Fast and Precise Detection of Object Grasping Positions with Eigenvalue Templates

Kousuke Mano[1,2], Takahiro Hasegawa[1], Takayoshi Yamashita[1], Hironobu Fujiyoshi[1], and Yukiyasu Domae[2]

Abstract— **Fast Graspability Evaluation (FGE) has been proposed as a method for detecting grasping positions on objects and is now being used for industrial robots. FGE uses convolution of hand templates with regions on the target object to estimate the optimum grasping posture. However, the hand opening width and rotation angles must be set with high resolution to achieve highly accurate results and the computational load is high. To address that issue, we propose a method in which hand templates are represented in compact form for faster processing by using singular value decomposition. Applying singular value decomposition enables hand templates to be represented as linear combinations of a small number of eigenvalue templates and eigenfunctions. Eigenfunctions take discrete values, but response values can be calculated with arbitrary parameters by fitting a continuous function. Experimental results show that the proposed method reduces computation time by two thirds while maintaining the same detection accuracy as conventional FGE for both parallel hands and three-finger hands.**

I. INTRODUCTION

Grasping objects is an important task for industrial robots and robots that support daily life. Implementing that capability requires detection of the optimum grasping positions for the object by using the robot's vision sensors to acquire an RGB image and a depth image. The grasping position detection methods can be categorized into machine learning methods and model fitting methods.

Model fitting for detection of grasping position includes using depth images or using 3D point clouds. The Fast Graspability Evaluation(FGE)[1] uses a depth image for detecting the grasping position using a hand model. For detecting grasping position from a point cloud, there have been proposals of methods in which simple models such as cylinders are fit to the point cloud [2], [3], [4], [5], [6] and methods in which a 3D model of the object to be grasped and a point cloud within the work area are used [7], [8], [9]. The machine learning approaches involve using a support vector machine or neural network to learn the grasping positions of an object by using feature vectors obtained from training images [10], [11], [12]. A method of using convolutional neural networks to detect grasping positions that are even more optimal has also been proposed [14], [15], [16], [17].

Of those methods, FGE can detect grasping positions at high speed by convolution of binary images of the hand model in the target object region. However, achieving highly accurate grasping position detection requires convolution of hand templates with more states, such as the hand opening width and rotation angle, with the binary image. Because computation cost increases with the number of hand states.

We propose a faster grasping position detection method, which introduces eigenvalue templates to FGE. In our method, a hand template group is generated for each region in which the robot hand and the object collide (collision region), and each region in which there is contact (contact region). Applying singular value decomposition to the generated hand template groups makes it possible to represent the groups in compact form and increase the processing speed. The accuracy of detection can also be improved by fitting the eigenfunctions with a continuous function and approximating the hand model to an arbitrary resolution.

II. Fast Graspability Evaluation

Figure 1 shows the overview of the Fast Graspability Evaluation (FGE) [1]. FGE constracts hand collision and contact regions from robot hand models as templates. Those hand regions are convoluted with collision and contact regions of a grasping object, respectively and we obtain the collisionless region of a hand and an object. Then, a graspability map is caliculated by convoluting the collisionless region and the Gaussian filter. The grasping positions are then detected as the positions at which the graspability map has peak values. The optimum grasping posture is detected by varying the resolution of the hand rotation angle and opening width. Formally, the hand collision region (H_c), in which the robot hand collides with the target object, and the hand contact region (H_t) are constructed from a depth image. For coordinates (x,y), the object contact region $O_t(x,y)$ and the object collision region $O_c(x,y)$ are obtained from the object depth image O, the depth to which the hand has proceeded in the grasping approach w, and the height of the target object h by using Eq. (1).

$$\begin{aligned} O_t(x,y) &= \begin{cases} 1 & (O(x,y) \geq h) \\ 0 & (\text{otherwise}) \end{cases} \\ O_c(x,y) &= \begin{cases} 1 & (O(x,y) \geq h-w) \\ 0 & (\text{otherwise}) \end{cases} \end{aligned} \tag{1}$$

The coordinates at which the values of the object and hand model contact region values are maximum and the object and hand model collision region values are minimum can be regarded as the grasping coordinates of high graspability. The contact region for the target object in the hand model

[1],Machine Perception and Robotics Group, Chubu University, 1200 Matsumoto-tho, Kasugai-shi, Aichi, 487-0027, Japan runrun70@mprg.cs.chubu.ac.jp

[2],Automation Research Team, Artificial Intelligence Research Center, Advanced Industrial Science And Technology, 2-4-7 Aomi, Koto-ku, Tokyo, 135-0064, Japan

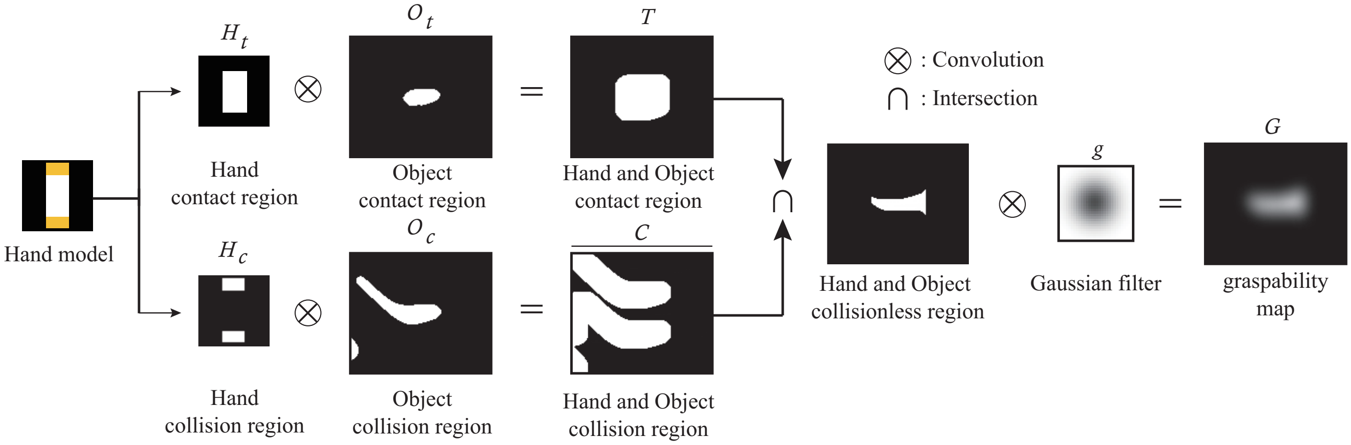


Fig. 1. Overview of the Fast Graspability Evaluation

T can be calculated by convolution of H_t and O_t, and the contact region C can be calculated by convolution of H_c and O_c. Denoting the Gaussian filter as g, the graspability region G for which there is no collision of the hand model and the target object can be obtained with Eq. (2).

$$G = (T \cap \overline{C}) \otimes g \tag{2}$$

The grasping posture can be estimated from the hand parameters and the coordinates at which G is maximum.

A. Issues for FGE

The hand opening width and rotation angle must be taken into account when detecting the grasping position for a parallel hand by FGE. For the values of $\{20\text{mm}, 30\text{mm}, \cdots, 60\text{mm}\}$ for hand opening width h and the values $\{0°, 10°, \cdots, 170°\}$ for hand rotation angle θ, there are 90 templates for the hand collision region and 90 templates for the contact region. In that case, there are 180 convolution operations. For accurate grasping position detection, the number of states is increased to $\{20\text{mm}, 25\text{mm}, \cdots, 60\text{mm}\}$ for the opening width and $\{0°, 5°, \cdots, 175°\}$ for the rotation angle, which results in 324 collision region templates and 324 contact region templates. Thus, the total number of convolution operations is 648 and the computational load is greatly increased.

III. PROPOSED METHOD

In our method, singular value decomposition (SVD) is applied to the large hand template groups to achieve high-speed object grasping position detection with approximate calculations.

A. Calculation of the eigenvalue templates by SVD

Applying SVD to the collision region and contact region of the robot hand makes it possible to approximate the convolution image of the two regions by combining a small number of eigenvalue templates. Doing so makes it possible to approximate the groups of 324 hand templates that result from the extended number of hand opening width values and rotation angle values with linear combinations of a few tens of eigenvalue templates and weight coefficients, thus suppressing the increase in processing time.

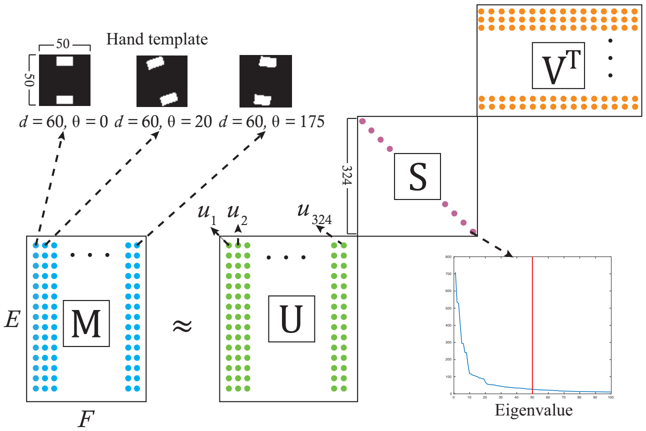


Fig. 2. Singular value decomposition of the hand template group

To apply SVD, the 324 collision region hand templates and the 324 contact region hand templates are respectively represented as the two-dimensional matrices M_c and M_t. The row vectors of the matrices $M_c \in \mathbb{R}^{E\times F}$ and $M_t \in \mathbb{R}^{E\times F}$ are respectively composed of the vectorized 324 hand templates $m_c^{(d,\theta)} \in \mathbb{R}^E$ and $m_t^{(d,\theta)} \in \mathbb{R}^E$. As shown in Figure 2, E is the hand template pixel count (2,500 pixels) and F is the number of robot hand states. Applying SVD to matrices M_c and M_t results in the respective three matrices U, S and V^{T} as shown by Eq. (3).

$$\begin{aligned} M_t &= U_t S_t V_t^{\mathrm{T}} \\ M_c &= U_c S_c V_c^{\mathrm{T}} \end{aligned} \tag{3}$$

Because the same processing is performed for contact region M_t and collision region M_c, the equations for the contact region are omitted in the following description. As shown in Figure 2, the matrices S_t and S_c are diagonal matrices that have the singular values s_{t_i} and s_{c_i} as the diagonal components. Only the upper elements of the singular values have large values; the lower elements have values near zero. Therefore, if we use the contribution rate obtained from the singular values of matrices S_t and S_c and denote the

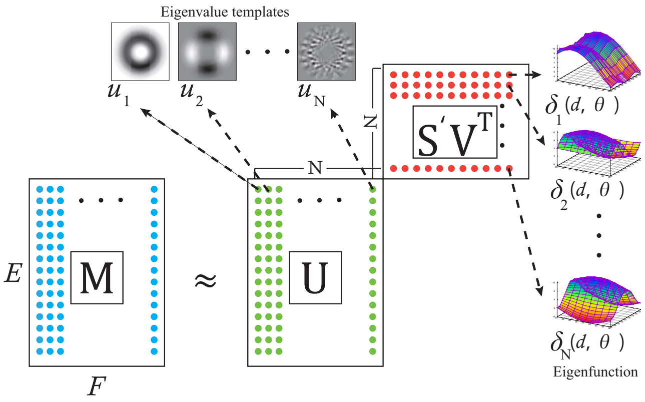


Fig. 3. Compact matrix representation using the contribution ratio

dimension-reduced matrices as S'_t and S'_c, Eq. (3) becomes Eq. (4).

$$M'_t = U_t S'_t V_t^{\mathrm{T}} \tag{4}$$

Let $u_{t_i} \in \mathbb{R}^E$ and $u_{c_i} \in \mathbb{R}^E$ denote the ith column vectors of matrices U_t, and U_c, and let $\delta_{t_i} \in \mathbb{R}^F$ and $\delta_{c_i} \in \mathbb{R}^F$ denote the ith row vectors of matrices $S'V_t^{\mathrm{T}}$ and $S'V_c^{\mathrm{T}}$. The templates $m_t^{(d,\theta)}$ and $m_c^{(d,\theta)}$ for the parameters (d,θ) can then be defined by Eq. (5).

$$m_t^{(d,\theta)} = \sum_{i=1}^{\mathrm{N}} \delta_{t_i}(d,\theta) u_{t_i} \tag{5}$$

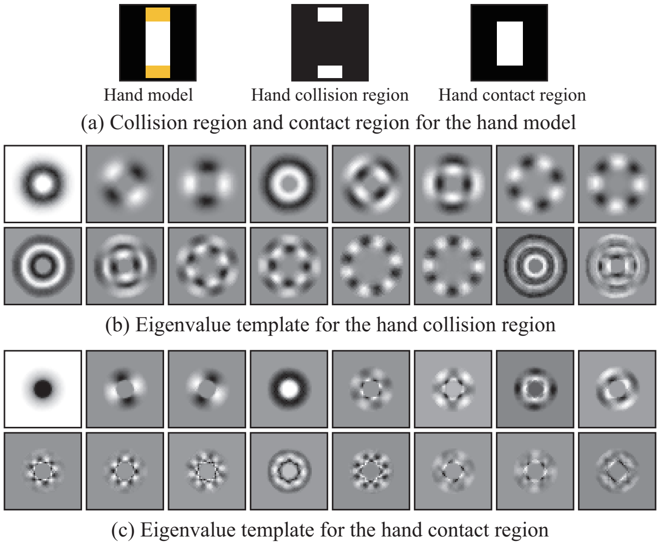
Because u_{t_i} and u_{c_i} can respectively be regarded as two-dimensional template images of a contact region and a collision region (Figure 3), we refer to them here as “*eigenvaluetemplates*” and we refer to the weight coefficients δ_{t_i} and δ_{c_i} as “*eigenfunctions*” An example visualization of an eigenvalue template for a parallel hand is presented in Figure 4 Eigenfunctions δ_{t_i} and δ_{c_i} are discrete functions that have a number of values equal to the number of robot hand templates prior to decomposition The compilation images of the contact region and the collision region ($T'^{(d,\theta)}$ and $C'^{(d,\theta)}$) can be calculated by convolution of the hand templates obtained by approximation of the collision region and contact region of the object (O_t and O_c)

$$T'^{(d,\theta)} = O_t \otimes \sum_{i=1}^{\mathrm{N}} \delta_{t_i}(d,\theta) u_{t_i} \tag{6}$$

B. Eigenfunction fitting with a continuous function

Because the eigenfunctions obtained from SVD (δ_{t_i} and δ_{c_i}) have only discrete values, it is to reconstruct the resolution of the hand movement by using the parameters used for SVD. We therefore apply either linear interpolation or continuous function fitting to the eigenfunctions as off-line processing. With linear interpolation, the hand opening width and rotation values can be approximated with arbitrary resolution. For continuous function fitting, we define the following function models $\gamma_{t_i}(d,\theta)$ and $\gamma_{c_i}(d,\theta)$ in the same way as described in reference [18].

$$\gamma_{t_i}(d,\theta) = \sum_{j=0}^{\mathrm{J}} \sum_{k=0}^{\mathrm{K}} \alpha_{j,k} d^j \cos(k\theta) + \sum_{j=0}^{\mathrm{J}} \sum_{k=1}^{\mathrm{K}} \beta_{j,k} d^j \sin(k\theta) \tag{7}$$



(a) Collision region and contact region for the hand model

(b) Eigenvalue template for the hand collision region

(c) Eigenvalue template for the hand contact region

Fig. 4. Visualization of the eigenvalue template for a parallel hand

In the above equation, J and K represent the numbers of hours for the continuous function models, and $\alpha_{j,k}$ and $\beta_{j,k}$ are unknown coefficients. The unknown constants are calculated as the following minimization problem.

$$\begin{gathered} \underset{\alpha,\beta}{\arg\min}\Big(\sum_{d}\sum_{\theta}(\delta_{t_i}(d,\theta) - \gamma_{t_i}(d,\theta))^2\Big), \\ d = \{20, 25, 30, \cdots, 60\}, \\ \theta = \{0, 5, 10, \cdots, 175\} \end{gathered} \tag{8}$$

We set $\mathrm{J} = 4$ and $\mathrm{K} = 12$ to approximate the original eigenfunctions. Representing the eigen function with continuous function models enables as to reconstruct the hand templates to arbitrary resolution.

C. Efficient graspability calculation with eigenvalue templates

The process of detecting grasping position by the proposed method is illustrated in Figure 5. Because the binary images of the target object (O_t and O_c) in the eigenvalue templates (u_{t_i} and u_{c_i}) can be convoluted in advance as $q_{t_i} = O_t \otimes u_{t_i}$ and $q_{c_i} = O_c \otimes u_{c_i}$, substitutions can be made in Eq. (6) as shown in Eq. (9) in the first step. In the second step, it is possible to efficiently calculate the output for an arbitrary hand opening width d and rotation angle θ by changing only the values of the eigenfunctions as shown in Eq. (9) for the contact region or the collision region ($T'^{(d,\theta)}$ or $C'^{(d,\theta)}$).

$$T'^{(d,\theta)} = \sum_{i=1}^{\mathrm{N}} \gamma_{t_i}(d,\theta) q_{t_i} \tag{9}$$

In the third step, the graspability map $G^{(d,\theta)}$ can be obtained by convolution with the Gaussian filter g as shown by Eq. (10).

$$G^{(d,\theta)} = (T'^{(d,\theta)} \cap \overline{C'^{(d,\theta)}}) \otimes g \tag{10}$$

The positions that have the maximum values in the graspability map are taken as the grasping positions.

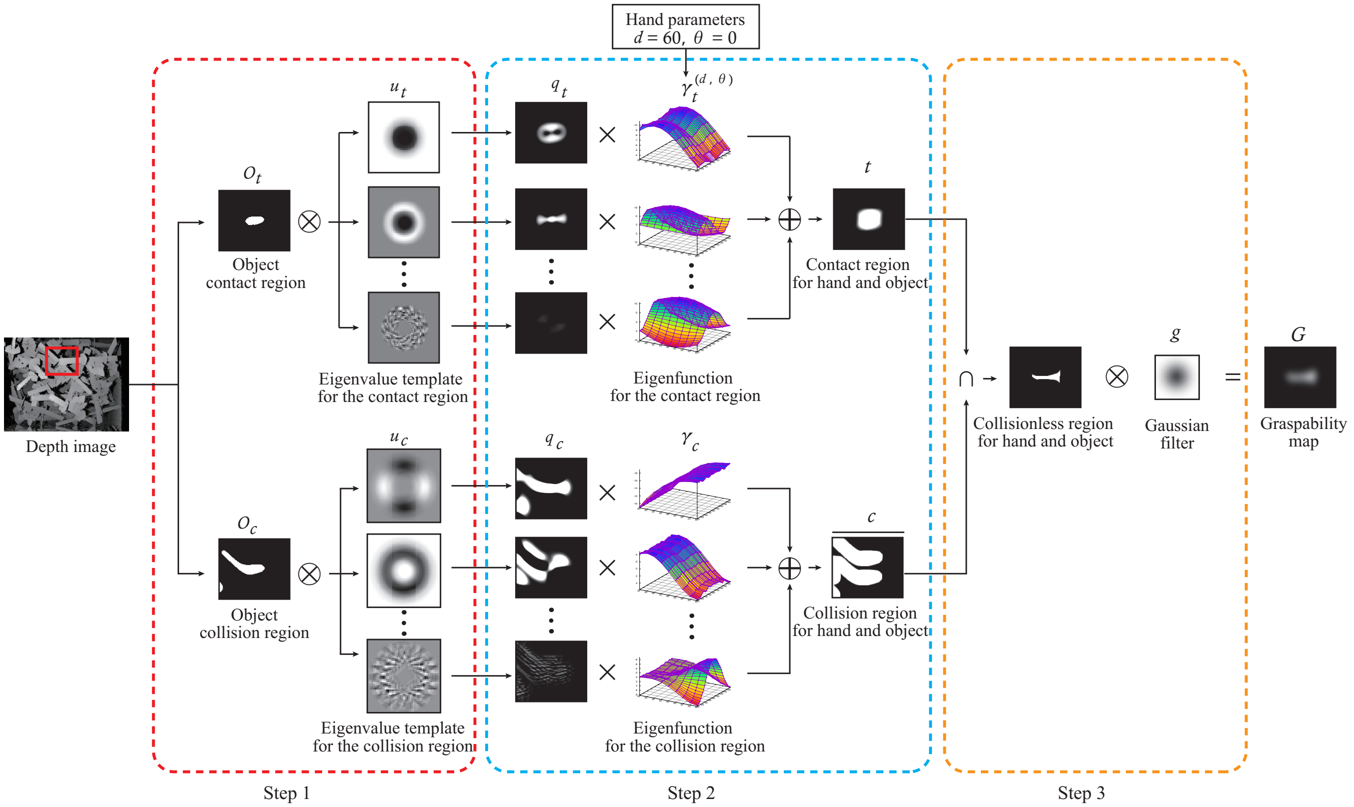
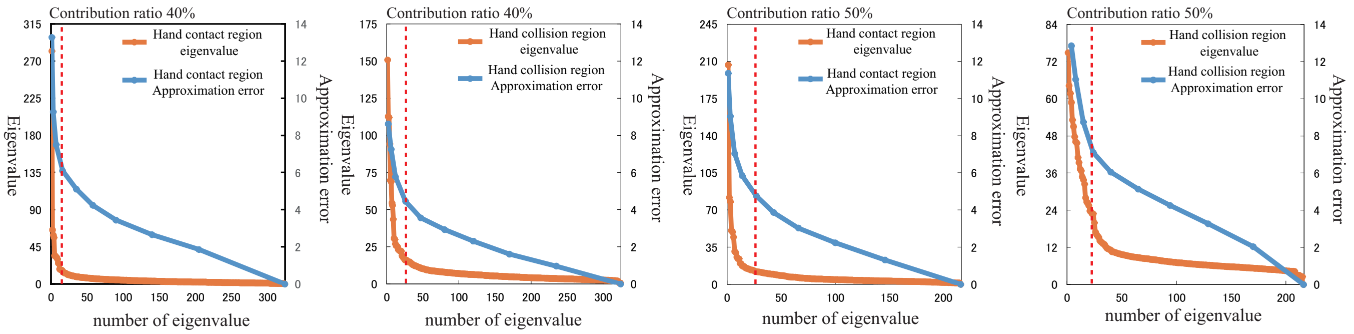

Hand parameters
$d = 60,\ \theta = 0$
u_t
q_t
$\gamma_t^{(d,\theta)}$
t
O_t
Object contact region
Contact region for hand and object
Eigenvalue template for the contact region
Eigenfunction for the contact region
Depth image
u_c
q_c
γ_c
c
O_c
Object collision region
Collision region for hand and object
Eigenvalue template for the collision region
Eigenfunction for the collision region
g
G
Collisionless region for hand and object
Gaussian filter
Graspability map
Step 1
Step 2
Step 3

Fig. 5. Grasping position detection by the proposed method



(a) Parallel hand contact region (b) Parallel hand collision region (c) Three-finger hand contact region (d) Three-finger hand collision region

Fig. 6. Relationship of the hand approximation error and eigenvalue

IV. Experimental results

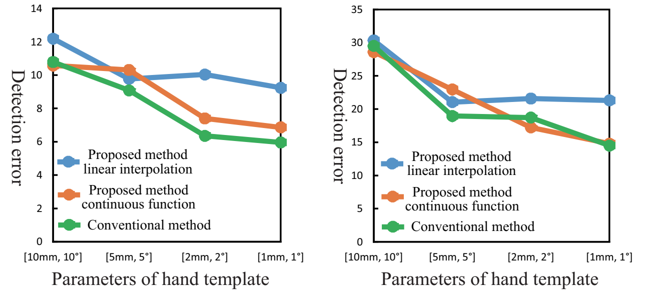
To evaluate the effectiveness of the proposed method, we compared the detection error and the processing time with the results for FGE.

A. Overview

We used 34 images of two types of bulk industrial components. The size of depth image is 640 × 480 pixels. The computer used in the experiment had a 3.40 GHz Intel core I7 6700. The range for the opening width was from 20 mm to 60 mm for both the parallel hand and the three-finger hand. The range for the rotation angle was from 0° to 175° for the parallel hand and from 0° to 115° for the three-finger hand. Parameters of hand template (i.e., opening width and totation angle) and denoted as $[d,\ \theta]$. In case that parameters were 5mm for opening width and 5° for rotation angle, parameters are denoted as [5mm, 5°] Using the FGE with [0.1mm, 0.1°] as reference values, we evaluated the detection error by varying the resolution for parallel hand and three-finger hand models. For the proposed method, SVD was performed using parameters [5mm, 5°]. The contribution ratios were calculated for the matrices S_c and S_t, and the optimum contribution rate was determined from the hand template approximation error and the singular value by varying the contribution rate.

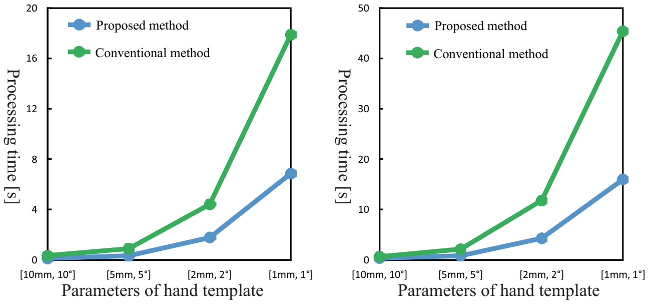
The approximation error was calculated by applying the Frobenius norm for the hand template and the hand template obtained by approximation (Eq. 11).

$$Approximation\ Error = \sqrt{\sum_i^I \sum_j^J (H_t^{(d,\theta)} - m_t^{(d,\theta)})^2} \qquad (11)$$



(a) Detection error for a parallel hand (b) Detection error for a three-finger hand

Fig. 7. Comparison of detection error for changes in resolution



(a) Processing time for a parallel hand (b) Processing time for a three-finger hand

Fig. 8. Comparison of processing time for changes in resolution

The grasping position detection error was calculated from the reference coordinates $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$ and the detection coordinates $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ using Eq. (12) as shown in Figure 9.

$$Detection\ Error = \sqrt{(\boldsymbol{R}_1 - \boldsymbol{A}_1)^2 + (\boldsymbol{R}_2 - \boldsymbol{A}_2)^2} \tag{12}$$

B. Approximation accuracy by changing contribution ratio

The relationship between the approximation error of the hand template and the singular value is shown in Figure 6 (a) and (b) for the parallel hand and in Figure 6 (c) and (d) for the three-finger hand. In the figures, the left vertical axis represents the singular value and the right vertical axis represents the approximation error of the hand template. For a contribution ratio of 90%, the singular value is often required for both the parallel hand and the three-finger hand, so a contribution rate that is before the point at which the change in singular value becomes constant and the approximate error changes greatly is used. As seen in Figure 6 (a) and (b), 16 singular values are used for the contact region of the parallel hand and 26 values are used for the collision region. In that case, the contribution ratio is 40%. From Figure 6 (c) and (d), we see that 27 singular values are used for the three-finger hand contact region and 40 are used for the collision region. In that case, the contribution ratio is 50%.

C. Comparison of detection accuracy and processing time

The effect of varying the resolution on the detection error for the proposed method and the conventional method (FGE) is shown in Figure 7. The grasping posture estimation time for one item is shown in Figure 8. From Figure 7, we can see that the accuracy is the same for both methods for both the parallel hand and the three-finger hand with linear interpolation of the eigenfunctions at [10mm, 10°] and [5mm, 5°]. For [2mm, 2°] and [1mm, 1°], however, the detection accuracy is the same as for [5mm, 5°], which indicates that linear interpolation of the eigenfunctions contributes little to improvement of grasping position accuracy. With continuous function fitting of the eigenfunctions, on the other hand, the detection error of the proposed method is the same as for the conventional method for [2mm, 2°] and [1mm, 1°] for both the parallel hand and the three-finger hand. We can therefore conclude that fitting the eigenfunctions with continuous functions is an effective method. From Figure 8(a), we can see that the proposed method can reduce the processing time for parallel hand grasping posture estimation by about 62% for [5mm, 5°] and by about 61% for [1mm, 1°]. From Figure 8(b), we can see reductions for the three-finger hand of about 45% for [5mm, 5°] and about 64% for [1mm, 1°]. These results demonstrate the possibility of achieving both high speed and high accuracy by continuous function fitting.

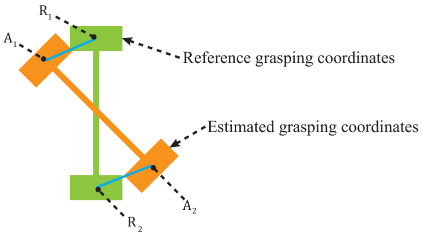

R1
A1
Reference grasping coordinates
Estimated grasping coordinates
A2
R2

Fig. 9. Explanation of the valuation equation

D. Example of grasping position detection

An example of grasping position detection for bulk parts is shown in Figure 10, where the red and green coloring indicates the estimated grasping posture. There are no large deviations of the grasping positions that result from the proposed method and the FGE method for parallel hands and three-finger hands. We can thus consider the detection error of the proposed method shown in Figure 7 to be acceptable.

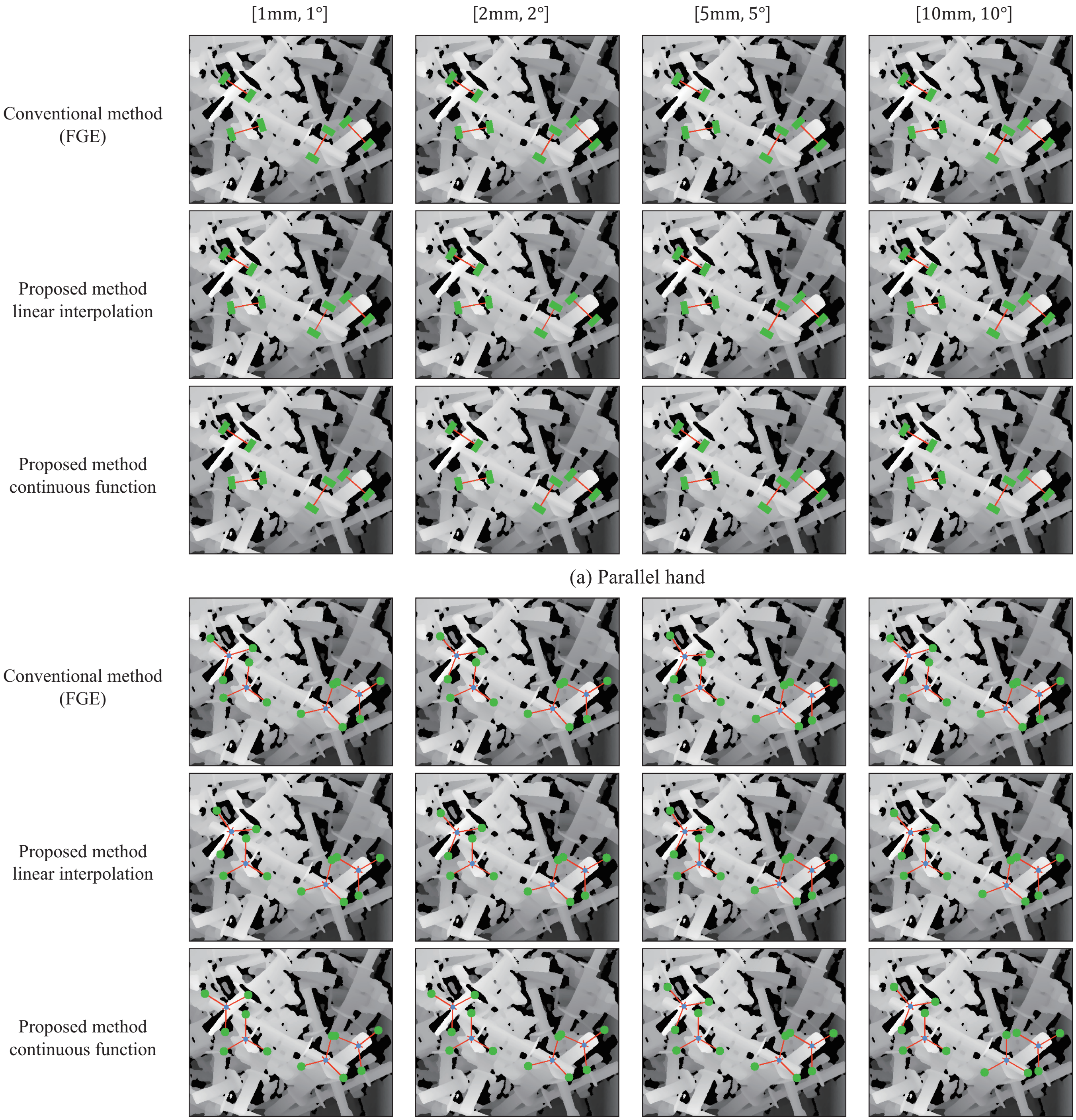
V. CONCLUSION

We proposed the use of eigenvalue templates for faster high-resolution grasping posture estimation. Introducing eigenvalue templates for approximate calculation enables as to reduce computational cost about one third while maintaining accuracy. Fitting the eigenfunctions with continuous functions achieves higher accuracy in the detection of grasping positions. In future work, we will evaluate this method with a robot manipulator.

Acknowledgement. This work was carried out based on the budget of NEDO “next generation artificial intelligence & robot core technology development”.

References

[1] 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.


[1mm, 1°]
[2mm, 2°]
[5mm, 5°]
[10mm, 10°]
Conventional method (FGE)
Proposed method linear interpolation
Proposed method continuous function
Conventional method (FGE)
Proposed method linear interpolation
Proposed method continuous function

(a) Parallel hand

(b) Three-finger hand

Fig. 10. Example of grasping position detection by the proposed method and FGE

[2] K. Harada, K. Nagata, T. Tsuji, N. Yamanobe, A. Nakamura, and Y. Kawai, "Probabilistic approach for object bin picking approximated by cylinders," International Conference on Robotics and Automation, pp. 3742–3747, 2013.

[3] M. Nieuwenhuisen, D. Droeschel, D. Holz, J. Stuckler, A. Berner, J. Li, R. Klein, and S. Behnke, "Mobile bin picking with an anthropomorphic service robot," International Conference on Robotics and Automation, pp. 2327–2334, 2013.

[4] M. Berger, G. Bachler, and S. Scherer, "Vision guided bin picking and mounting in a flexible assembly cell," Industrial Engineering Applications of Artificial Intelligence and Expert systems, vol. 1821, pp. 109–117, 2000.

[5] N. Curtis and J. Xiao, "Efficient and Effective Grasping of Novel Objects through Learning and Adapting a Knowledge Base," Intelligent Robots and Systems, pp. 2252–2257, 2008.

[6] K. Nagata, T. Miyasaka, D.N. Nenchev, N. Yamanobe, K. Maruyama, S. Kawabata, and Y. Kawai, "Picking up an Indicated Object in a Complex Environment," Intelligent Robots and Systems, pp. 2109–2116, 2010.

[7] B. Drost, M. Ulrich, N. Navab, and S. Ilic,"Model globally, match

locally: Efficient and robust 3D object recognition," Computer Vision and Pattern Recognition, pp. 998–1005, 2010.

[8] C. Choi, Y. Taguchi, O. Tuzel, M. Liu, and S. Ramalingam,"Voting-based pose estimation for robotic assembly using a 3D sensor," International Conference on Robotics and Automation, pp. 1724–1731, 2012.

[9] D. Chetverikov, D. Svirko, D. Stepanov, and P. Krsek,"The trimmed iterative closest point algorithm," International Conference on Pattern Recognition, pp. 545–548, 2002.

[10] Y. Jiang, S. Moseson, and A. Saxena, "Efficient grasping from rgbd images: Learning using a new rectangle representation," International Conference on Robotics and Automation, pp. 3304–3311, 2011.

[11] T. Joachims, "Optimizing search engines using clickthrough data," Knowledge Discovery and Data Mining, pp. 133—142, 2002.

[12] I. Lenz, H. Lee, and A. Saxena, "Deep Learning for Detecting Robotic Grasps," International Journal of Robotics Research, vol. 34, no. 4-5, pp. 705–724, 2015.

[13] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-Based Lerning Applied to Document Recognition," Proceedings of the IEEE, vol, 86, pp. 2278–2324, 1998.

[14] J. Redmon, and A. Angelova, "Real-time grasp detection using convolutional neural networks," International Conference on Robotics and Automation, pp. 1316–1322, 2015.

[15] J. Mahler, M. Matl, X. Liu, A. Li, D. Gealy, and K. Goldberg,"Dex-Net 3.0: Computing Robust Vacuum Suction Grasp Targets in Point Clouds using a New Analytic Model and Deep Learning,"International Conference on Robotics and Automation, pp. 1–8, 2018.

[16] A. Umar, T. Jianbin, and H. Stefan, "GraspNet: An Efficient Convolutional Neural Network for Real-time Grasp Detection for Lowpowered Devices," International Joint Conference on Artificial Intelligence, pp. 4875–4882, 2018.

[17] J. Edward, L. Stefan, and Andrew, D. j.,"Deep learning a grasp function for grasping under gripper pose uncertainty," Intelligent Robots and Systems, pp. 4461–4468, 2016.

[18] T. Hasegawa, M. Ambai, K. Ishikawa, G.Koutaki, Y. Yamauchi, T. Yamashita, and H. Fujiyoshi, "Multiple-hypothesis affine region estimation with anisotropic LoG filters," International Conference on Computer Vision, pp. 585–593, 2015.