Homework 0

Release: 01/09/2023 Due: 01/17/2023, 11:59 PM

- Homework 0 tests your background knowledge on physics, optimization, geometry processing, and network training, which are prerequisites that we will not thoroughly review in this course. You are **allowed** to consult any external resources, but you must cite them. You are also **allowed** to discuss with each other, but you need to acknowledge them. However, your submission must be your own work; specifically, you must **not** share your code or proof.

- We estimate that at most 6 hours are needed to finish it. If not, you might feel that the course is progressing very fast.

- Your submission should be a single PDF file containing solution, code, and results.

- This homework is worth 5/100 of your final grade. However, it is **mandatory**, meaning you have to finish and submit it.

- If you are on the waitlist and intend to enroll, you have to submit this homework on time.

Problem 1. Rigid Motion (6pts)

As shown in Fig. 1, there is a rigid thin square moving on a smooth XY-plane, on which an external force $F = 5N$ is always applied. For the square, the mass is $m = 1kg$, the side length is $d = 2m$, and the density distribution is uniform. For the external force F, the contact point and the direction are fixed relative to the square. Specifically, F is always orthogonal to one side, and the distance between the contact point and the dashed line (through the center and parallel to F) is always $l = 0.5m$. At time $t = 0$, the square is still with the center at the origin, and force F points to the $+y$ direction.

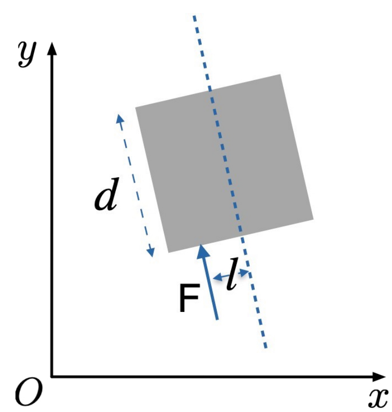

Figure 1: A moving square.

Please analyze the trajectory of the square for $t \in [0, 4s]$:

1. Write down the basic physical laws and and necessary derivations (differential equations). Let θ be the clockwise rotation angle of the moving square.

2. Write a program to solve the numerical integration for the differential equations.

3. Plot three curves: trajectories of the center and the contact point; how the rotation angle θ changes over time t ($\theta = 0$ when $t = 0$, clockwise is positive).

4. Write down the final positions ($t = 4s$) of the center and the contact point.

Hints: 1. There is no friction; 2. For a thin square, the moments of inertia is $\frac{1}{6}md^2$ (axis of rotation at the center).

Problem 2. Optimization (7pts)

We consider the following constrained least square problem:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|Ax-b\|_2^2 \\ \text{subject to} & x^T x \leq \epsilon, \end{array}$$

where $A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m, \epsilon \in \mathbb{R}, \epsilon > 0$. The variable is $x \in \mathbb{R}^n$.
The solution to this optimization problem has to satisfy the Karush–Kuhn–Tucker (KKT) conditions:

$$\begin{cases} \nabla_x \mathscr{L}(x,\lambda) = 0 & (1) \\ x^T x \leq \epsilon & (2) \\ \lambda \geq 0 & (3) \\ \lambda(x^T x - \epsilon) = 0 & (4) \end{cases}$$

where $\mathscr{L}(x,\lambda) = \frac{1}{2}\|Ax-b\|_2^2 + \lambda(x^T x - \epsilon)$ is the Lagrangian of the problem. Let us assume that A is of full rank and the linear system of $Ax = b$ is overdetermined.

1. Write down the gradient of the Lagrangian $\nabla_x \mathscr{L}$. (1pt)

   Next, We will solve this problem by considering 2 cases based on Condition (2).

2. Case 1: $x^T x < \epsilon$. Then, $\lambda = 0$ by Condition (4). To satisfy the KKT condition (1), it is equivalent to solving the unconstrained least square problem with objective $\frac{1}{2}\|Ax-b\|_2^2$. If its solution satisfies (2), we are done. Write down the closed-form solution to the unconstrained least-square problem. You do not need to show intermediate steps. (1pt)

3. Case 2: $x^T x = \epsilon$.

   (a) Set $\nabla_x \mathscr{L} = 0$, express x in terms of λ, i.e. $x = h(\lambda)$ (1pt)

   (b) Prove $h(\lambda)^T h(\lambda)$ is monotonically decreasing for $\lambda \geq 0$. *Hint:* You might need the fact that $A^T A = U\Lambda U^T$ (eigendecomposition). (2pts)

   By the monotone property of $h(\lambda)^T h(\lambda)$, we can solve $x^T x = \epsilon$ by line search over λ (e.g., bisection method or Newton's iterative method). You will implement it in the following programming assignment.

4. (Programming assignment) Solve a provided instance of this problem. (2pts)

   - You can load the data by

```
npz = np.load('./HW0_P1.npz')
A = npz['A']
b = npz['b']
eps = npz['eps']
```

- You are **not** allowed to use external optimization libraries that solve this problem directly. Linear algebra functions in numpy are allowed, e.g. numpy.linalg.eig, numpy.linalg.svd, etc.
- Please provide screenshots or pasted code in your submitted pdf together with your answer.

Problem 3. Geometry Processing & Network Training (5pts)

In this problem, you need to train a neural network to solve the 3D shape part segmentation task. Specifically, we focus on the ShapeNet chair category, and there are only four types of parts: arm, back, leg, and seat.

We provide 1,000 annotated shapes as training data. You can find input point clouds in `train/pts` and ground truth labels in `train/label`. For point cloud files, each line denotes the 3D coordinate of a point, and the number of points may vary. For label files, each line denotes the part annotation of the corresponding point, where 1-4 indicates arm, back, leg, and seat, respectively. Fig. 2 shows some examples of the training data.

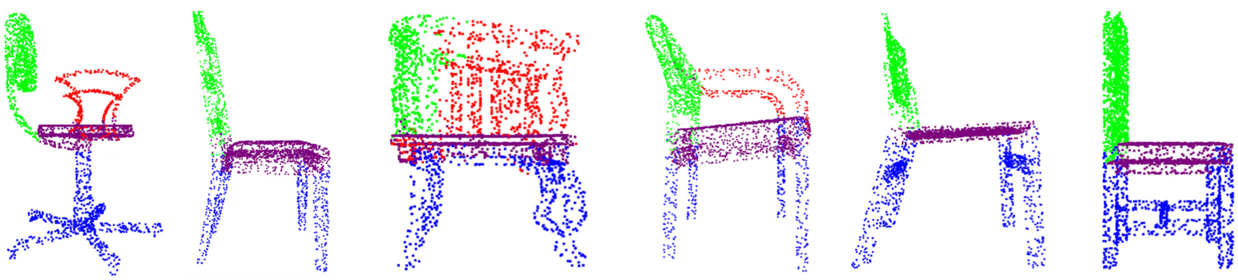

Figure 2: Training data: point clouds with gorund truth label.

After training, you need to test your model on six testing point clouds provided in `test/` and visualize the results in your report. You can use any 3D library (e.g., Open3D) to visualize the point clouds. Please color arm parts in red, back parts in green, leg parts in blue, and seat parts in purple.

You can leverage any existing 3D neural network (e.g., PointNet) as your network backbone. Since the segmentation task is relatively easy (only one category and four parts), you don't need very complex networks (a small PointNet should be good enough), and the network training should converge very fast. You can use quantitative metrics (e.g., mIoU) to check your implementation. However, your results will only be graded based on visual appearance, and you do not need to achieve very high performance. Please **visualize** the first six shapes in the test set with your prediction (colored parts) and also provide code in the pdf.

Hints:

1. If you do not have a GPU, you can try Google Colab. Login with your .edu Google account for free GPU resources.
2. You may need to pad the input point cloud to enable batch processing.
3. You can further split the training data to set up a validation set.

Problem 4. Feedback Questions (1 extra point)

Your feedback is important to us to improve the homework. You will earn one extra point by answering the following questions:

1. How long have you taken to finish the above problems?
2. Which problem is the most challenging for you?