Visual Grasping with Spectral Clustering and Heuristic Searching for Robot in Cluttered Environments

Wenjie Geng, Zhiqiang Cao, *Senior Member*, *IEEE*, Yingbo Tang, Shuo Wang, Fengshui Jing

Abstract—**Grasping the target object is an essential requirement for the robot to provide better services. It becomes complicated especially in cluttered environments, which still remains challenging. This paper proposes a novel grasping chain generation solution that enables the robot to grasp the target after other obstructed objects are moved in a good order. SSD is firstly adopted to acquire the information of detectable objects and then Euclidean clustering is employed to obtain the untrained objects. After that, the minimum bounding box of each object is obtained, which is then projected on the plane and represented by a smooth differentiable minimum ellipse. On this basis, an information density kernel function is designed to express the interaction between objects. By abstracting each ellipse as a node of the graph whose edge weight is calculated by this kernel function, the whole scene is described in a form of graph. To simplify the complexity of the scene graph, we use spectral clustering algorithm to classify the objects, and the task-oriented objects graph is constructed according to the objects closely related to the target one. As a result, the searching space is reduced. With space division of task-oriented objects graph, each candidate grasping chain is iteratively extended by using the heuristic searching and the best chain with the shortest length is determined. The proposed method solves the barrier caused by secondary obstruction and its effectiveness is testified by experiments.**

I. INTRODUCTION

Nowadays, more and more tasks rely on the participation of robots, and some complex ones even require the manipulation of objects, such as delivering [1-2]. For manipulation, grasping solution is the most popular form, where visual grasping plays a dominant role.

To realize visual grasping, it firstly needs to detect objects. The robustness of traditional detection is usually weak due to the variation of illuminations, and deep learning method [3] has received much attention, such as two-stage Faster R-CNN [4], single-stage YOLO [5] and SSD [6]. Besides, determining stable grasps on objects is also important. Generally, the robot can execute grasp based on the principal axis of object point cloud [7]. According to the result of grasp detection, the robot executes grasping operation for the target object. Zhao et al. adopted SSD to detect objects and obtained corresponding grasping point. A path planning approach based on RRT-Connect and Bezier curve is employed for grasping the target object [8]. However, the environments are simple without interference from other objects. In practice, the target object is often obstructed in complex environments, and the obstructed ones have to be firstly cleared. Wu et al. concerned table cleaning, and the moving order of objects is generated to avoid obstruction from other objects by assigning a given priority of left, top, and front directions [9]. Notice that the cleaning task is not target-oriented and each object has to be grasped. More researches focus on target-oriented solutions, where the obstructed objects are required to be moved out of the way for grasping the target one. Lozano-Pérez et al. presented a strategy for integrating task and motion planning based on a symbolic search for a sequence of high-level operations including pick, move and place [10]. Srivastava et al. provided an interface between task and motion planning, which can generate a new plan by task planner when the target is obstructed in cluttered environments [11]. A problem of references [10][11] is that only a clear path to the target object is provided and the best order is not considered.

Chitnis et al. formulated the grasping order problem caused by obstructed objects as a plan refinement graph, where its nodes contain high-level plans and edges reflect unsatisfied preconditions that explain a failed attempt at refinement [12]. Krontiris and Bekris proposed to search minimum constraint removal paths based on a graph structure in configuration space and chose a better sorting order that balances minimizing constraints, computational cost and path length [13]. These two methods can provide a sorting sequence of the obstructed objects, however, the grasping process is susceptible to influence from secondary obstruction where an object in this sequence may be further obstructed by others. It becomes worse with the increasing of the number of objects, and searching the best sequence tends to be time-consuming. Also, most of existing methods are verified by simulations due to the complexity of grasping problem in cluttered environments.

In order to solve the challenge from secondary obstruction, this paper proposes a novel grasping chain generation solution for manipulating robot in cluttered environments. Firstly, a scene graph is built, whose nodes and edges are objects and the influence between objects labeled as information density, respectively. On this basis, spectral clustering [14] is employed on this graph to obtain a division result, which is combined with target bundling to form a task-oriented searching space for grasping sorting. Compared to the original searching space with all objects, the generated one becomes smaller. The task-oriented searching space is divided into multiple regions to reduce complexity by searching respective scope. Specially, the object is recommended in a heuristic way according to its distances to the target and the robot as well as its orientation relative to the connection vector from the robot's center to the target. Combining the grasping status of the recommended object and the target, the grasping chain is iteratively extended

---

This work was supported in part by the National Natural Science Foundation of China under Grants 62073322, 61633020.

Wenjie Geng, Zhiqiang Cao, Yingbo Tang, Shuo Wang, Fengshui Jing are with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: gengwenjie2017@ia.ac.cn).

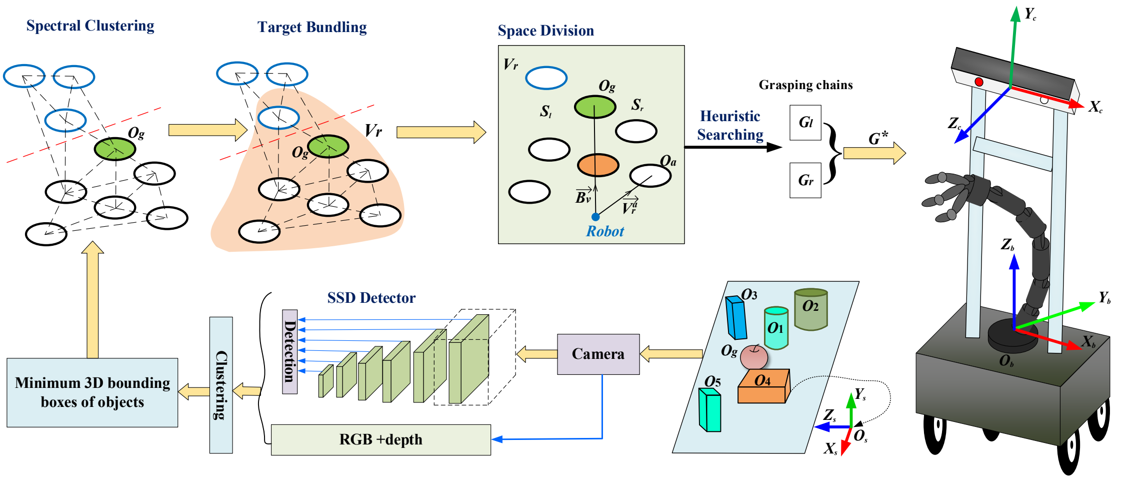


Fig. 1. The overview of visual grasping in cluttered environments. $O_bX_bY_bZ_b$, $O_cX_cY_cZ_c$ and $O_sX_sY_sZ_s$ refer to the robot base coordinate system, the camera coordinate system and local coordinate system on a specific object, respectively. O_g is the target object, and V_r is termed as the task-oriented objects graph. $\overrightarrow{B_v}$ and $\overrightarrow{V_r^a}$ reflect the vectors from the center of the robot to O_g and from the center of the robot to the object O_a, respectively. S_l and S_r are the results of space division and their corresponding grasping chains are G_l and G_r, respectively. G^* describes the best grasping chain.

until the target is appended to the end of the chain. The generated chain is expected to be shorter in length compared to the scheme of sequential traverse. By synthesizing the grasping chains from different regions, the best one can be confirmed. The hierarchical characteristics of our grasp chain solves the problem caused by secondary obstruction.

II. VISUAL GRASPING IN CLUTTERED ENVIRONMENTS

A. *Overview of the Method*

Fig. 1 presents the overview of visual grasping in cluttered environments, which includes object detection, spectral clustering and target bundling for removing objects that are distant from the target on the scene graph, heuristic searching with space division for generating the best grasping chain.

In the grasping process, the scene sensing is the first step. We employ SSD to detect the trained/detectable objects. Combining the RGB and depth information, the point clouds of the detectable objects are obtained. In Fig. 1, O_g, O_1 and O_2 belong to the detectable type. With the table plane fitting using RANSAC [15] and straight-pass filter, the point clouds of undetectable objects (see O_3, O_4 and O_5 in Fig. 1) are then obtained by Euclidean clustering [16]. For each detectable or undetectable object $O_j(j=1,2,\ldots,n)$, it is processed by PCA [17] to get a minimum 3D bounding boxes B^{O_j}, which is used as the representation of the object O_j. For each object, its 3D bounding box is projected on a plane and we get its minimum circumscribed ellipse for simplifying the calculation. The scene graph is then acquired. we further apply spectral clustering is employed on the scene graph to present an optimal division and combine target-bounding to supplement classification results to get task-oriented objects graph V_r. Combining the vector $\overrightarrow{B_v}$ from the center of the robot to the target object as well as the relationship of other objects relative to $\overrightarrow{B_v}$, V_r is divided into two regions S_l and S_r. Further, the corresponding grasp chains G_l and G_r of these regions can be obtained by heuristic searching. Finally, the best grasping chain G^* is determined, which provides the grasping sequence of moving objects for the robot.

We denote with the camera coordinate system $O_cX_cY_cZ_c$ whose origin locates at the center of the camera and Z_c-axis faces forward. $O_bX_bY_bZ_b$ is labeled as robot base coordinate system, where O_b locates in the center of its base, Y_b-axis is reverse to the moving direction of the robot, and Z_b-axis is perpendicular to the base in an upward direction. By the transformation matrix from $O_cX_cY_cZ_c$ to $O_bX_bY_bZ_b$, the position information of objects in $O_bX_bY_bZ_b$ can be obtained with the combination of the camera's intrinsic matrix. All the location information of objects is transformed under $O_bX_bY_bZ_b$, and we consider two grasping ways with top grasp (t_g) and side grasp (s_g) according to object size and the relationship relative to its neighbors. For the former, the robot's palm is required to be perpendicular to the table plane, and thus the grasping pose can be calculated only relying on the principal direction of top surface of B^{O_j}. However, the robot's palm is not fixed to a specific direction in the side grasp, and we need to build a local coordinate system $O_sX_sY_sZ_s$ corresponding to B^{O_j}. The origin O_s refers to the vertex on the undersurface of B^{O_j} with the smallest x coordinate in $O_bX_bY_bZ_b$, and X_s-axis and Z_s-axis are along the directions of short edge and the long edge on the undersurface of B^{O_j}, respectively. On this basis, the grasp point is chosen at the center of the side surface of B^{O_j} intersecting with the plane $O_sX_sY_s$. Combining transform matrix M_s between $O_sX_sY_sZ_s$ and $O_bX_bY_bZ_b$, the 6D grasp pose can be obtained.

B. *Graph representation for Scene and Information Density Spectral Clustering with Target Bundling*

For the case where there are many objects on the table, it is time-consuming to obtain a grasping chain by traversing all the objects, when the target object cannot be directly grasped. In this case, how to reduce the searching scope is the first problem.

Firstly, we proposed to represent the grasping scene by the scene graph $G(V,E)$, where each node v_j in V refers to an object, and E reflects the connection relationship between the objects. Due to the fact that different objects possess different orientations with varying lengths and heights, the representation of the edges E is complex. For each object O_j, $j=1,2,\ldots n$, it is described by minimum circumscribed ellipse E_j^e

of the quadrangle R_j based on the projection of B^{O_j} on the table. In the following, the connection relationships between objects are determined. There does not exist a connection between objects O_k and O_t when the following condition is satisfied.

$$\exists\, q|_{q=1,2,\ldots,n,q\neq k,t} \ \rightarrow\ (x,y)\in zone(P_k,P_t)$$
$$s.t.\ \begin{cases} A_l x + B_l y + C_l = 0 \\ A_q x^2 + B_q xy + C_q y^2 + D_q x + E_q y + F_q = 0 \end{cases} \quad (1)$$

where $P_k(x_{kc}, y_{kc})$ and $P_t(x_{tc}, y_{tc})$ are the centers of the objects O_k and O_t, k, t=1,2,…n, respectively. A_l, B_l and C_l are the parameters of the line l connecting P_k and P_t. A_q, B_q, C_q, D_q, E_q and F_q refer to the parameters of the represented ellipse E_q^e attached to other objects. In other words, if the line segment connecting the centers of two objects does not interact with ellipses corresponding to other objects, it is considered that there is an edge between these two objects in the graph.

For the edges E, its each edge corresponds to a weight termed as information density. Take the weight w_{kt} between the objects O_k and O_t as an example. It is calculated based on information density kernel function of a single object, which is given by:

$$w_k = \begin{cases} C_k & (x,y)\ in\ E_k^e \\ \frac{1}{1+e^{-h_k^2}} \exp\left(\frac{-\sqrt{A_k x^2 + B_k xy + C_k y^2 + D_k x + E_k y + F_k}}{\sigma * e_k}\right) & others \end{cases} \quad (2)$$

where w_k is relevant to the object O_k. $e_k = 1 + \exp(\frac{L_{stk} h_k}{L_{rstk}})$. h_k is the height of B^{O_k} and σ is a given value, which is generally set to a smaller value as a big σ leads to that the spectral clustering results are prone to errors. L_{stk} and L_{rstk} describe the lengths of R_k's long side and short side, respectively.

Fig. 2 visualizes the information density function, where Fig. 2 (a) and Fig. 2 (b) corresponds to 3D view and vertical view. It can be seen that the information density function takes into account the orientation and position of the object, and the influence of an object is reflected continuously.

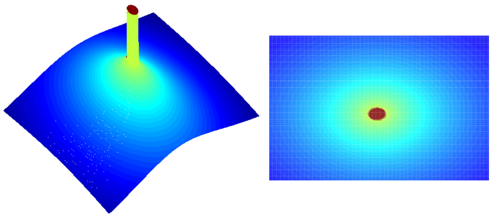

(a) (b)

Fig. 2. The visualization description of the information density kernel function. (a) 3D view. (b) The vertical view.

The information density function of an object reflects its influence on the environment. The influence of the object O_k on the object O_t is labeled as $w_{t/k}$, which is computed by substituting the center coordinate of O_t into w_k. On this basis, the weight w_{kt} is calculated as follows.

$$w_{kt} = w_{t/k} * w_{k/t} \quad (3)$$

where $w_{k/t}$ refers to the influence of the object O_t on the object O_k. According to (3), we get the adjacent matrix W=$[w_{kt}]_{n\times n}$. By adding every row of W, the degree matrix D of the scene graph $G(V, E)$ is obtained, where the degree d_k of each node is equal to $\sum_{t=1}^{n} w_{kt}$, k=1,2,…n. Then, Laplacian matrix is acquired by L=D-W. Note that L, D and W are symmetric matrixes [18].

$$L = D - W = \begin{bmatrix} d_1 & 0 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & 0 & 0 \\ \vdots & \ddots & d_k & \ddots & \vdots \\ 0 & 0 & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & d_n \end{bmatrix} - \left[w_{kt}\right]_{n\times n} \quad (4)$$

We denote with f the arbitrary eigenvector of L and one can get the following expression.

$$\begin{aligned} f^T L f &= f^T D f - f^T W f \\ &= \textstyle\sum_{i=1}^{n} d_i f_i^2 - \sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij} f_i f_j \\ &= \tfrac{1}{2}\textstyle\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}\,(f_i - f_j)^2 \end{aligned} \quad (5)$$

In this paper, $G(V, E)$ is divided into k_1 subgraphs without connection and the vertex set of each subgraph is expressed as A_1, A_2,…, A_{k1}. Thus, $A_1 \cup A_2 \cup \cdots \cup A_{k_1} = V$. For any two vertex sets A and B, where $A \cap B = \emptyset$, the weights of the graph cut between A and B are defined as $W(A,B) = \sum_{i\in A, j\in B} w_{ij}$. Therefore, considering the whole $G(V, E)$ with a more accurate cut result, the graph cuts can be solved by $\frac{1}{2}\sum_{p=1}^{k_1}\frac{W(A_p,A_p^-)}{vol(A_p)}$, where A_p^- is the complement of A_p and $vol(A_p)$ refers to the sum of the nodes' degrees belonging to A_p. The optimizing result of graph cuts is shown as follows.

$$\begin{aligned} & argmin\,\tfrac{1}{2}\textstyle\sum_{p=1}^{k_1}\frac{W(A_p,A_p^-)}{vol(A_p)} \\ &= argmin\,\tfrac{1}{2}\textstyle\sum_{p=1}^{k_1}\left(\sum_{m=1}^{n}\sum_{u=1}^{n} w_{mu}\,(h_{mp} - h_{up})^2\right) \\ &= argmin\,\textstyle\sum_{p=1}^{k_1} h_p^T L h_p \\ &= argmintr(H^T L H) \\ &= argmintr\left(F^T D^{-\frac{1}{2}} L D^{-\frac{1}{2}} F\right)_{|H=D^{-1/2}F} \end{aligned} \quad (6)$$

where $h_{ip} = \begin{cases} \frac{1}{\sqrt{vol(A_p)}} & v_i \in A_p \\ 0 & other \end{cases}$.

Then, spectral clustering is executed using (6) with k_1=2. We denote with the nearest object to the robot O_{nr}, and the objects involved in the grasping process are screened by analyzing the category relationship corresponding to O_g and O_{nr}.

Fig. 3 illustrates the results of spectral clustering, and the cases where O_{nr} and O_g are in the same category or in different ones are shown in Fig. 3(a) and Fig. 3(b), respectively. We label the category near to the robot as V_r, which includes the nodes of interest for grasping. It is noted that the clustering results do not take the robot position into account. Therefore, the target object is required to be absorbed into the category V_r. Furthermore, the target object is often obstructed by adjacent objects, and thus its neighboring objects should be also bundled. When the target object is in the category V_r, as shown in Fig. 3(a), the nodes within a certain distance d_{th} to the target object are added into V_r; otherwise (see Fig. 3(b)), the target node as well as its neighboring nodes shall be placed into V_r.

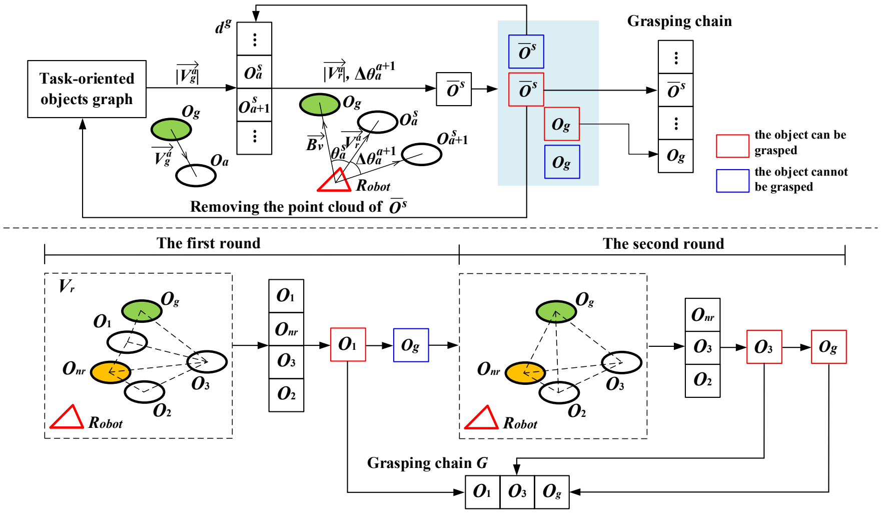
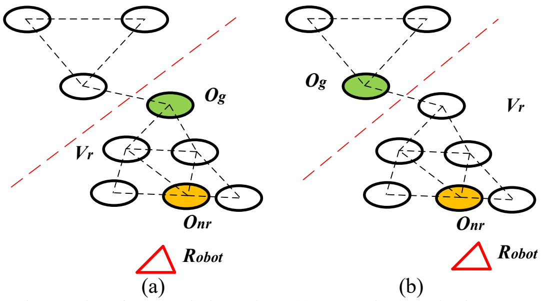


Fig. 4. Heuristic searching for generating the grasping chain.



(a) (b)

Fig. 3. The results of spectral clustering. (a) O_{nr} and O_g are in the same category. (b) O_{nr} and O_g are in different categories.

C. *Heuristic Searching for Grasping Chain*

With the aforementioned task-oriented objects graph V_r whose object number is n_r, the heuristic searching is employed for generating a grasping chain whose end is O_g (see Fig. 4). In this solution, we label as $\overrightarrow{V_g^a}$ and $\overrightarrow{V_r^a}$ the vectors from the center of the target O_g to the object O_a and from the center of the robot to O_a, respectively, where a=1,2,…,n_r. Also, the vector from the center of the robot to O_g is defined as benchmark vector $\overrightarrow{B_v}$, and thus we have the acute angle θ_a between $\overrightarrow{B_v}$ and $\overrightarrow{V_r^a}$. According to the distance $|\overrightarrow{V_g^a}|$, the objects in V_r are sorted in an ascending order to form an initial object list d^g. Then, two metrics of the distance $|\overrightarrow{V_r^a}|$ and the angle difference $\Delta\theta_a^{a+1}$ of adjacent objects O_a^s and O_{a+1}^s relative to $\overrightarrow{B_v}$ are applied to choose a candidate object $\overline{O_s}$. The detailed selection process is depicted by a function $S(O_a^s, O_{a+1}^s)$ in Algorithm 1, where d_{th}^g and d_{th}^p are distance thresholds, and θ_{th} is an angle threshold. $Id^s(*)$ is used to extract a specific object corresponding to $*$.

Algorithm 1. The selection function $S(O_a^s, O_{a+1}^s)$

```
Input: adjacent objects $O_a^s$ and $O_{a+1}^s$ in $d^g$, $O_g$
Output: the candidate object $\overline{O_s}$
compute the vectors $\overrightarrow{B_v}$, $\overrightarrow{V_g^a}$, $\overrightarrow{V_g^{a+1}}$, $\overrightarrow{V_r^a}$, $\overrightarrow{V_r^{a+1}}$;
obtain the acute angles $\theta_a^s$, $\theta_{a+1}^s$;
$\Delta d^g = \left|\overrightarrow{V_g^{a+1}}\right| - |\overrightarrow{V_g^a}|$;
$\Delta\theta_a^{a+1} = \theta_{a+1}^s - \theta_a^s$;
If $\Delta d^g \le d_{th}^g$ then
  $\overline{O_s} \leftarrow Id^s\left(min\left(|\overrightarrow{V_r^a}|, |\overrightarrow{V_r^{a+1}}|\right)\right)$;
else if $d_{th}^g < \Delta d^g < d_{th}^p$ && $\Delta\theta_a^{a+1} \le \theta_{th}$ then
  $\overline{O_s} \leftarrow Id^s\left(min\left(|\overrightarrow{V_r^a}|, |\overrightarrow{V_r^{a+1}}|\right)\right)$;
else if $d_{th}^g < \Delta d^g < d_{th}^p$ && $\Delta\theta_a^{a+1} > \theta_{th}$ then
  $\overline{O_s} \leftarrow Id^s(\max(\theta_a^s, \theta_{a+1}^s))$;
else if $\Delta d^g > d_{th}^p$ then
  $\overline{O_s} \leftarrow Id^s\left(min\left(|\overrightarrow{V_g^a}|, \left|\overrightarrow{V_g^{a+1}}\right|\right)\right)$;
return $\overline{O_s}$
```

If the object $\overline{O_s}$ cannot be grasped, the robot will search the next object in the list d^g. Otherwise, this object shall be added into the grasping chain, and the graspable status of O_g is judged after the point cloud of $\overline{O_s}$ is removed. When it cannot be grasped, a new round judgement starts until O_g can be grasped or all the objects have been traversed. Algorithm 2 presents pseudo-code to generate the grasping chain, where $F_{cg}(\overline{O_s})$ is used to judge whether the object $\overline{O_s}$ can be grasped.

Algorithm 2. A grasping chain generation

```
Input: task-oriented objects graph $V_r$
Output: a grasping chain $G$
compute the list $d^g$ of $V_r$ according to $|\overrightarrow{V_g^a}|$;
$n_{sr}$=$n_r$-1;
$b$=1;
while ($b$<$n_{sr}$) then
  $\overline{O_s} \leftarrow S(O_b^s, O_{b+1}^s)$;
  if $F_{cg}(\overline{O_s})$ is True then
    remove the point cloud of $\overline{O_s}$ from $V_r$;
    add $\overline{O_s}$ into $G$;
    $n_{sr}$=$n_{sr}$-1;
    if $F_{cg}(O_g)$ is True then
      add $O_g$ into $G$;
      return $G$;
    else then
      update $V_r$ as well as the list $d^g$;
    end if
  else then
    $b$=$b$+1;
  end if
end
```

An illustration of grasping chain is presented in the bottom section of Fig. 4, where there are four objects O_1, O_2, O_3 and O_{nr} besides O_g. In the first round, the objects is sorted and stored in d^g: $O_1 \rightarrow O_{nr} \rightarrow O_3 \rightarrow O_2$. According to Algorithm 1, the object O_2 is chosen as the candidate one $\overline{O_s}$. As it can be grasped by the robot, and thus O_2 is added into the grasping

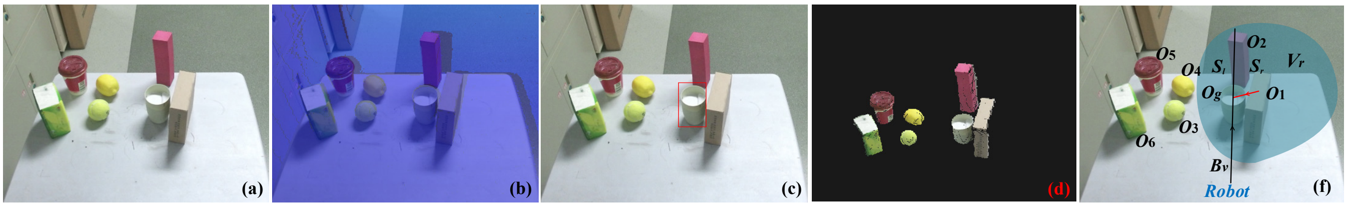


Fig. 5. The results of experiment 1. (a) RGB image of grasping scene. (b) depth image. (c) detection result. (d) point clouds of objects. (e) task-oriented objects graph V_r and the grasping chain.

chain. Accordingly, its point cloud is removed. Due to that the target O_g cannot be grasped, the robot starts to execute the second round. The object O_3 is chosen. O_3 and O_g are both in the graspable state and they are added into the grasping chain. Eventually, the grasping chain $O_1 \rightarrow O_3 \rightarrow O_g$ is obtained.

With the complexity of environments, the space division is considered where V_r is disassembled into left and right regions noted as S_l and S_r, respectively. The cross-product is calculated between $\overrightarrow{B_v}$ and the vector from the center of the robot to an object in V_r. When the cross-product result in z-axis is positive, it belongs to left region S_l. The negative and zero state in z-axis means that it locates at right region S_r. Algorithm 2 is separately executed in each region, and we acquire corresponding results. It is worth noting that a chain to O_g cannot be guaranteed reachable for a region. In this case, the objects in other regions have to be searched. Based on the outputted chains G_l and G_r, the best one G^* is determined by:

$$G^* = \underset{G\in\{G_l,G_r\}}{\operatorname{argmin}}\, len(G) \tag{7}$$

where $len(G)$ represents the length of the grasping chain G. Finally, the robot moves the objects according to the best result G^*. During the operation process, the robot grasps the object and put it in some position which is calculated by an elliptical cone potential field method [19]. Then, the grasping task of the target object in cluttered environments is fulfilled.

EXPERIMENT

The experiments are carried out to testify the effectiveness of the presented visual grasping method. A service robot with a 6-DOF (degree of freedom) Kinova manipulator is used to perform grasping tasks in the executable working space, and Kinect V2 is utilized for scene sensing. Objects are detected by SSD with 2D red bounding boxes, and the point clouds of all the objects are acquired according to PCL (point cloud library). The robot analyzes the scene by spectral clustering to get task-oriented objects graph, and then obtains the grasping chain by heuristic searching for robotic grasping. In the following experiments, apples and cups are detectable objects, and other objects belong to undetectable type.

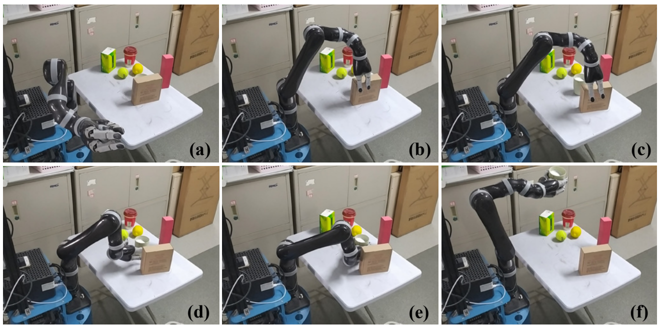


Fig. 6. The video snapshots of the experiment 1.

The scene of the experiment 1 is shown in Fig. 5(a), where there are seven objects and the cup is considered as the target object. Fig. 5(b) denotes the depth image, and Figs. 5(c) and 6(d) present the detection result and the point clouds of all objects, respectively. According to Fig. 5(d), the information density kernel function is used and the result of spectral clustering is obtained. Combining the target bounding on the scene graph $G(V, E)$, the task-oriented objects graph V_r is obtained, as illustrated in Fig. 5(e), where O_g, O_1 and O_2 are included in it. V_r is divided into two regions. For the right region with O_1 and O_2, a grasping chain of $O_1 \rightarrow O_g$ is generated using Algorithm 2. Due to that there is no object in the left region, it has to resort to the right region. The same grasping chain is obtained and it is also the best one. On this basis, the manipulator executes the grasping and the video snapshots are given in Fig. 6. The curves of joint angles during grasping are depicted in Fig. 7. The robot firstly moves the box away and put it on a new position (-0.19, -0.45) according to potential fields with elliptical cone [19]. Afterwards, the target cup is successfully grasped in the form of side grasp.

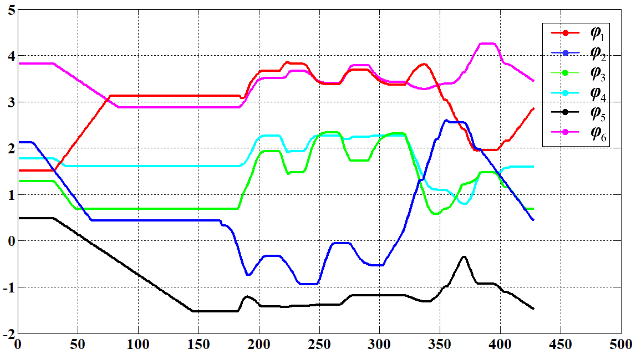


Fig. 7. The curves of joint angles of the experiment 1.

Experiment 2 considers a more complex scenario with 9 objects, where the red apple is the target object. Because the target object is classified into the category away from the robot, the target bundling becomes active, and the target one and other three objects O_1, O_3 and O_4 are absorbed into the other category. The task-oriented objects graph V_r is then constructed, as shown in Fig. 8(e). The objects O_2, O_5, O_6 and O_8 are in the left region, and O_1, O_3 and O_4 belong to the right region. Based on heuristic searching, the robot gets the left chain $O_6 \rightarrow O_5 \rightarrow O_2 \rightarrow O_8 \rightarrow O_4 \rightarrow O_1 \rightarrow O_g$ and the right chain $O_4 \rightarrow O_1 \rightarrow O_g$. Obviously, the right chain shall be chosen because of its shorter length. The video snapshots of the experiment 2 are shown in Fig. 9. The robot firstly grasps O_4 and puts it at (-0.28, -0.39) according to [19], and moves O_1 to the position (-0.17, -0.43) in the same way. Finally, the target apple is grasped and the task is smoothly completed.

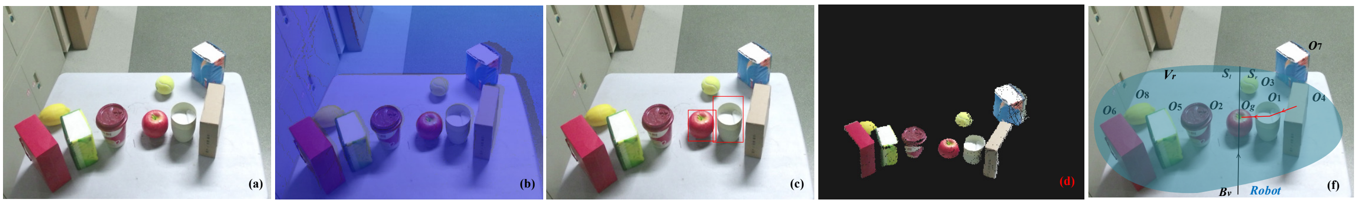


Fig. 8. The results of experiment 2. (a) RGB image of grasping scene. (b) depth image. (c) detection result. (d) point clouds of objects. (e) grasping chain.

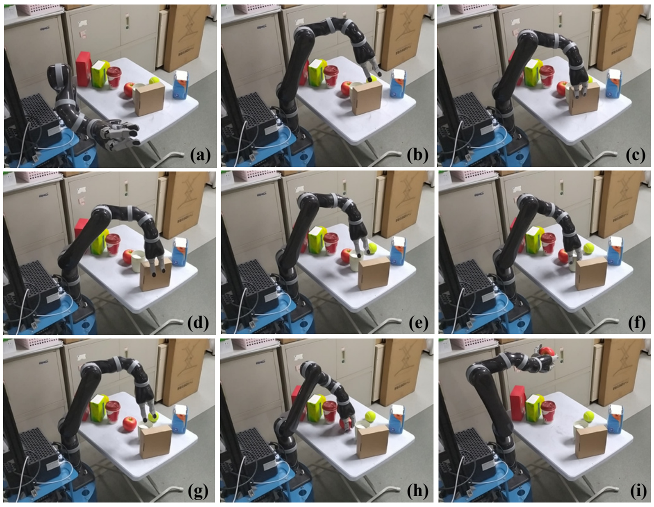


Fig. 9. The video snapshots of the experiment 2.

III. CONCLUSION

In this paper, a visual grasping method with spectral clustering and heuristic searching for robot in cluttered environments is proposed. The task-oriented objects graph is obtained based on spectral clustering and target bundling, and it is disassembled into multiple regions for simplifying complexity of searching. Heuristic searching is used to recommend an object, and the grasping chain can be generated in an iterative way according to the grasping status of the recommended one and the target. Eventually, the best chain is acquired, which provides the robot a decent path to clear the obstructed objects in the pursuit of the target object. The proposed method has been validated by experiments.

REFERENCES

[1] J. Wang, S. Li, “Grasp detection via visual rotation object detection and point cloud spatial feature scoring,” International Journal of Advanced Robotic Systems, vol.18, no.6, November 1, 2021.

[2] C. M. Mateo, P. Gil, F. Torres, “Visual perception for the 3D recognition of geometric pieces in robotic manipulation,” *The International Journal of Advanced Manufacturing Technology*, vol. 83, 1999–2013, 2016.

[3] A. Krizhevsky, I. Sutskever, G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” *Advances in the Neural Information Processing System*, 2012.

[4] S. Ren, K. He, R. Girshick, et al. “Faster R-CNN: Towards real-time object detection with region proposal networks,” *Advances in neural information processing systems*, 2015, 91-99.

[5] J. Redmon, S. Divvala, R. Girshick, et al. “You only look once: Unified, real-time object detection,” *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, 779-788.

[6] W. Liu, D. Anguelov, et al. “SSD: Single shot multibox detector,” European Conference on Computer Vision, arXiv: 1512.02325.

[7] T. Suzuki and T. Oka, “Grasping of unknown objects on a planar surface using a single depth image,” IEEE International Conference on Advanced Intelligent Mechatronics, 2016, 572-577.

[8] X. Zhao, Z. Cao, W. Geng, Y. Yu, M. Tan and X. Chen, “Path Planning of Manipulator Based on RRT-Connect and Bezier Curve,” *IEEE Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems*, 2019, 649-653.

[9] P. Wu, W. Chen, H. Liu, Y. Duan, N. Lin and X. Chen, “Predicting Grasping Order in Clutter Environment by Using Both Color Image and Points Cloud,” *WRC Symposium on Advanced Robotics and Automation*, 2019, 197-202.

[10] T. Lozano-Pérez and L. P. Kaelbling, “A constraint-based method for solving sequential manipulation planning problems,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, 3684-3691.

[11] S. Srivastava, E. Fang, L. Riano, R. Chitnis, S. Russell and P. Abbeel, “Combined task and motion planning through an extensible planner-independent interface layer,” *IEEE International Conference on Robotics and Automation*, 2014, 639-646.

[12] R. Chitnis *et al*., “Guided search for task and motion plans using learned heuristics,” *IEEE International Conference on Robotics and Automation*, 2016, 447-454.

[13] A. Krontiris and K. E. Bekris, “Computational Tradeoffs of Search Methods for Minimum Constraint Removal Paths,” International Symposium on Combinatorial Search, 2015.

[14] U. V. Luxburg. “A Tutorial on Spectral Clustering,” Statistics and Computing, vol. 17, no. 4, 395-416, 2004.

[15] G. Shi, X. Xu and Y. Dai, “SIFT Feature Point Matching Based on Improved RANSAC Algorithm,” *International Conference on Intelligent Human-Machine Systems and Cybernetics*, Hangzhou, 2013, 474-477.

[16] F. Carvalho and L. D. S. Pacifico, “A Weighted Partitioning Dynamic Clustering Algorithm for Quantitative Feature Data Based on Adaptive Euclidean Distances,” *International Conference on Hybrid Intelligent Systems*, Barcelona, 2008, 398-403.

[17] H. M. Ebied, “Feature extraction using PCA and Kernel-PCA for face recognition,” *International Conference on Informatics and Systems*, Cairo, 2012, pp. MM-72-MM-77

[18] J. P. Liu, “A summary of the principles of spectral clustering”, https://www.cnblogs.com/pinard/p/6221564.html.

[19] W. Geng, Z. Cao et al., “A Robotic Grasping Approach with Elliptical Cone-Based Potential Fields under Disturbed Scenes,” International Journal of Advanced Robotic Systems, vol. 18, no.1, 2021.