CSCI-UA 473 : Intro to Machine Learning - Final Project Description

October 26, 2022

1 Problem Statement

In this project, you will be working on state prediction of a 4-fingered robot hand given RGBD (RGB + Depth) images. You are required to implement a simple supervised learning algorithm where you **input RGBD images of the robotic hand from a top view, and output the positions (in meters) of the tip of each finger**. You are required to submit your code to the Kaggle Competition (http://www.kaggle.com/competitions/csci-ua-473-intro-to-machine-learning-fall-2022) and will be graded/ranked according to regression loss values of the output of your model. The evaluation metric for the competition will be the mean squared error (MSE) loss.

The dataset has already been collected for you and is uploaded on the given Kaggle competition. You should download the dataset and use it according to the instructions provided in Sec. 1.1.

Your code must include two major components:

- Dataset class: Where you load the data, apply necessary augmentations and return a `Tensor` at each iteration of your training procedure.
- A Convolutional Neural Network (CNN) which takes the RGBD image(s) as an input and returns the positions of each fingertip. This model must be trained in a supervised manner. You have complete freedom in choosing the architecture of the model, the training metric, and the optimizer.

1.1 Data Description

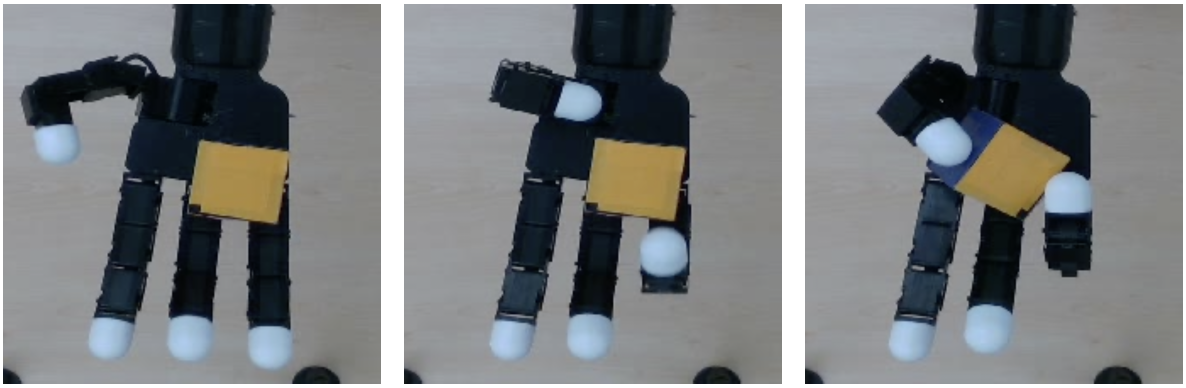

Figure 1: Example RGB images from the dataset

Figure 1 shows example RGB images from the dataset. The dataset has been provided in the form of 3 files - `trainX.pt`, `trainY.pt` and `testX.pt`. `trainX.pt` corresponds to the train images, `trainY.pt` corresponds to the hand states corresponding to each image, and `testX.pt` corresponds to the test images (used for evaluations). The dataset has been provided in the following format -

- `trainX.pt` has images for training your model. The data comprises $(rgb\_images, depth\_images, file\_ids)$. rgb_images have dimensions $(num\_data\_samples, num\_camera\_views, num\_channels, height, width)$. $depth\_images$ have dimensions $(num\_data\_samples, num\_camera\_views, height, width)$. $file\_ids$ contains the sample ID for each data sample.
- `trainY.pt` contains the robot states (outputs) for training your model. The data is present as a tensor of shape $(num\_data\_samples, 12)$ where 12 corresponds to the (x, y, z) coordinates of the four fingertips.
- `testX.pt` has a similar format to `trainX.pt`.

1.2 Instructions for using the data

- The dataset comprises RGB images and depth images from 3 camera angles. You are free to use as many or as few of the camera views provided for training your model.
- Consider normalizing the data when training your model. **Do not use the test data to compute the normalization metrics.**
- Consider shuffling the dataset when loading it using a dataloader.
- Consider applying augmentations to the input images during training.

2 Submission Instruction

Detailed instructions about making a submission on Kaggle will be provided in a few days. You are recommended to go through the problem statement and and get started with the project till then.