Article

Bio-inspired Proprioceptive Touch of a Soft Finger with Inner-Finger Kinesthetic Perception

Xiaobo Liu[1,2], Xudong Han[1,2], Ning Guo[1,2], Fang Wan[1,3,*] and Chaoyang Song[2,4,*]

[1] Shenzhen Key Laboratory of Flexible Manufacturing and Robotics, Southern University of Science and Technology, Shenzhen, Guangdong 518055, China.
[2] Department of Mechanical and Energy Engineering, Southern University of Science and Technology, Shenzhen, Guangdong 518055, China.
[3] School of Design, Southern University of Science and Technology, Shenzhen, Guangdong 518055, China.
[4] Guangdong Provincial Key Laboratory of Human-Augmentation and Rehabilitation Robotics in Universities, Southern University of Science and Technology, Shenzhen, Guangdong 518055, China.
* Correspondence: wanf@sustech.edu.cn (F.W.); songcy@ieee.org (C.S.)

Abstract: Manipulating objects inside the hand is an essential skill for humans and robots that requires the object's pose. In-hand object pose estimation is a challenging problem because of the heavy occlusion produced by the hand and object. Humans can perceive the position and orientation of objects with the finger's kinesthesia; analogously, robots can also estimate object pose with the gripper pose and tactile sensing. Inspired by human fingers, we designed a soft finger integrated inner vision with kinesthetic sensing and a framework for object state estimation based on kinesthetic sensing. This soft finger has a flexible skeleton and skin that is adaptive to different objects, and the skeleton deformations during interaction represent contact information obtained by the image from the inner camera. The framework is an end-to-end method that uses raw images from soft fingers to estimate in-hand object pose, and it consists of two parts: an encoder for kinesthetic information processing and an estimator for object pose and category. We test the fingers and framework on seven objects and get an error of 2.02 mm and 11.34 degrees for pose error and 99.05% for classification.

Keywords: object recognition; kinesthetic; humanoid finger; inner-finger vision; deep learning

Citation: Liu, X.; Han, X.; Guo, N.; Wan, F.; Song, C. Bio-inspired Proprioceptive Touch of a Soft Finger with Inner-Finger Kinesthetic Perception. *Biosensors* **2023**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

1. Introduction

Humans exhibit various manipulative behaviors with the ability to detect the interaction behaviors of handled objects and hands. Visual information provides whole and rich features for humans to feel objects' shapes. But without visual information, humans can still assess object properties, such as size, shape, position, and orientation, using the sense of touch alone [1]. There are many receptors in the skin at different depths on human hands to perceive mechanical stimulus during the interaction. Those receptors empower humans to feel objects relying on the sense of touch: cutaneous and kinesthetic [2]. The cutaneous sense is the modality that depends on direct contact between receptors and objects and is better for feeling the material properties. In contrast, the kinesthetic sense is the awareness of the position and movement of the body. It is better to feel the object's shape, orientation, etc., from the receptors within muscles, tendons, and joints [3]. Inspired by the kinesthetic sense, we present a soft finger with an embedded camera and a deep learning architecture for object recognition.

According to the structure of human hands, many methods have been proposed for hand pose estimation (HPE) problem [4–6]. As the shape of an object and the configuration of a hand (how many fingers are used to manipulate objects and fingers' positions) are constrained by each other [7], some hand object joint detection methods are proposed which are called hand-object pose estimation (HOPE) [8–10]. Like humans manipulating objects with HOPE, object pose recognition is also a fundamental and challenging task in robotics.

For a manipulation task, perception of the environment and objects is essential [11]. Vision sensors are standard solutions to perceive the environment, and many methods have been proposed for object localization and classification [12–15]. While deep learning significantly improves performance in object recognition problems, the inevitable occlusion is still challenging, especially in dexterous manipulation tasks. Even if we get an object pose with high precision before manipulation, the pose during manipulation in the gripper is still unknown as the inherent uncertainties, tolerances, and noise in the robotic system [16].

Inspired by the HOPE problem, we try to solve the robot gripper-object pose estimation problem with gripper pose estimation. For fully-actuated grippers, we can get the joints' angles of the gripper from motors and contact states from tactile and force sensors and estimate object pose and category with that information [17–19]. For under-actuated grippers, we need additional sensors to measure extra degrees of freedom (DoF), then estimate object pose, and category [20–22].

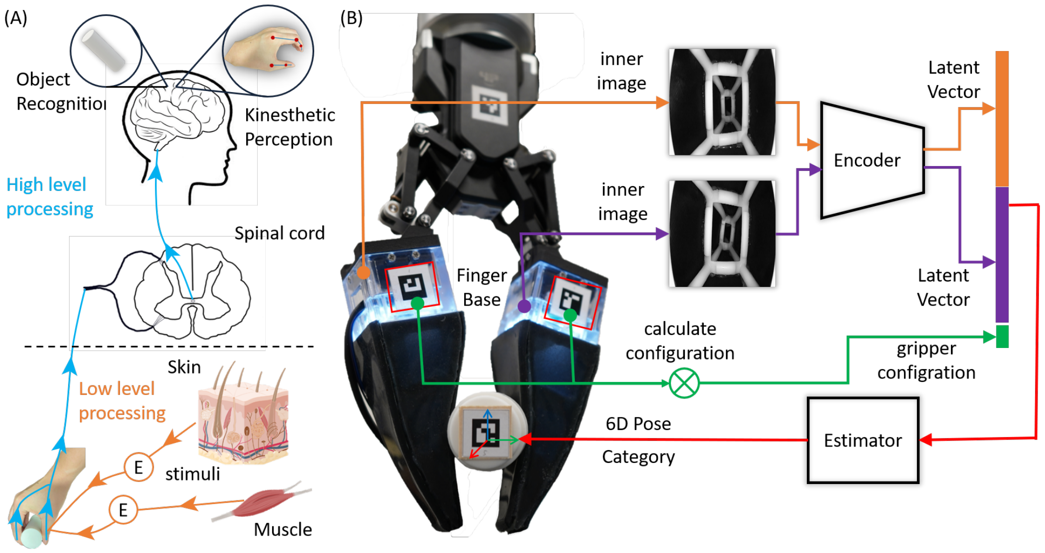


Figure 1. Overview of the bio-inspired finger and framework. (A) The raw stimulus is encoded with low-level processing and then transmitted to the central nervous system (CNS) for high-level processing such as object recognition; (B) The bio-inspired fingers with flexible skeleton and silicon gel skin and a framework for object recognition: the raw images from fingers are encoded as latent vectors, and then used for auxiliary tasks such as pose estimation and object classification.

Those methods mentioned above use multi-sensors in fingers for joints and tactile sensors in the fingertip for contact states and then estimate objects' poses and categories with CAD models. In this article, we propose a soft, adaptive finger with an integrated camera to infer the finger deformation during interaction with objects, as shown in **Figure 1**. We mount the soft fingers on a gripper to enhance the adaption of the gripper and recognize handle objects with their proprioceptive sensing. Our method uses raw images to estimate objects' pose and categories with a camera and unknown CAD modes. Instead of a two-stage method to recognize gripper state and pose, our method is one-stage to recognize handled objects' pose and categories from the raw images. To simplify the training and enhance the reusability of the method, we split the method into two parts: feature extractor for interaction information embedding and post-processor for further manipulation tasks. The feature extractor is an Encoder-Decoder architecture with ResNet block [23], and the post-processor is a multilayer perceptron (MLP) for classification and regression. The main contributions of this paper include the following: First, we design and fabricate a soft finger with an integrated camera inside for proprioception. Second, we propose a frame to extract and fuse fingers' data for objects' states in a gripper. Finally, we test the effectiveness of the proposed method and get high accuracy in pose estimation and classification.

2. Materials and Methods

2.1. Design and Fabrication of the Soft Finger with Inner Vision

In our previous work [24], we leveraged the soft finger with an ArUco marker inside to sense contact force and torque, which encoder the deformation of the finger. In this study, we introduce several improvements to the finger design as shown in **Figure 2**:

- Added silica gel coatings on the finger to isolate the outside environment for a clear background.
- Added an LED light for illumination as the coating blocked the outside light.
- Removed the ArUco marker and used the finger's skeleton as a deformation feature.

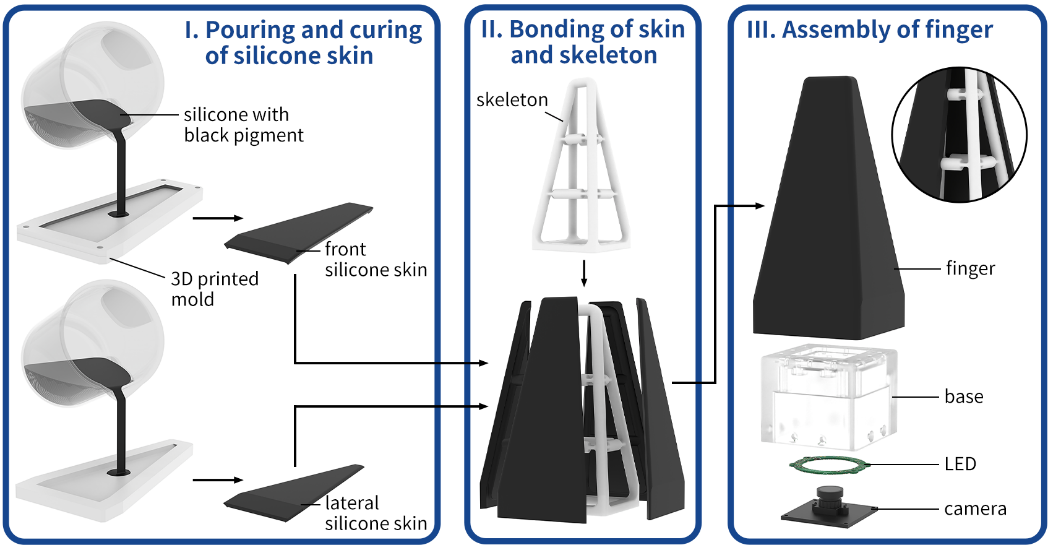


Figure 2. Design and fabrication of the soft finger. (I) The fabrication process of the silica gel skin; (II) attaching the gel skin to the basic finger skeleton; (III) the integrated finger with an LED and an inner camera.

As shown in Figure 2, this new design finger contains a finger skeleton with a black coating, a base frame, an LED light, and a camera. The finger skeleton is used vacuum molding for fabrication using polyurethane elastomers (Hei-cast 8400 from H&K) with a mixing ratio of 1:1:0 for its three components to achieve 90A hardness with robust performances. Alternatively, at a lower cost, one can use other fabrication methods, such as Fused Deposition Modeling (FDM) or Stereolithography (SLA). The coating is made of Smooth-On Ecoflex$^{\text{TM}}$ 00-30 silica gel, and we mixed black pigment to change the color to block the ambient light effectively. Moreover, the silica gel coating's thickness is 3mm, fabricated individually, and attached to the finger skeleton with an adhesive Valigooo® V-80. The white LED light has enough luminous flux for the camera's exposure. The chosen camera is Chengyue WX605 from Weixinshijie, with a 640×360 resolution at 330 fps, and the lens is manually adjustable.

When grasping objects, the finger skeleton and silica gel coating are deformed, which encodes the interaction information captured by the inner camera. So, we use these fingers to recognize the outside object with captured inner images.

2.2. Framework for Handled Object Recognition with the Soft Finger

In this section, we present a framework illustrated in **Figure 3** to extract kinesthesia features and estimate the object state handled by the gripper. This framework contains two parts: an Encoder-Decoder architecture for feature extraction and two auxiliary multilayer perceptrons (MLP) for estimating the object's pose relative to the gripper's coordinate

system and category, respectively. The input of the framework is two fingers' inner images 95
and the gripper configuration. 96

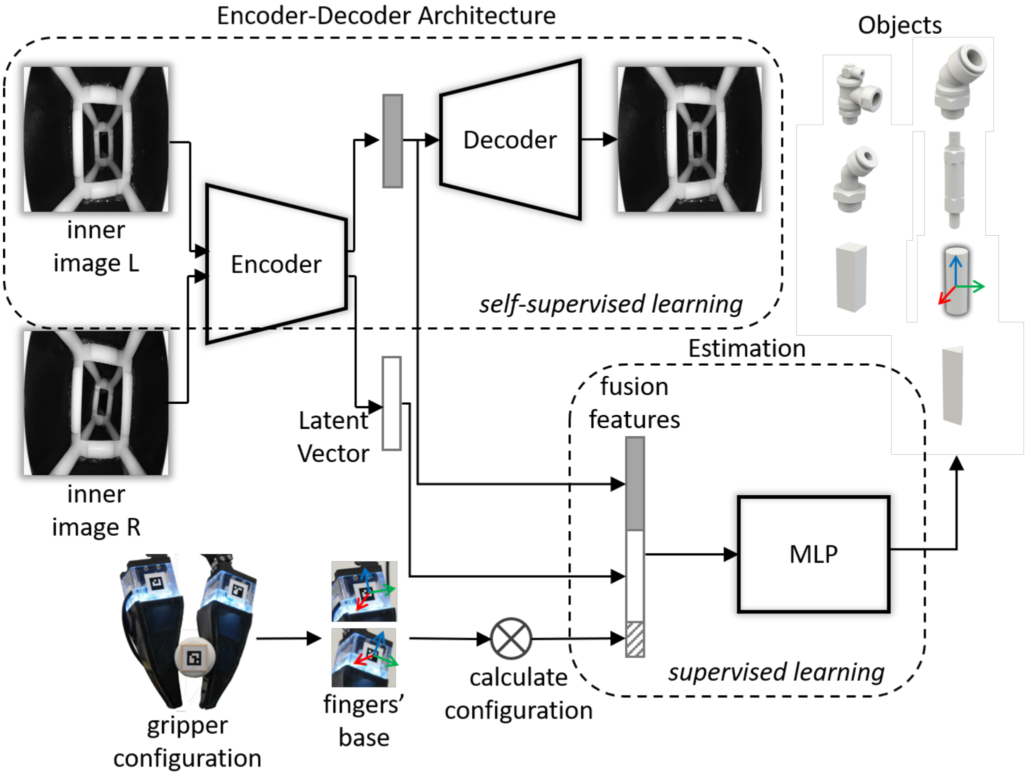


Figure 3. The architecture of the proposed framework: it takes two resized grayscale images and a gripper configuration as inputs and predicts the 6D pose and category of the object.

2.2.1. Encoder-Decoder Architecture 97

Specific details of the Encoder-Decoder architecture are shown in **Figure 4**, the blue 98
block is the encoder, the green block is the decoder, and the yellow vector is the extracted 99
latent vector. The Encoder-Decoder architecture is fully convolutional topology and takes 100
a resized grayscale image $I = \mathbb{R}^{1\times320\times320}$ as input. It extracts the features representing 101
the finger's deformation and outputs an N-dimension vector; the decoder reconstructs the 102
image from the feature vector. 103

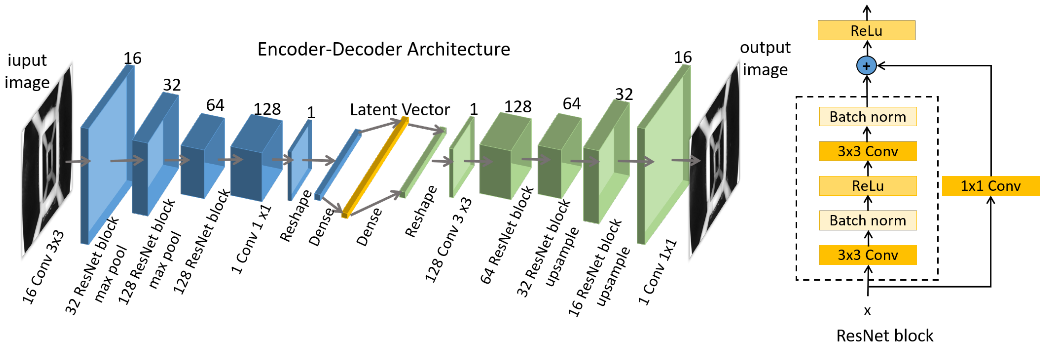


Figure 4. The Encoder-Decoder architecture of the feature extraction: one resized grayscale image as inputs and the same size image as output.

The basic blocks of the Encoder-Decoder architecture are 3x3 convolution and ResNet 104
block for extracting features and 1x1 convolution for compressing features. The dense layer 105

is set to change the latent vector's dimensions and explore the feature dimensions' effect to recover the image.

Define the input as I, the encoder function as E, the latent vector V, and the decoder function D, output Z. The Encoder-Decoder architecture can be described as:

$$V = E(I), \tag{1}$$

$$Z = D(V), \tag{2}$$

$$(\hat{\theta}_e, \hat{\theta}_d) = \underset{\theta_e, \theta_d \in \Theta}{arg} \min Loss(Z, I) \tag{3}$$

here $\hat{\theta}_e, \hat{\theta}_d$ is well trained encoder and decoder parameters, $Loss$ is the loss function between Z and I.

2.2.2. Pose Estimation and Classification

After extracting the latent feature V, we designed two MLP models to estimate the object's pose and category as shown in **Figure 5**. These two models have the same inputs, and the output of the regression model is a 6D pose. In contrast, the output of the classification model is seven classes with a softmax activation function. In this article, the input vector is 129 dimensions, aggregating the two fingers' feature V and gripper configuration. In the follow-up work, we set the dimension of V as 64 and the dimension of the gripper configuration as one since the gripper we used is one degree of freedom (DoF), so the input vector is $64 * 2 + 1 = 129$ dimensions. The regression model consists of five hidden layers with 200, 200, 100, 100, and 100 neurons, with activation function rectified linear unit (ReLu) [25] and batch normalization [26]. The classification model consists of three hidden layers with 200, 200, and 100 neurons, with activation function ReLu.

Define two images taken from the inner cameras of the fingers as $I^L = \mathbb{R}^{C \times H \times W}, I^R = \mathbb{R}^{C \times H \times W}$ with height H and width W, gripper configuration as G_c, regression model as F_r, classification model as F_c, object 6d pose S_p, and object category S_c, the two MLP models are described as:

$$V_L, V_R = E(I_L), E(I_R), \tag{4}$$

$$V_{aggregation} = Func(V_L, V_R, G_c), \tag{5}$$

$$S_p = F_r(V_{aggregation}), S_c = F_c(V_{aggregation}), \tag{6}$$

Here, $Func$ is a function to combine the vectors in order.

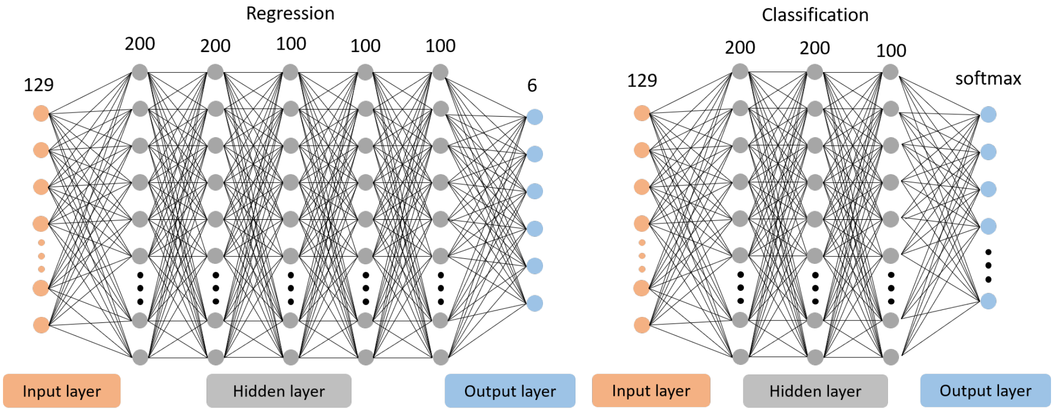


Figure 5. Two MLP models of object recognition. Left: a regression model for 6d pose estimation; Right: a classification model for object categories.

2.3. Data Collection and Training Setups

2.3.1. Data Collection Setup

We built an experimental platform to collect training data efficiently to train the framework above, as shown in **Figure 6**. The designed fingers are mounted on a DH-Robotics AG-160-95 adaptive gripper to replace its tips and pasted AruCo codes on the fingers and grippers to represent their poses. An extra camera is mounted on an optical breadboard to collect the AruCo marker poses, and two cameras in the fingers collect the interaction deformations. The AruCo markers are 4x4 squares of 16mm width with different indexes and are detected by OpenCV [27]. To increase the detection success rate and precision of AruCo markers detection, the outside camera's resolution is set to 1920x1080. The inner camera's resolution is 640x360 and resized to 320x320 to decrease the model's size and prediction time.

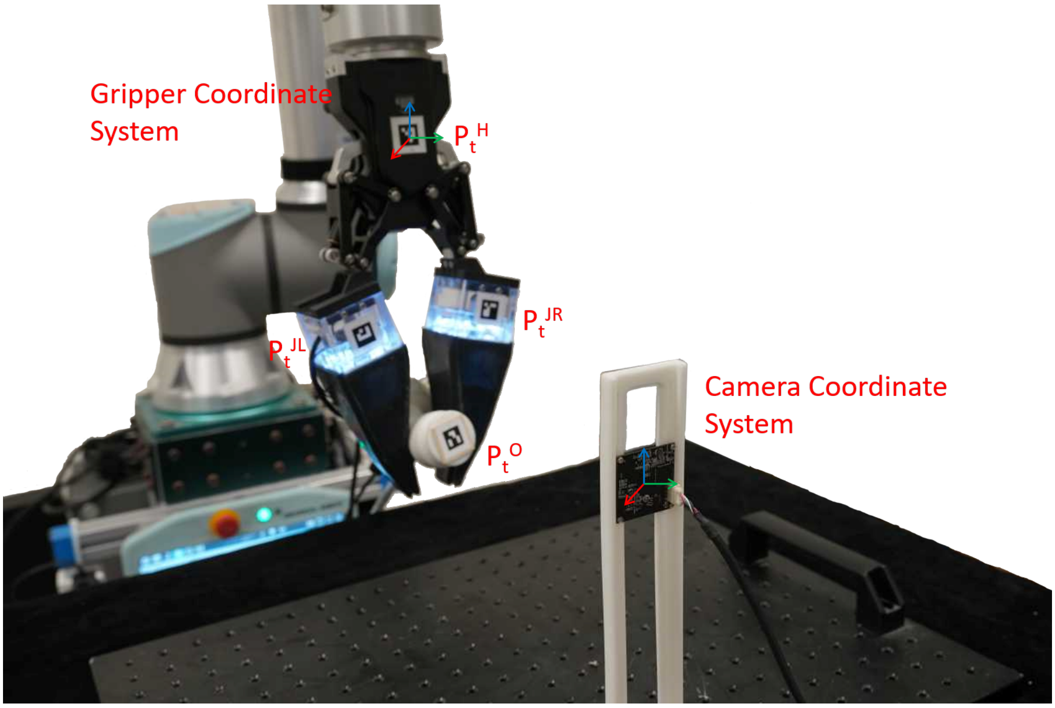


Figure 6. Date collection setup. Four markers are attached to the gripper, fingers, and object. An outside camera monitors the four markers for the object's pose; simultaneously, fingers' deformations are captured by two inner cameras.

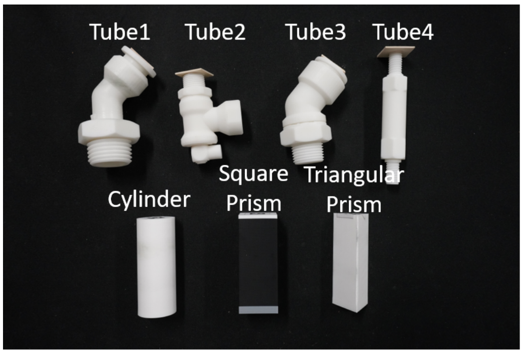
Referring to the article [28], we chose McMaster dataset[1] as our test objects. In addition to the objects from the McMaster dataset, we also chose three basic geometric solids. All objects are resized to adjust the gripper width and 3D-printed for final usage as shown in **Figure 7**. When collecting data, we set the gripper force mode to grasp the object, then shake the object manually to collect the object poses. We collect 5,000 samples for each object.

After collection, all poses are transferred to the gripper coordinate system for standardization. Define $\mathbf{P} = (x, y, z, rx, ry, rz)$ as a pose, here (x, y, z) is translation and (rx, ry, rz) is orientation. Instead of using the object CAD model, we use the relative change to represent the object's pose without the object model. Define reference pose $\mathbf{P}_0 = (x_0, y_0, z_0, rx_0, ry_0, rz_0)$, current pose $\mathbf{P}_t = (x_t, y_t, z_t, rx_t, ry_t, rz_t)$ at time t, the translation matrix $\mathbf{M}_t = [R|T]$, so

$$\mathbf{P}_t = \mathbf{P}_0\mathbf{M}_t, \tag{7}$$

and we use $\mathbf{M}_t$ to represent the current object pose.

[1] https://www.mcmaster.com



object	size (mm)	weight (g)
Tube1	125*85*55	153
Tube2	95*70*35	92
Tube3	115*80*50	118
Tube4	145*20*20	57
Cylinder	100*40*40	73
Square Prism	100*40*40	73
Triangular Prism	100*40*40	36

Figure 7. Test objects and their properties. Left: 3D printed objects; Right: objects' sizes and weights.

The left superscript G and C represent the gripper and camera coordinate system variables. G is the gripper coordinate system, and C is the camera coordinate. In-camera coordinate system, the gripper pose, gripper configuration, and object pose indicated by the Aruco marker attached are ${}^{C}\mathbf{P}_t^O$, gripper poses ${}^{C}\mathbf{P}_t^H$, gripper joint poses ${}^{C}\mathbf{P}_t^{JL}$ and ${}^{C}\mathbf{P}_t^{JR}$ in time t.

Transfer to gripper coordinate system:

$$
\begin{aligned}
{}^{C}\mathbf{P}_t^O &= {}^{C}\mathbf{P}_t^H \cdot {}^{G}\mathbf{P}_t^O, && (8)\\
{}^{G}\mathbf{P}_t^O &= [{}^{C}\mathbf{P}_t^H]^{-1} \cdot {}^{C}\mathbf{P}_t^O, && (9)\\
{}^{G}\mathbf{P}_0^O &= [{}^{C}\mathbf{P}_0^H]^{-1} \cdot {}^{C}\mathbf{P}_0^O, && (10)\\
{}^{G}\mathbf{P}_t^O &= {}^{G}\mathbf{P}_0^O \cdot^{G} \mathbf{M}_t, && (11)
\end{aligned}
$$

In the gripper coordinate system, the object transfer pose ${}^{G}\mathbf{M}_t$ is

$$
\begin{aligned}
{}^{G}\mathbf{M}_t &= [{}^{G}\mathbf{P}_0^O]^{-1} \cdot {}^{G}\mathbf{P}_t^O \\
&= [[{}^{C}\mathbf{P}_0^H]^{-1} \cdot {}^{C}\mathbf{P}_0^O]^{-1}[[{}^{C}\mathbf{P}_t^H]^{-1} \cdot {}^{C}\mathbf{P}_t^O] \qquad (12)
\end{aligned}
$$

The collected dataset comprises seven objects and 5,000 samples per object, each consisting of two inner images, four poses from the outside camera, and objects' categories. The resolution of the inner images is the same and is 640x360, and resized to 320x320 for input, and the values are normalized to 0-1. The objects' pose distributions are shown in **Figure** 8.

2.3.2. Network Training Setup

To improve the reusability and expansibility of the network, we trained the Encoder-Decoder reconstruction and the auxiliary tasks in two stages using the dataset collected in the previous section.

In the first stage, the Encoder-Decoder reconstruction is self-supervised learning. The dataset is randomly split into 8:2; 56,000 images are used for training, and 14,000 are used for evaluation. It was trained with a batch size of 32 using an Adam optimizer with a learning rate 0.001 on mean squared error loss (MSELoss). The latent vector V is set to 8, 16, 32, 64, 128, and 256 to determine the best network configuration. The training epoch is set to 200, and we save the weights with the lowest training loss.

We froze the Encoder's weights in the second stage and only trained the following auxiliary tasks. For the regression model, we trained an MLP model for each object. Using the split dataset above, 4,000 samples are used for training, and 1,000 are used for evaluation for each object. The batch size is 32, the optimizer is Adam optimizer, and the learning rate is 0.001. As the 6D pose consists of two parts, translation and orientation, we define the

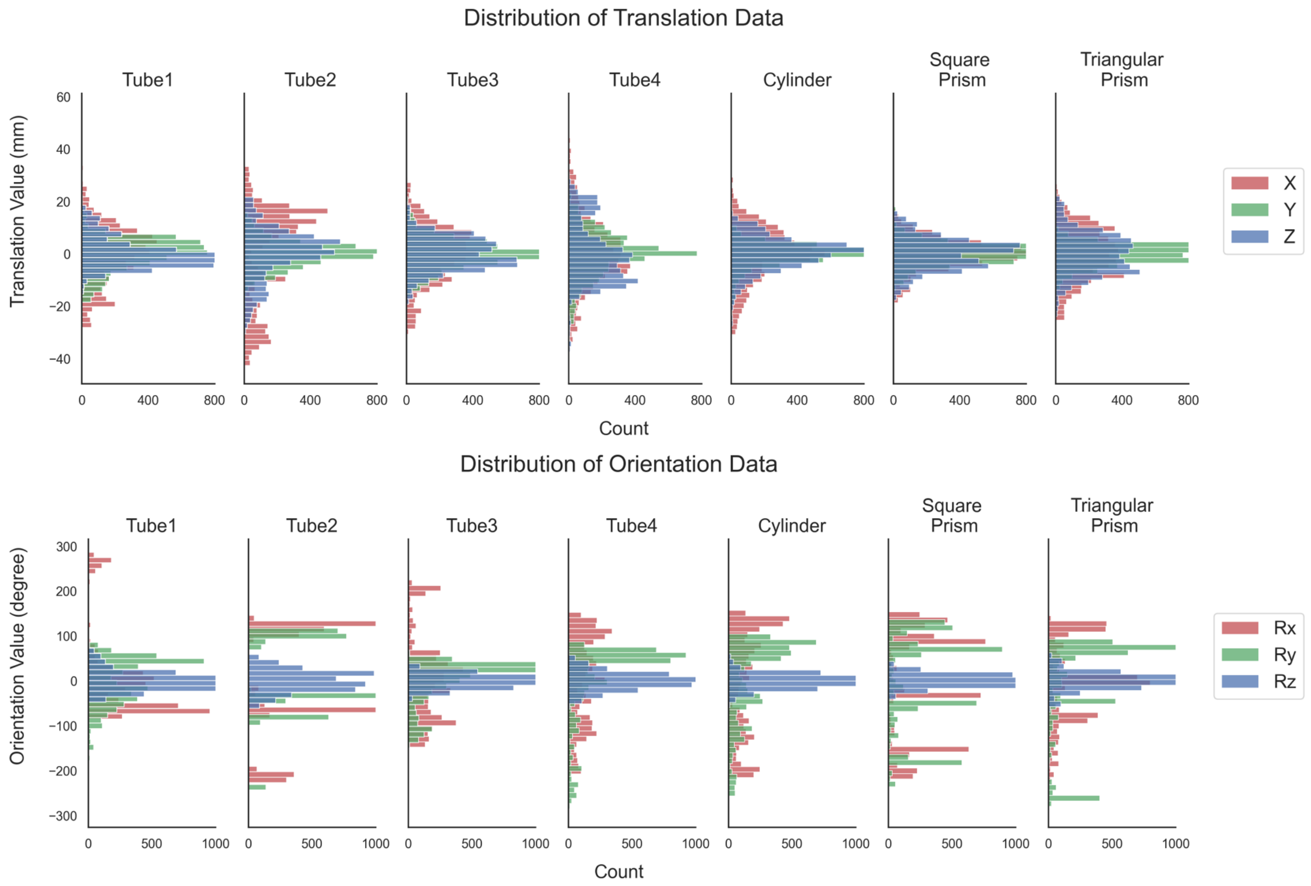


Figure 8. Distribution of the dataset. All data have been normalized by subtracting the mean value, (A) translation distribution, and (B) orientation distribution.

training loss in equations 13-15 where L_t is translation loss, and L_r is orientation loss. The hyper-parameters α and β are set to 0.01 and 10. The training epoch is set to 100.

$$L_t = \frac{1}{3}\sum_{n=1}^{3}(x_n^t - \hat{x}_n^t)^2, \tag{13}$$

$$L_r = \frac{1}{3}\sum_{n=1}^{3}(x_n^r - \hat{x}_n^r)^2, \tag{14}$$

$$L = \alpha L_t + \beta L_r. \tag{15}$$

For the classification model, we trained an MLP model for all objects together. Using the split dataset above, 28,000 samples are used for training and 7,000 for evaluation. The batch size is 256, the optimizer is Adam optimizer, the learning rate is 0.001, and the training loss is cross-entropy loss. The training epoch is set to 100.

3. Results and Discussion

3.1. Dimension of the Latent Vector

To find an optimal dimension of the latent space, we varied the dimension of the latent vector and compared the reconstruction errors using the same training and validation dataset. The dimension of the latent vector is set to 8, 16, 32, 64, 128, and 256, and the corresponding results are shown in **Table** 1.

We scaled all losses such that the loss of 256-dimensional latent space was one. As the dimension increases, the reconstruction loss decreases, and the number of parameters of the model increases. To balance the precision and computational efficiency of the auto-encoder,

Table 1. Effect of the Latent Vector Dimension

	Latent Vector dimension					
	8	16	32	64	128	256
normalized MSELoss	1.37	1.74	1.36	1.09	1.35	1
Parameters (M)	0.57	0.67	0.88	1.29	2.11	3.74

we chose the 64-dimensional latent space, whose loss is comparable to the 256-dimensional space with only 24% of the number of parameters. 181 182

3.2. Quantitative Evaluation of Object Recognition 183

In this section, we report the accuracy of pose estimation and classification. The translation error is measured as the Euclidean distance $\|p_{est} - p_{gt}\|_2$ between the estimated position $p_{est} = (x, y, z)_{est}$ and the ground truth position $p_{gt} = (x, y, z)_{gt}$ [29]. The orientation error $|\alpha|$, measured by an absolute angle error, is computed as:

$$2\cos|\alpha| = \mathrm{Tr}(R_{gt}^{-1}R_{est}) - 1, \tag{16}$$

where R_{gt} and R_{est} are the estimated and ground truth rotation matrices, Tr is the trace of the matrix. 184 185

As shown in **Figure** 9, the translation and orientation errors are significantly different for different objects. The translation error is between 2.02mm and 4.00 mm, and the orientation error is between 11.34 degrees and 31.87 degrees. Object Tube1 has the smallest translation error of 2.02 mm and the smallest orientation error of 11.34 degrees. The object Cylinder has the largest translation error of 4 mm and the largest orientation error of 31.87 degrees. 186 187 188 189 190

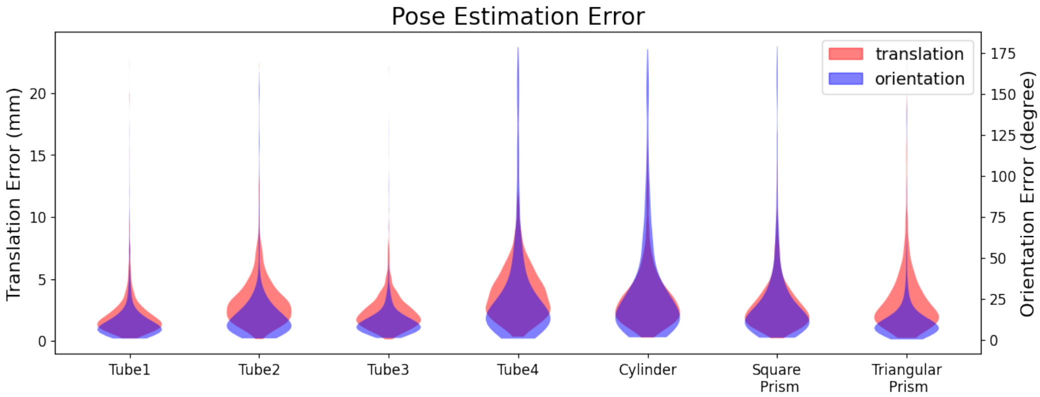


Figure 9. The histogram of pose estimation errors of each object. Translation error is the Euclidean distance, and rotation error is the absolute orientation error $|\alpha|$.

191

The inner camera can only perceive the objects' geometric shape and size as the coating isolates the ambient environment. Objects with complex shapes provide abundant shape features and improve the pose estimation accuracy. On the contrary, the geometric shapes of Tube4, Cylinder, Square prism, and Triangular prism are more similar to a cylinder, resulting in more significant orientation errors among the seven objects. Those objects are symmetrical, but the features around the symmetry axis lack uniqueness, increasing the difficulty of orientation estimation. Comparing the seven objects, the objects' cross-section shape influences translation error. The columnar objects (tube4, cylinder, and prisms) have a more significant translation error to their similar cross-section shape. 192 193 194 195 196 197 198 199 200

Objects' geometric and texture features are essential elements, and the designed finger is limited to obtain the texture as the black coat, influencing the pose estimation precision. A prominent method to improve the precision is to add more features, such as mounting a camera on the gripper or changing the transparent black coating to obtain 201 202 203 204 image features. More features increase the complexity of the device and algorithm but improve performance.

As the objects have unique 3D shapes and sizes, we get a high classification accuracy of 99.05%, as shown in **Figure 10**. With the proposed method, we can estimate the handled object's state with a high accuracy proprioceptive touch of the soft fingers.

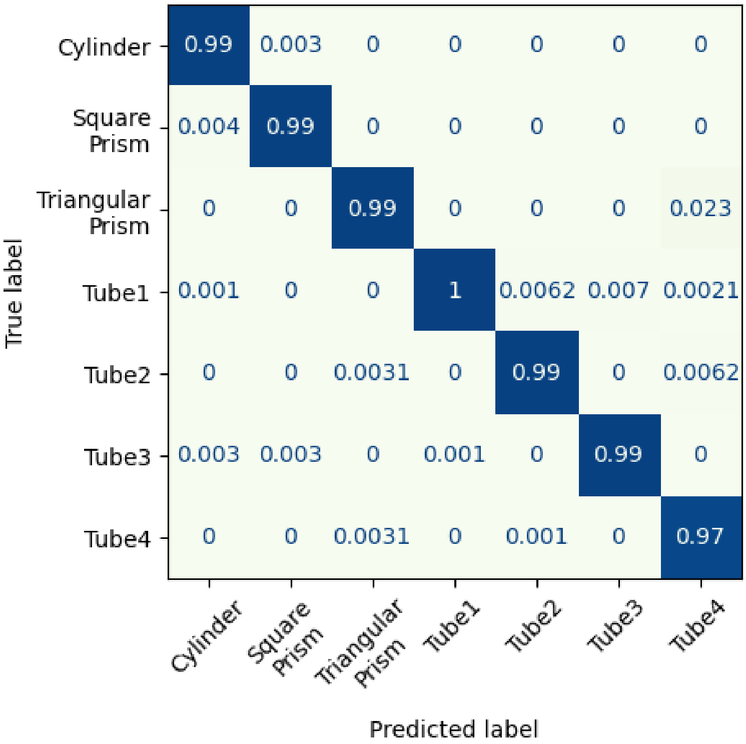


Figure 10. Confusion matrix of objects classification.

3.3. Reusability and Expansibility of the Framework

As described in the framework, the Encoder-Decoder architecture reduces the tactile feature dimensions and unite their format for different type of sensors. This makes the tactile information compact and simplifies the processing flow. For other sensors such as GelSight [30], BioTac[2] and magnetic skin [31], the different sensing information can also be represented as a latent vector with a convolutional neural network, graph neural network, or other methods according to the data structure.

Then, the extracted tactile features are fused depending on the gripper configuration. In this paper, the fusion features combine two-finger images and gripper configuration and are an input of the auxiliary tasks. For an N-finger gripper, we first extract the tactile information of each finger, then fuse each finger's features and the gripper configuration, such as joint rotation angles. The gripper configuration represents the joint's spatial position and can be described as a base pose and the DoF of each finger. As shown in this article, we use AruCo markers to monitor the finger base pose and tactile features of the soft fingers to represent the finger's DoF, which is independent of hardware.

Finally, the fused features are used for downstream tasks. We demonstrate two basic examples: pose estimation and classification of the handled object and get sound results. We can quickly adapt the frame to other tasks using the same fusion features. Benefited from the modular design of the framework, we can extract the tactile features independently, fuse them according to the configuration of the hand, and feed them to different auxiliary

[2] https://syntouchinc.com

task models to complete manipulation tasks; this framework applies to scenarios with multi-sensor, multi-gripper, and multitasking.

4. Conclusions

This paper presents a bio-inspired, soft proprioceptive sensor and a framework for object pose estimation and classification based on the sensor. The proposed soft proprioceptive sensor can be extended to different manipulators, providing extra shape adaptation and interaction information. Based on this sensor, we propose an extendable architecture to extract the tactile information and estimate the handled objects' state. This method achieves a high accuracy of 2.02 mm in translation, 11.34 degrees in orientation, and 99.05% classification accuracy for objects with an unknown CAD model.

The finger is not sensitive to small deformation. The pure black skin on the soft finger loses texture features, and the skeleton filters small shape features, limiting the soft finger to perceiving small objects and distinguishing similar shape objects.

Future work will explore the transferability of this method on different tactile sensors and grippers. This framework provides a uniform feature extractor for different types of tactile sensor information and an extendable structure for different grippers. Meanwhile, more manipulation tasks can be involved with this method.

Author Contributions: Conceptualization, X.L; methodology, X.L., X.H., and N.G.; software, X.L.; validation, X.L.; formal analysis, X.L.; investigation, X.L.; resources, F.W., and C.S.; data curation, X.L.; writing—original draft preparation, X.L.; writing—review and editing, F.W., and C.S.; visualization, X.L., and X.H. supervision, F.W., and C.S.; project administration, F.W., and C.S.; funding acquisition, F.W., and C.S.; All authors have read and agreed to the published version of the manuscript.

Funding: This work was partly supported by the Ministry of Science and Technology of China [2022YFB4701200], the National Natural Science Foundation of China [62206119], Shenzhen Science and Technology Innovation Commission [JCYJ20220818100417038, ZDSYS20220527171403009, SGDX20220530110804030], and Guangdong Provincial Key Laboratory of Human Augmentation and Rehabilitation Robotics in Universities.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Klatzky, R.L.; Lederman, S.J.; Metzger, V.A. Identifying objects by touch: An "expert system". *Perception & psychophysics* **1985**, *37*, 299–302. https://doi.org/10.3758/BF03211351.
2. Dahiya, R.S.; Metta, G.; Valle, M.; Sandini, G. Tactile sensing—from humans to humanoids. *IEEE transactions on robotics* **2009**, *26*, 1–20. https://doi.org/10.1109/TRO.2009.2033627.
3. Boivin, M.; Lin, K.Y.; Wehner, M.; Milutinović, D. Proprioceptive Touch of a Soft Actuator Containing an Embedded Intrinsically Soft Sensor using Kinesthetic Feedback. *Journal of Intelligent & Robotic Systems* **2023**, *107*, 28. https://doi.org/10.1007/s10846-023-01815-4.
4. Zimmermann, C.; Ceylan, D.; Yang, J.; Russell, B.; Argus, M.; Brox, T. Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In Proceedings of the Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 813–822. https://doi.org/10.1109/ICCV.2019.00090.
5. Wan, C.; Probst, T.; Gool, L.V.; Yao, A. Self-supervised 3d hand pose estimation through training by fitting. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 10853–10862. https://doi.org/10.1109/CVPR.2019.01111.
6. Chen, X.; Liu, Y.; Dong, Y.; Zhang, X.; Ma, C.; Xiong, Y.; Zhang, Y.; Guo, X. Mobrecon: Mobile-friendly hand mesh reconstruction from monocular image. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 20544–20554. https://doi.org/10.1109/CVPR52688.2022.01989.
7. Doosti, B.; Naha, S.; Mirbagheri, M.; Crandall, D.J. Hope-net: A graph-based model for hand-object pose estimation. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 6608–6617. https://doi.org/10.1109/CVPR42600.2020.00664.
8. Tekin, B.; Bogo, F.; Pollefeys, M. H+ o: Unified egocentric recognition of 3d hand-object poses and interactions. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 4511–4520. https://doi.org/10.1109/CVPR.2019.00464.

9. Hasson, Y.; Varol, G.; Tzionas, D.; Kalevatykh, I.; Black, M.J.; Laptev, I.; Schmid, C. Learning joint reconstruction of hands and manipulated objects. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 11807–11816. https://doi.org/10.1109/CVPR.2019.01208.
10. Hampali, S.; Sarkar, S.D.; Rad, M.; Lepetit, V. Keypoint transformer: Solving joint identification in challenging hands and object interactions for accurate 3d pose estimation. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 11090–11100. https://doi.org/10.1109/CVPR52688.2022.01081.
11. Mason, M.T. Toward robotic manipulation. *Annual Review of Control, Robotics, and Autonomous Systems* **2018**, *1*, 1–28. https://doi.org/10.1146/annurev-control-060117-104848.
12. Wan, F.; Wang, H.; Liu, X.; Yang, L.; Song, C. DeepClaw: A Robotic Hardware Benchmarking Platform for Learning Object Manipulation. In Proceedings of the 2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM). IEEE, 2020, pp. 2011–2018. https://doi.org/10.1109/AIM43001.2020.9159011.
13. Wang, G.; Manhardt, F.; Tombari, F.; Ji, X. Gdr-net: Geometry-guided direct regression network for monocular 6d object pose estimation. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 16611–16621. https://doi.org/10.1109/CVPR46437.2021.01634.
14. Lipson, L.; Teed, Z.; Goyal, A.; Deng, J. Coupled iterative refinement for 6d multi-object pose estimation. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 6728–6737. https://doi.org/10.1109/CVPR52688.2022.00661.
15. Su, Y.; Saleh, M.; Fetzer, T.; Rambach, J.; Navab, N.; Busam, B.; Stricker, D.; Tombari, F. Zebrapose: Coarse to fine surface encoding for 6dof object pose estimation. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 6738–6748. https://doi.org/10.1109/CVPR52688.2022.00662.
16. Von Drigalski, F.; Taniguchi, S.; Lee, R.; Matsubara, T.; Hamaya, M.; Tanaka, K.; Ijiri, Y. Contact-based in-hand pose estimation using bayesian state estimation and particle filtering. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2020, pp. 7294–7299. https://doi.org/10.1109/ICRA40945.2020.9196640.
17. Chalon, M.; Reinecke, J.; Pfanne, M. Online in-hand object localization. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2013, pp. 2977–2984. https://doi.org/10.1109/IROS.2013.6696778.
18. Pfanne, M.; Chalon, M.; Stulp, F.; Albu-Schäffer, A. Fusing joint measurements and visual features for in-hand object pose estimation. *IEEE Robotics and Automation Letters* **2018**, *3*, 3497–3504. https://doi.org/10.1109/LRA.2018.2853652.
19. Tu, Y.; Jiang, J.; Li, S.; Hendrich, N.; Li, M.; Zhang, J. PoseFusion: Robust Object-in-Hand Pose Estimation with SelectLSTM. *arXiv preprint arXiv:2304.04523* **2023**.
20. Wen, B.; Mitash, C.; Soorian, S.; Kimmel, A.; Sintov, A.; Bekris, K.E. Robust, occlusion-aware pose estimation for objects grasped by adaptive hands. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2020, pp. 6210–6217. https://doi.org/10.1109/ICRA40945.2020.9197350.
21. Álvarez, D.; Roa, M.A.; Moreno, L. Tactile-based in-hand object pose estimation. In Proceedings of the Iberian Robotics conference. Springer, 2017, pp. 716–728. https://doi.org/10.1007/978-3-319-70836-2_59.
22. Yang, L.; Han, X.; Guo, W.; Wan, F.; Pan, J.; Song, C. Learning-based optoelectronically innervated tactile finger for rigid-soft interactive grasping. *IEEE Robotics and Automation Letters* **2021**, *6*, 3817–3824. https://doi.org/10.1109/LRA.2021.3065186.
23. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778. https://doi.org/10.1109/CVPR.2016.90.
24. Wan, F.; Liu, X.; Guo, N.; Han, X.; Tian, F.; Song, C. Visual Learning Towards Soft Robot Force Control using a 3D Metamaterial with Differential Stiffness. In Proceedings of the Conference on Robot Learning. PMLR, 2022, pp. 1269–1278.
25. Nair, V.; Hinton, G.E. Rectified linear units improve restricted boltzmann machines. In Proceedings of the Proceedings of the 27th international conference on machine learning (ICML-10), 2010, pp. 807–814.
26. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International conference on machine learning. pmlr, 2015, pp. 448–456.
27. Bradski, G. The OpenCV Library. *Dr. Dobb's Journal of Software Tools* **2000**.
28. Villalonga, M.B.; Rodriguez, A.; Lim, B.; Valls, E.; Sechopoulos, T. Tactile object pose estimation from the first touch with geometric contact rendering. In Proceedings of the Conference on Robot Learning. PMLR, 2021, pp. 1015–1029.
29. Sattler, T.; Maddern, W.; Toft, C.; Torii, A.; Hammarstrand, L.; Stenborg, E.; Safari, D.; Okutomi, M.; Pollefeys, M.; Sivic, J.; et al. Benchmarking 6dof outdoor visual localization in changing conditions. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 8601–8610. https://doi.org/10.1109/CVPR.2018.00897.
30. Yuan, W.; Dong, S.; Adelson, E.H. Gelsight: High-resolution robot tactile sensors for estimating geometry and force. *Sensors* **2017**, *17*, 2762. https://doi.org/10.3390/s17122762.
31. Yan, Y.; Hu, Z.; Yang, Z.; Yuan, W.; Song, C.; Pan, J.; Shen, Y. Soft magnetic skin for super-resolution tactile sensing with force self-decoupling. *Science Robotics* **2021**, *6*, eabc8801. https://doi.org/10.1126/scirobotics.abc8801.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.