate
      for l = -1 : 2 : 1
       temp= 2* mvx (ceiling (i-1)/B, ceiling (j-1)/B );
       diff = Y22 (i+k +temp : i+k+temp +B-1,
                  j+l+temp : j+l+temp+B-1)
           -Y11 (i:i+B-1 , j : j+l-1);
```

```
MAD = sum(sum(abs(diff)));
if MAD < MAD_min
    MAD_min = MAD; dyy=k, dxx=l
end;
iblk1=1+floor((i-1-R)/B); jblk1=1+floor((j-1-R)/B);
mvx(iblk1, jblk1) = dy*2+dyy;
mvy(iblk1, jblk1) = dx*2+dxx;
% new motion vector is original one, multiplied by 2, plus the offset
% dxx and dyy
pimg = Y22(i+2*dy+dyy: i+dy*2+dyy+B,
           j+2*dx+dxx: j+2*dx+dxx+B);
end;
end;
```