2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
September 27 - October 1, 2021. Prague, Czech Republic

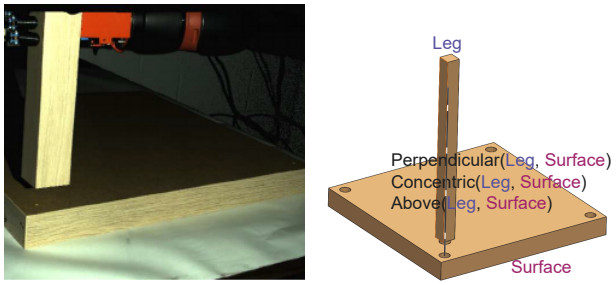
2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 978-1-6654-1714-3/21/$31.00 ©2021 IEEE | DOI: 10.1109/IROS51168.2021.9635939

Comprehension of Spatial Constraints by Neural Logic Learning from a Single RGB-D Scan

Fujian Yan, Dali Wang, and Hongsheng He*

***Abstract*—Autonomous industrial assembly relies on the precise measurement of spatial constraints as designed by computer-aided design (CAD) software such as SolidWorks. This paper proposes a framework for an intelligent industrial robot to understand the spatial constraints for model assembly. An extended generative adversary network (GAN) with a 3D long short-term memory (LSTM) network was designed to composite 3D point clouds from a single RGB-D scan. The spatial constraints of the segmented point clouds are identified by a neural-logic network that incorporates general knowledge of spatial constraints in terms of first-order logic. The model was designed to comprehend a complete set of spatial constraints that are consistent with industrial CAD software, including left, right, above, below, front, behind, parallel, perpendicular, concentric, and coincident relations. The accuracy of 3D model composition and spatial constraint identification was evaluated by the RGB-D scans and 3D models in the ABC dataset. The proposed model achieved 57.23% intersection over union (IoU) in 3D model composition, and over 99% in comprehending all spatial constraints.**

***Index Terms*—spatial constraints, neural-logic learning, logic rules**



(a) A sample RGB-D scan. (b) Comprehended spatial constraints.

Figure 1: Comprehended spatial constraints between parts of an IKEA table during autonomous robotic assembly. The left image shows the RGB-D scan of a table leg and a table surface, and the right image shows the spatial constraints comprehended by the robot between the leg and the surface.

I. Introduction

With the recent advancement of machine-learning and sensing technologies, there are increasing needs on automated planning of industrial robots on assembly lines recently [1], [2]. Traditional industrial robots are typically programmed to be deployed in well-constructed assembly lines, where they can assemble and operate precisely with familiar objects [3]. It may, however, take significant manual effort to reprogram robots to work with new objects or work in a new environment [4]. As compared to confined, immobile, and non-interactive industrial robots in structured environments [5], autonomous robotic systems would need to understand the spatial constraints on parts to achieve automated assembly under unfamiliar unstructured situations [6]. Autonomous robotic systems have potential to be agile, flexible, adaptive, and flexibility without explicit programming [7].

In this paper, we propose a method that enables industrial robots to understand spatial constraints by neural logic learning. According to CAD software such as SolidWorks, spatial constraints are useful in assembly. The proposed model can learn the complete set of spatial constraints defined in CAD software, which are left, right, above, below, front, behind, parallel, perpendicular, concentric, and coincident. Compared with our previous work [8], the new model can learn the set of spatial constraints by incorporating eight logic rules into the model structure. In this paper, a 3D composition model is designed to approximate point clouds of 3D models from an RGB-D scan. In contrast to approximated bounding volumes of objects [8], the composited 3D shapes are represented in point clouds that allow precise identification of spatial constraints. A result of spatial constraint understanding by using the proposed approach is illustrated in Fig. 1, where the left one is the RGB-D scan of the scene, and the right one is the robot comprehended spatial constraints.

Following [9], we propose a 3D composition model by extending the generative adversarial network (GAN) with 3D convolutional Long Short-Term Memory (LSTM) units to fulfill scanned object shapes. We further register the 3D composited shapes of objects with RGB-D scans of objects to prevent missing geometrical information such as distance, coordination. After the 3D shapes of objects are composited, a neural-logic network is designed to identify objects' spatial constraints based on the high-level features that are extracted by a grounding block. Spatial constraints are symbolic, and it is challenging to use raw point-cloud

Fujian Yan and Honegsheng He are with School of Computing, Wichita State University, Wichita, KS, 67260, USA.

Dali Wang is a Senior R&D Staff and a member of the Artificial Intelligence (AI) team at Oak Ridge National Laboratory (ORNL).

This work was supported by NSF 2129113 and Regional Institute on Aging R52110.

*Correspondence should be addressed to Hongsheng He, hongsheng.he@wichita.edu.

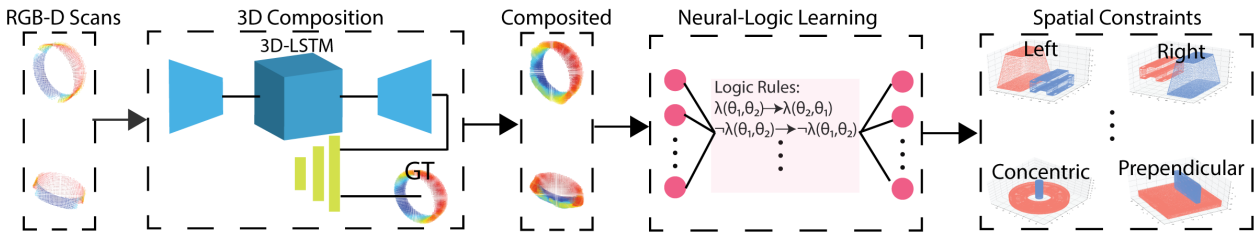


Figure 2: Structure of the Neural-Logic Network. It contains three part, which are 3D filling block, grounding block, and symbolic learning block.

data to represent. Integrating knowledge with data-driven methods such as neural network can help learn and represent knowledge. The proposed neural-logic network can learn the spatial constraints of objects represented by point clouds by using pre-defined logic rules. The rich knowledge that is presented by logic rules can improve learning performance. The logic operators such as "AND" and "OR" are mapping to numerical space by using fuzzy logic.

The major contributions include: 1) We proposed a 3D composition model extends the GAN with the 3D LSTM unit to composite the point clouds of objects from a single RGB-D input. 2) We designed a neural-logic learning model that can learn comprehensive spatial constraints using raw point cloud data. This paper focuses on the problem of understanding spatial constraints on constructed models from an RGB-D scan, while assuming that the point clouds corresponding to the models are segmented.

II. Spatial Constraints Understanding

The proposed model of spatial constraints understanding contains two parts: the first part is a 3D composition network, and the second part is a neural logic network, as shown in Fig. 2. The 3D composition model takes a single RGB-D scan of the objects to generate approximated 3D shapes of objects, and the neural logic learning network takes a pair of objects in as input to determine their spatial constraints.

A. 3D Composition Network

Completed point cloud of objects, which contains volumetric information, is essential for understanding spatial constraints. A single scan from an RGB-D sensor can only acquire the depth of objects from one point-of-view. Therefore, 3D volumetric information is missing. As shown in Fig. 3, there is missing volumetric information in RGB-D scan of the object. To address that problem, we propose a 3D model composition network with an extended GAN network to compose the missing point cloud of the object. Conventional GAN networks randomly generates data from latent distributions. In contrast, the proposed GAN model used RGB-D scans as input. The proposed GAN network formulates a smooth function f to map the 2.5D input x to the 3D composed model $y = f(x)$ where $x \in \mathbb{Z}^{64\times64\times64}$ and $\mathbb{Z} = \{0, 1\}$. The 2.5D input is presented in a 64^3 occupied grid. The 3D composited model is represented in the format of a 256^3 occupied voxel grid.

To minimize the difference between the 3D composited model y and the ground truth y', multiple-viewed RGB-D scans of a same object can be used as a sequence of data $x_0 \dots x_n$ to train the model to increase the fineness of the 3D composited model. To implement this idea, we extended the GAN network with 3D LSTM units [10]. To keep the original coordinations of objects in the scene, we register the composed 3D model to the RGB-D scan of objects with ICP [11].

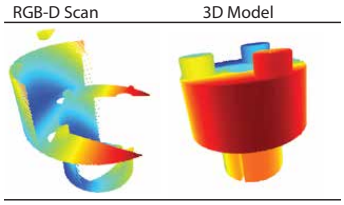


Figure 3: Comparison between a RGB-D scan and a complete point cloud of the same object in the view.

The 3D model composition network contains a generator and discriminator. The generator can produce a composited 3D shape of an object, and the discriminator can differentiate how realistic is the composited 3D shape against the 3D CAD model (ground truth) of an object [12]. The generator network takes a single RGB-D scan of the object $x \in \mathbb{Z}^{64\times64\times64}$ as the input, then create the composited 3D model $y \in \mathbb{Z}^{256\times256\times256}$ of the object. We introduced an encoder-decoder architecture into the GAN network to extract the features from RGB-D scan. The structure of the generator part of the 3D composition model is shown in Fig. 4.

Both the composited 3D model and the real CAD model are embedded as inputs for the discriminator. The composited 3D model is mapped into a 64^3 voxel grid, and the 3D model is mapped into a 256^3 voxel grid. There are six 3D convolutional layers in the discriminator, and each of the first five 3D convolutional layer has a ReLU activation function, and the last layer takes the sigmoid function as the activation function. The learning rate for the generator network is $1e^{-4}$, and that of the discriminator network is $5e^{-5}$. Generator and discriminator were optimized by the Adam optimizer. There are four different categories, which contain objects that are

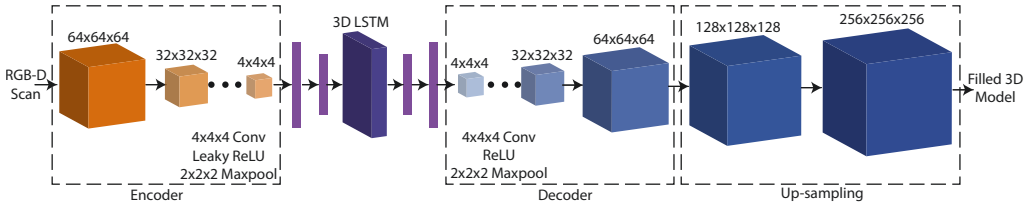


Figure 4: Structure of the generator network.

most commonly used in assembly, are selected to train the 3D model composition network. Each category has four objects, and another two objects in the category is selected to test, and another two objects are selected for validation. The RGB-D scans were generated by Blender, where each object was scanned 125 times with 2.88° angle difference along Z-axis. The field of view (FOV) for the simulated RGB-D camera is 49.2°, and RGB-D camera is placed 1.6 meters away from the object.

B. Neural Logic Network

We assumed the point clouds are perfectly clustered. Rather than manually designing features from 3D models, this paper seeks to learn a high-level representation from the training data. The training time can be reduced by using these high-level features. A five-layered fully-connected neural network is proposed in this paper to extract features from the raw point-cloud data. The input of the grounding block is from the 3D filling block's output. The generated 3D filling model is down-sampled and the sample factors are 600. There are 600 neurons in the first layer for the pairwise input, The ReLU were the activation functions for each layer. The mean and standard deviation of each layer are initialized using a random normalization method. The mean equals to zero, and the standard deviation equals to one.

The logic rules of spatial constraints are learned by using neural network with the pair-wise point clouds of objects in the scene. We designed a fully connected feed-forward neural network, which has four layers to enable robots to understand spatial constraints. We take extracted features from the ground block as input, and we instantiate the features that are extracted from the 3D fulfilled models into variables in the spatial rules to translate spatial rules to numerical space [13]

$$\mathrm{I(p)} = \sigma(\mu_{\mathrm{P}}^{\mathrm{T}} \tanh(\mathrm{x}^{\mathrm{T}} \mathrm{W}_{\mathrm{P}}^{[1:\mathrm{k}]} \mathrm{x})) \tag{1}$$

where $W_P^{[1:k]}$ is the weight of the network, and it is a tensor in $\mathbb{R}^{\mathrm{mn}\times\mathrm{mn}\times\mathrm{k}}$, σ is the sigmoid function. $I(p)$ is the output of one network. The x is the extracted features from the grounding block. The μ was t-norm the conjunction of the neural network. This encoding enables a network to determine the grounding of a clause (e.g. $\mathrm{above}(\theta_1, \theta_2) \rightarrow \mathrm{below}(\theta_2, \theta_1)$) by calculating the literals (e.g. $\mathrm{above}(\theta_1, \theta_2)$) of the clause and then combine those results to calculate the final result. Two networks are combined to map whole spatial rules to numerical space.

The proposed method can learn the logic rules. The spatial constraints include left (L), right (R), above (A), below (B), front (F), behind (Bh), parallel (Para), perpendicular (Perp), concentric (Conc), and coincident (Coin) The rules which are regulated for logic learning are based on these logic rules:

$$\forall\theta_1, \theta_2 : \dot{\lambda}(\theta_1, \theta_2) \rightarrow \dot{\lambda}(\theta_1, \theta_2) \tag{2}$$

$$\forall\theta_1, \theta_2 : \neg\dot{\lambda}(\theta_1, \theta_2) \rightarrow \neg\dot{\lambda}(\theta_1, \theta_2) \tag{3}$$

$$\forall\theta_1, \theta_2 : \lambda(\theta_1, \theta_2) \rightarrow \quad \bar{\lambda} \ (\theta_2, \theta_1) \tag{4}$$

$$\forall\theta_1, \theta_2 : \lambda(\theta_1, \theta_2) \rightarrow \neg\lambda(\theta_2, \theta_1) \tag{5}$$

where θ_1, θ_2 denote objects. Both $\dot{\lambda}$ and λ denote spatial constraints, $\dot{\lambda} \in \{\mathrm{L}, \mathrm{R}, \mathrm{A}, \mathrm{B}, \mathrm{F}, \mathrm{Bh}, \mathrm{Para}, \mathrm{Perp}, \mathrm{Conc}, \mathrm{Coin}\}$ and $\lambda \in \{\mathrm{L}, \mathrm{R}, \mathrm{A}, \mathrm{B}, \mathrm{F}, \mathrm{Bh}\}$.

- ◇ The rule (2) denotes that if the spatial constraints between two objects θ_1 and θ_2 is λ, then the spatial constraints between two objects θ_1 and θ_2 is λ.
- ◇ The rule (3) denotes that if the spatial constraints between two objects θ_1 and θ_2 are not λ, then it infers the spatial constraints of these two objects are not λ. In rule (4), the $\{\lambda, \overline{\lambda}\}$ is contrasting spatial constraints, such as {left, right}, {front, behind}, {above, below}, and {contact, non-contact}.
- ◇ The rule (4) denotes if the relation between two objects θ_1 and θ_2 is known, then the object θ_2 and the object θ_2 has the contrasting spatial relation such as left and right.
- ◇ The rule (5) denotes that if the spatial constraints between the object θ_1 and the object θ_2 is known, then the constraints between the object θ_2 and the object θ_1 cannot be the same.

These logic rules described as spatial constraints such as "left", "right", "above", "below", "front", and "behind" for the neural-logic learning network. For the spatial constraints such as "parallel", "perpendicular", "concentric", and "coincident" both the positive rule (2) and the negative rule (3) are applied. The concatenation of the rules $C = \sum_i^n I(\lambda)$ where $I(\lambda)$ is the instantiation of each rule regarding to ten different spatial constraints. The optimization for the learning is to minimize the loss $L = \underset{C}{\mathrm{argmin}}(\sum_j^m C)$, where C is the concatenation of each logic rule, and m is number of logic rules. The minimum of the loss L has been optimized with an adaptive gradient optimizer with a starting learning rate equal to $1e^{-3}$. The weights of the spatial logic network are randomly initiated with the mean equals to zero, and

the standard deviation equals to one. There are totally four layers in spatial logic network block, which the first three layers have 600 neurons and choose the Tanh function as activation function. The last layer has eight neurons and the activation function is the sigmoid function. The spatial logic block is a parallel structure, which each one translate logic atom into numerical space. The fuzzy semantic logic conjunct each atom in the logic sentence to represent the semantic of each logic sentence.

III. Experiment

We performed simulation and experiments to evaluate the proposed method including the accuracy of the spatial constraint understanding, and the agreement of the composited 3D models in each category with the ground truth.

A. Accuracy of 3D Composition Model

Some of the sample results of 3D composition based on RGB-D inputs were shown in Fig. 5. As compared with the ground truth (CAD model) of each object, the composited 3D model can represent the 3D shape of objects. By comparing the third row and the fourth row, the more the shape of objects the RGB-D scan have covered, the better the 3D composition of the objects was.

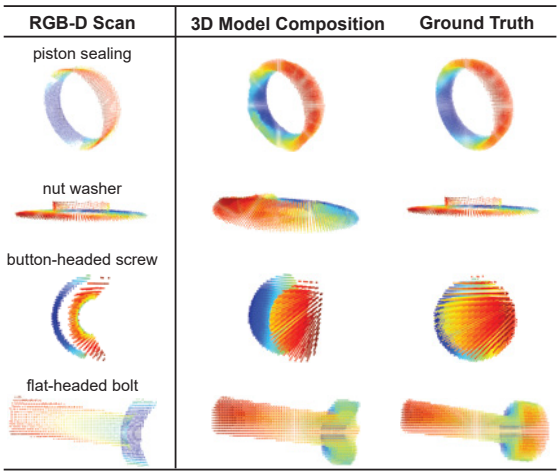


Figure 5: Results of the 3D composition model. The first column is the RGB-D scan of objects, the second column is the composited 3D model of objects, and the last column is the ground truth of objects.

We evaluated our 3D fulfilling results by computing the intersection over union (IoU), the composition error, and the missing voxel. The IoU was calculated between the 3D fulfilling models that were generated based on RGB-D scans and the CAD models of objects. IoUs are computed as $IoU = (\sum_{i=0}^{N} 1_e(y_i) \cap \hat{y}_i)/(\sum_{i=0}^{N} 1_e(y_i) \cup \hat{y}_i)$ where y_i was the i^{th} voxel in the 3D fulfilling model, and the $\hat{y}_i$ was the i^{th} voxel in the 3D CAD model. was the indicator function where the ϵ is the threshold.

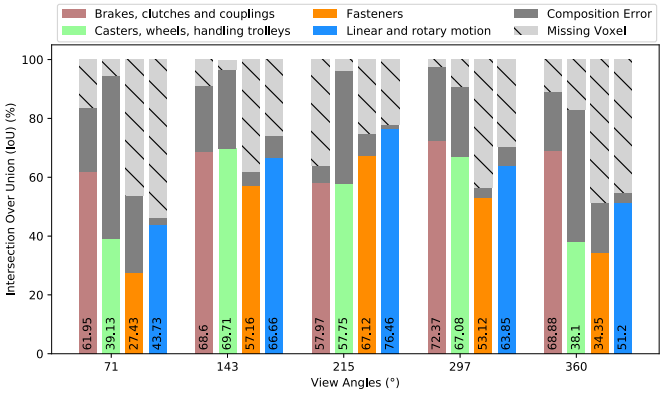


Figure 6: The evaluation of the 3D composition model in terms of IoU, the composition error, and the missing voxel. The view angle was along the Z-axis.

The composition error (IoU) and the missing voxel were evaluated in five different view angles along the Z-axis for each category, and the mean of each category is shown in the Table 6. There were 125 RGB-D scans acquired from different angles for each object in each category in the Blender environment. As shown in the table, an observation can be found that the "brakes, clutches, and coupling" category achieved better IoU in the oblique viewing angles $(71°, 143°, 297°, 360°)$, because objects in this category are thin and flat. Therefore, RGB-D scans can only acquire few data points from an oblique viewing angle. For objects in the second category ("casters, wheels, handling trolleys") achieved less composition error, because the objects in this category contain hollow structures. RGB-D scans that were taken from sides were not able to observe the hollow structure. For objects in the third, and fourth category ("fasteners" "linear and rotary motion") achieve better performance with the $215°$ viewing angle, since objects in these two categories are solid, and compounds with regular shape. So, the RGB-D scans can obtain the general shape of objects from $215°$. In conclusion, the more data points, more complicated shape of objects that were observed by RGB-D scans can achieve better IoU, less error composition, and less missing values.

Another experiment has been done to compare the projection overlapping between the composited point cloud and the ground truth. The point cloud of 3D composited model and the ground truth of objects have been projected orthogonally on X-Y plane, Y-Z plane, and Z-X plane. An example of results was shown in the Fig. 7, where the blue colored points are the ground truth, and the green colored points are the composited 3D model. The projection overlapping on each plane are shown in Table I. In Table I, objects in the second category (casters, wheels, handling trolleys) that contain hollow structures achieved better performance in X-Y direction. In contrast to that, objects in the third category (fasteners), and objects in the fourth category (linear and rotary motion), that are solid, and compounds with regular shapes achieved better performance in Y-Z and Z-

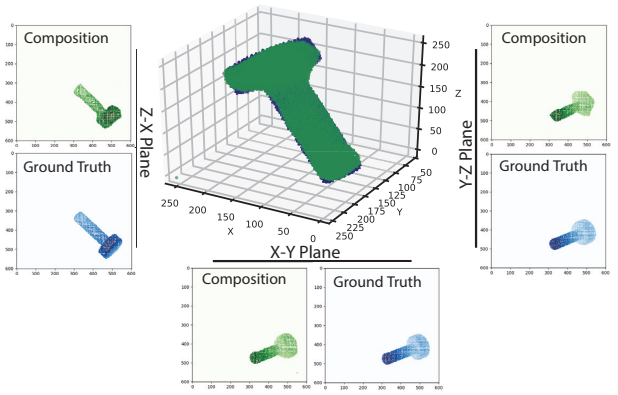


Figure 7: The orthogonal projection example.

Table I: Projection overlap between the 3D composited model and ground truth on different planes. Taxonomy of the categories were based on the Traceparts.

Category	X-Y	Y-Z	Z-X
Brakes, clutches and couplings	60%	47%	86%
Casters, wheels, handling trolleys	79%	12%	46%
Fasteners	43%	82%	72%
Linear and rotary motion	66%	77%	87%

X directions.

The evaluation results for both experiments were considered reasonable, since both experiment results were shown that the more data points were acquired by RGB-D scans, the more accurate the 3D composition model is. It is same for human beings, most scenarios that human can not guess the completed shape of an object by only observing a part of it.

An comparison results has been done between the proposed method with 3DR2N2 [10], the comparison results were shown in the Table II. In the experiment, a single scan of objects has been used as input for both models. In table II, the proposed 3D shapes composition of objects outperformed 3DR2N2 in IoU. Objects were divided based on the shapes for 3DR2N2 method. Based on the results of comparison, the generation of missing points in point clouds by using depth features were better than generating randomly from latent distribution.

Table II: Accuracy of the 3D composition (IoU).

Category	This Paper	**3DR2N2** [10]
Brakes, clutches and couplings	70.48%	64.2%
Casters, wheels, handling trolleys	68.39%	59.3%
Fasteners	67.12%	53.3%
Linear and rotary motion	76.46%	73.85%

B. Evaluation of the Spatial Constraints Understanding

There were ten spatial constraints evaluated in this experiment, which were “left”, “right”, “above”, “below”, “front”, “behind”, “parallel”, “perpendicular”, “concentric”, and “coincident”. We randomly selected 500 pairs of objects from the pre-formed dataset to evaluate the model, and the prediction accuracy of the proposed method is provided in Table III.

Some sample results of understanding spatial constraints are shown in Fig. 8. The inputs of the model were RGB-D scans of objects in different categories. We randomly selected CAD models of objects [14] to form pairs to train the proposed method. There were 535,760 pairs of objects that hold each of those ten spatial constraints are generated. Objects were randomly selected from the pre-formed dataset to formulate spatial constraints such as “left”, “right”, “above”, “below”, “front”, and “behind” objects, and the angle differences between two objects’ center points were less than 15°. The cylindrical objects such as tubes, screws, and nuts were selected to formulate the “concentric” relations. Objects with large flat surfaces were selected for formulating the “parallel”, “perpendicular”, and “coincident” relations.

Table III: Comparison study for the prediction accuracy of the spatial constraint

	This Paper	**RANSEM** [15]	**CPN** [16]
Left	99.7%	86.8%	79%
Right	99.5%	88.9%	79%
Above	99.6%	-	82%
Below	99.4%	-	79%
Front	99.5%	-	-
Behind	99.5%	-	-
Coincident	99.3%	-	-
Parallel	99.7%	-	-
Perpendicular	99.4%	-	-
Concentric	99.1%	-	-

The comparison of the prediction accuracy and the spatial constraints among the proposed method, RANSEM [15], and CPN [16] is shown in Table III. The proposed method’s prediction accuracy outperformed previously approaches [16], [17]. Our method is more comprehensive because all spatial constraints used in the CAD software for assembly are considered. One limitation for the proposed method is that our method is suitable to comprehend precise spatial constraints. Some blurred spatial constraints such as “near” and “far” cannot be learned because these blurred spatial constraints are subjective and not able to define in logic constraints exhaustively.

The pioneered CPN [16] utilized k-nearest neighbors method to classify features that were extracted from the projected point clouds. That method can recognize four spatial constraints, and they chose 128 images of working space and split 95 of them as training images and 33 of them as testing images. Instead of extracting features from the projected point cloud, this paper directly extracted features from point clouds, which were considered containing rich information.

The RANSEM [15] extracted relevant spatial features face-centric geometric descriptors (FGDs) to classify spatial

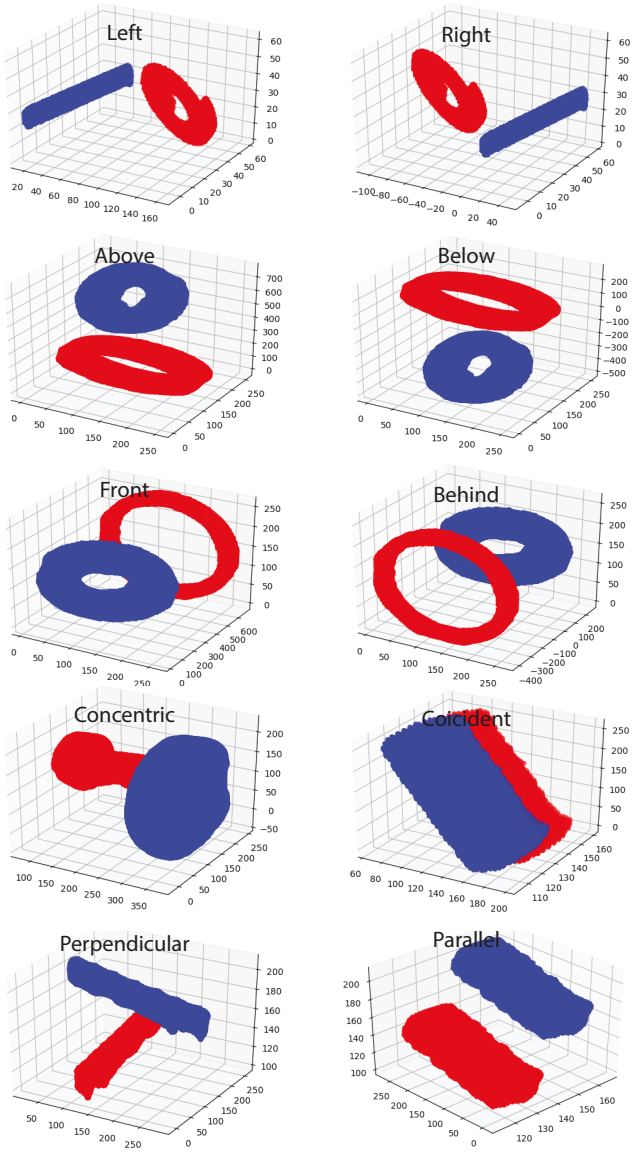


Figure 8: Experiment results of spatial constraints understanding of object: flat-head bolt, piston sealing, nut washer, rebar, gear, and supporting block.

constraints that are held between objects. The proposed paper using spatial symbolic knowledge, which was described by logic rules to enhance the network training. In contrast to the synthetic bounding volume of objects, the 3D shape of objects were composited by the proposed extended 3D GAN model. The composited 3D shape of objects has higher resolution and contains more features than synthetic bounding volumes, especially for complex objects.

IV. Conclusion

This paper has presented a method that takes a single RGB-D scan of the objects and learn the spatial constraints of objects. The proposed approach can recognize ten different spatial constraints, which are left, right, above, below, front, behind, parallel, perpendicular, concentric, and coincident relations. The overall accuracy spatial constraint understanding is 99%, demonstrating state-of-the-art performance. The intersection over union (IoU) has achieved 57.23% overall for objects in four different categories.

References

[1] H. He, Z. Shao, and J. Tan, "Recognition of car makes and models from a single traffic-camera image," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3182–3192, 2015.

[2] H. He, Y. Li, Y. Guan, and J. Tan, "Wearable ego-motion tracking for blind navigation in indoor environments," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 4, pp. 1181–1190, 2015.

[3] V. V. Unhelkar, P. A. Lasota, Q. Tyroller, R.-D. Buhai, L. Marceau, B. Deml, and J. A. Shah, "Human-aware robotic assistant for collaborative assembly: Integrating human motion prediction with planning in time," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2394–2401, 2018.

[4] C. Paxton, A. Hundt, F. Jonathan, K. Guerin, and G. D. Hager, "Costar: Instructing collaborative robots with behavior trees and vision," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 564–571.

[5] V. V. Unhelkar, S. Dörr, A. Bubeck, P. A. Lasota, J. Perez, H. C. Siu, J. C. Boerkoel, Q. Tyroller, J. Bix, S. Bartscher *et al.*, "Mobile robots for moving-floor assembly lines: Design, evaluation, and deployment," *IEEE Robotics & automation magazine*, vol. 25, no. 2, pp. 72–81, 2018.

[6] O. Mees, N. Abdo, M. Mazuran, and W. Burgard, "Metric learning for generalizing spatial relations to new objects," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 3175–3182.

[7] H. Li, J. Tan, and H. He, "Magichand: Context-aware dexterous grasping using an anthropomorphic robotic hand," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9895–9901.

[8] F. Yan, D. Wang, and H. He, "Robotic understanding of spatial relationships using neural-logic learning," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 8358–8365.

[9] B. Yang, S. Rosa, A. Markham, N. Trigoni, and H. Wen, "Dense 3d object reconstruction from a single depth view," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 12, pp. 2820–2834, 2018.

[10] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3d-r2n2: A unified approach for single and multi-view 3d object reconstruction," in *European conference on computer vision*. Springer, 2016, pp. 628–644.

[11] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor fusion IV: control paradigms and data structures*, vol. 1611. International Society for Optics and Photonics, 1992, pp. 586–606.

[12] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, vol. 29, 2016.

[13] L. Serafini and A. S. d. Garcez, "Learning and reasoning with logic tensor networks," in *Conference of the Italian Association for Artificial Intelligence*. Springer, 2016, pp. 334–348.

[14] S. Koch, A. Matveev, Z. Jiang, F. Williams, A. Artemov, E. Burnaev, M. Alexa, D. Zorin, and D. Panozzo, "Abc: A big cad model dataset for geometric deep learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9601–9611.

[15] J. Li, D. Meger, and G. Dudek, "Learning to generalize 3d spatial relationships," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 5744–5749.

[16] B. Rosman and S. Ramamoorthy, "Learning spatial relationships between objects," *The International Journal of Robotics Research*, vol. 30, no. 11, pp. 1328–1342, 2011.

[17] S. Fichtl, A. McManus, W. Mustafa, D. Kraft, N. Krüger, and F. Guerin, "Learning spatial relationships from 3d vision using histograms," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 501–508.