FULL PAPER

Grasp pose detection for deformable daily items by pix2stiffness estimation

Koshi Makihara[a,b*], Yukiyasu Domae[b], Ixchel G. Ramirez-Alpizar[b], Toshio Ueshiba[b] and Kensuke Harada[a,b]
[a]Graduate School of Engineering Science, Osaka University, Osaka, Japan; [b]National Institute of Advanced Industrial Science and Technology (AIST), Tokyo, Japan

ARTICLE HISTORY
Compiled May 9, 2022

ABSTRACT

While most of the existing works on grasp pose detection have assumed a complete 3D object model, this paper proposes a grasp pose detection method for unknown deformable objects, based on visual information. The proposed method is comprised two parts; (1) pix2stiffness estimation, which generates a stiffness map that indicates the object's stiffness for each pixel in an image using generative adversarial networks (GAN), and (2) grasp pose detection, which adapts a stiffness map to maximally reduce the object's deformation and avoid any possible damage. We demonstrate the validity of the proposed method and evaluate the estimation accuracy via simulations, and in a real environment. We also verify that the proposed approach can plan how to grasp an object using few 3D models of objects.

KEYWORDS
Grasping; Deformable objects; Stiffness estimation; Image translation; GAN

1. Introduction

Recently, robots are expected to be able to work in household environments, where there are several objects with different shapes, materials, mass, and other properties. In particular, robots have to grasp several types of deformable objects such as paper boxes including snacks. Although these objects are easy to grasp for a human, they are very difficult for robots that need to search for a grasp pose and control each finger's force. In most of the robotic grasping research, the main focus has been placed on rigid objects where the problem can be simplified by assuming a point contact model [1]. These methods exhibit optimal performance for several types of shapes, sizes, and other complexities. However, it remains difficult to successfully grasp deformable objects, because these rigid-body-based approaches do not consider the object's deformation.

Recent studies have proposed grasping methods for deformable objects, such as analysing grasp quality, considering the surface deformation by a contact wrench [2,3],

*Corresponding author. Email: makihara@hlab.sys.es.osaka-u.ac.jp

and effectively controlling a grasp wrench by sensing the contact wrench [4]. Also, some simulation-based methods that compute deformation using physics engines and then apply the grasp in the real-world [5–7] have been proposed. In these cases, they assumed that the grasping force is controllable by an electric unit, and can grasp various deformable objects using some force and/or tactile sensor. However, in most cases of industrial applications, a constant force is used to grasp objects (e.g. pneumatic gripper). Also, sometimes is impossible to force control the grasp of slippery objects even if the slip is detected. Therefore, there is a need to consider stiffness to avoid damages in the object without force control, which can be done by considering pre-grasping motion before any contact happens. In addition, many objects have an inhomogeneous stiffness like daily items, or are unknown, which makes it difficult, even though it might be possible, to apply force control. We propose a grasp pose detection method for unknown deformable objects using an image as input. This method comprises two parts; (1) stiffness estimation, which generates a "stiffness map" that indicates the object's stiffness for each pixel in an image using generative adversarial networks (GAN) [8] as an image translation method, and (2) grasp pose detection, which generates a grasp pose, thereby avoiding damage to the object by the robot's gripper, using the stiffness map and executing the grasping motion. The overview of the proposed method is shown in Figure 1. Our contributions are as follows:

(1) Our proposed pix2stiffness method can convert the image of objects to a map of the stiffness score for each pixel by adapting the pix2pix [9]. The image translation can be performed by training semi-automatically generated images using a physics simulator.
(2) By combining the obtained stiffness map with the grasp pose detection method, we can detect a grasp pose that can prevent damages to unknown (same category of a bottle or a box, but has different shape and size) deformable objects with fewer 3D object models used adopted in training the GAN.

This paper is organized as follows. First, we review related works in section 2. Next, we comprehensively present an overview of the proposed method (pix2stiffness and grasp pose detection) in section 3. In section 4, we evaluate these method in simulation. In section 5, we describe the experimental setup and compare the real-world results with the simulation results obtained in section 4. Finally, we conclude this study in section 6.

2. Related Works

There are several strategies available for planning how to grasp an object, which adopt different kinds of input data such as full 3D mesh model, RGB-D image, and 3D point cloud data. Using these data as input, almost all of the existing methods generate optimized grasp poses with different types of grasp quality measures to achieve successful grasps with high accuracy. In this section, we introduce some of the previously proposed methods for grasp quality evaluation and planning of rigid and non-rigid objects. We also present a few methods that estimate physical-information from some images, such as stiffness and depth.

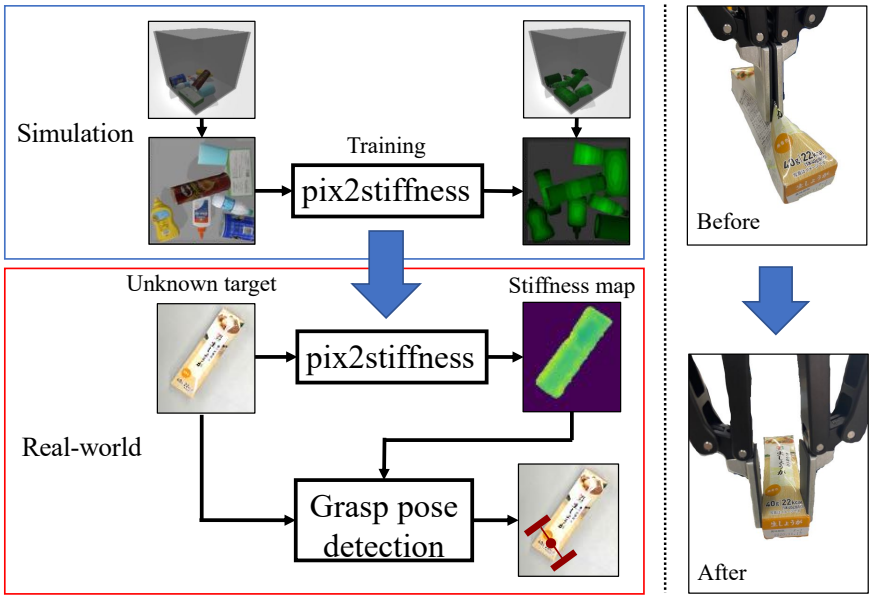


Figure 1. Overview of the proposed grasp pose detection method via stiffness estimation: we adopt an image as the input, and utilize it for image translation by pix2stiffness. After image translation, a stiffness map that indicates the object's stiffness score for each pixel is generated. Finally, grasp pose detection is executed using the map for the case of a 2-finger gripper (the red lines represent the grasp candidate).

2.1. Physics-based grasp quality evaluation

Grasp quality is often defined by considering some properties for rigid objects, such as disturbance resistance, stability, etc. [4]. These metrics often adopt grasping force and torque in the analysis of grasp candidates such as grasp wrench space (GWS) [10], and the expanded method for task completion such as task wrench space (TWS) [10]. For non-rigid objects, it is difficult to adopt these measures because they must consider the deformation generated by the grasping wrench. In recent works, a few methods have proposed the evaluation of grasp quality by analyzing the object's deformation [11,12]. In addition, the grasp quality evaluation is not only successful for grasping, but also for deformability and for preventing damage. Xu et al. [3] proposed quality metrics, considering task completion of deformable objects, including liquids. Using an elastic 3D model, the grasp quality is defined as the minimal grasping wrench, which reduces resistance according to Hooke's law. Also, there are some metrics considering contact dynamics [5] based on Finite Element Method (FEM) that can simulate grasping in a more real way than static analytical methods. However, these methods are difficult to apply to unknown objects (re-calculation is needed), and to objects with material properties like nonuniform stiffness (FEM assumes homogeneity).

2.2. Image-based grasp pose detection

Grasp planning often needs to recognize the object's location and/or grasping pose, using RGB and/or depth images, 3D point cloud, etc., as input. In recent research, there are several methods of synthesized grasp pose detection [13]. For example, robotics competitions such as the Amazon Picking Challenge [14] have encouraged the research community [15,16]. The fast graspability evaluation (FGE) [17] is an effective grasp pose detection method. The FGE can detect 4-DOF grasping points for unknown ob-

jects from a single depth image, without requiring object models. In addition, this method can also be applied to entangle objects like mechanical parts by combining a simulation environment and convolutional neural network (CNN) [18]. In contrast, methods based on CNN models use several object models [16], such as simulation-based learning for bin-picking, which can adapt well to real-world environments [19]. These methods can execute grasping with a high success rate, using different types of grippers (two-finger hand, suction and vacuum tools) for several types of shapes and sizes of objects. An FGE-based object picking system was developed for an actual industrial application [20]. However, for deformable objects, because grasp planning does not consider preventing the object's deformation, the grasped object may be damaged by the gripping force. For preventing the damages and failure to grasp, force-sensitive approaches were proposed using force, tactile, and an image to detect contact states [21]. In this paper, we focus on the pre-grasping motion that can consider the prevention of the damages before contact happens. There are also some methods to grasp or manipulate objects by estimating the deformation from images [6,7,22], however, these methods do not consider inhomogeneity stiffness (mentioned in previous section). Hence, additional estimation is required for deformable objects.

2.3. Estimation of physical properties from images

In this study, we propose a simple estimation of the object's stiffness based on images. Previous studies proposed stiffness estimation for specific types of objects, such as fruits. In this case, softness is classified via regression and mathematical models, based on optics principles, using high-frequency images from near-infrared, hyper spectrum cameras as input [23]. Methods for non specific objects measure the displacement on the object's surface, generated when an ultrasonic wave is irradiated [24]. However, these methods can solely be used for specific hardware settings, and are not easy to apply in robotic manipulation systems (e.g. Pick-and-Place).

In addition, methods that estimate various properties from RGB images exist, such as depth estimation [25]. This method adopts an RGB image as input to generate a depth image using a simple deep learning model, as well as deep generative models, such as GAN [8]. GAN comprises a generator G, which generates an image from random multi-dimensional noise, and a discriminator D, which discriminates whether the image is real, and mutually trains two networks. GAN can be applied for image translation tasks [9,26,27], style transfer, and color painting. Pix2pix [9] is a type of GAN, which uses before and after pair of images. This method adopts random noise attached to the image before translation as the input of G to generate the image after translation, and the same image attached as the input of D discriminates whether the pair of images is real; this is called conditional GAN (cGAN). There are also various methods of image translation via GAN's architecture, such as pix2pixHD [26], which generates a high resolution image, CycleGAN [27], which generates an image without using a pair of translation images. Thus, deep learning models have great potential to estimate some types of physical properties using image translation.

3. Proposed Method

In this section, we introduce the proposed methods for stiffness estimation and grasp pose detection. To estimate the object's stiffness, a stiffness map is constructed, which indicates a score of the stiffness for each pixel in an image using image translation

with GAN. Using this map, a grasp point is detected for the object to be grasped.

3.1. Stiffness map generation (pix2stiffness)

In this study, we employ pix2pix [9] network architecture to generate a stiffness map. We solely adopt one image as input; however, the representation of an object's stiffness depends not only on its texture, but also on its shape, material, and other physical properties. Therefore, other information is also required as input. If we consider them as conditions when using cGAN, then there is no need to train a new network, as we would simply need to adjust the inputs. For executing pix2stiffness, we employ the pix2pix architecture because the pair of images required for translation (for the tasks considered in this work) can be generated via simulations.

3.1.1. Data Collection

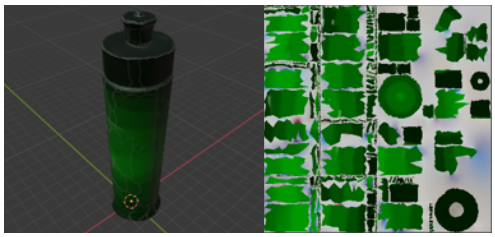
To train the pix2pix network, we need pairs of before and after translation images. To prepare the stiffness map, the annotation of a stiffness score for each pixel is required, and the cost of doing it manually is high. In addition, because stiffness can change in some parts of the object, it is necessary to prepare stiffness maps for various poses of each object, which further increases the cost of doing it manually. To address these problems, we propose a method that semi-automatically generates synthetic data via simulations. We adopt the 3D object model with its texture attached, as well as the Blender physics engine [28]; accordingly, data is prepared as follows:

I. Coloured stiffness map:
For each 3D model, we decide to divide a green color gradation into 10 tones, as a guideline for understanding the effect of damage triggered by grasping, and for representing the stiffness score of each object's surface. Accordingly, this approach generates a 3D stiffness map as a texture attached to the object's surface (Figure 2a). The stiffness scores are based on manual measurements by a hardness meter.

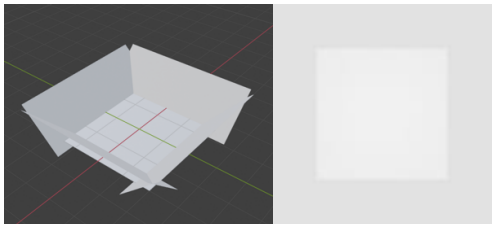
II. Execute simulation:
After preparing a white bin (Figure 2b), a simulation that involves dropping objects with random positions and postures from above the bin is computed by Blender.

III. Capture images:
We took images from the top of the bin when using the original texture of the object (Figure 2c) and also when using the stiffness map texture (Figure 2d). Accordingly, we obtained a pair of images. In the stiffness map, because the element value of green indicates the stiffness score for each position, we convert the map from 3-channel to 1-channel (green). And we use this map for training.

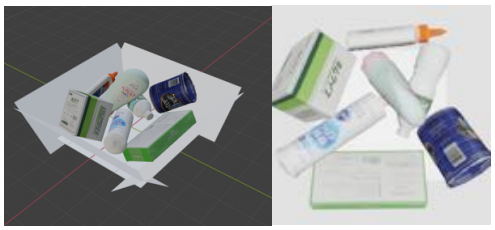
We semi-automatically generated the training data by repeating the above steps. Because we generated a clutter scene, the synthetic data exhibited various scenes with randomized object poses. The green tone solely represents the stiffness score of the created stiffness map.



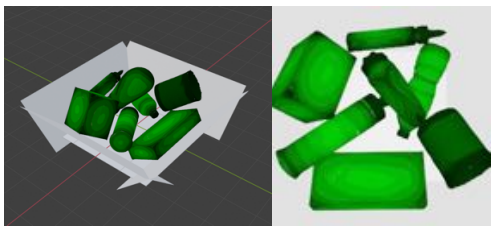
(a) Coloured stiffness map and map image of 3D object model



(b) Initial state and background image



(c) Dropped state and original image



(d) Dropped state and stiffness map image (same state as Figure 2c)

Figure 2. Data collection using a physics simulator

3.1.2. GAN Training

We employ the pix2pix architecture for pix2stiffness translation. The objective adversarial loss is defined by pix2pix [9]:

$$L_{GAN} = \mathbb{E}_s[log(D(x,s))] + \mathbb{E}_x[log(1 - D(x, G(x)))] \ , \tag{1}$$

where G is trained to minimize this objective and D is trained vice versa. x and s indicate the input (RGB or Depth) and stiffness map images, respectively. In addition, the loss function is also based on the $L1$ distance to obtain a generated image $G(x)$ closer to the ground truth s:

$$L_{L1} = \mathbb{E}_{x,s}[||s - G(x)||_1] \ , \tag{2}$$

Our problem is

$$G^* = \arg\min_G \max_D L_{GAN} + \lambda L_{L1} \ . \tag{3}$$

The image input and stiffness map output have a size of 256×256, the generator has a U-Nets structure [29], which has a seven-layer encoder and a seven-layer decoder with skip-connection and dropout for all layers. The discriminator value is calculated using PatchGAN, which judges True/False for each small region of an image (Figure 3). To train the network, we can use an arbitrary number of pair of image-stiffness map images (described in the previous section).

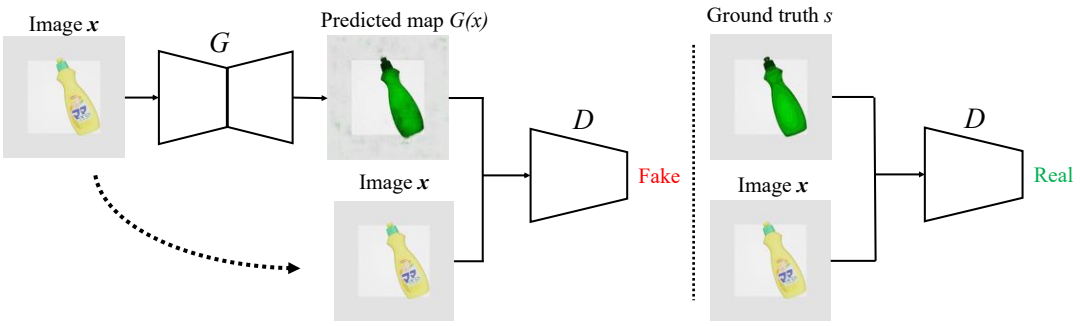


Figure 3. Image translation network architecture of pix2stiffness referenced by pix2pix [9]

3.2. Grasp pose detection using stiffness map

In this section, we describe the grasp pose detection method using the stiffness map generated by pix2stiffness. This stiffness map can be easily applied to a method that indicates a grasp score for each pixel in an image. In this study, we propose a 4-DOF grasp pose detection using a stiffness map and a depth image constrained to a grasping pose vertical to a plane located in the target objects. The proposed method adopts the stiffness score as the grasp quality score for each grasp candidate in an image.

3.2.1. Overview of the FGE [17]

The FGE is a method that detects a 4-DOF grasping pose using a single depth image. Using these depth and template images, FGE calculates contact and collision regions of the hand and a target object, then it computes a non-collision region that represents grasp pose candidates. Subsequently, a graspability map that indicates the points that are closer to the object's center of mass is generated by convoluting a Gaussian filter with the non-collision region. The optimal grasping pose is detected as the position with the highest graspability value. The contact $\mathbf{T}_t$ and collision $\mathbf{T}_c$ templates are predefined, while the contact $\mathbf{I}_t$ and collision $\mathbf{I}_c$ images are obtained from a single depth image.

Then, the contact region $\mathbf{A}_t$ can be calculated by convoluting $\mathbf{T}_t$ with $\mathbf{I}_t$:

$$\mathbf{A}_t = \mathbf{T}_t \otimes \mathbf{I}_t \ , \tag{4}$$

$\otimes$ denotes the convolution. The collision region $\mathbf{A}_c$ can be calculated by convoluting $\mathbf{T}_c$ with $\mathbf{I}_c$:

$$\mathbf{A}_c = \mathbf{T}_c \otimes \mathbf{I}_c \ . \tag{5}$$

To compute the non-collision region and the graspability score of each pixel, the graspability map is calculated as the region where the gripper does not interfere with the surrounding area near the center of gravity of the object region (each image can be seen in Figure 4).

Finally, the graspability map is calculated for each angle of the hand model, and the optimal grasp pose is the point with the highest graspability score.

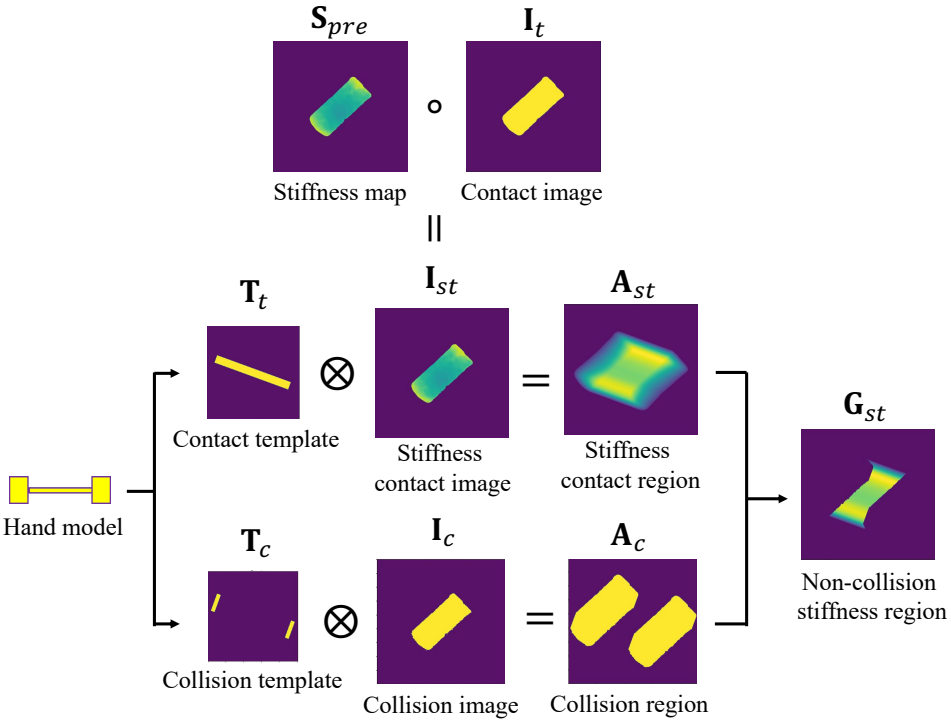


Figure 4. Processing pipeline of the grasp pose detection method

3.2.2. Grasp pose detection using stiffness map

FGE can select the grasp pose nearest to the position of the object's center of mass; however, for deformable objects, it may cause a large deformation that triggers permanent damages. By using a stiffness map, we can detect a grasp pose that addresses these problems.

At first, a stiffness map $\mathbf{S}_{pre}$ that indicates $G(x)$ for each pixel is generated by pix2stiffness, normalization and some pre-processing are used. In $\mathbf{S}_{pre}$, a larger value denotes that it is more difficult to deform. When a grasp pose with a high score is detected, this implies that it is possible to grasp and prevent damage simultaneously. By using this stiffness map, we proposed a modified FGE that fits the objectives of this study.

Secondly, the stiffness contact image $\mathbf{I}_{st}$ is calculated by multiplying the generated stiffness map $\mathbf{S}_{pre}$ and the contact image $\mathbf{I}_t$:

$$\mathbf{I}_{st} = \mathbf{S}_{pre} \circ \mathbf{I}_t \ , \tag{6}$$

$\circ$ denotes the Hadamard product. Then, it is convoluted with the contact template ($\mathbf{I}_t$ is replaced with $\mathbf{I}_{st}$). Subsequently, the stiffness contact region $\mathbf{A}_{st}$ is generated;

$$\mathbf{A}_{st} = \mathbf{T}_t \otimes \mathbf{I}_{st} \ , \tag{7}$$

$\mathbf{A}_{st}$ represents the average stiffness score of each pixel in the rectangular region surrounded by the 2-finger gripper (in the contact template $\mathbf{T}_t$). Via this convolution, this stiffness score differs from the original one, and the score of the locations near the

object's silhouette is slightly lower than those closer to its center. Additionally, the contact region $\mathbf{A}_t$ and the collision region $\mathbf{A}_c$ are also generated in the same way as FGE.

The grasp candidates which are collision free, are obtained using a logical AND operation between $\mathbf{A}_t$ and $\overline{\mathbf{A}_c}$, thus, the non-collision stiffness region $\mathbf{G}_{st}$ (similar to the graspability map) is generated as:

$$\mathbf{G}_{st} = \mathbf{A}_{st} \circ (\mathbf{A}_t \cap \overline{\mathbf{A}_c}) \ , \tag{8}$$

where $G_{st}(h, w)$ denotes the element value of the position (h, w) in $\mathbf{G}_{st}$. The objective function is defined as:

$$f(h, w, \theta) = \begin{cases} G_{st}(h, w) & if \ \ A_c(h, w) = 0 \\ 0 & otherwise \end{cases} \ , \tag{9}$$

where θ denotes the rotation angle of the detected grasp candidate, and $A_c(h, w)$ denotes the element value of the position (h, w) in $\mathbf{A}_c$. The calculated coordinate index is expressed as:

$$[H, W, \Theta] = \arg\max_{h,w,\theta} f(h, w, \theta) \ . \tag{10}$$

Here, we only utilize a two-finger gripper's hand template; hence, we can apply Eq. (6) as the objective function (described in Figure 4).

4. Simulation Results

In this section, we evaluate the accuracy of pix2stiffness estimation and the effectiveness of our grasp pose detection method in simulation scenes. For training, we prepared fifteen models of 3D objects in Figure 5a and stiffness maps annotated as explained in section 3.1. For validation, seven unknown (we define some categories such as bottle and box, then we target objects in the same category but with different shapes) models (Figure 5b) are prepared.

(a) Training (b) Validation

Figure 5. Dataset of 3D object models used in simulation.

4.1. Image quality evaluation

Using the training data presented above, we evaluate the results obtained with different input data types (RGB and Depth). The quantitative evaluation metrics of the adopted pix2stiffness estimation are: 1) root mean square error ($RMSE$) and 2) structural similarity index ($SSIM$) [30] between the ground truth stiffness map $\mathbf{S}$ and the predicted map $\hat{\mathbf{S}}$ using pix2stiffness. Especially, $SSIM$ considers changes in brightness, contrast and the entire structure. These metrics are usually used for depth estimation [31]. $RMSE$ is calculated as:

$$RMSE(\mathbf{S}, \hat{\mathbf{S}}) = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(s_i - \hat{s}_i)^2} \ , \tag{11}$$

where M denotes the number of $\mathbf{S}$ pixels (same as $\hat{\mathbf{S}}$). s_i and $\hat{s}_i$ denote each element i-th value of $\mathbf{S}$ and $\hat{\mathbf{S}}$, respectively. The closer $RMSE$ is to zero, the lower the pixel-wise error is. $SSIM$ is a metric based on appearance, which is computed for each of the evenly divided small areas. This metric can analyze spatial similarity. We adopt the mean of $SSIM$ ($MSSIM$) to evaluate the entire image quality:

$$MSSIM(\mathbf{S}, \hat{\mathbf{S}}) = \frac{1}{N}\sum_{j=1}^{N} SSIM(\mathbf{S}_j, \hat{\mathbf{S}}_j) \ . \tag{12}$$

$SSIM(\mathbf{S}_j, \hat{\mathbf{S}}_j)$ is calculated between $\mathbf{S}_j$ and $\hat{\mathbf{S}}_j$ (N is the number of areas, and we use $N = 100$) for each region j. The closer $MSSIM$ is to one, the better the similarity of the entire image. For evaluation, the predicted stiffness map $\hat{\mathbf{S}}$ is preprocessed map $\mathbf{S}_{pre}$. The obtained results are summarized in Table 1. It can be observed that higher accuracy is obtained with depth as input.

Table 1. Estimation results for different training dataset types

Dataset type	$RMSE$	$MSSIM$
RGB	47.68	0.7250
Depth	**32.91**	**0.8552**

4.2. Effectiveness of grasp pose detection

Furthermore, we also evaluate the influence of pix2stiffness on the grasp pose detection by calculating the mean of stiffness score in the rectangular space surrounded by a two finger gripper (grasp region) when using the ground truth stiffness map s. In this evaluation, we adopt the segmentation image from simulation as the contact and collision images, assuming a picking scene with an object placed on the table. After cropping the grasp region from $\mathbf{I}_{st}$, The mean of stiffness score (the higher the better) is calculated as:

$$R = \sum_{k=1}^{L} \begin{cases} 1 & \text{if } \mathbf{s}_k > 0 \\ 0 & \text{otherwise} \end{cases} \ , \tag{13}$$

$$Mean\ of\ stiffness = \frac{1}{R}\sum_{k=1}^{L} \mathbf{s}_k\ . \tag{14}$$

where L denotes the number of pixels in the grasp region, and R is calculated as the size of the object's region. The proposed grasp pose detection method described in section 3.2.2 is adopted in this evaluation, to demonstrate the validity of the proposed method for grasping deformable objects while preventing damage. Using the model of pix2stiffness with the depth image as input, Figure 6 presents examples of detected grasp poses in simulations, where the red line represents the detected pose for a two-finger gripper (the blue one is detected by FGE). Table 2 presents the evaluation of seven images for each single object when using the proposed method and FGE. The result of each object and the mean of stiffness are relatively higher than FGE's result. In the result of "Bottle 3", the score of FGE is close to the proposed method. The reason is that the lowest stiffness of this object is 0.7 which has overall a high stiffness. In the result in Figure 6, the grasp poses were detected far from the center of gravity, therefore it has some possibility of failure to grasp. Because we only verified the possibility to prevent damages by considering grasping to the hard part, more evaluation that the grasp pose can be executed successfully in the real-world is needed.

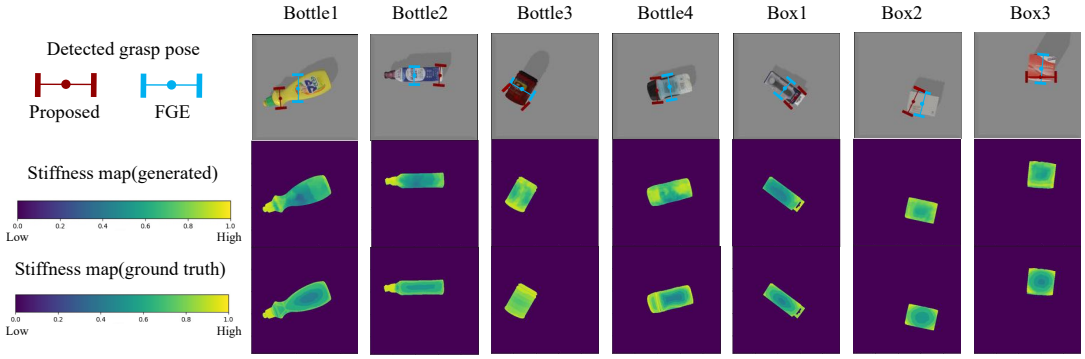


Figure 6. Detected grasp pose using the proposed method for seven objects in simulation: in the top row, the red line represents the detected pose for a two-finger gripper. In the middle row, the images represented the stiffness maps generated by pix2stiffness. In the bottom row, the images represent the ground truth stiffness maps generated via simulations.

Table 2. Mean of stiffness (*Mean of stiffness*) for single object in simulations. Each object's name is as described in Figure 6

Name	*Mean of stiffness*	
	FGE [17]	Proposed method
Bottle 1	0.6141	0.7149
Bottle 2	0.6293	0.6896
Bottle 3	0.7710	0.8138
Bottle 4	0.6043	0.7977
Box 1	0.5446	0.7207
Box 2	0.5897	0.7129
Box 3	0.6011	0.7125
Mean	0.6220	**0.7374**

5. Real-World Experiments

In this section, we evaluate the predicted stiffness map and detected grasp pose using the map for real images (adopt depth image as input, same as in section 4.2). Because real depth images have some noise, missing values, and errors, the estimation networks are trained with only simulation images, and it is inadequate to adopt raw images as the input for pix2stiffness. To address this problem, raw images are preprocessed via fast digital inpainting [32] and contrast emphasis. By using these simple pre-processing methods, the stiffness map can be generated more clearly. The target objects in grasping experiments are presented in Figure 7. The hardware used in the experiments are a UR5, a Robotiq 2-finger gripper (140mm stroke), and a Realsense SR305 attached to the gripper.

Figure 7. Target objects in real experiments that are not included in training data

5.1. Grasp experiments for single object scene

In this section, we evaluate the effectiveness of the proposed method on real-world images via grasping experiments. The experimental scene is assumed to be a single object placed on a table (same as in simulation). In addition, we manually set the height of the gripper from the table in the proposed 4-DoF grasping pose detection, where the silhouette of the object's region can be almost obtained in the contact/collision image.

Figure 8 presents the grasp pose detection results, generated stiffness map, and grasping behavior for each object. It can be observed that the proposed method can grasp the hard part of each of the objects. However, for "Object 1" and "Object 6", the poses of these objects changed during the lifting motion. This result indicates that the proposed method can fail to stably grasp the object, as the grasp pose for most of the objects is detected far from their center of gravity. However, because we adopt the strategy of grasping the hard part of the object, it can be considered to be successful because few deformations are generated by a posture change of the object.

As a quantitative evaluation, we analyze the deformation the object sustains by grasping. To evaluate the mean of stiffness in the simulation results, we adopt the stiffness map of the ground truth; however, it is difficult to prepare the same map for real-world experiments. Therefore, we use the grasping width after performing grasping for the evaluation. First, we manually measure the grasping width at the moment of contact with the object (called l_c), then, we measure the grasping width after grasping

with a certain grasping force (called l_g). The grasping force can be determined by using Robotiq's gripper function (10-125 [N]) provided by URCaps [33]. In this experiment, we set the constant grasping force to 62.5 [N] assuming the case of no-force control, this value is the smallest force that can grasp the heaviest object (Object 1 in Figure 8) in our experiments. After measuring the two grasping widths (l_c and l_g), we define the deformation rate using the following equation.

$$Deformation\ rate = \frac{l_c - l_g}{l_c} * 100\ [\%]\ . \tag{15}$$

Table 3 presents the deformation rate by grasping in Figure 8 for each object. For most of the results, the deformation rate is lower than FGE, which indicates that the proposed method can prevent the deformation of the object. However, in the result for "Object 2", the deformation rate is higher than FGE. The reason is that the grasping force is applied to a narrower surface than FGE's result because the finger surface was slightly inclined to the object's surface. By addressing this problem, it is expected that the object's deformation can be prevented.

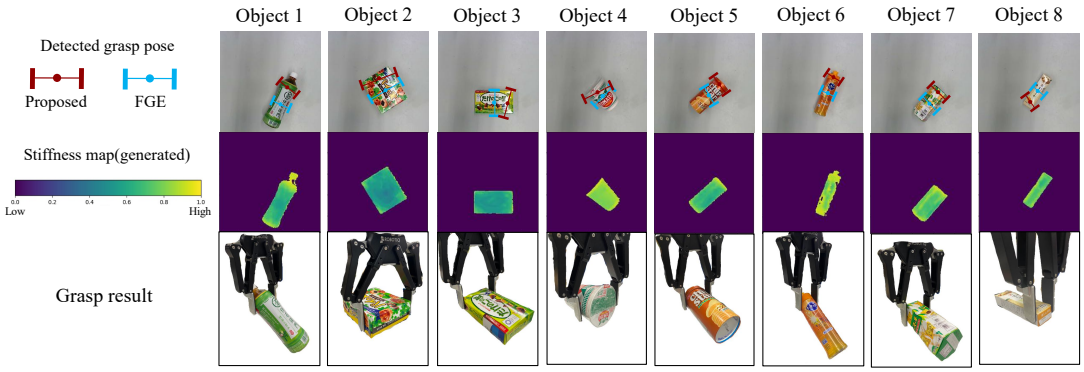


Figure 8. Detected grasp pose using the proposed method for eight objects in real-world: in the top row images, the red line represents the detected pose for a two-finger gripper. In the middle row, the stiffness maps generated by pix2stiffness are presented. In the bottom row, results obtained for grasping and lifting a single object placed on a table are presented.

Table 3. Deformation rate results for single objects in real-world

Name	*Deformation rate*	
	FGE [17]	Proposed method
Object 1	25.13	12.44
Object 2	7.913	9.428
Object 3	14.25	9.757
Object 4	13.74	7.610
Object 5	13.69	8.885
Object 6	18.18	10.91
Object 7	18.33	14.47
Object 8	66.14	21.64
Mean	22.17	**11.89**

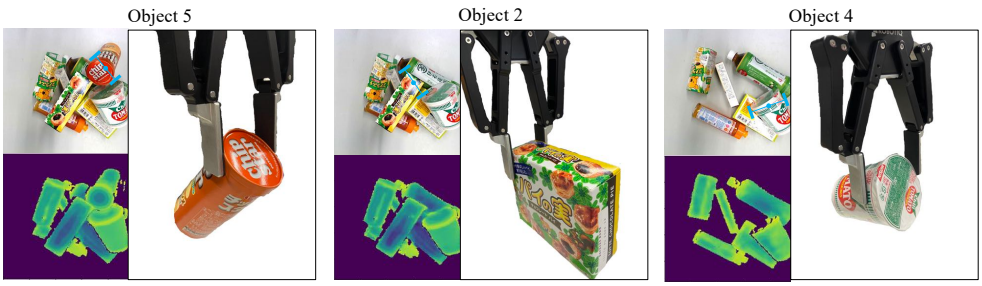


Figure 9. Some examples of successful grasping results in clutter scene. There are three successful cases of grasping each target object while preventing deformation and avoiding collision with other objects during grasping.

5.2. Grasp experiments for clutter scene

Similarly to section 5.1, we evaluate the deformation rate and success rate in a cluttered scene, and compare it with FGE. The robot repeats the trial until all eight objects are grasped successfully in the given cluttered scene. Although it is necessary to determine the grasping height in the proposed method, it is not trivial in a cluttered scene. In this experiment, the candidate grasping height is set at regular intervals (each of 10 [mm]), and the grasp pose is searched from the height where there is a certain amount of one object area in the contact image to a height five steps lower. This method is also applied to the FGE, and the grasp pose with the highest graspability score is selected. As explained in section 3.2.2, the score map $\mathbf{G}_{st}$ in our proposed method does not represent the original stiffness score; hence, it is difficult to select a relatively high score in all grasp candidates. Instead of the score, we calculate a new score, same as the mean of stiffness (described in Eq. (15)), and delete the grasp pose candidate whose size difference between the contact image and collision image (cropped grasping area) is higher than a threshold value (eliminating failure cases owing to the slippage of the hand and the object).

Figure 9 shows some of the successful grasping results. The deformation rate evaluation is presented in Table 4. Here, it can be observed that the deformation of the object is suppressed in several cases. For "Object 2", the grasping result is not optimal because the grasping direction differs from the object's surface; hence, the grasping force is applied to a narrower surface than the FGE's result. Regarding the success rate of grasping, FGE succeeded in all attempts, and the proposed method failed in three attempts. It is necessary to improve the method specifically introduced for the cluttered scene, as well as expand the method for 3D because the stiffness of the contact point with the gripper cannot be properly measured using only 2D images.

5.3. Discussion

In the two experiments in section 5.1 and 5.2, when the deformation rate is more than 20%, the damage was caused by large deformation in the FGE case (Figure 10), which suggests that the proposed method can reduce damage. The reason for the three failures in the experiments of section 5.2 is that the grasping pose selected was often close to the edge of the object. This makes the contact surface smaller, and the possibility of failure was increased by a small disturbance or error in the grasp pose control. Therefore, we need to accurately determine the grasping depth in the clutter scene, and determine the grasping pose based on the grasp stability.

Table 4. Deformation rate results for each cluttered object in real-world

Name	*Deformation rate*	
	FGE [17]	Proposed method
Object 1	12.19	8.373
Object 2	14.80	15.39
Object 3	24.35	12.42
Object 4	13.09	6.144
Object 5	16.59	9.129
Object 6	17.45	9.191
Object 7	19.10	16.14
Object 8	64.58	29.69
Mean	22.77	**13.31**

Figure 10. Some examples of grasping with significant deformation in the FGE case. Each deformation rate was more than 20%.

6. Conclusion

In this study, we proposed a pix2stiffness estimation method, which generates a stiffness map that indicates the object's stiffness for each pixel on an image using the pix2pix architecture. We demonstrated that the stiffness estimation has a higher accuracy when using depth images as input data than when adapting RGB. Furthermore, we introduced a grasp pose detection method using a stiffness map based on FGE, which we called GPD-stiffness. This method can robustly detect grasp poses in clutter scenes in the real-world. However, more experiments are required for various objects, and generating the stiffness map (data collection in section 3.1) is time consuming and cumbersome because it is manually done. In the future, we would like to automatically generate the annotated stiffness map using contact force (e.g. grasped distance for each object [3]). Also, we would like to introduce a force-adjustable method that can grasp with the smallest deformation by considering contact dynamics.

Acknowledgment

This work was supported by JST [Moonshot R&D][Grant Number JPMJMS2031].

References

[1] M. A. Roa, and R. Suárez, "Grasp quality measures: review and performance," Auton Robot 38, 65–88, 2015.

[2] M. Danielczuk, J. Xu, J. Mahler, M. Matl, N. Chentanez, and K. Goldberg, "Reach: Reducing false negatives in robot grasp planning with a robust efficient area contact hypothesis model," in Int. S. Robotics Research (ISRR), 2019.
[3] J. Xu, M. Danielczuk, Jeffrey Ichnowski, J. Mahler, E. Steinbach, and K. Goldberg, "Minimal Work: A Grasp Quality Metric for Deformable Hollow Objects," International Conference on Robotics and Automation, 2020.
[4] M. Pfanne, M. Chalon, F. Stulp, H. Ritter, and A. Albu-Schäffer, "Object-Level Impedance Control for Dexterous In-Hand Manipulation," in IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 2987-2994, April 2020.
[5] I. Huang, Y. Narang, C. Eppner, B. Sundaralingam, M. Macklin, T. Hermans, and D. Fox, "DefGraspSim: Simulation-based grasping of 3D deformable objects," arXiv:2107.05778 [cs.RO], 2021.
[6] B. Thach, A. Kuntz, and T. Hermans, "DeformerNet: A Deep Learning Approach to 3D Deformable Object Manipulation," arXiv:2107.08067 [cs.RO], 2021.
[7] T. N. Le, J. Lundell, F. J. Abu-Dakka, and V. Kyrki, "Deformation-Aware Data-Driven Grasp Synthesis, arXiv:2109.05320 [cs.RO], 2021.
[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," Proc. NIPS, pp. 2672–2680, 2014.
[9] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," 2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2017.
[10] Z. Li, and S. S. Sastry, "Task-oriented optimal grasping by multifingered robot hands," IEEE Journal on Robotics and Automation, vol. 4, no. 1, pp. 32–44, 1988.
[11] K. Gopalakrishnan, and K. Goldberg, "D-space and deform closure grasps of deformable parts," Int. Journal of Robotics Research (IJRR), vol. 24, no. 11, pp. 899–910, 2005.
[12] Y.-B. Jia, F. Guo, and H. Lin, "Grasping deformable planar objects: Squeeze, stick/slip analysis, and energy-based optimalities," Int. Journal of Robotics Research (IJRR), vol. 33, no. 6, pp. 866–897, 2014.
[13] M. Fujita, Y. Domae, A. Noda, G. A. Garcia Ricardez, T. Nagatani, A. Zeng, S. Song, A. Rodriguez, A. Causo, I. M. Chen, and T. Ogasawara, "What are the important technologies for bin picking? Technology analysis of robots in competitions based on a set of performance metrics," Advanced Robotics, 34:7-8, 560-574, DOI: 10.1080/01691864.2019.1698463.
[14] N. Correll, K. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Okada, A. Rodriguez, J. Romano, and P. Wurman, "Analysis and Observations From the First Amazon Picking Challenge", IEEE Trans. on Automation Science and Engineering (T-ASE), Vol.15, No.1, pp.172–188, 2018.
[15] H. Fujiyoshi, T. Yamashita, S. Akizuki, M. Hashimoto, Y. Domae, R. Kawanishi, M. Fujita, R. Kojima, and K. Shiratsuchi, "Team C2M: Two Cooperative Robots for Picking and Stowing in Amazon Picking Challenge 2016," Springer, Advances on Robotic Item Picking, pp.101–112, 2020.
[16] A. Zeng, S. Song, K. Yu, E. Donlon, F. Hogan, M. Bauza, D. Ma, O. Taylor, M. Liu, E. Romo, N. Fazeli, F. Alet, N.Dafle, R. Holladay, I. Morona, P. Nair, D. Gree, I. Taylor, W. Liu, T. Funkhouser, and A. Rodriguez, "Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching", 2018 IEEE International Conference on Robotics and Automation (ICRA), 2018.
[17] Y. Domae, H. Okuda, Y. Taguchi, K. Sumi, and T. Hirai, "Fast graspability evaluation on single depth maps for bin picking with general grippers," International Conference on Robotics and Automation, pp.1997– 2004, 2014.
[18] R. Matsumura, Y. Domae, W. Wan, and K. Harada, "Learning Based Robotic Bin-picking for Potentially Tangled Objects," 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2019, pp. 7990-7997, doi: 10.1109/IROS40897.2019.8968295.
[19] H. Tachikake, and W. Watanabe, "A Learning-based Robotic Bin-picking with

Flexibly Customizable Grasping Conditions," 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2020, pp. 9040-9047, doi: 10.1109/IROS45743.2020.9340904.
[20] Y. Domae, A. Noda, T. Nagatani, and W. Wan, "Robotic General Parts Feeder: Bin-picking, Regrasping, and Kitting," 2020 IEEE International Conference on Robotics and Automation (ICRA), 2020, pp. 5004-5010, doi: 10.1109/ICRA40945.2020.9197056.
[21] S. Cui, R. Wang, J. Wei, F. Li, and S. Wang, "Grasp State Assessment of Deformable Objects Using Visual-Tactile Fusion Perception," 2020 IEEE International Conference on Robotics and Automation (ICRA), 2020, pp. 538-544.
[22] Z. Hu, T. Han, P. Sun, J. Pan, and D. Manocha, "3-D Deformable Object Manipulation Using Deep Neural Networks," in IEEE Robotics and Automation Letters, vol. 4, no. 4, pp. 4255-4261, Oct. 2019.
[23] R. Lu, R. V. Beers, W. Saeys, C. Li, and H. Cen, "Measurement of optical properties of fruits and vegetables: A review," Postharvest Biology and Technology, vol. 159, 2020.
[24] M. Fujiwara, K. Nakatsuma, M. Takahashi, and H. Shinoda, "Remote measurement of surface compliance distribution using ultrasound radiation pressure," 2011 IEEE World Haptics Conference, 43–47, 2011.
[25] C. Zhao, Q. Sun, C. Zhang, Y.Tang, and F. Qian, "Monocular depth estimation based on deep learning: An overview," Sci. China Technol. Sci. 63, 1612–1627, 2020.
[26] T. Wang, M. Liu, J. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 8798-8807, doi: 10.1109/CVPR.2018.00917.
[27] J. Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," IEEE International Conference on Computer Vision (ICCV), 2017.
[28] Blender, https://www.blender.org/(Last viewed Oct 21, 2020).
[29] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in Medical Image Computing and Computer-Assisted Intervention (MICCAI), ser. LNCS, vol. 9351, pp. 234—241, Springer, 2015.
[30] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," in IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600–612, 2004.
[31] K. G. Lore, K. K. Reddy, Michael Giering, and E. A. Bernal, "Generative Adversarial Networks for Spectral Super-Resolution and Bidirectional RGB-To-Multispectral Mapping," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 926–933, 2019.
[32] M. M. Oliveira, B. Bowen, R. McKenna, and Y. S. Chang, "Fast Digital Image Inpainting," Proc. of Int. Conf. on Visualization, Imaging and Image Processing (VIIP), pp. 261–266, 2001.
[33] Universal Robotics, https://www.universal-robots.com/(Last viewed July 6th, 2021).