GelSight Simulation for Sim2Real Learning

Daniel Fernandes Gomes[1], Achu Wilson[2] and Shan Luo[1]

Abstract*—Grasping and manipulation of objects are common both in domestic and industrial environments. Recent works exploring learning based solutions have shown promising results on robotic manipulation tasks. One efficient approach for training such learning agents is to train them within a simulated environment, followed by their deployment on real robots (*Sim2Real*). Most current works leverage camera vision to facilitate such manipulation tasks. However, camera vision might be significantly occluded by robot hands during the manipulation. Tactile sensing is another important sensing modality that offers complementary information to vision and can make up the information loss caused by the occlusion. However, the use of tactile sensing is restricted in the *Sim2Real* research due to no simulated tactile sensors available in the current simulation platforms. To mitigate the gap, we introduce a novel approach for simulating a GelSight tactile sensor in the commonly used Gazebo simulator. Similar to the real GelSight sensor, the simulated sensor can produce high-resolution images by an optical sensor from the interaction between the touched object and an opaque soft membrane. It can indirectly sense forces, geometry, texture and other properties of the object and enables the research of *Sim2Real* learning with tactile sensing. Preliminary experiment results have shown that the simulated sensor could generate realistic outputs similar to ones captured by a real GelSight sensor.

I. Introduction and related work

The manipulation of objects is prevalent in various applications, e.g., grasping tools, untangling a cable and folding a piece of garment. Recent works on using robot platforms for the manipulation tasks have shown inspiring results, especially ones using Deep Learning based approaches [1]. Such methods usually require a large number of training iterations with many robotic arms being used in parallel, to learn the necessary manipulation policies, which would be costly to replicate. More recently, *Sim2Real* learning approaches have been proposed to mitigate this problem [2]: The agent is trained firstly in a simulated scene and then the learned policy is deployed on a similar real robot and environment. An example of Sim2Real learning is [3], in which the task is to grasp, fold and hang a towel using a robot arm. Camera vision is the mostly commonly used sensing modality to facilitate the manipulation tasks in these *Sim2Real* works. However, the visual perception can be easily occluded by either the robot hands or other objects, as observed in [3] where the main failures arise from weak or incorrectly centered grasps.

In addition to vision, tactile sensing can also be used to facilitate perception and grasping tasks, and can make up the information loss of vision due to the occlusion, demonstrated on the real robot setups [4]. However, limited works have been done on simulating tactile sensors, causing difficulties for *Sim2Real* learning with tactile sensing. Simulation models were built for piezoresistive tactile arrays (tactile sensors from Weiss Robotics) in [5], [6]. However, to the best of our knowledge, there has been no work on creating simulation models for optical tactile sensors [7], [8], that are of high-resolution, and sensitive to sense fine textures and shapes. In this paper, we develop a simulated sensor model for the GelSight sensor [7] that is one of the mostly widely used optical tactile sensors. Core ideas from our work can also be used to construct the simulation of other optical tactile sensors, like the TacTip [8]. The developed simulated sensor, shown in Figure 1, enables the *Sim2Real* learning with tactile sensing, and also avoids potential damage to the delicate soft elastomer of such tactile sensors due to frequent use in real experiments.

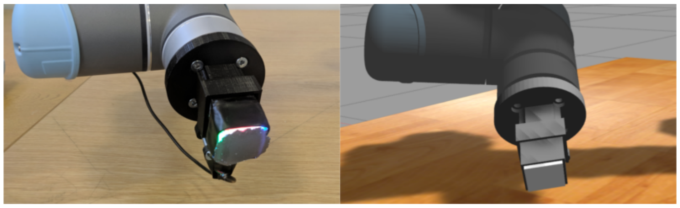

Fig. 1: **Our experimental setup**. Views of our real (left) and virtual (right) robot platform with a GelSight sensor mounted onto the end-effector of a UR5 robotic arm.

II. Simulation of a GelSight sensor

The real GelSight consists of a membrane that is internally illuminated by 4 opposite multi-color LEDs, producing a high-resolution tactile image that is captured by a webcam installed in the sensor core. However, in the current simulators used in robotic applications, such as Gazebo[1], PyBullet[2] and MuJoCo[3], the simulation of soft material deformations is not supported, or only with low resolution and accuracy. Thus, it would result in no or low quality tactile images when attempting to capture the external forces applied to the virtual membrane, with a 2D camera positioned in the sensor core. To overcome the limitations, we obtain the measured object surface directly from the simulation and generate the tactile images by taking it as an inverse problem of the Surface Reconstruction, which is described by:

$$I(x,y) = R(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}) \tag{1}$$

where I denotes the output image, f is the elastomer surface function, and R is the reflectance function that models both

[1] smARTLab, Department of Computer Science, University of Liverpool, Liverpool L69 3BX, U.K. [2] Perceptual Science Group, Computer Science and Artificial Intelligence Lab, Massachusetts Institute of Technology, Cambridge, MA, USA. Emails:{danfergo, shan.luo}@liverpool.ac.uk, achuwils@mit.edu.

We thank the EPSRC Doctoral Training Programme for supporting Daniel's work, and the Department of Computer Science from the University of Liverpool for funding his conference participation.

[1] http://gazebosim.org [2] https://pybullet.org [3] http://www.mujoco.org

the lighting conditions and the surface material reflectance properties.

The surface functions f can be obtained using a variety of approaches. The solution we propose here is by casting orthogonal rays to the sensor base plane and intersecting them with the closest object, which is similar to the behaviour of a depth camera. In our case, using the Gazebo simulator, we are able to add a depth camera to the simulation, to obtain the desired surface map. From the obtained depth map, we limit all depths to a maximum of 3mm above the simulated sensor shell, i.e., the elastomer outside surface, as shown in Figure 1, resulting in a depth map that captures the object part penetrating the elastomer. To obtain $\frac{\partial f}{\partial x}$ and $\frac{\partial f}{\partial y}$ we perform 2D convolutions over the the depth map matrix, as discrete centered derivatives using 3×1 and 1×3 kernels respectively. We normalize these values by dividing them by $2r$, where r is a pixel-to-meter ratio. To obtain r we put a cube of side size 5 mm against the virtual sensor, and measure the distance in pixels between the first and last in-contact pixels in a same row. Then, to implement the reflectance function R in Equation (**??**), we follow Phong's reflection model:

$$I(x,y) = k_a i_a + \sum_{m \in L} (k_d(\hat{L}_m \cdot \hat{N}) i_{m,d} + k_s(\hat{R}_m \cdot \hat{V})^\alpha i_{m,s}) \tag{2}$$

$$\hat{R}_m = 2(\hat{L}_m \cdot \hat{N})\hat{N} - \hat{L}_m \tag{3}$$

where $\hat{N}$ represents the surface normal vector, given by $(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, -1)$; L represents the set of 4 LEDs (white, blue, red and green, as in [7]); $\hat{L}_m$ the direction of the LED light, i.e., $(-1,0,0)$, $(0,-1,0)$, $(1,0,0)$ and $(0,1,0)$. By conducting extensive experiments, we set the shininess constant α to 5, the diffuse constants k_d to 0.5, and the specular constant k_s to 0.15. Instead of considering a solid k_a ambient light, we use an empty background image captured using the real sensor. Furthermore, we add a black darkening mask, dependant on the pixel penetration depth, and tilt the LEDs slightly towards the elastomer, i.e., we set the $\hat{L}_m$ 3rd component to 0.15. This results in better separation of touch and non-touch areas. Finally, because the depth camera measures the object in contact with the sensor and not the elastomer surface, we need emulate elastomer displacement around the touching area. For instance, a flat sharp surface touching the elastomer would result in (almost) no gradients and no tactile information observed. To this end, we produce a *displacement map* by blurring the depth map 5 times with a 15×15 mean kernel. We then select the non-touching pixels ($> 3mm$) from the *displacement map* and subtract them to the surface depth map.

III. Evaluation

We measured 4 elementary surfaces, shown in Figure 2, with our real GelSight sensor: a round smooth surface, a large flat displacement, a thin protrusion and a prism corner. We then obtained correspondent pairs by placing virtual objects in contact with our virtual sensor. We can understand the two main inaccuracies resulting from our method. 1) The imprecise shadowing particularly visible near the edge of the real sphere, that we mitigate with the darkening mask. 2) And, the less realistic elastomer displacement generated with the blur effect, more prominent in the real samples. We can also notice the micro details in the real objects that don't exist in the virtual correspondences, e.g., the prism bezel present on the last sample. Nonetheless, by using our method we should be able to generate tactile images for any complex surface/texture that can be displayed in the simulation.

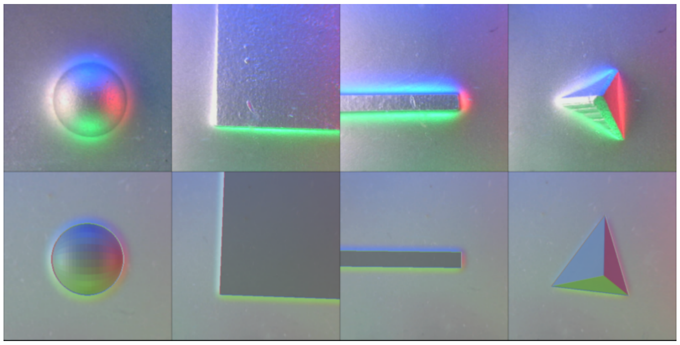

Fig. 2: **Real vs. virtual tactile images.** Samples captured by a real GelSight sensor (top) and generated virtual correspondences using our method (bottom).

IV. Conclusion and future work

In this paper we introduce a novel way of generating tactile images from a simulated GelSight sensor, to enable *Sim2Real* learning with tactile sensing. As the proposed method only depends on the surface function, it can be implemented in any current popular robotics simulator. Our proposed method will be used to augment real and/or generate entirely new synthetic datasets. Further improvements to our method can also be considered, such as leveraging contact mechanics theory to model the elastomer displacement. Quantitative analysis of the generated outputs [9] will also be conducted to further validate our proposed approach.

References

[1] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," *IJRR*, vol. 37, no. 4-5, pp. 421–436, 2018.

[2] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," in *ICRA*, 2018.

[3] J. Matas, S. James, and A. J. Davison, "Sim-to-real reinforcement learning for deformable object manipulation," in *CoRL*, 2018.

[4] R. Calandra, A. Owens, D. Jayaraman, J. Lin, W. Yuan, J. Malik, E. H. Adelson, and S. Levine, "More than a feeling: Learning to grasp and regrasp using vision and touch," in *CoRL*, 2018.

[5] S. Moisio, B. Len, P. Korkealaakso, and A. Morales, "Simulation of tactile sensors using soft contacts for robot grasping applications," in *ICRA*, 2012, pp. 5037–5043.

[6] J. Bimbo, S. Luo, K. Althoefer, and H. Liu, "In-hand object pose estimation using covariance-based tactile to geometry matching," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 570–577, 2016.

[7] W. Yuan, S. Dong, and E. Adelson, "Gelsight: High-resolution robot tactile sensors for estimating geometry and force," *Sensors*, vol. 17, no. 12, p. 2762, 2017.

[8] B. Ward-Cherrier, N. Pestell, L. Cramphorn, B. Winstone, M. E. Giannaccini, J. Rossiter, and N. F. Lepora, "The tactip family: Soft optical tactile sensors with 3d-printed biomimetic morphologies," *Soft robotics*, vol. 5, no. 2, pp. 216–227, 2018.

[9] J.-T. Lee, D. Bollegala, and S. Luo, ""Touching to See" and "Seeing to Feel": Robotic cross-modal sensorydata generation for visual-tactile perception," in *ICRA*, 2019.