# DITTOHAND: ZERO-SHOT GRIP-FORCE CONTROL ACROSS HAND MORPHOLOGIES WITH SURFACE-AWARE ACTUATOR TOKENS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

To a typical cross-embodiment policy, a hand is its kinematic structure; two hands that differ only in finger thickness are the same hand. Yet the same actuator command meets the object earlier on the thicker finger, at a different point and along a different normal. Existing cross-hand approaches either leave the hand–object geometry implicit or recover it with a retargeting step trained per hand. A policy that leaves this collision geometry implicit can often find a grasp, but it keeps that grasp less reliably and barely tracks the commanded grip force. We propose **DittoHand**, which treats a hand as a set of actuators and gives each an actuator token carrying the collision surface it moves and the closing rate at which that surface approaches the object as the actuator turns, so an action means the same thing on every hand. Trained on 24 hands, one policy is tested on them and zero-shot on 648 unseen hands, with 20 objects of which 17 are unseen in training. On both hand sets it leads two cross-hand baselines, most widely in keeping the object (47.5% vs. at most 34.3% on unseen hands) and in tracking the grip command, where it beats each baseline on at least 99.8% of the unseen hands. In a virtual reality (VR) study on three hands DittoHand had never seen, 12 participants rated it significantly more controllable and realistic overall than ForceGrip policies trained for each hand.

## 1 INTRODUCTION

To a typical cross-embodiment policy, a hand is its kinematic structure: actuators, phalanges, and axes (Patel & Song, 2025; Wu et al., 2026). Two hands that differ only in finger thickness are therefore the same hand. They are not. The same actuator command now meets the object earlier, at a different point, along a different normal. When the goal is only to hold on, a policy can hide this by squeezing harder; when the user specifies how hard to squeeze, it cannot, because a stable grasp at the commanded grip force depends on exactly where and how the surface meets the object. Interactive virtual reality (VR) is this setting, and no existing policy serves it without training per hand: the wrist follows the controller (Han et al., 2024), a trigger sets the commanded grip force (Han et al., 2025), and the hand is whatever avatar the user chose or edited moments ago, differing in kinematic structure and collision geometry alike and unseen in training. Users expect that hand to work at once, so retraining per hand is not an option; we seek one policy that transfers zero-shot to unseen hands and makes the same command mean the same grip on each (Figure 1).

A representation for this must answer two questions for every hand: which surface each actuator moves, and how moving it changes the contact with the object. Existing representations answer at most one. Morphology graphs (Patel & Song, 2025; Wu et al., 2026) describe the actuators but leave the surfaces they move implicit, so how a thicker phalanx changes contact is left for the policy to infer. Hand–object surface representations (She et al., 2024) describe the contact geometry but not which actuator alters it, and so still need hand-specific adaptation. Acting in a human action space (Yuan et al., 2025) sidesteps both questions but needs a retargeting network trained per hand.

We answer both by making the actuator the unit of representation. Fingers and phalanges vary in number, but every hand is a set of actuators, and every actuator does the same kind of thing: it moves a patch of surface toward or away from the object. That event looks the same on any hand, so each



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

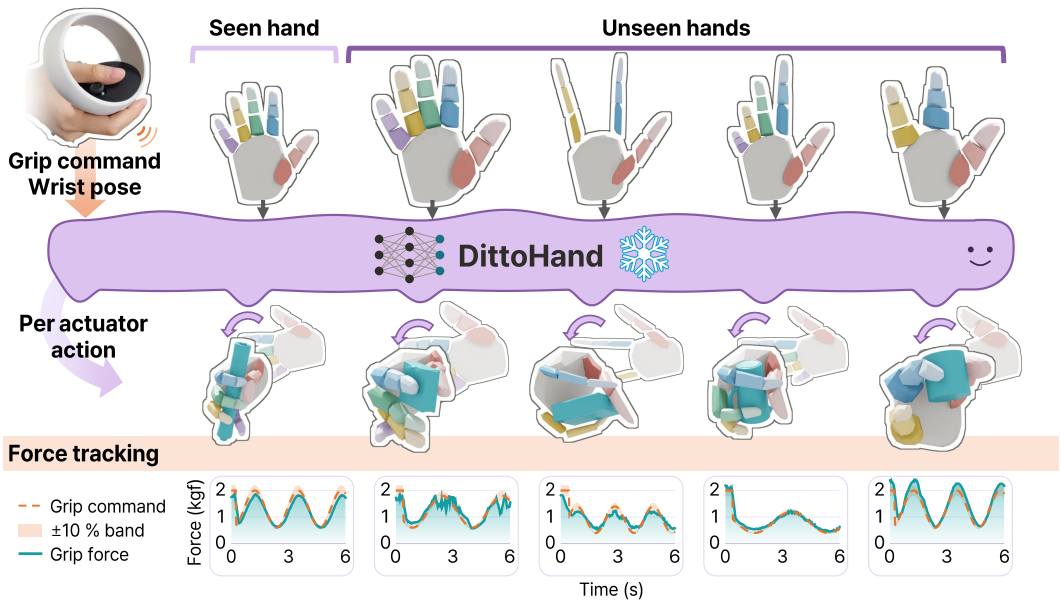


Figure 1: DittoHand. One policy drives hands that differ in shape, fingers, phalanges, thickness, and length (top), acting per actuator (middle) to keep the grip force near the command (bottom).

actuator becomes one token describing it. The token carries the actuator's kinematics and state, the surface it moves, where that surface currently sits relative to the object, and how fast turning the actuator would close that distance. This last quantity, the closing rate, tells the policy what its action does: how fast it closes the gap before contact and how hard it presses after, so an action means the same thing on hands with different joint axes and finger lengths. A Transformer (Vaswani et al., 2017), aware of which actuators are connected, turns these tokens into per-actuator actions. We call the result DittoHand, one policy that takes the form of whatever hand it is given.

We train one policy on 24 MANO hands (Romero et al., 2017) and test it on them and on 648 unseen hands, with 20 objects. A cross-hand policy must learn many hands at once and work on new ones as they are; ours leads two cross-hand baselines at each. When fingers or phalanges are removed from the training hands, all three methods grasp about equally often, yet ours keeps the object far more often and tracks the grip command more closely. In a VR user study on three hands our policy had never seen, 12 participants rated it significantly more controllable and realistic overall than ForceGrip (Han et al., 2025) policies trained for each hand.

Our contributions are as follows:

- **Surface-aware actuator tokens.** Each actuator is described by the surface it moves and where that surface sits relative to the object, so one policy controls hands that differ in kinematic structure and collision geometry alike, with no retargeting or adaptation.
- **A closing rate for grip-force tracking.** Tracking a grip command requires knowing how each actuator's motion changes its contact; we compute this closing rate from each hand's own surface at every step, which sharpens tracking on seen and unseen hands.
- **Collision geometry as a necessary axis of embodiment.** One policy trained on 24 hands keeps objects and tracks the grip command better than cross-hand baselines on seen and unseen hands, an ablation shows that this lead comes from the surfaces in its tokens, and 12 VR users rated it more controllable and realistic than per-hand policies.

## 2 Related Work

### 2.1 Hand interaction in virtual reality

Contact-based VR hand control either tracks user finger poses under contact constraints (Borst & Indugula, 2005; Jacobs & Fröhlich, 2011; Verschoor et al., 2018), retargeting them across hand

sizes and joint placements (Sorli et al., 2021), or synthesizes finger motion from wrist pose and a grasp command: rule-based flexion (Oprea et al., 2019), closed-loop control learned by VR-HandNet (Han et al., 2024), and trigger-specified grip-force regulation by ForceGrip (Han et al., 2025). Physics-based grasp motion synthesis in graphics (Christen et al., 2022) also spans several hand models (Zhang et al., 2024), but targets a stable grasp rather than tracking a user's grip command online. With no finger pose to retarget, morphology transfer must happen inside the policy; VR-HandNet and ForceGrip are trained per hand, whereas ours transfers to unseen hands without retraining.

### 2.2 Sharing one policy across morphologies

A shared policy must map a common representation to each hand's actuators. CrossDex (Yuan et al., 2025) shares fingertip observations and a human eigengrasp action space but needs a retargeting network trained per hand. Part-level modules (Huang et al., 2020; Bohlinger et al., 2024) share an observation-to-action mapping across parts; we apply it per actuator. Parts are coordinated by message passing along the body graph (Wang et al., 2018; Huang et al., 2020) or by self-attention (Kurin et al., 2021), with learned positional embeddings in MetaMorph (Gupta et al., 2022) or a joint-graph-distance attention bias in GET-Zero (Patel & Song, 2025). The anatomically aligned node graphs of DexGrasp-Zero (Wu et al., 2026) likewise describe the hand by its kinematic structure. None of these represents the surface each actuator moves. We keep GET-Zero's distance-based attention bias and no positional embedding, but bind every actuator to that surface.

### 2.3 Geometric representations for grasping

Grasp synthesis builds a target pose from contact maps (Li et al., 2023), hand–object distance matrices (Wei et al., 2025), link–object transforms (Fei et al., 2026), or canonical hand parameters that size each link as a capsule (Wei et al., 2026); force tracking, however, must react to the object after contact, so geometry is needed as an online observation. DexRepNet (Liu et al., 2023; 2026) observes hand-centric occupancy and keypoint-to-object distances and normals, but fixed queries can miss changes in finger thickness. DexPoint (Qin et al., 2022) encodes hand and scene point clouds, and other policies (She et al., 2022; 2024) the interaction bisector surface (IBS) equidistant from hand and object, with She et al. (2024) mapping keypoint displacements to joints through a hand-specific adaptation model. These expose surface geometry but not which actuator changes it; we describe each actuator by the surface it moves, its relation to the object, and the closing rate.

## 3 DittoHand

We build hands from MANO (Romero et al., 2017), whose shape parameters $\boldsymbol{\beta}$ span a range of human hand shapes learned from scans. A virtual avatar hand need not lie within this range, so we also edit finger thickness, finger length, and the number of fingers and phalanges beyond it (Figure 1). Removing fingers or phalanges changes the kinematic structure: which actuators exist and how they are arranged. Thickness changes only the collision geometry: the surfaces those actuators move. $\boldsymbol{\beta}$ and length change both. The same actuator turn then meets the object sooner on a thicker finger, and the same torque presses less hard on a longer one. Our policy therefore sees each actuator together with the surface it moves.

### 3.1 Task and grip force

At each step the user sets the wrist pose, which the simulated hand follows, and a grip command $g_t \in [0, 1]$ that expresses a force intention as the commanded grip force $F_{\mathrm{cmd},t} = g_t F_{\max}$, with $F_{\max} = 2\,\mathrm{kgf}$ for every hand and object. The policy acts at 30 Hz and outputs one action per actuator, a position target relative to the current angle for a 120 Hz proportional–derivative (PD) drive, so in contact the action sets how hard the actuator presses. Its goal is to grasp stably enough for the user to pick up, move, and place the object while its grip force reproduces $F_{\mathrm{cmd},t}$.

We define the grip force $F_{\mathrm{grip}}$ as the total normal load of the internal contact forces, those that squeeze the object without pushing or turning it (Kerr & Roth, 1986; Murray et al., 1994). With $\mathbf{f}_i$ the net contact force that the palm or phalanx $i$ exerts on the object, $\mathbf{f}$ their stack, and $\mathbf{G}$ the grasp

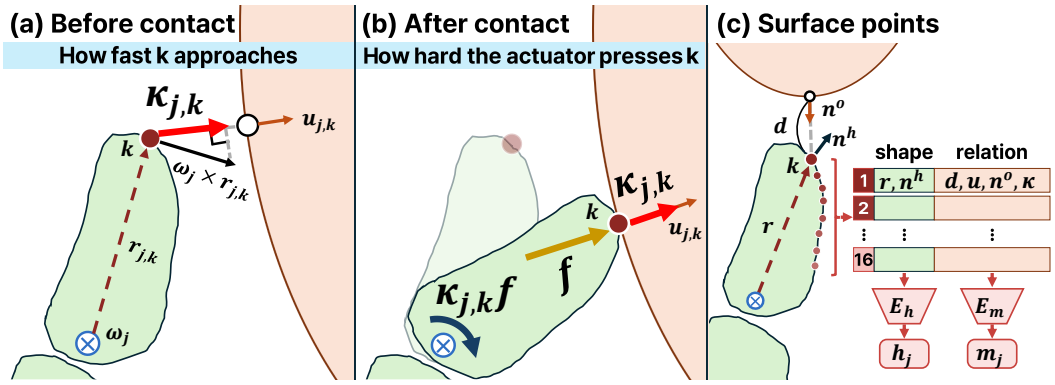


Figure 2: Surface points and closing rate. (a) Turning actuator $j$ about $\boldsymbol{\omega}_j$ moves surface point $k$ along $\boldsymbol{\omega}_j \times \mathbf{r}_{j,k}$; its component along $\mathbf{u}_{j,k}$, toward the nearest object point, is the closing rate $\kappa_{j,k}$. (b) After contact, pushing at $k$ with force $f$ along $\mathbf{u}_{j,k}$ takes a torque $\kappa_{j,k} f$. (c) $E_{\mathrm{h}}$ and $E_{\mathrm{m}}$ encode the shape ($\mathbf{r}$, $\mathbf{n}^h$) and relation ($d$, $\mathbf{u}$, $\mathbf{n}^o$, $\kappa$) of the 16 points on the contact side into $\mathbf{h}_j$ and $\mathbf{m}_j$.

map from $\mathbf{f}$ to the object's net wrench (Murray et al., 1994), the internal part is $\mathbf{f}_{\mathrm{int}} = (\mathbf{I} - \mathbf{G}^{+}\mathbf{G})\mathbf{f}$. $F_{\mathrm{grip}}$ sums its compressive normal components:

$$F_{\mathrm{grip}} = \sum_i \left[-\mathbf{n}_i^{\top} \mathbf{f}_{\mathrm{int},i}\right]_+, \tag{1}$$

with $\mathbf{n}_i$ the outward object normal where part $i$ touches. A single contact gives zero, since one contact cannot squeeze. The same $F_{\mathrm{grip}}$ is the command target, the reward, an observation, and every force metric.

### 3.2 Surface points and closing rate

$F_{\mathrm{grip}}$ reads the same on every hand, but how an actuator's action changes it does not: that depends on the surface the actuator moves and where it sits relative to the object. We therefore give each actuator that surface, as 16 points on the contact side of its phalanx, laid out so that each point refers to the same place on any finger. Point $k$ of actuator $j$ records its position $\mathbf{r}_{j,k}$ from the joint origin, its gap $d_{j,k}$ to the nearest object point and the unit direction $\mathbf{u}_{j,k}$ toward it, and the hand and object normals $\mathbf{n}^h_{j,k}$, $\mathbf{n}^o_{j,k}$ there, which tell whether the two surfaces face each other. All of these vectors are expressed in the coordinate frame of the joint of actuator $j$. Actuators on the same phalanx share its points.

Where a surface lies does not yet say what turning the actuator does to it. For actuator $j$ with axis $\boldsymbol{\omega}_j$ and angle $q_j$, the closing rate (Figure 2)

$$\kappa_{j,k} = \mathbf{u}_{j,k}^{\top}\left(\boldsymbol{\omega}_j \times \mathbf{r}_{j,k}\right) \tag{2}$$

answers this: it is the rate at which point $k$ approaches the object as $q_j$ increases, positive when turning closes the gap. It equals $-\partial d_{j,k}/\partial q_j$, since the shift of the nearest object point does not change the distance to first order (Danskin, 1967). Once point $k$ reaches the object, the same $\kappa_{j,k}$ tells how hard the actuator's torque presses there: pushing with force $f$ along $\mathbf{u}_{j,k}$ takes a torque $\kappa_{j,k} f$ (Murray et al., 1994). The kinematic structure alone does not determine $\kappa$, since the surface points belong to the collision geometry. A network that sees $\mathbf{r}_{j,k}$, $\mathbf{u}_{j,k}$, and $\boldsymbol{\omega}_j$ could learn $\kappa$, but only approximately, from the axes and surfaces it was trained on. We therefore compute $\kappa$ directly, like $d$ and $\mathbf{u}$, from each hand's own surface at every step. How quickly an actuator closes a gap then means the same on every hand.

### 3.3 Actuator tokens

We treat a hand as a set of actuators. Each actuator becomes one token, and one network with shared weights reads every token without joint indices (Figure 3), so a hand with fewer or differently placed actuators is simply a different set. Because the network sees an actuator only through its

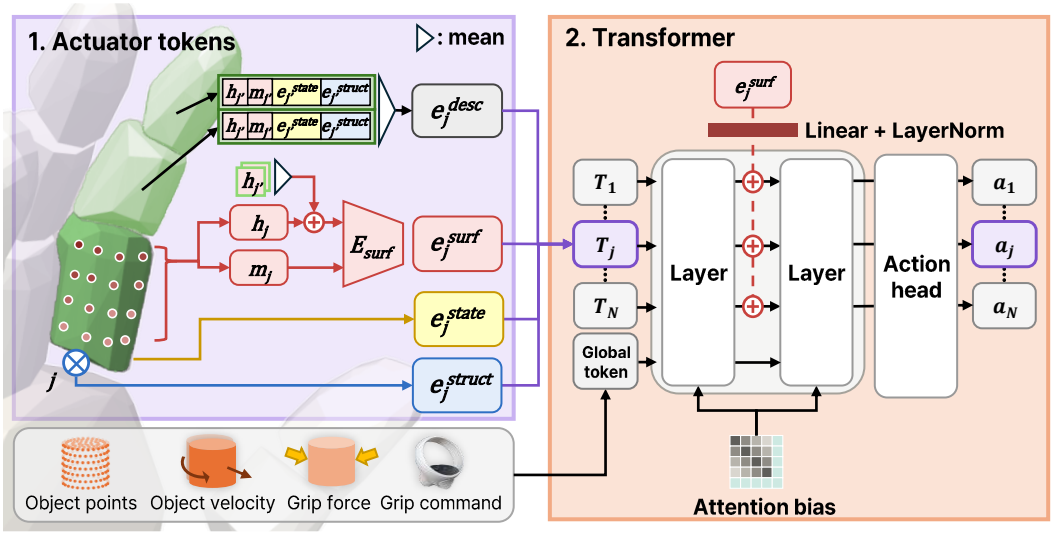


Figure 3: Policy architecture. Each actuator becomes one token from its state, structure, surface, and descendant embeddings. A Transformer with an attention bias processes the actuator tokens together with a global token, and a linear action head shared by all actuators maps each final token to its actuator's action.

token, the token must describe the actuator fully: its state, its structure, the surface it moves, and its descendants, the downstream phalanges and actuators that move with it.

Each of these four parts becomes an embedding from its own encoder, shared across actuators. The state embedding $\mathbf{e}_j^{\text{state}}$ says what the actuator is doing: its angle, velocity, previous action, and torque, and the motion and contact force of its phalanx. The structure embedding $\mathbf{e}_j^{\text{struct}}$ says what kind of actuator it is: its axis and angle limits, its rest-pose bone vectors, and its depth and distance from the wrist. The surface embedding says which surface the actuator moves, how that surface relates to the object, and how that relation changes as the actuator turns (Figure 2c). One encoder reads the shape of its phalanx, $\mathbf{h}_j = E_{\text{h}}(\{\mathbf{r}_{j,k}, \mathbf{n}^h_{j,k}\}_k)$. Another reads the relation from $d$, $\mathbf{u}$, and $\mathbf{n}^o$ and its change from $\kappa$, $\mathbf{m}_j = E_{\text{m}}(\{d_{j,k}, \mathbf{u}_{j,k}, \mathbf{n}^o_{j,k}, \kappa_{j,k}\}_k)$.

Each token keeps one actuator's own view, but turning the actuator also moves its descendants $\mathcal{D}(j)$, whose number differs across fingers and hands. So the shapes of the descendant phalanges join the surface embedding,

$$\mathbf{e}_j^{\text{surf}} = E_{\text{surf}}\Big(\mathbf{h}_j + \operatorname*{mean}_{j' \in \mathcal{D}(j)} \mathbf{h}_{j'},\ \mathbf{m}_j\Big). \tag{3}$$

The descendant embedding summarizes the state, structure, and surface of the actuators in $\mathcal{D}(j)$ (Huang et al., 2020),

$$\mathbf{e}_j^{\text{desc}} = \operatorname*{mean}_{j' \in \mathcal{D}(j)} E_{\text{desc}}\big(\mathbf{e}_{j'}^{\text{state}},\ \mathbf{e}_{j'}^{\text{struct}},\ \mathbf{h}_{j'},\ \mathbf{m}_{j'}\big). \tag{4}$$

Because the descendants enter both embeddings as an average, the token keeps the same form whether a finger has two or three phalanges. The token is $\mathbf{T}_j = E_{\text{T}}\big(\mathbf{e}_j^{\text{state}},\ \mathbf{e}_j^{\text{struct}},\ \mathbf{e}_j^{\text{surf}},\ \mathbf{e}_j^{\text{desc}}\big)$.

### 3.4 Whole-hand coordination and action

Each token describes one actuator and its descendants, but a grip comes from fingers pressing the object from opposite sides, so all actuators must act together. A Transformer (Vaswani et al., 2017) lets all tokens exchange information, and a global token adds what the actuator tokens do not carry. It holds the last six grip commands, the current $F_{\text{grip}}$, and its difference from $F_{\text{cmd}}$. It also holds the object's linear and angular velocity relative to the wrist, the gravity direction, and a PointNet (Qi et al., 2017) encoding of the object's points in the palm frame, which places the object relative to the palm.

The descendant terms cover only what moves with each actuator. To coordinate the whole hand, attention must also know how any two actuators are linked, including across fingers. We therefore add a learned attention bias (Ying et al., 2021; Patel & Song, 2025): each attention head adds a learned scalar to the attention score of a token pair according to how closely the two actuators are linked, from neighbors on one finger to actuators on different fingers. The bias depends on how two actuators relate rather than on which joints they are, so the same bias serves any hand.

The surface embedding $\mathbf{e}_j^{\mathrm{surf}}$ is re-added to its token between the Transformer layers through a linear layer and LayerNorm (Bai et al., 2019; Chen et al., 2020; Wu et al., 2026) so that attention does not dilute it. A linear action head shared by all actuators turns each final token into that actuator's action. Hands with fewer actuators are padded to a common number of tokens, and the padding is masked out of attention, the descendant means, and the action distribution. Appendix A gives the exact features, dimensions, and network.

### 3.5 Training

Training varies only $\boldsymbol{\beta}$ and the number of fingers, leaving the other edits for evaluation. Latin hypercube sampling (McKay et al., 1979) spreads the $\boldsymbol{\beta}$ of 24 hands evenly over $[-2, 2]^{10}$, and half lack one non-thumb finger. The objects are a cube, cylinder, and capsule, all 4 cm wide.

The reward $r = r_{\mathrm{force}} + r_{\mathrm{prox}} + 0.2\, r_{\mathrm{opp}}$ builds on ForceGrip (Han et al., 2025), with forces in kgf:

$$r_{\mathrm{force}} = \exp\bigl(-(F_{\mathrm{grip}} - F_{\mathrm{cmd}})^2\bigr). \tag{5}$$

Before contact $r_{\mathrm{force}}$ cannot guide the hand, so the proximity term $r_{\mathrm{prox}}$ rewards the fingers for being close to the object with their contact sides facing it. A single contact still gives $F_{\mathrm{grip}} = 0$, so the opposition term $r_{\mathrm{opp}}$ rewards the thumb and another finger for touching at once, not for pressing harder. Appendix B.4 defines both terms.

A random object placement often leaves the hand no reachable force-closure grasp. Each episode therefore starts with the fingers at a default pose and the object at a pose relative to the wrist, drawn from candidates that the grasp generator FSG (Wu et al., 2025) precomputes for each hand–object pair. The object is held fixed above a support for the first 0.5 s of the 3 s episode while the fingers close, and is then released under gravity. The episode ends early if the object surface is more than 7 cm from the palm, where the hand cannot recover it.

To learn a stable grasp and force tracking at once, we extend ForceGrip's curriculum (Han et al., 2025) to four stages that all environments enter together at fixed iterations. Stage 1 keeps the command constant per episode and the support in place. Stages 2–4 add in turn a time-varying command, support removal at the release, and random wrist motion. Training our policy with proximal policy optimization (PPO) (Schulman et al., 2017) takes about 4.2 days on one NVIDIA RTX 6000 Ada GPU. Appendix B gives the placement, curriculum, and PPO settings.

## 4 Experiments

We first test on seen and unseen hands whether a policy meets both requirements of the goal in Section 3.1: a stable grasp and a grip force that reproduces the commanded grip force (Section 4.3). Ablations then show what the token's surface and descendant information contributes (Section 4.4), and a VR user study asks users how controllable and realistic the policy feels (Section 4.5).

### 4.1 Evaluation protocol

**Seen hands** are the 24 hands used in training. **Unseen hands** are 648 hands that start from 36 new $\boldsymbol{\beta}$ (12 each with five, four, or three fingers), each built with two or three phalanges on every non-thumb finger, finger thickness scaled by $\{0.85, 1.0, 1.15\}$, and finger length scaled by $\{1.0, 1.1, 1.2\}$. To test one change at a time, we also report the 24 Unseen hands whose only change is a new $\boldsymbol{\beta}$ and 192 hands that each apply one edit to a Seen hand, with thickness and length changed beyond the levels of the Unseen hands (Figure 6). Each hand is tested on 20 objects in (Figure 7) : the three training objects, a sphere, and 16 YCB-Video objects (Xiang et al., 2018).

Table 1: Comparison with the baselines. Indented rows differ from the Seen hands in one aspect (Section 4.1). Topology means fewer fingers or phalanges. Grasp and Success come from the transport experiment, FTE and Gain from the force experiment. Bold: best per row and metric.

| | Grasp (%) ↑ | | | Success (%) ↑ | | | FTE (kgf) ↓ | | | Gain → 1 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Hands | DGZ | IBS | Ours | DGZ | IBS | Ours | DGZ | IBS | Ours | DGZ | IBS | Ours |
| Seen hands (24) | 57.8 | 63.9 | **74.6** | 54.8 | 41.0 | **66.6** | 0.67 | 0.93 | **0.35** | 0.28 | 0.00 | **0.65** |
| $\beta$ only (24) | 57.5 | 63.4 | **73.5** | 53.9 | 41.3 | **65.0** | 0.67 | 0.93 | **0.36** | 0.28 | −0.01 | **0.65** |
| Thickness only (96) | 54.9 | 61.3 | **71.2** | 45.4 | 35.1 | **58.4** | 0.75 | 1.04 | **0.47** | 0.27 | −0.01 | **0.64** |
| Length only (48) | 48.7 | 59.6 | **67.4** | 49.3 | 37.5 | **67.3** | 0.74 | 1.24 | **0.41** | 0.28 | −0.02 | **0.69** |
| Topology only (48) | 49.5 | **50.3** | 49.0 | 24.4 | 17.8 | **38.9** | 0.73 | 1.00 | **0.46** | 0.23 | 0.00 | **0.53** |
| **Unseen hands (648)** | 49.7 | 53.8 | **55.4** | 34.3 | 28.4 | **47.5** | 0.72 | 1.06 | **0.46** | 0.25 | 0.00 | **0.59** |

Each hand–object pair runs 10 episodes per experiment from FSG placements drawn at random as in training. Pairs without a feasible FSG grasp (234 of 17,280) are left out. The evaluation is zero-shot: no method is trained or tuned on any hand other than the Seen hands.

Two experiments test the two requirements one at a time. The transport experiment asks whether the hand keeps the object while the wrist carries it around. As in the fourth training stage, the support is removed at 0.5 s and the wrist then moves randomly. The command rises from 0 to 1 over the first 0.5 s and then stays at 1, so that holding does not depend on tracking a changing command. **Grasp** is the share of episodes in which the hand holds the object in a force-closure grasp when the support is removed, by the same test FSG uses to select start poses. **Success** is the share of episodes that end without early termination (Appendix B.4).

The force experiment asks whether $F_{\mathrm{grip}}$ reproduces $F_{\mathrm{cmd}}$ when nothing else changes: as in the second training stage, the support stays and the wrist is fixed. The command is a sinusoid clipped at 0 and 1, like a trigger fully released or pressed. **FTE** (force tracking error) is the time mean of $|F_{\mathrm{grip}} - F_{\mathrm{cmd}}|$ in kgf. **Gain** is the slope of $F_{\mathrm{grip}}$ regressed on $F_{\mathrm{cmd}}$ and is ideally 1. Both are computed per episode and then averaged.

### 4.2 Baselines and ablations

The baselines are cross-hand policies without the surface points and closing rate of our actuator tokens. DGZ-adapted brings DexGrasp-Zero (Wu et al., 2026) to our task: a graph network over the hand's anatomical parts that sees their kinematic structure, contacts, and offsets to the object. IBS-adapted keeps our architecture but replaces the surface points and the object's points with the IBS (She et al., 2022; 2024). Each phalanx's part of the IBS goes to that phalanx's actuators, and the whole IBS to the global token. Tables abbreviate the two baselines as DGZ and IBS. The ablations in Table 2 each remove one kind of information from the token. $-\kappa$ removes the closing rate, $-$ desc removes the descendant embedding and the descendant shapes in the surface embedding, and $-$ surf removes the surface embedding and the surface inputs to the descendant embedding. All methods share our environment, reward, and curriculum schedule. Appendix B details how the hands, objects, and commands are generated and how each baseline and ablation is built.

### 4.3 Zero-shot transfer

A cross-hand policy must learn many hands at once and then work on new hands as they are, and ours leads both baselines at each. On the 24 Seen hands it trains on and on the 648 Unseen hands, it leads on every metric, most widely in keeping the object (Unseen Success 47.5% vs. at most 34.3%) and tracking the grip command (Unseen Gain 0.59 vs. at most 0.25; Table 1). Hand by hand, it beats each baseline in FTE and in Gain on at least 99.8% of the Unseen hands (Figure 4), so its lead in tracking is not an average carried by a few hands. The baselines already track the command far less on the Seen hands (Gain 0.28 and 0.00 vs. 0.65), so their difficulty in regulating the grip force appears on the hands they learn together, not only on new ones. IBS-adapted, which shares our architecture but replaces the surface and object points with the IBS, does not track the command at all. Seeing the space between hand and object thus does not substitute for knowing the surface each actuator moves.

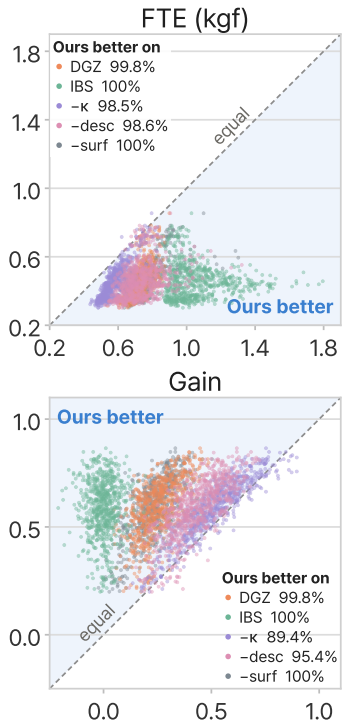


Figure 4: Per-hand FTE (top) and Gain (bottom) on the 648 Unseen hands: Ours ($y$) vs. each compared method ($x$).

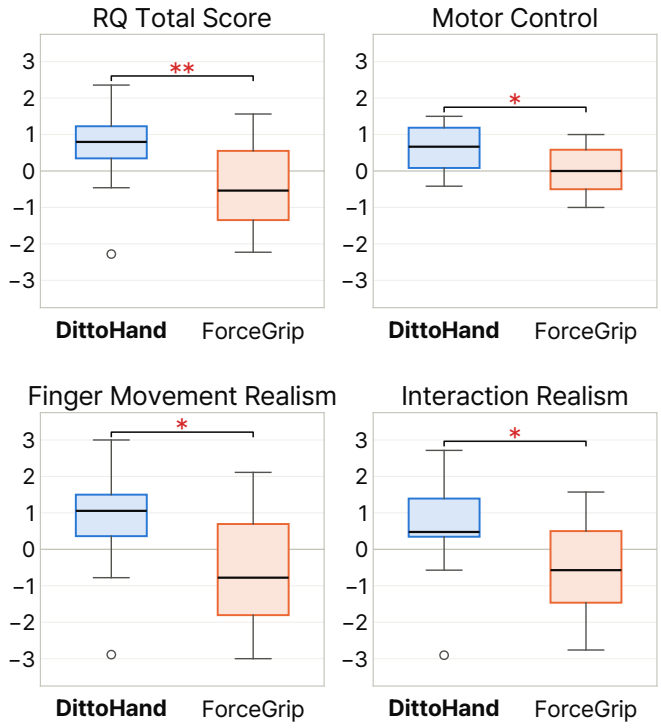


Figure 5: Realism Questionnaire (RQ) Total Score and subscores in the VR user study, from $-3$ to $+3$ (higher is better). Brackets mark significance after Holm correction ($*$: $p < 0.05$, $**$: $p < 0.01$).

Table 2: Ablations (Section 4.2). Grasp and Success are in %, FTE in kgf. Bold: best per column.

| | Seen hands (24) | | | | Unseen hands (648) | | | |
|---|---|---|---|---|---|---|---|---|
| Method | Grasp ↑ | Success ↑ | FTE ↓ | Gain → 1 | Grasp ↑ | Success ↑ | FTE ↓ | Gain → 1 |
| Ours | **74.6** | **66.6** | **0.35** | **0.65** | 55.4 | 47.5 | **0.46** | **0.59** |
| $-\kappa$ | 68.3 | 57.5 | 0.50 | 0.53 | 54.3 | **48.2** | 0.58 | 0.52 |
| − desc | 63.8 | 61.9 | 0.63 | 0.42 | **56.9** | 47.5 | 0.73 | 0.43 |
| − surf | 51.9 | 33.2 | 0.71 | 0.24 | 43.2 | 31.8 | 0.76 | 0.24 |

Two thirds of the Unseen hands have a topology no training hand has (three fingers or two-phalanx fingers). Such a topology gives every method its lowest Success (Topology only in Table 1), whereas a new $\boldsymbol{\beta}$ alone leaves every method near its Seen level. On the Topology-only hands, all three methods grasp about equally often (Grasp 49.0–50.3%), yet ours keeps the object far more often (Success 38.9% vs. at most 24.4%) and tracks the command more closely (Gain 0.53 vs. at most 0.23). On such hands a grasp can be found without knowing the surfaces, but keeping it at the commanded force depends on how each surface meets the object, which is what our actuator tokens carry. In every row of Table 1, including those with thickness and length beyond the levels of the Unseen hands, our policy leads on Success, FTE, and Gain, so its advantage covers hands that differ from the training hands in collision geometry, in kinematic structure, or in both.

### 4.4 Ablation Study

Removing all surface points, and with them the closing rate, drops the policy to the level of the baselines on both hand sets (Table 2; baselines in Table 1), even though it still observes the contact force on each phalanx. Our lead thus rests on what the tokens carry about the surface each actuator moves, even on the hands the policy trains on. Removing only the closing rate or the descendant

terms still raises FTE and lowers Gain on both hand sets, and ours has a lower FTE than $-\kappa$ and $-$ desc on 98.5% and 98.6% of the Unseen hands, respectively (Figure 4). These two parts also raise Grasp and Success on the Seen hands, but on the Unseen hands the two ablations grasp and keep the object about as often as ours: without them the policy can still infer a stable grasp on a new hand but matches the commanded force less well. Since the $-\kappa$ policy still sees the quantities from which $\kappa$ is computed, its weaker tracking supports giving $\kappa$ to the policy directly (Section 3.2).

### 4.5 VR USER STUDY

**Setup.** Scripted commands cannot show how a hand feels to the person driving it, so we ask users whether our single policy feels as controllable and realistic as ForceGrip (Han et al., 2025) trained on each hand separately. Our policy saw none of the three study hands, and each adds one kind of change (Figure 8). The mean MANO hand ($\boldsymbol{\beta} = \mathbf{0}$) differs from the five-fingered training hands only in $\boldsymbol{\beta}$. The same hand with finger thickness scaled by 1.15 also differs in collision geometry. A three-fingered hand (thumb, index, and ring finger) with a new $\boldsymbol{\beta}$ and finger thickness scaled by 1.135 differs in kinematic structure as well. Each ForceGrip policy keeps ForceGrip's observations and network and is trained on its study hand alone with our reward and curriculum.

**Procedure.** Twelve right-handed participants (nine men and three women, aged 21–33) took part, 11 of whom had used VR before. Wearing a Meta Quest 2 headset, they set the wrist pose with the controller in their right hand and the grip command with its trigger. In each trial they picked up one of the three training objects scaled by 0.75, 1, or 1.25, turned it to see the color of its underside, and placed it on the red or blue tray of that color (Figure 8). Each participant ran one block of 18 trials for each hand and method, one trial per combination of object, scale, and underside color. The blocks came in counterbalanced order, and participants did not know which method was running. Before each block they practiced with that hand and method until they felt ready. After each block they answered the 14-item Realism Questionnaire (RQ) of Oprea et al. (2019) and wrote what worked well and what did not. Before and after the session they filled in the Simulator Sickness Questionnaire (SSQ) (Kennedy et al., 1993).

**Results.** SSQ scores did not change significantly over the session (median 15.0 before and 16.8 after, $p = 0.96$). We score the RQ as Oprea et al. (2019) do, after mapping the 7-point answers to integers from $-3$ to $+3$, and compare the two methods on each participant's mean over the three hands with Wilcoxon signed-rank tests, Holm-corrected over the four measures. Participants rated our policy significantly higher than ForceGrip on the RQ Total Score and on all three subscores (Figure 5; $p \leq 0.012$, effect size $r \geq 0.77$). The largest gap is in Finger Movement Realism (median 1.06 vs. $-0.78$), and participants' comments echo it. Several described our policy's fingers as shaping to each object (P5, P9, P11), whereas four criticized ForceGrip's thumb (P1, P2, P5, P6). ForceGrip's ratings also spread more widely (RQ Total Score interquartile range 1.90 vs. 0.88), so its quality varied more from participant to participant. Overall, a policy that never saw the user's hand felt more controllable and realistic, and more consistently so, than policies trained for each hand, without training one per avatar. Appendix C details the ForceGrip training and the scoring.

## 5 CONCLUSION

To a policy that must reproduce a commanded grip force, a hand is its collision geometry as much as its kinematic structure. DittoHand builds this into one policy by tying each actuator to the collision surface it moves and to the closing rate at which that surface approaches the object as the actuator turns. An action then means the same on every hand, so the policy can act on a new hand from its actuator tokens alone. This lets a user choose or edit an avatar hand in VR and grasp with it at once, without training a policy for that hand.

However, the grip command sets only the total grip force, not each finger's share, and the tokens need the collision geometry of the hand and the object at run time. Feeding per-finger commands to the actuator tokens and estimating the surface points from observations could lift both limits.

### AI USE STATEMENT

In this work, we used generative AI tools to assist with method implementation, research methodology and experimental design, result interpretation, hypothesis refinement, translation, conceptual development, formulation of mathematical claims, dataset cleaning and reformatting, coding, figure preparation, experimental parameter selection, literature search, reference formatting, preparation of supplementary materials and the code release, and drafting and editing the paper. We did not use them for synthetic data generation or for qualitative and thematic data analysis, and proof-related tasks are not applicable because the paper contains no proofs. AI-assisted code was reviewed and run by the authors, reported results were obtained from our own experiments and user study, and citations were checked against their original sources. The authors take responsibility for the final content of this work.

### ETHICS STATEMENT

The user study involving human participants was reviewed and approved by the Institutional Review Board (IRB) of the authors' institution. All participants provided informed consent prior to participation. Identifying details of the IRB approval, including the institution and protocol number, are withheld during anonymous review and will be disclosed in the de-anonymized version.

### REPRODUCIBILITY STATEMENT

Section 3 and Appendix A describe the policy, with every token feature, its scaling, and the network sizes in Tables 3–5. Appendix B describes the experimental setup: how the Seen and Unseen hands are generated (Appendix B.1), how the objects are selected and prepared (Appendix B.2), how start poses are placed (Appendix B.3, Table 6), the simulation, reward, curriculum, PPO settings, and hardware (Appendix B.4, Tables 7–9), and how each baseline and ablation is built (Appendix B.5, Tables 10–11). Section 4.5 and Appendix C give the VR study protocol and scoring. The supplementary material contains a video and anonymized code; the code generates the hand sets, re-implements the FSG grasp generator, trains our policy and its ablations, and runs both evaluation experiments and builds the result tables. The MANO model and the YCB-Video objects are not redistributed because of their licenses; the README explains how to obtain them, and the code converts them. Code for the adapted baselines is not included; Appendix B.5 describes how each was built from its released code. The released code includes a small fix to how the object's center of mass is cached, so the first observation of each episode differs slightly from the runs reported here. Every method is evaluated at its final checkpoint after 25,000 iterations. The VR study data are not released to protect the participants' privacy.

## REFERENCES

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. Deep equilibrium models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Nico Bohlinger, Grzegorz Czechmanowski, Maciej Piotr Krupka, Piotr Kicki, Krzysztof Walas, Jan Peters, and Davide Tateo. One policy to run them all: An end-to-end learning approach to multi-embodiment locomotion. In *Conference on Robot Learning (CoRL)*, volume 270 of *Proceedings of Machine Learning Research*, pp. 3356–3378, 2024.

Christoph W. Borst and Arun P. Indugula. Realistic virtual grasping. In *IEEE Virtual Reality Conference (VR)*, pp. 91–98, 2005. doi: 10.1109/VR.2005.65.

Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *International Conference on Machine Learning (ICML)*, pp. 1725–1735, 2020.

Sammy Christen, Muhammed Kocabas, Emre Aksan, Jemin Hwangbo, Jie Song, and Otmar Hilliges. D-Grasp: Physically plausible dynamic grasp synthesis for hand-object interactions. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20577–20586, 2022.

John M. Danskin. *The Theory of Max-Min and its Application to Weapons Allocation Problems*, volume V of *Ökonometrie und Unternehmensforschung / Econometrics and Operations Research*. Springer, Berlin, Heidelberg, 1967. doi: 10.1007/978-3-642-46092-0.

Xin Fei, Zhixuan Xu, Huaicong Fang, Tianrui Zhang, and Lin Shao. T(R,O) Grasp: Efficient graph diffusion of robot-object spatial transformation for cross-embodiment dexterous grasping. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2026.

Carlo Ferrari and John Canny. Planning optimal grasps. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2290–2295, 1992. doi: 10.1109/ROBOT.1992.219918.

Agrim Gupta, Linxi Fan, Surya Ganguli, and Li Fei-Fei. MetaMorph: Learning universal controllers with transformers. In *International Conference on Learning Representations (ICLR)*, 2022.

DongHeun Han, RoUn Lee, KyeongMin Kim, and HyeongYeop Kang. VR-HandNet: A visually and physically plausible hand manipulation system in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 30(7):4170–4182, 2024. doi: 10.1109/TVCG.2023.3255991.

DongHeun Han, Byungmin Kim, RoUn Lee, KyeongMin Kim, Hyoseok Hwang, and HyeongYeop Kang. ForceGrip: Reference-free curriculum learning for realistic grip force control in VR hand manipulation. In *ACM SIGGRAPH 2025 Conference Papers*, 2025. doi: 10.1145/3721238.3730738.

Wenlong Huang, Igor Mordatch, and Deepak Pathak. One policy to control them all: Shared modular policies for agent-agnostic control. In *International Conference on Machine Learning (ICML)*, pp. 4455–4464, 2020.

Jan Jacobs and Bernd Fröhlich. A soft hand model for physically-based manipulation of virtual objects. In *IEEE Virtual Reality Conference (VR)*, pp. 11–18, 2011. doi: 10.1109/VR.2011.5759430.

Robert S. Kennedy, Norman E. Lane, Kevin S. Berbaum, and Michael G. Lilienthal. Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. *The International Journal of Aviation Psychology*, 3(3):203–220, 1993.

Jeffrey Kerr and Bernard Roth. Analysis of multifingered hands. *The International Journal of Robotics Research*, 4(4):3–17, 1986. doi: 10.1177/027836498600400401.

Vitaly Kurin, Maximilian Igl, Tim Rocktäschel, Wendelin Böhmer, and Shimon Whiteson. My body is a cage: the role of morphology in graph-based incompatible control. In *International Conference on Learning Representations (ICLR)*, 2021.

Puhao Li, Tengyu Liu, Yuyang Li, Yiran Geng, Yixin Zhu, Yaodong Yang, and Siyuan Huang. GenDexGrasp: Generalizable dexterous grasping. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8068–8074, 2023. doi: 10.1109/ICRA48891.2023.10160667.

Qingtao Liu, Yu Cui, Qi Ye, Zhengnan Sun, Haoming Li, Gaofeng Li, Lin Shao, and Jiming Chen. DexRepNet: Learning dexterous robotic grasping network with geometric and spatial hand-object representations. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3153–3160, 2023. doi: 10.1109/IROS55552.2023.10342334.

Qingtao Liu, Zhengnan Sun, Yu Cui, Haoming Li, Gaofeng Li, Lin Shao, Jiming Chen, and Qi Ye. DexRepNet++: Learning dexterous robotic manipulation with geometric and spatial hand-object representations. *IEEE Transactions on Robotics*, 42:799–818, 2026. doi: 10.1109/TRO.2026.3651669.

Michael D. McKay, Richard J. Beckman, and William J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979. doi: 10.1080/00401706.1979.10489755.

Richard M. Murray, Zexiang Li, and S. Shankar Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

Sergiu Oprea, Pablo Martinez-Gonzalez, Alberto Garcia-Garcia, John A. Castro-Vargas, Sergio Orts-Escolano, and Jose Garcia-Rodriguez. A visually realistic grasping system for object manipulation and interaction in virtual reality environments. *Computers & Graphics*, 83:77–86, 2019. doi: 10.1016/j.cag.2019.07.003.

Austin Patel and Shuran Song. GET-Zero: Graph embodiment transformer for zero-shot embodiment generalization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 14262–14269, 2025. doi: 10.1109/ICRA55743.2025.11127922.

Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 652–660, 2017.

Yuzhe Qin, Binghao Huang, Zhao-Heng Yin, Hao Su, and Xiaolong Wang. DexPoint: Generalizable point cloud reinforcement learning for sim-to-real dexterous manipulation. In *Conference on Robot Learning (CoRL)*, volume 205 of *Proceedings of Machine Learning Research*, pp. 594–605, 2022.

Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (SIGGRAPH Asia)*, 36(6), 2017.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Qijin She, Ruizhen Hu, Juzhan Xu, Min Liu, Kai Xu, and Hui Huang. Learning high-DOF reaching-and-grasping via dynamic representation of gripper-object interaction. *ACM Transactions on Graphics (SIGGRAPH)*, 41(4), 2022. doi: 10.1145/3528223.3530091.

Qijin She, Shishun Zhang, Yunfan Ye, Ruizhen Hu, and Kai Xu. Learning cross-hand policies of high-DOF reaching and grasping. In *European Conference on Computer Vision (ECCV)*, pp. 269–285, 2024.

Suzanne Sorli, Dan Casas, Mickeal Verschoor, Ana Tajadura-Jiménez, and Miguel A. Otaduy. Fine virtual manipulation with hands of different sizes. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 304–309, 2021. doi: 10.1109/ISMAR52148.2021.00046.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Mickeal Verschoor, Daniel Lobo, and Miguel A. Otaduy. Soft hand simulation for smooth and robust natural interaction. In *IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 183–190, 2018. doi: 10.1109/VR.2018.8447555.

Tingwu Wang, Renjie Liao, Jimmy Ba, and Sanja Fidler. NerveNet: Learning structured policy with graph neural networks. In *International Conference on Learning Representations (ICLR)*, 2018.

Zhenyu Wei, Zhixuan Xu, Jingxiang Guo, Yiwen Hou, Chongkai Gao, Zhehao Cai, Jiayu Luo, and Lin Shao. D(R,O) Grasp: A unified representation of robot and object interaction for cross-embodiment dexterous grasping. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4982–4988, 2025. doi: 10.1109/ICRA55743.2025.11127754.

Zhenyu Wei, Yunchao Yao, and Mingyu Ding. One hand to rule them all: Canonical representations for unified dexterous manipulation. In *Robotics: Science and Systems (RSS)*, 2026. doi: 10.15607/RSS.2026.XXII.124.

Liqi Wu, Haoyu Jia, Kento Kawaharazuka, Hirokazu Ishida, and Kei Okada. Dexterous grasp data augmentation based on grasp synthesis with fingertip workspace cloud and contact-aware sampling. *Advanced Robotics*, 39(12):730–747, 2025. doi: 10.1080/01691864.2025.2524553.

Yuliang Wu, Yanhan Lin, WengKit Lao, Yuhao Lin, Yi-Lin Wei, Wei-Shi Zheng, and Ancong Wu. DexGrasp-Zero: A morphology-aligned policy for zero-shot cross-embodiment dexterous grasping. In *Robotics: Science and Systems (RSS)*, 2026. doi: 10.15607/RSS.2026.XXII.122.

Table 3: Features of the token of actuator $j$, grouped by the encoder that reads them. Scales apply to SI units, with forces in kgf. Per-point vectors and $\boldsymbol{\omega}_j$ are in the joint frame of $j$, and the motion and contact force of the phalanx in the wrist frame.

| Feature | Dim |
|---|---|
| *Structure* ($\mathbf{e}_j^{\text{struct}}$) | |
| lower and upper angle limits, $/\pi$ | 2 |
| rest-pose bone vectors from the origin of the palm or phalanx carrying the joint to the joint and from the joint to the origin of the phalanx it turns, $\times 10$ | 6 |
| axis $\boldsymbol{\omega}_j$ | 3 |
| distance from the wrist, summed from the bone-vector lengths | 1 |
| depth along its finger over the largest depth | 1 |
| *State* ($\mathbf{e}_j^{\text{state}}$) | |
| angle $q_j$, velocity $\dot{q}_j/60$, and previous action | 3 |
| applied torque | 1 |
| position of the phalanx's center of mass, $\times 10$, and its linear velocity | 6 |
| magnitude and vector of the contact force on the phalanx, $/10$ | 4 |
| *Shape of the phalanx* ($\mathbf{h}_j$) | |
| positions $\mathbf{r}_{j,k}$, $\times 10$ | 48 |
| hand normals $\mathbf{n}_{j,k}^h$ | 48 |
| *Relation to the object* ($\mathbf{m}_j$) | |
| gaps $d_{j,k}$, $/0.1$ | 16 |
| unit directions $\mathbf{u}_{j,k}$ toward the nearest object point | 48 |
| object normals $\mathbf{n}_{j,k}^o$ at that point | 48 |
| gap derivatives $\partial d_{j,k}/\partial q_j = -\kappa_{j,k}$, $\times 10$ | 16 |

Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. In *Robotics: Science and Systems (RSS)*, 2018. doi: 10.15607/RSS.2018.XIV.019.

Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform badly for graph representation? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Haoqi Yuan, Bohan Zhou, Yuhui Fu, and Zongqing Lu. Cross-embodiment dexterous grasping with reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2025.

Hui Zhang, Sammy Christen, Zicong Fan, Otmar Hilliges, and Jie Song. GraspXL: Generating grasping motions for diverse objects at scale. In *European Conference on Computer Vision (ECCV)*, pp. 386–403, 2024.

# A Policy implementation

## A.1 Surface points

The surface points are sampled once per hand from the collision geometry of each phalanx and then move rigidly with it. We keep the contact-side quarter of this geometry and pick 16 points from it by farthest-point sampling. Each point is then assigned to one cell of a $4 \times 4$ grid over the length and width of the phalanx, so point $k$ lies in the same cell on every phalanx of every hand.

The joint frame of actuator $j$ has its origin at the joint and turns with the palm or phalanx that carries the joint. Where two actuators turn one phalanx at the same joint, the axis of the second turns with the first, so $\boldsymbol{\omega}_j$ is updated at every step.

## A.2 Token features

Table 3 lists the features of one actuator token and Table 4 the encoders that turn them into the token $\mathbf{T}_j$.

Table 4: Encoders from the token features to the token. All actuators share them. Each linear layer is followed by GELU, except the single linear layer of $E_{\mathrm{T}}$.

| Output | Input | Layers |
|---|---|---|
| $\mathbf{e}_j^{\mathrm{struct}}$ | structure features | $13 \to 64 \to 64$ |
| $\mathbf{e}_j^{\mathrm{state}}$ | state features | $14 \to 64 \to 64$ |
| $\mathbf{h}_j$ ($E_{\mathrm{h}}$) | shape of the phalanx | $96 \to 64 \to 64$ |
| $\mathbf{m}_j$ ($E_{\mathrm{m}}$) | relation to the object | $128 \to 64 \to 64$ |
| $\mathbf{e}_j^{\mathrm{surf}}$ ($E_{\mathrm{surf}}$) | $\mathbf{h}_j + \mathrm{mean}_{j' \in \mathcal{D}(j)} \mathbf{h}_{j'}$ and $\mathbf{m}_j$ | $128 \to 64$ |
| $\mathbf{e}_j^{\mathrm{desc}}$ ($E_{\mathrm{desc}}$) | $\mathbf{e}^{\mathrm{state}}$, $\mathbf{e}^{\mathrm{struct}}$, $\mathbf{h}$, and $\mathbf{m}$ of each descendant, then averaged | $256 \to 64 \to 64$ |
| $\mathbf{T}_j$ ($E_{\mathrm{T}}$) | the four embeddings | $256 \to 128$ |

Table 5: Network that reads the tokens.

| Part | Setting |
|---|---|
| Global features | last six $g$; $F_{\mathrm{grip}}$ and $F_{\mathrm{grip}} - F_{\mathrm{cmd}}$ in kgf, $/10$; object linear and angular velocity relative to the wrist; gravity direction; 17 numbers, vectors in the wrist frame |
| Object points | 512 points on the object surface, drawn once per object, in the palm frame, $\times 10$ |
| PointNet | shared MLP $3 \to 64 \to 128$ with ReLU, max pooling, linear $128 \to 64$ |
| Global token | learned vector plus a linear map $(17 + 64) \to 128$ |
| Transformer | 2 pre-norm layers, $d_{\mathrm{model}} = 128$, 4 attention heads, feed-forward $128 \to 256 \to 128$ with GELU, final LayerNorm |
| Attention bias | one learned scalar per attention head and bin ($4 \times 6$), shared by both layers |
| Surface re-addition | linear $64 \to 128$ and LayerNorm, scaled by one learned scalar that starts at 1, added to the actuator tokens before layer 2 |
| Action head | linear $128 \to 1$, shared by all actuators; the PD target is $q_j$ plus 0.5 rad times the action |
| Value head | linear $128 \to 128$, ELU, linear $128 \to 1$, on the critic's global token |

### A.3 Descendants and padding

Every hand gets 21 tokens, one for each actuator of a five-fingered hand with three phalanges per finger (five on the thumb and four on each other finger), and a hand with fewer actuators pads the rest. The tokens form five fixed chains, one per finger from its base to its tip. A padding token keeps its place in its chain, so it still counts as a step in the distances of the attention bias (Appendix A.4).

The descendants $\mathcal{D}(j)$ of actuator $j$ are the actuators after $j$ in its chain and the phalanges they turn. Padding in the middle of a chain does not cut it, so the distal actuator of a two-phalanx finger stays a descendant of the actuators at the finger's base. The mean in Equation 3 runs over the distinct phalanges in $\mathcal{D}(j)$ other than the one $j$ turns. The means in Equation 3 and Equation 4 are zero when $\mathcal{D}(j)$ is empty.

### A.4 Network architecture

Table 5 lists the network that reads the tokens. The attention bias sorts token pairs into six bins: a token with itself; actuators one, two, or three steps apart on one finger; actuators further apart or on different fingers; and any pair that involves the global token. The critic has the same architecture and inputs but its own weights.

## B Experimental setup

### B.1 Hand sets

Each component of $\boldsymbol{\beta}$ lies in $[-2, 2]$, two standard deviations of MANO's shape space along that component. A separate Latin hypercube (McKay et al., 1979) spreads the $\boldsymbol{\beta}$ of each finger count

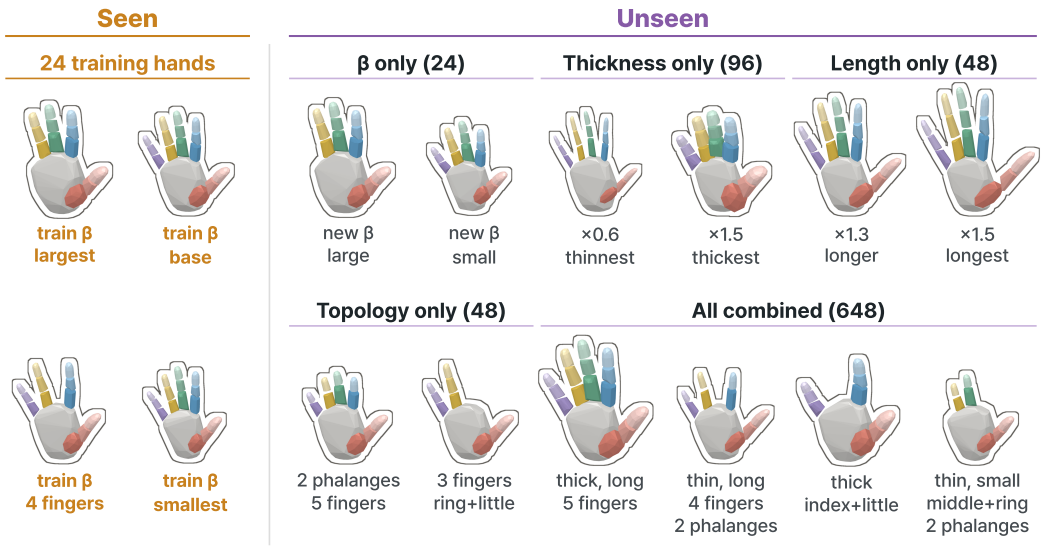


Figure 6: Example hands from each set in Table 1. Only the Seen hands are used in training.

over this range, and these $\beta$ are assigned at random to that count's finger layouts. Samples with a norm above 4.6 are discarded, since in these corners of the range many components are extreme at once. Among the Seen hands, each non-thumb finger is missing on three of the four-fingered hands. The four-fingered new $\beta$ have three per missing finger, and the three-fingered ones two for each of the six ways to keep two non-thumb fingers.

The thumb keeps three phalanges on every hand. A two-phalanx finger drops its middle phalanx and the actuator that bent it, so its distal phalanx follows the proximal one directly. Each missing finger removes four of the 21 actuators of a five-fingered hand and each two-phalanx finger removes one, so the hands have 11 to 21 actuators. Thickness scales the cross-section of every phalanx and moves no joint. Length stretches every phalanx along its bone with the palm and the finger roots fixed. This moves the joints farther along each finger but keeps each actuator's axis direction. Both edits also apply to the thumb. Length stays at or above 1.0 because the smallest hands in the $\beta$ range could not close around the objects with shorter fingers. The palm and each phalanx take their mass and inertia from their volume at one density of about 939 kg/m$^3$, so a thicker or longer finger is also heavier.

In Table 1, *$\beta$ only* holds the 24 Unseen hands with five or four fingers, three phalanges, and thickness and length at 1.0. *Thickness only* builds each Seen hand at thickness 0.6, 0.7, 1.3, and 1.5, and *Length only* at length 1.3 and 1.5, beyond the levels of the Unseen hands. *Topology only* builds each Seen hand once with two phalanges on every non-thumb finger and once with three fingers. With the Seen hands, the evaluation covers $24 + 648 + 192 = 864$ hands and thus the $864 \times 20 = 17{,}280$ hand–object pairs of Section 4.1.

### B.2 Objects

Figure 7 shows the 20 evaluation objects. A sphere of 4.5 cm diameter adds the continuously curved shape that YCB-Video lacks. Each YCB-Video object whose bounding box has a middle side above 4.5 cm is scaled down uniformly until that side is 4.5 cm, just above the 4 cm of the training objects. This side is the width the fingers wrap around. Of the 21 YCB-Video objects, we keep those whose thinnest side after scaling is at least 1.5 cm (about the thickness of a distal phalanx) and whose volume is at least 20% of their convex hull. A thinner part touches a finger at an edge, where the contact normal and hence $F_{\text{grip}}$ jump from step to step. A hollow shell loses its shape when its collision geometry is built by convex decomposition. These rules remove the scissors, the two clamps, the power drill, and the bowl. Every object has the mass, damping, and friction in Table 7.

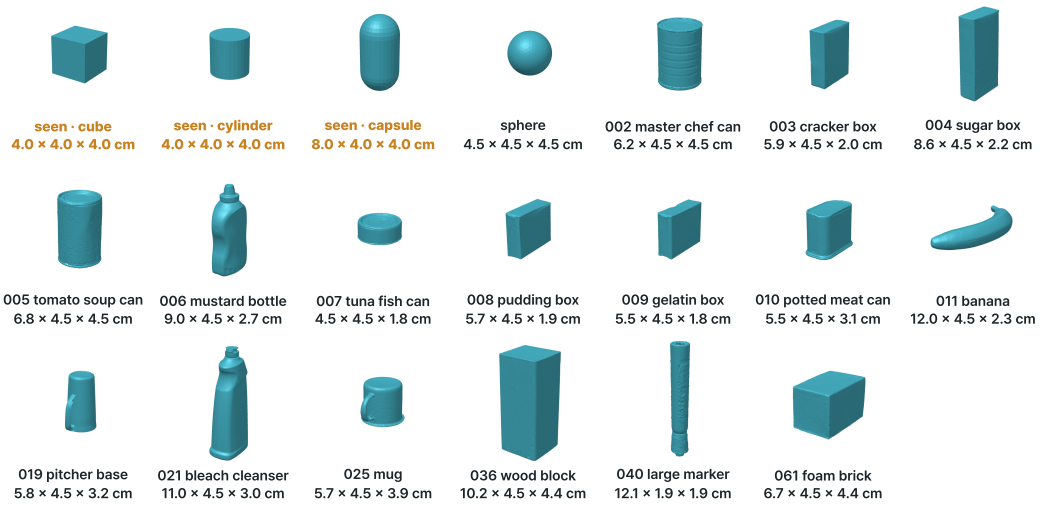


Figure 7: The 20 evaluation objects: the three training objects, the sphere, and 16 YCB-Video objects.

Table 6: Placement settings.

| | Value |
|---|---|
| Contact surfaces | thumb middle phalanx, non-thumb distal phalanges |
| Contact region | contact-side 25% of each contact phalanx, minus the distal 20% of this region on non-thumb fingers |
| Friction coefficient (FSG selection) | 0.5 |
| Object center | beyond 60% of the way from the wrist to the finger roots |
| Ferrari–Canny epsilon | $> 10^{-8}$ |
| Clearance at the default pose | $\geq 4$ mm |
| Sampling pool | better half by epsilon, at least 32 candidates |
| Position jitter | $\pm 1$ mm per axis |

### B.3 Object placement

For each hand–object pair, FSG (Wu et al., 2025) searches offline for object poses relative to the wrist at which the thumb and at least two other fingers (a minimum we impose) can reach a force-closure grasp without penetrating the object. Each candidate also keeps a clearance from the hand at the default finger pose, so that no finger starts against the object. Each reset draws a pose uniformly from the pair's best candidates by the Ferrari–Canny epsilon (Ferrari & Canny, 1992) and jitters its position (Table 6). For a training object, the hand and the object are then turned together by the smallest rotation that lets the object rest stably on the support. Sphere and YCB-Video poses are used as drawn. In evaluation all methods start from the same poses; only the random wrist motion and the command noise are drawn separately for each method. The Grasp metric (Section 4.1) applies FSG's force-closure test with a friction coefficient of 1.5 instead of the simulated 0.5, since at 0.5 the test rejects many grasps that go on to carry the object through the transport experiment.

### B.4 Simulation and training

All methods share the simulation settings in Table 7, the episode and curriculum below (Table 8), and the PPO (Schulman et al., 2017) settings in Table 9. Each of the 3,600 training environments keeps one hand and one object, so each training hand–object pair runs in 50 environments. Every method is evaluated at its last checkpoint, after 25,000 iterations.

Table 7: Simulation settings and training hardware.

| | Value | | Value |
|---|---|---|---|
| Simulator | Isaac Sim 5.1.0 (PhysX) | Hand–object friction | 0.5 |
| Solver position iterations | 8 | Support and floor friction | 0.5 |
| Solver velocity iterations | 1 | Restitution | 0.0 |
| PD stiffness | 1.0 N m/rad | Friction combine | min |
| PD damping | 0.1 N m s/rad | Object mass, damping | 0.1 kg, 0.5 $\mathrm{s}^{-1}$ |
| Joint torque limit | 0.72–2.37 N m | Support size | $0.5 \times 0.5 \times 0.04$ m |
| GPU | 1× NVIDIA RTX 6000 Ada (48 GB) | | |
| CPU | 2× Intel Xeon Gold 6526Y | | |

Table 8: Curriculum stages. The Support and Wrist columns apply after the object's release at 0.5 s.

| Stage | Iterations | Support | Wrist | Grip command | $g_{\min}$ |
|---|---|---|---|---|---|
| 1 | 0–3,000 | kept | fixed | constant per episode | 0.05 |
| 2 | 3,000–7,000 | kept | fixed | clipped sinusoid + noise | 0.05 |
| 3 | 7,000–14,000 | removed | fixed | clipped sinusoid + noise | 0.25 |
| 4 | from 14,000 | removed | random motion | clipped sinusoid + noise | 0.25 |

The proximity and opposition terms of the reward in Section 3.5 are

$$r_{\mathrm{prox}} = \sum_{p} w_p \exp\!\big(-0.07\, d_p^2\big)\big(0.25 + 0.75\,[-\mathbf{a}_p^{\top}\mathbf{n}_p^{o}]_+\big), \tag{6}$$

$$r_{\mathrm{opp}} = c\big(\max_{i\in\mathcal{T}}\|\mathbf{f}_i\|\big)\, c\big(\max_{i\in\mathcal{O}}\|\mathbf{f}_i\|\big). \tag{7}$$

The points $p$ of $r_{\mathrm{prox}}$ are the base of every phalanx and every fingertip. The weight $w_p$ is 0.04 at the bases of the proximal and middle phalanges and 0.06 at the base of the distal phalanx and at the fingertip, rescaled to sum to 1 over the points a hand has. Each $p$ has a distance $d_p$ to the object in cm, the contact-side normal $\mathbf{a}_p$ of its phalanx, and the outward object normal $\mathbf{n}_p^o$ at the nearest object point. The sets $\mathcal{T}$ and $\mathcal{O}$ of $r_{\mathrm{opp}}$ hold the phalanges of the thumb and of the other fingers, and $c$ is the cubic smoothstep from 0 at 0.03 kgf to 1 at 0.05 kgf.

While the object is held for the first 0.5 s, the wrist is also held still, and the grip command and all rewards are already active. Table 8 lists what each stage changes. Besides the 7 cm rule of Section 3.5, an episode ends early when a contact force on the hand exceeds 10 kgf (five times $F_{\max}$). Early termination adds no penalty.

In stage 1, $g$ is drawn uniformly from $[0, 1]$ once per episode. From stage 2, $g = 0.5 + b + A\sin(2\pi\nu t + \varphi)$ plus per-step Gaussian noise of standard deviation 0.05 clipped at $\pm 0.1$. Each episode draws $\nu$, $A$, $b$, and $\varphi$ uniformly from $[1/12, 1/2]$ Hz, $[0, 2]$, $[-1, 1]$, and $[0, 2\pi)$. In every stage $g$ is then clipped to $[g_{\min}, 1]$ with the stage's minimum $g_{\min}$ in Table 8. The minimum rises in stage 3 because a command near the object's 0.1 kgf weight could not be tracked without dropping the object once the support is gone. From the release in stage 4, the wrist tracks random linear and angular velocities of 50–100 cm/s and 270–540 deg/s, resampled every 0.2 s and smoothed with a 0.1 s time constant. The force experiment (Section 4.1) uses the stage-2 command generator with $g_{\min} = 0$, and the transport experiment uses the stage-4 wrist motion.

### B.5 Baselines and ablations

Besides these shared settings, both baselines use our action interface, so they differ from our policy only in what they observe and in the network that reads it. The actors of DGZ-adapted, IBS-adapted, and Ours have 497,676, 495,514, and 396,122 parameters. The ablations set what they remove to zero and keep every layer and width, so they remove information and not capacity.

**DGZ-adapted.** We keep the released graph network of DexGrasp-Zero (Wu et al., 2026) for the actor and the critic but drop its wrist head, since the user sets the wrist (Section 3.1). A fixed table

Table 9: PPO settings.

| | Value | | Value |
|---|---|---|---|
| Learning rate | $1 \times 10^{-4}$, adaptive | Value loss coefficient | 1.0 |
| Desired KL | 0.01 | Value clipping | yes |
| $\gamma$ | 0.99 | Max gradient norm | 1.0 |
| $\lambda$ | 0.95 | Learning epochs | 6 |
| Clip | 0.2 | Mini-batches | 4 |
| Entropy coefficient | 0.001 | Steps per environment | 16 |
| Action std | initial 0.4 | | |

Table 10: DGZ-adapted configuration.

| | Value |
|---|---|
| Nodes | 21: wrist and 5 fingers $\times$ 4 parts (proximal, middle, and distal phalanx, tip) |
| Graph layers | 8, widths 128 then 256 |
| Adjacency | anatomical tree with self-loops, learned |
| Node input | 12: offset to the object (3), actuator angles (3) and applied torques (3), contact force, grip command, grip-force error |
| Static input | 33, encoded $33 \rightarrow 32 \rightarrow 16$ |
| Decoder | per node, 256 features and the mask of its 3 motion primitives $\rightarrow 128 \rightarrow$ 3, one output per primitive (flexion, abduction, rotation) |

maps each node's motion-primitive outputs to its actuators. Each node's input replaces the original observation with what our task needs (Table 10). The offset to the object runs from the part's surface point nearest the object to the nearest object point. DGZ-adapted thus sees one offset per part instead of 16 surface points, and no closing rate. A static input codes each node's finger and part and holds its joint limits, axes, and bone vector. As in the original, it enters every graph layer of the actor after the first. A missing finger removes its nodes and edges, and a missing phalanx leaves its node without actuators.

**IBS-adapted.** We build the IBS (She et al., 2022; 2024) from candidate points on a $20^3$ grid centered on the palm (Table 11). Following the original, a candidate is kept when its distances to the object and to the nearest surface point differ by less than a threshold. The 512 kept points with the smallest difference form the IBS, and when fewer qualify the rest are marked invalid. The original uses 4,096 points, whereas 512 matches the object points our policy receives. Each point then moves to the midpoint between its nearest surface point and its nearest object point, so it lies locally equidistant from both surfaces. The original pools the IBS over the palm and the five fingers and so loses which phalanx is near the object. We pool it over the palm and each phalanx instead, so the IBS reaches each actuator by the same route as our surface points (Section 4.2).

## C VR USER STUDY

**Policies.** Each ForceGrip policy uses the observations and network of the released ForceGrip code (Han et al., 2025) and is trained for 25,000 iterations, so the two methods differ in input design as well as in per-hand training. Our policy is the checkpoint of the simulation experiments. For both methods and every hand, the analog trigger value is used as $g$ without filtering.

**Scoring.** By Eqs. 3–4 of Oprea et al. (2019), each subscore averages its items with the negatively worded ones subtracted, and the RQ Total Score averages the three subscores with Interaction Realism counted twice. The signed-rank tests are exact, and the effect size is $r = |Z|/\sqrt{n}$, with $Z$ from the normal approximation and $n$ the number of participants whose two ratings differ. Per hand, the median RQ Total Score was 1.10 for our policy and $-0.12$ for ForceGrip on the mean MANO hand ($p = 0.001$), 0.93 and 0.03 on the thicker hand ($p = 0.32$), and 0.60 and $-1.09$ on the three-fingered hand ($p = 0.003$), by the same tests Holm-corrected over the three hands. On the thicker hand the median still favored our policy, but ratings split (six higher, five lower). Several participants noted

Table 11: IBS-adapted configuration.

| | Value |
|---|---|
| Candidate grid | $20^3$ points over a 0.36 m cube centered on the palm |
| Keep threshold | distance difference below 1.9 cm (grid spacing) |
| IBS points | 512 |
| Parts | palm and each phalanx (up to 16) |
| Point features | 23: position (3), distances to both surfaces (2), validity, part one-hot (16), palm-facing component of the nearest hand normal at rest |
| Per-part encoder | PointNet $23 \to 64 \to 128$, max over each part, to that phalanx's actuators |
| Global encoder | PointNet $23 \to 64 \to 128$, max, linear to 512, in place of the object PointNet; the global token's linear map becomes $(17 + 512) \to 128$ |

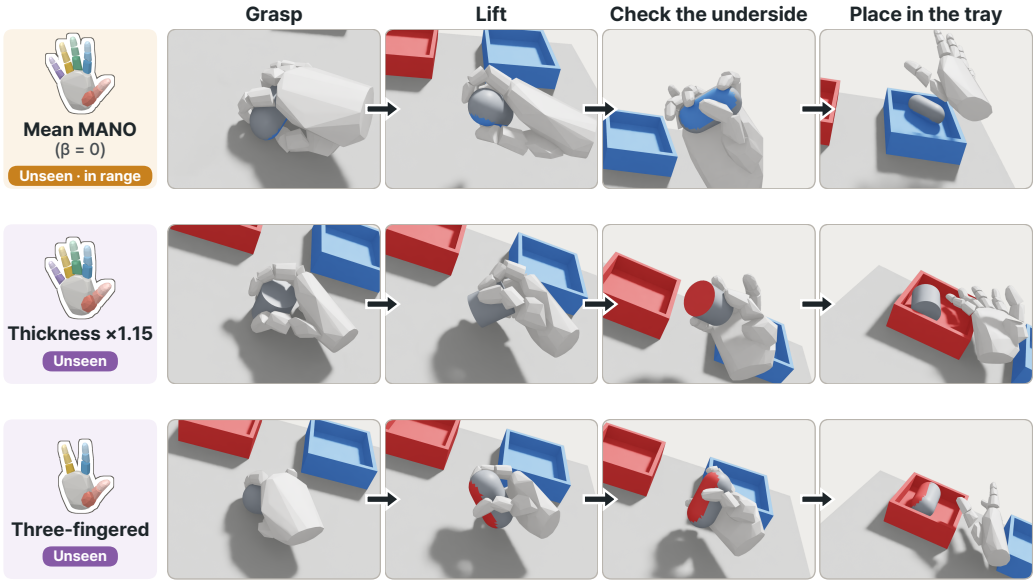


Figure 8: The participant's view in the VR user study. Each row shows one study hand, and the columns follow the task sequence of a trial. The underside color of the object tells which tray it belongs to.

objects slipping from our policy's grasp on this hand (P5, P6, P7, P12). However, ForceGrip drew similar complaints: objects slipped from its grasp (P1, P9, P12) or its grip felt weaker than intended (P2, P11). The SSQ total uses the weights of Kennedy et al. (1993), and its change over the session is tested with the same signed-rank test.