Bidirectional Sim-to-Real Transfer for GelSight Tactile Sensors With CycleGAN

Weihang Chen, Yuan Xu, Zhenyang Chen, Peiyu Zeng, Renjun Dang, Rui Chen, and Jing Xu, *Member, IEEE*

***Abstract*—GelSight optical tactile sensors have high-resolution and low-cost advantages and have witnessed growing adoption in various contact-rich robotic applications. Sim2Real for GelSight sensors can reduce the time cost and sensor damage during data collection and is crucial for learning-based tactile perception and control. However, it remains difficult for existing simulation methods to resemble the complex and non-ideal light transmission of real sensors. In this letter, we propose to narrow the gap between simulation and real world using CycleGAN. Due to the bidirectional generators of CycleGAN, the proposed method can not only generate more realistic simulated tactile images, but also improve the deformation measurement accuracy of real sensors by transferring them to simulation domain. Experiments on a public dataset and our own GelSight sensors have validated the effectiveness of our method. The materials related to this letter are available at https://github.com/RVSATHU/GelSight-Sim2Real.**

***Index Terms*—Deep learning methods, force and tactile sensing, transfer learning.**

I. Introduction

TCTILE sensing is essential in robot's interaction with objects and the environment [1], as it directly provides contact states and offers complementary information aside from visual sensors. Therefore, the adoption of tactile sensors have become increasingly popular in various robotic applications, e.g., cable manipulation [2], peg-in-hole insertion [3].

Learning-based tactile perception and control methods have shown great success compared with classical counterparts, since they can extract task-relevant representations in a data-driven fashion [3]–[5]. However, collecting data in real environment with real robots and tactile sensors can be time-consuming and damaging to robots and sensors. One common strategy to address the data collection issue is to first train robots in a simulated environment, and then transfer it into realistic setups (*Sim2Real*).

In this letter, we focus on Sim2Real of GelSight optical tactile sensors [6], which employ CMOS cameras to capture the deformation of the elastic membrane. Then, the contact states are derived from the surface deformation. GelSight sensors have the advantage of high resolution and low cost, thus being increasingly adopted in robotic manipulation tasks.

For GelSight sensors, there are two main challenges in simulation: a) the dynamic deformation of hyperelastic material is hard to simulate, and b) the complex and non-ideal illumination condition and light transmission in real sensors makes it hard to tune simulation parameters. Here in this letter, we mainly focus on the second challenge: how to generate optically realistic tactile images from simulation. Several simulation methods based on traditional rasterization method have been proposed, but the similarity between Sim and Real is limited [7], [8]. On the contrast, differentiable rendering based on the path-tracing algorithm is also implemented [9], but the computational cost is high.

From another aspect, the non-ideal illumination condition not only makes accurate simulation difficult, but also decreases the depth reconstruction accuracy, because the assumptions of photometric stereo method [10] do not hold. Although end-to-end pixelwise neural network can mitigate this issue, a huge amount of aligned data is needed [11].

Considering the aforementioned problems, we aim at narrowing the sim-real gap bidirectionally in an unsupervised manner, to solve the simulation and depth reconstruction problems simultaneously (See Fig. 1). In the context of optical simulation, the differences between Real and Sim for GelSight mainly lie in color and light distributions [7]–[9]. Therefore, we get inspiration from the image-style-transfer task successfully completed by Cycle-Generative-Adversarial-Network (CycleGAN) [12]. Following this *Domain Adaptation* approach, by training CycleGAN with unpaired data collected from simulation and real world, we can enhance the simulated images to better mimic the real ones. Moreover, thanks to the bidirectional generators of CycleGAN, we can reduce the effect of non-ideal illumination in real GelSight sensors by transferring the real images to simulated ones (*Real2Sim*), and improve the deformation measurement accuracy. We evaluated our method on a public tactile Sim2Real dataset and our own GelSight sensors. Experimental results show that our method outperforms existing Sim2Real methods

Manuscript received November 4, 2021; accepted March 25, 2022. Date of publication April 13, 2022; date of current version April 25, 2022. This letter was recommended for publication by Associate Editor R. Liu and Editor D. Popa upon evaluation of the reviewers' comments. This work was supported in part by the National Natural Science Foundation of China under Grants 51935010 and 62173198, in part by the Natural Science Foundation of Beijing Municipality under Grant L192001, in part by the China National Postdoctoral Program for Innovative Talents under Grant BX2021152, and in part by the State Key Laboratory of Tribology of China under Grant SKLT2022C17. *(Weihang Chen, Yuan Xu, and Zhenyang Chen contributed equally to this work.) (Corresponding author: Jing Xu.)*

Weihang Chen, Peiyu Zeng, Renjun Dang, Rui Chen, and Jing Xu are with the Department of Mechanical Engineering, Tsinghua University, Beijing 100084, China (e-mail: chen-wh18@mails.tsinghua.edu.cn; zengyuqi0307@gmail.com; drjamesown@gmail.com; chenruithu@mail.tsinghua.edu.cn; jingxu@mail.tsinghua.edu.cn).

Yuan Xu and Zhenyang Chen are with the Department of Mechanical Engineering, Southern University of Science and Technology, Shenzhen 518055, China (e-mail: xuyuan2966@gmail.com; 11811917@mail.sustech.edu.cn).

Digital Object Identifier 10.1109/LRA.2022.3167064

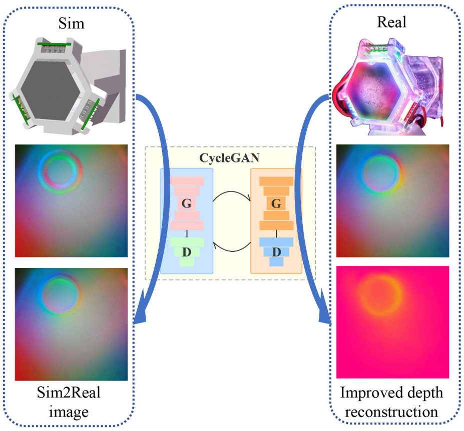


Fig. 1. Overview of the proposed method. On the Sim2Real side, the proposed method produces realistic tactile image; on the Real2Sim side, the depth reconstruction quality is improved.

in the object-classification task; besides, the Real2Sim approach can improve the deformation measurement accuracy for real GelSight sensors by 30%.

The remainder of this letter is organized as follows. Firstly, Section II introduces the related work. Then, our methodology of bidirectional Sim-Real transfer is presented in Section III. Next, in Section IV, experiments on a public dataset are conducted to test the Sim2Real transferability and generalizability of the proposed method; in Section V, depth reconstruction experiments are performed on our self-made sensor. Finally, Section VI concludes this letter.

II. Related Work

A. Simulation of GelSight Sensors

One challenge of simulating tactile sensors is the complex deformation of the elastomer surface. There are mainly two approaches to tackling this challenge: physics-based method or geometry-based method with post-processing. Finite element method (FEM) is commonly used as a physics-based method, with relatively better accuracy, but it relies on a massive amount of computation, which may affect simulation efficiency [13], [14]. Geometry-based methods are usually based on the intersection of object meshes, and then filters are applied to smooth the contact edge [8], [9]. In this work, we adopt the geometry-based method for higher simulation efficiency.

With the deformation of the sensor surface, one can develop different simulation methods according to the specific kind of sensing principle. Among the optical tactile sensors, GelSight is more complicated to simulate because of the complex light system.

There are currently two ways to generate synthetic GelSight images. The first is from the rasterization algorithm in computer graphics, including OpenGL renderer [7] and Phong's shading model [8]. These methods can reach high throughput, but is less realistic because of the simplifications made by the shading model. The other approach [9] utilizes the path-tracing algorithm, where multiple bounces of light are considered, resulting in more realistic synthetic images. The authors also implement differentiable rendering, and therefore the parameters of simulation can be efficiently optimized. However, the realistic effect comes at the cost of large computation consumption. A recent work proposes an example-based simulation method [15], where a polynomial look-up table is used to model the optical response. This method requires less than 100 calibration examples, and its performance reaches the state of the art.

In this work, we aim to narrow the Sim-Real gap using a data-driven approach. We use the traditional rasterization method to efficiently produce simulated images, and then train a transformation model from the unpaired Sim and Real dataset. Details will be presented in Section III.

B. Sim2Real Transfer for GelSight Sensors

Simulation method varies as the tactile sensing principle changes, and so does the Sim2Real method. For the *TacTip* sensor, whose image features are not rich, researchers ran simulations under random-dynamics environments to better transfer to reality [16]. For an optical-flow-based sensor, researchers built a sensor-dependent calibration layer to map between real images and simulated features [17]. The method is concise and has good generalizability, but it is not suitable for GelSight-like sensors. For GelSight-like sensors, Fernandes *et al.* [8] proposed to add random texture noises in the depth image before rendering the RGB image, and increased the Sim2Real classification accuracy by over 30 percent. This method can be categorized into *Domain Randomization*.

To the best of our knowledge, Sim2Real for GelSight-like sensors has not been extensively studied. Aside from the *Domain Randomization* method introduced above, we adopt *Domain Adaptation* method by using CycleGAN to enhance simulated images. In Section IV, comparisons will be made between the texture-based method and the proposed method.

C. GAN for Domain Adaptation

Generative Adversarial Networks (GANs) [18] construct a learning pattern where the adversarial loss will force the generator to produce images that are indistinguishable by the discriminator, making it suitable for tasks like image generation. CycleGAN [12] is a popular variant which utilizes two sets of GANs, where the images generated by GAN A will be put into the other GAN B to test the invertibility. CycleGAN has demonstrated a significant effect on style-transfer, super-resolution and image-generation tasks on unpaired datasets. CyCADA [19] introduces additional semantic losses into CycleGAN, which is suitable for domain adaptation tasks with distinct categories.

Many robotic applications have used GANs for Domain Adaptation. In [20], RL-CycleGAN is proposed, which is a reinforcement-learning-aware Sim2Real method applicable in robot grasping tasks. In [21], Real2Sim transfer is performed for visual control, by translating the real images back to the synthetic domain during policy deployment. In [22], Sim2Real is used to bridge the dynamics domain gap in robot navigation, while Real2Sim enabled by CycleGAN is used to bridge the visual domain gap.

Very related to our method is the concurrent work of Church *et al.*, where Real2Sim for the optical tactile sensor

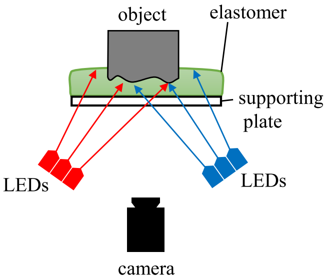


Fig. 2. Structure of GelSight tactile sensor.

TacTip is proposed [5]. In their work, a pix-to-pix GAN [23] was used to map *TacTip* tactile images to simulated depth maps. Their work differs from ours in the following aspects. Firstly, the concurrent work focuses on contact simulation and the input images are not rich in features. In this work, we preserve the detailed geometry of tactile images; we realize this by increasing the optical similarity between simulated and real images. Secondly, in their work, a supervised pix-to-pix translation network was used. Therefore, the simulated image and real image are required to be strictly paired. Consequently, the data collection procedure needs to be carefully designed; accurate relative pose between the object and the sensor should be guaranteed. Contrarily in this letter, only unpaired data are needed, resulting in minimal manual effort.

III. Methodology

In this section, the principle of GelSight is firstly introduced. Then, we introduce the adopted simulation method. Next, to narrow the Sim2Real gap, we introduce the *Domain Adaptation* method based on CycleGAN, through which the real and simulated images can be transferred to each other.

A. Depth Reconstruction Principle of GelSight

As can be seen in Fig. 2, in GelSight, the elastomer surface deforms as an object is pressed against it, causing the color distribution to change. Therefore, with the image captured by the camera, the surface shape can be solved, and further, the force distribution can be obtained using the constitutive relation of the elastomer material. It can be naturally concluded that, to exploit the sense of touch with GelSight, the depth map of sensor surface should be solved accurately.

The depth reconstruction principle of GelSight sensors was initially introduced in [10], where a modified photometric stereo method was proposed. As stated in [10], when some certain assumptions are made, the RGB intensity at a point (x, y) in the image is related to its surface normal:

$$\boldsymbol{I}(x,y) = \boldsymbol{R}\left(\frac{\partial h}{x}, \frac{\partial h}{y}\right), \tag{1}$$

where $h = f(x, y)$ is the surface height map and $\boldsymbol{R}(\cdot)$ is the reflectance function. Since there are 3 intensity values (RGB) and 2 unknown gradients, this equation can be over-constrained under appropriate conditions. After a calibration procedure, one can build a lookup table (LUT) through which the surface gradient $(\frac{\partial h}{x}, \frac{\partial h}{y})$ can be determined by the RGB intensities. Then, the surface depth map can be obtained by solving the Poisson Equation [6].

In this letter, we also follow this procedure to calibrate the sensor and calculate depth maps by referring to the open-source code.[1] However, it should be noted that, the above method assumes uniformly distributed illumination and surface reflection, which is rare in reality because of shadows and internal reflections.

B. Simulation Method

Since a data-driven approach for bidirectional Sim-Real transfer is adopted, it is less necessary for the simulation method to have high fidelity. Instead, we require the simulation method to be computationally efficient and make it strictly meet the requirements of the depth reconstruction algorithm introduced in III-A.

With that purpose, we synthesize our simulation method from existing rendering algorithms [8]. Specifically, we use *Tacto* [7] to acquire the depth image, then apply the *Difference of Gaussians (DoG)* method [8] to approximate the real elastomer deformation, and finally get the RGB tactile image using only the diffusion part of Phong's shading model:

$$\boldsymbol{I}_{Phong}(x,y) = \sum_{m \in L} k_d(\hat{\boldsymbol{L}}_m \cdot \hat{\boldsymbol{N}})\boldsymbol{i}_{m,d}, \tag{2}$$

where L is the set of light sources (i.e., LEDs), $\hat{\boldsymbol{L}}_m$ is the emission direction of a given light source m; $\hat{\boldsymbol{N}}$ is the surface normal; $\boldsymbol{i}_{m,d}$ is the intensity of the diffuse reflection of light source m; k_d is the reflectance property of the surface related to diffusion. We only include the diffusion part because the specular reflection part depends on pixel positions, which will cause errors when using the LUT method and is unnecessary in this ideal simulation. Then, to blend with the background acquired from the real sensor, the RGB intensity change caused by contact is added to the background:

$$\boldsymbol{I} = \boldsymbol{I}_{Phong} - \boldsymbol{I}_{Phong,original} + \boldsymbol{I}_{background}. \tag{3}$$

C. Domain Adaptation for GelSight With CycleGAN

We utilize the CycleGAN architecture proposed by Zhu *et al.* [12] to realize the invertible transfer between real and simulated tactile images. We believe in such methodology because the difference between real and simulated tactile images mainly includes color and illumination, while this type of difference is successfully tackled in several CycleGAN applications. Next, we first introduce the losses in CycleGAN training; then, the Sim2Real approach is introduced, which is followed by the Real2Sim approach for depth reconstruction. The complete procedure is shown in Fig. 3.

1) Losses in CycleGAN: In the proposed method, two pairs of generators and discriminators in CycleGAN are trained together with unpaired Sim and Real datasets, as shown in Fig. 3(a). Suppose the Sim2Real generator is G_{S2R}, and the corresponding

[1] https://github.com/mcubelab/gelslim

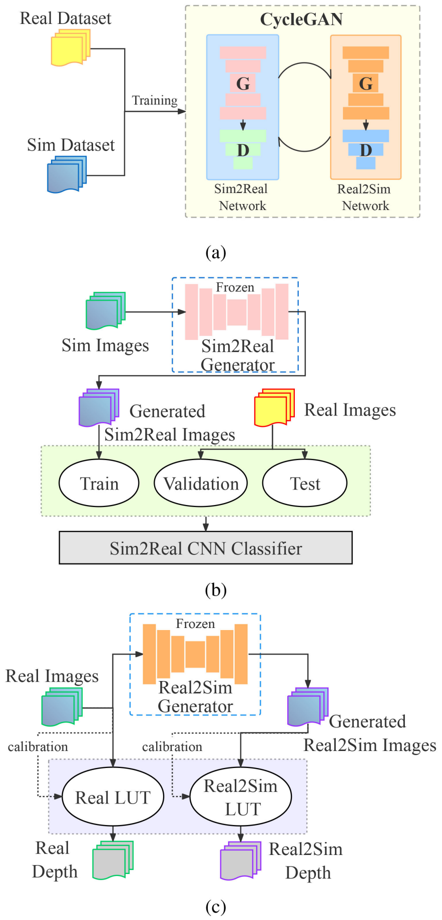


Fig. 3. Narrowing the Sim-Real gap using CycleGAN: (a) training of CycleGAN with unpaired datasets; (b) the Sim2Real procedure for classification; (c) the Real2Sim Procedure for depth reconstruction. As indicated by the blue dashed box in (b)(c), the parameters of generators are frozen after training.

discriminator is D_{S2R}; the Real2Sim generator is G_{R2S}, and the corresponding discriminator is D_{R2S}. The goal of G_{S2R} is to make $G_{S2R}(\boldsymbol{I}_{Sim})$ (the generated image from simulated image) resemble $\boldsymbol{I}_{Real}$ (the real tactile image) as close as possible, and vice versa. So, the adversarial loss [18] is utilized:

$$L_{adv}(G_{S2R}, D_{S2R}) = (D_{S2R}(G_{S2R}(\boldsymbol{I}_{Sim})) - 1)^2 + D_{S2R}(\boldsymbol{I}_{Real})^2. \tag{4}$$

On this basis, CycleGAN enforces cycle consistence by introducing an additional loss, which encourages the reconstructed image $G_{R2S}(G_{S2R}(\boldsymbol{I}_{Sim}))$ to be the same as its origin $\boldsymbol{I}_{Sim}$ [12]:

$$L_{cycle}(G_{R2S}, G_{S2R}) = \|G_{R2S}(G_{S2R}(\boldsymbol{I}_{Sim})) - \boldsymbol{I}_{Sim}\|_1 + \|G_{S2R}(G_{R2S}(\boldsymbol{I}_{Real})) - \boldsymbol{I}_{Real}\|_1. \tag{5}$$

Next, to preserve the color information, an identity loss is introduced [12]. The identity loss is aimed at making generators preserve the original image when it is already in the target domain, i.e.,

$$L_{identity}(G_{R2S}, G_{S2R}) = \|G_{R2S}(\boldsymbol{I}_{Sim}) - \boldsymbol{I}_{Sim}\|_1 + \|G_{S2R}(\boldsymbol{I}_{Real}) - \boldsymbol{I}_{Real}\|_1. \tag{6}$$

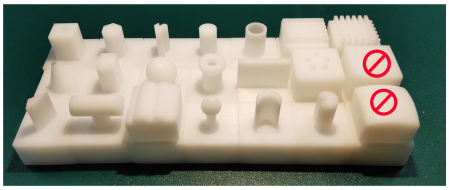


Fig. 4. The public object set includes 21 objects. We printed them using SLA technology, so their surfaces are smoother than the ones in the public dataset. Because the deformation caused by ‘curved surface’ and ‘flat slab’ is difficult to distinguish, we remove them from the original dataset.

Finally, the total loss is the weighted sum of the aforementioned losses:

$$\begin{aligned}\mathcal{L}(G_{R2S}, G_{S2R}, D_{R2S}, D_{S2R}) &= L_{adv}(G_{R2S}, D_{R2S}) \\ &+L_{adv}(G_{S2R}, D_{S2R}) \\ &+\lambda_{cycle} L_{cycle}(G_{R2S}, G_{S2R}) \\ &+\lambda_{identity} L_{identity}(G_{R2S}, G_{S2R})\end{aligned} \tag{7}$$

With these losses, a CycleGAN can be properly trained on the unpaired Sim and Real dataset. After training, we will get a Sim2Real generator G_{S2R}, as well as a Real2Sim generator G_{R2S}. Next, these two generators are used to perform bidirectional Sim-Real transfer, which is introduced as follows.

2) Sim2Real for Classification: As shown in Fig. 3(b), for Sim2Real, we input simulated images into the trained Sim2Real generator G_{S2R}, and the generated Sim2Real images can be used to train the classification model. The Sim2Real images are supposed to have similar characteristics as real images. The validation and test split of the classification model is from the real images. With the classification accuracy, the transferability of the proposed method can be proved.

3) Real2Sim for Depth Reconstruction: For the reconstruction of depth maps from tactile images, we propose a Real2Sim approach, whose procedure is illustrated in Fig. 3(c). After proper training, the Real2Sim generator G_{R2S} can be used to transfer real images to simulation-like ones. The generated Real2Sim image will capture both the real object geometry and the illumination condition in simulation, thus mitigating the non-ideal issues in reality that would decrease the depth reconstruction quality. With part of the real images and Real2Sim images, we perform calibration and get two LUTs. Then, we can reconstruct depth maps from the real images or the Real2Sim images. Comparison between the depth errors will be made for validation.

IV. Experiment on the Public Dataset

A. The Public Dataset and Data Preprocessing

Fernandes *et al.* [8] built a public dataset of tactile images from a GelSight 2014 sensor. In total, tactile images of 21 objects (Fig. 4) are collected with both real and simulated sensors. In the real tactile images, one can see rich textures in the contact area, because the objects were printed using fused deposition modeling (FDM) technology, different from ours in Fig. 4.

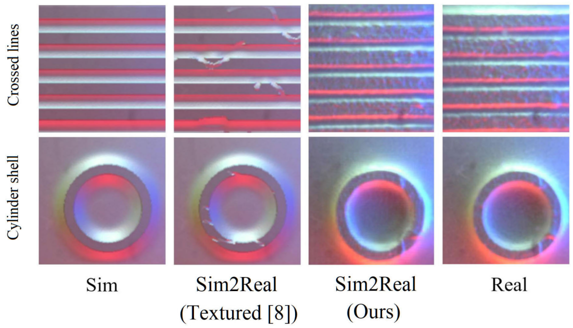


Fig. 5. Sim2Real performance of the proposed method. The image generated by our method looks similar to real images, better than the existing texture-augmentation method [8].

In this dataset, we found some poorly performing pictures in nearly every classification set in the simulation set, which does not correspond to the real pictures. The indentation of some simulated pictures is shallower than the corresponding real pictures, or even blank, which may be caused by the inconsistency of the pose correction between the simulation platform and the real platform. In practice, these shallow or blank simulated pictures can hardly be distinguished, and they may interfere with the training of the discriminators. Therefore, we preprocessed the data.

We used the Canny operator from OpenCV to perform edge extraction on all images, experimentally found suitable threshold parameters, filtered the blank and shallow images, and finally removed 649 simulated images. The corresponding real images with the same names are also removed. Because the indentation of classes “Flat slab” and “curved surface” is too shallow to distinguish, they are completely removed in the Canny screening, and we will perform classification network training on the remaining 19 classes of objects with 1429 simulated images and 1429 real images each.

B. *Data Generation for Training Classifier*

1) Data Generation Based on the Texture-Augmentation Method: In the previous research [8], a texture-augmentation method for simulated images was proposed: before Phong’s shading model is applied to obtain the RGB tactile image, several preset textures are randomly transformed and added to the simulated depth map. This method was reported to improve the Sim2Real classification accuracy considerably.

We implement this texture-augmentation method according to the open-source code and the generated Sim2Real images are shown in the second column of Fig. 5. These Sim2Real images are used to perform Sim2Real classification tests to compare with our method.

2) Data Generation Based on CycleGAN: First, we divide the Sim and Real dataset into training, validation and test sets according to the reference letter [8]. Then, we follow the method in Section III-C1 to train the CycleGAN. The unpaired training data include the training split of Sim data, the training and validation split of real data. After the training of CycleGAN is completed, we follow Section III-C2 to generate enhanced Sim data for training the classifier.

TABLE I
TRAINING, VALIDATION AND TEST SPLITS FOR SIM2REAL CONTRAST EXPERIMENT ON THE PUBLIC DATASET

Groups	**Training set**	**Val set**	**Test set**
Sim2Sim	Sim	Sim	Sim
Real2Real	Real	Real	Real
Sim2Real (Direct)	**Sim**	Real	Real
Sim2Real (Textured)	**Sim with textures**	Real	Real
Sim2Real (Ours)	**CycleGAN Generated**	Real	Real

TABLE II
SIM2REAL CLASSIFICATION ACCURACIES OF 10 EXPERIMENTS ON THE PUBLIC DATASET

Type	**Test Acc. (Std)**
Sim2Sim	99.28% (0.78%)
Real2Real	99.29% (0.67%)
Sim2Real (Direct)	84.82% (2.71%)
Sim2Real (Textured)	88.04% (2.44%)
Sim2Real (Ours)	**98.30% (0.27%)**

As shown in Fig. 5, compared with the original simulated images, CycleGAN effectively enhances the texture and shade distribution. Further comparison and analysis will be shown in classification tasks.

C. *Transfer-Learning for Shape Classification*

We firstly perform Sim2Sim and Real2Real experiments as shown in Table I. Then, for Sim2Real, there are three groups: ‘Direct,’ ‘Textured’ and ‘CycleGAN (Ours)’. As Table I shows, the validation and test sets of the three groups remain the same, while the only difference lies in the training sets. For the ‘Direct’ group, the training set is directly extracted from the original simulation data; in the other two comparison groups, the training sets are changed accordingly.

The classification network is basically the same as the one in [8], whose backbone is Resnet50. Before training, we adopted normalization for all images, and “earlystop” is used in the training process; detailed settings are provided in the independent appendix file. Each group is trained 10 times to get the average accuracy and standard deviation.

The classification accuracies and standard deviations are reported in Table II. In the table, the Sim2Sim and Real2Real results are firstly reported for reference. Both accuracies are close to 100%, which is reasonable since the objects mostly have distinct features. The test accuracies of the three Sim2Real experiments prove that adding random textures can slightly improve training performance; however, our method of using CycleGAN is advantageous over the texture-augmentation method [8].

D. *Generalizability of CycleGAN*

In the above experiment, CycleGAN has access to the images from all classes of objects during training. However, in reality, the contact shapes of the tactile sensor are too diverse to be completely preset. We expect that the CycleGAN trained with

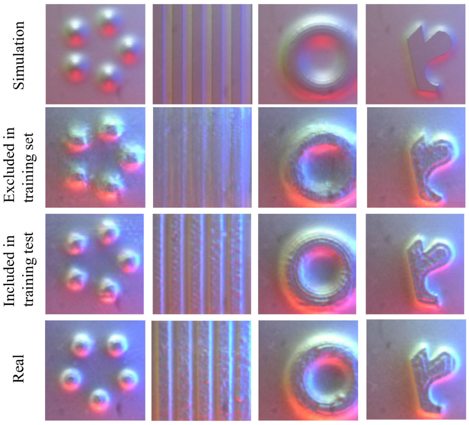


Fig. 6. Generalizability of CycleGAN. Four classes are tested: dots, straight lines, torus and random shape. The first row is original simulation pictures; the second row is $CycleGAN_{15}$ enhanced pictures (trained without these four object sets); the third row is a comparison to the second row, which is trained with $CycleGAN_{19}$ and the fourth row is real pictures. The generalized results capture the main difference between simulation and real pictures and look similar to the comparison.

a limited number of objects will perform well on new shapes. Therefore, we study the generalizability of CycleGAN.

We split the complete dataset for CycleGAN based on features. Now, the training set (for CycleGAN) only contains 15 basic shapes. The remaining 4 shapes are intentionally excluded due to their features: "dots" is a duplication of spheres; "parallel lines" is the rotated version of "cross lines"; "torus" is similar to "cylinder shell"; "random" is an arbitrarily generated shape. For simplicity, let $CycleGAN_{19}$ denote the CycleGAN trained on all 19 classes, and let $CycleGAN_{15}$ denote the one trained on 15 classes.

The visual results are shown in Fig. 6. For comparison, the $CycleGAN_{19}$ results are shown in the third row. The generalized results in the second row capture the main difference between simulation and real pictures in shades, light, and other details and look similar to real pictures, while texture details on edges are partly lost.

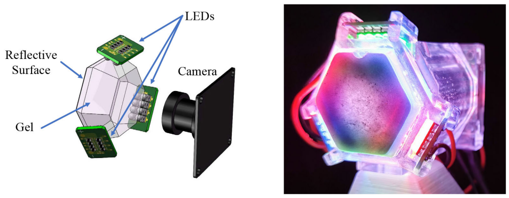
In order to quantitatively test the generalizability of CycleGAN, we designed new Sim2Real experiments with three control groups. For a fair comparison, the validation and test sets are kept consistent among the three groups. The only difference between the three groups is the training set, where each group contains four classes from different sources, but all the 15 remaining classes are generated by $CycleGAN_{19}$ (see Table III for details). As shown in Table III, CycleGAN has considerable generalizability even faced with unseen objects. When 4 classes in training data are replaced by generalized data, the test accuracy decreased by only 0.6%. This ability is valuable when tactile sensors are applied to complex tasks.

V. Experiment on Our Self-Made Sensor

To validate the proposed Real2Sim method for depth reconstruction, we conduct a series of experiments on our self-made sensor.

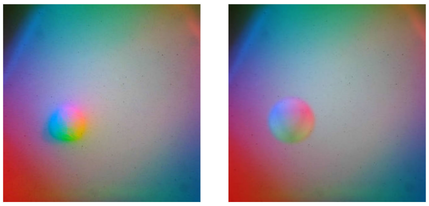
TABLE III
Training Conditions and Accuracies of CNN classifier for CycleGAN Generalizability Test

Training data for the classifier		Val Acc	Test Acc (Std)
common part: 15 classes from $CycleGAN_{19}$	4 classes from Simulation	83.33%	86.51% (3.71%)
	4 classes from $CycleGAN_{15}$ (generalized)	91.66%	97.68% (1.45%)
	4 classes from $CycleGAN_{19}$	97.91%	98.30% (0.27%)



(a) (b)

Fig. 7. The schematic (a) and photo (b) of our self-made sensor.



(a) (b)

Fig. 8. The real (a) and simulated (b) tactile image of our self-made sensor. The simulated image is blended with the real background image.

A. Self-Made Sensor and Simulation

Our tactile sensor (Fig. 7(b)) is developed based on GelSight 2017 [24]. Its structure is shown in Fig. 7(a). This letter mainly focuses on the RGB tactile image, so we do not laser-cut the markers. When collecting data, the camera is configured to have the resolution of 640 × 480, and the captured images are later cropped to 320 × 320 in the center, as shown in Fig. 8(a).

The simulation method for our self-made sensor has been illustrated in Section III-B. The simulation parameters are summarized in the independent appendix file, and the resulting simulated tactile image is shown in Fig. 8(b).

B. Data Collection for the Self-Made Sensor

The objects set is introduced in Section IV-A. The authors have released the 3D models of the objects [8], so we printed them using stereolithography (SLA) 3D printing technology, as shown in Fig. 4. Obviously, our objects have smoother surfaces. In the previous experiments, we only used 19 objects. Here for our self-made sensor, we further remove the 'cross lines' object, since it is nearly the same as 'parallel lines' (in the public dataset, their only difference is the rotation angle around Z-axis). For the remaining 18 objects, we manually press them against the

TABLE IV
DATASETS FOR CLASSIFICATION EXPERIMENTS ON SELF-MADE SENSOR

Experiment name	Training set	Validation set	Test set
Sim2Sim	Sim	Sim	Sim
Real2Real	Real	Real	Real
Sim2Real (Direct)	Sim	Real	Real
Sim2Real (Ours)	G_{S2R}(Sim)	Real	Real
Real2Sim (Direct)	Real	Sim	Sim
Real2Sim (Ours)	G_{R2S}(Real)	Sim	Sim

TABLE V
SIM2REAL AND REAL2SIM CLASSIFICATION ACCURACIES OF 10 EXPERIMENTS ON SELF-MADE SENSOR

Type	Validation (Std)	Test (Std)
Sim2Sim	99.49%(0.47%)	99.53%(0.52%)
Real2Real	98.98%(0.44%)	98.89(0.36%)
Sim2Real (Direct)	87.56%(1.43%)	86.88%(2.37%)
Sim2Real (Ours)	**97.91%(0.75%)**	**97.79%(0.63%)**
Real2Sim (Direct)	93.70%(2.34%)	93.25%(2.30%)
Real2Sim (Ours)	**97.06%(1.33%)**	**96.62%(1.63%)**

sensor with different poses, and finally 1200 tactile images for each object are collected.

To generate the simulation dataset, we use PyBullet to change the pose of the object, and force it to contact the sensor surface. For each object, 400 poses are randomly generated, and for each pose, 3 levels of force are exerted to the object. Therefore, 1200 tactile images for each object are collected in the simulation dataset. Since our method does not require paired simulated and real data, the sophisticated alignment and registration process is no longer needed, so the time cost of data generation and collection is decreased significantly.

C. CycleGAN Training on Self-Collected Datasets

For the collected simulation and real datasets, we split them into training, validation and test sets respectively, in the ratio of 7:1:2. Because we want to conduct both Sim2Real and Real2Sim experiments, we use the training sets of both datasets for training CycleGAN. The validation and test sets are used to verify whether CycleGAN improves the classification accuracy. In CycleGAN training, most of the training settings are the same as the default ones except that we change the number of layers of the discriminator from 3 to 2 to balance the generator and the discriminator.

D. Classification Accuracy on Self-Collected Datasets

In this section, the main purpose is to validate the bidirectional transferability of CycleGAN. With the assist of the trained CycleGAN, we perform a series of classification experiments. Table IV summarizes the data source of each dataset split for each experiment. The Real2Real and Sim2Sim experiments provide the theoretical upper bound for Sim2Real and Real2Sim performance. Each experiment was run 10 times to get more accurate statistical results.

The test accuracies in Table V show that both Real2Sim and Sim2Real performance are close to the upper bound with the assist of CycleGAN. The results agree with those on the public dataset, which further demonstrates the effectiveness of the proposed method.

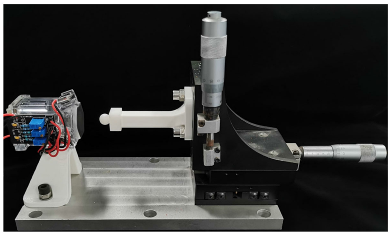


Fig. 9. The 3-axis translation stage provides ground truth depth information for the quantitative depth reconstruction experiment.

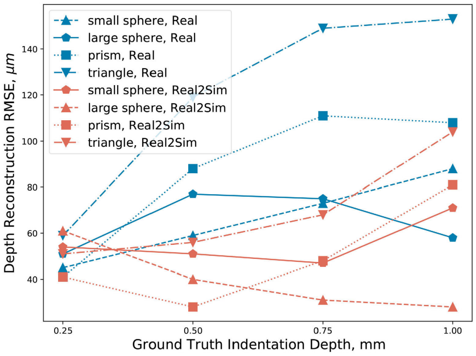
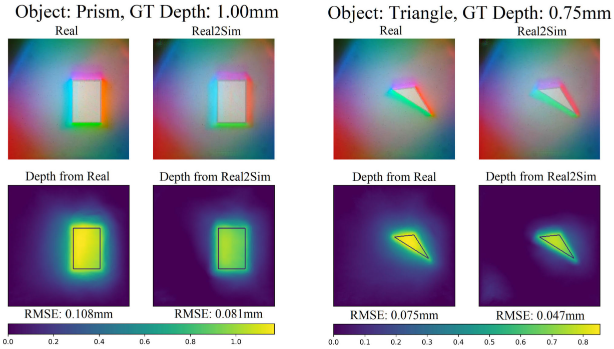


Fig. 10. The results of quantitative depth reconstruction experiments. Depth errors from the real images are colored blue, while those from Real2Sim images are colored red. In most cases, the proposed Real2Sim method significantly outperforms the original method, with an average of 30%.

E. Real2Sim Depth Reconstruction Result

To validate whether the proposed Real2Sim method can improve the depth reconstruction quality, quantitative experiments are performed. With the 3-axis translation stage shown in Fig. 9, tactile images of 4 objects (small sphere, large sphere, triangle and prism) with ground truth (GT) indentation depth are collected. The GT indentation depth has 4 values: 0.25 mm, 0.5 mm, 0.75 mm and 1.00 mm respectively. Following the procedure introduced in Section III-C3, two LUTs are generated for real images and generated Real2Sim images respectively. With the LUTs, depth maps are reconstructed from the real images and the generated Real2Sim images.

The depth errors against the GT depth are calculated, and the results are summarized in Fig. 10. Two of them are shown in detail in Fig. 11. Fig. 11 shows the original RGB image, the Real2Sim image, the depth reconstructed from original images, and the depth reconstructed from the Real2Sim image. The white masks in RGB images and the contour line in depth images show the area which is taken into account for calculating the root-mean-square error (RMSE) of depth. From the depth image, we can visually find that the depth images reconstructed from Real2Sim images tend to have a clearer background; this shows that the proposed method can partly eliminate the noise, interreflection or shadows. From Fig. 10, we find that in most cases, the Real2Sim method performs significantly better than the original real images. Only in some cases, when the



(a) (b)

Fig. 11. Two examples of the quantitative depth reconstruction experiment. The masks in RGB images and the contours in depth maps show the area which is taken into account for calculating depth error.

indentation is shallow, the Real2Sim method performs equally or worse. On average, the Real2Sim method can decrease the depth reconstruction error by 30%.

VI. Conclusion

In this letter, we propose to utilize CycleGAN to narrow the gap between simulation and reality for GelSight tactile sensors. On one hand, our method can generate realistic simulated tactile images for Sim2Real shape classification; on the other hand, it can significantly improve the depth reconstruction accuracy of real sensors by transforming them to simulation domain and mitigating the non-ideal illumination issues.

However, there are still some limitations in our work, that only the geometrical and optical properties of tactile sensors are taken into account and the physical property is neglected. For future work, we will study the physical simulation of tactile sensors and integrate the sensor simulation into robot simulation environment for Sim2Real control policy of contact-rich manipulation tasks.

Acknowledgment

The authors would like to thank Shaoxiong Wang from MIT, for providing technical advice on the manufacture of GelSight.

References

[1] S. N. Flesher *et al.*, "A brain-computer interface that evokes tactile sensations improves robotic arm control," *Science*, vol. 372, no. 6544, pp. 831–836, May 2021.

[2] Y. She, S. Wang, S. Dong, N. Sunil, A. Rodriguez, and E. Adelson, "Cable manipulation with a tactile-reactive gripper," *Int. J. Robot. Res.*, vol. 40, no. 12–14, pp. 1385–1401, Aug. 2021.

[3] S. Dong, D. K. Jha, D. Romeres, S. Kim, D. Nikovski, and A. Rodriguez, "Tactile-RL for insertion: Generalization to objects of unknown geometry," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 6437–6443.

[4] C. Wang, S. Wang, B. Romero, F. Veiga, and E. Adelson, "SwingBot: Learning physical features from in-hand tactile exploration for dynamic swing-up manipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5633–5640.

[5] A. Church, J. Lloyd, R. Hadsell, and N. F. Lepora, "Tactile sim-to-real policy transfer via real-to-sim image translation," in *Proc. 5th Conf. Robot Learn.*, 2022, pp. 1645–1654.

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

[7] S. Wang, M. M. Lambeta, P.-W. Chou, and R. Calandra, "TACTO: A fast, flexible, and open-source simulator for high-resolution vision-based tactile sensors," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 3930–3937, Apr. 2022.

[8] D. Fernandesgomes, P. Paoletti, and S. Luo, "Generation of GelSight tactile images for Sim2Real learning," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 4177–4184, Apr. 2021.

[9] A. Agarwal, T. Man, and W. Yuan, "Simulation of vision-based tactile sensors using physics based rendering," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 1–7.

[10] M. K. Johnson and E. H. Adelson, "Retrographic sensing for the measurement of surface texture and shape," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1070–1077.

[11] J. Li, S. Dong, and E. H. Adelson, "End-to-end pixelwise surface normal estimation with convolutional neural networks and shape reconstruction using GelSight sensor," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2018, pp. 1292–1297.

[12] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2242–2251.

[13] C. Sferrazza, A. Wahlsten, C. Trueeb, and R. D'Andrea, "Ground truth force distribution for learning-based tactile sensing: A finite element approach," *IEEE Access*, vol. 7, pp. 173438–173449, 2019.

[14] D. Ma, E. Donlon, S. Dong, and A. Rodriguez, "Dense tactile force estimation using GelSlim and inverse FEM," in *Proc. Int. Conf. Robot. Automat.*, 2019, pp. 5418–5424.

[15] Z. Si and W. Yuan, "Taxim: An example-based simulation model for Gelsight tactile sensors," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 2361–2368, Apr. 2022.

[16] Z. Ding, N. F. Lepora, and E. Johns, "Sim-to-real transfer for optical tactile sensing," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 1639–1645.

[17] C. Sferrazza and R. D'Andrea, "Sim-to-real for high-resolution optical tactile sensing: From images to three-dimensional contact force distributions," *Soft. Robot.*, Nov. 2021.

[18] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, vol. 27, pp. 2672–2680.

[19] J. Hoffman *et al.*, "CyCADA: Cycle consistent adversarial domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1989–1998.

[20] K. Rao, C. Harris, A. Irpan, S. Levine, J. Ibarz, and M. Khansari, "Rl-CycleGAN: Reinforcement learning aware simulation-to-real," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11157–11166.

[21] J. Zhang *et al.*, "VR-Goggles for robots: Real-to-sim domain adaptation for visual control," *IEEE Robot. Automat. Lett.*, vol. 4, no. 2, pp. 1148–1155, Apr. 2019.

[22] J. Truong, S. Chernova, and D. Batra, "Bi-directional domain adaptation for Sim2Real transfer of embodied navigation agents," *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 2634–2641, Apr. 2021.

[23] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5967–5976.

[24] S. Dong, W. Yuan, and E. H. Adelson, "Improved GelSight tactile sensor for measuring geometry and slip," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 137–144.