2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 23-27, 2022, Kyoto, Japan

6D Robotic Assembly Based on RGB-only Object Pose Estimation

Bowen Fu*, Sek Kun Leong*, Xiaocong Lian and Xiangyang Ji

***Abstract*— Vision-based robotic assembly is a crucial yet challenging task as the interaction with multiple objects requires high levels of precision. In this paper, we propose an integrated 6D robotic system to perceive, grasp, manipulate and assemble blocks with tight tolerances. Aiming to provide an off-the-shelf RGB-only solution, our system is built upon a monocular 6D object pose estimation network trained solely with synthetic images leveraging physically-based rendering. Subsequently, pose-guided 6D transformation along with collision-free assembly is proposed to construct any designed structure with arbitrary initial poses. Our novel 3-axis calibration operation further enhances the precision and robustness by disentangling 6D pose estimation and robotic assembly. Both quantitative and qualitative results demonstrate the effectiveness of our proposed 6D robotic assembly system.**

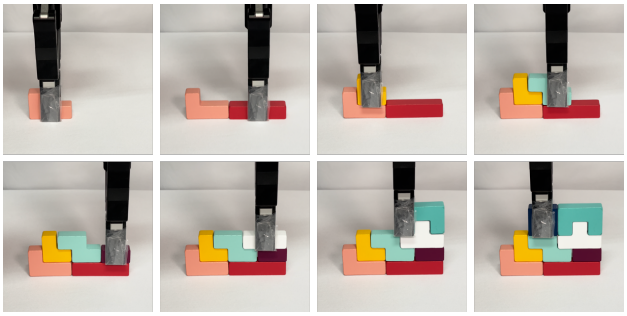


Fig. 1. Example of an assembly process. Given a desired architecture, our system autonomously and precisely assembles the blocks.

I. Introduction

Although building blocks is natural for humans, it is quite challenging for robots. A block needs to be perceived, grasped, manipulated, and then appropriately assembled with an extremely tight tolerance, thus requiring a highly robust and precise vision algorithm. In this work, we focus on establishing a flexible 6D robotic system to assemble blocks based on monocular 6D object pose estimation (Fig. 1).

Robotic assembly tasks including peg-in-hole and block stacking have been studied for decades. Peg-in-hole tasks are typically implemented on a board [1]–[3], thus only 2D information is required. Some works exploit 2D object detection [4], [5] or 3D pose estimation [6] to perceive the objects. By simplifying the assembly task from 3D space into a 2D plane with only 2 or 3 degrees of freedom, the information along the principal axis is lost. Therefore, only unidirectional assembly can be implemented, rendering flexible multi-angle block assembly tasks impossible. Additionally, force-torque sensors are employed in some works [5], [7], [8], whereas the high cost oftentimes prohibits their practical applications.

Some works leverage reinforcement learning to conduct block stacking tasks [9], [10], which attempt to teach robots to stack one block on top of another one. The learned policies are capable of dealing with multiple object combinations and demonstrate a large variety of stacking skills. However, only simple stacking operation can be implemented. There is still a long way to meet the requirement of robotic assembly.

To conduct more complex grasping and assembly tasks, a robot needs to interact with objects in 3D space with full 6 degrees of freedom. Thanks to the rapid development in convolutional neural networks (CNNs), recently, a few methods attempt to conduct robotic assembly through learning-based 6D object pose estimation. For instance, [11] formulates the task of assembly as predicting the 6D poses of template geometries to enable manipulating objects in arbitrary contexts. However, the object is manually placed into the gripper to avoid the grasp error accumulating to the assembly process, which lacks system integrity. Additionally, [12] conducts multiple open-world assembly tasks leveraging an RGBD-based 6D object pose tracker. However, the dependence on surrounding objects and depth information may restrict its application.

In this work, an integrated robotic system is established to autonomously and precisely assemble blocks with tight tolerances. Only RGB input is required and the prior information of the object is strictly confined to an easily acquired 3D model, making it easy to extend to real-world applications. Our system is built upon a high-precision real-time monocular 6D object pose estimation methodology, with all the training and validation data generated by physically-based rendering. The learned model can be directly applied to real-world robotic manipulation without further training or using domain randomization techniques. Aided by the proposed pose-guided 6D transformation strategy, our system is capable of assembling arbitrary given structures with arbitrary initial block poses. To further enhance the precision and robustness, 3-axis calibration is introduced to decouple 6D pose estimation and the robotic assembly process, which eliminates the effect of pose error on assembly.

Our contributions can be summarized as follows:

- We establish an integrated 6D robotic assembly system to assemble blocks with arbitrary initial poses to any given structure, only requiring RGB input.
- We propose a pose-guided 6D transformation strategy

* Equal contribution

This work was supported by the National Key R&D Program of China under Grant 2018AAA0102801, National Natural Science Foundation of China under Grant 61620106005.

All authors are with the Department of Automation and BNRist, Tsinghua University, Beijing, China. {fbw19, lxq20}@mails.tsinghua.edu.cn, {lian900625, xyji}@tsinghua.edu.cn

2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 978-1-6654-7927-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/IROS47612.2022.9982262

along with an RGB-based 6D pose regression network [13] with pure synthetic training, providing high-precision real-time interaction between blocks and the robot system.
- We further promote the precision and robustness by 3-axis calibration. The effectiveness of our methodology is demonstrated by robotic assembly tasks with 1mm tolerance.

II. Related Work

The presented work relates to two major strands of research: 6D object pose estimation and vision-based robotic assembly.

A. 6D Object Pose Estimation

6D object pose estimation has received a lot of attention in both robotics and computer vision communities in recent years. Some works conduct indirect approaches to predict the 6D pose. A popular approach is to establish 2D-3D correspondences, which are subsequently exploited for computing the 6D poses by P*n*P algorithm. For instance, BB8 [14] and YOLO6D [15] compute the 2D projections of 3D bounding box corners. To enhance the robustness, SegDriven [16] and PVNet [17] employ segmentation paired with voting for each correspondence. Meanwhile, CDPN [18] and DPOD [19] predict dense rather than sparse correspondences and achieve remarkable performance. Similarly, EPOS [20] represents the object by compact surface fragments to handle symmetries.

Another branch of works directly regresses the 6D pose. PoseCNN [21] and DeepIM [22] leverage a point matching loss and CosyPose [23] extends DeepIM [22] by introducing multi-view information and implementing global scene refinement. SingleStage [24] and GDR-Net [13] infer intermediate 2D-3D correspondences and directly regress the 6D pose via the network rather than the P*n*P algorithm, showing that the learned P*n*P is capable of producing more robust estimates than standard P*n*P. Additionally, SSD6D [25] discretizes the pose space and conducts classification rather than regression.

Noteworthy, the majority of these methods exploit annotated real data for training. However, labeling real data requires tremendous consumption of time and labor. Hence, some works completely rely on synthetic data to avoid this defect [26], [27]. Though the performance of these methods falls behind those using real annotations, the domain gap between synthetic and real images can be narrowed by physically-based rendering [28], [29] or domain randomization strategies.

B. Vision-based Robotic Assembly

With the blossom of deep learning, vision-based robotic grasping and assembly progress rapidly. Compared with robotic grasping tasks [30], [31], robotic assembly tasks require higher precision because the grasped object interacts not only with the gripper but also with the target object. Peg-in-hole insertion tasks and block stacking tasks have drawn much attention among others.

Peg-in-hole insertion tasks require robots to assemble gears, shafts, etc. Various challenges have been proposed in the last few years, *e.g.*, the NIST Assembly Task Boards [1] and the World Robot Summit (WRS) Assembly Challenge [2], [3]. The targets are typically placed on a plane, thus many peg-in-hole insertion methods leverage 2D object detection [4] or 3D pose estimation (*i.e.*, the 2D object center $(x,\ y)$ and the orientation angle θ) [6]. [5] utilizes YOLOv5 to detect target objects and accomplish assembly using visual and force servoing. Their interaction within the 3D space is limited and the flexible multi-angle assembly is beyond their reach.

Meanwhile, several works deal with vision-based robot stacking tasks by reinforcement learning [9], [10], trying to teach robots to stack one block on top of the other block. Their methods leverage few prior and rely on physical simulation to perform large-scale robot training, which proves the possibility of learning a vision-based policy to stack multiple object combinations. However, only simple tasks, *i.e.*, stack one block on top of the other, can be conducted. There is still a long way towards high-precision block stacking tasks leveraging reinforcement learning.

Recently, some works combine pose estimation and robotic assembly or stacking. [11] introduces a new robotic assembly task, which requires reasoning about local geometry that is surrounded by arbitrary context and formulates the task as the 6D pose estimation of template geometries. [12] conducts plug insertion, box packing and cup stacking tasks leveraging RGBD-based 6D object pose tracking along with within-hand visual feedback control.

III. Method

In this paper, we aim to exploit a 6D robotic assembly system to build blocks with tight tolerances by RGB-only input (Fig. 2). To implement the task autonomously and precisely, we first perceive all the blocks leveraging 2D object detection and 6D object pose estimation. After conducting multiple grasp and place operations to interact with each block, pose-guided 6D transformation and collision-free assembly are proposed to deal with arbitrary initial poses and designed structures. Moreover, the key challenge of the assembly task is the gap between the precision and robustness required by the robot system and those provided by the computer vision algorithm. Therefore, 3-axis calibration is proposed to further enhance the precision and robustness by disentangling 6D pose estimation and the 6D assembly process.

A. Data Generation

For training the 2D object detection network and the 6D object pose estimation network, we generate photorealistic images leveraging physically-based rendering. Only 3D object models are required in this process. Since the blocks are standard and regular, we manually model them by SolidWorks and re-sample the surface utilizing Poisson-disk sampling to obtain dense point clouds.

Some 6D pose estimation methods [17], [18], [22] render 3D object models in an arbitrary pose and randomly choose a

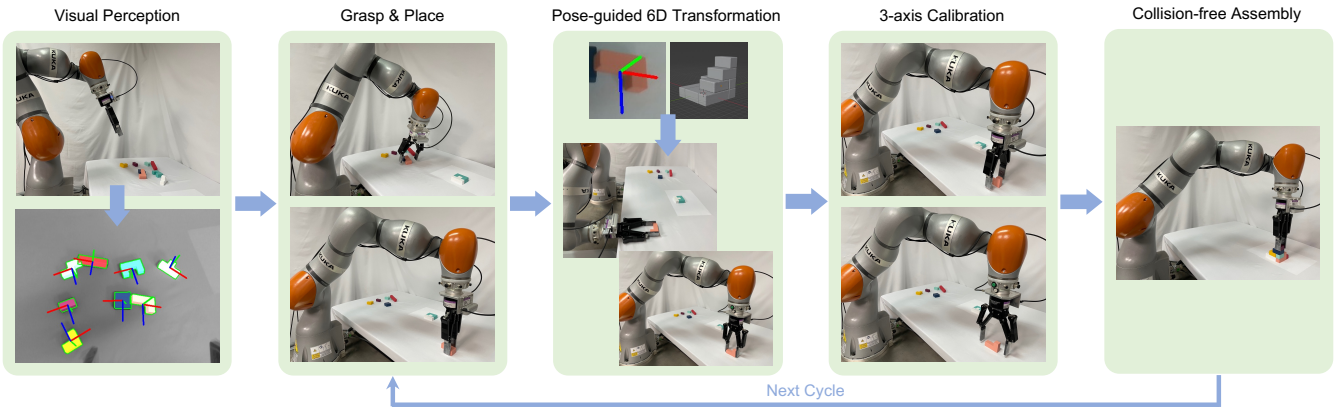


Fig. 2. Flow chart of our system. After perceiving all the blocks by 2D object detection and 6D object pose estimation, we implement grasp & place, pose-guided 6D transformation, 3-axis calibration and collision-free assembly for each block until the intact architecture is constructed.

background image from PASCAL VOC [32] or SUN397 [33] dataset to generate synthetic data. However, the generated images are unrealistic due to the fragmented foreground and background, which triggers the domain gap between synthetic and real data.

In contrast to them, we exploit BlenderProc [29], a physically-based rendering pipeline, to generate synthetic training and validation data. Specifically, each object is assigned with a uniformly sampled initial pose and then dropped to the ground of a room with a random photorealistic material from the CC0 Textures library. The synthetic images are rendered leveraging Blender with properly set light and camera positions. Since the Pybullet physics engine records the real-time object poses, we simultaneously acquire both 2D detection and 6D pose annotations of the images.

As shown in Fig. 4 (a), abundant high-quality images with precise annotations can be generated for any given model, which is highly efficient and labor-saving. Our methodology greatly narrows the domain gap between synthetic and real images. Subsequently, the 2D detection network and 6D pose estimation network, which are solely trained on synthetic data, can be directly applied to real scenes without further training or using domain randomization techniques.

B. 6D Pose Estimation

Given an RGB image and a set of objects together with their corresponding 3D models, we aim to estimate the 6D pose $P = [R|t]$ w.r.t. the camera for each object, with R representing the 3D rotation and t denoting the 3D translation.

Fig. 3 presents a schematic overview of the proposed methodology. We follow GDR-Net [13] to directly regress the 6D pose. Leveraging physically-based rendering, we train the 2D detection network and the 6D pose regression network solely on synthetic data. Specifically, we first detect all objects of interest using the trained detection network. For each corresponding Region of Interest (RoI), we feed it to the pose network exploiting Dynamic Zoom-In (DZI) [18] strategy, which decouples the 2D detection process and 6D pose estimation process. Then several intermediate geometric feature maps indicating the 2D-3D correspondences are predicted, and finally, a CNN module that simulates the PnP algorithm is exploited to directly regress the 6D object pose.

Following [13], we employ a disentangled 6D pose loss, individually supervising the rotation R, the scale-invariant 2D object center [18] and the distance. We parameterize rotation R as the first two columns of the rotation matrix, which has been demonstrated effective [13], [34].

To sum up, with a glance of the camera mounted to the gripper, we acquire the current poses of all blocks w.r.t. the camera by our 6D pose estimation methodology. Aided by the high quality of training data and the generalization ability of the network, our system requires no prior of working scenario, which vastly extends its application range.

C. Grasp Strategy and Collision-free Assembly

Based on the 6D poses we have acquired, the robot system is capable of interacting with the blocks. For implementing subsequent grasp operations, we first consider two coordinate systems: the object coordinate system O_{obj}, defined in the 3D object model, and the robot base coordinate system O_{base}. The 6D pose represents the transformation T_{obj}^{cam} from O_{obj} to the camera coordinate system O_{cam}. With the offline calibration between the camera and the robot flange and the robot kinematics parameters, the transformation from O_{obj} to O_{base} is obtained by

$$T_{obj}^{base} = T_{flange}^{base}\ T_{cam}^{flange}\ T_{obj}^{cam}, \tag{1}$$

where T denotes the homogeneous transformation matrix. For simplification, we consider the axis of O_{obj} with the minimum angle to the vertical axis of O_{base} as the reference axis

$$I_{ref} = \arg\max_{v} \langle e_z,\ R_{obj}^{base}\ v \rangle, \tag{2}$$

where v denotes the unit vector of the axes of O_{obj} and e_z denotes the unit vector of the vertical axis of O_{base}.

We conduct a grasp towards the block center along the reference axis to avoid the unreachable point of the robot arm and collisions with the other blocks. However, in the assembly process, a proper grasp position needs to be found for the gripper to avoid collisions with the other blocks.

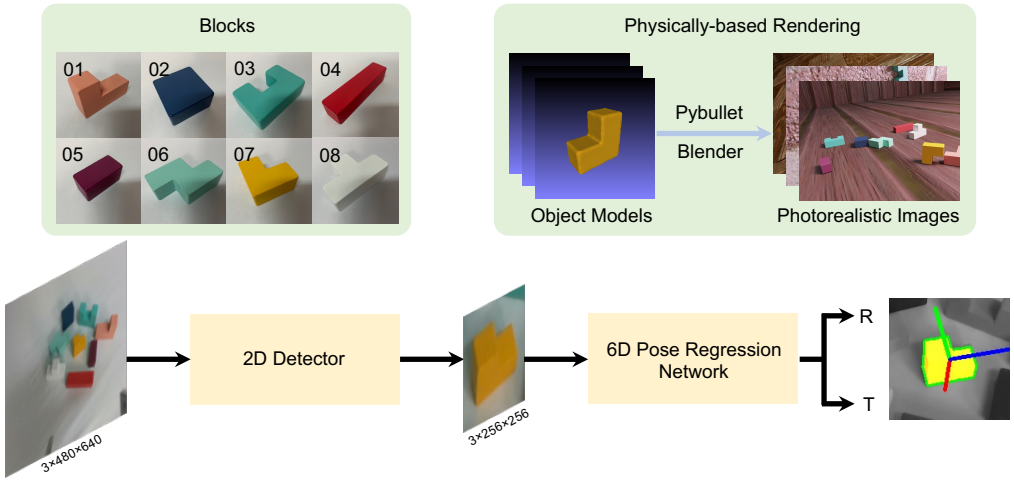


Fig. 3. Overview of our 6D pose estimation methodology. We train the 2D detection network and the 6D pose estimation network with pure synthetic images leveraging physically-based rendering. During inference, we feed a real image captured by the camera to the networks and obtain the 6D pose. 8 blocks are employed in our experiments.

Considering the blocks are regular, we conduct grasping along the axes in O_{obj}, including $\pm x$, $\pm y$ and $\pm z$. For each direction, we pre-define two orthogonal grasps. For instance, the grasp along $+x$ includes two orthogonal grasps in x-y plane and x-z plane. Considering directly grasping the object center may be infeasible due to the collisions with the designed structure, we also pre-define two extra grasp positions for each direction, with offsets relative to the center. To sum up, 36 grasp candidates are pre-defined for each block in O_{obj}.

Knowing the block poses and the grasp candidates, we detect collisions between the gripper and the designed structure leveraging the Flexible Collision Library (FCL) [35] to exclude infeasible grasp candidates. Noteworthy, the 2-finger gripper can be formulated as two cuboids, which is enough for filtering out infeasible grasp candidates.

D. Pose-guided 6D Transformation

We design a target structure manually in Blender, where the relative 6D poses between each block can be obtained. Simultaneously, the assembly sequence is also allocated. As long as we assign the absolute 6D pose of the first block in O_{base}, all the poses of blocks can be inferred, which makes our method easy to transfer to other desired structures.

Therefore, the arbitrary initial and target block poses have been obtained respectively, which guides the robot to implement 6D transformation for each block. Similar to the grasping process, the rotation along the vertical axis of O_{base} is easy to deal with while the rotation along the other two axes may be out of range due to the limited workspace of the robot arm. We would like to avoid wide angle rotation along the horizontal axes of O_{base}.

If the target pose shares the same reference axis with the current pose, the assembly can be conducted simply by virtue of the rotation along the vertical axis of O_{base}. Otherwise, we move the block to a rotation workspace and rotate it along one horizontal axis of O_{base} to make the target reference axis upturned. Noteworthy, up to two rotations are sufficient for any arbitrary conditions and the number of rotations is related to the angle between the reference axes of the current pose and the target pose. Note that to account for symmetries, we compute all correct poses by multiplying the estimated pose and rotation matrix of symmetry and select the most convenient transformation strategy.

E. 3-axis Calibration

Although we achieve relatively accurate 6D pose estimation, the assembly precision is still not satisfied. To be specific, in 6D pose estimation tasks, the average error of model points is usually considered acceptable up to 10% of the diameter. The grasp and place operation shares a similar tolerance and can still be implemented stably and robustly. However, in assembly tasks, the pose estimation error can lead to collisions or gaps. Considering the block only interacts with the gripper in the grasp and place operation but needs to interact with both the gripper and other blocks in the assembly operation, the precision required by the assembly operation is much higher than that required by the grasp and place operation. To eliminate the impact of pose error on assembly, we propose 3-axis calibration to decouple 6D pose estimation and assembly process.

Assume that there is a small rotation error ΔR and translation error Δt between ground-truth pose $(\bar{R}, \bar{t})$ and estimated pose (R, t). We first grasp and place the block on a plane. Since the block is regular, ΔR_x, ΔR_y and Δt_z can be eliminated due to the limitation of the plane. Then we conduct two orthogonal grasps along the x axis and the y axis of O_{obj} to the estimated pose, forcing the block to shift to the estimated pose. Therefore, ΔR_z, Δt_x and Δt_y are eliminated, which are tiny yet crucial to assembly process.

Noteworthy, 3-axis calibration shares the same plane prior with the preceding process, thus does not leverage supernumerary prior. On account of 3-axis calibration, the blocks can still be anywhere with arbitrary initial poses, even be stacked

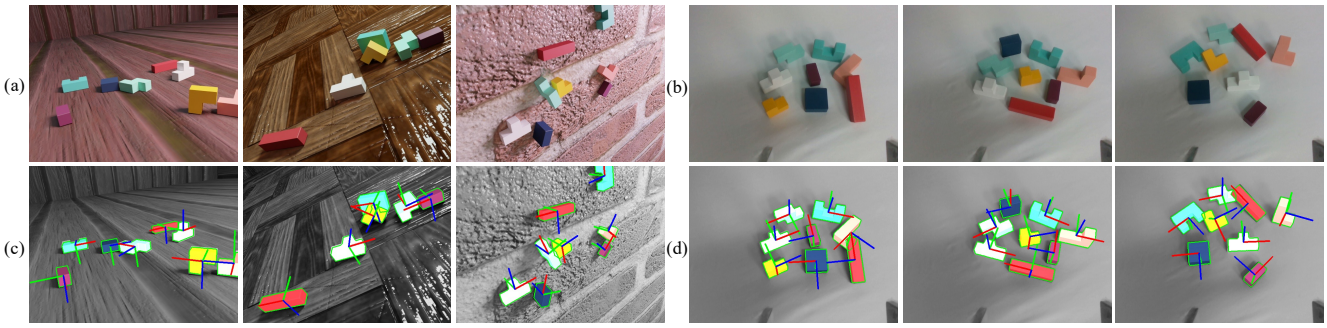


Fig. 4. The qualitative 6D pose estimation results of photorealistic and real-time images. (a) The photorealistic images we generate leveraging physically-based rendering. (b) The real-time images we capture. (c, d) The corresponding qualitative 6D pose estimation results by rendering the 3D models and overlaying the contours and coordinate axes on the image.

TABLE I
RESULTS FOR 6D POSE ESTIMATION ON VALIDATION SET

Block	ADD(-S) 0.02d	0.05d	0.1d	5° 5cm	2cm
01	57.39	82.91	88.10	80.95	89.26
02	51.28	81.47	88.59	53.53	91.22
03	61.23	83.40	89.69	82.35	90.43
04	54.50	79.81	87.69	73.81	87.81
05	43.74	78.26	88.71	82.32	92.71
06	60.83	85.33	90.61	82.75	91.16
07	44.79	79.63	88.82	80.91	91.32
08	53.17	83.96	90.11	84.39	91.79
Mean	53.37	81.84	89.04	77.63	90.71

together. The pose error during perception, grasp and place process cannot be accumulated to the assembly process, thus the success rate is improved dramatically.

IV. EXPERIMENTS

We use eight blocks for 6D pose estimation and robotic assembly experiments. In this section, we first evaluate our 6D object pose estimation algorithm. Then robot assembly experiments are implemented to demonstrate the effectiveness of our methodology.

A. 6D Pose Inference

1) Implementation Details: We generate 38000 photorealistic synthetic images for training and 2000 for validation. All eight blocks are dropped randomly from 0.2 to 0.4 m in a -0.15 to 0.15 m square with a uniformly sampled rotation matrix in Blender. The camera is placed within an annulus with 0.55 to 0.85 m radius and 0.5 to 0.9 m elevation. The 2D detection and 6D pose estimation experiments are implemented using PyTorch. We leverage YOLOv4 [36] for detection. The pose network is trained end-to-end using Ranger optimizer with a batch size of 64 and a base learning rate of 1e-4.

2) Evaluation Metrics: We employ ADD(-S) and n° n cm for evaluation. The ADD metric [37] measures whether the average distance of the model vertices between the predicted pose and the ground-truth pose is less than a certain percentage of the object's diameter. For symmetric objects, the ADD-S metric is employed to measure the error as the average distance to the closest model vertices [37] [38]. The n° n cm metric measures whether the rotation error is under n° and the translation error is less than n cm. It is computed w.r.t. the smallest error for all possible ground-truth poses for symmetric objects.

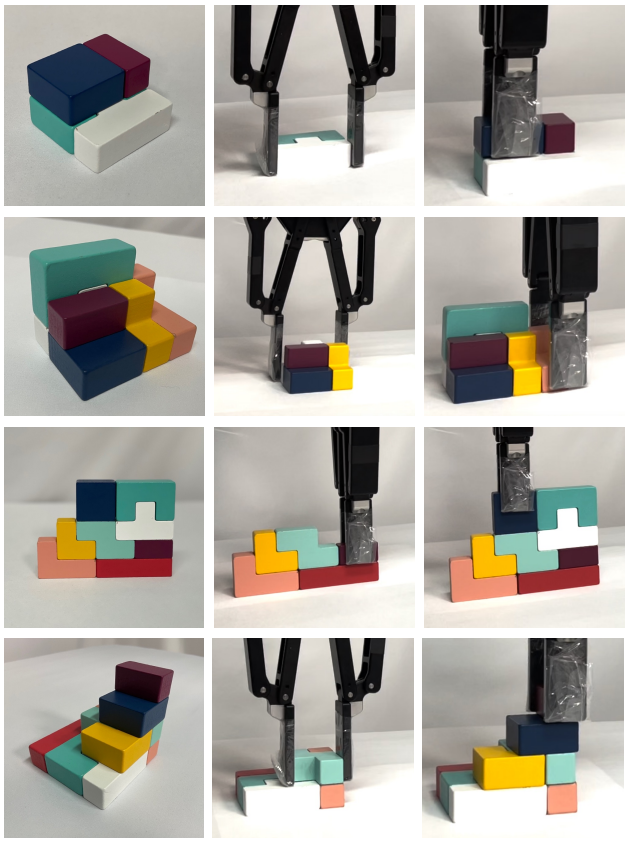
3) Analysis: Table I shows the performance of our 6D pose estimation methodology. As can be seen, we achieve 89.04 in ADD(-S)-0.1d and 77.63 in 5° 5cm on the validation set. With regard to more rigorous metrics ADD(-S)-0.05d and ADD(-S)-0.02d, we achieve 81.84 and 53.37, respectively. We also demonstrate qualitative results in Fig. 4 for synthetic and real-world data. Although we do not acquire the ground-truth 6D poses in real scenes, the visualization reflects the high precision of our methodology. Aided by our 6D pose estimation methodology, arbitrary initial block poses can be handled even when the blocks are stacked together with occlusion.

4) Inference time: With an AMD Ryzen 7 5800 CPU and an NVIDIA RTX 3090 GPU, given a 640×480 image, using YOLOv4 [36] detector, our approach takes 45ms for eight objects, including 21ms for detection.

B. Robot Assembly Experiments

The robotic system consists of a KUKA LBR iiwa R820 robot arm and a 2-finger Robotiq 2F-140 gripper, with an Intel RealSense D435i camera mounted to the gripper. The offset between the camera and gripper is obtained in an off-line calibration procedure. We conduct robot experiments leveraging Robotic Operating System (ROS). IIWA STACK [39], a ROS metapackage for KUKA LBR iiwa robots, is employed to communicate between ROS and KUKA Sunrise software.

We design four structures for assembly (Fig. 5 (a)) and conduct 15 trials for each structure. As demonstrated in Table II, our methodology achieves a 100% 2D detection rate and the 6D poses of all blocks are predicted, respectively. A trial is considered successful if all the blocks are properly assembled without severe collision and the assembly gap is less than 1mm as well. Among all the 60 trials, 52 are successful with the target structure constructed, achieving an 86.7% success rate. To further analyze the whole construction process, we consider the perception, grasp, manipulation and assembly of a single block as a step. A trial consists of a



(a) (b) (c)

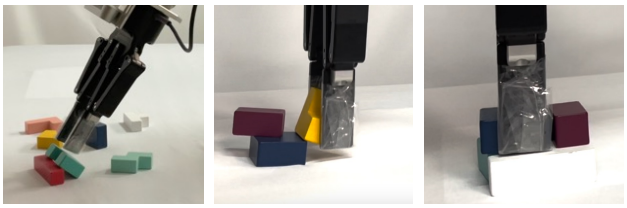
Fig. 5. The results of robotic assembly. (a) Target structures. (b, c) The corresponding robotic assembly process.

different number of steps according to the number of blocks. We achieve a 93.5% step success rate among all the steps in four structures.

Noteworthy, the target structures are designed with tight tolerance and no gap is reserved between the blocks, thus only 1mm error is admissible during the assembly process. With a precise and robust 6D pose estimation algorithm, pose-guided 6D transformation strategy and 3-axis calibration, the assembly with tight tolerance is successfully implemented as shown in Fig. 5.

The system error is principally derived from 6D pose estimation and calibration process. When the block is severely occluded or in certain ambitious conditions, our system may perceive a rough 6D pose. In most cases, the grasp can still be implemented and then the 6D pose error is eliminated by 3-axis calibration, which contributes to one successful assembly. Conversely, the 6D pose error that exceeds the tolerance of robotic grasping leads to a grasping failure (Fig. 6 (a)).

For the assembly error less than 1mm, only slight collisions and weeny gaps occur with no damage to the whole structure, while the assembly error larger than 1mm may wreck the whole structure (Fig. 6 (b)). To summarize, our robotic assembly system is capable of assembling blocks with a tolerance of 1mm by RGB-only input.



(a) (b) (c)

Fig. 6. Failure cases. (a) caused by 6D pose estimation error. (b) caused by calibration error. (c) without 3-axis calibration.

TABLE II

RESULTS FOR ROBOTIC ASSEMBLY TASK

Structure	Number of Blocks	Detection Rate	Step Success Rate	Trial Success Rate
1	4	100.0%	94.7%	93.3% (14/15)
2	6	100.0%	91.8%	86.7% (13/15)
3	8	100.0%	95.8%	86.7% (13/15)
4	8	100.0%	91.7%	80.0% (12/15)
Mean		100.0%	93.5%	86.7% (52/60)

C. Ablation Study on 3-axis Calibration

To demonstrate the effectiveness of 3-axis calibration, we attempt to assemble structure 1, which consists of only four blocks, without exploiting the 3-axis calibration operation for the last 3 blocks. Experimental results show that 13 out of 15 trials failed with severe collision and the collapse of the whole structure (Fig. 6 (c)). Our 3-axis calibration operation outperforms direct assembly by a large margin, which indicates the indispensability of the disentanglement of 6D pose estimation and assembly process.

It is worth noting that we make some assumptions in our approach, *e.g.*, blocks with regular shapes, flat surfaces and stable flipping poses. Further exploration is needed for more general robotic assembly tasks.

V. CONCLUSION

In this paper, we establish an integrated 6D robotic assembly system to assemble blocks with only RGB input. Leveraging a monocular 6D object pose estimation methodology, our system is capable of perceiving and interacting with the blocks with arbitrary initial poses. Abundant photo-realistic training and validation data are generated exploiting physically-based rendering, which avoids complicated and labor-consuming 6D pose labeling in real scenes. Trained solely with synthetic data, our 2D detection network and 6D pose regression network are capable of directly transferring to real scenes without further training or using domain randomization techniques.

Correspondingly, pose-guided 6D transformation along with collision-free assembly is proposed to deal with arbitrary initial poses and target poses. Our novel 3-axis calibration operation further promotes the precision and robustness of our system by decoupling 6D pose estimation and the assembly process. The experimental results demonstrate that our system is capable of robustly assembling blocks with 1mm tolerances.

REFERENCES

[1] K. Kimble, K. Van Wyk, J. Falco, E. Messina, Y. Sun, M. Shibata, W. Uemura, and Y. Yokokohji, "Benchmarking protocols for evaluating small parts robotic assembly systems," *IEEE robotics and automation letters*, vol. 5, no. 2, pp. 883–889, 2020.

[2] Y. Yokokohji, Y. Kawai, M. Shibata, Y. Aiyama, S. Kotosaka, W. Uemura, A. Noda, H. Dobashi, T. Sakaguchi, and K. Yokoi, "Assembly challenge: a robot competition of the industrial robotics category, world robot summit–summary of the pre-competition in 2018," *Advanced Robotics*, vol. 33, no. 17, pp. 876–899, 2019.

[3] F. Von Drigalski, C. Schlette, M. Rudorfer, N. Correll, J. C. Triyonoputro, W. Wan, T. Tsuji, and T. Watanabe, "Robots assembling machines: learning from the world robot summit 2018 assembly challenge," *Advanced Robotics*, vol. 34, no. 7-8, pp. 408–421, 2020.

[4] J. C. Triyonoputro, W. Wan, and K. Harada, "Quickly inserting pegs into uncertain holes using multi-view images and deep network trained on synthetic data," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 5792–5799.

[5] K. Almaghout, R. A. Boby, M. Othman, A. Shaarawy, and A. Klimchik, "Robotic pick and assembly using deep learning and hybrid vision/force control," in *2021 International Conference" Nonlinearity, Information and Robotics"(NIR)*. IEEE, 2021, pp. 1–6.

[6] Y. Litvak, A. Biess, and A. Bar-Hillel, "Learning pose estimation for high-precision robotic assembly using simulated depth images," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3521–3527.

[7] F. Furrer, M. Wermelinger, H. Yoshida, F. Gramazio, M. Kohler, R. Siegwart, and M. Hutter, "Autonomous robotic stone stacking with online next best object target pose planning," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 2350–2356.

[8] H. Fakhurldeen, F. Dailami, and A. G. Pipe, "Cara system architecture-a click and assemble robotic assembly system," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 5830–5836.

[9] Y. Zhu, Z. Wang, J. Merel, A. Rusu, T. Erez, S. Cabi, S. Tunyasuvunakool, J. KramÃ¡r, R. Hadsell, N. de Freitas, and N. Heess, "Reinforcement and imitation learning for diverse visuomotor skills," in *Proceedings of Robotics: Science and Systems*, Pittsburgh, Pennsylvania, June 2018.

[10] A. X. Lee, C. M. Devin, Y. Zhou, T. Lampe, K. Bousmalis, J. T. Springenberg, A. Byravan, A. Abdolmaleki, N. Gileadi, D. Khosid *et al.*, "Beyond pick-and-place: Tackling robotic stacking of diverse shapes," in *5th Annual Conference on Robot Learning*, 2021.

[11] S. Stevšić, S. Christen, and O. Hilliges, "Learning to assemble: Estimating 6d poses for robotic object-object manipulation," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1159–1166, 2020.

[12] A. S. Morgan, B. Wen, J. Liang, A. Boularias, A. M. Dollar, and K. Bekris, "Vision-driven Compliant Manipulation for Reliable, High-Precision Assembly Tasks," *Robotics: Science and Systems XVII*, 2021.

[13] G. Wang, F. Manhardt, F. Tombari, and X. Ji, "Gdr-net: Geometry-guided direct regression network for monocular 6d object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 611–16 621.

[14] M. Rad and V. Lepetit, "Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 3828–3836.

[15] B. Tekin, S. N. Sinha, and P. Fua, "Real-time seamless single shot 6d object pose prediction," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 292–301.

[16] Y. Hu, J. Hugonot, P. Fua, and M. Salzmann, "Segmentation-driven 6d object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3385–3394.

[17] S. Peng, Y. Liu, Q. Huang, X. Zhou, and H. Bao, "Pvnet: Pixel-wise voting network for 6dof pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4561–4570.

[18] Z. Li, G. Wang, and X. Ji, "Cdpn: Coordinates-based disentangled pose network for real-time rgb-based 6-dof object pose estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7678–7687.

[19] S. Zakharov, I. Shugurov, and S. Ilic, "Dpod: 6d pose object detector and refiner," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 1941–1950.

[20] T. Hodan, D. Barath, and J. Matas, "Epos: Estimating 6d pose of objects with symmetries," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 703–11 712.

[21] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes," in *Robotics: Science and Systems (RSS)*, 2018.

[22] Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox, "Deepim: Deep iterative matching for 6d pose estimation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 683–698.

[23] Y. Labbé, J. Carpentier, M. Aubry, and J. Sivic, "Cosypose: Consistent multi-view multi-object 6d pose estimation," in *European Conference on Computer Vision*. Springer, 2020, pp. 574–591.

[24] Y. Hu, P. Fua, W. Wang, and M. Salzmann, "Single-stage 6d object pose estimation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2930–2939.

[25] W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab, "Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1521–1529.

[26] M. Sundermeyer, Z.-C. Marton, M. Durner, M. Brucker, and R. Triebel, "Implicit 3d orientation learning for 6d object detection from rgb images," in *Proceedings of the european conference on computer vision (ECCV)*, 2018, pp. 699–715.

[27] F. Manhardt, D. M. Arroyo, C. Rupprecht, B. Busam, T. Birdal, N. Navab, and F. Tombari, "Explaining the ambiguity of object detection and 6d pose from visual data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6841–6850.

[28] T. Hodaň, M. Sundermeyer, B. Drost, Y. Labbé, E. Brachmann, F. Michel, C. Rother, and J. Matas, "Bop challenge 2020 on 6d object localization," in *European Conference on Computer Vision*. Springer, 2020, pp. 577–594.

[29] M. Denninger, M. Sundermeyer, D. Winkelbauer, D. Olefir, T. Hodan, Y. Zidan, M. Elbadrawy, M. Knauer, H. Katam, and A. Lodhi, "Blenderproc: Reducing the reality gap with photorealistic rendering," in *International Conference on Robotics: Sciene and Systems, RSS 2020*, 2020.

[30] V. Loing, R. Marlet, and M. Aubry, "Virtual training for a real application: Accurate object-robot relative localization without calibration," *International Journal of Computer Vision*, vol. 126, no. 9, pp. 1045–1060, 2018.

[31] B. Wen, C. Mitash, B. Ren, and K. E. Bekris, "se (3)-tracknet: Data-driven 6d pose tracking by calibrating image residuals in synthetic domains," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 10 367–10 373.

[32] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[33] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "Sun database: Large-scale scene recognition from abbey to zoo," in *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE, 2010, pp. 3485–3492.

[34] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the continuity of rotation representations in neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5745–5753.

[35] J. Pan, S. Chitta, and D. Manocha, "Fcl: A general purpose library for collision and proximity queries," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 3859–3866.

[36] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[37] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes," in *Asian conference on computer vision*. Springer, 2012, pp. 548–562.

[38] T. Hodaň, J. Matas, and Š. Obdržálek, "On evaluation of 6d object pose estimation," in *European Conference on Computer Vision*. Springer, 2016, pp. 606–619.

[39] C. Hennersperger, B. Fuerst, S. Virga, O. Zettinig, B. Frisch, T. Neff, and N. Navab, "Towards mri-based autonomous robotic us acquisitions: a first feasibility study," *IEEE transactions on medical imaging*, vol. 36, no. 2, pp. 538–548, 2017.