# [ENPM673] Homework 0

Anton Mitrokhin, Kanishka Ganguly, Cornelia Fermüller

January 30, 2019

## 1 Homework 0

On behalf of the instructor and TAs of ENPM673, we would like to welcome you all to the course this semester with an introductory homework. This homework serves two main purposes: one is to get you familiar with (or recall) the mathematical foundations you will need to have, to successfully navigate this course. The other purpose is to let you get started on some basic programming tools and techniques that you will encounter during the course.

There are two main parts to this homework. The first part covers a number of mathematical topics which you will attempt to solve “manually” (i.e. with pen and paper) and all of which you will encounter later in homeworks and in projects. The second part involves a simple programming assignment using Python and OpenCV and will allow you to setup the basic vision algorithm development pipeline.

All of this material is a prerequisite for this course and, while this homework will not be graded and will not count towards your scores, it is still a prerequisite and you may encounter similar problems on the midterm.

**Please keep in mind, that our course policy would require you not only to show the solutions to the homework but also (as per TA request) detailed step-by-step explanations for those solutions.** Homework 0 is designed to be solved by hand, so any usage of mathematical toolkits is disallowed.

## 2 Some Problems, No Solutions

### 2.1 Transforms

In this set of problems, you will be working with three coordinate systems (*’frames’*): $A$, $B$ and $C$. Please consult Fig. 1 for some clarifications. Note that this figure is not required to solve any of the problems. Please use radians to express all angles.

1. You are given three vectors: $T_1 = (1, 1, -1)$, $r_1 = (1 + \sqrt{2}, 1 + \sqrt{2}, 0)$, $r_2 = (2, 1, -1 - \sqrt{2})$ - all three are given in frame $A$. Write the equation of a plane defined by $r_1$ and $r_2$.

2. Assuming that vector $T_1$ is the translation vector between the centers of frames $A$ and $B$, and $r_1$ and $r_2$ point to some points on the $x$ and $y$ axes of $B$ (respectively), compute all *basis* vectors of $B$ in frame $A$ so that $B$ is an orthonormal basis (refer to Fig. 1).

3. Compute the transformation matrix $(R_1, T_1)$ from frame $A$ to frame $B$. What is the determinant of this matrix?

4. How will the equation of the plane computed in (1) change if written in frame $B$?

5. Frame $C$ is defined by the transformation from $B$ to $C$: $(R_2, T_2)$, where $R_2$ is given by Eq. 1 and $T_2 = (0, 0, 0)$. Vector $m = (-\sqrt{3}, \sqrt{3}, 1)$ is given in frame $C$. Provide the vector $m$ coordinates in frame $A$.

6. Do the same but for $T_2 = (-1, -1, 1)$ and $R_2$ given by Eq. 2.

7. What is the Euler angle representation of the $C$ to $B$ transformation given by Eq. 2 (note the $R_2$ provides 'forward' $B$ to $C$ transformation)?

$$R_2 = \frac{1}{15}\begin{pmatrix} 3\sqrt{(10)} & 3\sqrt{(10)} & 3\sqrt{(5)} \\ 5\sqrt{(3)} & 0 & -5\sqrt{(6)} \\ -2\sqrt{(15)} & 3\sqrt{(15)} & -\sqrt{(30)} \end{pmatrix} \tag{1}$$

$$R_2^{'} = \begin{pmatrix} \frac{\sqrt{3}}{4} & \frac{1}{4} & \frac{\sqrt{3}}{2} \\ \frac{3}{8} - \frac{\sqrt{3}}{4} & \frac{3}{4} + \frac{\sqrt{3}}{8} & -\frac{1}{4} \\ -\frac{1}{4} - \frac{1}{8}(3\sqrt{3}) & -\frac{3}{8} + \frac{\sqrt{3}}{4} & \frac{\sqrt{3}}{4} \end{pmatrix} \tag{2}$$

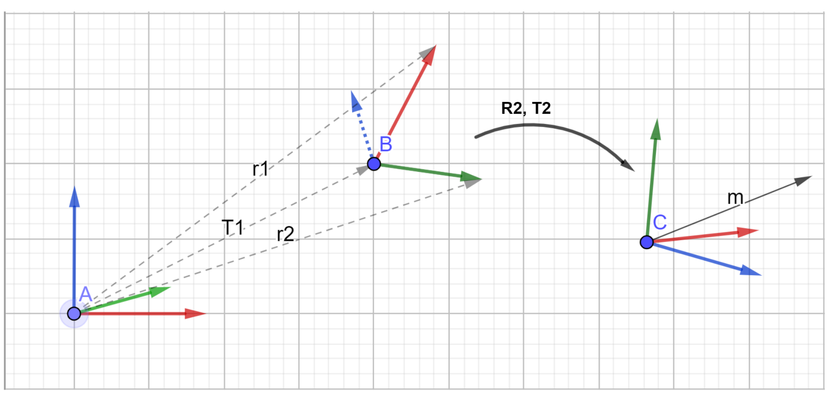


Figure 1: Illustration for problem 2.1. Note that the this image may not exactly correspond to the values of transformations in the assignment.

## 2.2 Least Squares Approximation

In computer vision, least squares approximation is a commonly used, important tool. When dealing with real world data, one often encounters noisy input from sensors (such as cameras), which causes errors in the interpretation of the same data from different sensors. A typical example for this is image stitching. In image stitching, one is given two images, of the same scene, taken slightly farther apart and it is required to "stitch" the images together to get a seamless, larger image combining the two. This is done by first finding out a set of *features* in both images that are similar, such that they may be used as the basis for stitching the images.

In this set of problems, you will attempt to utilize least squares approximation to find a solution to a set of linear equations and find a line that passes through a set of given points.

### 2.2.1 Solving a set of linear equations

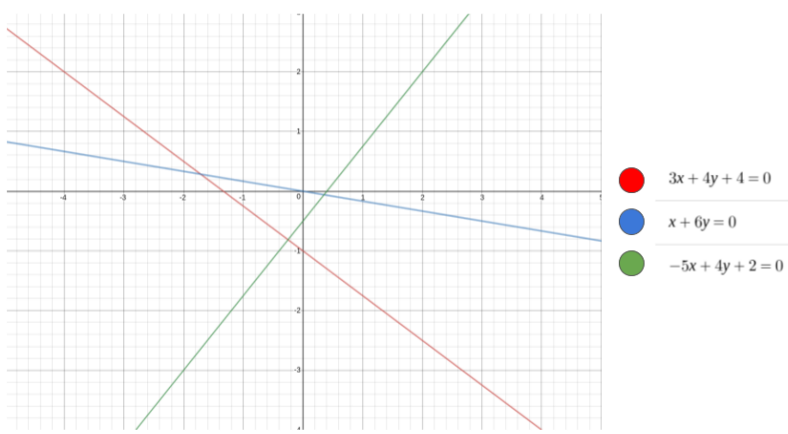


Figure 2: Illustration for problem 2.2.1. Note that this figure may or may not help you solve your question.

You are given the following lines:

1. $3x \; + \; 4y \; + \; 4 = 0$
2. $x + 6y \; = \; 0$
3. $-5x + 4y + 2 = 0$

Now, upon initial computation, you will find that this system of linear equations is unsolvable and hence, an approximate solution is needed.

Compute the approximate solution to this set of equations using the standard least squares method.

### 2.2.2 Finding a line to pass through given points

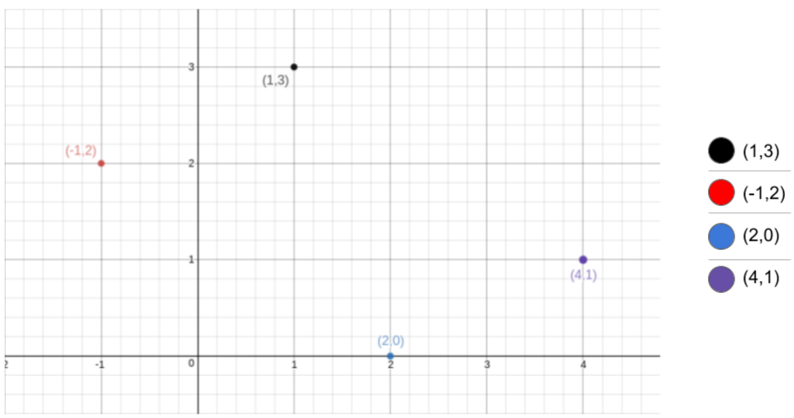


Figure 3: Illustration for problem 2.2.2. Note that this figure may or may not help you solve your question.

You are given the following points:

1. (1,3)
2. (-1,2)
3. (2,0)
4. (4,1)

By looking at the figure, it is clear that there is no single line that can fit all these points simultaneously.

- Compute the equation of the line that using least squares approximation method that best fits these points.
- Provide a plot of this line with respect to the other points provided.

# 3 Candy Counter

As part of getting familiar with Python and OpenCV functions, you will be developing an algorithm for segmenting objects from an image based on color.

(a) Candy Image

(b) Pins Image

Figure 4: The images provided for color segmentation

You are provided two images to test your code on, namely `candy.jpg` and `pins.jpg`.

Your task is to count the number of objects with the same color (i.e. ”five blue pins” or ”two red candies”, etc.) and output it as text overlayed on the original image.

By the end of this project, you should be familiar with (in approximate order):

- *Reading an image* from file, into Python, using OpenCV.
- *Resizing* images.
- Understanding the representation of an *image as matrices of values.*
- Converting an image *to and from different color spaces.*
- Applying processing techniques such as *blur* or *sharpen.*
- *Selecting individual channels* from the image, and manipulating them independently.
- Selectively *“masking”* certain parts of the image based on *color thresholds.*
- *Write text* overlaid on images.

Your output must also show individual “masked” images for each of the colors, which show only the pins under consideration. For example, you must have separate images for red, blue, yellow, etc. which show **ONLY** those colored objects on a black or white background.