IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Point Cloud-based Grasping for Soft Hand Exoskeleton

Chen Hu[1], Enrica Tricomi[2], Eojin Rho[3], Daekyum Kim[4], Lorenzo Masia[2], Shan Luo[1] and Letizia Gionfrida[1]

***Abstract*—Grasping is a fundamental skill for interacting with and manipulating objects in the environment. However, this ability can be challenging for individuals with hand impairments. Soft hand exoskeletons designed to assist grasping can enhance or restore essential hand functions, yet controlling these soft exoskeletons to support users effectively remains difficult due to the complexity of understanding the environment. This study presents a vision-based predictive control framework that leverages contextual awareness from depth perception to predict the grasping target and determine the next control state for activation. Unlike data-driven approaches that require extensive labelled datasets and struggle with generalizability, our method is grounded in geometric modelling, enabling robust adaptation across diverse grasping scenarios. The Grasping Ability Score (GAS) was used to evaluate performance, with our system achieving a state-of-the-art GAS of 91 ± 2% across 15 objects and healthy participants, demonstrating its effectiveness across different object types. The proposed approach maintained reconstruction success for unseen objects, underscoring its enhanced generalizability compared to learning-based models.**

***Index Terms*—Adaptive Grasping assistance, Soft hand exoskeleton, 3D Vision Perception.**

I. Introduction

INDIVIDUALS with grasping impairments can experience restriction of hand function [1]. For instance, individuals with spinal cord injury or post-stroke lose the ability to extend their fingers, but retain unaffected motor and sensory abilities outside the hand [2]. Such impairments make it nearly impossible to perform essential grasping tasks, posing significant challenges to independence and quality of life [1].

Over the past decade, soft hand exoskeletons have been developed to assist users in performing daily grasping tasks [3] by providing additional force to users' fingers to support grasping activities [4]. Unlike rigid hand exoskeletons, which use 3D-printed stiff materials to immobilise fingers and execute predefined movements, soft hand exoskeletons [5] are made of flexible materials such as fabric and silicone, offering enhanced comfort during use.

Current soft hand exoskeletons enable assistance through pneumatic-based [6] or tendon-driven [7] mechanisms, facilitating finger flexion and extension to aid users in grasping and releasing objects. Tendon-driven soft hand exoskeletons have garnered attention due to their ability to mimic the natural movement patterns of tendons during object grasping and releasing [8]. A challenge for such exoskeletons involves determining how to effectively regulate control based on user intention [3], [9]. Many such devices rely on surface electromyography (sEMG) [10], detecting muscle electrical activity to compensate for reduced muscle activity [11]. However, sEMG signals can be weak or unreliable in individuals with motor impairments, such as those recovering from stroke, limiting their effectiveness in intention-based control.

The integration of visual perception into the control strategies of wearable robots has gained increasing attention in the robotics community [3], offering the potential to enhance adaptability and decision-making in dynamic environments.

Current state-of-the-art methods predominantly rely on data-driven approaches that incorporate deep learning-based visual perception [5], [9]. While these techniques demonstrate strong performance in controlled settings, their application in wearable robots is hindered by three key factors: (1) the requirement for extensively trained vision algorithms, (2) the lack of task-specific datasets tailored to wearable robotic applications, and (3) the computational cost associated with processing high-dimensional visual data in real-time. These constraints pose significant challenges in seamlessly integrating visual feedback into wearable robotic systems.

An alternative to data-driven methods is to leverage geometric modelling for scene understanding [12], which reduces the reliance on large-scale annotated datasets. Depth-based object modelling, for instance, allows the inference of scene geometry [13]. Unlike learned feature representations, which are often sensitive to changes in camera viewpoints, geometric modelling exploits the underlying spatial structure of the environment, making it inherently more adaptable to unseen scenarios [14]. By shifting the computational focus from high-cost inference to geometric reasoning, these methods can enable real-time visual processing, enhancing responsiveness in wearable robotics [15]. Integrating geometric modelling into vision-based control strategies can improve robustness, adaptability, and efficiency, offering a scalable alternative to deep learning-based solutions for real-world assistive applications.

Geometric point cloud has also been used to detect grasp points for robotic grippers [19], but this approach has not yet been explored for wearable robots. Although individuals without impairments can determine grasp points intuitively,

Manuscript received: February, 12, 2025; Revised April, 30, 2025; Accepted August, 6, 2025.

This paper was recommended for publication by Editor Pietro Valdastri upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by The Royal Academy of Engineering Research Fellowship (RF2324-23-229).

The corresponding author is Chen Hu.

[1]C. Hu, S. Luo, and L. Gionfrida are with King's College London, London, WC2R 2LS, UK (email: {tyrone.hu, shan.luo, letizia.gionfrida}@kcl.ac.uk)

[2] E. Tricomi and L. Masia are with the Munich Institute for Robotics and Machine Intelligence, Technical University of Munich, 80333 Munich, Deutschland (email: {enrica.tricomi, lorenzo.masia}@tum.de)

[3]E. Rho is with the School of Computing, KAIST, Daejeon 34141, South Korea (email: djwls9453@kaist.ac.kr).

[4]D. Kim is with the School of Mechanical Engineering and the School of Smart Mobility, Korea University, Seoul 02841, South Korea (email: daekyum@korea.ac.kr).

Digital Object Identifier (DOI): see top of this page.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

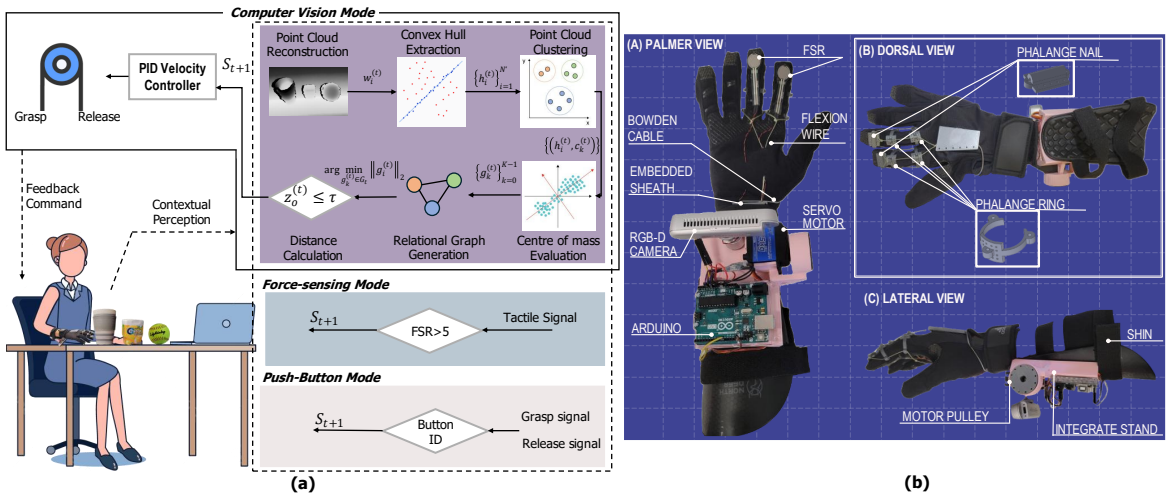


Fig. 1. System design: **(a)** The vision-based controller reconstructs a 3D point cloud from depth frames and sorts points by neighbourhood density. The largest planar model (table) and object convex hulls are identified using PROSAC [16], and point categories are determined using DBSCAN [17]. Centroids computed by Principal Component Analysis (PCA) [18] form an object relationship graph, and the centroid closest to the camera's optical axis is selected as the grasp target. When the object's distance is below the threshold τ, a velocity-based PID controller initiates grasp assistance. *Force-sensing mode:* A fingertip-mounted force-sensitive resistor (FSR) triggers grasp commands based on pressure thresholding. *Push-button mode:* Commands are activated via button presses. **(b)** The soft hand exoskeleton, adapted from [7], comprises an embedded actuator, a customised tendon-driven glove, and integrated sensors. The actuator is mounted on a forearm-worn shin guard for optimised weight distribution. Additionally, 3D-printed nails and rings replicate anatomical tendon paths near finger joints.

those with impairments may struggle to identify when and where to grasp due to cognitive challenges [20]. Point detection can assist those who approach a target object but struggle to determine how to initiate the grasp, enhancing stability, functionality, and the overall effectiveness of the assistance.

Building on these insights, this work introduces a geometric vision-based control system for a soft hand exoskeleton. The proposed framework integrates a wrist-mounted depth camera with a soft wearable robotic hand exoskeleton (Fig. 1). By leveraging ga eometric point cloud, the system reconstructs the 3D scene to identify grasp points and actuates the tendon-driven exoskeleton to execute hand closure. The approach is validated through experiments, including performance comparisons with existing force-sensing and push-button control modalities and tests involving healthy participants performing diverse grasping tasks with different objects. The key contributions of this work include:

1) A geometric point cloud framework to improve generalizability and computational speed compared to traditional data-driven vision controllers.
2) A vision-based controller to detect grasping points that improve the grasping ability score in comparison to other control approaches.

II. Methodology

The proposed vision-based control framework (Fig. 1 **(a)**) was evaluated using a tendon-driven soft hand exoskeleton (Fig. 1 **(b)**), based on an existing design [7], and tested across different objects and grasp types against two existing control modes: force-sensing [7], and push-button [21].

A. Dataset

The dataset comprises 15 objects, including 7 (banana, strawberry, softball, apple, pear, orange, and plum) selected from the Yale-CMU-Berkeley (YCB) dataset [22], designated as *'seen'* objects, and 8 additional everyday objects (chewing gum box, small storage box, purse, small chips can, cup, coffee can, peach can, and chilli can), classified as *'unseen'*, chosen to evaluate the generalizability of the proposed algorithm. Items were chosen to be classified into three distinct grasp types: pinch, spherical grip, and cylindrical grip, as in the study proposed by Calli et al. [22], ensuring heterogeneity across the objects in aspects such as shape, mass, dimensions, and material composition.

B. Hardware Design

The implemented soft hand exoskeleton used in the study was based on an existing design [7], and consisted of three main components (Fig. 1 (b)): 1) a Tendon-Sheath Mechanism (TSM) actuator; 2) a custom glove that transferred force to the finger joints; and 3) a sensing module.

The actuation system, powered by a LiPo battery (Crazepony, 1400mAh, 11.1V, 64.1g, Shenzhen, China), weighed 0.5 kg and was mounted on a forearm-worn shin guard (Super Comfortable Shin Pad, Northdeer, China). A flat servo motor (Digital Servo, 4.8V, 24 kgcm, CHICIRIS, China) drove a 30mm diameter pulley, around which the actuation cable was wound. The system employed a microcontroller (Arduino Uno, Ivrea, Italy) to manage force-sensing analog-to-digital conversion and motor pulse-width modulation (PWM).

The exoskeleton was composed of flexible polyamide, with 3D-printed rings near the metacarpophalangeal (MCP), proximal interphalangeal (PIP), and distal interphalangeal (DIP) joints, replicating human hand tendons for effective grasping and releasing motions. A TSM connected the actuation system to the glove and was designed to remain fixed, reducing the impact of dynamic sheath bending angles. To estimate the fingertip force generated by the TSM, we adopted a joint-

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Object	Name	Grasp Gesture	Grasp Type	Mass (g)	Dimension (mm)	Material	Seen
	Banana		Pinch	80	180 x 37 x 36	Rubber	Y
	Strawberry		Pinch	10	45 x 40 x 52	Plastic	Y
	Chewing gum		Pinch	9	59 x 45 x 77	PP	N
	Storage box		Pinch	50	120 x37 x 55	Plastic	N
	Purse		Pinch	90	120 x 29 x 120	Fabric	N
	Softball		Spherical	200	95	Leather	Y
	Apple		Spherical	60	73 x 73 x 72	Plastic	Y
	Pear		Spherical	40	65 x 65 x 100	Plastic	Y
	Orange		Spherical	50	70 x 70 x 71	Plastic	Y
	Plum		Spherical	30	51 x 51 x 52	Plastic	Y
	Crisp can		Cylindrical	20	74 x 86	Aluminium	N
	Cup		Cylindrical	100	79 x 127	PP	N
	Coffee can		Cylindrical	150	73 x 120	Metal	N
	Peach can		Cylindrical	250	66 x 77	Metal	N
	Sauce can		Cylindrical	210	62 x 116	Glass	N

Fig. 2. Objects used for evaluating the grasping ability across different control modes. Each object is associated with a predefined grasp gesture (Pinch, Spherical, or Cylindrical), and characterized by its mass, dimensions, material, and whether it was included in the training set (Seen). The set includes common household items with varied shapes, sizes, textures, and surface materials to test generalizability under realistic conditions.

level torque transmission model. Based on known design parameters of our glove, the resulting force ranged from nearly 0N to approximately 6.81N (at 90° flexion), reflecting the biomechanical leverage of flexed postures. This force range aligns with fingertip forces commonly observed during activities of daily living (ADL) and ensures safe yet effective grasp assistance [23].

The sensing module, following the existing design [7], incorporated a force-sensing resistor (FSR) sensor (Oumefar, China), integrated into an RC circuit, with a resistance value of R_M set at 10 kΩ and an input voltage V_+ of 5V. The circuit was mounted on the tip of the index finger and an actuation command was activated when the pressure exceeded a defined threshold. To introduce and evaluate the vision-based framework, a wrist-mounted RealSense D415 RGB-D camera (Intel RealSense, California, USA) was incorporated into the soft exoskeleton original design, to capture environmental data. The camera is attached to the shin pad via a miniature gimbal. Depth images are sent to a server (Dell, Precision 7680, US) for inference with a 10 frames per second (fps) frame rate and 640 × 480 resolution. To assess the effectiveness of the vision-based approach, apart from the existing force-sensing mode, the sensing module incorporated an additional control modality: a push-button [21], where actuation commands were transmitted via the corresponding button presses.

C. Control Design

1) Vision-Based control mode: The vision-based process began with the reconstruction of a 3D point cloud from depth frames. The largest planar model (representing the tabletop) and convex hulls (representing objects) were then identified using PROSAC (Progressive Sample Consensus) [16]. PROSAC is used to detect the dominant plane as an outlier by sorting points based on confidence (e.g., local neighbourhood density) and progressively expanding the sample set from high- to low-confidence points, thereby improving model fitting efficiency and robustness, and enabling rapid plane extraction with fewer iterations [24]. As a variant of RANSAC [25], PROSAC [16] differs by prioritising high-confidence samples rather than drawing uniformly at random from the entire dataset. This prioritisation effectively reduces computational overhead and minimises the influence of noisy outliers [16]. In our vision mode, structured object surfaces serve as inliers, while the supporting tabletop is treated as an outlier.

The remaining points were categorised using DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [17]. DBSCAN defines clusters as regions of high point density, identifying core points with sufficient neighbours within a defined radius, and expanding clusters by recursively aggregating density-connected points. It distinguishes between core points, border points, and noise, without requiring the number of clusters to be specified [26], yielding a robust segmentation of object contours in the unstructured point

Algorithm 1: Proposed vision-based control approach

Data: Depth matrix $M_t(u, v)$, camera intrinsic matrix Φ, parameters ϵ, τ, m_0, μ
Result: Estimated PID control state S_{t+1}

while *not thread_stop* **do**
  if $M_t(u, v)$ *exists* **then**
    $P \leftarrow \{p_i^{(t)} = (x_i^{(t)}, y_i^{(t)}, z_i^{(t)})\}_{i=1}^{N}$ from $M_t(u, v)$ and Φ
    $N_\epsilon(p_i^{(t)}) \leftarrow \{p_j^{(t)} \mid \|p_i^{(t)} - p_j^{(t)}\| \leq \epsilon\}$
    $\rho_i^{(t)} \leftarrow |N_\epsilon(p_i^{(t)})|$
    $w_i^{(t)} \leftarrow \frac{\rho_i^{(t)} - \rho_{\min}}{\rho_{\max}^{(t)} - \rho_{\min}^{(t)}}$
    $P \leftarrow \text{Sort}(P, w_i^{(t)})$
    $m(0) \leftarrow m_0$
    repeat
      Select top $m(n)$ points with highest $w_i^{(t)}$
      Fit plane π: $Ax + By + Cz + D = 0$
      Update $m(n+1) \leftarrow \min(N, m_0 + n)$
    until *plane model* π^* *is found with sufficient support*;
    $H_t \leftarrow P \setminus \pi^*$
    $clusters \leftarrow \text{DBSCAN}(H_t, \mu)$
    $G^* \leftarrow \arg\min_{G_k} \|c_k^{(t)}\|_2$
    $distance\_to\_camera \leftarrow \|c_{G^*}^{(t)}\|_2$
    if *motor_state* == *True* **then**
      if $distance\_to\_camera < \tau$ **then**
        $S_{t+1} \leftarrow v_{t+1}$
      else
        $S_{t+1} \leftarrow 90$

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

cloud [27]. PCA (Principal Component Analysis) [18] was subsequently used to compute object centroids, enabling the identification of the target object closest to the camera's optical axis.

These centroids were used to construct a relationship graph among segmented objects, from which the centroid closest to the camera's optical axis was selected as the target. When the Euclidean distance between the target centroid and the camera plane fell below a fixed threshold of 400 millimetres (mm), a PID controller was triggered to initiate grasping. This threshold was selected empirically based on calibration trials (e.g., grasping a banana across 30 repetitions yielded an average distance of approximately 400 mm). While the threshold was fixed, it demonstrated consistent applicability across a diverse set of object shapes and sizes within the 200 mm–1000 mm working range of the RealSense D415, providing a practical balance between robustness and responsiveness.

The essence of the proposed control method, $\mathcal{F}$, was to predict the grasping target based on the current depth frame to determine the velocity PID control state at the next moment. We established the following mathematical model:

$$S_{t+1} = \mathcal{F}(M_t(u,v); \Phi, \epsilon, \tau, m_0, \mu) \tag{1}$$

where $M_t(u,v)$ is the depth matrix obtained by the camera at time t, u and v are the depth matrix coordinates, Φ is the depth camera intrinsic matrix, ϵ is the neighbourhood radius, τ is the PROSAC [16] threshold for determining the maximum plane model's error value, m_0 is the initial sampling range size, μ is the minimum number of points required by DBSCAN [17] to evaluate core points, and S_{t+1} is the velocity PID controller state at time $t+1$. The intrinsic matrix of the RealSense camera is defined as:

$$\mathbf{\Phi} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

Using Open3D [28], we reconstructed the depth matrix $M_t(u,v)$ into a point cloud $P = \{p_i^{(t)}\}_{i=1}^{N}$ (Fig. 1(a)), where N is the total number of points in the point cloud. The point cloud reconstruction outputs are visualised for 15 objects using the proposed vision-based approach (Fig. 5) to illustrate the output of the method.

Any point in this dataset was defined as $p_i^{(t)} = \{x_i^{(t)}, y_i^{(t)}, z_i^{(t)}\}$. Each point's coordinates in the camera coordinate system were calculated as follows:

$$Z = M_t(u,v), X = \frac{(u-c_x)\cdot Z}{f_x}, Y = \frac{(v-c_y)\cdot Z}{f_y} \tag{3}$$

For any point $p_i^{(t)}$ in the scene point cloud, its neighborhood set $N_\epsilon(p_i^{(t)})$ was defined as:

$$N_\epsilon(p_i^{(t)}) = \{p_j^{(t)} \in P_t \mid \|p_i^{(t)} - p_j^{(t)}\| \le \epsilon\} \tag{4}$$

where $\|p_i^{(t)} - p_j^{(t)}\|$ is the Euclidean distance between $p_i^{(t)}$ and $p_j^{(t)}$. The neighborhood density of $p_i^{(t)}$ was defined as $\rho_i^{(t)} = |N_\epsilon(p_i^{(t)})|$. Based on the density $\rho_i^{(t)}$, the confidence of $w_i^{(t)}$ was defined as:

$$w_i^{(t)} = \frac{\rho_i^{(t)} - \rho_{\min}^{(t)}}{\rho_{\max}^{(t)} - \rho_{\min}^{(t)}} \tag{5}$$

Here, $w_i^{(t)} \in [0,1]$, and $\rho_{\min}^{(t)}$ and $\rho_{\max}^{(t)}$ are the minimum and maximum neighbourhood densities among all points. Sorting points by confidence $w_i^{(t)}$ in descending order results in an ordered point set. The top m_0 points with the highest confidence are selected as the initial sampling range. These points were considered most likely to belong to the target plane. In the n-th iteration, the sampling range was progressively expanded, and the expansion was expressed as:

$$m(n) = \min(N, m_0 + n) \tag{6}$$

where $m(n)$ represents the sampling range size in the n-th iteration. The sampling range in the n-th iteration is denoted as P_t'. We randomly select three points from P_t' to fit a plane model $\{P_a', P_b', P_c'\} \subseteq P_t'$. Early iterations used smaller sampling ranges $m(n)$, containing only the highest-confidence points, which were more likely part of the plane. As iterations proceed, the range gradually expands to include more points, ultimately encompassing all points. This ensures that points with initially low confidence but actually part of the plane can be included. Using $\{P_a', P_b', P_c'\}$, the plane normal vector is calculated as:

$$\vec{l} = (P_b' - P_a') \times (P_c' - P_a') = (A, B, C) \tag{7}$$

This gives the plane equation $Ax + By + Cz + D = 0$. The distance from point $p_i^{(t)}$ to the plane model π is expressed as:

$$Dist(p_i^{(t)}, \pi) = \frac{|Ax_i^{(t)} + By_i^{(t)} + Cz_i^{(t)} + D|}{\sqrt{A^2 + B^2 + C^2}} \tag{8}$$

Based on the above formula, the distance of each point to the plane π can be calculated. The support of the plane is evaluated as $I = \sum_{i=1}^{N} 1(Dist(p_i^{(t)}, \pi))$.

After all points were processed, the plane model with the maximum support was selected as the maximum plane model π^*. The complement of the maximum plane model was obtained to get the convex hull point cloud, denoted as: $H_t = \{h_i^{(t)}\}_{i=1}^{N'}$, where $H_t \subseteq P_t$ and $N' < N$. According to DBSCAN, all points were classified into core points, boundary points, and noise points. A core point must satisfy $N_\epsilon(h_i^{(t)}) \ge \mu$; a boundary point satisfies $N_\epsilon(h_i^{(t)}) < \mu$ but is within the neighborhood of a core point; a noise point satisfies $N_\epsilon(h_i^{(t)}) < \mu$ and is not in any core point's neighborhood. DBSCAN first traverses H_t to identify all core points. Then, starting from any unvisited core point h, a new cluster was created. All points within the ϵ neighbourhood of h were added to the cluster. If these points included core points, their neighbourhoods were added to the cluster. Boundary points were automatically added to the nearest core point's cluster. This process repeats until the current cluster no longer expands. Unclustered points were marked as noise. Once all points in H_t were processed, a set of clusters with K categories was obtained: $C_t = \{c_k^{(t)}\}_{k=0}^{K-1}$, where $i = -1$ represents noise points. After removing noise points, the point-class relationship was as: $\{(h_i^{(t)}, c_k^{(t)}) | h_i^{(t)} \in H_t, c_k \in \{0, ..., K-1\}\}$.

Using PCA [18], we computed the centroids of each cluster to represent the scene graph of objects on the table at time t: $G_t = \{g_k^{(t)}\}_{k=0}^{K-1}$.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

The node in G_t closest to the camera optical axis Z was selected as the target object:

$$(x_o^{(t)}, y_o^{(t)}, z_o^{(t)}) = \arg \min_{\mathbf{g}_i^{(t)} \in \mathcal{G}_\sqcup} \|\mathbf{g}_k^{(t)}\|_2 \tag{9}$$

If the distance between the target object and the camera plane at time t was greater than or equal to a threshold $\tau \in (200, 10000)$, measured in mm, the velocity PID at time $t+1$ was set to 90, keeping the motor stationary. If the distance is less than or equal to τ, the velocity PID at time $t+1$ was set to v_{t+1}, making the motor rotate at a constant speed of v_{t+1}, driving the exoskeleton to continuously close.

$$S_{t+1} = \begin{cases} 90, & \text{if } z_o^{(t)} \geq \tau \\ v_{t+1}, & \text{if } z_o^{(t)} < \tau \end{cases} \tag{10}$$

2) Force-sensing control mode: The FSR sensor fixed on the tip of the index finger of the glove, following the original design proposed by [7], was used to initiate grasping. This configuration initiated the grasp command when the fingertip sensors came into contact with an object. The motor received a grip command to rotate forward and output a constant torque of 24 kgcm ($\approx$ 2.35 Nm) when the power system provided 4.8V voltage, assisting the user in grasping and maintaining actions. Based on the benchmarking comparison, this precise torque aided users in grasping objects. The FSR was also connected in series with a 3 kΩ resistor R_M and referenced to a 5V source V_+. The output voltage V_{OUT} increased with increasing force. The relationship between force and voltage under different resistances was derived from the Seeed Studio 101020553 Datasheet [29]. As the grasping motion commenced upon activation of the pressure sensor in contact with the object, a small pressure dead zone was established at a threshold of $\Delta 5$ ADC readings to prevent unintended actions.

3) Push-button control mode: Another common control mode was implemented, where the soft hand exoskeleton control was triggered by pressing a button, as in [21]. When the user needed to grasp an object, they approached the object with their right hand and pressed the grasp button with their left hand. This triggered the motor to rotate forward, delivering a constant torque of 24 kgcm. A PID controller adjusted the phase currents to maintain these parameters, enabling the object to be grasped and maintained stably. Similarly, the user could press the release button to trigger the motor to rotate backwards, resulting in a releasing action.

D. Study Protocol and Evaluation

To assess the grasping performances of the proposed vision-based control architecture against the force-sensing and push-button modes, 10 right-handed healthy participants were involved in the study: seven males and three females aged 25.0±6 years, measuring 1.80±0.22 m in height. All participants demonstrated standard hand motor functions. Participants were instructed to execute grasping tasks, ensuring that these actions were performed without any discomfort or pain. The study received ethical approval (*MRPP-23/24-40750*) from the College Research Ethics Committee at King's College London. Information detailing the study's aims and procedures was provided to all participants, who subsequently gave their written informed consent before involvement.

Once they agreed to participate, participants sat adjacent to a table, as sketched in Fig. 1 (a). Participants had to keep their hands relaxed and avoid applying force on the object during the grasp. A researcher handed the objects to the participants, who were instructed to maintain their grasp for three seconds. Subsequently, participants were asked to rotate their hand 90° to a palm-down position, holding the grasp for another three seconds before the system initiated the release.

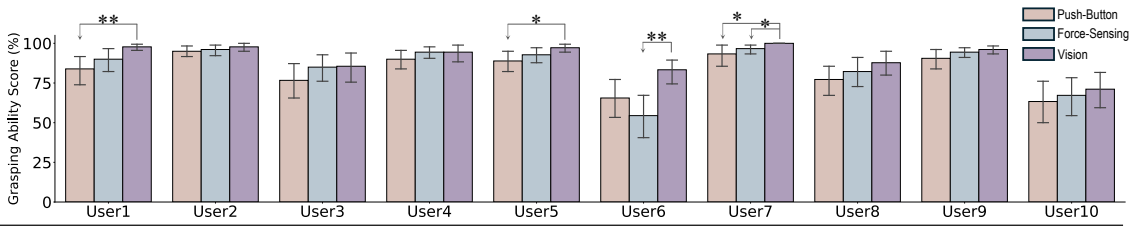
While this controlled setup was adopted for consistency and ease of testing, the vision-based control system is capable of autonomously detecting and responding to objects placed on a tabletop. As demonstrated in Fig. 1 (a) and the supplementary video, it can trigger grasp assistance in multi-object scenarios, supporting extension to practical applications beyond the structured lab setting.

1) Grasping Ability Score: Following the protocol outlined in [30], we evaluated each object's grasp performance across the three control modes. Each object was grasped three times per control mode to comprehensively assess both the initiation of grasp (Grasping) and retention stability (Maintaining). To mitigate cognitive burden and operational complexity for participants, we employed a partially fixed testing sequence within each evaluation cycle (push-button $\rightarrow$ force-sensing $\rightarrow$ vision-based), repeated consecutively for three cycles per object. Participants were explicitly informed about the specific control mode used in each trial because distinct user interactions were necessary for the correct operation of each interface. A limitation of this protocol is that, during the first evaluation cycle for each object, the vision-based controller was always tested last. This ordering raises the possibility that participants may have habituated to the object or improved their interaction with the exoskeleton prior to vision-based trials, potentially inflating success rates (e.g., by adjusting hand orientation or approach angle). However, this partially ordered design balanced the potential adaptation effects from repeated interactions against participant cognitive load and task confusion.

During each trial, Grasping scores were assigned at three distinct levels: 0 indicated grasp failure, 0.5 denoted a successful grasp but incorrect hand positioning, and 1 represented a fully correct grasp posture. Similarly, maintaining scores were rated based on grasp stability: 0 if the object was dropped, 0.5 if held with noticeable instability or movement, and 1 if stably maintained without significant movement. Individual grasp and maintain scores were combined to yield a comprehensive trial score, from which the cumulative Grasping Ability Score (GAS) [31] was calculated, quantifying the overall grasp performance efficacy across the tested modalities.

2) Vision-based mode vs. data-driven approaches: Performance of the proposed visual method over data-driven approaches were evaluated from two key perspectives: response time, and generalizability across *unseen* objects. For data-driven approaches, pre-trained YOLO11n-seg and YOLO11x-seg models [32] were used. We selected YOLO11-seg for instance segmentation of objects within the scene, followed by 3D reconstruction of the target object using Open3D [28].

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.



	User1	User2	User3	User4	User5	User6	User7	User8	User9	User10
Thumb	110	140	110	150	150	130	150	120	140	130
Index finger	170	190	170	180	190	160	200	170	180	170
Middle finger	180	200	190	190	200	170	210	180	190	180

Fig. 3. Grasping Ability Scores (GAS) in percentage (%) for 10 users when grasping 15 objects across three control modes: push-button mode (orange), force-sensing mode (blue), and vision-based mode (purple) are displayed in the bar chart. The table below shows the distances (mm) between the tips of the thumb, index finger, middle finger, and wrist. Longer fingers correlate with higher GAS. Significance levels are indicated by asterisks (*), where * denotes p $\leq$ 0.05, ** denotes p $\leq$ 0.01, and *** denotes p $\leq$ 0.001.

As the user approached the object, we extracted keyframes every 20 frames from the first 100 frames, resulting in a total of 5 keyframes. For these keyframes, we calculated the reconstruction success rates (RSR) as:

$$RSR = \frac{Frame\_Number_{success\_reconstruction}}{Total\_Frame\_Number} \tag{11}$$

3) Kinematics analysis of fingers: The biomechanical impact of the proposed controller on the range of motion (ROM) was evaluated using video recordings captured during the experiment. The ROM was measured at key joints— MCP, PIP, and DIP—during grasping tasks performed with and without the hand exoskeleton across three grasp types: cylindrical (C), spherical (S), and pinch (P). To extract the ROM, YOLO11 [32] was fine-tuned on the 11k Hands Dataset [33] over 100 epochs. The output was then analyzed to compare the ROM with and without external actuation, providing an assessment of the proposed controller's performance.

III. Results and Discussion

A. Grasping Ability Score (GAS)

Figures 3 and 4 report the Grasping Ability Score (GAS) for the vision-based, force-sensing, and push-button control modes, evaluated across 10 participants and 15 test objects. The vision-based mode exhibited slightly superior performance (average GAS: 91±2%), outperforming force-sensing and push-button methods by 6% and 9% in Fig. 3, respectively. This advantage can be attributed to the vision-based system's activation strategy, which includes a brief delay before glove closure, enabling users additional cognitive preparation time to initiate grasps successfully. Object properties, especially surface friction and texture, significantly influenced the GAS; lower-friction objects, such as oranges and coffee cans, presented particular difficulty due to insufficient frictional contact between the exoskeleton and object surfaces by jointly examining Fig. 2 and Fig. 4). Furthermore, Fig. 4 revealed higher success rates for initial grasping compared to maintaining grasp stability, highlighting challenges in prolonged holding, especially for smoother objects. Consequently, enhancing the frictional interface between the exoskeleton and grasped objects may substantially improve grasp stability over extended durations. Statistical comparisons between control modes were performed using paired two-tailed Student's t-tests [34], with

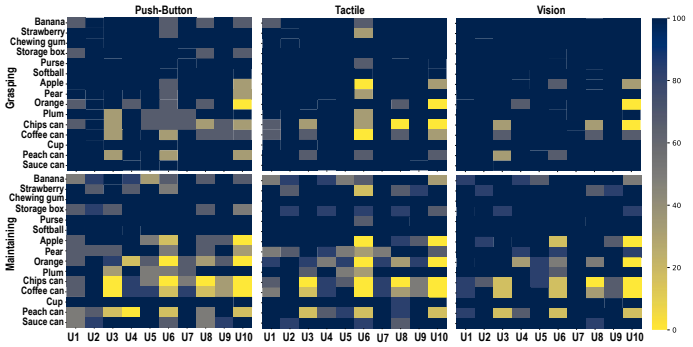


Fig. 4. Confusion matrix depicting the average Grasping Ability Scores (GAS) across 10 users for 15 objects evaluated in three modes: push-button, force-sensing, and vision. The intensity of the colour reflects the GAS, with darker blue indicating higher scores and lighter yellow indicating lower scores.

data normality verified via Shapiro–Wilk tests [35]. Results are presented as mean ± standard deviation, and significance was set at p $\leq$ 0.05. For significant pairwise comparisons, we also report the effect size using Cohen's d. For instance, User1 exhibited a statistically significant increase in GAS (p $<$ 0.01, d = 0.42), indicating a small to moderate effect size.

Furthermore, these figures illustrate consistently smaller error bars for the vision-based mode across participants and object conditions, indicating reduced trial-to-trial variability compared to push-button and force-sensing modes. As detailed in the protocol, participants were explicitly informed of the current control mode during testing. Each evaluation cycle followed a fixed sequence, repeated three times per object to balance cognitive load against potential learning effects. Although this partially ordered approach likely limited pronounced adaptation effects, the reduced variability in vision-based trials may still reflect inherently lower cognitive or motor demands. Future work will adopt a more systematic benchmarking framework, incorporating randomized and counterbalanced trial designs to account for task repetition effects and enable a more rigorous evaluation of performance differences across control modalities.

Tab. I compares grasping scores, maintaining scores, and overall GAS across three grip types, pinch, spherical, and cylindrical, for the proposed vision approach [30]. These results highlight the efficacy of the vision-based method in enhancing grasping performance, particularly in scenarios that require precision (Grasping score) and stability (Maintaining score).

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

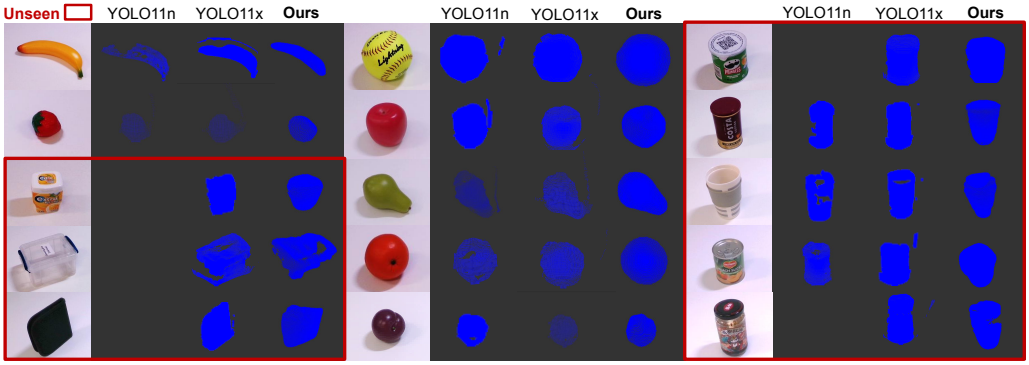


Fig. 5. Visual comparison of the proposed method and data-driven approaches in reconstructing 15 objects. Data-driven models (YOLO11n-seg, YOLO11x-seg) struggle with *'unseen'* objects (red boxes), particularly irregular shapes (strawberry, pear), due to segmentation errors under varying viewpoints. Reconstructions also degrade as the camera moves closer. In contrast, our method achieves consistent, complete reconstructions across textitseen' and textit'unseen' objects, showing generalizability to viewpoint variations without relying on dataset-specific training.

TABLE I
GRASPING PERFORMANCE AVERAGES DIVIDED INTO GRASPING, MAINTAINING, AND TOTAL GAS SCORES FOR THE THREE TYPES OF GRASP.

Method	GAS (%)	Pinch	Spherical Grip	Cylindrical Grip
Push-button	Grasping score	94.67 ± 0.72	88.00 ± 1.89	84.00 ± 1.84
	Maintaining score	89.33 ± 1.19	74.33 ± 3.12	92.33 ± 1.25
	GAS score	92.00 ± 0.95	81.16 ± 2.1	74.16 ± 2.29
Force-sensing	Grasping score	96.67 ± 0.75	91.33 ± 1.89	85.33 ± 1.97
	Maintaining score	89.33 ± 1.59	78.00 ± 2.44	91.73 ± 2.05
	GAS score	93.00 ± 1.17	84.66 ± 2.17	78.33 ± 2.23
Maldonado-Mejía et al. [30]	Grasping score	59.44 ± 3.15	73.33 ± 1.66	68.33 ± 9.61
	Maintaining score	93.33 ± 5.77	94.44 ± 4.81	92.22 ± 4.19
	GAS score	76.39 ± 1.34	83.89 ± 3.15	80.28 ± 3.78
Proposed approach	Grasping score	**100.00 ± 0.00** ↑	**95.33 ± 0.27** ↑	**91.33 ± 0.55** ↑
	Maintaining score	**95.67 ± 0.85** ↑	**87.67 ± 1.72** ↑	76.67 ± 2.25
	GAS score	**97.84 ± 0.43** ↑	**91.50 ± 0.59** ↑	**84.00 ± 0.99** ↑

TABLE II
COMPARISON OF THE PROPOSED VISUAL METHOD AND DATA-DRIVEN APPROACHES IN CPU PROCESSING TIME AND RECONSTRUCTION RATE.

Methods	CPU Processing Time (fps)	Seen (%)	Unseen (%)
YOLO11n-seg	2.89 ± 0.27	77.14 ± 16.08	35.00 ± 17.07
YOLO11x-seg	0.27 ± 0.02	80.00 ± 15.34	77.50 ± 14.94
Proposed approach	**10.72 ± 0.58** ↑	**94.29± 3.36** ↑	**92.50 ± 3.33** ↑

B. Vision-based mode vs. data-driven approaches

To evaluate the advantages of the proposed method over data-driven approaches, we assess its performance across: computational efficiency and generalizability.

1) Computational efficiency and real-time constraints: Table II presents a comparison of CPU processing fps time among the proposed method and two data-driven baselines, YOLO11n-seg and YOLO11x-seg. The proposed approach achieves a processing speed of 10.72 ± 0.58 fps, which is approximately 3.7× faster than YOLO11n-seg, and 39.7× faster than YOLO11x-seg. This substantial improvement highlights the efficiency of real-time applications.

2) Generalisation and zero-shot performance: To evaluate generalizability, we assessed the reconstruction success rate (RSR) on both *'seen'* and *'unseen'* objects (Tab. II). Data-driven models demonstrated varying degrees of generalization failures. YOLO11n-seg achieved 77.14 ± 16.08% RSR for *'seen'* objects but exhibited poor performance (35.00 ± 17.07%) on *'unseen'* instances, highlighting limited adaptability. YOLO11x-seg, with a higher model capacity (62.1M parameters vs. YOLO11n-seg's 2.9M), showed improved but still limited generalization (77.50 ± 14.94% for *'unseen'*). In contrast, the proposed method achieved consistently high RSR of 94.29 ± 3.36% and 92.50 ± 3.33% for *'seen'* and *'unseen'* objects respectively, demonstrating robust zero-shot generalization without object-specific training. Additionally, data-driven models exhibited sensitivity to object geometry and camera viewpoint, with irregularly shaped objects (e.g., strawberries and pears) often failing reconstruction under varied viewing angles due to segmentation inaccuracies and incomplete boundary delineation (Fig. 5).

C. Kinematics Analysis of Fingers

Figure 6 summarizes the joint-level range of motion (ROM) of the index and middle fingers under three grasp types, cylindrical (C), spherical (S), and pinch (P), with and without the soft exoskeleton. For cylindrical grasp, ROM was reduced at nearly all joints, particularly at the DIP and PIP joints (p ¡ 0.001), where deep flexion is typically required to wrap around objects. The MCP joint of the index finger also showed a statistically significant reduction, although the effect size was relatively small. This modest change likely reflects the glove's allowance for proximal joint mobility, while the tension generated along the tendon path may still introduce subtle constraints. The overall reduction in ROM during cylindrical grasping may stem from mechanical interference between the glove and the palmar surface during high-flexion postures and increased resistance from tendon tension routing around the dorsal hand.

For the spherical grasp, a similar trend was observed, though the reductions were slightly less pronounced. The PIP joints again showed the largest ROM suppression, while the MCP joints remained largely unaffected. This suggests that while the exoskeleton imposes constraints on mid-joint movement, it preserves enough proximal flexibility to accommodate the larger, more open-hand configuration typical of spherical grasping.

In the pinch grasp, ROM was largely preserved at the MCP joints, with only selective reductions at the PIP or DIP joints. The pinch task relies more on distal precision and requires less angular excursion at the proximal joints, which may

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

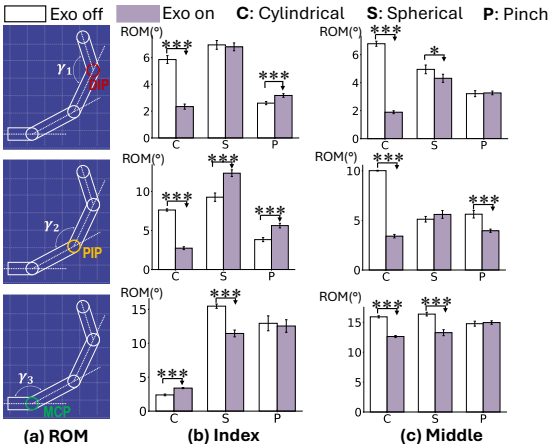


Fig. 6. Hand kinematics analysis showing the Range of Motion (ROM) with the hand exoskeleton on (shaded bars) and off (unshaded bars) during three grasp types: Cylindrical (C), Spherical (S), and Pinch (P). The ROM is measured for the metacarpophalangeal (MCP), proximal interphalangeal (PIP), and distal interphalangeal (DIP) joints of the index and middle fingers.

explain the reduced sensitivity to exoskeletal constraint in this condition.

IV. Conclusion and Future Work

This study introduced a vision-based control framework tested on a tendon-driven soft hand exoskeleton. The approach can perform contextual perception, target object detection, and relational graph generation for PID control, which achieves a GAS of 91±2%. Compared to other control strategies, such as push-button, force-sensing, or data-driven methods, the proposed approach increases accuracy and decreases computational complexity.

Several avenues for future research can further enhance the proposed system. Future efforts will focus on developing dynamic actuators able to infer and modulate grasp strength in response to material properties to handle soft or fragile items, improving versatility and safety in various tasks.

While the current study focuses on the technical development and evaluation in able-bodied participants, future work should involve clinical validation in individuals with motor impairments. Such trials are essential to assess the true functional benefits of the system, such as potential improvements in kinematics and goal attainment, that are likely to be more pronounced in users with reduced mobility.

References

[1] P. Raghavan, "The nature of hand motor impairment after stroke and its treatment," *Current treatment options in cardiovascular medicine*, vol. 9, no. 3, pp. 221–228, 2007.

[2] S. Silver *et al.*, "Peripheral nerve entrapment and injury in the upper extremity," *American Family Physician*, vol. 103, no. 5, pp. 275–285, 2021.

[3] L. Gionfrida *et al.*, "Wearable robots for the real world need vision," *Science Robotics*, vol. 9, no. 90, p. eadj8812, 2024.

[4] T. Du Plessis *et al.*, "A review of active hand exoskeletons for rehabilitation and assistance," *Robotics*, vol. 10, no. 1, p. 40, 2021.

[5] D. Kim *et al.*, "Eyes are faster than hands: A soft wearable robot learns user intention from the egocentric view," *Science Robotics*, vol. 4, no. 26, p. eaav2949, 2019.

[6] L. Ge *et al.*, "Design, modeling, and evaluation of fabric-based pneumatic actuators for soft wearable assistive gloves," *Soft robotics*, vol. 7, no. 5, pp. 583–596, 2020.

[7] E. Rho *et al.*, "Learning fingertip force to grasp deformable objects for soft wearable robotic glove with tsm," *RA-L*, vol. 6, no. 4, pp. 8126–8133, 2021.

[8] S. Coyle *et al.*, "Bio-inspired soft robotics: Material selection, actuation, and design," *Extreme Mechanics Letters*, vol. 22, pp. 51–59, 2018.

[9] E. Tricomi *et al.*, "Environment-based assistance modulation for a hip exosuit via computer vision," *T-RO*, vol. 8, no. 5, pp. 2550–2557, 2023.

[10] B. A. De la Cruz-Sánchez *et al.*, "Emg-controlled hand exoskeleton for assisted bilateral rehabilitation," *Biocybernetics and Biomedical Engineering*, vol. 42, no. 2, pp. 596–614, 2022.

[11] C. Siviy *et al.*, "Opportunities and challenges in the development of exoskeletons for locomotor assistance," *Nature Biomedical Engineering*, vol. 7, no. 4, pp. 456–472, 2023.

[12] R. Diaz Garcia, "Strong geometric context for scene understanding," Ph.D. dissertation, UC Irvine, 2016.

[13] C. Hu *et al.*, "Pointgrasp: Point cloud-based grasping for tendon-driven soft robotic glove applications," *arXiv preprint arXiv:2403.12631*, 2024.

[14] A. Pérez-Yus *et al.*, "Detection and modelling of staircases using a wearable depth sensor," in *ECCV Workshops*. Springer, 2015, pp. 449–463.

[15] D. Driess *et al.*, "Learning geometric reasoning and control for long-horizon tasks from visual input," in *ICRA*. IEEE, 2021, pp. 14 298–14 305.

[16] O. Chum *et al.*, "Matching with prosac-progressive sample consensus," in *CVPR*, vol. 1. IEEE, 2005, pp. 220–226.

[17] M. Ester *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*. AAAI Press, 1996, pp. 226–231.

[18] K. Pearson, "On lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.

[19] H. Duan *et al.*, "Robotics dexterous grasping: The methods based on point cloud and deep learning," *Frontiers in Neurorobotics*, vol. 15, p. 658280, 2021.

[20] N. El Husseini *et al.*, "Cognitive impairment after ischemic and hemorrhagic stroke: a scientific statement from the american heart association/american stroke association," *Stroke*, vol. 54, no. 6, pp. e272–e291, 2023.

[21] R. Alicea *et al.*, "A soft, synergy-based robotic glove for grasping assistance," *Wearable Technologies*, vol. 2, p. e4, 2021.

[22] B. Calli, "Benchmarking in manipulation research: Using the yale-cmu-berkeley object and model set," *RA-M*, vol. 22, no. 3, pp. 36–52, 2015.

[23] N. Kalita *et al.*, "Hand grip forces during adl: Reference data and implications for assistive devices," *South African Journal of Occupational Therapy*, vol. 54, no. 1, pp. 12–20, 2024.

[24] S. Choi *et al.*, "Performance evaluation of ransac family," in *BMVC*. BMVA Press, 2009, pp. 81.1–81.12.

[25] S. Choi, Q.-Y. Zhou, and V. Koltun, "Robust reconstruction of indoor scenes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5556–5565.

[26] E. Schubert *et al.*, "Dbscan revisited, revisited: why and how you should (still) use dbscan," *TODS*, vol. 42, no. 3, pp. 1–21, 2017.

[27] S. M. Ahmed and C. M. Chew, "Density-based clustering for 3d object detection in point clouds," in *CVPR*, 2020, pp. 10 608–10 617.

[28] Q.-Y. Zhou *et al.*, "Open3d: A modern library for 3d data processing," *arXiv preprint arXiv:1801.09847*, 2018.

[29] "Datasheet - seeed studio 101020553," https://uk.rs-online.com/web/p/sensor-development-tools/1887119?gb=s.

[30] Maldonado-Mejía *et al.*, "A fabric-based soft hand exoskeleton for assistance: the exhand exoskeleton," *Frontiers in Neurorobotics*, vol. 17, p. 1091827, 2023.

[31] I. Llop-Harillo *et al.*, "The anthropomorphic hand assessment protocol (ahap)," *Robotics and Autonomous Systems*, vol. 121, p. 103259, 2019.

[32] G. Jocher and J. Qiu, "Ultralytics yolo11," 2024.

[33] M. Afifi, "11k hands: gender recognition and biometric identification using a large dataset of hand images," *Multimedia Tools and Applications*, 2019.

[34] Student, *The Probable Error of a Mean*. Oxford University Press, 1908, vol. 6, no. 1.

[35] S. S. Shapiro *et al.*, "An analysis of variance test for normality (complete samples)," *Biometrika*, vol. 52, no. 3/4, pp. 591–611, 1965.