Asaoka *et al. Robomech J (2018) 5:23*
https://doi.org/10.1186/s40648-018-0118-6

ROBOMECH Journal

RESEARCH ARTICLE

Open Access

Detection of object arrangement patterns using images for robot picking

Tadashi Asaoka[1*], Kazuyuki Nagata[2], Takao Nishi[2] and Ikuo Mizuuchi[1]

Abstract

This paper focuses on robot picking of objects in warehouses and stores. Objects are often regularly stacked or aligned in specific arrangement patterns to increase storage efficiency. There are typical patterns in arrangement patterns. A specific picking strategy set is often linked to specific arrangement patterns. By linking the arrangement patterns of various object categories to picking strategy sets, the picking performance of a robot is expected to improve. In this paper, we propose a method in which groups of regularly arranged objects are detected from an image, and the arrangement pattern of each group is identified. In this paper, we describe the effectiveness of the proposed method based on experiment results for "book" as the target.

Keywords: Object arrangement pattern, Robot picking, Deep learning

Introduction

Automated object picking by robots in warehouses and stores is expected to become a reality. Distribution warehouses of Amazon.com, Inc. are currently using the mobile shelf Kiva Pod, which stores objects inside, and delivers objects automatically for packaging. However, objects are still picked from the shelf by hand. Therefore, including Amazon Robotics Challenge [1, 2], there have been many researches on robot picking of objects.

For a robot to pick an object, picking strategies must be created. The picking strategies of a robot depend on (1) the hardware of the robot hands, (2) the shape, (3) the use [3], and (4) the arrangement of the object picked by the robot. With regard to (1), there are various types of robot hands being developed (e.g., multi-fingered hands [4] and vacuum hands [1]). Whereas, (2) and (3) depend on the type of the object. There have been many researches on the detection of general objects using deep learning. For example, R-CNN [5], Fast R-CNN [6], Faster R-CNN [7], YOLO [8], and SSD [9] have been proposed. Based on the achievements of these methods, many researches have been conducted on the robot picking of diverse types of objects. Lenz et al. [10] derived the grasping posture of robots using deep learning from RGB-D images of various objects. Mahler et al. [11, 12] built a system (Dex-Net) that selects the suitable picking by comparing 3D data of the target object with the shapes of the objects that were previously picked. Using deep learning, Levine et al. [13] predicted the probability of successful picking of diverse objects stacked in bulk from monocular camera images.

This paper focuses on the arrangement of the objects. Previous researches on robot picking, which depends on the arrangement of the objects, have mainly handled random bin picking problems (e.g., [10–23]). The random picking indicates the problem in which randomly arranged objects, as shown in Fig. 1, are picked automatically. With random picking, a picking strategy must be created for each object. In random bin picking, objects adjacent to the target object are considered obstacles. Thus the object for picking is handled as a simplex (objects are handled as a complex in this paper as described below). For a target object surrounded by obstacles, Domae et al. [15] derived the grasping posture of a robot that does not collide with the obstacles. When picking objects that are stacked in bulk, Harada et al. [21] predicted that, using machine learning, a gripper will be able to successfully pick the target object even when in contact with adjacent objects.

*Correspondence: t-asaoka@aist.go.jp
[1] 2-24-16 Naka-cho, Koganei-shi, Tokyo 184-8588, Japan
Full list of author information is available at the end of the article

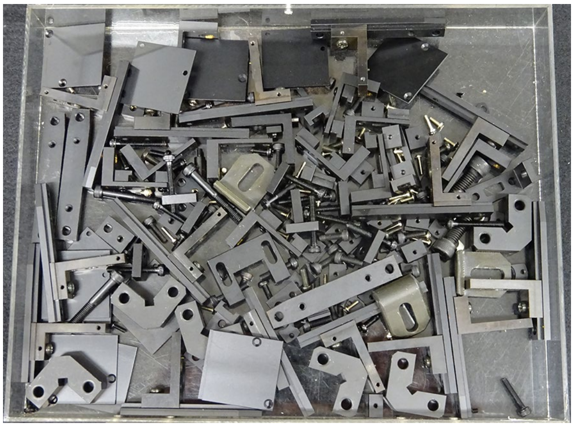


Fig. 1 Randomized objects

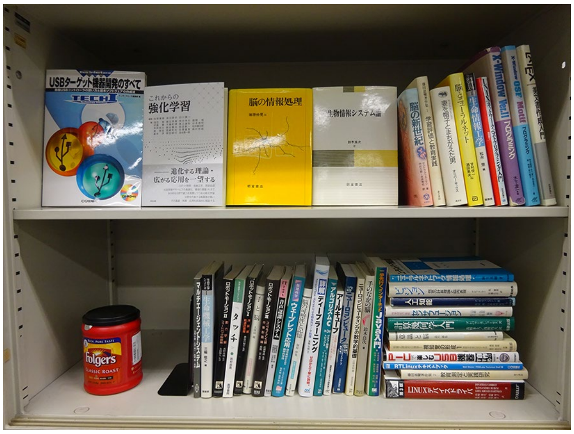


Fig. 2 Regularly arranged books

On the other hand, as shown in Fig. 2, objects are often and regularly arranged in specific arrangement patterns, such as stacked or aligned, in order to increase storage efficiency. When objects are regularly arranged, a common picking strategy set can be applied to the objects. Thus, when picking regularly arranged objects, a set of objects can be handled as a complex.

As shown in Fig. 3, there are the typical patterns of object arrangements. As shown in Fig. 4, in many cases, a specific picking strategy set can be linked to a specific arrangement pattern. In other words, if the category and the arrangement pattern of objects can be detected from an image, a picking strategy set is expected to be created for a robot. To the best of the authors' knowledge, there have been no researches on robot picking that have focused on the object arrangement patterns as described above.

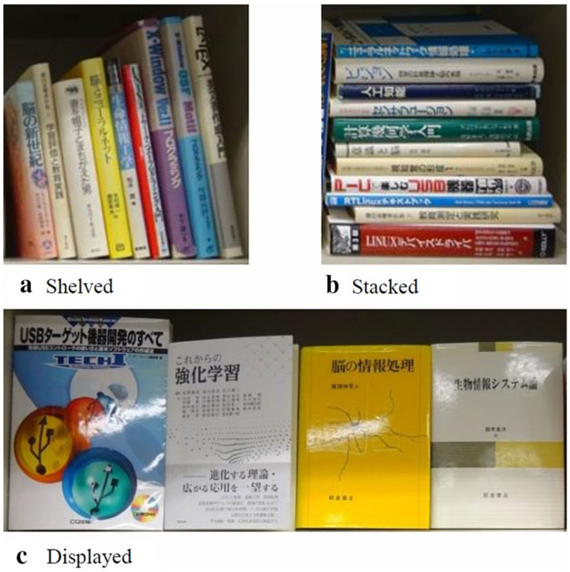
In this paper, we propose a method through which the areas of objects that are regularly arranged are detected from an image, and the arrangement pattern of the each area is identified. In the proposed method, we use a general object detection method based on deep learning. Based on the general object detection, the objects of diverse categories can be detected. An object area is detected as a bounding box (BB), and arrangement patterns are identified based on the BBs for each object category. In "Relationship between object arrangement patterns and picking strategies" section, we discuss the relationship between the arrangement patterns and the picking strategy in detail. In "Detecting BB of object for picking" section, we discuss a method for detecting the object area using a BB. In "Extraction of the area for a group of regularly arranged objects" section, we discuss a method for extracting a group of objects that are arranged in a regular manner. In "Identifying arrangement patterns" section, we discuss a method for identifying the arrangement patterns. In "Evaluations" section, we provide some example results. In "Discussion" section, we discuss the results of the previous section. For simplification, we present the results of only three types of arrangement patterns for books (Fig. 3) in this paper.

Relationship between object arrangement patterns and picking strategies

The objects are usually picked through prismatic or circular precision grasps [24]. The prismatic precision grasp is a grasp in which the opposing faces of the target object are grasped with two fingers (or virtual fingers; as there are cases where one surface may be grasped by more than one fingers, all those fingers can be considered as one virtual finger). The circular precision grasp is a grasp in which the circumference of a spherical or cylindrical object is grasped using three fingers (or virtual fingers).

As shown in Fig. 3, typical patterns can be found in the arrangement of objects (shelved, C_h; stacked, C_v; displayed, C_f). With these arrangement patterns, because objects are in contact with or in close proximity to each other, the picking surfaces of the object (for the prismatic precision grasp) are hidden. Therefore, when humans are picking an object for a specific arrangement pattern, a specific picking strategy is often observed as shown in Fig. 4. First, to obtain the picking surfaces, the target object is manipulated through a grasp-less manipulation (tilting for shelved and displayed objects, and sliding for stacked objects). Then the picking surfaces obtained are grasped.

In cognitive science, the concept of affordance is often considered [25], which is an idea in which information is ubiquitous within an environment. The affordance refers to the possibility of a specific behavior based on

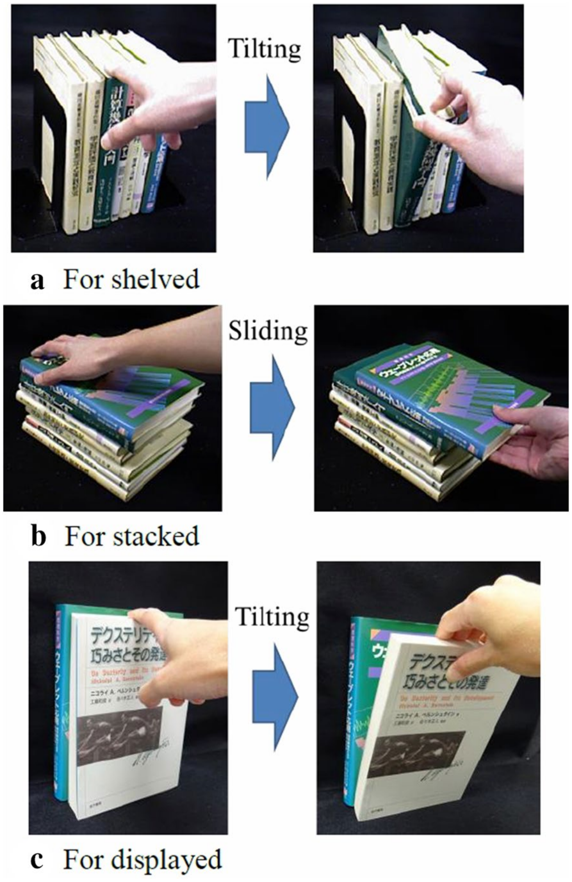


a Shelved

b Stacked

c Displayed

Fig. 3 Typical arrangement patterns



a For shelved

b For stacked

c For displayed

Fig. 4 Picking strategies for arrangement patterns

the environment. According to this idea, instead of living organisms arbitrarily expressing a behavior, the environment affords the possibility of expressing the specific behavior on living organisms, and as living organisms receive the possibility, the behavior is expressed.

There have been many researches [26, 27] conducted on robot picking based on the affordance. Using Fig. 4 as examples, it is supposed that not only the types but also the arrangement patterns of objects afford the picking behavior. Human behaviors provide many suggestions regarding the behavior generation of a robot with a physical structure similar to that of a human. Therefore, it is supposed that detecting arrangement patterns from images is meaningful for robot picking.

Methods

Deep learning shows high performance in general object detection. In deep-learning-based general object detection, the position (area) and category of objects are detected by learning a large amount of training data for each object category. Assuming that the area and category (type of arrangement patterns) of a group of regularly arranged objects are detected on the basis of the idea of the general object detection, it is easily predicted that a huge amount of training data ({object category} × {arrangement pattern}) is required.

In the proposed method, only training data set for each object category is required by using the pipeline process, as shown in Fig. 5. Also, in the proposed method, it is expected that the processing of the latter stages (the area extraction and the arrangement pattern identification) will be simplified, since the processing of the latter stages is executed for each object category. Each processing is described below.

Detecting BB of object for picking

In the proposed method, we first detect the object to be grasped. The area (BB) and the category of the object are detected. In this paper, we detect the object using Single Shot MultiBox Detector (SSD) [9].

In general object detection including the SSD, non-maximum suppression (NMS) is applied such that multiple BBs are not detected for the same object (Fig. 6). In the proposed method, we use Soft-NMS [28]. Each BB B_i has the probability of belonging to an object category (category score) s_i. In the end, only a BB with a score s_i of the threshold or higher is produced. In a standard NMS (greedy-NMS), when two BBs in the same category (B_i and B_j) have a high overlapping rate, by giving the BB with the smaller score s_i ($s_i < s_j$) a value of zero,

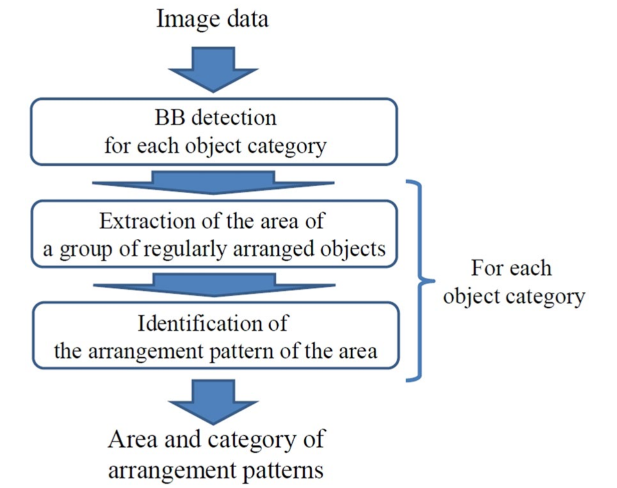
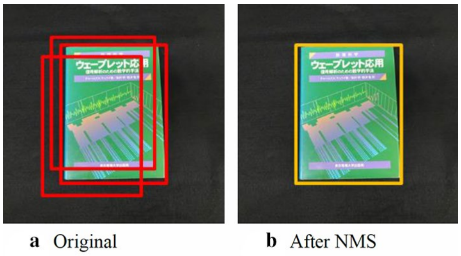


Image data
BB detection for each object category
Extraction of the area of a group of regularly arranged objects
Identification of the arrangement pattern of the area
For each object category
Area and category of arrangement patterns

Fig. 5 Overview of the proposed method



a Original **b** After NMS

Fig. 6 NMS

the detection of multiple BBs for the same object is prevented [28], as shown in Eq. (1).

$$\text{if } s_i < s_j, \quad \text{then } s_i = \begin{cases} s_i & \text{if } \mathrm{iou}(B_i, B_j) < O_{th} \\ 0 & \text{if } \mathrm{iou}(B_i, B_j) \geq O_{th} \end{cases} \tag{1}$$

where iou(·) is a function that expresses the intersection over union (IoU), and O_{th} is the threshold of the overlapping rate. If area function is represented as area(·), iou(·) is expressed using Eq. (2) (Fig. 7).

$$\mathrm{iou}(B_i, B_j) = \frac{\mathrm{area}(B_i \cap B_j)}{\mathrm{area}(B_i \cup B_j)} \tag{2}$$

Therefore, as shown in Fig. 8a, if objects are adjacent to each other, the BB of an adjacent object may not be detected (Fig. 9).

Thus, instead of the greedy-NMS, we used the soft-NMS. With the soft-NMS, the score s_i is attenuated according to the overlapping rate, as shown in Eq. (3).

$$\text{if } s_i < s_j, \text{then } s_i = s_i \exp\left\{-\frac{\mathrm{iou}(B_i, B_j)^2}{\sigma_s}\right\} \tag{3}$$

where σ_s is a parameter that determines the rate of the attenuation. Using the soft-NMS, the detection of multiple BBs for the same object can be prevented, and the BB of adjacent objects can be detected (Fig. 9).

Figure 8a shows the detection results of object areas belonging to the book category. In this paper, we used a model fine-tuned from an existing model [29] pre-trained with the MS-COCO dataset [30]. The images contained in each red BB in Fig. 8a are shown in Fig. 8b as examples. The extraction precision of areas belonging to the book category is low (the number of books and the number of BBs do not match). This issue is discussed in "BB detection" section.

Extraction of the area for a group of regularly arranged objects

In the proposed method, by clustering each detected BB B_i, we extract the area of a group of objects that are regularly arranged for each object category.

Definition of BB vector

Each BB B_i has four characteristic values: the coordinates of the center position (x_i, y_i), width w_i, and height h_i, where $x_i$andy_i express the position of the object, and w_i and h_i express its scale and attitude. A vector expressing each characteristic of the BB (BB vector), $\boldsymbol{B}_i$, is defined as follows:

$$\boldsymbol{B}_i = (x_i, y_i, w_i, h_i) \tag{4}$$

Here, each BB can be expressed as points in a four-dimensional space with x, y, w and h as coordinate axes. By clustering $\boldsymbol{B}_i$, the area of a group of objects that are regularly arranged can be extracted. In the clustering, we extract clusters with a similar shape and position of $\boldsymbol{B}_i$. The area in which the shape and position of $\boldsymbol{B}_i$ are similar is regarded as the area of a group of regularly arranged objects.

Definition of distance between BBs

For the clustering, we define the distance (dissimilarity) between $\boldsymbol{B}_i$ and $\boldsymbol{B}_j$, $\mathrm{d}(\boldsymbol{B}_i, \boldsymbol{B}_j)$, as follows:

$$\mathrm{d}(\boldsymbol{B}_i, \boldsymbol{B}_j) = \mathrm{S}(\boldsymbol{B}_i, \boldsymbol{B}_j) + \mathrm{P}(\boldsymbol{B}_i, \boldsymbol{B}_j) \tag{5}$$

where S is the dissimilarity in shape, and P is the dissimilarity in position.

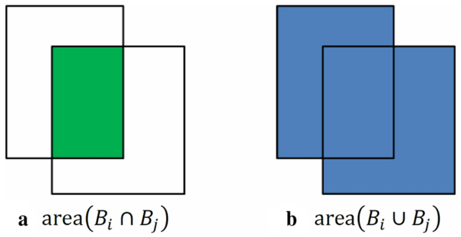
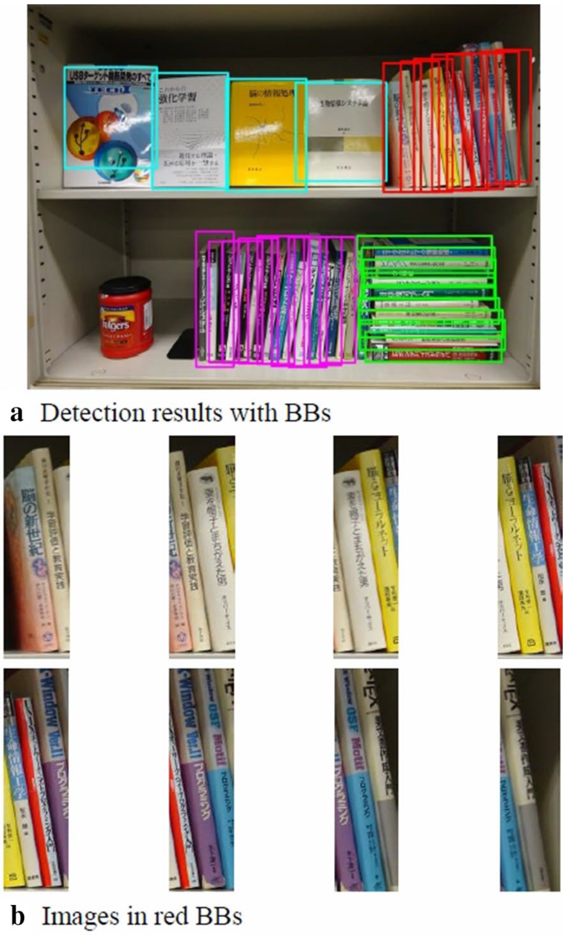


Fig. 7 IoU



a Detection results with BBs

b Images in red BBs

Fig. 8 Detection results of book category using the SSD

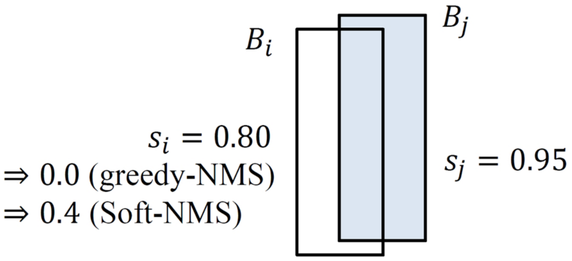


Fig. 9 Difference between the Soft-NMS and the greedy-NMS

(a) S: The value of S is calculated using the IoU as described (2); however, to ignore the impact of the position, it is calculated as $x_i = x_j = 0$ and $y_i = y_j = 0$. Here, iou $(\boldsymbol{B}_i, \boldsymbol{B}_j)$ $(0 \leq \text{iou}\,(\boldsymbol{B}_i, \boldsymbol{B}_j) \leq 1)$ expresses the similarity of the shape of BBs. Thus, we define dissimilarity S as follows:

$$\mathrm{S}(\boldsymbol{B}_i, \boldsymbol{B}_j) = -\log\left(\mathrm{iou}(\boldsymbol{B}_i, \boldsymbol{B}_j)\right) \tag{6}$$

(b) P: First, we define the similarity of position $\mathrm{Q}(\boldsymbol{B}_i, \boldsymbol{B}_j)$ as follows (Fig. 10):

$$\mathrm{Q}(\boldsymbol{B}_i, \boldsymbol{B}_j) = \frac{X_{ij} + Y_{ij}}{2},$$

where

$$\begin{aligned} X_{ij} &= \frac{\bar{X}_{ij}}{\bar{X}_{ij} + |x_i - x_j|}, \quad \bar{X}_{ij} = \frac{w_i + w_j}{2}, \\ Y_{ij} &= \frac{\bar{Y}_{ij}}{\bar{Y}_{ij} + |y_i - y_j|}, \quad \bar{Y}_{ij} = \frac{h_i + h_j}{2} \end{aligned} \tag{7}$$

Here, Q expresses the similarity of the position when considering the shape of the BBs.

As shown in Fig. 8a, depending on the arrangement, even object of the same category will have a different sized BB. The distances between the centers of adjacent BBs change depending on the size of each BB. In other words, the larger the BBs are, the longer the distance is between their centers. Here, Q considers the size of the BBs, as shown using $1/X_{ij} = 1 + |x_i - x_j|/\bar{X}_{ij}$. The wider the BB $(\bar{X}_{ij})$ is, the smaller the value of $1/X_{ij}$ is $(X_{ij}$ becomes larger). This is the same for Y_{ij}. Therefore, when the center distance is the same, the larger the BBs are, the greater the similarity Q is (Fig. 11).

Dissimilarity P is defined using $\mathrm{Q}(\boldsymbol{B}_i, \boldsymbol{B}_j)$ $(0 < \mathrm{Q}(\boldsymbol{B}_i, \boldsymbol{B}_j) \leq 1)$ as follows:

$$\mathrm{P}(\boldsymbol{B}_i, \boldsymbol{B}_j) = -\log\left(\mathrm{Q}(\boldsymbol{B}_i, \boldsymbol{B}_j)\right) \tag{8}$$

The effectiveness of distance d defined in (5) is shown in Fig. 12. Figure 12a shows the results of presenting the BB, shown in Fig. 8a, using the distance defined in (5) (however, it was mapped two-dimensionally using multidimensional scaling [31]). Figure 12b shows the results of using the Euclidean distance $\mathrm{d}'(\boldsymbol{B}_i, \boldsymbol{B}_j) = \|\boldsymbol{B}_i - \boldsymbol{B}_j\|$ as a comparison. Figure 12a and b show that, with the proposed method, BBs with a higher similarity of shape and position are distributed more closely together.

Extracting areas through clustering

In this paper, we use the density-based spatial clustering of applications with noise (DBSCAN) [32] for clustering.

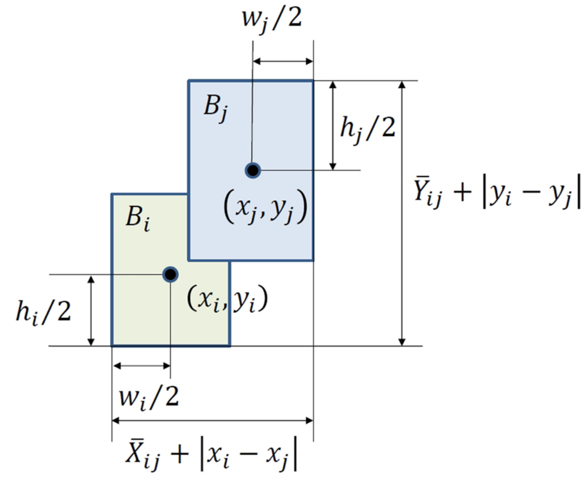


Fig. 10 Q. $\bar{X}_{ij} + |x_i - x_j|$ and $\bar{Y}_{ij} + |y_i - y_j|$ show the maximum distance between BBs in the horizontal and vertical direction, respectively

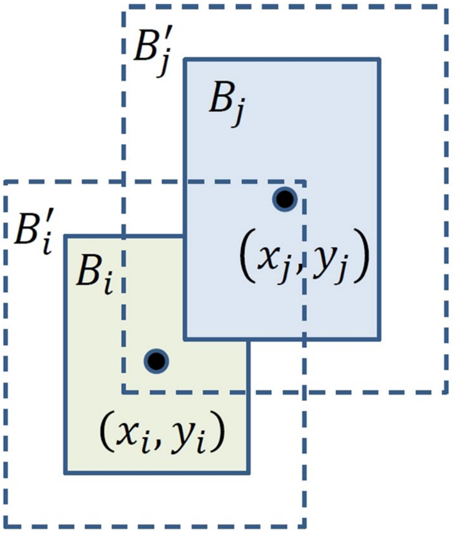


Fig. 11 Impact on the shape of BBs for Q. $Q(\boldsymbol{B}_i, \boldsymbol{B}_j) < Q\left(\boldsymbol{B}_i^{'}, \boldsymbol{B}_j^{'}\right)$

As discussed below, the DBSCAN is well suited to data clustering, as shown in Fig. 12a.

The DBSCAN uses two parameters: distance threshold ε and data number threshold *MinPts*. When there are *MinPts* or more data within the radius ε, they are extracted as a single cluster. Therefore, a group of dense data is classified into the same cluster. This enables extracting clusters of any shape that are not hyper-spherical. Unlike a well-known clustering method such as K-means, the number of clusters is automatically estimated. Because data that are not included in any cluster are considered noise, DBSCAN is considered robust against various noises (outlier). For example, by setting $MinPts = 2$, an isolated BB can be considered noise (not as a group of objects that are regularly arranged).

The clustering is applied for each object category. Figure 13 shows the results of clustering BBs of the category book, shown in Fig. 8a. Each cluster obtained through clustering is represented as R_k. A bounding box of all BBs included in the same cluster indicates the area of a group of regularly arranged objects.

Identifying arrangement patterns

In the proposed method, the arrangement pattern is identified for each cluster R_k, as described in "Extraction of the area for a group of regularly arranged objects" section.

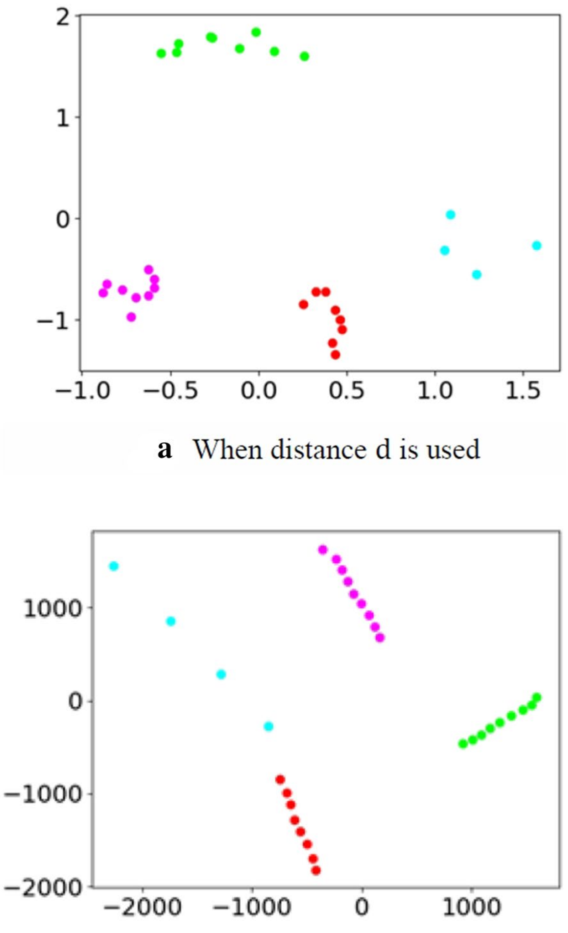
Identifying object arrangement patterns using BBs

The identification of arrangement patterns is based on the set of BBs, $\left\{B_i^k | B_i^k \in R_k\right\}$. Because the shape and position of $B_i^k \in R_k$ are similar, the set of BBs, $\left\{B_i^k | B_i^k \in R_k\right\}$ expresses the characteristics of the arrangement patterns. A feature matrix $\boldsymbol{F}_k$, which expresses the arrangement patterns, is defined as follows (N_k is the number of $B_i^k \in R_k$):

$$\begin{aligned} \boldsymbol{F}_k &= \begin{pmatrix} \hat{\boldsymbol{B}}_1^k \\ \vdots \\ \hat{\boldsymbol{B}}_{N_k}^k \end{pmatrix} \\ &= \begin{pmatrix} \hat{x}_1^k, \hat{y}_1^k, w_1^k, h_1^k \\ \vdots \\ \hat{x}_{N_k}^k, \hat{y}_{N_k}^k, w_{N_k}^k, h_{N_k}^k \end{pmatrix}, \end{aligned} \tag{9}$$

where $\hat{x}_i^k = x_i^k - \bar{x}^k, \quad \hat{y}_i^k = y_i^k - \bar{y}^k$.

Here, $\bar{x}^k$ and $\bar{y}^k$ are the mean of x_i^k and y_i^k, respectively. In other words, $\hat{x}_i^k$ and $\hat{y}_i^k$ are the relative positions from the center of each area R_k. In addition, $\boldsymbol{F}_k$ is a $N_k \times 4$ matrix. In this paper, to identify the arrangement patterns using machine learning, we prepare training dataset $E = \{\boldsymbol{F}_k\}$ for shelved, stacked and displayed books. The data size is 100 for each. All training data $\boldsymbol{F}_k$ include some BBs $\hat{\boldsymbol{B}}_i^k$. The training data (BBs) were created by using computer graphics.



a When distance d is used

b When Euclidean distance d′ is used

Fig. 12 BB vector distributions. The colors of the points correspond to the colors of the BBs in Fig. 8a

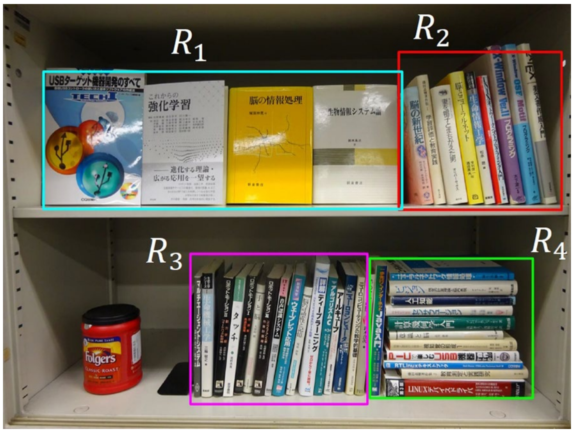


Fig. 13 Extraction results for the area of a group of regularly arranged objects

Conversion into BoW vector

When using $\boldsymbol{F}_k$ as input to the classifier, the following problems arise. As shown in Fig. 8a, the number of BBs N_k varies depending on the area R_k. Therefore, $\boldsymbol{F}_k$ is a variable length matrix. Furthermore, $\boldsymbol{F}_k$ changes even with the order of $\hat{\boldsymbol{B}}_i^k$.

In the proposed method, the feature matrix $\boldsymbol{F}_k$ is converted into a Bag of Words (BoW) vector [33], and input into the classifier. BoW vectors are often used in natural language processing.

A BoW vector $\boldsymbol{F}_k^{bow}$ is calculated as the sum of 1-of-K vectors as follows:

$$\boldsymbol{F}_k^{bow} = \mathrm{e}\left(\hat{\boldsymbol{B}}_1^k\right) + \cdots + \mathrm{e}\left(\hat{\boldsymbol{B}}_{N_k}^k\right) \tag{10}$$

where $\mathrm{e}\left(\hat{\boldsymbol{B}}_i^k\right)$ shows the conversion of $\hat{\boldsymbol{B}}_i^k$ into the 1-of-K vector. The 1-of-K vector is a vector with one element containing a 1 and all other elements containing a 0.

$$\mathrm{e}\left(\hat{\boldsymbol{B}}_i^k\right) = (0, \ldots, 0, 1, 0, \ldots, 0) \tag{11}$$

Here, $\boldsymbol{F}_k^{bow}$ is a fixed-length vector that does not depend on the order of $\hat{\boldsymbol{B}}_i^k$.

Conversion into the 1-of-K vector using SOM

To convert the continuous vector $\hat{\boldsymbol{B}}_i^k$ into the 1-of-K vector, $\hat{\boldsymbol{B}}_i^k$ needs to be quantized (discretized). The proposed method used self-organizing maps (SOM) [34] for quantization.

The SOM is a type of neural network (Fig. 14), and consists of two layers: input and output layers. The neuron (unit) of the output layer U_p has a reference vector $\boldsymbol{W}_p = \left(W_x, W_y, W_w, W_h\right)$ with the same dimension as the input vector $\hat{\boldsymbol{B}}_i^k$. We calculate the Euclidean distance between the input and each reference vector, $\|\hat{\boldsymbol{B}}_i^k - \boldsymbol{W}_p\|$, and select the output unit with the reference vector that is the closest to the input vector as the winner unit U_c.

$$U_c = \arg\min_p \left\|\hat{\boldsymbol{B}}_i^k - \boldsymbol{W}_p\right\| \tag{12}$$

Equation (12) indicates a mapping in which the input vector $\hat{\boldsymbol{B}}_i^k$ and the winner unit U_c correspond with each other. By assigning 1 to the winner unit U_c and 0 to the other units $U_p\ (p \neq c)$, the input vector can be converted into the 1-of-K vector.

In this paper, we use a two-dimensional SOM in which the output unit is aligned in the two-dimensional lattice with $L \times M$ (the reference vector of each output unit is $\boldsymbol{W}_{l,m}$). Therefore, the BoW vector becomes a $L \times M$ vector (Fig. 14).

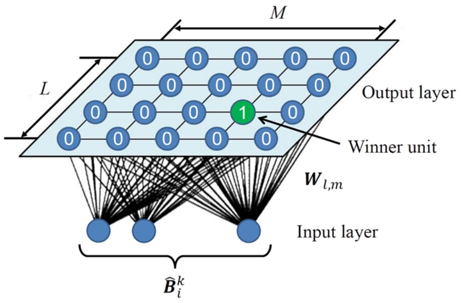
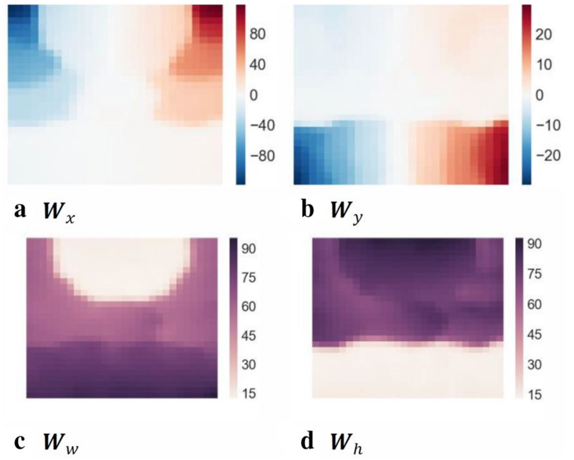


Fig. 14 Conversion into the 1-of-K vector using SOM



a $\boldsymbol{W}_x$ **b** $\boldsymbol{W}_y$ **c** $\boldsymbol{W}_w$ **d** $\boldsymbol{W}_h$

Fig. 15 $\boldsymbol{W}_{l,m}$

Using the training dataset E, the learning results of the SOM ($L = M = 28$) are shown in Fig. 15. The initial vector $\boldsymbol{W}_{l,m}^{init}$ is calculated using the principal component analysis (PCA) method [34]. The PCA is performed based on the variance–covariance matrix of the training data $\hat{\boldsymbol{B}}_i^E \in E$.

$$\hat{\boldsymbol{B}}_i^E = \left(\hat{x}_i^E, \hat{y}_i^E, w_i^E, h_i^E\right),$$
$$\text{where} \quad \hat{x}_i^E = x_i^E - \bar{x}^E, \quad \hat{y}_i^E = y_i^E - \bar{y}^E. \tag{13}$$

Here, $\bar{x}^E$ and $\bar{y}^E$ are the mean of x_i^E and y_i^E, respectively. As shown in (14), we set the vector $\boldsymbol{W}_{l,m}^{init}$ along the two axes $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ ($\boldsymbol{b}_1$ is the first principal component, and $\boldsymbol{b}_2$ is the second principal component).

$$\boldsymbol{W}_{l,m}^{init} = \bar{\boldsymbol{b}} + \lambda\left[\boldsymbol{b}_1\left(l - \frac{L}{2}\right) + \boldsymbol{b}_2\left(m - \frac{M}{2}\right)\right] \tag{14}$$

where $\bar{\boldsymbol{b}}$ is the mean vector of $\hat{\boldsymbol{B}}_i^E$, and λ is the maximum eigenvalue of the variance–covariance matrix.

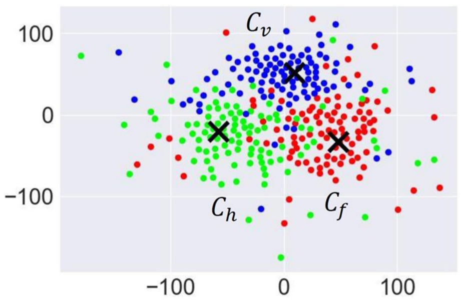


Fig. 16 Training data distribution

Identification of arrangement patterns with SVM

In this paper, we use the support vector machine (SVM) [35] to identify the arrangement patterns. The conversion results of the training data into the BoW vector $\boldsymbol{F}_k^{bow}$ is shown in Fig. 16. The figure shows the distribution of $\boldsymbol{F}_k^{bow}$, which is expressed two-dimensionally through a dimension reduction using the t-SNE [36]. Figure 17a–c show representatives of BoW vectors for each category. Each BoW vector corresponds to data represented as "×" in Fig. 16.

Considering the detection error of BBs, in the conversion into the BoW vector, we provided the output units with the following values h $(0 < h \leq 1)$ in response to the distance from the winner unit.

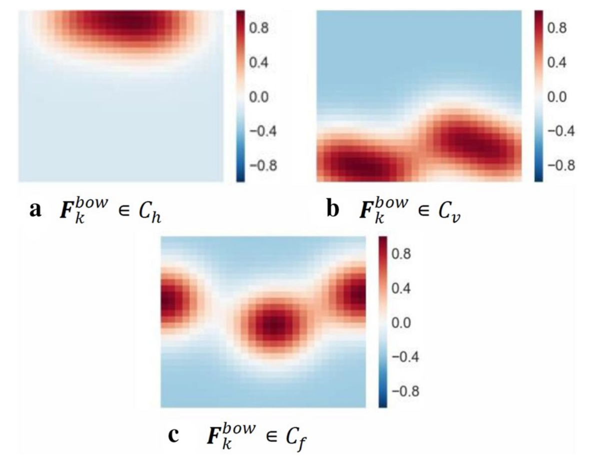
$$h = \exp\left(-\frac{\left\|\hat{\boldsymbol{B}}_i^k - \boldsymbol{W}_{l,m}\right\|^2}{2\sigma^2}\right) \tag{15}$$

Here, σ is a parameter that determines the size of the nearby areas. Each BoW vector is converted such that the mean is zero, and normalized using the maximum value.

The characteristics in the BoW vector $\boldsymbol{F}_k^{bow}$ for each arrangement pattern are as follows:

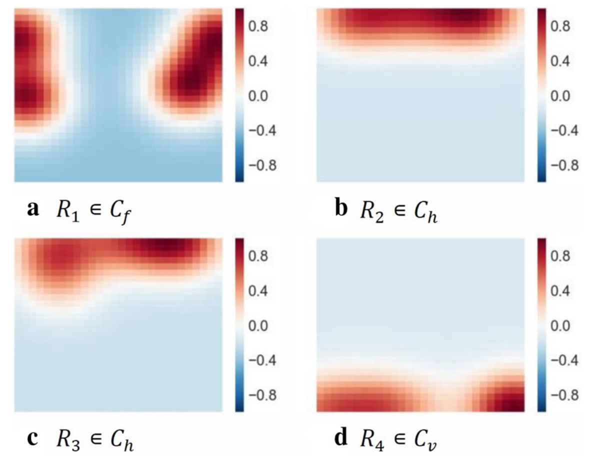
- The case of $\boldsymbol{F}_k^{bow} \in C_h$, large values are mainly found in the upper part.
- The case of $\boldsymbol{F}_k^{bow} \in C_v$, large values are mainly found in the lower part.
- The case of $\boldsymbol{F}_k^{bow} \in C_f$, large values are mainly found in the middle part.

Figure 18 shows BoW vectors for R_1, R_2, R_3 and R_4 in Fig. 13. The SVM was able to correctly identify the arrangement pattern of each area. The BoW vector of each arrangement pattern has notable characteristics, as mentioned above. Therefore, we suppose that identifying arrangement patterns based on the BB can provide



a $F_k^{bow} \in C_h$ **b** $F_k^{bow} \in C_v$ **c** $F_k^{bow} \in C_f$

Fig. 17 Representatives of BoW vectors for the training data



a $R_1 \in C_f$ **b** $R_2 \in C_h$ **c** $R_3 \in C_h$ **d** $R_4 \in C_v$

Fig. 18 Identification results of arrangement patterns based on the BoW vector

greater accuracy than directly identifying arrangement patterns from images (refer to "Evaluation of the identification" section).

Evaluations

Evaluation of the extraction

Proposed method

Evaluation of the extraction of the area for a group of regularly arranged objects was performed (refer to "Extraction of the area for a group of regularly arranged objects" section). Figures 19 and 20 show the results of IoU evaluation between ground-truth BBs and predicted BBs. If the set of the ground-truth BBs is denoted by T, and the set of the predicted BBs is denoted by P, the IoU is calculated by the following equation.

$$\mathrm{IoU} = \frac{\mathrm{area}(B_T \cap B_P)}{\mathrm{area}(B_T \cup B_P)}, \qquad (16)$$

where $B_T = \{B_i | B_i \in T\}, \quad B_P = \left\{B_j \middle| B_j \in P\right\}$.

The average of IoU values was 0.68 (the maximum was 0.95, and the minimum was 0.39). The images used for the evaluation were sampled from the MS-COCO dataset (the ground-truth BBs were given by us). A discussion on the results is given in "BB detection" section.

Area extraction by a baseline method

The area extraction was performed by a baseline method, which is based on the HOG (histograms of oriented gradients) [37] and the SVM (Linear SVM). The HOG is established as one of the most popular hand-crafted features, which provides excellent performance for object recognition [38].

For the training of the SVM, 150 positive images (refer to "Evaluation of the identification" section) and 500 negative images were used (the negative images were sampled from the MS-COCO dataset). In the training of the SVM, HOG features were extracted from the training images (converted into grayscale images), and the SVM was trained using the HOG features. The sliding window and image pyramid techniques combined with the trained SVM are used for the area extraction [39]. The detection results were post-processed by the greedy-NMS.

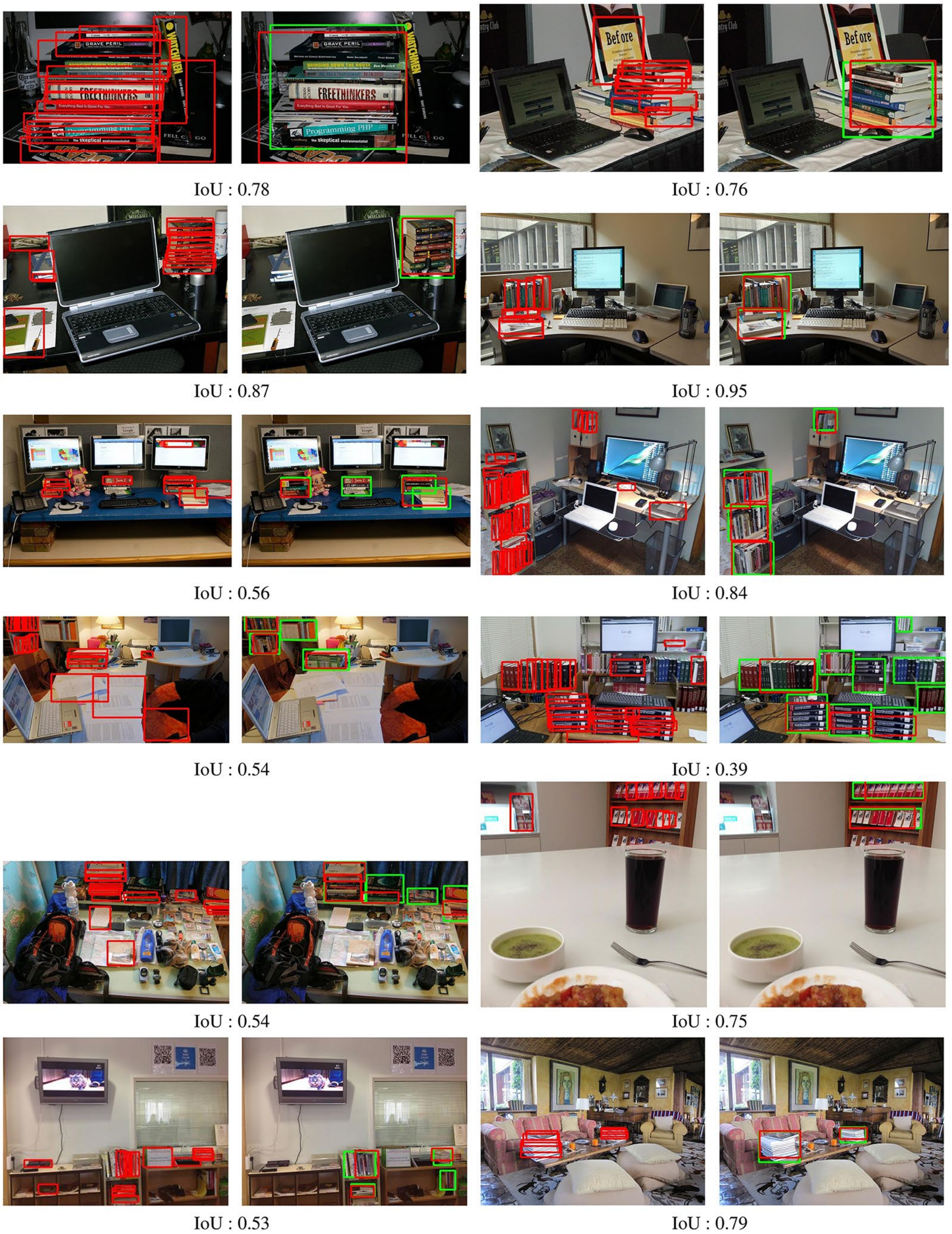
The area extraction results are shown in Figs. 21 and 22. The average of IoU values was 0.17 (the maximum was 0.70, and the minimum was 0.00). The proposed method shows higher performance than the baseline method (however, because the experiment conditions are different (e.g., the number of training data), they can not be fairly compared).

Evaluation of the identification

Proposed method

Evaluation of the identification of object arrangement patterns was performed (refer to "Identifying arrangement patterns" section). For the evaluation, 150 images (shelved patterns: 50 images, stacked patterns: 50 images, displayed patterns: 50 images), that differed in terms of not only arrangement patterns but also the type, number, arrangement order of books, angle of a camera, etc., were used (a part of the images is shown in Fig. 23).

The evaluation was performed using cross-validation. The images were split to training images and testing images. The stratified K-fold method was adopted. The stratified K-fold method splits the data into training and



IoU : 0.78

IoU : 0.76

IoU : 0.87

IoU : 0.95

IoU : 0.56

IoU : 0.84

IoU : 0.54

IoU : 0.39

IoU : 0.54

IoU : 0.75

IoU : 0.53

IoU : 0.79

Fig. 19 Extraction examples by the proposed method (1). For each pair, the left side is the result of the SSD and the right side is the result of extraction by the proposed method (red BB: predicted, green BB: ground-truth)

testing sets by preserving the percentage of the samples for each category [40].

(a) *Training* Ground-truth BBs were given to each object in the training images (Fig. 23). The learning of the SOM was performed using the ground-truth BBs. BoW vectors were generated from the training images (the ground-truth BBs) using the trained SOM. The learning of the SVM (Linear SVM) was performed using the BoW vectors.

(b) *Testing* Predicted BBs were detected from the testing images using the SSD. BoW vectors were generated from the predicted BBs using the trained SOM. The arrangement patterns of the testing images were

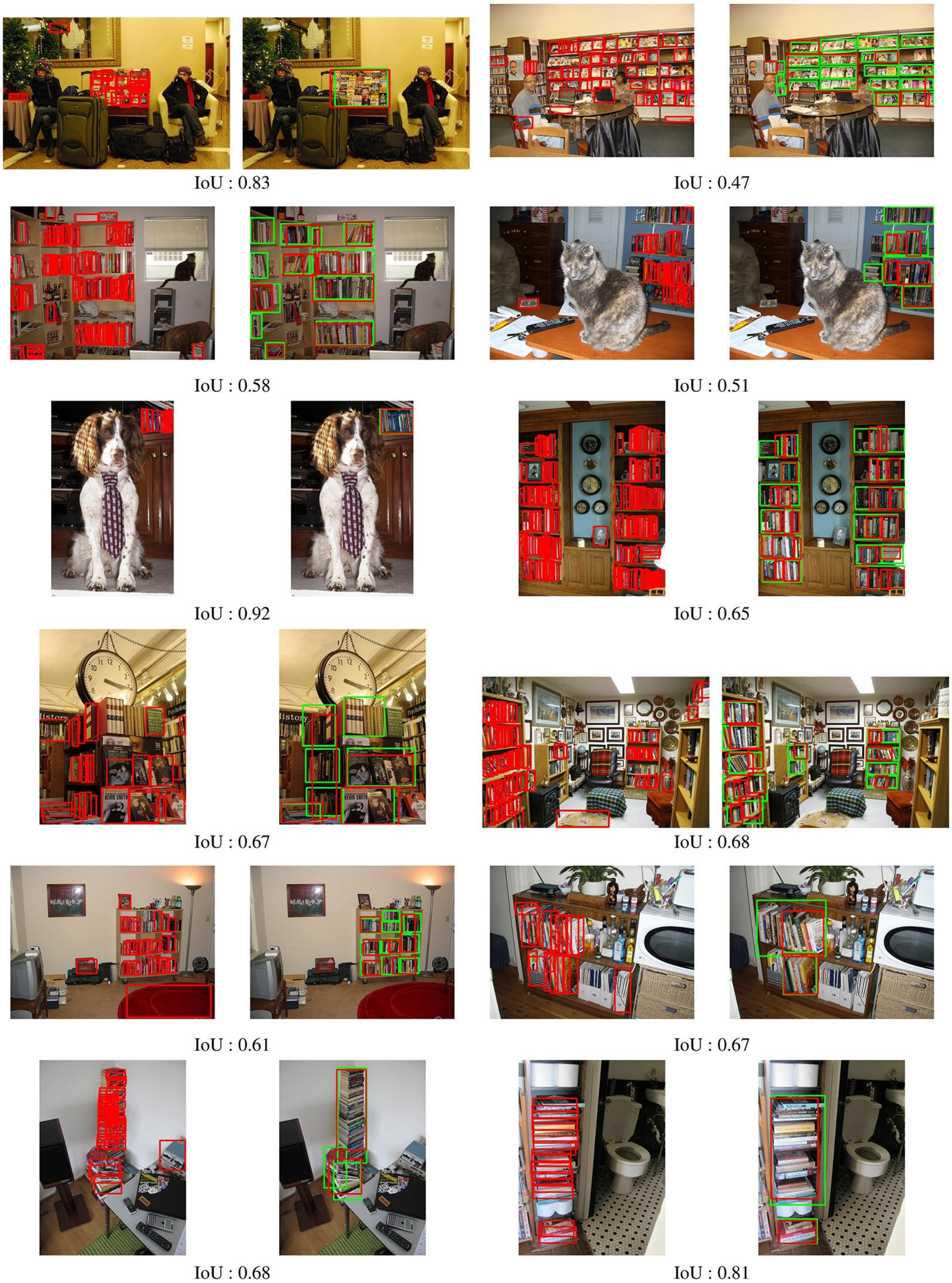


Fig. 20 Extraction examples by the proposed method (2). For each pair, the left side is the result of the SSD and the right side is the result of extraction by the proposed method (red BB: predicted, green BB: ground-truth)

identified by inputting the BoW vectors into the trained SVM. Table 1 shows identification accuracy.

In the proposed method, the predicted BBs belonging to the target area (which is the area of a group of regularly arranged objects) were obtained by clustering BBs detected by the SSD (refer to “Extraction of the area for a group of regularly arranged objects” section). However, in this evaluation, the predicted BBs belonging to the target area were obtained by detecting using the SSD

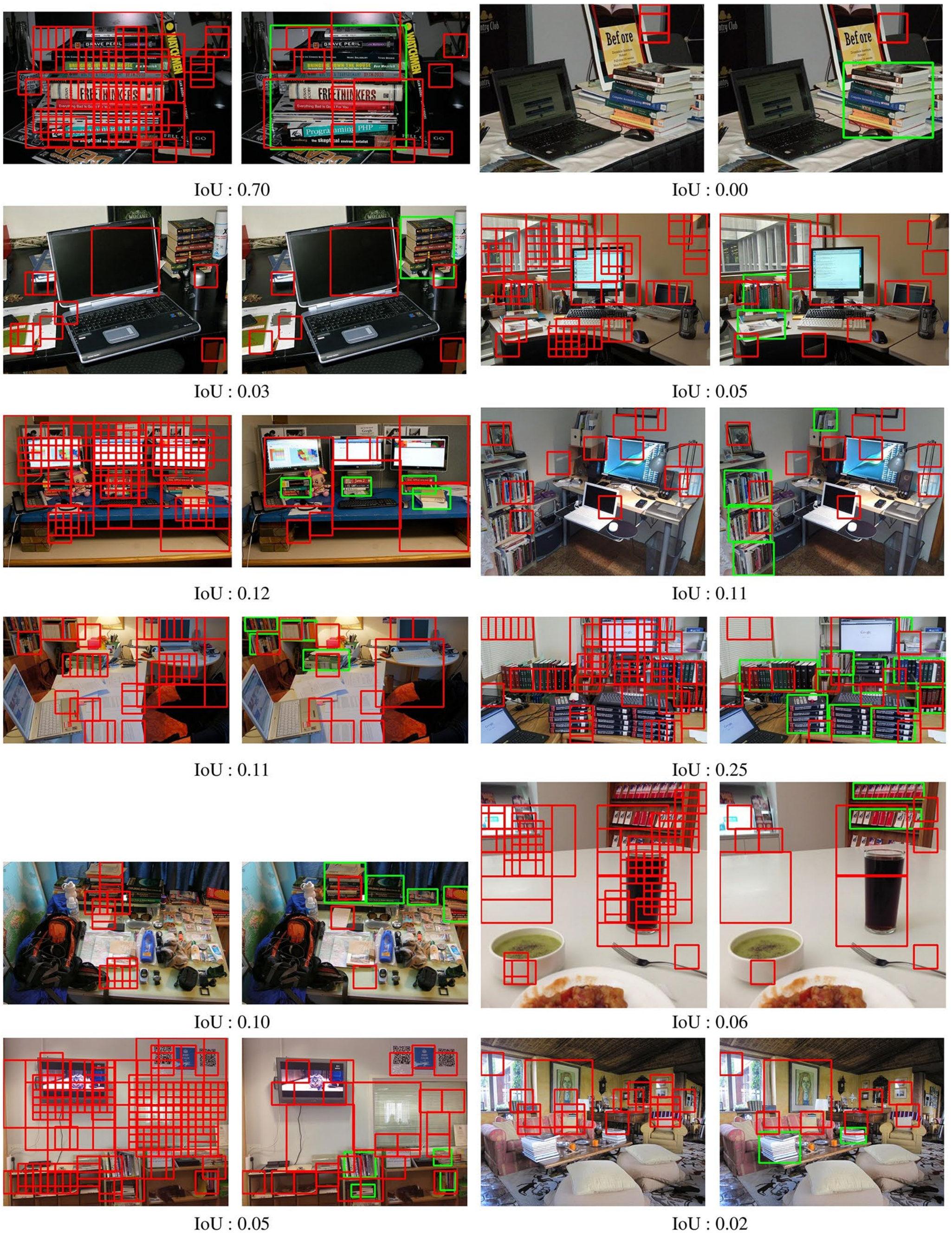


Fig. 21 Extraction examples by the baseline method (1). For each pair, the left side is the extraction result before NMS and the right side is the extraction result after NMS (red BB: predicted, green BB: ground-truth)

from an image cropped to the target area beforehand. There is no essential difference between both.

Comparison with a baseline method

A comparison was performed between the proposed method and a baseline method. As the baseline method, a method based on the HOG and the SVM (Linear SVM) was used.

In the training of the SVM, HOG features were extracted from the training images (converted into gray-scale images), and the SVM was trained using the HOG features. In the testing of the SVM, the identification of arrangement patterns was performed by inputting HOG

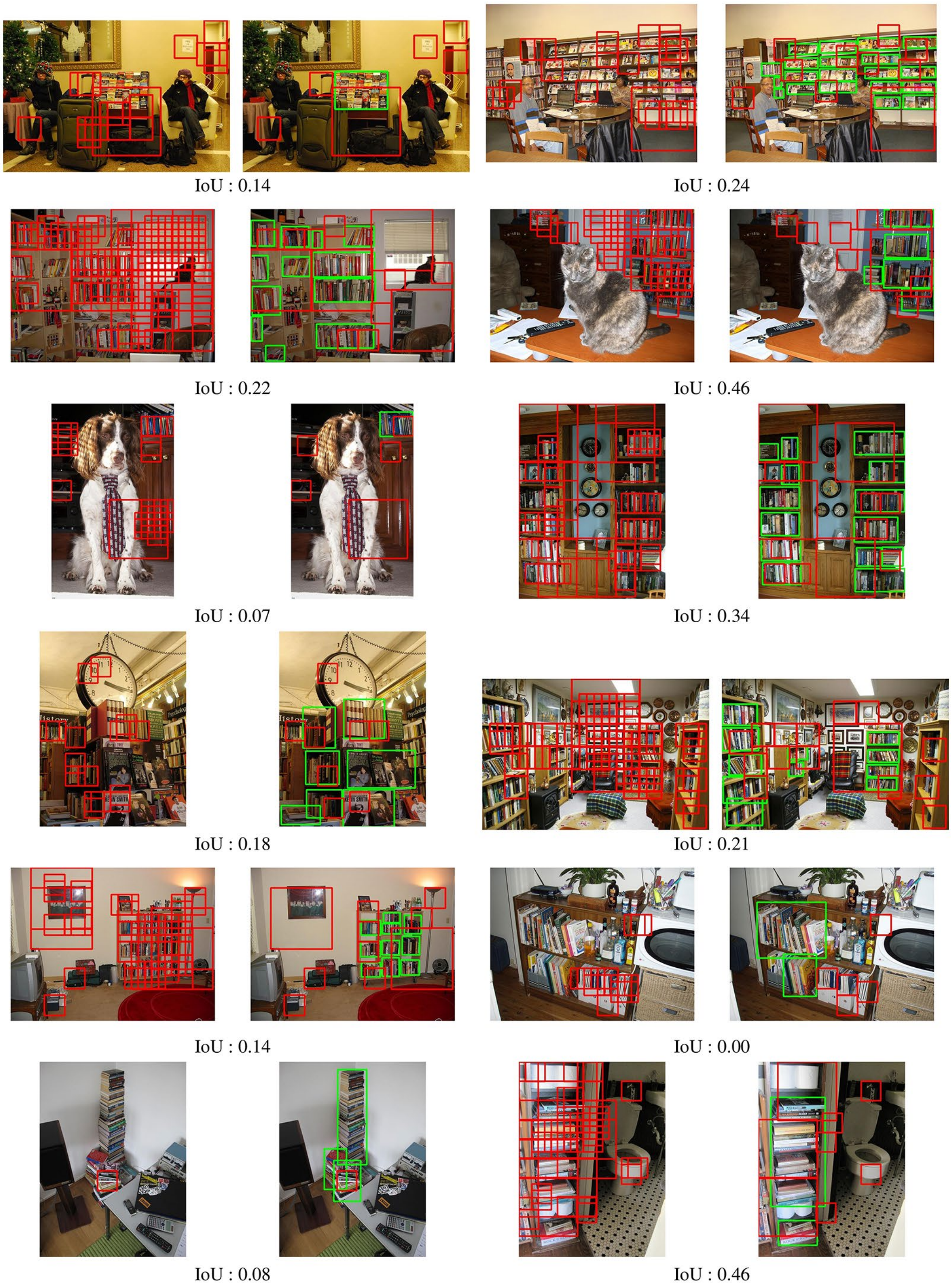


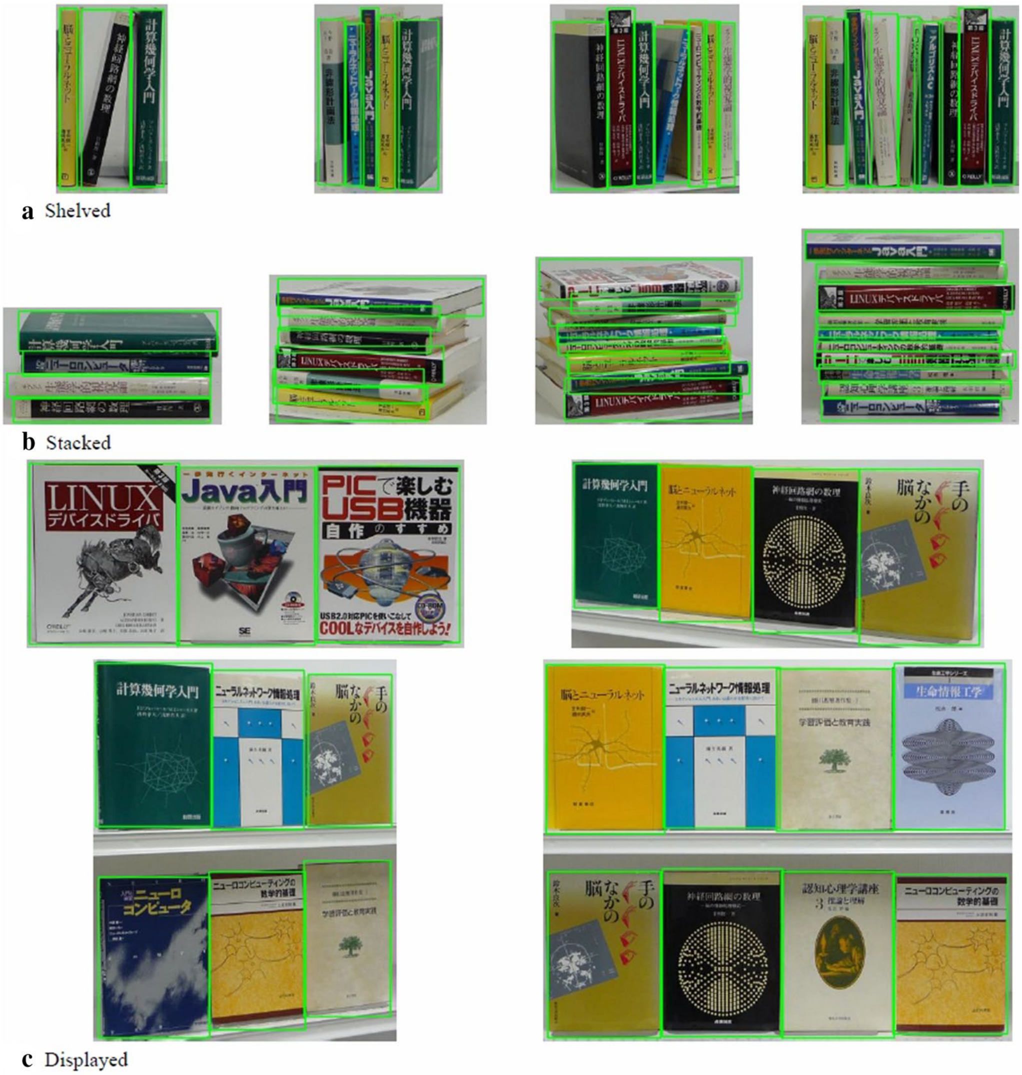
Fig. 22 Extraction examples by the baseline method (2). For each pair, the left side is the extraction result before NMS and the right side is the extraction result after NMS (red BB: predicted, green BB: ground-truth)

features extracted from the testing images (converted into grayscale images) into the trained SVM. Table 1 shows identification accuracy. The proposed method shows higher accuracy than the baseline method.

Discussion

BB detection

The proposed method is based on the BB. Therefore, the proposed method depends on the detection performance of the BB. In this paper, the SSD was used for the



a Shelved

b Stacked

c Displayed

Fig. 23 Examples of images used for the evaluation of the identification (green BB: ground-truth)

detection of the BB. The SSD has low performance on small objects and dense scenes [41] (these weak points are expected to be overcome by the development of novel general object detection algorithms).

Table 1 Identification accuracy using stratified K-fold cross-validation (the average of 5 trials performed using data shuffled at random order)

	Proposed (%)	HOG + SVM (%)
K = 2	98.9	88.9
K = 3	99.2	93.0
K = 4	99.2	93.8
K = 5	99.2	94.3
K = 150 (leave-one-out)	100.0	95.3

In "Evaluation of the extraction" section, the IoU evaluation with respect to the area extraction for a group of regularly arranged objects was performed. The reason why the IoU was small (the average of IoU values was 0.68) is mainly due to the failure of the BB detection (refer to Figs. 19, 20). If the performance of the BB detection improves, the improvement of IoU can be expected.

As shown in Fig. 8b, the accuracy of BB regression on dense scenes is low. However, it is considered that this has a small influence on the proposed method. In the area extraction, the area is extracted as the minimum rectangle surrounding all BBs belonging to the area (refer to "Extraction of the area for a group of regularly arranged objects" section). Moreover, in the identification of the arrangement patterns, the BoW vector independent of

Fig. 24 Various objects in stores

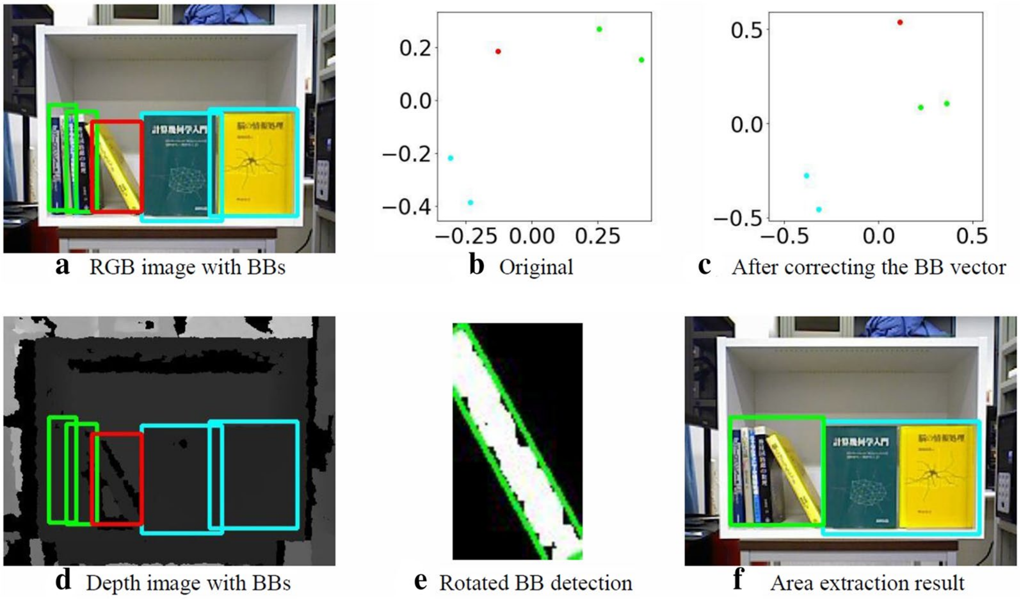
the number of BBs is used (refer to "Identifying arrangement patterns" section). Therefore, it is considered that the low accuracy of BB regression has a small influence on the proposed method.

Also, in this paper, the Soft-NMS is used instead of the greedy-NMS. Using the soft-NMS could increase false positive. However, as with the low accuracy of BB regression, it is considered that this has a small influence on the proposed method.

Approximation of objects using BB

This research aims at robot picking in warehouses and stores. Most of objects in warehouses and stores can be approximated by rectangular parallelepipeds because most of the objects are packaged (Fig. 24).

The visual appearance of the objects in the image depends on the camera angle. If the camera is mounted on a robot, it is possible to control the angle of the camera. In warehouses and stores, objects are generally placed on shelves, and the shelves are generally aligned along the passage. It is possible to create a situation where the robot moves through the passage while orienting the



a RGB image with BBs **b** Original **c** After correcting the BB vector **d** Depth image with BBs **e** Rotated BB detection **f** Area extraction result

Fig. 25 Correction of the BB of an inclined object

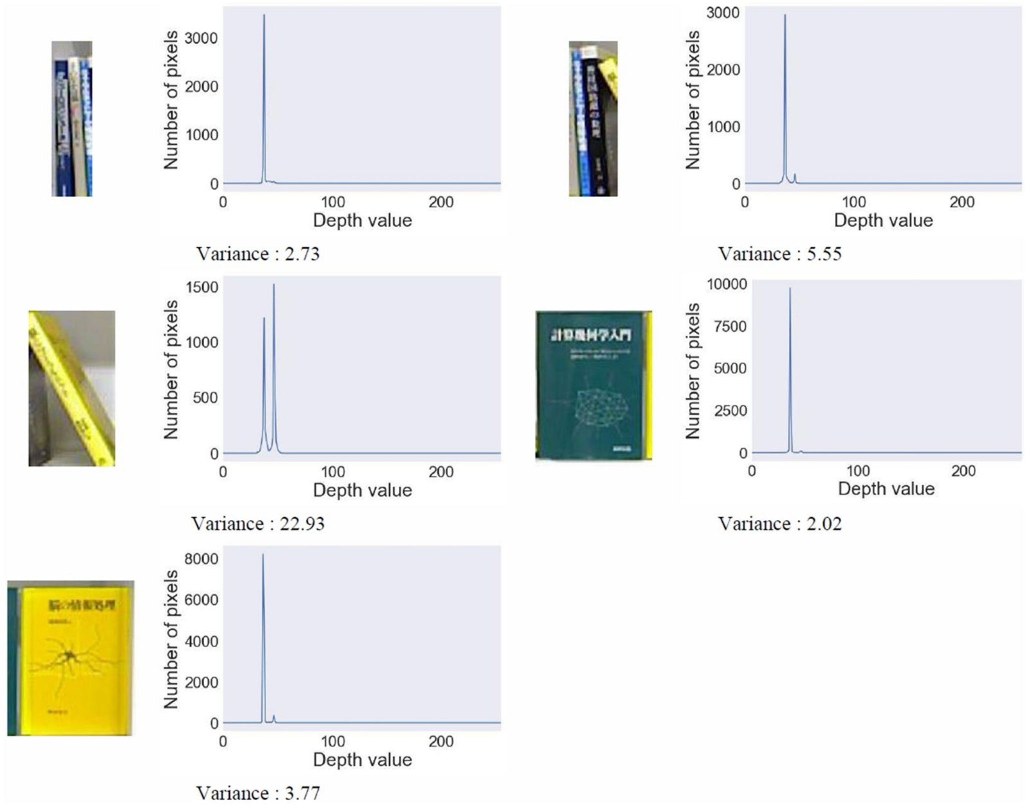


Fig. 26 Histogram and variance of depth values

angle of the camera to be orthogonal to the shelves (e.g., shelf-scanning robot [42]). In this case, most of the visual appearances of the objects can be approximated by BBs. It is considered that the proposed method can be applied to many scenarios in warehouses and stores.

Inclination of objects

The shape and size of BBs change depending on the inclination angles of objects (Fig. 25a). The distribution of the BBs ($\boldsymbol{B}_i$) in Fig. 25a is shown in Fig. 25b (it was mapped two-dimensionally using multidimensional scaling). Thus, if the object is greatly inclined, the area of a group of regularly arranged objects may not be extracted correctly. An example of the countermeasure is shown in the following.

In this countermeasure, inclined objects are detected, and then the BBs of the inclined objects are corrected. Depth images (Fig. 25d) are used for the detection of the inclined objects. The histogram and variance of depth values in each BB are shown in Fig. 26. If the variance is higher than a given threshold, the BB is corrected as that of the inclined object, as follows.

The depth image is binarized to detect the inclined object. Then, a rotated BB with minimum area for the object is obtained. The obtained BB is shown in Fig. 25e (the threshold for the binarization was obtained by Otsu's discriminant analysis method [43]). The BB vector $\boldsymbol{B}_i$ is corrected using the width $w_i^{'}$ and the height $h_i^{'}$ of the obtained BB (there is no correction of the position of the BB (x_i, y_i)).

$$(x_i, y_i, w_i, h_i) \rightarrow \left(x_i, y_i, w_i^{'}, h_i^{'}\right) \tag{17}$$

The distribution of the corrected BB vectors is shown in Fig. 25c. As shown in Fig. 25f, the area could be correctly detected by the correction of BBs. In the identification, the problem could be solved by including inclined objects in training images (refer to "Evaluation of the identification" section).

Conclusion

In this paper, we proposed a method in which the area of a group of regularly arranged objects is extracted from an image and the arrangement pattern of the extracted area is identified. Objects are often stacked and aligned in specific arrangement patterns to improve storage efficiency. There are typical patterns in the arrangement patterns. A specific picking strategy set is often linked to a specific arrangement pattern. By linking diverse arrangement patterns with specific picking strategies, the robot picking performance is expected to improve. In the future, we plan to continue with researches on the practical robot picking based on object arrangement patterns.

Authors' contributions

TA carried out the main part of this research and drafted the manuscript. KN, TN and IM contributed concepts of this research and revised the manuscript. All authors read and approved the final manuscript.

Author details

[1] 2-24-16 Naka-cho, Koganei-shi, Tokyo 184-8588, Japan. [2] 1-1-1 Umezono, Tsukuba, Ibaraki 305-8568, Japan.

Acknowledgements

This work was supported by JSPS KAKENHI Grant Number JP17H01805.

Competing interests

The authors declare that they have no competing interests.

Availability of data and materials

Not applicable.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 29 January 2018 Accepted: 28 August 2018
Published online: 10 September 2018

References

1. Eppner C et al (2017) Lessons from the Amazon Picking Challenge: four aspects of building robotic systems. In: Proceedings of International Joint Conference on Artificial Intelligence (IJCAI), pp 4831–4835
2. Jonschkowski R, Eppner C, Höfer S, Martín RM, Brock O (2016) Probabilistic multi-class segmentation for the Amazon Picking Challenge. In: Proceedings of IEEE/RSJ international conference on intelligent robots and systems (IROS)
3. Shiraki Y et al (2014) Modeling of everyday objects for semantic grasp. In: Proceedings of IEEE international symposium on robot and human interactive communication (RO-MAN), pp 750–755
4. Odhner LU et al (2014) A compliant, under actuated hand for robust manipulation. Int J Robot Res 33(5):736–752
5. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of IEEE conference on computer vision and pattern recognition (CVPR)
6. Girshick R (2015) Fast R-CNN. In: Proceedings of IEEE international conference on computer vision (ICCV)
7. Ren S, He K, Girshick R, Sun J (2015) Faster R-CNN: towards real-time object detection with region proposal networks. In: Proceedings of international conference on neural information processing systems (NIPS), pp 91–99
8. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: unified, real-time object detection. In: Proceedings of IEEE conference on computer vision and pattern recognition (CVPR)
9. Liu W et al (2016) SSD: single shot multibox detector. In: Proceedings of European conference on computer vision (ECCV)
10. Lenz I, Lee H, Saxena A (2015) Deep learning for detecting robotic grasps. Int J Robot Res 34(4–5):705–724
11. Mahler J et al (2016) Dex-Net 1.0: a cloud-based network of 3D objects for robust grasp planning using a Multi-Armed Bandit model with correlated rewards. In: Proceedings of IEEE international conference on robotics and automation (icra), pp 1957–1964
12. Mahler J et al (2017) Dex-Net 2.0: deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. In: Proceedings of robotics: science and systems conference (RSS)
13. Levine S, Pastor P, Krizhevsky A, Quillen D (2016) Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. In: Proceedings of international symposium on experimental robotics (ISER)
14. Al-Hujazi E, Sood A (1990) Range image segmentation with applications to robot bin-picking using vacuum gripper. IEEE Trans Syst Man Cyber 20(6):1313–1325
15. Domae Y, Okuda H, Taguchi Y, Sumi K, Hirai T (2014) Fast graspability evaluation on single depth maps for bin picking with general grippers. In: Proceedings of IEEE international conference on robotics and automation (ICRA), pp 1997–2004
16. Dupuis DC, Léonard S, Baumann MA, Croft EA, Little JJ (2008) Two-fingered grasp planning for randomized bin-picking. In: Proceedings of robotics: science and systems manipulation workshop
17. Fuchs S, Haddadin S, Keller M (2010) Cooperative bin-picking with Time-of-Flight camera and impedance controlled DLR lightweight robot III. In: Proceedings of IEEE/RSJ international conference on intelligent robots and systems (IROS)
18. Ghita O, Whelan PF (2003) A bin picking system based on depth from defocus. J Mach Vision Appl 13(4):234–244
19. Harada K et al (2013) Probabilistic approach for object bin picking approximated by cylinders. In: Proceedings of IEEE international conference on robotics and automation (ICRA), pp 3742–3747
20. Harada K et al (2014) Project on development of a robot system for random picking—grasp/manipulation planner for a dual-arm manipulator. In: Proceedings of IEEE/SICE international symposium on system integration (SII), pp 583–589
21. Harada K et al (2016) Initial experiments on learning-based randomized bin-picking allowing finger contact with neighboring objects. In: Proceedings of IEEE international conference on automation science and engineering (CASE), pp 1196–1202
22. Kirkegaard J, Moeslund TB (2006) Bin-picking based on harmonic shape contexts and graph-based matching. In: Proceedings of international conference on pattern recognition (ICPR), vol. 2, pp 581–584
23. Zuo A, Zhang JZ, Stanley K, Wu QMJ (2004) A hybrid stereo feature matching algorithm for stereo vision-based bin picking. Int J Pattern Recognit Artif Intell 18(8):1407–1422
24. Cutkosky MR (1989) On grasp choice, grasp models, and the design of hands for manufacturing tasks. IEEE Trans Robot Autom 5(3):269–279
25. Gibson JJ (1979) The ecological approach to visual perception. Houghton Mifflin, Boston
26. Detry R et al (2009) Learning object-specific grasp affordance densities. In: Proceedings of IEEE international conference on development and learning (ICDL), pp 1–7
27. Geng T, Wilson J, Sheldon M, Lee M, Hülse M (2013) Synergy-based affordance learning for robotic grasping. Int J Robot Auton Syst 61(12):1626–1640
28. Bodla N, Singh B, Chellappa R, Davis LS (2017) Soft-NMS—improving object detection with one line of code. In: Proceedings of IEEE international conference on computer vision (ICCV)
29. Github. weiliu89/caffe. https://github.com/weiliu89/caffe/tree/ssd. Accessed 1 Sept 2017
30. Lin TY et al (2014) Microsoft COCO: common objects in context. In: Proceedings of European conference on computer vision (ECCV), pp 740–755
31. Kruskal JB, Wish M (1978) Multidimensional scaling. Sage Publications, New York
32. Ester M, Kriegel HP, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of international conference on knowledge discovery and data mining (ICKDDM), pp 226–231
33. Liu D, Sun DM, Qiu ZD (2009) Bag-of-words vector quantization based face identification. In: Proceedings of international symposium on electronic commerce and security (ISECS), vol .2, pp 29–33
34. Kohonen T (2001) Self-organizing maps. Springer, Berlin
35. Vapnik VN (1995) The nature of statistical learning theory. Springer, Berlin
36. van der Maaten LJP, Hinton GE (2008) Visualizing high-dimensional data using t-SNE. J Mach Learn Res 9:2579–2605

37. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: Proceedings of conference on computer vision and pattern recognition, pp 886–893
38. Sharma R, Savakis A (2015) Lean histogram of oriented gradients features for effective eye detection. J Elect Imaging 24(6):063007
39. Github, VladKha/object_detector. https://github.com/VladKha/object_detector. Accessed 10 Aug 2018
40. Pektas A, Acarman T (2017) Ensemble machine learning approach for android malware classification using hybrid features. In: Proceedings of international conference on computer recognition systems, pp 191–200
41. Fu CY, Liu W, Ranga A, Tyagi A, Berg AC (2017) Dssd: deconvolutional single shot detector. arXiv preprint arXiv:1701.06659
42. Bossa Nova Robotics. http://www.bossanova.com. Accessed 9 May 2018
43. Otsu N (1978) Discriminant and least-squares threshold selection. In: Proceedings of international joint conference on pattern recognition, pp 592–596