2025 IEEE International Conference on Robotics and Automation (ICRA)
May 19-23, 2025. Atlanta, USA

Bring Your Own Grasp Generator: Leveraging Robot Grasp Generation for Prosthetic Grasping

Giuseppe Stracquadanio[1], Federico Vasile[1,2], Elisa Maiettini[1], Nicolò Boccardo[3] and Lorenzo Natale[1]

***Abstract*— One of the most important research challenges in upper-limb prosthetics is enhancing the user-prosthesis communication to closely resemble the experience of a natural limb. As prosthetic devices become more complex, users often struggle to control the additional degrees of freedom. In this context, leveraging *shared-autonomy* principles can significantly improve the usability of these systems. In this paper, we present a novel *eye-in-hand* prosthetic grasping system that follows these principles. Our system initiates the approach-to-grasp action based on user's command and automatically configures the DoFs of a prosthetic hand. First, it reconstructs the 3D geometry of the target object without the need of a depth camera. Then, it tracks the hand motion during the approach-to-grasp action and finally selects a candidate grasp configuration according to user's intentions. We deploy our system on the Hannes prosthetic hand and test it on able-bodied subjects and amputees to validate its effectiveness. We compare it with a multi-DoF prosthetic control baseline and find that our method enables faster grasps, while simplifying the user experience. Code and demo videos are available online at this https URL.**

I. INTRODUCTION

Upper limb amputation can drastically change the quality of life of people, impacting their ability to carry out actions that seemed trivial. Recovering the functionality of the amputated limb becomes, up to a certain extent, possible with prostheses. Modern upper-limb prosthetic devices try to push this extent, seeking a seamless integration and embodiment with the user and trying to replicate the key properties of a real human hand. Currently, commercial upper-limb prostheses are based on *electromyography* (EMG) or *mechanomyography* (MMG) as user-input interfaces, allowing users to control the Degrees of Freedom (DoFs) of the prosthetic device by relating the user signals to motors' velocities [1]. A standard approach to multi-DoF control consists in the Sequential Switching and Control (SSC) paradigm. Following this method, the user drives each joint individually through EMG signals and an explicit trigger is required to switch between them. However, as the number of available DoFs increases, the control becomes complex and unintuitive [2]. For this reason, simplifying the user-prosthesis communication is one of the most interesting research problems in prosthetics. An emerging research direction proposes a *shared-autonomy* [3], [1] framework, in which the collaboration with a semi-autonomous system relieves the users from exhausting and unnatural actions, while still being able to operate the device according to their intentions. This framework usually leverages additional input modalities, such as inertial measurements, images and depth information to accomplish the task [4], [5], [6], [7]. For instance, previous works have already considered the integration of a RGB camera for visual recognition to predict a grasp pre-shape [4], [5], [6]. Following such approaches, in this work, we introduce a novel eye-in-hand vision-based prosthetic grasping pipeline, drawing inspiration from the human cognitive process involved in the action of grasping. Generally, when we observe an object that we want to grasp, we perceive its geometry and, unconsciously, evaluate the most suitable grasp among different possibilities. Immediately, we also begin to plan the trajectory that our hand will follow to reach the object, and finally grasp it. Each module of our semi-autonomous control system is designed to emulate the steps of this process. First of all,

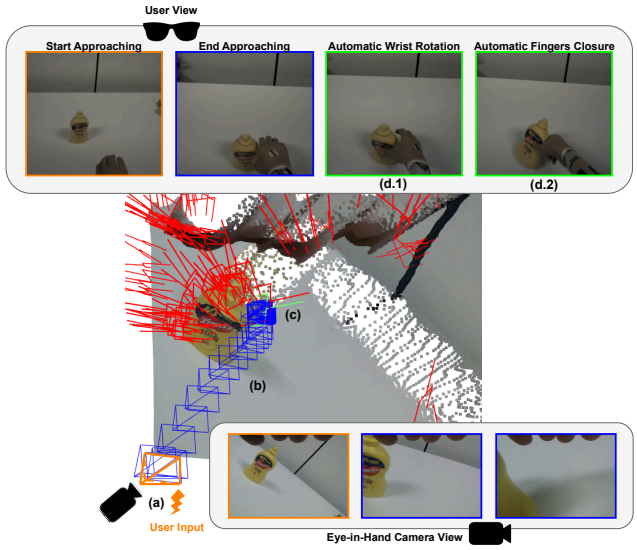

Fig. 1: The phases of our grasping pipeline. A demonstration of semi-autonomous grasp is shown through the user perspective.

[1]Giuseppe Stracquadanio, Federico Vasile, Elisa Maiettini and Lorenzo Natale are with the Istituto Italiano di Tecnologia, Humanoid Sensing and Perception, 16163 Genoa, Italy (email: name.surname@iit.it).

[2]Federico Vasile is also with the Dipartimento di Informatica, Bioingegneria, Robotica e Ingegneria dei Sistemi (DIBRIS), University of Genova, 16146 Genoa, Italy.

[3]Nicolò Boccardo is with the Istituto Italiano di Tecnologia, Rehab Technologies, 16163 Genoa, Italy, and also with the Open University Affiliated Research Centre at Istituto Italiano di Tecnologia (ARC@IIT), 16163 Genoa, Italy (email: nicolo.boccardo@iit.it) .

This work received support by the European Union's Horizon-JU-SNS-2022 Research and Innovation Programme under the project TrialsNet (Grant Agreement No. 101095871) and the project RAISE (Robotics and AI for Socio-economic Empowerment) implemented under the National Recovery and Resilience Plan, Mission 4 funded by the European Union – NextGenerationEU.

2025 IEEE International Conference on Robotics and Automation (ICRA) | 979-8-3315-4139-2/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICRA55743.2025.11127995

we use a depth estimation network to reconstruct geometric information about the object. Then, we exploit a grasp pose generation model to predict multiple grasp candidates (Fig. 1a). Subsequently, a visual odometry model estimates the hand trajectory during the approach (Fig. 1b), allowing us to select a candidate according to the user's intentions (Fig. 1c). Finally, we execute this pose on our prosthesis (Fig. 1d). We make the following contributions: (a) we propose a novel eye-in-hand prosthetic grasping pipeline, designed to improve the user experience by automatically setup a grasp configuration according to user's intentions, (b) we deploy it on Hannes [8], validating its effectiveness on able-bodied subjects and (c) testing its robustness and embodiment on amputees, while conducting (d) an early-stage analysis on cognitive load.

II. RELATED WORK

A. Monocular Depth Estimation

Monocular depth estimation is the task of estimating the depth for each pixel of a single RGB image. Knowing the depth, together with the camera calibration matrix K, allows to un-project each image pixel back to 3D. Current state-of-the-art methods are mostly based on transformer foundational models, such as DINOv2 [9] or Depth-Anything [10]. To allow generalization across multiple datasets, these methods are trained with an affine-invariant depth estimation objective [11], regressing a *relative* (up-to-scale) depth. However, some applications, like ours, would require a *metric* (i.e., with *absolute* scale) depth. In this case, zero-shot inference to a new scenario is not directly possible, before proper fine-tuning on a custom dataset. Therefore, in this work, we generate a synthetic dataset to fine-tune our model.

B. Vision-based Prosthetic Grasping

Several vision-driven prosthetic grasping pipelines have been proposed in literature. In [4], the authors propose an *eye-in-hand* system based on visual recognition of a *known* target object and sensor-fusion to guide the selection of the candidate grasp trajectory from an existing database. The automatic execution of a grasp pre-shape was studied by [12], simultaneously predicting the target object in clutter and the current step in the temporal evolution of the grasping action. To support different grasp types for a single object, [6] proposes a synthetic data generation pipeline leveraging the approaching trajectory to automatically label the object parts with grasp pre-shapes [8], [13]. A pipeline based on depth-perception from a depth camera is described in [5], where a grasp size and wrist rotation were estimated after fitting geometric primitives to the object. Recently, [7] proposed to reconstruct the object geometry to infer information (i.e., the object diameter) for automatic finger closure, relying on a time-of-flight depth sensor. In this work, similarly to [5] and [7], we rely on geometric structure of the objects, but employ a monocular depth estimation network and avoid the need for any depth sensor. Moreover, differently from previous works, we aim to leverage the know-how from robotic grasping methods and apply it to a prosthetic scenario. Specifically, grasping hypotheses are computed at the beginning of the action, when the full object is visible in the camera frustum and are subsequently tracked during the action using a visual odometry module. This allows detecting the most suitable grasping candidate from the initial hypothesis and finally execute it on the prosthetic hand.

C. Vision-based Robotic Grasping

Two different research directions can be identified in the vision-based robotic grasping literature, depending on the type and DoFs of the robot end-effector. The first, dealing with parallel-jaw grippers, includes works that present several gripper parameterizations and end-to-end architectures for learning to *generate* [14], [15] or *regress* [16], [17] grasp distributions over point clouds. The second direction, instead, studies grasping with humanoid and dexterous hands [18], [19], [20], [21]. Although these methods allow a higher diversity of dexterous grasps, we decided to build on top of the current state-of-the-art with grippers that, in addition to being more mature, has an immediate simplicity of use without making particular hypotheses on hand kinematics. In this work, we adapt the Contact-GraspNet [16] gripper representation to our multi-DoFs Hannes hand [13]. Other similar works, such as [17], can be adapted with little modifications to account for a different gripper parameterization.

D. Visual Odometry

Visual Odometry (VO) is the task of estimating camera position and orientation using visual information. A closely related problem is Simultaneous Localization and Mapping (SLAM), where a stream of images is processed in real time to estimate the camera position and simultaneously build a 3D map of the environment [22], [23], [24]. Both VO and SLAM typically use feature tracking at the pixel level to establish 2D-2D correspondences across consecutive frames, which are then used as inputs for optimization objectives, to solve for the camera poses and 3D point coordinates. SLAM methods further continuously compute corrections to improve map global consistency and mitigate the drift resulting from accumulated errors in VO. We rely on a SLAM method to perform VO, in order to benefit from this additional robustness. More specifically, we integrate DPVO [24] to process streams of images in real time. This method performs sparse RGB patch tracking. This increases inference speed and lowers the memory usage, making its adoption suitable in a prosthetic scenario. However, it is fair to say that any other real-time VO technique can be easily adapted into our grasping architecture.

III. METHODS

A. Overview

We designed our vision-based prosthetic grasping system following the shared-autonomy framework. Specifically, the user is responsible for pointing their hand at the target object and triggering the starting signal, using the EMG user-interface. Then, the user approaches the object, while the system performs online tracking of the hand position

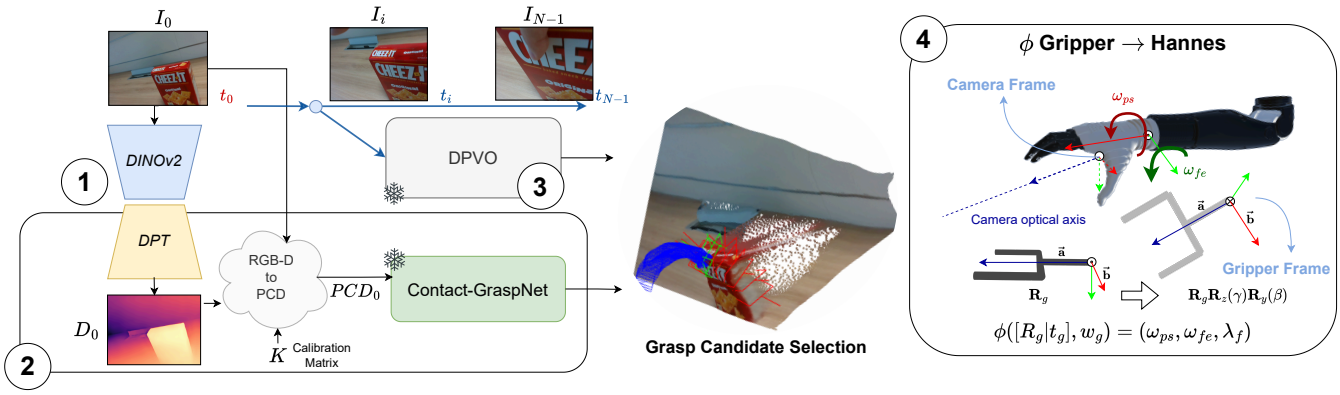

Fig. 2: Details of our grasping pipeline. (1) A depth D_0 is estimated from the first frame of the grasping sequence. (2) D_0 is used to build the point cloud PCD_0 and generate a distribution of grasp poses. (3) The hand-trajectory is estimated by a visual odometry module, and used to select a grasp candidate. (4) The candidate pose is mapped to Hannes.

and finally predicts the most proper Hannes configuration for grasping. More precisely, as soon as the starting signal is triggered, an image from the eye-in-hand camera is captured and grasp candidates are predicted (Fig. 2.2). However, since the grasp generation method requires a point cloud as input, we first estimate a depth map (Fig. 2.1) and reconstruct a point cloud. Then, given the grasp candidates in the scene, the most suitable one is selected according to the approach-to-grasp trajectory performed by the user. To this end, during the approach, the visual odometry module continuously processes the RGB images from the eye-in-hand camera and estimates the camera trajectory in space (Fig. 2.3). Finally, the closest grasp candidate to the camera is selected and executed on the Hannes prosthesis (Fig. 2.4).

B. Architecture Details

*1) **Monocular Depth Estimation**:* We rely on a model with a DPT [25] head and a DINOv2 [9] backbone for monocular depth estimation. Given the first RGB frame of the grasping sequence $\mathbf{I}_0 \in \mathbb{R}^{H\times W}$, we estimate a depth-map $\mathbf{D}_0 \in \mathbb{R}^{H\times W}$, where each value $d_{u,v}, u \in [0,...,H-1], v \in [0,...,W-1]$ is the depth in absolute scale (i.e., *metric* depth) estimated for the corresponding pixel in $\mathbf{I}_0$.

*2) **Grasp Generation**:* Given $\mathbf{I}_0$, $\mathbf{D}_0$ and the intrinsic camera parameters K, we build our point-cloud $\mathbf{PCD}_0$ by computing the 3D coordinates (x,y,z) for each pixel $(u,v) \in \mathbf{I}_0$ in the camera coordinate system. We use Contact-GraspNet [16] to process the resulting point cloud. Contact-GraspNet associates each gripper pose to a visible contact point $\mathbf{c}$, reducing the complexity of the learning problem. More specifically, the $\mathbb{SE}(3)$ position of a gripper is parameterized using two orthonormal vectors, the *approach* vector $\mathbf{a} \in \mathbb{R}^3$ and the *grasp baseline* vector $\mathbf{b} \in \mathbb{R}^3$, intersecting on the contact point $\mathbf{c}$. Furthermore, to reduce the memory requirements, the point cloud is first downsampled to $n = 20k$ points. In order to increase the number of grasps generated on object contact points, Contact-GraspNet can rely on object segmentation masks and sequentially process different point-cloud segments. We decided to not rely on additional object segmentation models to avoid introducing biases to known objects. We also experimented with the same unknown object segmentation model [26] used in [16] and observed lower performance in our scenario. We modify the Contact-GraspNet downsampling strategy in order to assign a higher sampling probability to closer points to the *eye-in-hand* camera, usually corresponding to contact points on the target object, and lower probabilities to farther points, commonly associated to background (e.g., a wall). For doing that, we sample pixels from a distribution having the probability scores $p_{u,v} = softmax(1/d_{u,v}),\ \forall (u,v) \in \mathbf{I}_0$, where $d_{u,v}$ is the estimated depth for pixel at location (u,v) and the softmax is computed over all the pixels.

*3) **Visual Odometry**:* We use DPVO [24] to estimate the position and orientation of the Hannes hand, using the sequence of RGB frames collected from the eye-in-hand camera while approaching the target object. DPVO performs sparse patch-tracking to build 2D-2D correspondences between consecutive frames. Each patch $\mathbf{P}_k$ is represented as the homogeneous array $\mathbf{P}_k = (\mathbf{u}_k, \mathbf{v}_k, \mathbf{1}, \mathbf{d}_k),\ \mathbf{u}_k, \mathbf{v}_k, \mathbf{d}_k \in \mathbb{R}^{1\times p^2}$, containing the pixel coordinates and the depth of each pixel in a $p \times p$ patch. At each new incoming RGB frame, DPVO estimates the 2D motion of tracked patches and then solves for both camera poses $\mathbf{T}_j \in \mathbb{SE}(3)$ and patch representations $\mathbf{P}_k \in \mathbb{R}^{4\times p^2}$. Because DPVO works with RGB frames, the optimized camera poses $\mathbf{T}_j = [\mathbf{R}_j|\mathbf{t}_j]$ will have up-to-scale translation components. We compute a scaling factor by comparing the optimized patch representations $\mathbf{P}_k$ having the initial frame (i.e., $\mathbf{I}_0$) as *source* frame with the *absolute*-scale dense depth estimated by our monocular depth estimation model, $\mathbf{D}_0$. Specifically, as DPVO assumes the same depth value for every pixel in a patch, we consider the center pixel of each patch. Then, we sample from the dense depth map at the same coordinates, and compute the *median* ratio between the two depth values for each sampled pixel. Formally, we compute the new camera poses $\mathbf{T}^*_j = [\mathbf{R}_j|\alpha^*\mathbf{t}_j]$, with $\alpha^* = \psi(\mathbf{D}_0, \{\mathbf{P}_{k|\mathbf{I}_0}\}_k)$, where ψ is the operator that takes the center coordinates u_c, v_c of each patch $\mathbf{P}_k$, samples $\mathbf{D}_0$ at the same coordinates and computes the *median* of the ratios $\{d^{\mathbf{D}_0}_{u_c,v_c}/d^{\mathbf{P}_k}_{u_c,v_c}\}_{k|\mathbf{I}_0}$ ($k|\mathbf{I}_0$ means "patch k sampled from $\mathbf{I}_0$"). We use the last estimated camera position to select the

nearest grasp pose. We compute a euclidean distance between the camera position and the middle point of each gripper. We select as the candidate grasp pose the one having the shortest distance to the camera, and thus, to the Hannes hand. Finally, if this distance is below a given threshold (e.g., 5cm), DPVO stops running and the candidate grasp is automatically executed on the Hannes prosthesis.

*4) **Mapping grasp candidates to the Hannes hand**:* A Hannes pre-shape configuration is defined by the triplet $(\omega_{ps}, \omega_{fe}, \lambda_f)$, where λ_f is the Hannes opening in the fingers *opening-closing* (FOC) range, and $(\omega_{ps}, \omega_{fe})$ are, respectively the wrist *pronation-supination* (WPS) and *flexion-extension* (WFE) angles. Finding λ_f from gripper parameters is straightforward, as it only requires to scale the gripper width w_g to the Hannes FOC range. We also define the optimal $(\omega_{ps}, \omega_{fe})$ to be the joint angles that make the eye-in-hand camera optical axis match the gripper approaching direction $\vec{\mathbf{a}}$, while the direction $\mathbf{f}$ in which the Hannes fingers close (in which the other four fingers close to reach the thumb) matches the gripper baseline vector $\vec{\mathbf{b}}$. We first apply a rotation of $\gamma = -\pi/2$ around the z-axis and $\beta = -\pi/4$ around y-axis to the gripper pose, such that a gripper with identity pose resembles the Hannes hand in the *home* position (Fig. 2.4). Thus, if $\mathbf{R}_g \in \mathbb{SO}(3)$ is the rotation component of the *candidate* grasp pose, we computed the *desired* camera pose to perform the grasp as $\mathbf{R}_c^{des} = (\mathbf{R}_c^{N-1})^T \mathbf{R}_g \mathbf{R}_z(\gamma) \mathbf{R}_y(\beta)$, where $\mathbf{R}_c^{N-1}$ embeds the rotation of the last estimated camera pose (i.e., at frame $N-1$, with N the number of processed frames). The transpose of $\mathbf{R}_c^{N-1}$ is pre-multiplied to project a gripper pose into the new reference frame of the camera. Then, we minimize the $\mathbb{SO}(3)$ error $\mathbf{e} = (\theta r_x, \theta r_z)^T$, where $\theta\mathbf{r} = axisangle(\mathbf{R}_c^{des})$. We project this error to the joint space, using the Jacobian in the end-effector frame and the control law $\dot{\mathbf{q}} = \lambda \left({}^c\mathbf{J}_e[\dot{\theta}_x, \dot{\theta}_z](\mathbf{q}) \right)^{\dagger} \mathbf{e}$, where ${}^c\mathbf{J}_e(\mathbf{q}) = {}^c\mathbf{Ad}_e \, {}^e\mathbf{J}_e(\mathbf{q})$, ${}^c\mathbf{Ad}_e$ is the adjoint matrix that converts Jacobian velocities expressed in the end-effector frame to the camera frame, ${}^c\mathbf{J}_e[\dot{\theta}_x, \dot{\theta}_z]$ is the $\mathbb{R}^{2\times 2}$ matrix built by extracting from the Jacobian the rows corresponding to the angular velocities around x and z-axis, $(\cdot)^{\dagger}$ is the Moore-Penrose pseudo-inverse of a matrix, and $\mathbf{q} \in \mathbb{R}^2$ is the vector encoding the Hannes wrist joint angles, $\mathbf{q} = (\omega_{ps}, \omega_{fe})^T$. We run this optimization until the norm of the error $\mathbf{e}$ is smaller than an error threshold, or for a fixed amount of steps (for constant time). We use the final $(\omega_{ps}, \omega_{fe})$ angles to control the Hannes wrist in the joint space.

IV. EXPERIMENTAL SETUP

A. Deployment and Embodiment

We tested and deployed our grasping pipeline on our Hannes prosthesis. A small eye-in-hand RGB camera is embedded into the prosthesis palm. User input is handled using two EMG sensors, placed on the forearm flexor and extensor muscles. We use a pre-defined and easily identifiable EMG signal to let the user trigger the start of the *approaching* stage. All our experiments were conducted by using a flexor and extensor *co-contraction* as the triggering starting signal. When triggering the signal, the target object should be visible on the eye-in-hand camera in order for grasp candidates to be generated on that object. At this point, the user can approach the target object. The end of the approaching stage, and thus the start of the *grasping* stage, is automatically detected based on distance between the estimated camera and nearest gripper position. At the start of the *grasping* stage, we control the wrist position (*open-loop* control) by specifying joint angles $(\omega_{ps}, \omega_{fe})$. After a few seconds ($t_{grasp}$), we also send the λ_f command to the fingers motor. We set $t_{grasp} = 2s$, after observing that this time is enough to let the user wrap the hand around the object, before the fingers are automatically closed for performing the grasp.

B. Subjects, Goal and Target Objects

We performed a first validation of our system on 10 able-bodied subjects, measuring a grasp success rate (GSR) and the average time to grasp (ATG). After validating the approach, we tested it on 3 amputee subjects. Moreover, for amputees, we also compared our method with a control based solely on EMG sensors using the SSC paradigm (now referred to as EMG-SSC). Evaluating this baseline was possible only with amputee subjects, because already familiar with similar control strategies. Performing the same trials with able-bodied subjects was impractical, as it turned out to be too complex for people that had no prior experience with the embodiment. We used 5 objects to conduct our experiments. The 3D model of three objects was included in synthetic data used to train our depth estimation model (*known* objects). The other two, or similar objects, were not included (*unknown* objects). The other components of our pipeline are object-agnostic. Objects are shown in Fig. 3 and Fig. 4. Every user (able-bodied or amputee) was asked to perform 6 grasps for each object, for a total of 30 trials for each subject. Amputees also performed the same number of trials with EMG-SSC. Finally, we conducted an experimental study on *cognitive load*, using the pupil dilation as a fatigue measure to further compare our method with EMG-SSC. The study adhered to the standard of the Declaration of Helsinki and was approved by the CET - Liguria ethical committee (Protocol code: IIT_REHAB_HT01). We refer to the supplementary video for demonstrations on both amputee and able-bodied subjects.

C. Hardware

We run our modules on a single NVIDIA RTX 3080 GPU. On this GPU, the pipeline can run at 30 Hz during the approaching stage. To record pupil dilation during our trials with subjects, we used Tobii Pro Glasses 3, a wearable eye-tracking device able to record absolute measures of pupil diameter (at 100 Hz) and other gaze measurements.

V. RESULTS

A. Ablation Study on Monocular Depth Estimation

To learn a monocular depth estimator, we use synthetic data to benefit from a significant amount of ground-truth depth values. We devised a synthetic data generation pipeline using the Unity engine and the Perception package [27].

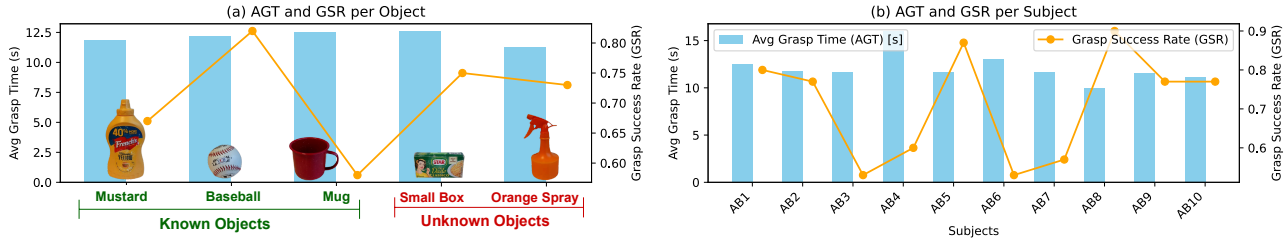

Fig. 3: AGT and GSR measured on able-bodied subjects. Statistics are shown per object (a) and per subject (b).

Our pipeline is based on [6] (*SynV1*), but we extended it to specifically increase the robustness of a depth estimation model. We propose several offline data augmentations (*SynV2*). We simulate scenarios with clutter and occlusions by randomizing the number of objects in the scene. We randomize the camera optical axis direction to avoid biases to fixed object positions. Finally, we randomize light direction and intensity to perform renderings under variable light conditions. To evaluate our depth estimation model, we use the same test data from [6], collected using an Intel RealSense D435 depth camera. We compare two different backbone checkpoints (DINOv2 [9] and DepthAnything [10]). We also experimented with two initialization strategies for the decoder head: random initialization and a checkpoint trained on the NYU-DepthV2 [28] depth dataset. The encoder features were frozen (*) in some of our experiments. When fine-tuning the encoder features, we use the same recipe of DepthAnything. Results shown in Table I demonstrate that our *SynV2* synthetic dataset is more effective for depth estimation and that random initialization of the decoder, with a fine-tuned encoder, produces the best results.

TABLE I: MDE evaluation on real test-set [6]

Dataset	Encoder	Decoder	[29] $\delta_1 \uparrow$	RMSE $\downarrow$
SynV2	DepthAnything	Random	0.689	0.115
SynV2	DINOv2	Random	**0.698**	**0.113**
SynV2	DINOv2 (*)	NYUv2	0.658	0.166
SynV1	DINOv2 (*)	NYUv2	0.412	0.476

B. Experiments with Able-bodied Subjects

Results for grasp success rate (GSR) and average grasp time (AGT) obtained on our 10 able-bodied subjects are shown in Fig. 3. We report an analysis of GSR and AGT found on different objects (Fig. 3a) and achieved by all the able-bodied subjects (Fig. 3b). We can already extract useful information from this validation on able-bodied subjects. First, we can notice how our method does not exhibit generalization issues to unknown objects. We can also observe how the *Mug* object was the most difficult object to grasp, achieving a GSR of 0.58. Specifically, we were expecting a gripper to be predicted on the mug handle, or on the mug borders, such that the antipodal contact-points are placed, respectively, on the external and internal surface of the object. Instead, during our trials, most of the grippers were predicted with both contact-points on the external surface, with a large gripper aperture. However, even when fully-opened, our Hannes hand is not able to grasp the *Mug* object this way, resulting in a failed attempt. Finally, we have evidence of a variable GSR on different able-bodied subjects (0.71 ± 0.14), with AB3 and AB6 achieving only 0.53 GSR, while AB5 and AB8 achieved 0.87 and 0.90 GSR, respectively. We believe a variable performance can be explained with a different subject response to the initial training stage or to muscle fatigue (e.g., due to holding the prosthesis for long periods of time). The AGT (12.09 ± 2.14 over *all* the trials) does not depend significantly on the object, as expected. Instead, subjects can approach the objects at different speeds or become familiar more quickly with the embodiment (especially with the EMG user-input interface).

C. Experiments with Amputees

We now report results and insights obtained from our analysis on our actual target users. Results for GSR and AGT are shown in Fig. 4. We also report a comparison with results obtained for EMG-SSC baselines. We show a comparison summary in Table II. All our invited amputee subjects had already experience with the use of EMG sensors for prosthesis control. One of them (Amputee 2) also had experience with the same multi-DoF control paradigm used for our baselines. The level of prior experience and skill is indicated in Fig. 4 through the use of stars. We experimented with a EMG-SSC baseline that uses a muscle co-contraction to trigger the joint switch (CC-SSC) and another EMG-SSC baseline in which the joint switch is operated through a button (B-SSC), held by the user using the other free hand. We decided to use B-SSC to facilitate the continuation of the experiments, when we noticed that the amputees struggled to continuously use a co-contraction as a switch method, while having to use EMGs to control the joints of the prosthesis. Amputee 1 performed the first 40% of the trials (12/30) with EMG-SSC. The remaining were performed using B-SSC. Amputee 2 performed all the trials with CC-SSC, and Amputee 3 performed all of them with B-SSC. Objects were picked in a random order. To avoid biasing our results, we never asked to grasp the same object for more consecutive trials for the EMG-SSC baselines. We were able to observe higher GSR results with amputee subjects, compared to able-bodied ones. This can be easily explained with a better response to fatigue over trials and with prior experience with the prosthetic embodiment. For EMG-SSC baselines, we expected nearly perfect GSR results with a trade-off on AGT. Interestingly, we measured a perfect GSR score with B-SSC on every object except the *Mug*, confirming the

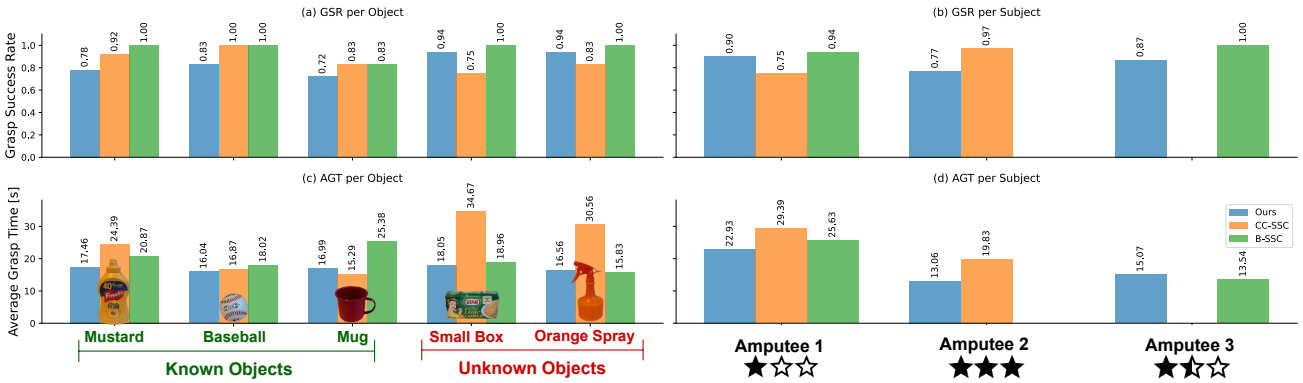

Fig. 4: AGT and GSR measured on amputees. Results with our method are compared with CC-SSC and B-SSC baselines.

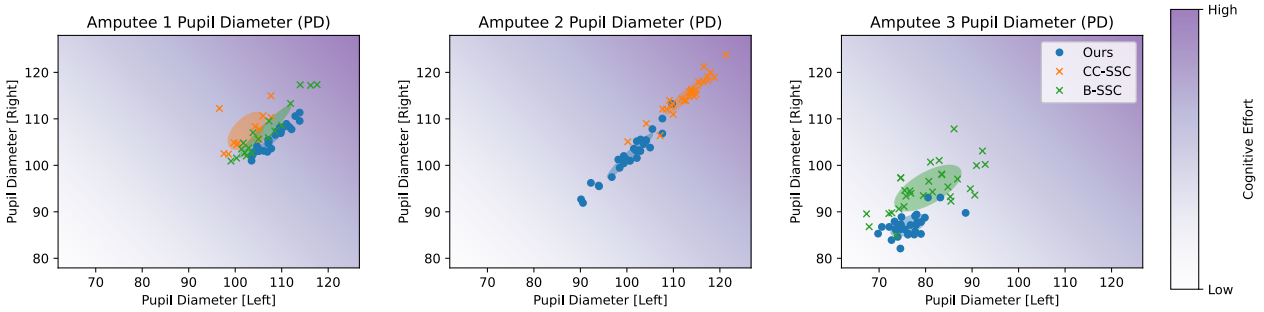

Fig. 5: Distributions of pupil diameter (PD) recorded while performing trials.

TABLE II: Summary table of results obtained on amputees

Method	↑ GSR	↓ AGT [s]
Ours	0.84 ± 0.12	**17.02 ± 4.75**
CC-SSC	0.87 ± 0.19	24.36 ± 15.07
B-SSC	*0.97 ± 0.15*	19.81 ± 8.65

intrinsic difficulty of grasping the object with our device. We also expected lower AGT times with B-SSC compared to CC-SSC, as the control is simplified at the cost of a more unhandy embodiment. We found that our method has, on average, lower AGT compared to both CC-SSC and B-SSC (Table II). Notably, this is also true for a user who had already experience with the control baseline (Amputee 2), but no experience with our method (Fig. 4d). This demonstrates that the proposed strategy enables faster grasps, while preserving the overall success rate.

D. Analysis on Cognitive Load

We conducted an analysis on cognitive load using eye-tracking data from Tobii Pro Glasses 3. Every amputee subject performed all the trials while wearing the eye-tracking glasses. For every subject, we also recorded a *baseline* experiment, in which users were asked to grasp objects with their other real hand (from now on, *Real-Hand*). Following prior works doing the same analysis in similar scenarios [30], [31], we relate the pupil diameter (PD) to the user's cognitive load. Indeed, it has been observed that a dilation of pupil diameter is related to an increasing cognitive effort [32], [33]. Similarly to [31], for every user, we express PD as the percentage of the diameter measured during *Real-Hand*. For our analysis, we only consider PD values on *fixations*, to only include variations in PD which are actually due to cognitive load and not other factors, such as changes in gaze directions. Notice that our goal is not to present a statistical exhaustive study on the topic, but rather to demonstrate the applicability of this analysis, compared to other more obtrusive setups to evaluate cognitive load. We show amputees' PD plots in Fig. 5. It is, indeed, interesting to observe how PD values are distributed for the three subjects. For Amputee 1, we do not observe significant differences in PD between the distributions. However, for Amputee 2 and 3, we observed, on average, an higher PD value while using a control baselines, compared to our method. Following these observations, we can hypothesize that our method was easier and more intuitive to use for Amputee 2 and 3, resulting in a lower measured cognitive effort.

VI. CONCLUSIONS

In this work, we introduced a novel vision-based prosthetic grasping pipeline, based on the shared-autonomy principles to enhance user-prosthesis interaction. Our system leverages components from the robotic literature and showcases how these can be applied in a prosthetic scenario. We demonstrated the effectiveness of our framework by testing it on amputee subjects and comparing it with a standard multi-DoF control paradigm. Finally, we performed an experimental analysis on the mental workload, demonstrating the potential of our method to reduce the cognitive burden on the user.

REFERENCES

[1] A. Marinelli, N. Boccardo, F. Tessari, D. Di Domenico, G. Caserta, M. Canepa, G. Gini, G. Barresi, M. Laffranchi, L. De Michieli, *et al.*, "Active upper limb prostheses: A review on current state and upcoming breakthroughs," *Progress in Biomedical Engineering*, vol. 5, no. 1, p. 012001, 2023.

[2] S. Amsuess, P. Goebel, B. Graimann, and D. Farina, "Extending mode switching to multiple degrees of freedom in hand prosthesis control is not efficient," in *IEEE Eng. Med. Biol. Soc.*, 2014, pp. 658–661.

[3] L. Seminara, S. Dosen, F. Mastrogiovanni, M. Bianchi, S. Watt, P. Beckerle, T. Nanayakkara, K. Drewing, A. Moscatelli, R. L. Klatzky, *et al.*, "A hierarchical sensorimotor control framework for human-in-the-loop robotic hands," *Science Robotics*, vol. 8, no. 78, p. eadd5434, 2023.

[4] J. Starke, P. Weiner, M. Crell, and T. Asfour, "Semi-autonomous control of prosthetic hands based on multimodal sensing, human grasp demonstration and user intention," *Robotics and Autonomous Systems*, vol. 154, p. 104123, 2022.

[5] M. N. Castro and S. Dosen, "Continuous semi-autonomous prosthesis control using a depth sensor on the hand," *Frontiers in Neurorobotics*, vol. 16, p. 814973, 2022.

[6] F. Vasile, E. Maiettini, G. Pasquale, A. Florio, N. Boccardo, and L. Natale, "Grasp pre-shape selection by synthetic training: Eye-in-hand shared control on the hannes prosthesis," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 13 112–13 119.

[7] F. Hundhausen, S. Hubschneider, and T. Asfour, "Grasping with humanoid hands based on in-hand vision and hardware-accelerated cnns," in *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*. IEEE, 2023, pp. 1–7.

[8] M. Laffranchi, N. Boccardo, S. Traverso, L. Lombardi, M. Canepa, A. Lince, M. Semprini, J. A. Saglia, A. Naceri, R. Sacchetti, *et al.*, "The hannes hand prosthesis replicates the key biological properties of the human hand," *Science robotics*, vol. 5, no. 46, p. eabb0467, 2020.

[9] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. HAZIZA, F. Massa, A. El-Nouby, *et al.*, "Dinov2: Learning robust visual features without supervision," *Transactions on Machine Learning Research*, 2024.

[10] L. Yang, B. Kang, Z. Huang, X. Xu, J. Feng, and H. Zhao, "Depth anything: Unleashing the power of large-scale unlabeled data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 10 371–10 381.

[11] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun, "Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 3, pp. 1623–1637, 2020.

[12] X. Shi, W. Xu, W. Guo, and X. Sheng, "Target prediction and temporal localization of grasping action for vision-assisted prosthetic hand," in *2022 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2022, pp. 285–290.

[13] N. Boccardo, M. Canepa, S. Stedman, L. Lombardi, A. Marinelli, D. Di Domenico, R. Galviati, E. Gruppioni, L. De Michieli, and M. Laffranchi, "Development of a 2-dofs actuated wrist for enhancing the dexterity of myoelectric hands," *IEEE Transactions on Medical Robotics and Bionics*, 2023.

[14] A. Mousavian, C. Eppner, and D. Fox, "6-dof graspnet: Variational grasp generation for object manipulation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 2901–2910.

[15] A. Murali, A. Mousavian, C. Eppner, C. Paxton, and D. Fox, "6-dof grasping for target-driven object manipulation in clutter," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 6232–6238.

[16] M. Sundermeyer, A. Mousavian, R. Triebel, and D. Fox, "Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 13 438–13 444.

[17] A. Alliegro, M. Rudorfer, F. Frattin, A. Leonardis, and T. Tommasi, "End-to-end learning to grasp via sampling from object point clouds," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9865–9872, 2022.

[18] L. Shao, F. Ferreira, M. Jorda, V. Nambiar, J. Luo, E. Solowjow, J. A. Ojea, O. Khatib, and J. Bohg, "Unigrasp: Learning a unified model to grasp with multifingered robotic hands," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2286–2293, 2020.

[19] K. Karunratanakul, J. Yang, Y. Zhang, M. J. Black, K. Muandet, and S. Tang, "Grasping field: Learning implicit representations for human grasps," in *2020 International Conference on 3D Vision (3DV)*, 2020, pp. 333–344.

[20] K. Li, N. Baron, X. Zhang, and N. Rojas, "Efficientgrasp: A unified data-efficient learning to grasp method for multi-fingered robot hands," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 8619–8626, 2022.

[21] P. Li, T. Liu, Y. Li, Y. Geng, Y. Zhu, Y. Yang, and S. Huang, "Gendexgrasp: Generalizable dexterous grasping," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 8068–8074.

[22] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[23] Z. Teed and J. Deng, "Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras," *Advances in neural information processing systems*, vol. 34, pp. 16 558–16 569, 2021.

[24] Z. Teed, L. Lipson, and J. Deng, "Deep patch visual odometry," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[25] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 12 179–12 188.

[26] C. Xie, Y. Xiang, A. Mousavian, and D. Fox, "Unseen object instance segmentation for robotic environments," *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1343–1359, 2021.

[27] Unity Technologies, "Unity Perception package," https://github.com/Unity-Technologies/com.unity.perception, 2020.

[28] P. K. Nathan Silberman, Derek Hoiem and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *ECCV*, 2012.

[29] L. Ladicky, J. Shi, and M. Pollefeys, "Pulling things out of perspective," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[30] K. Y. Cheng, M. Rehani, and J. S. Hebert, "A scoping review of eye tracking metrics used to assess visuomotor behaviours of upper limb prosthesis users," *Journal of NeuroEngineering and Rehabilitation*, vol. 20, no. 1, p. 49, 2023.

[31] S. Manz, T. Schmalz, M. Ernst, T. M. Köhler, J. Gonzalez-Vargas, and S. Dosen, "Using mobile eye tracking to measure cognitive load through gaze behavior during walking in lower limb prosthesis users: A preliminary assessment," *Clinical Biomechanics*, vol. 115, p. 106250, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0268003324000822

[32] P. Van der Wel and H. Van Steenbergen, "Pupil dilation as an index of effort in cognitive control tasks: A review," *Psychonomic bulletin & review*, vol. 25, pp. 2005–2015, 2018.

[33] K. Walter and P. Bex, "Cognitive load influences oculomotor behavior in natural scenes," *Scientific Reports*, vol. 11, no. 1, p. 12405, June 2021, publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41598-021-91845-5