Signed-Distance Collision Detection using a Predictive Net: Mid-Term Report

Cade Parkison, Braxton Smith
March 7, 2019

***Abstract*—We propose a CNN to compute the signed-distance to the nearest object given RGBD point-cloud data and joint configuration of a KUKA robot. The system is trained using simulated data from the LL4MA's Gazebo/DART environment. Preliminary research shows the promise for significant improvements in compute time as compared to state of the art collision detection libraries that use the inverse calculations.**

***Index Terms*—CNN, Signed-Distance Collision Function**

I. Introduction

COMPUTING the signed-distance to objects requires numerous calculations and presents itself as a bottleneck for practical implementation in real-world systems. There is need in robotics for an efficient alternative.

II. Prior Work

Traditional collision detection in robotics typically involves computing the forward dynamics or kinematics to determine where a robot is located. These queries can be computationally expensive and limit the speed at which robotic systems can plan in real time.

Other approaches to real-time collision detection use hierarchical data structures to speed up the detection time. These typically use Bounding Volume Hierarchies to represent the environment objects in ways that make the collision computations more efficient but error on the side of caution and eventually converge to the traditional method above (when the accuracy is needed). While this does speed things up, the calculations involved are still expensive.

More recent work has used machine learning techniques to formulate the question posed to various "prediction nets" in terms of collision detection and avoidance. Many try to inform the motion planning algorithm of areas where collision is less probable. Examples include [1], [2], [3].

Hybrid's of the two exist in which a neural net is used to speed up collision checking. "Fastron" [4], developed by Nikhil Das, Naman Gupta and Michael Yip, uses kernel perceptrons (a non-linear variant of the perceptron) to represent the systems configuration space and an efficient search for collision status changes in a time-variant environment. While shown to be much better than traditional method, significant problem formulation is required for this approach.

III. Proposed Model

To take advantage of the fact that both authors have access to a NVIDIA 1080 GPU, we will compare two different CNN architectures for inference of the signed-distance function. The first structure will be based off of [5] while the second which we have yet to come up with will be of our own creation. We believe structure won't be a significant factor but are curious to see. We believe the significant factor will be in the training data.

IV. Proposed Data Collection

Data collection will be done using simulations in a Gazebo/DART environment of a KUKA robot. Sensors mounted on the end-effector and above the environment will generate a depth map of the surrounding environment. Encoders in the joints of the robot will measure the joint angles of the motors. The depth maps will first be combined and reduced via an autoencoder before being passed along with the joint angles to the CNN for signed-distance predicition.

This environment consists of a robot arm mounted on a table with objects placed in the robot's workspace. To achieve good generalization in our models, we will simulate many random distributions of objects in the robot workspace. These objects include spheres, cubes, cones, and other basic shapes of various sizes. The number of placed objects will vary from one object up to ten objects. With the above limitations, we will create 20 distinct environments. For each of these environments, many different robot joint configurations are needed to thoroughly test for robot-environment collisions. These include both "elbow up" and elbow down" configurations for the same end effector positions. Additional environments with "canyon walls", holes and other difficult obstacles will be considered as well.

The LL4MA's Gazebo/DART environment has a built-in kinematic based collision checker, which reports the signed distance to the nearest object, will be used to collect the ground truth data. We believe our trained model will be able to perform this calculation much faster, since the expensive forward kinematic calculations are not needed.

V. Conclusion

Using the proposed outline and with a large volume of data, we believe we can train a CNN model to accurately compute the signed distance of the KUKA arm and it's environments. With enough diverse data and accurate simulation models, this should be able to generalize well to various robot arm environments in the same space. The improvements in computation time of the trained model will help in the motion planning of robot systems in real time.

REFERENCES

[1] J.H. Graham, et. al, *A neural network approach for safety and collision avoidance in robotic systems*. University of Lousiville, 1996.
[2] P. Long, et. al, *Deep-Learned Collision Avoidance Policy for Distributed Multi-Agent Navigation*. IEEE Robotics and Automation Letters, 2016
[3] A. Sharkawy, et. al, *Human-Robot Collision Detection Based on Neural Networks*. 2018
[4] N. Das, et. al, *Fastron: An Online Learning-Based Model and Active Learning Strategy for Proxy Collision Detection*, University of California, 2017.
[5] Q. Lu et. al., *Planning Multi-Fingered Grasps as Probabilistic Inference in a Learning Deep Network*. University of Utah, 2017.