Open Access

Automatic grasp planning for self-adjustable gripper frames



Abstract

In high-mix, low-volume manufacturing, end-of-arm tooling (EOAT) must be able to handle various products. Conventional EOATs usually comprise fixed or limited-adjustable frames (i.e. manually or using short-stroke pistons) over-equipped with multiple grippers, which are selectively activated based on the product to be grasped. This paper presents a smart gripper frame equipped with only four grippers capable of automatically adjusting to a product's unique geometry. To this end, a two-dimensional grasp planner has been developed that is supplied with product contours from depth images. The proposed approach has been successfully validated in multiple industrial use cases involving objects with different dimensions and materials, and applying various grippers.

Keywords Handling, Planning, Man-machine system

Introduction

Nowadays, the manufacturing industry is increasingly dominated by mass customisation with the production of goods in large variations but low quantities. These products often need to be manipulated multiple times, e.g. during machining, inspection, and packaging [1]. More specifically, 15% to 70% of the product cost is spent on handling [2] with the price of end-of-arm tooling (EOAT) often reaching 20% of the robot cost [3]. These EOATs constitute multiple grippers mounted on a gripper frame: a solid structure carried by a handling device such as a robot or overhead crane. In practice, two main types of gripper frames can be distinguished: fixed frames with an immutable gripper placement and adjustable frames allowing variable gripper positioning through manual adjustments or by employing short-stroke pistons. To be able to handle a wide variety of products and reduce manual interventions, EOATs often end up being

Jeroen Cramer

jeroen.cramer@kuleuven.be

¹ KU Leuven, Dept. Mechanical Engineering, Research unit ACRO,

² Flanders Make @ KU Leuven, B-3001 Heverlee, Belgium



© The Author(s) 2024. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

over-equipped with multiple grippers selectively controlled based on the dimensions of the product at hand; or being fitted out with large-area vacuum grippers that can handle vacuum leakages to a limited extent when not completely being covered by the product.

Besides existing model-based and model-free grasp planners tuned to a particular vacuum gripper's characteristics (e.g. [4–8]), this work discusses the development and validation of a two-dimensional grasp planner for controlling a smart-adjustable gripper frame composed of long-stroke linear guideways (Fig. 1). Provided with only four grippers, this EOAT can cover a wide range of products between 580×275 [mm] and 1100×1050 [mm] with a maximum weight of 300 kg, therewith outstripping conventional fixed and short-stroke gripper frames [9]. In addition, the grippers mounted on the discussed EOAT can be modularly exchanged depending on the characteristics of the products to be lifted and the task to be executed [10]. Where relevant, these grippers can be selected automatically by using, for instance, the web application presented in [11]. Due to the EOAT's scalable design, both industrial robots as well as manually operated hoist systems can act as potential handling devices.

^{*}Correspondence:

Diepenbeek Campus, B-3000 Leuven, Belgium



Fig. 1 3D-rendering of the tetrapodal smart-adjustable gripper frame used to elucidate the presented grasp planner (left) and the kinematic tree superimposed on the top-down view of this gripper frame (right). In addition, different gripper geometries and arrangements, i.e. offsets and angles are illustrated

This paper is organised as follows. Section 2 describes the modelling of gripper frame configurations and the proposed grasp planner. Whether the term "grasp" is applicable to every grasping principle is the cause of some debate. This terminology semantically stems from the act of holding onto something by enclosing or pinching it between one's fingers or hands. Likewise, when an impactive gripper (e.g. jaw gripper) firmly secures an object by means of a force- or form-closure, it is referred to as "grasping". The stability of such grasps depends not only on finding appropriate contact points but also on the impactive forces applied in the three-dimensional space. Therefore, one may question the use of this terminology for astrictive grippers (e.g. suction cups or magnets), which generate a force field and solely rely on the contact area in the two-dimensional plane. However, to preserve consistency with similar publications, this paper also applies the term "grasp planning" to astrictive grippers, although wherever appropriate, "grasp" and the more general term "pick" are used interchangeably. In Section 3, the mechanical setup of the self-adjustable EOAT and the developed product contour extractor from depth images are presented. Section 4 discusses their experimental validation in industrial scenarios, addresses current challenges, and points out future research directions. Finally, Section 5 concludes with a summary of this work.

Two-dimensional grasp planning Modelling gripper frame configurations

The presented approach formalises gripper frames in a similar fashion as the Unified Robot Description Format



Fig. 2 Excerpt from the URDF-like definition of a gripper frame's links, fixed and linear joints, and grippers using the JSON-format. The coordinate frames in all three links are visualised to reveal the underlying transformations

(URDF) [12]. Conceptually, this well-established format among roboticists composes robot models of both links and joints. Links are rigid entities used for connecting joints. Joints, on the other hand, are entities responsible for the dynamics of the robot model by allowing different types of relative motions between neighbouring links.

Programmatically, a gripper frame configuration is formatted in the JavaScript Object Notation (JSON) [13] as displayed in Fig. 2. The highest level specifies the gripper frame's links, joints, and grippers, each holding an array of their respective sub-objects. Besides a unique identifier that characterises the link, joint, and gripper objects, a joint is composed of four additional properties: the type of motion it exhibits, its origin, the adjacent parent and child links, and optionally the identifiers of its interdependent joints (i.e. joints driven by the same motor). In turn, a gripper is composed of four additional properties: its shape, angle, dimensions, and offsets from the gripper's centroid to the link it is attached to.

At this point, two types of joint motions have been considered: *fixed* joints that retain their size, and *linear* joints that can contract or extend along their axes of orientation. The origin property consists of an object representing the 2D-coordinate transformation from the joint's parent to child link. As a result, similar to URDF, the origin of a joint aligns with that of its child link. The translational property corresponds to the displacement vector between both reference frames, whereas the rotational property equals the rotation around the z-axis in the counterclockwise direction. In accordance with best



Fig. 3 Top-down view of some tripod (left) and tetrapod (right) gripper frame configurations. Red lines visualise fixed joints; dark blue lines represent joints that exhibit a linear motion

practices, all values are specified in SI-units and all coordinate systems are right-handed. Finally, the link property contains an object holding the identifiers of the joint's neighbouring parent and child links. It is defined that a linear joint enables its child link to move along the x-axis of its parent link's reference frame within the range dictated by the x-coordinate of the origin's translational property. Consider for instance the example shown in Fig. 2. Due to linear joint *j*2, child link *l*3 is able to translate within a range of 0.0 to 0.5 metres with respect to the local reference frame of *l*2 and this in the direction of *l*2's x-axis. Furthermore, child link *l*3 rotates clockwise at an angle of $\pi/2$ with respect to *l*2's reference frame.

Figure 3 depicts two gripper frame configurations that will be used throughout this work to elucidate the presented grasp planning approach. The first gripper frame, hereinafter referred to as the tripod configuration, is composed of a star-shaped arrangement of six fixed and six linear joints equipped with three gripper mounts at their endings. The second gripper frame, hereinafter referred to as the tetrapod configuration, consists of an orthogonal arrangement of six fixed and six linear joints provided with four gripper mounts at the linear joints' ends.

After specifying the gripper frame's configuration in the described JSON-format, it is converted into the kinematic tree representation as rendered in Fig. 4 for the tripod and tetrapod configurations. A kinematic tree can formally be described as a directed rooted tree: a directed acyclic graph T = (V, E) whose vertices $V_T = L = \{l_1, l_2, \ldots, l_f\}$ correspond to the set of links (i.e. coordinate systems) with length f that make up the entire gripper frame. The directed edges $E_T \subset \{(l_u, l_v) \in L^2 | l_u \neq l_v \land u, v = 1, 2, \ldots, f\}$ represent the connections between neighbouring links as defined by the parent-child relations following from the gripper frame's description in JSON. The mapping from these edges to the EOAT's joints, $m : E_T \mapsto J$



Fig. 4 Kinematic tree representations of the tripod (left) and tetrapod (right) gripper frame configurations. Red lines visualise fixed joints; dark blue lines represent joints that exhibit a linear motion

with $J = \{j_1, j_2, \dots, j_h\}$ and length *h*, is stored in a hash table. The kinematic tree originates from a common root vertex $v_T^{Src} \in V_T$ corresponding to the world link $l_W = v_T^{src}$ (i.e. the world's coordinate system) and terminates in one or more leaf vertices $V_T^{snk} \subset V_T$ representing the gripper links $L_G = V_T^{snk}$ (i.e. the grippers' coordinate systems) with G being the set of gripper indices. Following the definition of a directed rooted tree, each gripper link $l_g \in L_G$ in T is connected to the world link l_W by exactly one directed path $P_g = ((p_r, p_{r+1}) \in E_T | p_0 = v_T^{src} \land p_z \in V_T^{snk} \land r = 0, 1,$ $\ldots, z-1$) with length z. Subsequently, every directed path P_g can be translated into its corresponding joint chain $C_g = (j_1, j_2, \dots, j_z)$ by iteratively applying map function *m* to P_g 's elements. Note that since a rooted tree only allows at most one directed path to each leaf vertex, the existence of redundant joint chains-which would simultaneously be responsible for the same gripper's motion—is ruled out.

Aligning the gripper frame to the product's pose

To lift the object as stably as possible, the EOAT is initially positioned in the centre of area of the product and rotated according to its principal axes. To this end, the centroid of the product's polygon in the two-dimensional Cartesian plane is calculated—assuming a uniform mass distribution-using the area of this polygon, while considering its shell and every cut-out. Subsequently, the moments and product of inertia relative to the object's centroid are obtained by applying the parallel axis theorem. Note that the inertia quantities of all cut-outs are subtracted from those of the product's shell. Finally, the orientations of the principal axes are determined based on these inertia quantities. Oftentimes, these geometrical polygon properties can simply be calculated using offthe-shelf functions from software libraries available in any contemporary programming language.



Fig. 5 Composing the gripper zones for the tripod (left) and tetrapod (right) configurations. The gripper frame is automatically aligned with the product's centre of area and principal axes of inertia. Legend: grey shape: product polygon; black dots: links; red lines: fixed joints; dark blue lines: linear joints; blue rectangles: gripper polygons; black dashed lines: gripper zones

Composing the gripper zones

The EOAT's gripper zones delineate the regions described by all of its grippers when every linear joint is operated at its upper and lower bounds (Fig. 5, dashed lines). Algorithm 1 describes the process of composing zone Q_g of gripper index $g \in G$. For each joint j of this gripper's joint chain C_g , lines 1 to 15 determine the outer motion limits *M* considering the type of this joint. These motion ranges are then grouped into sequence \mathbb{M} in reverse order as opposed to C_g . Line 16 subsequently calculates the Cartesian product of M resulting in all possible outer configurations \mathbb{S} , i.e. sequences of joint sizes, in which the joints of C_g can be adjusted simultaneously. Indeed, joints that operate interdependently are not considered thus far. These constraints will be addressed in Section 2.5. Subsequently, lines 17 to 29 apply forward kinematics to determine the cluster of corners that makes up gripper g's zone. To this end, every chain of transformations $S \in \mathbb{S}$ is traversed from the respective gripper link to the world link for all corners of polygon Y_g that describe g's geometry. These transformed corners are finally clustered together in set Q_c of which the enclosing convex hull represents gripper g's zone polygon Q_g (see Algorithm 1, line 30).¹

Algorithm 1 Composing gripper q's zone

```
Input: Chain C_g = (j_1, j_2, \dots, j_z) of z joint objects,
                 Polygon Y_g = (y_1, y_2, \dots, y_k, y_1)^{[1]} with k corners
                (k \ge 3) representing gripper g's geometry at the
                origin O = (0, 0);
    Output: Polygon Q_g = (q_1, q_2, \dots, q_x, q_1)^{[1]} with x corners (3 \le x \le k \cdot 2^z) representing gripper g's
                   zone.
 1 Joint motion limits \mathbb{M} \leftarrow ();
    Joint orientations \Omega \leftarrow ();
 2
    for i \leftarrow z - 1 to 0 by
                                        -1 do
 3
           Joint j \leftarrow C_g(i);
 4
 5
           Joint orientation \omega \leftarrow \texttt{origin.rot}(j);
 6
           Joint size s \leftarrow \|\texttt{origin.trans}(j)\|;
           Joint motion limit \tilde{M} \leftarrow ();
 7
           if type(j) = fixed then
 8
 9
                 M \leftarrow (s, s);
10
           else if type(j) = linear then
            M \leftarrow (0,s);
11
12
           end
13
           \mathbb{M} \leftarrow \mathbb{M} \cup (M);
           \Omega \leftarrow \Omega \cup (\omega);
14
15 end
16 Chains of joint sizes \mathbb{S} \leftarrow \prod_{i=0}^{z-1} \mathbb{M}(i);
    Transformed corner cluster Q_c \leftarrow \emptyset;
17
     for i \leftarrow 0 to k-1 by 1 do
18
           for
each chain of joint sizes S \in \mathbb{S} do
19
                 Transformed corner y' \leftarrow Y_q(i);
20
                 for j \leftarrow 0 to z - 1 by 1 do
21
                       Joint orientation \omega
22
                                                    -\Omega(i):
                       Joint size s \leftarrow S(j);
23
24
                       Homogeneous transformation matrix
                                   \cos(\omega)
                                                -\sin(\omega) \quad s \cdot \cos(\omega)
                                   \sin(\omega)
                                                 \cos(\omega)
                                                                s \cdot \sin(\omega)
                                                                               :
                                                                      1
                                    0
                                                     0
                          \leftarrow H \cdot y';
\mathbf{25}
26
                 end
                 Q_c \leftarrow Q_c \cup \{y'\};
27
28
           end
29 end
so Gripper zone (i.e. convex hull) Q_q \leftarrow \text{Conv}(Q_c);
```

Determining the product-zone overlaps

However, the previously constructed gripper zones $Q_g \in \mathbb{Q}$ for every gripper index $g \in G$ can still contain sub-regions in which the EOAT's grippers will not be able to establish a steady grasp due to the absence of contact with the product. To discard these regions, the overlapping areas between the product polygon \mathcal{P} and every gripper zone Q_g are determined using set-theoretic intersection operations: $\mathbb{Q}' = \{\mathcal{P} \cap Q_g | Q_g \in \mathbb{Q}\}$. These result in the light orange polygons $Q'_g \in \mathbb{Q}'$, for every gripper index $g \in G$, pictured in Fig. $\tilde{6}$ that visualise the common regions between the product to be grasped and the formerly calculated gripper zones. Note that some grasping devices, including large-area vacuum grippers and magnetic grippers, do not require full coverage of the grasping surface but instead can handle vacuum and field leakages to a limited extent. This degree can be tuned by the overlapping factor Ψ described in the following section.

 $^{^1}$ By convention, the first corner of a polygon should be repeated at its end to obtain a valid closed object.



Fig. 6 Determining the product-zone overlaps for the tripod (left) and tetrapod (right) configurations. Legend: grey shape: product polygon; black dots: links; red lines: fixed joints; dark blue lines: linear joints; blue rectangles: gripper polygons; black dashed lines: gripper zones; light orange areas: product-zone overlaps

Computing and refining the gripper centroid areas

The gripper centroid areas represent the regions in which the centroid of every gripper can be located (Fig. 7, dark orange areas) considering the gripper frame's kinematics and a pre-specified minimum degree of contact between the product and the grippers' surfaces. The latter is controlled by the overlapping factor $\Psi \in \mathbb{R}^{[0,1]}$, which could be useful for grippers that do not require full coverage of the grasping area such as large-area vacuum grippers and magnetic grippers. Lowering Ψ will relieve the constraints set on the grasp planning process by enlarging the gripper centroid areas resulting in more suitable picking points. Algorithm 2 describes the process of computing centroid area A_g of gripper index $g \in G$. From lines 1 to 10, each corner y of this gripper's polygon Y_g in origin O is rotated according to the reverse order of orientations of each joint *j* in this gripper's joint chain C_{σ} . For each line segment of this gripper's product-zone overlap Q'_g , lines 11 to 30 determine the inward offset d_g by which each line segment should be displaced. This shift is determined by the overlapping factor Ψ and the maximum of distances D_g between the gripper's centroid and



Fig. 7 Computing the gripper centroid areas for the tripod (left) and tetrapod (right) configurations. Legend: grey shape: product polygon; black dots: links; red lines: fixed joints; dark blue lines: linear joints; blue rectangles: gripper polygons; black dashed lines: gripper zones; light orange areas: product-zone overlaps; dark orange areas: gripper centroid areas for the individual product

its corners along the normal direction of that line segment. The intersections between these newly created line segments delineate the gripper centroid areas (see Algorithm 2, lines 31 to 37).

Algorithm 2 Computing gripper g's centroid area

```
Input: Chain C_q = (j_1, j_2, \dots, j_z) of z joint objects,
                     Polygon Q_g' = (q_1', q_2', \dots, q_n', q_1')^{[1]} with n corners
                     (n \ge 3) representing gripper g's product-zone
                     overlap,
                     Polygon Y_q = (y_1, y_2, \dots, y_k, y_1)^{[1]} with k corners
                     (k \ge 3) representing gripper g's geometry at the
                     origin O = (0, 0),
                     Overlapping factor \Psi \in \mathbb{R}^{[0,1]};
       \begin{array}{l} \textbf{Output: Polygon } A_g = (a_1,a_2,\ldots,a_m,a_1)^{[1]} \text{ with } m \\ \text{ corners } (3 \leq m \leq n) \text{ representing gripper } g\text{ 's } \end{array} 
                         centroid area:
  1 Rotated gripper polygon Y'_a \leftarrow ();
  2 foreach corner y \in Y_g do
              Rotated gripper corner y' \leftarrow y;
  з
              for i \leftarrow z - 1 to 0 by -1 do
| Joint j \leftarrow C_g(i);
  4
  5
                      Joint orientation \omega \leftarrow \texttt{origin.rot}(j);
  6
  7
                     Gripper corner rotation: y' \leftarrow R_{O,\omega} \cdot y';
  8
              end
              Y'_g \leftarrow Y'_g \cup (y');
  9
10 end
11 Transformed zone line segments \mathbb{L}'_{Q} \leftarrow ();
12 for i \leftarrow 0 to n-1 by 1 do

13 | Zone corner q' \leftarrow Q'_g(i) with q' = (q'_x, q'_y);
              Transformed gripper polygon Y_g^* \leftarrow ();
foreach corner y' \in Y'_g with y' = (y'_x, y'_y) do
Gripper corner translation:
14
15
16
                    \begin{array}{c} T_{q'_xq'_y}:(y'_x,y'_y) \to (y'_x+q'_x,y'_y+q'_y);\\ Y_g^* \leftarrow Y_g^* \cup (T_{q'_xq'_y}\cdot y'); \end{array}
17
              end
18
              Vector \vec{l} \leftarrow Q'_g(i+1) - Q'_g(i) with \vec{l} = (l_x, l_y)
19
                representing the zone line segment;
              Unit normal vector \vec{n}_u \leftarrow (-l_y, l_x)/||\vec{l}||;
20
              Offsets D_g \leftarrow () storing k distances between zone
21
              corner q' and gripper g's individual corners;
for j \leftarrow 0 to k - 1 by 1 do
22
                     Displacement vector \vec{\delta} \leftarrow Y_q^*(j) - q';
23
                     D_g \leftarrow D_g \cup (\vec{\delta} \cdot \vec{n}_u);
24
\mathbf{25}
              end
              Maximum gripper offset d_g \leftarrow max(D_g) \cdot \Psi;
Displacement vector \vec{\sigma} \leftarrow -\vec{n}_u \cdot d_g for the zone line
26
27
                segment, with \vec{\sigma} = (\sigma_x, \sigma_y);
28
              Zone line segment's translation
              \begin{split} & T_{\sigma_x \sigma_y} : (q'_x, q'_y) \to (q'_x + \sigma_x, q'_y + \sigma_y); \\ & \mathbb{L}'_Q \leftarrow \mathbb{L}'_Q \cup (T_{\sigma_x \sigma_y} \cdot Q'_g(i), T_{\sigma_x \sigma_y} \cdot Q'_g(i+1)); \end{split} 
29
30 end
31 Centroid area polygon A_g \leftarrow () of gripper g;
32 for i \leftarrow 0 to n-1 by 1 do
              Line-line intersection corner a \leftarrow \mathbb{L}'_{Q}(i) \cap \mathbb{L}'_{Q}(i+1);
33
              \begin{array}{c} \text{if } a \text{ is inside } Q'_g \text{ then} \\ \mid & \text{Gripper centroid area } A_g \leftarrow A_g \cup (a); \end{array}
34
35
36
              end
37 end
```

In general, outward or inward offset curves (i.e. gripper centroid areas) are determined by assuming that the offset curve's points are at a certain distance in respectively the outward or inward normal direction of the original curve at each point. Besides this definition, implemented in Algorithm 2, the offset curves could also be considered as the (outward or inward) enclosure of copied addition polygons, with their centroids at each point of the original curve. This resembles the Minkowski sum $\mathcal{A} \bigoplus \mathcal{B} = \bigcup_{i} (a_i \bigoplus \mathcal{B}) = \{a + b | a \in \mathcal{A}, b \in \mathcal{B}\}$ of the original curve's polygon \mathcal{A} and the addition polygon \mathcal{B} . To determine the outward offset curve, both polygons have to be oriented counterclockwise; to determine the inward offset curve, the addition polygon has to be oriented clockwise as opposed to the original curve's polygon. This approach is mainly employed when both polygons are convex. However, if at least one polygon is non-convex, the following is required: the decomposition of the non-convex polygon (e.g. \mathcal{A}) into convex sub-polygons $\mathcal{A} \bigoplus \mathcal{B} = \bigcup_i (\mathcal{A}_i \bigoplus \mathcal{B});$ the convolution of the (non-)convex polygons or $\mathcal{A} * \mathcal{B} = \{a + b | a \in \mathcal{A}, b \in \mathcal{B}, t_A = t_B\}$ with t_A and t_B the tangent values of both curves. For the latter approach, the number of counterclockwise turns the original curve makes around any point p (i.e. winding number Ind) must be positive to determine the outward offset curve $\mathcal{A} \bigoplus \mathcal{B} = \{p \in \mathbb{R}^2 | \operatorname{Ind}_{\delta A * \delta \mathcal{B}}(p) > 0\},\$ and equal to one to determine the inward offset curve $\mathcal{A} \bigoplus \mathcal{B} = \{ p \in \mathbb{R}^2 | \operatorname{Ind}_{\delta \mathcal{A} * \delta \mathcal{B}}(p) = 1 \} [14].$

When grippers share linear joints in their kinematic chains, their gripper centroid areas are influenced by each other. Given each gripper's kinematic chain C_g , the motion limits of each joint $j \in J$ is determined by the inverse kinematics of every corner of the centroid areas $A_g \in \mathbb{A}$, as demonstrated by lines 1 to 16 of Algorithm 3. Subsequently, lines 17 to 22 supplement the motion limits of interdependent joints, sharing the same motor, with each other's motion limits. Finally, for each joint $j \in J$, the minimum and maximum motion limits (m_{\min}, m_{\max}) respectively correspond to the maximum lower limit and minimum upper limit of the joint's set of motion limits (see Algorithm 3, lines 23 to 27).

Algorithm 3 Determining shared and/or interdependent motion limits for all joints

Input: Polygons $\mathbb{A} = (A_1, A_2, \dots, A_t)$ representing the centroid areas of all $t = G $ grippers on the gripper frame:						
Output: Interdependent motion limits $\mathcal{M} = (M_1, M_2, \dots, M_h)$ for all h joints of the gripper frame;						
1 Motion limits $\mathcal{M} \leftarrow (\emptyset i = 0, 1, \dots, h)$;						
2 Motion sizes $\mathbb{M} \leftarrow (\emptyset i = 0, 1, \dots, h);$						
s foreach centroid area $A_g \in \mathbb{A}$ do						
4 Gripper g's chain $C_g = (j_1, j_2, \dots, j_z)$ of z joint objects;						
5 foreach corner $a \in A_g$ do						
6 Chain of joint sizes						
$S \leftarrow \texttt{inverse_kinematics}(a, C_g);$						
7 for $i \leftarrow 0$ to $z - 1$ do						
8 Joint $j \leftarrow C_g(i);$						
$\begin{array}{c c} \mathbf{y} \\ $						
$10 \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad \qquad $						
11 end						
12 for $i \leftarrow 0$ to $h - 1$ do						
$14 \qquad \qquad$						
15 end						
16 end						
17 Copy of motion limits $\mathcal{M}' \leftarrow \mathcal{M}$:						
18 Identifiers of joint i 's interdependent joints						
$K \leftarrow \text{interdeps}(j);$						
19 foreach joint $j \in J$ do						
20 $\chi \leftarrow id(j);$						
21 $\mathcal{M}(\chi) \leftarrow \mathcal{M}'(\chi) \cup \bigcup_{\kappa \in K} \mathcal{M}'(\kappa);$						
22 end						
23 for $i \leftarrow 0$ to $h-1$ do						
24 Minimum limit $m_{\min} \leftarrow \max(\{M(0) M \in \mathcal{M}(i)\});$						
25 Maximum limit $m_{\max} \leftarrow \min(\{M(1) M \in \mathcal{M}(i)\});$						
26 Resulting motion limits $\mathcal{M}(i) \leftarrow (m_{\min}, m_{\max});$						
27 end						

Algorithm 4 describes the process of refining centroid area A_g of gripper index $g \in G$ due to the presence of shared and/or interdependent joints. For each joint j of this gripper's joint chain C_g , lines 1 to 9 determine the joint's sequence of motion limits \mathbb{M} and orientations Ω in reverse order as opposed to C_g . Line 10 then calculates the Cartesian product of M resulting in all possible outer configurations \mathbb{S} , i.e. sequences of joint sizes. Subsequently, lines 11 to 21 apply forward kinematics to determine the cluster of corners that makes up gripper g's bounding box. To this end, every chain of transformations $S \in \mathbb{S}$ is traversed from the respective gripper link to the world link for gripper g's centroid c' at the origin O. These transformed centroids are finally clustered together in set C_c of which the enclosing convex hull represents bounding box B. The intersection of this bounding box B with



Fig. 8 Refining the gripper centroid areas for the tripod (left) and tetrapod (right) configurations. Grey shape: product polygon; black dots: links; red lines: fixed joints; dark blue lines: linear joints; blue rectangles: gripper polygons; black dashed lines: gripper zones; light orange areas: product-zone overlaps; dark orange areas: gripper centroid areas for the individual product, which are refined based on shared and interdependent joints

gripper *g*'s centroid area A_g results in the refined version A'_g (see Algorithm 4, lines 22 to 23). Figure 8 depicts these refined areas in which the gripper centroids can be placed (i.e. dark orange areas), taking into account the motion restrictions imposed by shared and interdependent joints. The three central linear joints of the tripod configuration (i.e. j_1 , j_5 and j_9 in Fig. 4) and the two central linear joints of the tetrapod configuration (i.e. j_1 and j_7 in Fig. 4) are driven by the same motor and therefore interdependent.

Algorithm 4 Refining gripper g's centroid area

```
Input: Chain C_g = (j_1, j_2, \dots, j_z) of z joint objects,
                 Polygon A_g = (a_1, a_2, \dots, a_m, a_1)^{[1]} with m
                corners (m \ge 3) representing gripper g's centroid
                 area
                 Motion limits \mathcal{M} = (M_1, M_2, \dots, M_h) for all h
                joints of the gripper frame;
     \textbf{Output:} \ \mathsf{Polygon} \ A'_g = (a'_1,a'_2,\ldots,a'_w,a'_1)^{[1]} with w
                    corners (3 \le w \le m) representing gripper g's
                    refined centroid area;
 1 Joint motion limits \mathbb{M} \leftarrow ()
     Joint orientations \Omega \leftarrow ();
 2
 s for i \leftarrow z - 1 to 0 by -1 do
           Joint j \leftarrow C_g(i);
 4
           Joint orientation \omega \leftarrow \text{origin.rot}(j);
 5
           \Omega \leftarrow \Omega \cup (\omega);
 6
           \chi \leftarrow \operatorname{id}(j);
 7
           \mathbb{M} \leftarrow \mathbb{M} \cup \mathcal{M}(\chi);
 8
     end
 9
10 Chains of joint sizes \mathbb{S} \leftarrow \prod_{i=0}^{z-1} \mathbb{M}(i);
11 Transformed gripper g's centroid cluster C_c \leftarrow \emptyset;
12 foreach chain of joint sizes S \in \mathbb{S} do
           Transformed gripper g's centroid c' \leftarrow O;
for j \leftarrow 0 to z - 1 by 1 do
13
14
                 Joint orientation \omega \leftarrow \Omega(j);
15
16
                  Joint size s \leftarrow S(j);
17
                 Homogeneous transformation matrix
                             \cos(\omega)
                                          -\sin(\omega) \quad s \cdot \cos(\omega)
                             \sin(\omega)
                                            \cos(\omega)
                                                            s \cdot \sin(\omega);
                               0
                                                0
                                                                 1
                     \leftarrow H \cdot c';
18
                 c'
           \mathbf{end}
19
                \leftarrow C_c \cup \{c'\};
20
           C_c
21 end
22 Bounding box (i.e. convex hull) B \leftarrow \text{Conv}(C_c);
23 Refined gripper centroid area A'_q \leftarrow A_g \cap B;
```

Deriving feasible picking points on individual products

After executing the previously discussed steps, Fig. 9 visualises the resulting areas on four manually composed polygons in which the gripper centroids can be positioned securely. In order to obtain the most optimal set of grasping points within these gripper centroid areas (Fig. 9, dark orange areas), several metrics can be devised among which the smallest or largest Euclidean distance from the gripper frame's centre, respectively aiming at the most compact configuration or the least deflection of the product during lifting. The remainder of this paper will employ this second metric.

Extra: deriving feasible picking points on product families

When provided with a (limited) set of product polygons, the presented approach could additionally be adopted for dimensioning a single rigid gripper frame capable of handling every product in that family $\mathcal{P} \in \mathbb{P}$. After calculating the refined centroid area $A'_g \in \mathbb{A}'$ for every gripper index $g \in G$, the common gripper centroid areas of the static frame are found by computing the intersection of the set of corresponding gripper centroid areas across the range of products: $\mathbb{F} = \left(\bigcap_{j=0}^{\|\mathbb{P}\|-1} \mathbb{A}'(i) |\mathbb{A}'(i) \sim \mathbb{P}(j) \right)_{i=0}^{\|G\|-1}$. Figure 10 visualises the resulting common areas for four manually composed polygons in which the gripper centroids can be positioned securely using a static gripper frame. In order to derive the most optimal set of picking points, the metrics mentioned in Section 2.6 apply here as well.



Fig. 9 Deriving feasible picking points on individual products for the tetrapod gripper frame configuration. Legend: grey shape: product polygon; light orange areas: product-zone overlaps; dark orange areas: gripper centroid areas for the individual product, which are refined based on shared and interdependent joints



Fig. 10 Deriving feasible picking points on product families for the tetrapod gripper frame configuration. Legend: grey shape: product polygon; light orange areas: product-zone overlaps; black dashed lines: gripper centroid areas for the individual product, which are refined based on shared and interdependent joints; green areas: gripper centroid areas optimised for handling all four products using a single adjustment of the gripper frame



Fig. 11 SmartHandler tetrapodal gripper frame operating on (left) a nearly cylindrical aluminium fuel tank for trucks, and (right) a wooden EPAL-pallet. The intuitive GUI designed for assisting the operator can be viewed in the left image

Self-adjustable gripper frame Mechanical setup

The grasp planning method described in Section 2 has been deployed on a tetrapod gripper frame configuration (Fig. 11) in the context of the Flanders Make² Smart-Handler project [9]. This gripper frame is horizontally extendable and foldable due to six linear guideways (i.e. rack and pinion drives) powered by servo motors. The two lengthwise guideways at the frame's centre change position symmetrically due to a flip-flop arrangement of two opposite racks moved by one joint pinion. The remaining four guideways along the widthway of the gripper frame displace asymmetrically by four independent servo motors. The four gripper mounts can be modularly equipped with any type of astrictive (e.g. suction cups and magnets) or even ingressive (i.e. needles) grippers. For this work, the gripper frame was mounted on an overhead crane (i.e. hoist) and guided by an operator provided with the following controls: a switch for activating the grippers, and three push buttons for (1) homing the incrementally controlled guideways, (2) capturing a depth image of the current scene containing the product to be manipulated, and (3) automatically adjusting the linear guideways to the optimal picking positions dictated by the grasp planner.

Software setup

A user-friendly graphical user interface (GUI) has been designed for supporting the operator in running the grasp planner and subsequently positioning the gripper frame with respect to the product. In addition to the outcome of the grasp planner (Figs. 12c, d), this GUI displays the live depth stream registered by the LiDAR-camera below the gripper frame. These depth images are supplied to the planning algorithm for determining appropriate picking positions. As depicted in Fig. 12a, the depth data from the Intel® RealSenseTM L515 LiDAR-camera are converted to greyscale images in which darker colours correspond to increased distances with respect to the camera. Using a Gaussian mixture model, every pixel of these depth images is segmented into different clusters (Fig. 12b) according to the layer they belong to, i.e. background, middle ground, and foreground. When there are multiple objects visible in the scene, the product



Fig. 12 Collection of images showing the grasp planning process for a wooden EPAL-pallet: **a** grey-scaled depth image acquisition, **b** product-background segmentation (n = 3), **c** world coordinate conversion, contour smoothing and grasp planning ($\Psi = 0.4$), and **d** the resulting view presented to the user via the GUI

² https://www.flandersmake.be/.

intended by the operator for lifting is estimated based on two metrics: the products' heights and distances to the image centre, allowing to select the product closest to the camera. Finally, the product contours are extracted using Suzuki's topological structural analysis [15] in OpenCV. Additional contour smoothening is achieved by means of the Ramer-Douglas-Peucker's algorithm [16, 17] for straight edges with sharp corners or the Savitzky-Golay's algorithm [18] for curved edges (Fig. 12c).

Results and discussion

Experimental validation

The previous sections offered an insight into the presented two-dimensional grasp planning approach and its application on an in-house developed self-adjustable gripper frame. To validate the robustness of the developed method, five industrial scenarios have been established involving products with various dimensions and materials, different gripper geometries and changing lighting conditions.

These experiments consist of handling of a wooden EPAL-pallet (Fig. 12d), several cardboard sheets with three distinct cut-out patterns (Figs. 13a–c), and a truck's nearly cylindrical aluminium fuel tank (Fig. 13d) using the tetrapodal SmartHandler gripper frame. To this end, four sets of commercially available grippers have been adopted among which Piab's square Kenos[®] KCS (Fig. 13a) and rectangular Kenos[®] KVG (Fig. 13d) largearea vacuum grippers, Goudsmit's circular MagVacu[®]



Fig. 13 Depth images displaying the extracted product contours and picking positions for cardboard sheets with three different cut-out patterns (**a–c**), and a nearly cylindrical aluminium fuel tank for trucks provided with a protruding refuelling opening and a hole for installing a level sensor (**d**). Four commercially available grippers with various geometries were applied during grasp planning: **a** Piab Kenos KCS, $\Psi = 1.0$; **b** FormHand FH-R150, $\Psi = 0.6$; **c** Goudsmit HGC-RO-100-VR-M-G-F, $\Psi = 1.0$; **d** Piab Kenos KVG 480 × 122 [mm], $\Psi = 0.2$

combi gripper (Fig. 13c), and FormHand's circular FH-R150 gripper combining vacuum technology with granulate-filled gripping pads (Fig. 13b). Thanks to the URDF-like modelling of the gripper frame's kinematics, various frame layouts as well as gripper geometries can be configured easily.

Figures 12d and 13 picture the products' inner and outer contours extracted from the depth images in yellow whereas the light-blue polygons represent the most optimal picking positions calculated by the two-dimensional grasp planner. During operation, these images are displayed via the intuitive GUI to guide the operator in positioning the SmartHandler frame before activating its grippers. Currently, the four positions with the largest Euclidean distance from the gripper frame's centre are selected to guarantee a stable manipulation. A future extension might entail implementing different metrics that could be selected automatically or manually during product handling.

Table 1 lists the online computing times for the five experiments. The total time needed for extracting the product's contours and calculating the optimal locations for picking range on average between 5.99 and 8.37 seconds depending on the use case. The largest share is dedicated to the former with the depth image segmentation and contour smoothing being the most computationally demanding sub-operations. However, this is still faster than manually reconfiguring the gripper frame and/or changing grippers. Moreover, if the products are known in advance, the picking positions could be determined offline based on the products' CAD models and uploaded to the self-adjusting gripper frame. On average, grasp planning amounted to 36 percent of the overall computing time using the rather restricted single-board computer installed on the SmartHandler gripper frame. These experiments were performed on an Odroid XU4 with an octa-core Samsung Exynos 5422 processor (2.1 GHz + 1.4 GHz) and 2 GB RAM.

Similar tests on a PC with an i7–7820HQ Intel processor (2.9 GHz) and 32 GB RAM, involving the idealised polygons in Fig. 14 of the actual products, show significant speed gains with the two-dimensional grasp planning requiring on average between 0.028 and 0.11 seconds to complete (see Table 2). In addition to the improved hardware, this is also likely due to the rather limited number of points that describe these ideal polygons. However, simply increasing the smoothing parameters with the aim of reducing the complexity of real product contours will adversely affect the time needed by this pre-processing step. Future research could therefore focus on optimising or implementing faster smoothing algorithms.

Product ID	Figure 12d	Figure 13a	Figure 13b	Figure 13c	Figure 13d
Contour extraction [s]	4.71 ± 0.14	4.29 ± 0.26	4.57 ± 0.17	4.60 ± 0.14	4.64 ± 0.23
Grasp planning [s]	2.71 ± 0.26	1.71 ± 0.14	2.62 ± 0.55	3.77 ± 0.26	2.18 ± 0.28
Total comp. time [s]	7.42 ± 0.34	5.99 ± 0.31	7.19 ± 0.71	8.37 ± 0.40	6.82 ± 0.26

Table 1 Computing times (mean \pm SD, n = 5) for finding the contours and picking positions for the five products displayed in Figs. 12 and 13



Fig. 14 Grasp planning results for the self-adjustable gripper frame applied on four polygons that idealise the actual products from Figs. 12d and 13a–c. In addition, the same grippers were applied on these manually engineered polygons and positioned by the metric with the largest Euclidean distance from the gripper frame's centre. Note that due to the selected overlapping percentage, part of the gripper centroid area calculated for product *b* partially covers one of its holes. Legend: grey shape: product polygon; blue rectangles: gripper polygons; light orange areas: product-zone overlaps; dark orange areas: gripper centroid area for the individual product, which are refined based on shared and interdependent joints

Moreover, it has been demonstrated that the developed grasp planning algorithm is also useful for dimensioning a rigid gripper frame based on the geometries of a set of products. In this study, this functionality was extensively tested on multiple families of ideal product polygons. Acquiring this information from a company's CAD-database could also have been among the possibilities. Going forward, determining the common gripper areas of product families will additionally be validated on real product

Table 2 Computing times (mean \pm SD, n = 5) for calculating the picking positions for the four hand-crafted polygons displayed in Fig. 14

Product ID	Figure 14a	Figure 14b	Figure 14c	Figure 14d
Grasp planning [s]	0.028 ± 0.002	0.082 ± 0.003	0.11 ± 0.01	0.04 ± 0.01

contours obtained by the developed contour extractor from depth images.

Besides a hoist system, the SmartHandler gripper frame has been designed to be additionally mounted on an industrial robot manipulator. Since the open-source Robot Operating System (ROS) supports most robots nowadays, the developed software is currently being incorporated in a ROS2-driver.

Challenges

Despite their promising results with regard to performance and generality, the presented two-dimensional grasp planner and product contour extractor currently face some challenges. This section addresses them pointby-point and outlines potential refinements.

During the experiments conducted in Section 4.1, it is observed that when working with fairly reflective or metallic products, the infrared (IR) laser beam emitted by the LiDAR depth camera underneath the gripper frame could be occasionally (and rather locally) disturbed by direct incident sunlight. In addition, scattering materials or curved surfaces that reflect the emitted IR light in odd ways tend to challenge this camera's underlying depth technology. In those cases, the contour extractor developed in this study might encounter difficulties resulting in rough or partly missing edges. For this reason, the top surface of the fuel tank mentioned in Section 4.1 was covered with masking tape in preparation for the experiments. Note that this concerns a hardware limitation of the selected depth camera rather than a constraint of the developed grasp planning algorithm. Therefore, in environments suffering from bright ambient light, stereo vision cameras equipped with optical filters have shown to yield superior performance. For dealing with featureless two-dimensional surfaces (e.g. paper sheets, evenlypainted panels) or objects with small repetitive patterns, one can rely on active IR stereo vision cameras. A software solution could entail applying more sophisticated refinement techniques to the extracted contours for improving adversely affected sections.

Since the SmartHandler gripper frame approaches the product along its top view, it was initially decided to create a two-dimensional grasp planner. As demonstrated, this resulted in a performant and versatile algorithm

enabling the control of adjustable handling devices, with a preference for products consisting of flat top surfaces. Nonetheless, the application in Fig. 13d proved that slightly curved surfaces can be considered as well, provided that the selected grippers are suited. As such, largearea vacuum grippers with thick compressible foam pads at their gripping surfaces were adopted for picking the near-cylindrical fuel tank. Another interesting application might be the simultaneous lifting of individual objects, e.g. each gripper of the frame handling a cardboard box of a different height. For this, the contours of every product in the depth frame must be combined into a single disjoint polygon for the grasp planner to work on. While the proposed algorithm is capable of dealing with multiple products at once, the gripper frame used in this study is restricted by its kinematics and therefore not able to vertically move each gripper independently. To mitigate this hardware restriction, the product contour extractor was instructed to output only one contour, the closest to the camera (see Section 3.2), for each depth frame. Again, since this is rather a hardware limitation, connecting each gripper via a vertical compensator to the gripper framewhether actively or passively controlled-could potentially open up a novel field of application.

With the aim of achieving a stable pick, the gripper frame is aligned with the product's centroid and principal axes at the start of the planning process (see Section 2.2). For this, it is assumed that the product possesses a uniform thickness and homogeneous mass distribution. This is true for two-dimensional products such as sheet metal, plasterboards, and plywood panels. However, in case of three-dimensional objects and non-homogeneous materials, this initial alignment of the gripper frame might turn out to be sub-optimal. In addition, it may happen that when aligning the gripper frame along its centroid and principle axes, an overlap could not be found for every gripper if the product features large cut-outs. Due to its actual mass distribution and weight, lifting the object in such case might result in an unsatisfactory operation. Also gravitational and dynamic loads due to accelerations are presently not factored in by the grasp planner. A potential solution could be allowing the algorithm to optimise the orientation of the gripper frame with respect to the product's pose prior to grasp planning. Moreover, the user can always set the minimum number of grasping points s/he considers necessary for the intended application, the available gripper frame and its grippers. Ultimately, it remains the responsibility of the operator to assess whether the calculated gripper positioning is capable of providing the necessary lifting force and stability to manipulate the product correctly and safely. If necessary, the relatively fast computing times reported in Section 4.1 allows the grasp planner to be run more than once until a satisfactory gripper positioning is found.

Finally, besides the object's contour, no other product properties are provided as input to the planner. As a result, the operator's ability is called upon to verify that the grippers installed on the gripper frame match the material properties of the product to be lifted, e.g. magnetic grippers for ferromagnetic materials and vacuum grippers for non-porous materials. For adequate gripper selection, one could rely on existing web-based selection tools as presented in [11].

Conclusion

This work presents a two-dimensional grasp planner and depth-based contour extractor for the robust handling of diverse products using a self-adjustable gripper frame. The dimensioning of rigid gripper frames based on product families is additionally discussed. The overall approach has been successfully validated in five industrial use cases involving products with different dimensions and materials, and various astrictive grippers. Future work will target software and hardware related improvements to further enhance the robustness and speed of the proposed method, in particular (1) devising additional metrics that could be selected automatically or manually during product handling, e.g. objectives that favour picking locations with higher contact rates, and (2) implementing faster contour refinement algorithms.

Abbreviations

- 2D Two-dimensional 3D Three-dimensional CAD Computer-aided design EOAT End-of-arm tooling FPAI European Pallet Association GUI Graphical user interface IR Infrared JSON JavaScript Object Notation Lidar Light detection and ranging PC Personal computer RAM Random-access memory ROS Robot Operating System
- URDF Unified Robot Description Format

Acknowledgements

The authors would like to thank Simon-Pierre Verpoort (KU Leuven) and Gert Schouterden (KU Leuven, ACRO) for welding, wiring, and assembling the selfadjustable gripper frame.

Author contributions

J.C. and K.K.: conceptualised the gripper frame that (1) can automatically adapt to the shape and dimensions of the products at hand, and (2) can be modularly equipped with various grasping systems. Formulating the research objectives also fell within their remit. J.C. and M.C.: jointly devised, coded, and integrated the presented 2D grasp planning software into the existing SmartHandler gripper frame. In addition, they carried out the experiments, collected the data, and analysed the results. J.C. and M.C.: both wrote the draft and final versions of the manuscript. K.K.: critically reviewed and edited the manuscript, took care of the project's supervision and administration, and was instrumental in acquiring the necessary funding. All authors read and approved the final manuscript.

Funding

The presented research was conducted in the context of the ICON-project SmartHandler (HBC.2018.0251) of Flanders Make, funded by Flanders Innovation & Entrepreneurship (VLAIO). Martijn Cramer is a SB PhD fellow at FWO (Research Foundation - Flanders) under grant agreements 1SA6919N and 1SA6921N.

Availability of data and materials

The grasp planner's source code together with the datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

Declarations

Competing interests

The authors declare that they have no competing interests.

Received: 29 June 2023 Accepted: 4 February 2024 Published online: 02 March 2024

References

- Fantoni G, Santochi M, Dini G, Tracht K, Scholz-Reiter B, Fleischer J, Kristoffer Lien T, Seliger G, Reinhart G, Franke J, Nørgaard Hansen H, Verl A (2014) Grasping devices and methods in automated production processes. CIRP Ann 63(2):679–701. https://doi.org/10.1016/j.cirp.2014.05.006
- 2. Tompkins JA, White JA, Bozer Y, Frazelle EH, Tanchoco JMA, Trevino J (2003) Facil Plan, 3rd edn. Wiley, New York
- Pham DT, Tacgin E (1992) Grippex: a hybrid expert system for selecting robot gripper types. Int J Mach Tools Manuf 32(3):349–360. https://doi. org/10.1016/0890-6955(92)90007-4
- Gabriel F, Römer M, Bobka P, Dröder K (2021) Model-based grasp planning for energy-efficient vacuum-based handling. CIRP Ann 70(1):1–4. https://doi.org/10.1016/j.cirp.2021.03.019
- You F, Mende M, Štogl D, Hein B, Kröger T (2018) Model-free grasp planning for configurable vacuum grippers. In: You F (ed) IEEE/RSJ International conference on intelligent robots and systems (IROS). IEEE, Madrid, pp 4554–4561. https://doi.org/10.1109/IROS.2018.8594227
- Wan W, Harada K, Kanehiro F (2021) Planning grasps with suction cups and parallel grippers using superimposed segmentation of object meshes. IEEE Transact Robot 37(1):166–184. https://doi.org/10.1109/TRO. 2020.30140
- Zhang H, Peeters J, Demeester E, Kellens K (2021) A CNN-based grasp planning method for random picking of unknown objects with a vacuum gripper. J Intell Robot Syst 103(4):64. https://doi.org/10.1007/ s10846-021-01518-8
- Zhang H, Peeters J, Demeester E, Kellens K (2023) Deep learning reactive robotic grasping with a versatile vacuum gripper. IEEE Transact Robot 39(2):1244–1259. https://doi.org/10.1109/TRO.2022.32261
- Bey-Temsamani A, Schouterden G, den Bergh JV, Meskens J, Incirci T, Kellens K (2022) Intelligent overhead crane improves operator ergonomics and productivity. Proc CIRP 106:132–137. https://doi.org/10.1016/j.procir. 2022.02.167.
- Schmalz J, Reinhart G (2014) Automated selection and dimensioning of gripper systems. Proc CIRP 23:212–216. https://doi.org/10.1016/j.procir. 2014.10.080.
- Cramer J, Demeester E, Kellens K (2022) Development of an assistive webtool for robotic gripper selection. Proc CIRP 106:250–257. https://doi. org/10.1016/j.procir.2022.02.187. (9th CIRP Conference on Assembly Technology and Systems)
- 12. Open Robotics: urdf—package summary. https://wiki.ros.org/urdf. Accessed 26 June 2023.
- Crockford D. Introducing JSON. https://www.json.org/. Accessed 26 June 2023.

- Russo D, Clementini E (2011) Offset curve: Find an inward and outward offset of a polygon. https://www.slideserve.com/colm/offset-curve. Accessed 26 June 2023.
- Suzuki S, Be K (1985) Topological structural analysis of digitized binary images by border following. Comput Vis Graphics Image Process 30(1):32–46. https://doi.org/10.1016/0734-189X(85)90016-7
- Ramer U (1972) An iterative procedure for the polygonal approximation of plane curves. Comput Graphics Image Process 1(3):244–256. https:// doi.org/10.1016/S0146-664X(72)80017-0
- Douglas DH, Peucker TK (1973) Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. Cartographica 10(2):112–122. https://doi.org/10.3138/FM57-6770-U75U-7727
- Savitzky A, Golay MJE (1964) Smoothing and differentiation of data by simplified least squares procedures. Anal Chem 36(8):1627–1639. https:// doi.org/10.1021/ac60214a047

Publisher's Note

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

Jeroen Cramer is a doctoral researcher at KU Leuven, Department of Mechanical Engineering. He received a bachelor's and master's degree in Engineering Technology, specialisation Electromechanics, jointly awarded by KU Leuven and Hasselt University in 2016 and 2017. His research focusses on stiffness-adaptable robotic grasping and gripper selection methodologies. Since 2021, he is affiliated with the Belgian strategic research centre Flanders Make as part of the Manufacturing & Automation core lab.

Martijn Cramer is a doctoral researcher at KU Leuven, Department of Mechanical Engineering. He received a bachelor's and master's degree in Engineering Technology, specialisation Electromechanics, jointly awarded by KU Leuven and Hasselt University in 2016 and 2017. His research focusses on intention-based human-robot collaborative assembly. Since 2022, he is affiliated with the Belgian strategic research centre Flanders Make as part of the Manufacturing & Automation core lab.

Karel Kellens is a professor of Flexible Handling and Assembly Technology and the co-chair of the ACRO (Automation, Computer Vision and Robotics) research group at the Department of Mechanical Engineering of KU Leuven, Belgium. In 2013, he obtained a PhD degree in Engineering Science from KU Leuven. Since 2021, he is affiliated with the Belgian strategic research centre Flanders Make and the manager of the Manufacturing & Automation core lab.