Lifelong 3D Object Recognition and Grasp Synthesis using Dual Memory Recurrent Self-Organization Networks

Krishnakumar Santhakumar[a], Hamidreza Kasaei[a]

[a]*Department of Artificial Intelligence, University of Groningen, Groningen, 9747 AG , Groningen, Netherlands*

1. Dataset Generation

The continual learning model needs to be trained on the sequential dataset, where the spatio-temporal relations of the task are progressively available over time (e.g dataset comprises of videos). Real-world objects are three-dimensional, the dataset like Cornell [1], Jacquard [2], ModelNet [3] and ShapeNet [4] which comprises of 3D objects of different categories are used for object recognition, and grasp synthesis prediction. These dataset provides the static representation of the objects. To address lifelong learning, we need a dataset which sequential represents the input data distribution. CORe50 dataset [5] provides the sequential representation of the real world object in terms of video captured scenes to address the continuous object recognition problem. Since the dataset is made of 2D images it cannot be used for grasp synthesis prediction. So, we developed our custom dataset which represents the sequential point cloud of the 3D objects in the form of different collections to address both object recognition and grasp prediction. In which each collection consists of different 3D object instances and categories, similar to the representation of the scenes in the CORe50 dataset.

Table 1: Objects used in the dataset generation process.

Cooking pan	Hammer
Airplane	Toy pistol
Guitar	Clock
Bowl	Pencil
Bottle - wine	Staple

We developed the dataset using the robot operating system (ROS) platform and 3D objects sampled from the ShapeNet dataset [4] and gazebo environment. In our dataset generation process, we used 50 objects belongs to 10 categories in the representation of 15 collections. In which each object instance in the collections consists of 500 point cloud samples of the respective object in a sequential manner. Table 1 comprises the list of objects used in the dataset generation. Since we used only 50 objects to create 15 collections, the object instances from one collection are used to generate the dataset samples for another collection. For example, object from collection 1 is used in collection 6 and collection 11, objects from collection 2 is used in collection 7 and 12 and

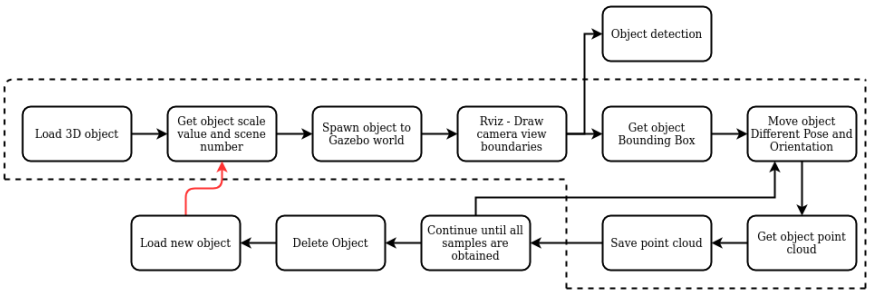
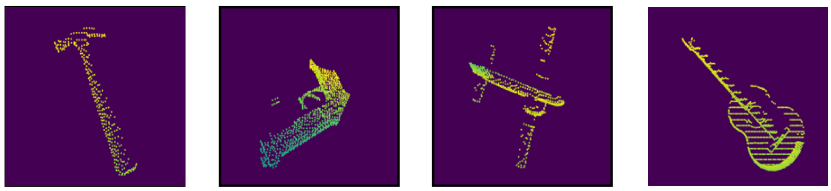


Figure 1: Architecture pipeline for the dataset generation.

follows. Even though we used the objects from one collection to another we make sure that object samples differ from one collection to another collection. To create more diversity between the object samples in each collection and to make the model prediction robust to external noise, we augmented the dataset. The dataset augmentation includes changing the position (x, y, and z-axis) and orientation (roll, pitch, and yaw) of the object, adding Gaussian noise to the point cloud during dataset generation, down-sampling the point cloud data, adding random occlusion, and by adding Gaussian noise along with point cloud down-sampling.



(a) Hammer (b) Pistol (c) Airplane (d) Guitar

Figure 2: RGB-D representation of the point cloud data samples generated during the dataset generation process.

The dataset generation architecture pipeline is shown in figure 1. The system initially spawns the 3D objects to the gazebo world using the ROS spawn model service based on the object scale value and the collection number. With our dataset generation algorithm, the object of any size can be used in the dataset generation by the custom scaling feature to fit the object to the gazebo environment. The scale values are determined prior to the dataset generation by experimenting with the different scale values to fit the 3D object into the gazebo world. Then the object region of interest (ROI) and its associated bounding box is detected. After which the object is moved to a different position and orientation (at the beginning the position and orientation will be zero). For the current position and orientation of the given 3D object, the point cloud data is generated. The point cloud with binary or RGB information can be generated based on the user requirement during the point cloud generation process. In the process along with the change in position and orientation, the above-mentioned augmented techniques are

introduced. The data sampling continues until the maximum frames (here 500 frames) are obtained. Once the required point cloud samples are generated and saved, the current object will be removed from the gazebo world and a new object will be spawned. This process continues until all the object instances of 10 categories in all the collections are generated. Figure 2 shows the examples of point cloud data for the hammer, pistol, airplane, and guitar object.

1.1. Parameter selection

The parameters and its range for different augmentation techniques used in dataset generation are as follows: in gazebo environment the position of the x-axis ranges from $0.176m$ to $-0.7m$, the position of the y-axis ranges from $0.072m$ to $-0.2475m$, and the position of the z-axis is set to a constant value of $0.003076m$. The position of the z-axis value is set to constant to prevent the object elevation from the wooden floor. For a change of orientation, the roll value ranges from 0 degree to 360 degree with the offset of 90 degree, the pitch value ranges from 0 degree to 360 degree with the offset of 60 degree, and the yaw rotation value ranges from 0 degree to 720 degree with the offset of 30 degree. For the Gaussian noise, the mean (μ) value changed between 0.01 to 0.51 with the difference of 0.01 and standard deviation (σ) values changed between 0.005 to 0.011 with the difference between values of .001. Both μ and σ values ranges are determined based on experimentation and both of these values are randomly sampled during the dataset generation process. For down-sampling, the point cloud data obtained from the kinect camera are down-sampled based on voxel size ranges from 0.01 to 0.1. The point samples are down-sampled from 1% to 10% of the total size of the respective objects point cloud. The percentage ranges are determined to ensure that even though the point cloud data points are down-sampled, the overall structure of objects needs to be retained. We inducted this constrain to have the point cloud data with a considerable amount of points to extract the features for object recognition and grasping.

References

[1] Y. Jiang, S. Moseson, A. Saxena, Efficient grasping from rgbd images: Learning using a new rectangle representation, in: 2011 IEEE International Conference on Robotics and Automation, 2011, pp. 3304–3311. doi:10.1109/ICRA.2011.5980145.

[2] A. Depierre, E. Dellandréa, L. Chen, Jacquard: A large scale dataset for robotic grasp detection, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018, pp. 3511–3516. doi:10.1109/IROS.2018.8593950.

[3] M. A. Uy, Q.-H. Pham, B.-S. Hua, T. Nguyen, S.-K. Yeung, Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019.

[4] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, et al., Shapenet: An information-rich 3d model repository, arXiv preprint arXiv:1512.03012 (2015).

[5] V. Lomonaco, D. Maltoni, Core50: a new dataset and benchmark for continuous object recognition, in: S. Levine, V. Vanhoucke, K. Goldberg (Eds.), Proceedings of the 1st Annual Conference on Robot Learning, Vol. 78 of Proceedings of Machine Learning Research, PMLR, 2017, pp. 17–26.
URL http://proceedings.mlr.press/v78/lomonaco17a.html