Polytechnic University, Dept. Electrical and Computer Engineering
EL6123 --- Video Processing, S12 (Prof. Yao Wang)
Solution to Midterm Exam
Closed Book, 1 sheet of notes (double sided) allowed

---

1. (5 pt)

To solve this problem, we need to determine the maximum horizontal and vertical angular frequency that the viewer can see if the screen of the size $w \times h$ is represented by $N \times M$ pixels.

To determine vertical angular frequency, we need to express vertical viewing angle θ in terms of h and d.

From the given figure,

$$\tan\frac{\theta}{2} = \frac{h/2}{d}$$

$\theta = 2\tan^{-1}\frac{h}{2d} \approx \frac{2h}{2d} = \frac{h}{d} = \frac{h}{d}\frac{180^\circ}{\pi}$(degree). (1 pt)

With M rows, there are at most $M/2$ cycles/picture height. (1 pt)

So the maximum angular frequency is

$f_{v,max} = \frac{M/2}{\theta} = \frac{Md\pi}{2h\cdot 180^\circ}$ (cycles/degree) (1 pt)

Let $f_{v,max} \le 10$, $M \le \frac{3600\cdot h}{\pi d}$. (1 pt)

Similarly, $f_{h,max} = \frac{N/2}{\theta} = \frac{Nd\pi}{2w\cdot 180^\circ}$.

Let $f_{h,max} \le 10$, $N \le \frac{3600\cdot w}{\pi d}$. (1 pt)

For example, for an SDTV monitor, $w = \frac{4}{3}h$. Assuming$d = 3h$, we have $M \le \frac{3600\cdot h}{\pi 3h} \approx 383$, $N \le \frac{4}{3}M \approx 511$.

Note that the above solution takes the view that having a resolution greater than what the human eye can tell is useless and waste of resources, and hence require $f_{v,max} \le 10$. An alternative view, which is also reasonable is to require $f_{v,max} \ge 10$, which says that the display should be designed to at least match the human sensitivity. In that case, the final solution would be $M \ge \frac{3600\cdot h}{\pi 3h} \approx 383$ and $N \ge \frac{4}{3}M \approx 511$. Both solutions are considered correct.

2. (10pt)

Method a) Pro: can render highest motion up to temporal frequency $f_{t,\max}/2$ correctly, without temporal aliasing; Con: cannot render rapid vertical changes correctly. When the actual video contains patterns with vertical frequency above $f_{v,\max}/4$, vertical aliasing can occur. If prefiltering is used vertically to limit the vertical frequency to $f_{v,\max}/4$, vertical blurring will appear.

Method b) Pro: can render highest vertical details up to $f_{v,\max}/2$ correctly; Con: cannot render fast moving objects with temporal frequency beyond $f_{t,\max}/4$ well. If prefiltering in termporal direction is not applied, a video with fasting moving objects will appear jumpy. With filtering, the motion will appear blurred.

Method c) Pro: can render highest motion up to temporal frequency $f_{t,\max}/2$ correctly. In regions containing non-moving objects, the effective vertical sampling rate is $f_{v,\max}$ because the two fields with alternative lines are merged. Hence the camera can render vertical frequency up to $f_{v,\max}/2$ in stationary regions. When objects with high spatial details are also fast moving, the human eye is less sensitive to the spatial details because of the motion, so the spatial aliasing may not be easily noticeable. Cons: fast moving objects (especially vertical motion) with high vertical frequency cannot be rendered properly. Visually these appears as line crawls. Horizontally fast moving vertical lines can appear jagged.

Overall: under the given constraint on the total line rate, Method c) provides the best tradeoff, taking into account of the effective vertical sampling rate in stationary regions, and the human visual system properties.

Grading note: Some of you just state method a) has high temporal frequency, but low spatial frequency. First of all, “frequency” should be changed to “sampling rate”. Second, this merely restates the method, but not its advantage and disadvantage in terms of video artifacts that may or may not occur. For method c): if you do not mention that the effective vertical sampling rate is $f_{v,\max}$ for stationary regions, and the fact that human eye is less sensitive to spatial details in fast moving objects, 3 points will be taken away. For the question on which method is best, some students said it depends on the video content. But the question really is which method provides the best tradeoff among all likely video scenes that the camera may capture. Hence, those answers are not acceptable. The discussion on each method counts 3 pts, the last one counts 1 pt.

3. (10pt)

Recall that that the temporal frequency ft is related to the spatial frequency (fx, fy) and motion (vx, vy) by: ft = fx vx + fy vy

We will easily observe change in time when vertical bars moving horizontally. This is because in this case fx \= 0, vx \= 0, so that ft \= 0.

We will not observe change in time when horizontal bars moving horizontally. This is because in this case fx=0, vy=0, so that ft=0. (the spatial frequency and motion are orthogonal).

Grading: if you answer regarding whether a change is observed is correct but you did not explain correctly in terms of relation between ft, fx,fy, vx,vy, you will only get 2 pt in each part. The explanation counts 3 pt each. Some students try to explain this in terms of progressive vs. interlaced scan. Since I did not ask question related to scanning method, those answers are totally irrelevant.

4. (10pt)

a. If we ignore the operations for frame interpolation, the number of operation is $MN(4R+1)^2 \cdot f_s$. (2 pt)

b. At top level, with search range of R/2 and integer-pel accuracy, the number of operation is $\frac{M}{2}\frac{N}{2}(R+1)^2 \cdot f_s = \frac{1}{4}MN(R+1)^2 \cdot f_s$. (1 pt)

At the bottom level, with search range of 1 and half-pel accuracy, the candidates to be searched in each dimension are in $[-1, -0.5, 0, 0.5, 1]$, therefore the total number of candidates for each block is 5^2=25, and the number of operation is $25MN \cdot f_s$. (1 pt)

(The above result can also be obtained by using the general solution from (a) with $R = 1$)

The total number of operation is $MN\left[\frac{1}{4}(R+1)^2 + 25\right] \cdot f_s$. (1 pt)

For $R > 1$, method (b) always has lower complexity than (a). When R is large, the reduction ratio is roughly $\frac{(4R+1)^2}{\frac{1}{4}(R+1)^2} = 64!$ (1 pt)

c. Method (a) will provide more accurate motion estimation based on MSE. (2 pt)

Method (b) is more likely to yield more physically correct, smoother motion field. (2 pt)

5. (10 pt)

To be added later

6. (10 pt)

a. The entropy for X_1 is $H(X_1) = -\sum_{l=1}^{L} P_l \log_2 P_l$. (2 pt)

The entropy for X_2 is $H(X_2) = -\sum_{l=1}^{L} P_l \log_2 P_l$. (1 pt)

If we code X_1 and X_2 separately, the lower bound of bit rate for coding one symbol is

$$\frac{H(X_1) + H(X_2)}{2} = H(X_1) = -\sum_{l=1}^{L} P_l \log_2 P_l$$

(1 pt)

b. The joint probability mass function of $\{X_1, X_2\}$ taking $\{s_i, s_j\}$ is

$$P(X_1, X_2) = P(X_1 = s_i)P\big(X_2 = s_j \big| X_1 = s_i\big) = \begin{cases} P_i & i = j \\ 0 & i \neq j \end{cases}$$

(1 pt)

The joint entropy of $\{X_1, X_2\}$ is

$$H(X_1, X_2) = -\sum_{X_1 \in \{s_l\}} \sum_{X_2 \in \{s_l\}} P(X_1, X_2) \log_2 P(X_1, X_2) = -\sum_{l=1}^{L} P_l \log_2 P_l = H(X_1)$$

(2 pt)

The lower bound of bit rate is

$$\frac{H(X_1, X_2)}{2} = -\frac{1}{2}\sum_{l=1}^{L} P_l \log_2 P_l = \frac{H(X_1)}{2}$$

(1 pt)

Method (b) is more efficient, which is expected. Since X_2 is correlated with X_1, it contains no extra information, therefore there is no need to code X_2 separately. The joint entropy would be exactly the same as the entropy of coding X_1 alone. (2 pt)

7. (10 pt)

(a) For each block, the number of bits is K^2R. (2 pt)

The codebook length is $L = 2^{K^2R}$. (1 pt)

(b) To compare the vector with each codeword, we need $K \times K = K^2$ operations. (2 pt) With L codewords, the total number of operation is $LK^2 = 2^{K^2R}K^2$. (1 pt)

(c) Assume the image size is $M \times M$. For each block, the number of operation is LK^2. There are $\frac{M}{K} \times \frac{M}{K}$ blocks, therefore the total number of operation is $\frac{M}{K} \times \frac{M}{K} \times LK^2 = M^2L = M^2 2^{K^2R}$, which increases exponentially as block size K increases. (2 pt)

With a larger block size, one can exploit the redundancy over a larger region better, and hence can represent a block more accurately when the bit rate per pixel is the same, or represent a block with the same accuracy using a lower bit rate.

8. (15 pt)

a. Fig.(b) is likely to yield less quantization error, (3 pt) because there are more points in Fig. (c) that are farther away from their closest codewords (e.g. those points in the bottom left and right corners of the triangle are farther away from their codeword). (2 pt)

b. Fig.(b) satisfies the nearest neighbor condition. (2 pt) Fig.(c) does not satisfy the nearest neighbor condition because the partition line is not halfway between the two codewords. (2 pt)

c.

Method (1)

Because of symmetry, we only need to calculate the right partition.

$$MSE = 2\int_0^1 \int_0^{1-x} p(x,y)[(x-a)^2 + (y-b)^2]dydx$$
$$= \int_0^1 \int_0^{1-x} 2Ae^{-x}e^{-y}[(x-a)^2 + (y-b)^2]dydx$$

(This integration is difficult to calculate, however, we could calculate a and b without knowing MSE.) (2 pt)

The minimal MSE requires $\frac{\partial}{\partial a}MSE = 0$ and $\frac{\partial}{\partial b}MSE = 0$, which yields, for determining a:

$$\int_0^1 \int_0^{1-x} p(x,y)(x-a)dydx = 0 \rightarrow$$

$$\text{a} = \int_0^1 \int_0^{1-x} p(x,y)x\, dydx \Big/ \int_0^1 \int_0^{1-x} p(x,y)\, dydx = 2\int_0^1 \int_0^{1-x} p(x,y)x\, dydx$$

Note that we made use of the fact that the denominator = ½ because that is integration of the pdf over half of its symmetric range. A similar equation can be obtained for b.

Completing the above integral yields $a = b = \frac{2e-5}{2e-4} \approx 0.3039$. (4 pt. You will get full points if you have the above equation correctly without the actual values.)

Method (2)

Using centroid condition directly

$$a = \frac{\int_0^1 \int_0^{1-x} p(x,y)xdydx}{\int_0^1 \int_0^{1-x} p(x,y)dydx} = \frac{\left(1-\frac{5}{2e}\right)A}{\left(1-\frac{2}{e}\right)A} = \frac{2e-5}{2e-4}$$

(2 pt)

Similarly, $b = a = \frac{2e-5}{2e-4}$. (2 pt)

MSE is the same as in method (1). (2 pt)

9. (20 pt)

Major steps:

(1) Down-sample img1 and img2 to prepare for top level EBMA. (1 pt)

(2) Pre-filtering should be applied in the down-sampling. (1 pt)

(3) Using down-sampled images to perform EBMA at top level (with integer-pel precision and search range R1). (5 pt)

(4) Perform HBMA at bottom level (with integer-pel precision, and search range R2 centered at the solution obtained from top level). (5 pt)

(5) Interpolate img2 (target frame) with pre-filtering. (2 pt)

(6) Perform half-pel refinement at bottom level, with search range 1.5, centered at the solution obtained from step 4). (5 pt)

(7) You should ensure that the block used for prediction is within frame boundary. (1 pt)