ASSIGNMENT #1 (EC720 A1)
"Digital Video Processing"
Date: September 13, 2017
Due date: September 27, 2017

1. *3-D sampling* (30 points).

   Assume a camera can capture scene luminance with sensor that applies 3-D sampling in $x-y-t$ space: either progressive (P), or line-interlaced (LI) or field-quincunx (FQ), described by the corresponding sampling matrices:

$$\boldsymbol{V}_P = \begin{bmatrix} X & 0 & 0 \\ 0 & X & 0 \\ 0 & 0 & X \end{bmatrix}, \qquad \boldsymbol{V}_{LI} = \begin{bmatrix} X & 0 & 0 \\ 0 & X & X/2 \\ 0 & 0 & X \end{bmatrix}, \qquad \boldsymbol{V}_{FQ} = \begin{bmatrix} X & 0 & X/2 \\ 0 & X & X/2 \\ 0 & 0 & X \end{bmatrix},$$

   (a) Sketch the projection of each lattice onto the $x-y$ plane by marking the locations from different times t with different symbols (e.g., circle, square, triangle, ...).

   (b) Sketch the projection of each lattice onto the $t-y$ plane by marking the locations from different horizontal positions x with different symbols.

   (c) Find the sampling matrix of the reciprocal lattice in each case by hand (you may verify your result using *Matlab* or *Maple*).

   (d) Sketch the projection of each reciprocal lattice onto the $f_x - f_y$ plane by marking the locations from different frequencies f_t with different symbols.

   (e) Sketch the projection of each reciprocal lattice onto the $f_t - f_y$ plane by marking the locations from different frequencies f_x with different symbols.

   (f) Sketch a perspective view of the Voronoi cell of the reciprocal lattice in each case to the best degree you can (make your sketch sufficiently large).

2. *3-D sampling – Matlab* (30 points).

   For the above sampling matrices assume certain value of X, for example $X = 1.0$, and perform the following steps:

   (a) Write a function in *Matlab* to plot $10\times10\times10$ points (first octant) of each lattice from Problem 1. Use different colors for different times t. Remember to label the axes. Print this plot for each lattice from the same viewing angle to clearly show differences between lattices.

   (b) Use the above function to plot the reciprocal lattices. Use different colors for different frequencies f_t. Remember to label the axes.

   (c) Plot the Voronoi cell of each reciprocal lattice. For example, you can display a dense set of points around the origin of the coordinate system that are within this Voronoi cell (closer to the origin than to the nearest lattice point). You may need to experiment with the number of points to show. Does your result support the shape of the Voronoi cell you drew in Problem 1?

Print the lattices, reciprocal lattices and Voronoi-cell cloud points from a suitable perspective to show differences. Include your *Matlab* code with your solutions.

3. *Luminance equalization* (20 points).

   Some of you may remember the famous special effect in the movie "Matrix" – camera swings around a body suspended in air. In this case a few dozen cameras were placed around the scene and recorded video in sync. Then, same-time frames from all cameras were played back as a video sequence thus creating a "frozen look around" effect.

   In this case, cameras must be calibrated with respect to one other. Otherwise, images may differ in terms of luminance and color. A similar danger of luminance and color variations can occur in a single camera across time, for example when camera's automatic gain control (AGC) reacts aggressively to illumination change. Since most motion and depth (structure) estimation algorithms assume luminance/color similarity between video frames, erroneous estimates result. However, if the luminance/color differences are global (i.e., apply to the whole image not to a local area), then one can attempt to compensate them.

   Let I_k and I_l be luminances of two frames captured by two different cameras, for example left and right cameras in a stereo setup. Assume that these luminances have sample means μ_k and μ_l, and variances σ_k^2 and σ_l^2, respectively. The goal is to equalize the luminance between these two frames.

   (a) Assuming a memoryless, linear transformation $\widehat{I}_k[\boldsymbol{x}] = \alpha I_k[\boldsymbol{x}] + \beta$, where $\boldsymbol{x}$ is the spatial location, and $\widehat{I}_k$ is an equalized luminance frame, find the parameters α and β such that the variance and mean of $\widehat{I}_k$ are equal to those of I_l.

   (b) Explain what type of luminance changes between those two frames will be accurately compensated for by this transformation and for what type of changes it will not work well.

4. *Luminance equalization – Matlab* (20 points).

   Write a *Matlab* function to implement the transformation from Problem 3, and apply it to image *Boxes_left.jpg* in order to match its appearance to the image *Boxes_right.jpg*, both available on the course web site in "Assignments/Images". Since these are color images, apply your transformation to each of the color components (R, G, B). Do not use *Matlab's* functions to compute the component's mean and variance, but instead write your own.

   Print the original images and the transformed left image. With your results include a table with means and variances of R, G, B components for all three images as well as the computed transformation parameters α and β. Include your *Matlab* code.