



# Kinematic Motion Retargeting for Contact-Rich Anthropomorphic Manipulations

ARJUN S. LAKSHMIPATHY, School of Computer Science, Carnegie Mellon University, Pittsburgh, United States

JESSICA K. HODGINS, School of Computer Science, Carnegie Mellon University, Pittsburgh, United States

NANCY S. POLLARD, School of Computer Science, Carnegie Mellon University, Pittsburgh, United States

Hand motion capture data are now relatively easy to obtain, even for complicated grasps; however, these data are of limited use without the ability to retarget it onto the hands of a specific character or robot. The target hand may differ dramatically in geometry, number of degree of freedom (DOF), or number of fingers. We present a simple but effective framework capable of kinematically retargeting human hand-object manipulations from a publicly available dataset to diverse target hands through the exploitation of contact areas. We do so by formulating the retargeting operation as a nonisometric shape matching problem and use a combination of both surface contact and marker data to progressively estimate, refine, and fit the final target hand trajectory using inverse kinematics. Foundational to our framework is the introduction of a novel shape matching process, which we show enables predictable and robust transfer of contact data over full manipulations (pregrasp, pickup, in-hand re-orientation, and release) while providing an intuitive means for artists to specify correspondences with relatively few inputs. We validate our framework through demonstrations across five different hands and six motions of different objects. We additionally demonstrate a bimanual task, perform stress tests, and compare our method against existing hand retargeting approaches. Finally, we demonstrate our method enabling novel capabilities such as object substitution and the ability to visualize the impact of hand design choices over full trajectories.

CCS Concepts: • **Computing methodologies** → **Procedural animation; Parametric curve and surface models;**

Additional Key Words and Phrases: Character Animation, Retargeting, Hands, Contact-Driven Methods, Dexterous Manipulation, Motion Synthesis, Shape Matching, Grasping, Inverse Kinematics, Optimization

## ACM Reference Format:

Arjun S. Lakshmiopathy, Jessica K. Hodgins, and Nancy S. Pollard. 2025. Kinematic Motion Retargeting for Contact-Rich Anthropomorphic Manipulations. *ACM Trans. Graph.* 44, 2, Article 14 (April 2025), 20 pages. <https://doi.org/10.1145/3723872>

## 1 Introduction

Advancements in motion capture technology have made it possible to collect motion data at high levels of detail, including large-scale

Authors' Contact Information: Arjun S. Lakshmiopathy, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, United States; e-mail: aslakshm@andrew.cmu.edu; Jessica K. Hodgins, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, United States; e-mail: jkh@cmu.edu; Nancy S. Pollard, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, United States; e-mail: npollard@andrew.cmu.edu.



This work is licensed under a Creative Commons Attribution 4.0 International License.

© 2025 Copyright held by the owner/author(s).  
ACM 0730-0301/2025/04-ART14  
<https://doi.org/10.1145/3723872>

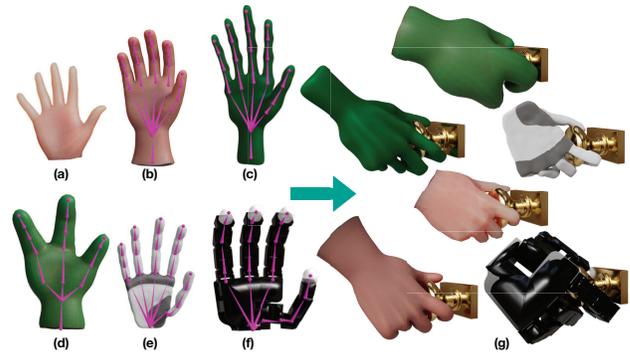


Fig. 1. All hands used in our experiments, including the (a) source MANO, (b) alternate human, (c) witch, (d) alien, (e) custom prosthetic, and (f) Allegro Hand. (g) We retarget demonstrations performed by the source hand to all these hands by procedurally transferring contact areas over the entire time series via shape matching.

body movements and fine-grained hand movements together in a single capture [Romero et al. 2017; Taheri et al. 2020]; however, using these data are still difficult, because the target embodiment for the data is almost never the same as that of the demonstrator. For hands in particular, retargeting is often required because of differences in hand models and the precision required to make grasps, especially those with many points of contact, look compelling. Unfortunately, this process often generates artifacts such as motion misalignment or lack of contact that are difficult to repair in post-processing. The problem becomes even more challenging when the target hand is not humanlike, whether that includes different finger proportions, **degrees of freedom (DOFs)**, or number of fingers.

The lack of reliable retargeting methods for adapting existing contact-rich hand motion data remains a bottleneck for multiple communities. In film and game production, considerable animator time is spent customizing common manipulations to different character rigs. In robot learning, the difficulty in adapting existing manipulation data to different hands significantly contributes to the data sparsity problem [Sivakumar et al. 2022], which in turn limits the feasibility of supervised and semi-supervised machine learning algorithms. A standardized approach for re-purposing existing data, and in particular data easily collectible from humans, would provide a much-needed solution for a broad range of applications.

This article presents an approach for kinematic retargeting of existing contact-rich *anthropomorphic* hand-object manipulations through the use of dense correspondences (bijections of discretized point sets) between contact areas. To do so, we treat time-varying

contact distributions as a foundational retargeting medium and formulate the mapping of distributions between different hands as a nonisometric shape matching problem. We then show that exploiting retargeted contact correspondences between hands and objects leads to the development of a straightforward motion synthesis pipeline. Despite its simplicity, our approach is robust, predictable, and effective, which we demonstrate by mapping motions of varying complexity from the publicly available GRAB dataset [Taheri et al. 2020] to five kinematically, morphologically, and geometrically diverse hands sourced as-is from artist rigs intended for media production and robot manipulators intended for simulation.

We make the following novel technical contributions:

- A local shape matching algorithm for robust time-series contact transfer based on an intuitive markup scheme
- A straightforward multi-stage optimization pipeline to compute the retargeted motion

We evaluate our method with the following experiments:

- Thirty retargeted single-hand demonstrations from a publicly available dataset (5 hands  $\times$  6 motions of different objects—details in Appendix B) as well as a bimanual task
- Quantitative evaluation via distance and intersection metrics
- Stress tests, including cyclic retargeting and re-using contacts across hand size variations
- Baseline comparisons and ablations to validate the importance of contact information
- Novel extensions, including visualizing the impact of hand design parameters and object substitution

The techniques and algorithms proposed in this article are designed for *standardization and simplicity*, which we show enable *reliable generation of high-quality retargeted hand motion trajectories*. Our decision to focus on kinematics, rather than dynamics, is primarily motivated by reliability and secondarily by speed, as well as the conjecture that, for this problem, kinematic solutions are sufficiently constrained through the use of multiple large contact areas. We show that our method produces high-quality results, generalizes well across manipulators, and gracefully handles complex manipulation tasks with many and changing contacts. Depending on the application, our solutions can be considered complete as-is or a first pass for additional processing.

## 2 Related Work

We consider three categories of relevant prior work: motion retargeting, shape matching, and contact-driven grasp synthesis. These categories collectively provide both an overview of existing retargeting strategies and background for our shape matching contribution.

### 2.1 Motion Retargeting

Adapting existing motion data to new characters is a long-standing and well-studied problem for both full-body and hand animation.

Joint-based retargeting remains a common approach in full-body character animation; however, resolving environmental and self-contact events complicates the process. Proposed solutions include adapting to variations in body shape using learning-based

methods exclusively [Aberman et al. 2020] or augmented by physics-based simulation [Ryu et al. 2021; Won and Lee 2019], differentiating between unwanted self-contacts and desired foot contacts during optimization [Villegas et al. 2021], exploiting spatial maps [Jang et al. 2024; Kim et al. 2016], standardized wrapper meshes [Jin et al. 2018] or interaction graphs [Zhang et al. 2023a], and dynamics-augmented projected motion optimization [Lee and Lee 2019]. These methods have produced useful results; however, a key distinction between hand and full-body retargeting is the number of contact points required to make a motion visually compelling. More concretely, single point contacts between objects and different body parts have proven largely sufficient at the full-body scale even for transferring highly complex interactions between multiple characters and objects [Zhang et al. 2023a]. Conversely, a small number of points has been shown to insufficiently model the complexity of realistic interaction between hands and objects [Lakshmipathy et al. 2023]. Additionally, as illustrated in Figure 1, the problem is further compounded by the fact that hand shapes, morphologies, and kinematics can vary widely between different characters.

Common approaches to retargeting motions from hand tracking include direct joint mapping [Kumar and Todorov 2015; Rajeswaran et al. 2018], keypoint-based **inverse kinematics (IK)** [Antotsiou et al. 2018; Dasari et al. 2023; Humberston and Pai 2015; Qin et al. 2022], inverse dynamics from input joint angles [Kry and Pai 2006], and relative keyvector distances [Handa et al. 2019; Sivakumar et al. 2022]. The latter methods remain state-of-the-art approaches when the hands are more divergent. These approaches can also be adapted to previously collected data; however, we show that doing so often creates significant artifacts that are difficult to clean up in postprocessing.

Physics simulation is also commonly used to improve visual plausibility and Sim2Real transfer success. Due to the complexities of frequent making and breaking of contacts during manipulation, several approaches employ learning from demonstration to generate controllers from human motion data [Dasari et al. 2023; Qin et al. 2022; Wu et al. 2022a]. Standard kinematic retargeting techniques (e.g., direct joint mapping, keypoint or keyvector IK) are often used in these approaches to create expert trajectories and subsequently prepopulate reward tables; however, this strategy can result in failures or unexpected results if the retargeted hand trajectory is poor. Identifying retargeting failures, as well as disentangling such error contributions from policy learning and reward shaping, is especially challenging at scale. Our goal in this work is to provide a solution for reliable upstream expert trajectory estimation that can be used in conjunction with any downstream physics simulator.

There is currently *no standardized solution* for reliably retargeting existing hand-object motion data. We argue in this work that contact information is vital to produce high-quality results across a broad range of motions and hands.

### 2.2 Shape Matching

Shape matching addresses the problem of finding geometrically meaningful correspondences between different shapes. Existing literature in this space can be approximately divided into isometric

and nonisometric problem domains. We provide a brief summary of each variant.

Isometric shape matching concerns instances where the baseline shape is (approximately) the same. Common examples include character meshes that have undergone skeleton-driven deformation or deformable objects. The key assumption in this domain is that the Riemannian metric (e.g., distances, angles) is preserved. This assumption enables the use of methods such as functional maps with consistent descriptor functions [Attaiki et al. 2021; Ovsjanikov et al. 2012; Pai et al. 2021] and heat diffusion [Sharp et al. 2022] to identify equivalences; consequently, problems in this domain are comparatively easier. Unfortunately, these techniques are rarely applicable to our domain due to the large range of geometric variations in hand models.

Nonisometric shape matching relaxes the preservation assumption, thereby extending the problem space to fully differentiated shapes at the cost of substantially escalating problem difficulty. The majority of existing works in this space consider mapping of global media (e.g., textures) and require some form of user input to compensate for the lack of reliable quantitative measures of equivalence, typically in the form of landmark points [Ezuz et al. 2019; Panine et al. 2022] or curves [Gehre et al. 2018]. Landmarks are subsequently used as boundary conditions [Gehre et al. 2018; Panine et al. 2022], as keypoints for the extraction of hyperbolic Tutte embeddings [Aigerman and Lipman 2016; Takayama 2022], or other demarcations. Although we initially tried several such techniques, we found that they were not able to handle all of the variations in our hand and object geometries and generally struggled with exact mappings of individual points (point-to-point transfers). We show in this work that relaxing the need to map globally smooth media enables the use of alternate strategies capable of performing precise point-to-point transfers from similar landmark annotations.

Efforts have been made to fully automate the nonisometric case, including automating the discovery of landmarks through heuristics [Edelstein et al. 2020; Marin et al. 2020], aligning extrinsic correspondences such as normals [Li et al. 2007], and automatically segmenting meshes via data-driven techniques [Kalogerakis et al. 2010]. However, the immense variety in hand designs (number of finger segments, finger lengths, finger counts, palm shapes, etc.) compounded with their ability to change shape with pose render the application of heuristics or extrinsic metrics particularly challenging. Existing pose data for arbitrary hand models is also difficult to obtain as it is infeasible to cover the entire space of all possible hand designs. Furthermore, fully automated methods typically permit little control over results, which can render such techniques unusable for quality-critical applications. In this work, we instead focus on creating highly reliable mappings for *direct*, local data transfer using simple curve-based annotations easily obtainable from an artist. We also note that our approach is *dependent only on intrinsic quantities*, which inherently makes it robust to isometric changes in geometry (e.g., deformation induced by pose changes).

### 2.3 Contact-driven Grasp Synthesis

Contact points have played a vital role in the generation of grasps and manipulations. Techniques include analyzing optimal indepen-

dent regions within which point contacts can be placed [Roa and Suarez 2009] or using contacts as error terms in the training of generative models [Christen et al. 2022; Wu et al. 2022b]. Approaches in manipulation planning switch between contact modes [Cheng et al. 2021] or optimize single point contact placement and force for physically plausible results [Hazard et al. 2020; Mordatch et al. 2012; Ye and Liu 2012]. The Ferrari-Canny metric [Ferrari and Canny 1992]—the most widely used grasp quality metric—is fundamentally rooted in its analysis at individual contact points.

However, characterization of grasps as single points highly simplifies the complexities of real interactions [Brahmbhatt et al. 2019a, 2020] and cannot account for geometric consistency outside the designated point. Consequently, there has been considerable recent interest in collecting, analyzing, and exploiting contact areas kinematically [Brahmbhatt et al. 2019b; Fan et al. 2023; Lakshminpathy et al. 2022, 2021; Taheri et al. 2020] and dynamically [Pang et al. 2023; Turpin et al. 2022]. We focus on kinematics in this article, following recent works that generate and optimize poses by matching contact patches [Brahmbhatt et al. 2019b; Grady et al. 2021; Lakshminpathy et al. 2022, 2023; Wei et al. 2023]. Our primary goal in this work is to establish the techniques necessary to adapt contact patch-driven grasping strategies to full manipulations. Although this idea appears straightforward, it was not possible to do so without completely rethinking the problem of shape matching due to the dense, complex, and constantly changing collections of contact areas that had to be transferred meaningfully between different hands.

## 3 Method

Our retargeting pipeline is shown in Figure 2 and can be divided into the following steps:

- (1) Extract a dense corresponding set of contact areas between the object and source hand per frame
- (2) Procedurally transfer all contacts from the source to target hand across all frames
- (3) Estimate an initial trajectory for the target hand using fixed markers and the transferred contacts
- (4) Refine the estimates to improve temporal consistency
- (5) Construct the final trajectory through B-Spline fitting

Importantly, the design of our pipeline assumes that the desired solution is one that *attempts to match the interaction mechanics of the source manipulation as exactly as possible*. For this reason, we assume contacts on the object are the same across all hands. We detail the inputs and processing steps in the following subsections.

### 3.1 Inputs

Our method requires existing hand-object motion data as input, which we expect to minimally include the following:

- (i) Accurate meshes of the original object and source hand
- (ii) A set of dense per-frame contact annotations on either the object or hand mesh
- (iii) A complete set of frames defining the motion sequence

The GRAB [Taheri et al. 2020] and ARTIC [Fan et al. 2023] datasets contain all three types of data. We select the former as

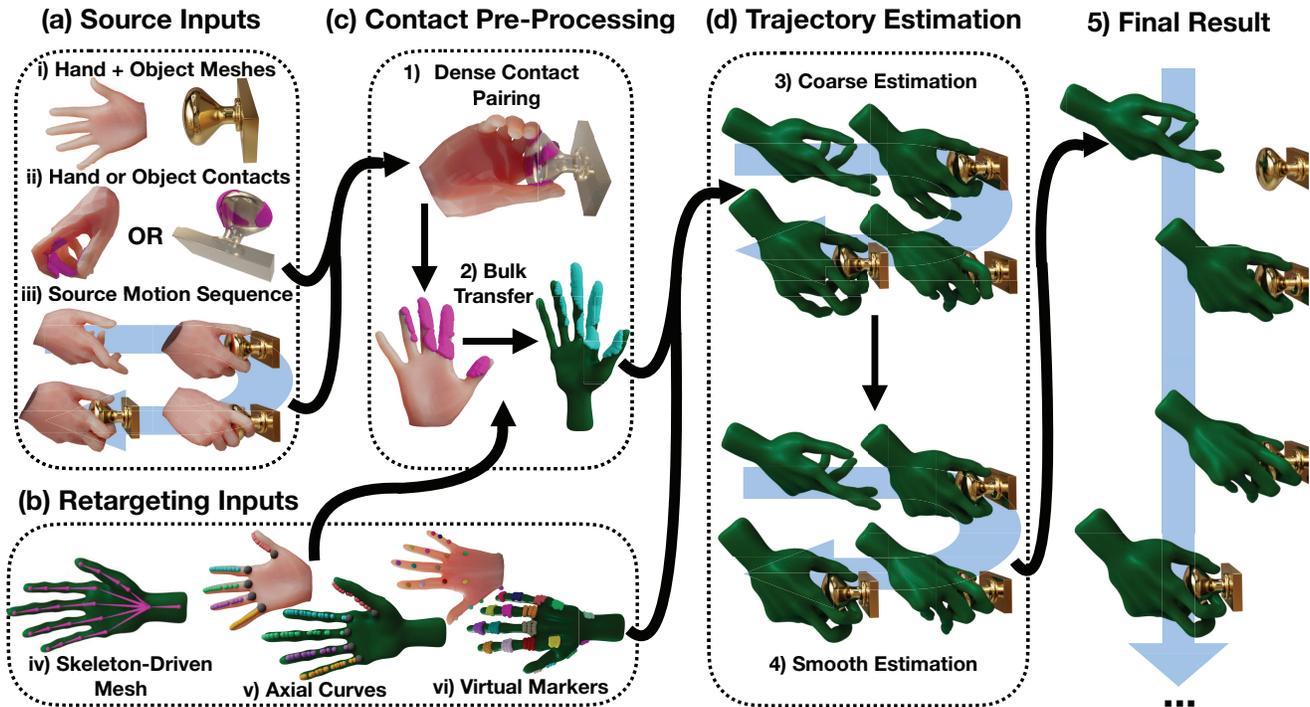


Fig. 2. Overview of our retargeting framework. (a) Our approach requires inputs of accurate meshes of the original object and source hand, per-frame contacts on either the object or source hand, and complete motion sequences of the object and source hand. (b) To perform the retarget, we require a skeleton-driven target hand mesh as well as a set of artist-annotated corresponding virtual markers and axial curves. (c) After recovering a dense set of contacts between the object and source hand, we transfer hand contacts across the entire time series and (d) use the virtual markers and transferred contacts to synthesize motion for the target hand from scratch.

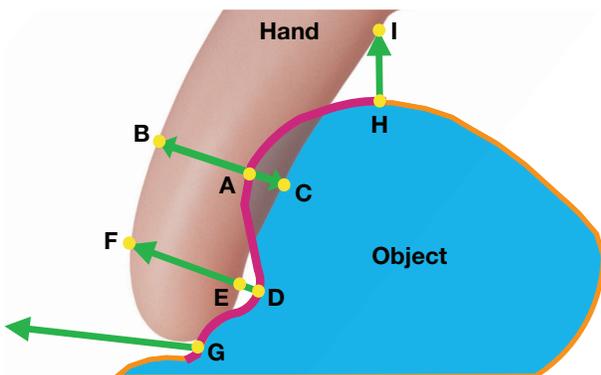


Fig. 3. Illustration of the raycasting procedure used to obtain one-to-one correspondences of a contact region (magenta) between the object (blue interior, orange boundary) and hand (skin-colored). Starting from contact points on the object boundary (yellow points A, D, G, and H), we cast rays along surface normals (green) and must resolve three possible events. If a single hand intersection is found (B), then we assume the contact region is located within the hand and invert the direction to obtain the contact point on the palmar surface (C). If two intersections are found (E and F, I and beyond), then we take the closer of the two intersections by distance (E, I). However, we discard instances that exceed a specified distance threshold (H, I) as errors. Finally, if no intersections are found (G), then the contact point is considered an error and discarded.

the data source for our experiments. Importantly, we only require one set of dense contacts and do *not* require the original hand skeleton.

For retargeting to a new hand, we require the following:

- (iv) A skeleton-driven target hand mesh
- (v) A set of artist-annotated *corresponding virtual markers* on the source and target hand
- (vi) A set of artist-annotated *corresponding axial curves* [Lakshmiopathy et al. 2023] on the source and target hand in their respective default configurations

We expand on the required artist annotations and their usages in the following sections but note that the work required to prepare a new hand is typically under 1 hour. We also require the source and target hands to be manifold; however, it is possible to overcome this constraint by using an approximate manifold wrapper. We demonstrate a result with the fully articulated Allegro Hand<sup>1</sup> as an example. Importantly, we do not assume any similarity between the source and target hand (e.g., identical finger counts, triangulations, finger or palm shape, finger length).

### 3.2 Dense Contact Pairing

Consistent with existing datasets, we assume contact areas are represented by a finite set of discrete points per frame across the

<sup>1</sup><https://www.wonikrobotics.com/research-robot-hand>

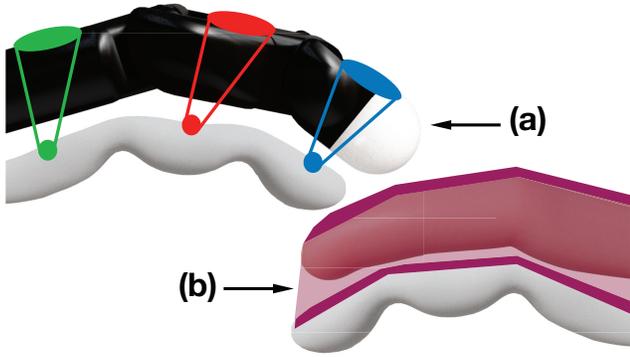


Fig. 4. Virtual markers can be configured as traditional single-point one-to-one, or alternatively (a) heterogeneous many-to-one, or (b) dense, area-based configurations. Configuration (a) can be utilized to model virtual marker location uncertainty between differing source and target hands, which can be useful when finger link lengths are different sizes. Configuration (b) can be used to weight the importance of matching the deformed hand states over large regions, which can be useful when deformation behaviors diverge despite similar link lengths.

entire motion sequence. We assume a one-to-one correspondence between each hand-object contact point pair. Contact points are stored as barycentric coordinates to render our approach sampling agnostic, and thus robust, to variances in triangulations across different meshes. This approach ensures that data can be collected and transferred from coarsely triangulated hands (e.g., the MANO hand [Romero et al. 2017]) to arbitrarily fine or irregularly sampled target hand meshes without risk of discretization error from vertex clamping.

In the event of only one dense set being available, as in the case of the GRAB dataset, we can generate a corresponding dense set through raycasting. We trace contacts out from source mesh locations along element normals until the opposite mesh is intersected, automatically generating a paired barycentric point. In the event of penetration into the source mesh, we invert the trace direction and retry. Points that do not intersect the opposite mesh or are further apart than an  $\epsilon$ -metric are considered errors and discarded. We note that while this technique proved reasonably reliable in practice, results tend to be inaccurate if the source manifold exhibits high local curvature at the contacts. Figure 2 illustrates an example of dense pairing obtained from raycasting, while Figure 3 illustrates the procedure in more detail.

### 3.3 Hand Shape Matching

We use the deformed state of the source hand and a (possibly empty) set of dense corresponding contacts between the object and source hand to compute the target hand pose per frame. As demonstrated in the grasping literature [Lakshminpathy et al. 2022, 2023; Wei et al. 2023], we can robustly compute target hand poses during contact-rich frames if we know the target hand contact distribution. To create these contact distributions, we transfer contacts from source to target hands over the entire time series using a one-time set of artist annotations consisting of *virtual markers* and *axial curves*. Virtual markers help to define hand poses in situations where contact information is sparse or absent



Fig. 5. Single point virtual marker configuration on the source MANO hand and area based corresponding marker sets on all other hands used in our experiments. We use a manifold wrapper of the Allegro Hand for contact processing. Marker points were selected at vertex indices for convenience.

altogether, while axial curves provide an intuitive control scheme for the placement of landmark points and adjustment of landmark parameters. Landmarks and parameters are subsequently used to enable scalable and customizable contact transfer through the use of local charts, as described in Section 3.3.2. Importantly, our annotation requirements are no greater than existing state of the art methods that exploit user-provided landmark points or curves for nonisometric shape matching [Aigerman and Lipman 2016; Gehre et al. 2018; Panine et al. 2022; Takayama 2022] or keypoint tracking [Dasari et al. 2023; Humberston and Pai 2015; Qin et al. 2022; Sivakumar et al. 2022; Wei et al. 2023], yet they adapt well to dense and rapidly changing contacts. Details of annotations and usage procedures are described below, while Figure 2 illustrates the shape matching inputs.

**3.3.1 Virtual Marker Alignment.** Our pipeline expects a corresponding set of artist-annotated virtual markers on both the source and target hand to assist with pose computation, particularly when contact data are sparse or unavailable. We define virtual markers as an arbitrary collection of *fixed* corresponding sets of mesh points between the source and target hand. Markers can either be traditional single points or areas. In the former case, aligning virtual markers for pose computation reduces to traditional keypoint-based IK solving consistent with existing literature [Dasari et al. 2023; Humberston and Pai 2015; Qin et al. 2022; Sivakumar et al. 2022; Wei et al. 2023]. Area-based correspondences can be generated by applying existing contact transfer methods [Lakshminpathy et al. 2022, 2023] to produce an automatic mapping between such markers, which can be viewed as analogous to matching “contacts” in mid-air. Notably, however, an area-area configuration requires both sets to contain a 1:1 correspondence of discretized elements. Heterogeneous mappings between single point and area based markers are also possible and trivial to designate as a one-to-many association under this modality. Figure 4 illustrates each configuration and instances where heterogeneous and area-based correspondences may be beneficial.

The annotation procedure is manual but was straightforward for an artist and only has to be performed once per source and target hand pair. Details of the annotation process are provided in Appendix A. Figure 5 illustrates the heterogeneous virtual marker sets received from the annotation process and subsequently used as-is for all results, in which the source hand was allocated a set of single-point markers and the target hand an area-based set. In am-



Fig. 6. (Top row) Object contacts, (second row) source hand contacts, (third row) computed target hand contacts, and (Bottom Row) source and retargeted hand motion for four different stages of a phone manipulation: (first column) table pickup, (second column) in-hand dialing, (third column) holding for use, and (last column) movement back toward the table for release. Although poses and contact distributions vary dramatically during the manipulation, our method can successfully produce target hand motion by using source hand contact distributions as a foundational retargeting medium.

biguous cases such as the three-fingered alien hand, the received configuration mapped each human finger marker set to a single finger on the target hand (e.g., thumb, index, middle on the alien hand); however, configurations that map multiple source finger marker sets to a single target finger are supported as well. We also found that area-based virtual markers proved useful in modeling uncertainty on target hands, because it was not always clear where a corresponding single-point virtual marker should be placed.

**3.3.2 Contact Alignment.** Unlike markers, contact distributions are dynamic and can vary greatly between motion frames (e.g., Figure 6). We model bulk transfer as an intrinsic nonisometric shape matching problem, with the key insight of formulating the correspondence as being governed by an atlas of multiple coordinate charts [Jost 2008]. Atlases can be comprised of one or many coordinate charts, provided the union of all charts generates a cover of the underlying manifold [Jost 2008].

Techniques targeting transfer of global media (e.g., textures), however, typically consider only single chart correspondences and require careful landmark placement to use effectively. Multiple chart correspondences are uncommon in such contexts, because careful handling of interpolation across transition regions and hard chart boundaries is required to maintain global smoothness [Jost 2008]. Additionally, we found that the inherently high sensitivity to landmark placement makes artist annotation difficult in practice.

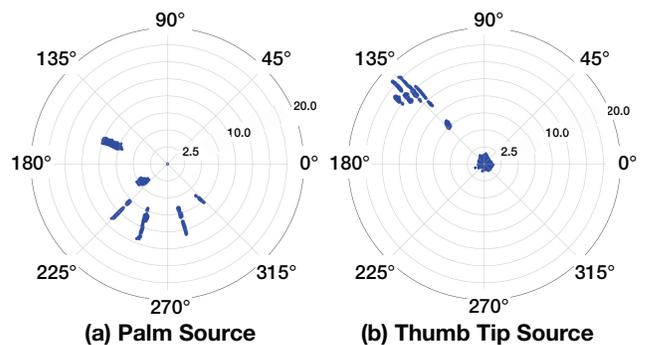


Fig. 7. (a) When examining geodesic distance and outgoing tangent vector direction from a landmark point on the palm, clusters of contact points are clearly separable. (b) Viewing the same distribution from a landmark on the thumb tip cleanly isolates thumb contacts from those of remaining fingers, allowing us to easily introduce discontinuities to filter out the remaining contacts and subsequently perform arbitrary transformations exclusively on the isolated distribution.

The highly local nature of contact areas, however, allows us to sidestep many of the drawbacks of mapping global features. For example, a chart used to parameterize contacts on the index finger need not have any influence on a chart used for the middle finger (qualitative semantic boundary), and a chart toward the tip of

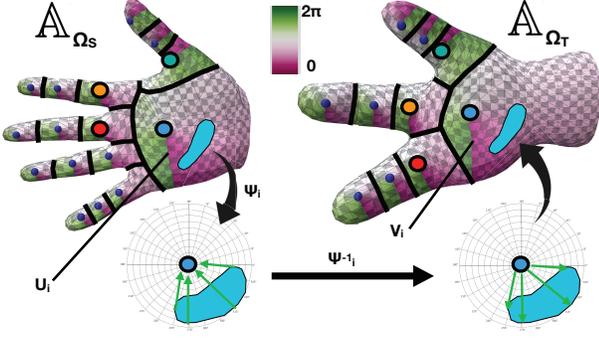


Fig. 8. Cover generated from a set of landmarks (dark blue), chart boundaries (black), and example corresponding landmarks on each surface (red, orange, teal, light blue) obtained from annotation. A sample contact distribution (cyan) within the boundary of chart  $U_i$  is parameterized against the chart's affiliated landmark using logmap transform  $\psi_i$ . Inverting  $\psi_i$  allows the distribution to be reconstructed from the corresponding landmark on  $\mathbb{A}_{\Omega_T}$ , while  $V_i$  determines the location of the contact's new embedding.

a finger need not influence a chart toward the base (quantitative geodesic distance boundary). Quantitative boundaries are particularly useful, because they can be reliably used even when semantic information is missing or unreliable. Figure 7 illustrates an example, where clusters of contact areas between different fingers are quantitatively distinguishable from each other when viewed from a sample reference point on the palm. It is therefore possible to isolate each finger region by introducing a hard discontinuity in the region of separation, allowing for subsequent fine-grained control over the parameters of each chart independently of the others. The atlas produced from the union of such disjoint charts is thus a segmentation of the manifold; however, contacts parameterized by any constituent chart can be fully reconstructed through chart inversion. We select the well-understood logarithmic map (logmap) [Schmidt et al. 2006] as the template for each chart due to its low dimensional parameterization, easy inversion via the exponential map (expmap), and ability to be quickly and accurately computed via heat diffusion [Sharp et al. 2019]. The atlas of the source hand manifold  $\Omega_S$ , termed the *source contact atlas*, is thus formulated as follows:

$$\begin{aligned} \mathbb{A}_{\Omega_S} &:= \{(U_i, \psi_i) : i \in [1, M], \psi_i = \log_q(c)\} \\ \text{s.t. } &\bigcup_{i=1}^M U_i = \Omega_S \quad U_i \cap U_j = \emptyset \quad \forall i, j \in [1, M], \end{aligned} \quad (1)$$

where  $(U_i, \psi_i)$  represents each of the  $M$  constituent charts such that  $U_i$  is the subdomain of  $\Omega_S$  governed by chart  $i$  and  $\psi_i$  is the logmap function that transforms contact  $c$  from barycentric  $U_i$  coordinates into logmap coordinates  $(r_c, \theta_c)_q$  relative to origin  $q$  in the transformed space. We then postulate the atlas of target manifold  $\Omega_T$ , termed the *target contact atlas*, to be of the form:

$$\begin{aligned} \mathbb{A}_{\Omega_T} &:= \{(V_i, \psi_i^{-1}) : i \in [1, M]\} \\ \text{s.t. } &\bigcup_{i=1}^M V_i = \Omega_T, \end{aligned} \quad (2)$$

where  $\psi_i^{-1} = \exp_q c$ , which can be computed by tracing a geodesic originating from  $q$  with parameters  $(r_c, \theta_c)_q$ . Including the disjoint condition is not necessary, because chart assignment is already determined on  $\Omega_S$ . Note also that while segmentations of some contact distributions may not always be clear (i.e., full hand power-grasps), any partition of  $\Omega_S$  is guaranteed to produce a contact point reconstruction on  $\Omega_T$ . Figure 8 illustrates the aforementioned terms and proposed formulation.

Our formulation requires only the ability to compute Laplacians and geodesics, which are both intrinsic quantities. Unlike extrinsic methods that exploit properties related to a geometry's embedding in space (e.g., vertex locations, normals) [Li et al. 2007], intrinsic methods instead only consider the connectivity of the structure. Intrinsic properties are highly advantageous for hands, because such metrics are isometry invariant and therefore allow us to move data between hand geometries without knowledge of either the source or target hand pose.

We start by determining each region  $U_i \in \Omega_S$  using a set of  $M$  landmarks extracted from artist annotation performed once for each new source hand. For convenience, we designate each landmark as the origin ( $q$ ) of each logmap. We next determine the closest landmark to each contact point ( $q^*$ ) using the Vector Heat Method [Sharp et al. 2019] and MMP [Mitchell et al. 1987] to extract the logmap coordinates of each contact point  $(r_c, \theta_c)_{q^*}$ . Importantly, automatically determining  $q^*$  means that we *do not require* contact data to be annotated or strictly associated with a pre-determined individual or grouped set of landmarks as mandated in previous work [Lakshminpathy et al. 2023] while also ensuring that the set of contact points for each source chart is unique. Notably, while any partition is suitable under our formulation, our approach of taking the closest landmark to each contact is special in that it generates an equivalent point partition to that of a geodesic Voronoi segmentation [Herholz et al. 2017] with the added benefit of providing an exact method of reconstruction.

However, rather than requiring landmarks to be provided as individual points, we instead adopt a curve-based input approach [Gehre et al. 2018]. We select axial curves [Lakshminpathy et al. 2023] as the annotation implementation. Axial curves contain the following:

- (1) a finite set of points  $\{a_1, \dots, a_n\} \in \Omega$  that in our case serve as landmarks, with a *shortest geodesic*  $g_i$  connecting each pair of adjacent points  $(a_i, a_{i+1})$  for  $i = 1, \dots, n-1$ ;
- (2) *turning angles*  $\{\phi_i\}_{i=2}^{n-1}$ , where each  $\phi_i$  is the angle of rotation from the ending direction of  $g_{i-1}$  to the initial direction of  $g_i$ , expressed in the tangent space of  $a_i$ .

We found that axial curves enabled our artist to easily generate and tweak multiple landmarks via simplified control handles, thereby greatly reducing annotation time. Axial curves also substantially reduced the need for careful landmark placement, automatically resolved the annotation overhead of consistently orienting the logmap zero angle of all constituent axis points, simplified the process of designating corresponding consistently oriented landmarks on  $\Omega_T$ , and provided a representation that enabled straightforward modification of expmap reconstruction parameters (discussed shortly). In practice, we found that even simple collections



Fig. 9. Illustration of contact transfer between two widely varying hand shapes. An isometry-preserving-as-possible transfer ( $\lambda_A = \lambda_S = 1$ ) results in undesirable squishing of contacts on the target domain that fails to capture semantic equivalence between fingers. Altering  $\lambda_A$  and  $\lambda_S$  fixes the problem.

of curves (Figure 10) easily obtainable from an artist yielded remarkably effective contact atlases capable of transferring contact areas reliably.

In typical cases, corresponding axial curves can be easily drawn by an artist; however, there are several interesting alternate cases. In the case of  $\Omega_T$  having fewer fingers than  $\Omega_S$ , such as the three-fingered alien hand illustrated in Figure 8, it is possible to safely discard unwanted contact groups by setting any  $V_i = \emptyset$  or, equivalently, removing unwanted  $U_i$ . By doing so, contacts that would originally be bound to the removed  $U_i$  regions will instead be bound to the nearest available  $U_i$ . Such contacts are typically unwanted (e.g., contacts on the human ring finger or pinky) and can easily be filtered out by a geodesic distance threshold  $\mathcal{E}_g$  due to their significantly further distances from their respective new landmark bindings. Similarly, hands with additional fingers do not necessarily require all fingers to be used—the chart nearest the unused finger simply extends to the region, ensuring a cover of  $\Omega_T$  is maintained.

To address variations in finger or palm shape, we introduce an expmap scaling metric  $\lambda_s(\theta)$  and reformulate  $\psi_i^{-1}$  to trace geodesics of the form  $(\lambda_s(\theta)r, \theta)$ . This modification effectively permits “deforming” contact distributions parameterized on  $\Omega_S$  into any shape on  $\Omega_T$ , although in practice we found that uniform scaling was largely sufficient. Although such a parameter breaks the assurance of the expmap providing an isometry-preserving-as-possible reconstruction of contacts embedded by  $(U_i, \psi_i)$  on  $\Omega_T$ , as illustrated in Figure 9, we found that such distortion is often desirable to meaningfully capture *semantic* similarity between geometric variations.

Finally, we address finger length variations by altering geodesic lengths from landmarks and by extension the desired locations of  $V_i$ . We therefore introduce one more parameter  $\lambda_a$  that can be used to extend or contract the length of the geodesics connecting the axial curve points on  $\Omega_T$ . As also illustrated in Figure 9, the combination of  $\lambda_s$  and  $\lambda_a$  enables fine-grained, predictable adaptation of contact distributions across even widely varying geometries. Importantly, the flexibility to perform such alterations is made possible by the source and target contact atlases.

We obtain a corresponding set of landmarks on both  $\Omega_S$  and  $\Omega_T$ , as well as hyperparameters  $\lambda_a$  and  $\lambda_s$ , from one-time artist annotation. We implement and build on tools from existing published work [Lakshmipathy et al. 2023] to facilitate the process and detail the annotation procedure in Appendix A. Contacts across entire manipulation time series are subsequently procedurally transferred between the source and target hand. Figure 10 illustrates the received annotations and a representative sample transfer.

### 3.4 Motion Retargeting

We perform motion reconstruction for the target hand in three steps using a consistent objective formulation: (1) estimation of an initial trajectory for the target hand, (2) refinement of the initial estimate, and (3) construction of the final trajectory through B-Spline fitting.

**3.4.1 Objective Formulation.** We use a simple and common objective formulation in all steps for computing solutions per frame:

$$\begin{aligned} \theta_f^* &= \arg \min_{\theta} \lambda_m \Gamma_M + \lambda_c \Gamma_C + \lambda_t \Gamma_T + \lambda_j \Gamma_J \\ \text{s.t. } & \theta_L \leq \theta \leq \theta_U, \end{aligned} \quad (3)$$

where  $\Gamma_M$ ,  $\Gamma_C$ ,  $\Gamma_J$ , and  $\Gamma_T$  are the penalty terms,  $\theta$  is the DOF vector,  $\theta_L$  and  $\theta_U$  define the lower and upper bounds of  $\theta$ , respectively, and  $\lambda_c$ ,  $\lambda_m$ ,  $\lambda_j$ , and  $\lambda_t$  are weighting hyperparameters.

We next describe each penalty term. The marker penalty ( $\Gamma_M$ ) encourages aligning source hand and target hand virtual markers at frame  $f$ . Assuming  $M$  virtual marker points,  $\Gamma_M$  is defined as:

$$\Gamma_M = \sum_{m=0}^M \Gamma_{MD,m}, \quad (4)$$

where  $\Gamma_{MD,m}$  represents the  $L_2$  distances between corresponding marker points. The contact penalty ( $\Gamma_C$ ) encourages aligning target hand and object contacts at frame  $f$ . Assuming  $C$  contact points,  $\Gamma_C$  is defined as follows:

$$\Gamma_C = \sum_{c=0}^C (\lambda_{cd} \Gamma_{CD,c} + \lambda_{cn} \Gamma_{CN,c}), \quad (5)$$

where  $\Gamma_{CD,c}$  represents the  $L_2$  distances between corresponding contact points,  $\Gamma_{CN,c}$  penalizes deviation from surface normal inversion at the contact points, and  $\lambda_{cd}$  and  $\lambda_{cn}$  are weighting hyperparameters. The table penalty  $\Gamma_T$  discourages hand-table intersection. Assuming  $S$  sampling points on the target hand,  $\Gamma_T$  is defined as follows:

$$\Gamma_T = \sum_{s=0}^S \max(0, -\Gamma_{SD,s}), \quad (6)$$

where  $\Gamma_{SD,s}$  represents the **signed distance function (SDF)** of the table evaluated at the location of point  $s$ . For simplicity, we use the vertex set of the target hand or its affiliated manifold wrapper as  $S$ . We also assume a box geometry for the table, whose SDF can be computed analytically.<sup>2</sup> Last, the “prior” penalty ( $\Gamma_J$ ) serves as a regularizer against either the default rest pose or the previously

<sup>2</sup><https://iquilezles.org/articles/distfunctions/>

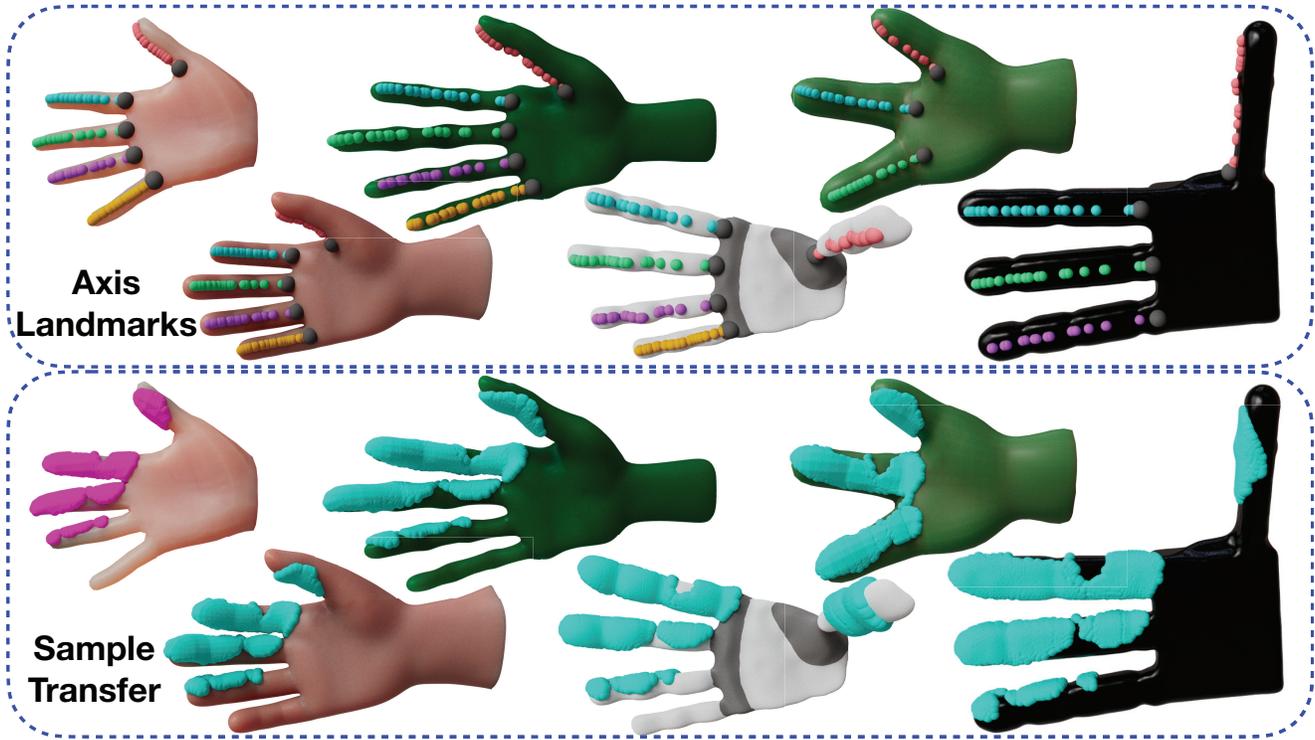


Fig. 10. One-time axis annotation (top) enables scalable retargeting of original contacts (magenta) to customized configurations per target hand (cyan). Axis colors indicate matching annotations.

existing keyed value at frame  $f$ . Assuming  $J$  DOFs, we obtain the following:

$$\Gamma_J = \sum_j^J \Gamma_{P,j}, \quad (7)$$

where  $\Gamma_{P,j}$  represents the deviation between DOF  $j$  and its existing value. We use the Method of Moving Asymptotes [Svanberg 2002], a local gradient-based solver, implemented in the NLOPT library [Johnson 2017] to compute solutions to Equation (3). All gradients are computed via a finite difference.

**3.4.2 Initial Trajectory Estimation.** Because no baseline data are available for the target hand motion, we must synthesize it from scratch. We do so by first estimating a per-frame optimal initial trajectory estimate over  $F$  total motion frames, which we define as follows:

$$\Theta^* = \{\theta_0^*, \theta_1^*, \dots, \theta_F^*\}. \quad (8)$$

Importantly, the above formulation entails that each frame of the optimal trajectory *is independent of the estimates of its neighbors*, which runs counter to many existing works that add explicit conditioning terms on the estimate of the previous pose [Handa et al. 2019; Qin et al. 2021; Sivakumar et al. 2022]. This crucial distinction proved vital in mitigating error buildup over long sequences, which caused the estimates of frames later in the motion to converge to undesirable local minima. Independent estimation also allowed us to reliably prune poor local optima during later processing stages.

We perform the trajectory estimation in two passes. We first solve for only the root joint position per frame. We then solve for the full pose per frame using the root estimate as the seed. Default poses for all hands are illustrated in Figure 1.

**3.4.3 Trajectory Refinement.** We next refine the estimated trajectory to improve temporal consistency using acceleration as the smoothing metric. To do so, we impose a threshold  $\mathcal{E}_{acc}$  and remove all frames that violate the threshold. We then replace each violated frame with a linear interpolation between its nearest valid left and right neighbors and re-solve Equation (3) to obtain more temporally consistent estimates that are. Frames are removed on a per-DOF basis and in a single pass. We impose the same threshold across all DOFs in our results regardless of joint coupling (e.g., codependence between ball joint DOFs) or units (e.g., angular vs. linear DOFs) in support of standardization.

However, updating violations can introduce new violations in the resulting trajectory. We thus perform the refinement procedure iteratively until either until no further violations are found or the number of iterations exceeds a maximum cap. All unresolved violations at the end of iteration exhaustion are ignored during fitting.

Applying this refinement strategy to the raw initial trajectory slowed the method considerably due to high numbers of violations resulting from independent pose computation per frame. Instead, following precedent [Handa et al. 2019; Qin et al. 2022; Sivakumar et al. 2022], we found it beneficial to re-solve for the estimates after applying low-pass and peak removal filters. Preprocessing

substantially reduced the number of acceleration violations, and thus the overall computation time required for this stage.

**3.4.4 B-Spline Fitting.** Finally, we use the refined frame solutions as sample points and fit a cubic B-Spline to each DOF across the computed time series, ensuring our final solution guarantees  $C^2$  continuity. We designate a fixed number of control points and solve simultaneously for both values and locations in time using least squares pseudoinverse approximation [Eberly 2005]. This representation also permits artist control over the fitted spline, and better approximations, at the cost of possible overfitting, can be easily obtained either by increasing the number of control points or hierarchically compositing splines to reduce error residuals [Lee and Shin 1999]. Notably, we found that most motions could be represented with relatively few control points.

## 4 Results

We performed a number of experiments to both qualitatively and quantitatively evaluate our method. In addition to evaluating our thirty retargeted tasks, we also assess our method’s effectiveness across a bimanual task, hand size variations, cyclic motion retargeting, two baseline comparisons, and two ablations.

### 4.1 Qualitative Evaluation

Results for our 30 retargeted demonstrations can be viewed in the supplementary video. Result computation times ranged from 4 to 22 hours when run on a single Intel Xeon W-1250 3.3-GHz processor without parallel threading or GPU acceleration. Following our commitment to standardization, we use a uniform set of optimization hyperparameters across all hands and motions, the same transfer coefficients for each hand across all motions, and an identical number of control points per motion across all hands and DOFs. We note that no parameter required careful tuning, following our commitments to reliability and simplicity. We also did not modify the generated DOF spline motion curves in any of our examples. Parameter values for all hands, transfer coefficients, optimization weighting coefficients, and control point counts are tabulated in Appendix B. Overall, we found the retargeted trajectories to be of generally high-quality despite blanket standardization. Such generalization is particularly notable, because results confirm that parameterization of contacts across arbitrary motions is possible *without knowledge of distributions in advance* and without customization of landmarks to individual distributions.

### 4.2 Quantitative Evaluation

**Distance Metrics.** Figure 11 quantifies the average marker and contact distances generated by our method across all results. As can be observed, average contact distance during segments of high contact density are reasonably stable across all hands and tasks. In contrast, contact distance tends to be higher in low density regions and spikes at region boundaries due to the competing table penalty (Equation (6)). Marker distances, in contrast, appear elevated during high contact density regions due to the competing, and typically dominant, contact penalty term. Nonetheless, distances for both quantities are typically no more than a few centimeters. Unsurprisingly, the Allegro Hand appears to perform the worst due to its large size and limited DOFs.

**Intersection Metrics.** Figure 12 quantifies the total amount of hand-object, self, and table intersection generated by our method across all results. To compute these quantities, we first determine all penetrating hand vertices via raycasting as detailed in Section 3.2. The resulting points are then clustered via depth-first search on the hand mesh. Finally, we compute the convex hull of each cluster and extract its volume. Total intersection is defined as the sum of all such hull volumes relative to the total volume of the hand, expressed as a percentage. All reported percentages are overestimates due to convex hull approximation. Despite not performing any hand-object or self-intersection resolution, we observe that intersection volumes are nonetheless low. Our results indicate that contact areas, in part due to their implicit encoding of natural grasp states, are viable as a cheap approximation of both physical motion and intersection avoidance in the absence of a full physics simulation.

### 4.3 Bimanual Manipulation Retargeting

Our method can trivially be extended to bimanual manipulations via superpositioning of independently computed left- and right-hand trajectories. We demonstrate this capability through retargeting a game controller manipulation task, which can be viewed in the supplementary video. However, our method is currently limited to instances where hands interact only with the object. We do not anticipate a successful result for hands interacting with each other, because the state of the contacts will vary with the state each hand during the optimization process. Resolving the resulting cyclic dependency is an interesting problem for future work.

### 4.4 Evaluation across Hand Size Variations

We stress tested our method against variations in hand size to determine whether identical contact distributions can be re-used. We selected dimensions representative of the fifth percentile of 7-year-old female children [Malina et al. 1973] and the 95th percentile of adult males [NASA 2000] as variants. Figure 13 illustrates the hand size variations. A complete result for the stapler task is available in the supplementary video. We found that the reconstructed motions to be of high quality, indicating that our method is capable of supporting hand scale variations without the need to re-map contact distributions.

### 4.5 Cyclic Motion Retargeting

We further stress tested our approach through an additional experiment of cyclic retargeting. We select the phone task and witch hand as the intermediary, which we found to be the most challenging task and most difficult five-fingered hand to work with respectively. To do so, we first retargeted the original motion to the witch hand. We then used the witch hand virtual markers as references for tracking the new MANO hand motion. We used the original contacts on the MANO hand, because the contact atlas for the witch hand was generated via the fully invertible expmap transform from the MANO hand. The reconstructed motion can be viewed in the supplementary video. Figure 14 illustrates the motion curve differentials between our cyclically retargeted result and the original motion. We observed that strong alignment between the original and intermediate hand roots (orange, green,

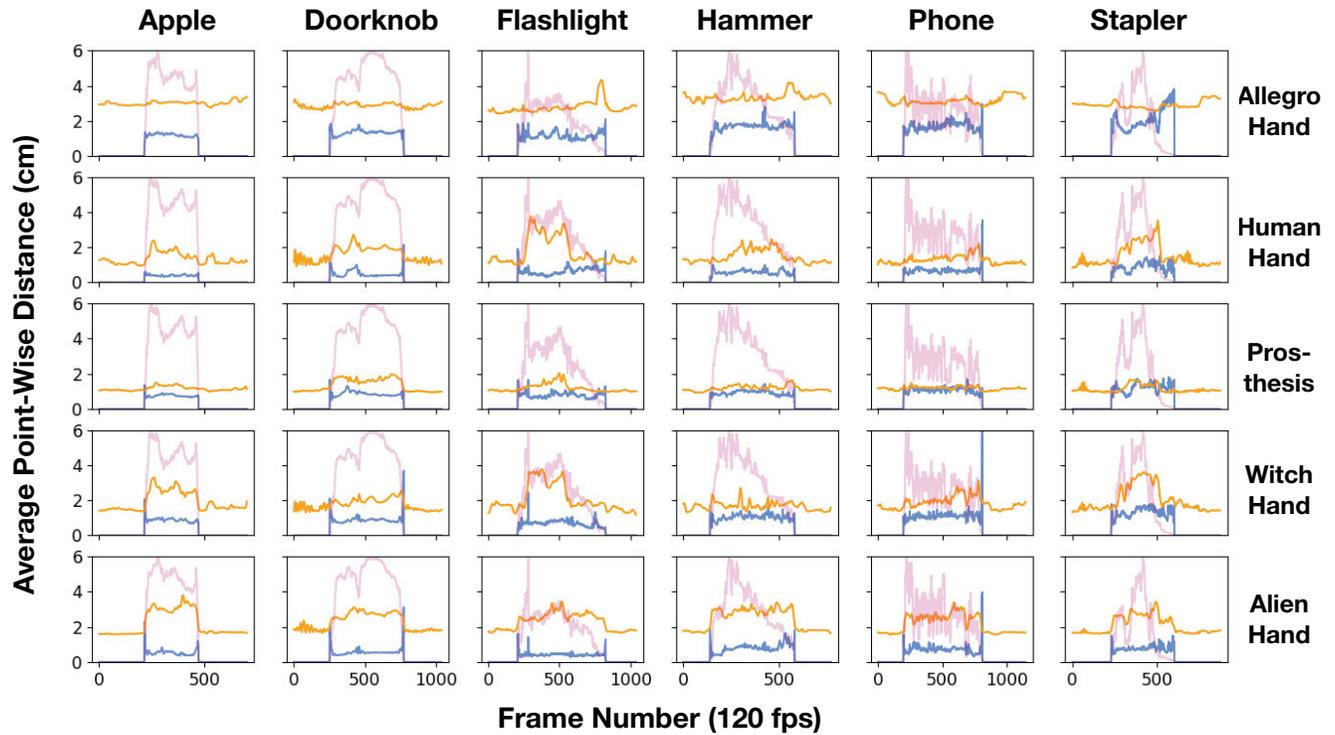


Fig. 11. Plots of all average marker (orange) and contact (blue) distances. (Light pink) The number of contact points per frame, normalized across the max number of contacts per task and scaled by the maximum chart distance value, is also overlaid for reference.

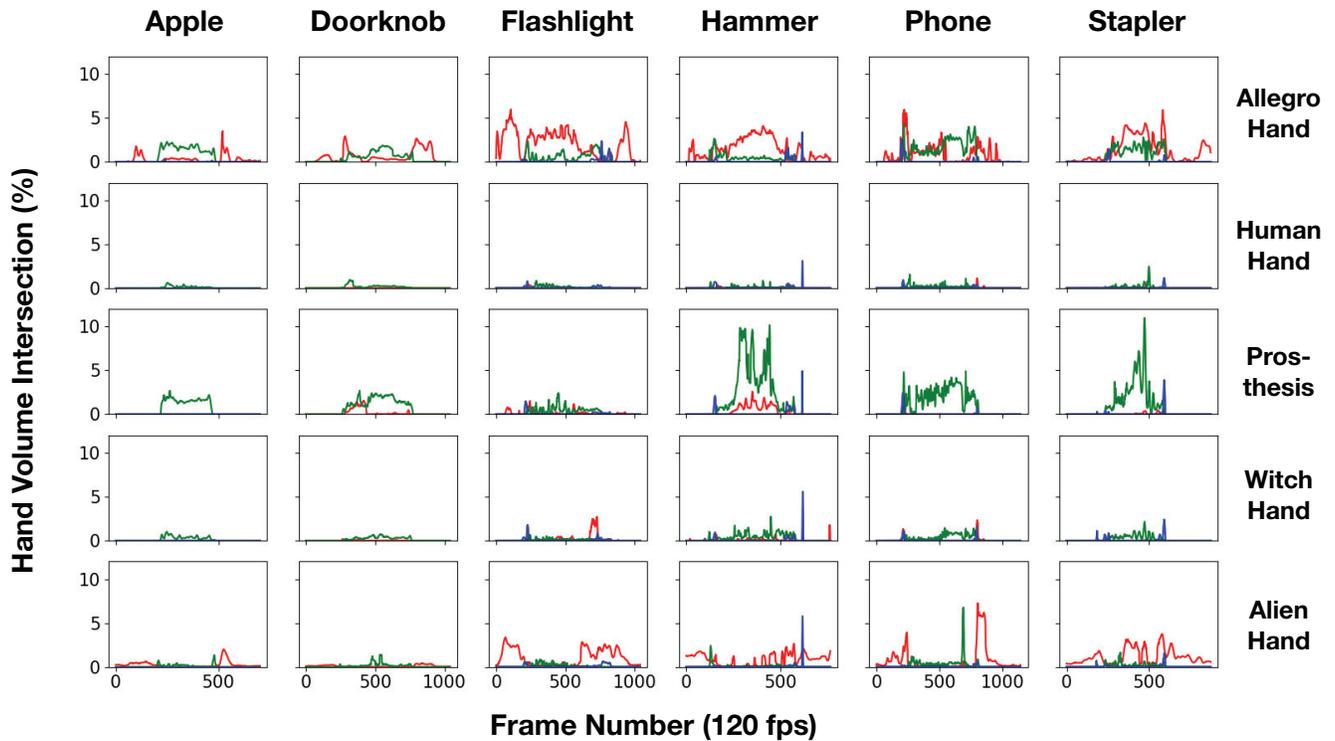


Fig. 12. Plots of all hand-object (green), self (red), and table (blue) intersection.

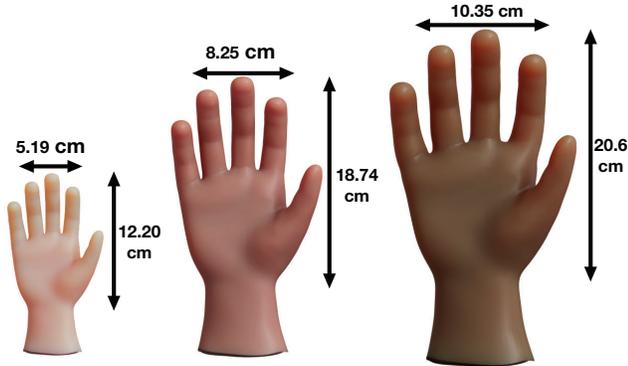


Fig. 13. Different human hand sizes used in our evaluations, including the original human hand (center), a small hand representing the fifth percentile of 7-year-old female children (left), and a large hand representing the 95th percentile of adult males (right).

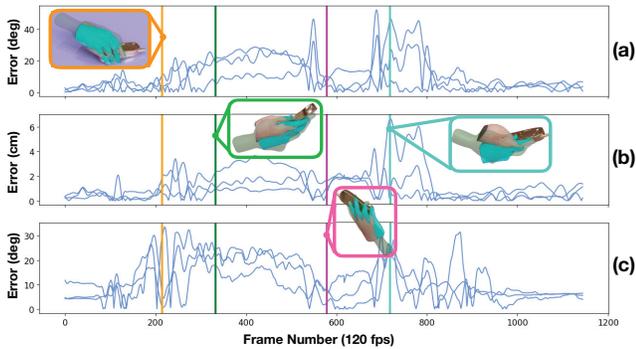


Fig. 14. Reconstructed motion curve error differentials between our cyclically retargeted result and the original motion, including the root rotation (a), root translation (b), and a representative finger joint (c). Rotations are expressed as Euler angles following the same order as the original motion. Colored boxes illustrate alignment between the cyclically retargeted hand (skin-colored), original hand (cyan), and intermediate witch hand (green silhouette) at various stages of manipulation.

magenta) typically resulted in high-quality cyclic reconstructions, while weaker alignments (turquoise) resulted in deterioration. Discrepancies between finger DOFs had relatively minor qualitative impact on results compared to the root (orange). Figure 15 provides an alternate evaluation by instead comparing contact and marker distance metrics. These errors are generally small, with the contact error routinely being under 1 cm outside the first and last few frames. Spikes in contact errors at the boundaries are similarly induced by large divergences in the witch hand motion from the MANO reference, which subsequently leads to unreliable virtual marker guidance during cyclic motion construction.

#### 4.6 Baseline Comparisons

We next validate the importance of contact information by comparing our approach against two existing contact-free methods used in tele-operation: fingertip keypoint tracking [Dasari et al. 2023; Qin et al. 2022] and whole-hand relative keyvectors [Handa et al. 2019; Sivakumar et al. 2022]. We use the Allegro Hand for consistency with the baselines. To isolate the objective formulation,

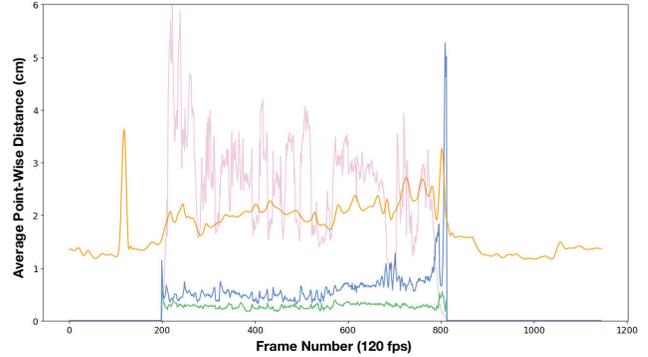


Fig. 15. Reconstructed average marker (orange) and contact (blue) distance errors between the original motion and our cyclically retargeted results. Average contact error for the original motion (green) and contact density (light pink) curves are also provided for reference. Marker error for the original motion is always zero.

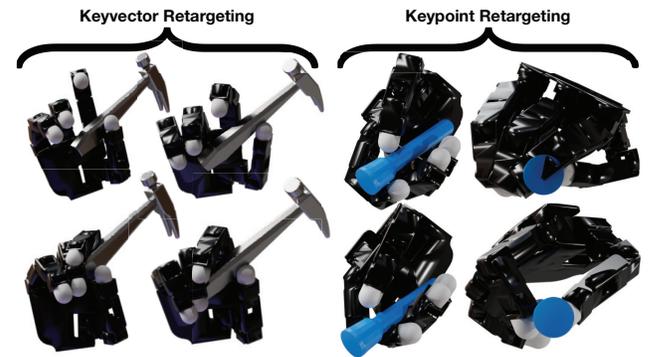


Fig. 16. (Top row) Poses computed using keyvectors or keypoints visibly struggle encouraging hand-object contact and motion alignment especially when source and target hand geometries greatly diverge. (Bottom row) Using contact areas greatly reduces such artifacts.

Table 1. Aggregate Statistics of Average Contact and Marker Distance Error Metrics across the Entire Task Suite

	Baseline Distance Metric Comparisons (cm)			
	Marker		Contact	
	Med.	$\mu \pm \sigma$	Med.	$\mu \pm \sigma$
Keypoint	<b>0.549</b>	<b>0.854</b> $\pm$ 0.810	3.22	3.2250 $\pm$ 0.607
Keyvector	4.705	4.675 $\pm$ 0.766	4.773	5.026 $\pm$ 1.478
Ours	2.967	2.996 $\pm$ <b>0.300</b>	<b>1.438</b>	<b>1.511</b> $\pm$ <b>0.429</b>

Lower values are better. Bold values indicate the best statistics.

we augment both methods with information about the known full source trajectory. Specifically, we provide access to ground-truth root joint estimation, independent pose computation per frame (Section 3.4.2), and retroactive smoothing of the whole sequence (Section 3.4.3).

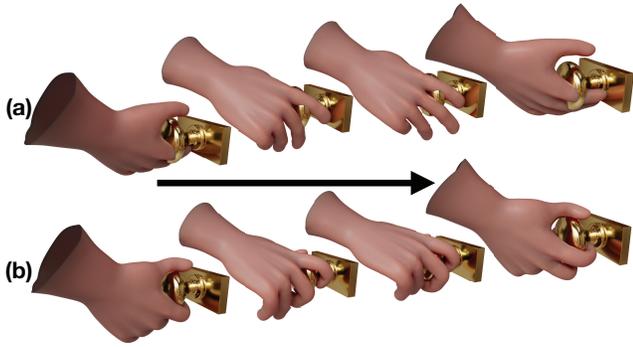


Fig. 17. Sample motion frames (a) without and (b) with contact information during a doorknob manipulation.

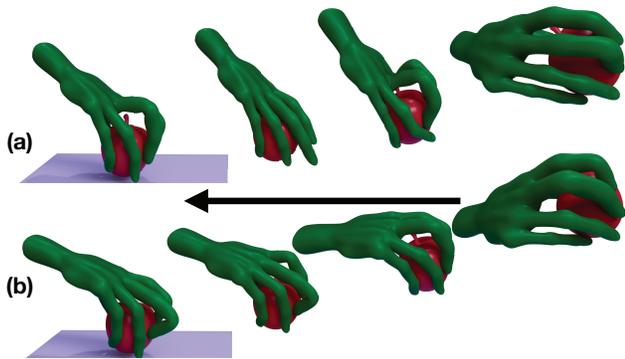


Fig. 18. Example (a) without and (b) with root preconditioning.

**Fingertip Keypoint Tracking.** We select four keypoints at the tips of each Allegro finger to track the human thumb, index, middle, and ring fingertip positions. The corresponding Allegro hand pose is computed using an existing optimization formulation [Qin et al. 2022], which is equivalent to Equation (3) with  $\lambda_C = 0$ .

**Whole-hand Relative Keyvectors.** We next consider the strategy of generating target hand poses using pairs of keypoints (i.e., keyvectors) [Handa et al. 2019; Sivakumar et al. 2022]. Specifically, we use the objective formulation proposed by DexPilot [Handa et al. 2019] as the basis for pose computation. Exact keyvectors and detailed term explanations are available in the cited work.

Full trajectory comparisons to both methods are available in the supplementary video. As illustrated in Figure 16, we observe that both baseline methods, while capable of producing smooth trajectory estimates, generally struggle to make contact with the object and produce nontrivial motion misalignment artifacts. Table 1 tabulates the performance of all methods across the entire task suite using average contact and marker distance error metrics. Because keypoint tracking only optimizes for marker error and tracks only fingertip positions, the overall marker error is less than that of our method; however, doing so comes at the cost of contact distance, which we argue results in comparatively worse artifacts qualitatively. It is also interesting to observe that our marker error standard deviation is significantly better despite the lower median of keypoint tracking, which indicates that dense virtual markers

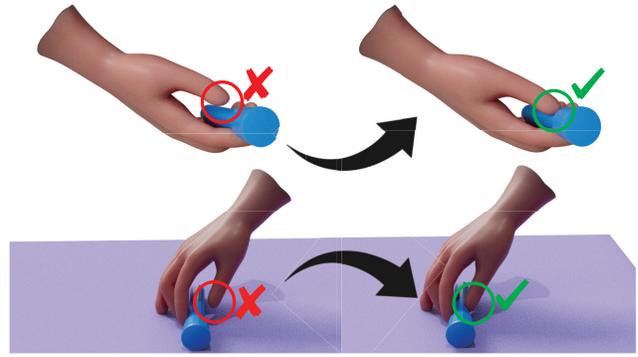


Fig. 19. (a) The original design candidate thumb is unable to reach the flashlight switch while (b) the revised candidate thumb length adequately closes the gap in all such frames.

tend to be more consistent than single-point tracking. Both our method and keypoint tracking, however, are superior to keyvector tracking in relation to both metrics by substantial margins.

Because both of these techniques have been used for retargeting in teleoperation scenarios, a person in-the-loop can interactively correct for errors such as those shown in Figure 16. In contrast, the approach presented in this article provides results with good contact without the need for such intervention.

#### 4.7 Ablations

**No Contact Data.** We perform an ablation against a pure marker tracking pipeline to test the importance of contact information. We select the alternate human hand for simplicity and perform the ablation by setting  $\lambda_c = 0$ . We examine results after per-frame full pose estimation, because large divergences in behavior are already apparent. Figure 17 provides several comparison frames. Notably, the hand lacking contact information fails to conform to the doorknob at numerous points of the manipulation, resulting in a motion estimate that is poor at the outset. We note that this failure was compounded in the case of less humanlike hands, where marker estimates were less reliable. The dramatic difference in motion reconstruction, even at a coarse level, clearly indicates that marker tracking information alone is not sufficient for motions involving rich object interaction—contact information is essential.

**No Root Preconditioning.** We perform an additional ablation to examine the necessity of computing root transforms as a preprocessing step. Figure 18 illustrates the outcome for a representative sequence of motion frames. Notably, omission of root preconditioning creates instances of root “drift,” in which locally optimal solutions in subsequent frames are found by moving other joints in the kinematic chain. The root joint is typically only updated when the accumulated marker error buildup becomes large, which results in undesirable sudden jumps in translation. We found results to be similar regardless of the hand, object, or motion and argue that it is because the root transformation has a disproportionately large influence over gradients. A poor starting root transform can thus bias solutions toward undesirable local minima that can only be escaped from when marker error buildup becomes substantial.



Fig. 20. (a) A typical three-finger axis mapping and its resulting pose compared to (b) an unusual “Spock”-like configuration achieved by assigning multiple curves to a single finger.

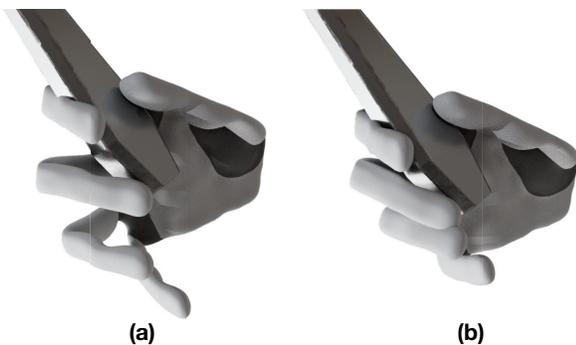


Fig. 21. Frame comparison of the prosthetic with (a) one DOF and (b) two DOFs at the proximal phalange joint.

Solving for the root transformation first directly combats this issue by reducing problem dimensionality.

## 5 Extensions

We next discuss two extensions of our method beyond hand motion retargeting: visualizing hand design choice impacts over full trajectories and retargeting demonstrations to different objects.

### 5.1 Visualizing Design Choice Impacts

Our method enables task-specific visualization of design choices as well as insights into how such parameters can be adjusted. For example, determining appropriate finger lengths is a common problem in both rigging and rapid prototyping. Figure 19 illustrates an example in which the original thumb is not able to reach the object when the source contact distribution is used, while the revised candidate is able to do so. Another such example is testing alternate finger mappings. As illustrated in Figure 20, it is possible to change the three-fingered hand’s interaction with the phone by assigning multiple axial curves to a single finger, effectively semantically mapping contacts from two human fingers (e.g., index and middle) to a single alien finger. Such flexibility is particularly useful for hands with different finger counts or other morphological

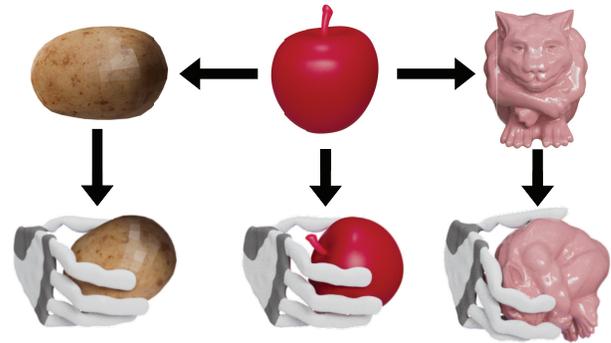


Fig. 22. Our method can be extended to accommodate retargeting generic motions between different objects. Hand grasps successfully make subtle, but important adjustments to adapt to simple shapes (left) as well as more dramatic adjustments for more complex features (right).

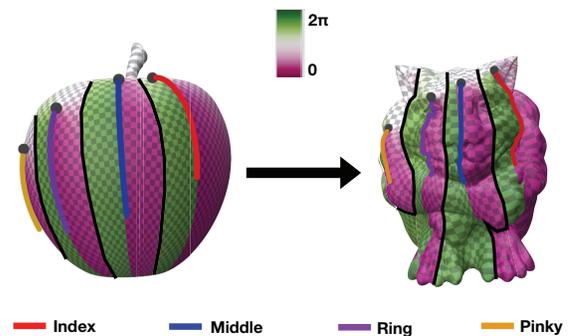


Fig. 23. Approximate atlases used to retarget contacts of a handoff manipulation between a simple apple and a more geometrically complex gargoyle figurine. Axial curves corresponding to each finger are denoted by different curve colors, while black spheres denote the start point of each curve. Curve placements on the target surface can be used to implicitly modify grasps with respect to the surface to any extent desired.

differences. Full trajectories for both examples are in the supplementary video.

A third example is using our method to visualize the impact of DOFs. Because our method does not make assumptions about the kinematics of the system, it is possible to modify the underlying skeleton and automatically recompute the entire motion sequence. This functionality is useful for rapid prototyping of new hand designs or selecting the best hand for a particular task set [Bauer et al. 2022], especially for driving decisions such as adding or removing motors. We illustrate this capability in a case study of the prosthetic hand. We start by assuming 1-DOF knuckle joints at each finger controlling flexion and extension and perform a comparison against a 2-DOF knuckle joint variant capable of adduction and abduction. Figure 21 illustrates a representative frame of noticeable difference in computed solutions during the hammer task. Under the same settings, the 1-DOF variant generates intermediate solutions containing physically implausible hammer grips. In comparison, the 2-DOF variant provides a more realistic grip. These observations suggest the 1-DOF variant does not possess the dexterity required to manipulate the tool in an anthropomorphic manner

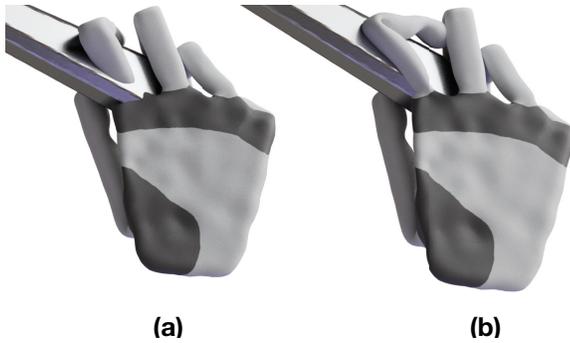


Fig. 24. When optimized from the same start position with the same marker and contact distribution targets, (a) a 2-DOF MCP joint converges to a solution that penetrates through the hammer while (b) a 3-DOF MCP joint is able to conform to the hammer’s handle with much less intersection. We found that the torsional DOF was utilized in this particular grip across all hands that included a 3-DOF MCP joint.

and that including a second motor per finger might be worthwhile. Our method can thus be used to evaluate task-centric dexterity mid-manipulation, which would otherwise be challenging to do manually.

## 5.2 Object Substitution

Our method also enables retargeting generic manipulations to different objects. Contact sets can be generated on new objects through any means of correspondance, including by reversing the raytrace process from the source hand, tracing from the original object, through diffusion models [Wei et al. 2023], or, perhaps most interestingly, by using our contact atlas approach. The remainder of the pipeline is subsequently applied normally to produce the final result. We illustrate examples in Figures 22 and 23, and the supplementary video of re-purposing an existing apple “handoff” manipulation to a potato, where contacts on the potato are generated by tracing the original contacts outward from the apple, and a more geometrically complex gargoyle sourced from the Thingi10K dataset [Zhou and Jacobson 2016], where contacts are instead retargeted using a contact atlas between the apple and the gargoyle.

## 6 Discussion

We next provide additional analyses of our results and a comparison of our framework against learning-based retargeting methods.

### 6.1 Assessment of Results

We were pleasantly surprised to find that high-quality retargets were possible even with a uniform set of parameters across all hands, although certain hands were comparatively easier to retarget than others. Surprisingly, the three fingered alien hand was the easiest to work with despite its unusual morphology. Its comparatively thick size was highly beneficial for limiting pose estimate uncertainty with respect to both markers and contacts. The only noticeable drawback was higher relative self-intersection compared to other hands; however, this behavior was unsurprising, because the large majority of these intersections were caused by overlap between the second and third phalange, which were mapped to

to the index and middle finger of the human hand—two fingers that commonly moved together in close proximity within the task suite we examined. In contrast, despite its more anthropomorphic appearance, estimations of the prosthesis wrist position were comparatively far noisier due to its thin profile. The prosthesis also exhibited relatively high hand-object intersection for the hammer and stapler tasks, which we found could be attributed to its small size and limited DOFs. In the case of the stapler, the prosthesis had difficulty curling fingers cleanly around the entire object. In the case of the hammer, as illustrated in Figure 24, the lack of a torsional metacarpophalangeal (MCP) DOF caused the index finger to penetrate through the hammer rather than wrapping around the handle. The witch and Allegro hand were the most difficult to retarget due to their unusually long finger lengths, which we found to be one of the most significant factors in determining retargeting complexity between hands.

However, despite the Allegro hand’s general difficulty in retargeting, our method was still able to produce remarkably high-quality solutions. As demonstrated in recently published work [Lakshmiathy and Pollard 2024], retargeted kinematic trajectories can be utilized as high-quality priors for building downstream physics-based controllers. The cited work even demonstrates that such controllers can be synthesized *online in real time* through simple predictive sampling in a local search space.

Additionally, although we focus on hands in this article, we showed that our shape matching approach is general and can be used to precisely define transfer of arbitrary local data between shapes. Additional applications include defining complex interactions between surfaces (e.g., between two deformables) or porting assets from one surface to another. We are interested in exploring additional extensions of our approach to broader contexts in the future.

Finally, we notably did not receive any hand pairs following an area-area virtual marker configuration; however, because our goal in this work was to examine a wide variety of dissimilar hands, we suspect that the annotator’s decision to use only many-to-one configurations is largely a result of experiment specifics. Even the retargeted human hand we used had different finger length sizes than the source. Retargeting between human hands of similar proportion or highly anthropomorphic robot hands (e.g., the Shadow Hand<sup>3</sup>) may provide a more compelling use case for area markers.

### 6.2 Comparison with Machine Learning Approaches

Our straightforward contact-driven motion retargeting process contrasts with many modern retargeting approaches from the full-body character animation literature, the large majority of which are learning-based. We include a brief discussion of such approaches and how we position our work relative to them.

A major advantage of our method is agnosticism to each hand’s underlying skeleton. Previous approaches have demonstrated the ability to generalize over skeleton variations between different characters in terms of bone lengths [Won and Lee 2019; Zhang et al. 2023b, a] or entire kinematic chains [Aberman et al. 2020; Reda et al. 2023]. Our work falls into the latter category. The latter two cited works, however, either assume that retargeted skeletons

<sup>3</sup><https://www.shadowrobot.com/dexterous-hand-series/>

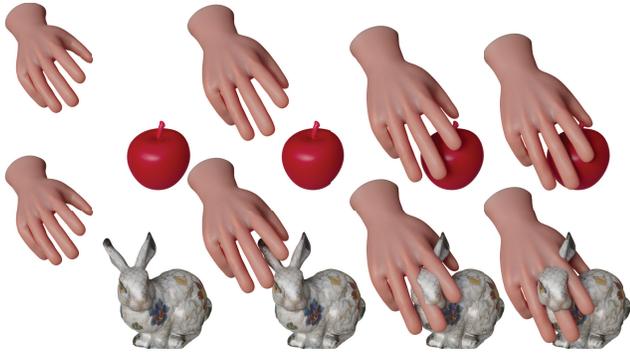


Fig. 25. Our method does not adapt to object substitution instances that require significant pregrasp or release adjustments. (Top) The retargeted approach for the original apple object produces a clean trajectory; however, (bottom) our method fails to accommodate geometries with widely differing features, because no contact data are available during these segments. Any smoothed transitions made during final reconstruction within these segments is purely incidental and a consequence of interpolation rather than a response to the underlying problem.

are homomorphic in kinematic chain endpoints [Aberman et al. 2020] or requires far more computationally expensive reinforcement learning [Reda et al. 2023]. To put reinforcement learning requirements into perspective, Zhang et al. required *3–9 days of training on a cluster of 640 CPUs* [Zhang et al. 2023a] while one of our most expensive retargets (prosthetic hand  $\times$  doorknob task) required 20.57 hours of single-thread processing on one CPU, even accounting for an inefficient front end and finite-difference gradient computations. We also discuss methods to improve computation time in Appendix C. Our method thus offers a more direct approach and is comparatively much faster while requiring far less computational resources.

Another broad category of learning-based approaches in cross-skeleton retargeting involves discovering mappings between either source and target character motions in a joint embedding space [Choi et al. 2020; Li et al. 2023] or a set of reference poses [Kim et al. 2022; Rhodin et al. 2014]. The former approach assumes that motion data for the target embodiment already exist, while the latter assumes that a small number of representative poses is sufficient to capture the entire space of possible mappings. But in this work, as is common in real-world settings, the hand assets are custom and therefore have no existing motion data. Additionally, the space of possible manipulations is massive—it is unclear if a small number of representative poses will sufficiently generalize over such a space. Our retargeting method, in contrast, is *motion agnostic* and *assumes that no motion data for the target hand is available*, which we argue more closely captures the limitations of real-world settings.

In summary, the primary benefits of our approach are its limited target embodiment assumptions with respect to both structure (skeleton, geometry, etc.) and mechanics (prior motion, operation space, etc.), direct and intuitive approach, and minimal computation overhead requirement in comparison to ML-based approaches. We acknowledge that it is possible, and even probable, that many aforementioned techniques can offer compelling solutions for some of our hands and tasks; however, we maintain that

our approach offers a viable alternative to situations where method assumptions or computational overhead can limit adoption.

## 7 Drawbacks and Limitations

While effective in retargeting motions across a wide range of hands, objects, and manipulations, our method contains several limitations.

The atlas generated by our shape matching approach is notably discontinuous and nondifferentiable, making it unsuitable for gradient computations or transfer of global media such as smooth functions or textures. Although these properties were not critical to our problem scope, formulating a  $C^\infty$  atlas can potentially enable a broader range of useful applications.

Our method does not guarantee a retargeted solution free of artifacts. Common artifacts included wobbles resulting from leftover trajectory estimation noise or under/over fitting splines. However, both of these artifacts can generally be resolved by tweaking the acceleration cutoff for trajectory refinement or adjusting the number of B-Spline control points. We found that a fixed set of parameters was suitable for most motions, but note that further refinement is possible with per-hand-per-motion customization if desired.

Our method sometimes has difficulty resolving table intersections in a manner that appears natural, which can result in uncanny finger contortions during object pickup and release. This issue arose in the flashlight and stapler manipulations by the witch hand, and the problem is generally more common in unconstrained long-fingered hands. These observations suggest that long-fingered hands may require root solutions that substantially deviate from standard hands, and that using a more similar reference hand may improve results.

Although we have discussed two interesting instances of object retargeting, our method is currently highly limited in the range of object variations it can support. In particular, we observe two primary failure modes: failure to adapt to different pregrasp and release configurations and failure to adapt to instances where contacts are not reachable. Figure 25 illustrates an instance where pregrasp adaptation negligence can produce severe penetration artifacts. We also provide examples in the supplementary video to illustrate similar issues for scaled objects; however, we show that our method is capable of handling such variations to a limited extent.

Finally, our formulation of matching source hand object contacts is generally less effective for target hands with large morphological or kinematic divergences. The Allegro hand is a prime example of a challenging retargeting problem because of its size and limited DOFs. The main difficulty caused by the large size of this manipulator was greater self-intersection, the likelihood of which is generally higher in larger hands. In such instances, and in others where the target hand diverges enough to substantially alter the general expected interaction mechanics (e.g., a human vs. a Barrett Hand<sup>4</sup>) or with respect to a particular manipulation (e.g., a pinch grasp on a larger hand vs. a human power grasp), matching original contacts exactly may not be the best problem formulation. While we demonstrated that our method can still handle considerable divergences, assuming identical object contacts

<sup>4</sup><https://robots.ros.org/barrett-hand/>

between hands is a limiting factor of our approach. In these instances, it may be worth considering alternative representations for mapping between hand motions [Karunratanakul et al. 2020; Khargonkar et al. 2023; She et al. 2022, 2024] and possibly augmenting these representations with methods introduced in this article. Addressing such divergences is an interesting area for future work.

## 8 Conclusion

We have presented a simple, reliable, and standardized framework capable of kinematically retargeting contact-rich anthropomorphic manipulations to a wide variety of target hands. Central to our method is the utilization of contact areas, for which we have presented both a novel, atlas-based shape matching algorithm capable of transferring localized contact data procedurally with high control and precision, and an optimization pipeline capable of utilizing said information to create high-quality retargeted results. We have also shown that our method performs well on bimanual tasks and under stress tests of hand size variations and cyclic retargeting. We have demonstrated that our method is capable of enabling unique extensions, including object substitution and visualizing the impact of hand design parameters over full trajectories. Finally, we have shown the value of contact information and key aspects of our processing pipeline through baseline comparisons and ablations. Importantly, we have maintained our commitment to standardization, simplicity, and reliability in the design of our pipeline, which we hope will encourage its adoption in downstream applications.

## Acknowledgments

The authors thank Yu Xi Lee, Dominik Bauer, and Wonik Robotics for the witch, alien, prosthetic, and Allegro hand models respectively; Melanie Danver and Moshe Mahler for assistance with presentation of results; and Vikash Kumar for referral to the GRAB dataset. This work was primarily conducted while Arjun Lakshminpathy was a visiting researcher at Fundamental AI Research (FAIR) at Meta and was partially funded by an AI Research and Mentorship (AIM) award. Partial support was also provided by the National Institute of Biomedical Imaging and Bioengineering of the National Institutes of Health under Award Number R01EB036842.

## References

- K. Aberman, P. Li, D. Lischinski, O. Sorkine-Hornung, D. Cohen-Or, and B. Chen. 2020. Skeleton-aware networks for deep motion retargeting. *ACM Trans. Graph.* 39, 4, Article 62 (2020).
- N. Aigerman and Y. Lipman. 2016. Hyperbolic orbifold Tutte embeddings. *ACM Transactions on Graphics* 35, 6, Article 217 (2016).
- D. Antotsiou, G. Garcia-Hernando, and Kim T. K. 2018. Task-oriented hand motion retargeting for dexterous manipulation imitation. In *Proceedings of the European Conference on Computer Vision Workshops*.
- S. Attaiqi, G. Pai, and M. Ovsjanikov. 2021. DPFM: Deep partial functional maps. In *Proceedings of the International Conference on 3D Vision*. 175–185.
- D. Bauer, C. Bauer, A. Lakshminpathy, R. Shu, and N. S. Pollard. 2022. Towards very low-cost iterative prototyping for fully printable dexterous soft robotic hands. In *Proceedings of the IEEE International Conference on Soft Robotics*. 490–497.
- S. Brahmabhatt, C. Ham, C. C. Kemp, and J. Hays. 2019a. ContactDB: Analyzing and predicting grasp contact via thermal imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8709–8719.
- S. Brahmabhatt, A. Handa, J. Hays, and D. Fox. 2019b. ContactGrasp: Functional multi-finger grasp synthesis from contact. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2386–2393.
- S. Brahmabhatt, C. Tang, C. D. Twigg, C. C. Kemp, and C. Hays. 2020. ContactPose: A dataset of grasps with object contact and hand pose. In *Proceedings of the European Conference on Computer Vision*. 361–378.
- X. Cheng, E. Huang, Y. Hou, and M. T. Mason. 2021. Contact mode guided sampling-based planning for quasistatic dexterous manipulation in 2D. In *Proceedings of the IEEE International Conference on Robotics and Automation*. 6520–6526.
- S. Choi, M. K. X. J. Pan, and J. Kim. 2020. Nonparametric motion retargeting for humanoid robots on shared latent space. In *Robotics: Science and Systems*.
- S. Christen, M. Kocabas, E. Aksan, J. Hwangbo, J. Song, and O. Hilliges. 2022. D-Grasp: Physically plausible dynamic grasp synthesis for hand-object interactions. In *Proceedings of the IEEE/CVF Computer Vision and Pattern Recognition*. 20577–20586.
- S. Dasari, A. Gupta, and V. Kumar. 2023. Learning dexterous manipulation from exemplar object trajectories and pre-grasps. In *Proceedings of the IEEE International Conference on Robotics and Automation*. 3889–3896.
- D. Eberly. 2005. Least Squares Fitting of Data with B-Spline Curves. Retrieved from <https://www.geometrictools.com/Documentation/BSplineCurveLeastSquaresFit.pdf>
- M. Edelstein, D. Ezuz, and M. Ben-Chen. 2020. ENIGMA: Evolutionary non-isometric geometry MAtching. *ACM Trans. Graph.* 39, 4, Article 112 (2020).
- D. Ezuz, J. Solomon, and M. Ben-Chen. 2019. Reversible harmonic maps between discrete surfaces. *ACM Trans. Graph.* 38, 2, Article 15 (2019).
- Z. Fan, O. Taheri, D. Tzionas, M. Kocabas, M. Kaufmann, M. J. Black, and O. Hilliges. 2023. ARCTIC: A dataset for dexterous bimanual hand-object manipulation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 12943–12954.
- C. Ferrari and J. Canny. 1992. Planning optimal grasps. In *Proceedings of the IEEE International Conference on Robotics and Automation*. 2290–2295.
- A. Gehre, M. M. Bronstein, L. Kobbelt, and J. Solomon. 2018. Interactive curve constrained functional maps. *Comput. Graph. Forum* 37, 5 (2018), 1–12.
- P. Grady, C. Tang, C. D. Twigg, M. Vo, S. Brahmabhatt, and C. C. Kemp. 2021. ContactOpt: Optimizing contact to improve grasps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1471–1481.
- A. Handa, K. V. Wyk, W. Yang, J. Liang, Y. W. Chao, Q. Wan, S. Birchfield, N. D. Ratliff, and D. Fox. 2019. DexPilot: Vision-based teleoperation of dexterous robotic hand-arm system. In *Proceedings of the IEEE International Conference on Robotics and Automation*. 9164–9170.
- C. Hazard, N. S. Pollard, and S. Coros. 2020. Automated design of robotic hands for in-hand manipulation tasks. *Int. J. Human. Robot.* 17, 1, Article 1950029 (2020).
- P. Herholz, F. Haase, and M. Alexa. 2017. Diffusion diagrams: Voronoi cells and centroids from diffusio. *Comput. Graph. Forum* 36, 2 (2017), 163–175.
- B. Humberston and D. K. Pai. 2015. Hands on: Interactive animation of precision manipulation and contact. In *Proceedings of the ACM SIGGRAPH / Eurographics Symposium on Computer Animation*. 63–72.
- I. Jang, S. Choi, S. Hong, C. Kim, and J. Noh. 2024. Geometry-aware retargeting for two-skinned characters interaction. *ACM Trans. Graph.* 43, 6, Article 203 (2024).
- T. Jin, M. Kim, and S. H. Lee. 2018. Aura mesh: Motion retargeting to preserve the spatial relationships between skinned characters. *Comput. Graph. Forum* 37, 2 (2018), 311–320.
- S. G. Johnson. 2017. The NLOpt Nonlinear-optimization Package. Retrieved from <http://ab-initio.mit.edu/nlopt>
- J. Jost. 2008. *Riemannian Geometry and Geometric Analysis*. Springer.
- E. Kalogerakis, A. Hertzmann, and K. Singh. 2010. Learning 3D mesh segmentation and labeling. *ACM Trans. Graph.* 29, 4, Article 102 (2010).
- K. Karunratanakul, J. Yang, Y. Zhang, M. J. Black, K. Muandet, and S. Tang. 2020. Grasping field: Learning implicit representations for human grasps. In *Proceedings of the International Conference on 3D Vision*. 333–344.
- N. Khargonkar, N. Song, Z. Xu, B. Prabhakaran, and Y. Xiang. 2023. Neuralgrasps: Learning implicit representations for grasps of multiple robotic hands. In *Proceedings of the Conference on Robot Learning*. 516–526.
- S. Kim, M. Sorokin, J. Lee, and S. Ha. 2022. Humanconquid: Human motion control of quadrupedal robots using deep reinforcement learning. In *Robotics: Science and Systems*.
- Y. Kim, H. Park, S. Bang, and S.H. Lee. 2016. Retargeting human-object interaction to virtual avatars. *IEEE Trans. Vis. Comput. Graph.* 22, 11 (2016), 2405–2412.
- P. G. Kry and D. K. Pai. 2006. Interaction capture and synthesis. *ACM Trans. Graph.* 25, 3 (2006), 872–880.
- V. Kumar and E. Todorov. 2015. MuJoCo HAPTIX: A virtual reality system for hand manipulation. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*. 657–663.
- A. Lakshminpathy, D. Bauer, C. Bauer, and N. S. Pollard. 2022. Contact transfer: A direct, user-driven method for human to robot transfer of grasps and manipulations. In *Proceedings of the IEEE International Conference on Robotics and Automation*. 6195–6201.
- A. Lakshminpathy, D. Bauer, and N. S. Pollard. 2021. Contact tracing: A low cost reconstruction framework for surface contact interpolation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. 5165–5172.

- A. S. Lakshminpathy, N. Feng, Y. X. Lee, M. Mahler, and N. S. Pollard. 2023. Contact edit: Artist tools for intuitive modeling of hand-object interactions. *ACM Trans. Graph.* 42, 4, Article 45 (2023).
- A. S. Lakshminpathy and N. S. Pollard. 2024. ContactMPC: Towards online adaptive control for contact-rich dexterous manipulation. In *Robotics: Science and Systems Workshops*.
- J. Lee and S. Y. Shin. 1999. A hierarchical approach to interactive motion editing for human-like figures. In *ACM Annual Conference on Computer Graphics and Interactive Techniques*. 39–48.
- S. Lee and S. H. Lee. 2019. Projective motion correction with contact optimization. *IEEE Trans. Vis. Comput. Graph.* 25, 4 (2019), 1746–1759.
- T. Li, J. Won, A. Clegg, J. Kim, A. Rai, and S. Ha. 2023. Ace: Adversarial correspondence embedding for cross morphology motion retargeting from human to nonhuman characters. In *SIGGRAPH Asia Conference Proceedings*. 1–11.
- Y. Li, J. L. Fu, and N. S. Pollard. 2007. Data-driven grasp synthesis using shape matching and task-based pruning. *IEEE Trans. Vis. Comput. Graph.* 13, 4 (2007), 732–747.
- R. M. Malina, P. V. V. Hamill, and S. Lemeshow. 1973. *Selected Body Measurements of Children 6-11 Years, United States*. National Center for Health Statistics.
- R. Marin, S. Melzi, E. Rodolà, and U. Castellani. 2020. FARM: Functional automatic registration method for 3D human bodies. *Comput. Graph. Forum* 39, 1 (2020), 160–173.
- J. S. B. Mitchell, D. M. Mount, and C. H. Papadimitriou. 1987. The discrete geodesic problem. *SIAM J. Comput.* 16, 4 (1987), 647–668.
- I. Mordatch, Z. Popović, and E. Todorov. 2012. Contact-invariant optimization for hand manipulation. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. 137–144.
- NASA. 2000. Anthropometry and biomechanics. In *Man Systems Integration Standards* (2000).
- M. Ovsjanikov, M. Ben-Chen, J. Solomon, A. Butscher, and L. Guibas. 2012. Functional maps: A flexible representation of maps between shapes. *ACM Trans. Graph.* 31, 4, Article 30 (2012).
- G. Pai, J. Ren, S. Melzi, P. Wonka, and M. Ovsjanikov. 2021. Fast sinkhorn filters: Using matrix scaling for non-rigid shape correspondence with functional maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 384–393.
- T. Pang, H. J. T. Suh, L. Yang, and R. Tedrake. 2023. Global planning for contact-rich manipulation via local smoothing of quasi-dynamic contact models. *IEEE Trans. Robot.* 39, 6 (2023), 4691–4711.
- M. Panine, M. Kirgo, and M. Ovsjanikov. 2022. Non-isometric shape matching via functional maps on landmark-adapted bases. *Comput. Graph. Forum* 41, 6 (2022), 394–417.
- Y. Qin, H. Su, and X. Wang. 2022. From one hand to multiple hands: Imitation learning for dexterous manipulation from single-camera teleoperation. *IEEE Robot. Autom. Lett.* 7, 4 (2022), 10873–10881.
- Y. Qin, Y. H. Wu, S. Liu, H. Jiang, R. Yang, Y. Fu, and X. Wang. 2021. DexMV: Imitation learning for dexterous manipulation from human videos. In *Proceedings of the European Conference on Computer Vision*. 570–587.
- A. Rajeswaran, V. Kumar, A. Gupta, G. Vezzani, J. Schulman, E. Todorov, and S. Levine. 2018. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. In *Robotics: Science and Systems*.
- D. Reda, J. Won, Y. Ye, M. van de Panne, and A. Winkler. 2023. Physics-based motion retargeting from sparse inputs. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* 6, 3 (2023), 1–19.
- H. Rhodin, J. Tompkin, K. In Kim, K. Varanasi, H. P. Seidel, and C. Theobalt. 2014. Interactive motion mapping for real-time character control. *Comput. Graph. Forum* 33, 2 (2014), 273–282.
- M. A. Roa and R. Suarez. 2009. Computation of independent contact regions for grasping 3-D objects. *IEEE Transactions on Robotics* 25, 4 (2009), 839–850.
- J. Romero, D. Tzionas, and M. J. Black. 2017. Embodied hands: Modeling and capturing hands and bodies together. *ACM Trans. Graph.* 36, 6, Article 245 (2017).
- H. Ryu, M. Kim, S. Lee, M. S. Park, K. Lee, and J. Lee. 2021. Functionality-driven musculature retargeting. *Comput. Graph. Forum* 40, 1 (2021), 341–356.
- R. Schmidt, C. Grimm, and B. Wyvill. 2006. Interactive decal compositing with discrete exponential maps. *ACM Trans. Graph.* 25, 3 (2006), 605–613.
- N. Sharp, S. Attaiki, K. Crane, and M. Ovsjanikov. 2022. DiffusionNet: Discretization agnostic learning on surfaces. *ACM Transactions on Graphics* 41, 3, Article 27 (2022).
- N. Sharp, Y. Soliman, and K. Crane. 2019. The vector heat method. *ACM Transactions on Graphics* 38, 3, Article 24 (2019).
- Q. She, R. Hu, J. Xu, M. Liu, K. Xu, and H. Huang. 2022. Learning high-DOF reaching-and-grasping via dynamic representation of gripper-object interaction. *ACM Transactions on Graphics* 41, 4, Article 97 (2022).
- Q. She, S. Zhang, Y. Ye, M. Liu, R. Hu, and K. Xu. 2024. Learning cross-hand policies for high-DOF reaching and grasping. In *Proceedings of the European Conference on Computer Vision*. 269–285.
- A. Sivakumar, K. Shaw, and D. Pathak. 2022. Robotic telekinesis: Learning a robotic hand imitator by watching humans on Youtube. In *Robotics: Science and Systems*.
- K. Svanberg. 2002. A class of globally convergent optimization methods based on conservative convex separable approximations. *SIAM Journal on Optimization* 12, 2 (2002), 555–573.
- O. Taheri, N. Ghorbani, M. J. Black, and D. Tzionas. 2020. GRAB: A dataset of whole-body human grasping of objects. In *Proceedings of the European Conference on Computer Vision*. 581–600.
- K. Takayama. 2022. Compatible intrinsic triangulations. *ACM Transactions on Graphics* 41, 4, Article 57 (2022).
- D. Turpin, L. Wang, E. Heiden, Y. C. Chen, M. Macklin, S. Tsogkas, S. Dickinson, and A. Garg. 2022. Grasp'D: Differentiable contact-rich grasp synthesis for multi-fingered hands. In *Proceedings of the European Conference on Computer Vision*. 201–221.
- R. Villegas, D. Ceylan, A. Hertzmann, J. Yang, and J. Saito. 2021. Contact-aware retargeting of skinned motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9720–9729.
- W. Wei, P. Wang, and S. Wang. 2023. Generalized Anthropomorphic Functional Grasping with Minimal Demonstrations. *arXiv preprint arXiv:2303.17808* (2023).
- J. Won and J. Lee. 2019. Learning body shape variation in physics-based characters. *ACM Transactions on Graphics* 38, 6, Article 207 (2019).
- Y. Wu, J. Wang, Y. Zhang, S. Zhang, O. Hilliges, F. Yu, and S. Tang. 2022b. SAGA: Stochastic whole-body grasping with contact. In *Proceedings of the European Conference on Computer Vision*. 257–274.
- Y. H. Wu, J. Wang, and X. Wang. 2022a. Learning generalizable dexterous manipulation from human grasp affordance. In *Proceedings of the Conference on Robot Learning*. 618–629.
- Y. Ye and C. K. Liu. 2012. Synthesis of detailed hand manipulations using contact sampling. *ACM Trans. Graph.* 31, 4, Article 41 (2012).
- J. Zhang, J. Weng, D. Kang, F. Zhao, S. Huang, X. Zhe, L. Bao, Y. Shan, J. Wang, and Z. Tu. 2023b. Skinned motion retargeting with residual perception of motion semantics and geometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13864–13872.
- Y. Zhang, D. Gopinath, Y. Ye, J. Hodgins, G. Turk, and J. Won. 2023a. Simulation and retargeting of complex multi-character interactions. In *Proceedings of the ACM SIGGRAPH Conference Proceedings*. 1–11.
- Q. Zhou and A. Jacobson. 2016. Thing10K: A dataset of 10,000 3D-printing models. *arXiv preprint arXiv:1605.04797* (2016).

## Appendices

### A Artist Annotation Procedures

All virtual markers and axial curves used in our results were generated by an artist with basic training in our annotation tools. We roughly detail the artist’s annotation process, the requests we made, and tools we provided in the following subsections.

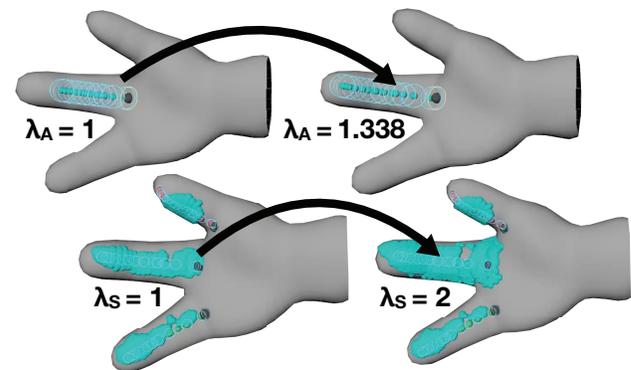


Fig. 26. Depictions of sample starting and final values of  $\lambda_A$  and  $\lambda_S$ , which typically are adjusted to target finger length and girth, respectively.

#### A.1 Virtual Markers

We provided the artist with a representative photo of a full hand marker set commonly used in motion capture. We requested the artist to designate equivalent sets of markers on the source and all target hands. The artist performed the annotation by selecting

individual or groups of vertices on each mesh using default brush tools commonly available in three-dimensional modeling software. The received annotations were processed and stored as corresponding virtual marker sets for all experiments.

## A.2 Axial Curves

We requested the artist to identify similar features on the palmar surface of each hand but provided no further directions.

We provided the artist a custom tool set to assist with the axis creation process. To do so, the artist first selects two points on the source hand. A geodesic is then traced between the two points in the order of selection, and all intermediate edge crossings are automatically designated as discrete axis points. Traversal between points is stored via the exponential map with the turning angle of the outgoing tangent vector being computed relative to that of the incoming tangent vector. This representation allows for approximate isometric reconstruction independently of the tangent basis at any point except the starting location, which is critical for the transfer procedure. A more comprehensive explanation of the axial curve model is available in existing work [Lakshmipathy et al. 2023].

The artist then selects two points on the target hand to initiate the transfer process. We use the first point as the origin for the axis reconstruction and the geodesic trace between the two points to determine the initial turning angle. The remainder of the axis curve is then reconstructed automatically from the exponential map computed on the source domain. After reconstruction, the artist determines  $\lambda_A$  for the target hand axis. In practice, a satisfactory  $\lambda_A$  extended the axis along the full length of the finger (Figure 26). The procedure is repeated until all axial curves have been designated on the source hand, transferred to the target hand, and fitted with an appropriate  $\lambda_A$ . The source hand curves were reused to accelerate annotation of the remaining target hands.

To determine  $\lambda_S$ , we provided the artist three sample frames of contact data on the source hand from a single motion sequence. Contacts were first loaded and parameterized on the source hand. After parameterization, contacts were transferred to the target hand and reconstructed from corresponding axial points under the linear scaling previously determined by  $\lambda_A$ .  $\lambda_S$  was then adjusted until an appropriate semantically meaningful distribution was produced (Figure 26). The artist then validated the parameter choice by repeating the transfer procedure with the remaining two frames of data. The procedure was repeated with all axial curves and hands to complete the annotation. The determined parameter values were used to procedurally transfer all frames of contact data in all results without any additional adjustment.

## B Experiment Details

Tables 2 and 3 tabulate the parameters of our target hand models and transfer coefficients. The optimization weighting coefficients  $\lambda_m$ ,  $\lambda_{cd}$ ,  $\lambda_{cn}$ ,  $\lambda_c$ , and  $\lambda_t$  were all set to 1.0, while  $\lambda_j$  was set to 50.0. The same coefficients were used in all retargets. The MANO hand does not contain a skeleton and was instead animated via vertex positions. Table 4 tabulates the total frames, contacts per frame, and number of fitted B-Spline control points for each task in our suite.

Table 2. Parameters for All Target Hands

Hand Parameters				
	Root	No. Fingers	No. VMarkers	No. DOFs
MANO	Wrist	5	19	N/A
Human	Forearm	5	19	54
Witch	Forearm	5	19	54
Alien	Forearm	3	13	42
Allegro	Wrist	4	16	22
Prosthetic	Wrist	5	19	26

Table 3.  $\lambda$  Coefficients Used for Bulk Contact Transfer from the MANO Hand

Hand Alignment Parameters ( $\lambda_A$ , $\lambda_S$ )					
	Thumb	Index	Middle	Ring	Pinky
Human	1.0, 1.2	1.0, 1.0	1.0, 1.0	1.0, 1.0	1.0, 1.0
Witch	1.15, 1.46	1.55, 1.5	1.55, 1.59	1.55, 1.51	1.55, 1.5
Alien	1.22, 1.45	1.27, 1.5	1.23, 1.57	N/A	N/A
Allegro	1.8, 1.9	1.8, 1.913	1.8, 1.97	1.8, 2.0	N/A
Prosthetic	1.33, 1.5	1.33, 1.4	1.33, 1.4	1.33, 1.3	1.33, 1.43

Table 4. Parameters for All Tasks in Our Suite

Task Parameters			
	Frames	No. Contacts / Frame ( $\mu \pm \sigma$ )	Control Points
Apple	703	1648.75 $\pm$ 360.17	60
Doorknob	1040	4766.07 $\pm$ 1334.91	40
Flashlight	1040	552.75 $\pm$ 274.48	80
Hammer	768	1235.53 $\pm$ 536.34	90
Phone	1145	415.62 $\pm$ 161.28	120
Stapler	877	516.27 $\pm$ 387.64	70

Motions all run at 120 FPS.

We set  $\mathcal{E}_{acc}$  to  $500^\circ/s^2$  for all angular motion and  $500 \text{ cm}/s^2$  for all linear motion. We cap the number of iterations at 20. The same bounds were applied to all DOFs across all manipulators indiscriminately. Note that the units of linear motion are arbitrary and vary depending on the scene scale. While these thresholds are high, we found that tighter bounds were not necessary, because the provided estimates were only data points for spline fits. Brief and intermittent noise from remaining large jumps in acceleration were automatically smoothed out due to our splines consisting of relatively few control points (<12% of total frame counts).

## C Compute-time Analysis

Although our computation times are currently somewhat slow, there do exist several cheap acceleration strategies. We consider four techniques in this section: reducing mesh resolution, reducing the number of DOFs, reducing the number of virtual marker points, and reducing the number of contact points. We then conclude with a discussion of further potential improvements.

We consider the initial trajectory estimation phase of the prosthesis hand doorknob task for our analysis, which is notable for being the slowest step of one of the slowest results to compute

among all 30 tasks. Consequently, this step has considerable room for improvement among our four axes of analysis.

### C.1 Results

Table 5. Compute-time Variations Induced by Mesh Resolution Alterations

Compute Times by Mesh Resolution (Res.)			
Original	50% Res.	25% Res.	12.5% Res.
13.76 hr	13.34 hr	12.54 hr	12.18 hr

*Mesh Resolution Reduction.* The original prosthesis mesh is comprised of 27,780 faces. We consider three levels of face reduction: 50%, 75%, and 87.5%. Table 5 tabulates the results of the findings.

Table 6. Compute-time Variations Induced by DOF Alterations

Compute Times by DOFs		
Original (26-DOFs)	3-DOF PCP (31-DOFs)	1-DOF PCP (21-DOFs)
13.76 hr	15.73 hr	11.56 hr

*DOF Reduction.* The original prosthesis rig is composed of 2-DOF proximal phalange (PCP) joints at each finger. We examine two alterations: 3-DOF PCP finger joints and 1-DOF PCP finger joints. Table 6 tabulates the results of the findings.

*Virtual Marker Reduction.* The original prosthesis hand consists of 19 virtual markers with approximately 60 points each. We examine an alteration where each marker is summarized into a single point via averaging. Table 7 tabulates the results of the findings.

Table 7. Compute-time Variations Induced by Virtual Marker Counts

Compute Times by Marker Counts	
Original (19 Markers, ~60 points each)	Reduced (19 Markers, 1 point each)
13.76 hr	11.56 hr

Table 8. Compute-time Variations Induced by Altering the Number of Total Contact Points across All Frames

Compute Times by No. of Contacts		
Original	50% Contacts	25% Contacts
13.76 hr	9.40 hr	7.15 hr

*Contact Point Reduction.* The doorknob task contains the largest number of contact points out of all tasks by a large margin. We

can systematically reduce the number of contact points across all frames by reducing the filter tolerance as outlined in Section 3.3.2, paragraph 3. We select tolerances such that the total number of contacts are reduced by 50% and 75% and show results in Table 8.

### C.2 Discussion

Tables 5–8 clearly indicate that compute times can be reduced by reducing mesh resolution, DOFs, number of marker points, and number of contact points, although performance gains vary depending on the axis of reduction. Reducing the number of contact points provides the most substantial speedup; however, as observed from the baseline comparisons, ignoring contacts can significantly degrade motion reconstruction quality. From these results, we recommend the following steps to improve computation times:

- (1) Reduce the total number of DOFs in the system if possible.
- (2) Reduce mesh resolution as much as possible. Reductions in resolution will not impact the final retarget.
- (3) Reduce the number of virtual marker points, but verify free-space trajectory estimates will remain largely unaltered.
- (4) Reduce the number of contact points such that the initial trajectory estimates remain acceptable. It is reasonable to predict that the final retarget will approximate a smoothed-out version of the initial trajectory estimate.

### C.3 Future Improvements

The compute-time reduction techniques proposed in this section are intended to be inexpensive and easy to perform under our proposed method; however, there is still ample room for further improvement. Two immediate pathways are worth considering.

First, our entire retargeting pipeline was implemented in Autodesk Maya. Although Maya’s UI was useful, we found that it was not optimal for compute-time performance. Because the software suite is closed source and does not provide a detailed overview of scene graph updates, it is possible that there is considerable “bookkeeping” overhead induced during our optimization pipeline. A more bare-bones software suite optimized for optimization pipelines could therefore provide substantial speedup out of the box.

Second, our optimization is currently bottlenecked by **finite-difference (FD)** approximations of gradients. We used FD due to both its ease of implementation and ability to easily adapt to different skinning policies, contact distributions, and scene geometries. Analytical or chain-rule computed gradients thus also have the potential to provide substantial speedup, albeit at the cost of additional implementation or memory overhead.

Received 11 April 2024; revised 13 February 2025; accepted 16 February 2025