Computer Vision **Name:** _____
Prof. Bolton **Net ID:** _____
Assignment 2

This assignment contains 2 pages (including this cover page) and 3 questions. Total of points is 100.

Conditions: Groups of 2 or less.

Write or type your answers neatly and clearly on standard paper. Include your name and Net ID. Follow submission instructions as indicated on Canvas.

---

1. (10 points) What is a BDRF? Identify and describe each of its parameters. Create an illustration to support your description.

2. (10 points) Denote the transformation from RGB color space to XYZ color space. Solve for the reverse map: from XYZ to RGB.

3. (80 points) Coding Exercise. Note that there are a variety of image formats: jpg, tiff, png, raw, ... Most formats are compressed. Some are lossy (some information is lost in the compression), some are lossless (full image recovery). In the future we will investigate these encodings, but for now, we will simply assume that images are simply a 2-d arrays of pixels where each pixel x is a value for intensity in grayscale $x \in R$ or in color $x \in \mathbb{R}^3$.

    A. Using an existing image loading function, load a color image (e.g. text.jpg). (EG if you are using Matlab, use imread). Investigate the image. What is the size of the image plane (in pixels)? Estimate the color resolution, that is, given the data type used and range of values observed, what is the maximum number of colors possible for a single image pixel?

    B. Transformation. Assume that you are a professional photographer taking a picture of a book. Assume that you visit of professional photo studio and prepare and orient the book for an optimal perspective, i.e., the normal which joins the center of mass of the book is also normal to the image plane and intersects the optical center. Further assume that you take an excellent image of the text. After you return to the office, you realize you need an image of the text from another perspective, specifically, as though you were viewing the text from below and to the left. For example, after your "perfect shot" assume you took two steps to the left, then got down on the floor and pointing the camera toward the center of the book. If effect, you have translated and rotated the camera. (See visual examples in Prince Figures 15.5 and 15.7) Rather than go back to the studio, you instead decide to create the new image by transforming the existing image you have.

Your goal is to implement this transformation as a function. Write a function named belowLeft (...) that takes as an input the filename $< filename >$. $< ext >$ of a color image and a list of parameters needed to complete the transformation. The function then transforms the image as though the camera was located below and to the left.

As noted, please assume the cameras optical axis still coincides with the texts center of mass. Please include the parameters that are necessary for this transformation as inputs to your function. The function then writes a new image named $< filename > BL. < ext >$.

Please organize the following information in the pdf submission. Also, please present your description as though it were a brief report. For example, please briefly discuss and explain your responses, supporting images, etc:

1. Experiment with using Euclidean, affine, and/or projective transformations to get the desired effect. What is the result?
2. Identify the type of transformation necessary to complete this transformation with the desired effect. Also identify and describe the input parameters required by your implementation. Justify your assertion and implementation with a mathematical discussion and imagery of the resulting transformation(s).
3. Include the source code in the pdf that is submitted. PLEASE comment your code.
4. Include "before" and "after" image examples to support your discussion.

Conditions for Coding: For this coding exercise, your implementation of this transformation can only make use of basic operations matrix multiplication, addition, ... (Please, no built-ins which perform transformations.) The only built-in functions that should be used are those which read in an image file, write out an image file, those that might allocate memory, and possibly IO operations (if you wish to have an interactive implementation).