ECE 4554 / 5554: Computer Vision: Homework 3

Fall 2022

Instructions

- The assignment is due at Canvas on Oct. 1 before 11:59 PM. Late submissions are allowed at the cost of 1 token per 24-hour period. A submission received only a minute after midnight will cost an additional token.
- Please review the Honor Code statement in the syllabus. This is an "individual" assignment, not a "team" assignment. The work that you submit for a grade must be your own.
- The assignment consists of 4 analytical problems, which are presented here, and 2 "machine problems", which require work using Colab. Notice that one of the analytical problems is required for 5554 students, but is optional (extra credit) for 4554 students.
- For the analytical problems, prepare an answer sheet that contains all of your written answers in a single file named `Homework3_Problems1-4_USERNAME.pdf`. (Use your own VT Username.) Show your work. Handwritten solutions are allowed, but they must be easily legible to the grader.
- Your solutions to the machine problems should be submitted together as one Jupyter notebook file. Details are provided at the end of this assignment. For any coding problem, the notebook file that you submit must be compatible with Google Colab. Your code should execute after the grader makes only one change to your file, which is the location of the working directory. If the notebook file does not execute, then the grader will be tempted to assign a grade of zero for the coding problems.
- After you have submitted to Canvas, please download the files that you submitted and verify that they are correct and complete. *The files that you submit to Canvas are the files that will be graded.*

Problem 1. (10 points.) We have discussed the 2D *planar perspective transformation*, also known as 2D *homography*, which maps a point (x, y) to a new location (x', y') in the plane. This transformation can be represented using the equation $\begin{bmatrix} s \cdot x' \\ s \cdot y' \\ s \end{bmatrix} = \boldsymbol{H} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$, where $\boldsymbol{H} = \begin{bmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & h_{22} \end{bmatrix}$. When using homogeneous coordinate representation as shown here, recall that s is simply a scalar term that is to be eliminated when solving for (x', y').

As discussed in the textbook near equation (2.20), there are only 8 degrees of freedom in this equation. For this reason, some formulations set $h_{22} = 1$. It may help you when working on coding problems later if you do *not* constrain h_{22} to be 1.

a) Consider the case that you are given n corresponding pairs of points, where $n \geq 4$, and you want to use those points to determine the parameters h_{ij}. For example, assume that the following correspondences are known:

$$
\begin{aligned}
(x_1', y_1') &\leftrightarrow (x_1, y_1) \\
(x_2', y_2') &\leftrightarrow (x_2, y_2) \\
(x_3', y_3') &\leftrightarrow (x_3, y_3) \\
(x_4', y_4') &\leftrightarrow (x_4, y_4) \\
&\;\;\cdots \\
(x_n', y_n') &\leftrightarrow (x_n, y_n)
\end{aligned}
$$

Show how to derive one matrix equation that represents the relationship between these all of these scalar values (not including s). The form of the equation should be $\boldsymbol{Q}\,\boldsymbol{a} = \boldsymbol{0}$, where $\boldsymbol{a}$ is a 9x1 vector that contains the individual homography parameters only; $\boldsymbol{Q}$ is a matrix of size $2n$x9 that you specify containing known values; and $\boldsymbol{0}$ represents the $2n$x1 vector containing only values of 0. For this part of the problem, you do not need to solve for the parameter vector $\boldsymbol{a}$. *Hint*: you may find some inspiration in the derivation near the end of packet 9, although those lecture slides are discussing a problem that is different from 2D homography.

b) Continuing from part (a), a least-squares solution to parameter vector $\boldsymbol{a}$ is the eigenvector associated with the smallest eigenvalue of the matrix $\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{Q}$. Use this approach to find a numerical solution for $\boldsymbol{H}$ for the following point correspondences:

$$
\begin{aligned}
(x_i', y_i') &\leftrightarrow (x_i, y_i)\\
(3.0, 2.0) &\leftrightarrow (0, 0)\\
(3.67, 2.0) &\leftrightarrow (1, 0)\\
(3.5, 2.5) &\leftrightarrow (1, 1)\\
(3.0, 3.0) &\leftrightarrow (0, 1)
\end{aligned}
$$

You may use any matrix solver to find the numerical values. For example, the NumPy functions `np.linalg.eig()` or `np.linalg.eigh()` might be used. If you use a matrix solver, cut and paste your code as part of your solution. To help with the grading, please normalize your numerical solution by dividing all parameters of $\boldsymbol{H}$ by h_{22}.

Problem 2. (10 points.) Consider the 2-dimensional *affine transformation*, which maps a point (x, y) to a new location (x', y') in the plane. In homogeneous form, the transformation can be represented as

$$
\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \boldsymbol{A} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \text{ where } \boldsymbol{A} = \begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ 0 & 0 & 1 \end{bmatrix}.
$$

a) Show that $\boldsymbol{A}^{-1}$ is also an affine transformation, when the inverse of $\boldsymbol{A}$ exists.

b) Consider the 3 points $(x, y) = (0, 0)$, $(1, 0)$, and $(0, 1)$. For each of these points, write an expression (using scalar values, in terms of the individual a_{ij}) for the corresponding new location (x', y'). (Notice that these 3 points can provide insights about how one coordinate reference frame maps onto another coordinate reference frame. Another take-away is that an affine transformation can map the vertices of *any triangle* to and from the vertices of a reference triangle.)

c) If $\boldsymbol{B}$ and $\boldsymbol{A}$ are both 3×3 affine transformation matrices, show that the product $\boldsymbol{BA}$ also represents a 3×3 affine transformation.

Problem 3. (10 points.) Consider a camera with focal length f that is positioned above a flat table. The camera is aimed vertically downward to observe the table, so that the image plane is parallel to the table. The distance between the table and the camera's point of projection is z_c.

On the table is a short object, such as a nail, that appears as a line segment in the image. Assume ideal perspective projection, and assume that the nail is always entirely within the camera's field of view. Show that the length of the nail *as it appears in the image* does not depend on the nail's position or orientation on the table.

Problem 4. (10 points. For 5554 students, this problem is required. For 4554 students, this problem is optional and can be submitted for extra credit.)

Using the definition of a derivative, show analytically that the following kernel is a good discrete approximation of the second derivative, $\frac{\partial^2 I}{\partial x^2}$.

1	-2	1

Machine Problems.
You have been given a Jupyter notebook file `Homework3_USERNAME.ipynb` and several image files. Replace "`USERNAME`" with your Virginia Tech Username. Then upload these files to Google Drive. Open the `ipynb` file in Google Colab. Follow the instructions that you will find inside the notebook file.

What to hand in: After you have finished, you will have created the following 3 files. Upload these 3 files to Canvas before the deadline. Do not combine them in a single ZIP file.

`Homework3_Problems1-4_USERNAME.pdf` ← Your solutions to problems 1 through 4

`Homework3_Code_USERNAME.zip` ← Your zipped Jupyter notebook file

`Homework3_Notebook_USERNAME.pdf` ← A PDF version of your Colab session
