**CS 5335: Robotic Science and Systems**

**HW6 report**

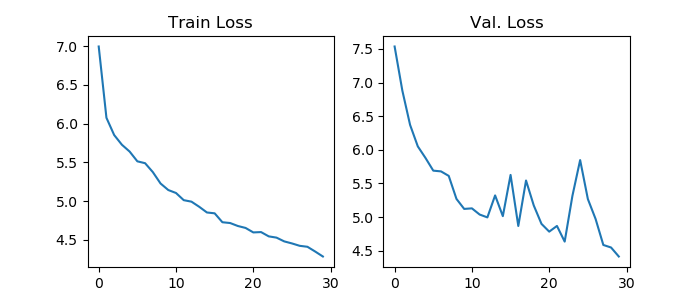
**Programming Partners: Aditya Bondada, Mani Chandhan Chakinala**

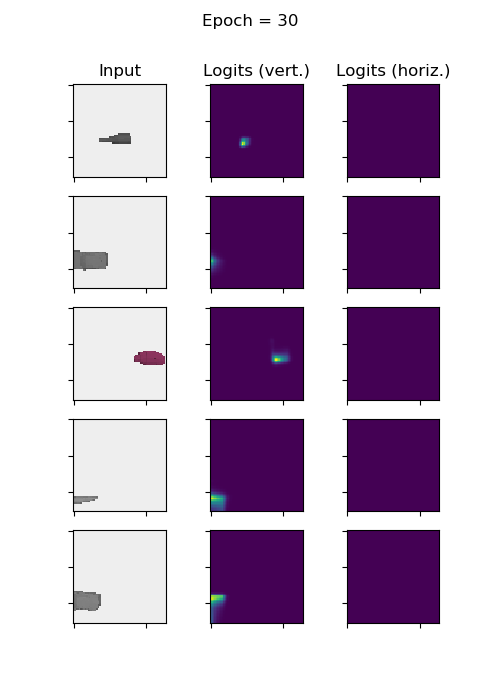
**Q1.a)** PyBullet simulator offers speed and safety benefits for generating grasp attempt datasets. It enables faster experimentation, rapid prototyping, and reduced training time. Additionally, it provides a risk-free environment, protects human operators, and facilitates debugging and failure analysis. Overall, PyBullet contributes to more efficient development and deployment of effective grasping algorithms in real-world applications.

**Q1.b)** I noticed that the objects are being pressed inside the floor before being picked up by the robot. This is quite unrealistic.

**Q2.b)** Both local and global information contribute to predicting grasp success. Local information, such as object geometry, texture, orientation, and contact points, is crucial for determining the robot's interaction with objects. Global information, although not strictly required for simple grasping tasks, becomes relevant in more complex tasks like pick-and-place, where a broader understanding of the environment and object interactions is needed. Overall, local information is key for grasp success, while global information helps in tasks involving multiple objects or a wider environment.

**Q3.a)**

****

****

The fluctuation in the validation loss curve can be mainly attributed to two factors:

1.The limited size of the validation set, consisting of just 200 examples, may not accurately represent the underlying distribution, leading to significant fluctuations in validation loss.

2.A small batch size of 12 makes the model's performance more susceptible to noise within each batch, resulting in oscillations in the loss and hindering generalization across the entire validation dataset.

**Q3.b)** Validation loss is an important metric for evaluating neural network performance, but it doesn't perfectly indicate the network's ability to predict grasps. Several factors contribute to this limitation, such as false negatives, which occur when the network fails to predict a valid grasp location even though one exists, leading to a higher validation loss. Imbalanced datasets, with unequal distributions of valid and invalid grasp examples, can also cause the validation loss to inaccurately represent the network's performance on a balanced dataset. To better understand a network's grasp prediction performance, it's essential to consider other evaluation metrics like precision, recall, F1-score, and success rate, in addition to the validation loss. Analyzing these metrics collectively can provide a more accurate assessment of the network's grasp prediction capabilities. Issues like overfitting, where the network performs well on training data but doesn't generalize to validation data, can be addressed by employing regularization or other techniques to improve generalization.

**Q3.c)**

****

I got a success rate of 40% and yes, I observed that the objects are not only positioned along the axis but also exhibit random rotations.

**Q4.b)**

****

I got a success rate of 64%

**Q4.c)** To enhance generalization for object translations in a scene, I would apply these data augmentation methods:

Image translation: Shift the image horizontally or vertically by a set number of pixels, helping the model recognize objects at varying locations. Ensure consistent translation for both RGB and depth images if used in the network.

Gripper translation: In tandem with image translation, modify the gripper's position (x and y coordinates) in the 2D plane to preserve its relative position to the object, maintaining accurate targeting of grasp points after translation.

Random cropping and padding: Crop a section of the original image randomly and pad the cropped area with a constant value (e.g., background color or depth value). This exposes the model to diverse object translations and aids in generalizing to different positions.

Object-wise translation: Translate each object in the scene independently to distinct positions, helping the model learn to recognize and manipulate objects in various spatial arrangements and improving generalization in translated object scenarios.

Implementing these data augmentation techniques improves the model's ability to generalize to object translations in the scene, making it more robust and adaptable in real-world situations.