Grasp'D: Differentiable Contact-rich Grasp Synthesis for Multi-fingered Hands

Dylan Turpin[1,2,3], Liquan Wang[1,2,3], Eric Heiden[3], Yun-Chun Chen[1,2], Miles Macklin[3], Stavros Tsogkas[4], Sven Dickinson[1,2,4], and Animesh Garg[1,2,3]

[1] University of Toronto, [2] Vector Institute, [3] Nvidia, [4] Samsung
dylanturpin@cs.toronto.edu

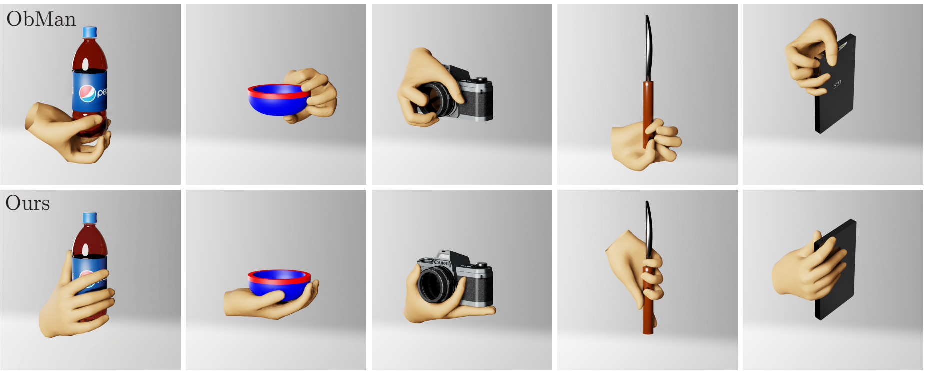

Fig. 1: Multi-finger grasp synthesis with Differentiable Simulation. Analytically synthesized grasps, such as in ObMan [39] based on the GraspIt! [63], plan sparse contacts at the fingertips. Our method (Grasp'D) for grasp synthesis discovers stable, contact-rich grasps that conform to detailed object surface geometry. Grasp'D creates larger contact-areas that better match the contact distribution of real human grasps.

Abstract. The study of hand-object interaction requires generating viable grasp poses for high-dimensional multi-finger models, often relying on analytic grasp synthesis which tends to produce brittle and unnatural results. This paper presents Grasp'D, an approach to grasp synthesis by differentiable contact simulation that can work with both known models and visual inputs. We use gradient-based methods as an alternative to sampling-based grasp synthesis, which fails without simplifying assumptions, such as pre-specified contact locations and eigengrasps. Such assumptions limit grasp discovery and, in particular, exclude high-contact power grasps. In contrast, our simulation-based approach allows for stable, efficient, physically realistic, high-contact grasp synthesis, even for gripper morphologies with high-degrees of freedom. We identify and address challenges in making grasp simulation amenable to gradient-based optimization, such as non-smooth object surface geometry, contact sparsity, and a rugged optimization landscape. Grasp'D compares favorably to analytic grasp synthesis on human and robotic hand models, and resultant grasps achieve over 4× denser contact, leading to significantly higher grasp stability. Video and code available at: graspd-eccv22.github.io.

Keywords: Multi-finger grasping, grasp synthesis, vision-based grasping

1 Introduction

Humans use their hands to interact with objects of varying shape, size, and material thousands of times throughout a single day. Despite being effortless -- almost instinctive -- these interactions employ a complex visuomotor system, with components that correspond to dedicated areas of computer vision research. Visual inputs from the environment are processed in our brain to recognize objects of interest (object recognition [16, 22, 26, 34, 85]), identify modes of interaction to achieve a certain function (affordance prediction [7, 20, 53, 72, 76]), and position our hand(s) in a way that enables that function (pose estimation [2, 6, 32, 37, 80, 91], grasping [25, 51, 82]). Proficiency in this task comes from accumulated experience in interacting with the same object over time, and readily extends to new categories or different instances of the same category.

This is an intriguing observation: humans can leverage accumulated knowledge from previous interactions, to quickly infer how to successfully manipulate an unknown object, *purely from visual input*. Granting machines the same ability to directly translate visual cues into plausible grasp predictions can have significant practical implications in the way robotic manipulators interact with novel objects [25, 77] or in virtual environments in AR/VR [18, 30].

Grasp prediction has previously been considered in the context of computer vision [42, 45, 67, 89] and robotics [69]. It amounts to predicting the base pose (position and rotation) and joint angles of a robotic or human hand that is stably grasping a given object. This prediction is usually conditioned on visual inputs, such as RGB(D) images, point clouds, etc., and is typically performed online for real-time applications. Predicting grasps from visual inputs can be naturally posed as a learning problem, using paired visual data with their respective grasp annotations. However, capturing and annotating human grasps is laborious and not applicable to robotic grasping, so researchers often rely on datasets of synthetically generated grasps instead (see Table 1 for a list of recent works). Consequently, high-quality datasets of plausible, diverse grasps are crucial for any modern vision system performing grasp prediction, motivating the development of better methods for grasp synthesis.

Grasp synthesis assumes that the complete object geometry (e.g., mesh) is known, and is usually achieved by optimizing over a grasping metric which can be computed analytically or through simulation. *Analytic metrics* are handcrafted measures of a grasp's quality. For example, the epsilon metric [27] measures the magnitude of the smallest force that can break a grasp, computed as a function of the contact positions and normals that the grasp induces. While analytic metrics can be computationally faster, they often transfer poorly to the real world. *Simulation-based metrics* [24, 48, 90] measure grasp quality by running a simulation to test grasp effectiveness, e.g., by shaking the object and checking whether it is dropped. These can achieve a higher degree of physical fidelity, but require more computation. In both cases, optimization is usually black box, as neither the analytic metric or simulator is differentiable. Black box optimization can find good grasps in a reasonable number of steps as long as the search space is low-dimensional, e.g., when searching the pose space of parallel-jaw

Year	Name	Hand Model(s)	Analytic (A) or Human Capture (HC)
2019	ObMan [39]	MANO	A (GraspIt! [63])
2019	ContactDB [7]	MANO	HC
2020	Hope-net [21]	MANO	A (ObMan [39])
2020	UniGrasp [78]	Various	A (FastGrasp [71])
2020	ContactPose [9]	MANO	HC
2020	GANHand [15]	MANO	Other (manual)
2020	Grasping Field [49]	MANO	A (ObMan)
2020	GRAB [81]	MANO	HC
2021	Multi-Fin GAN [56]	Barrett	A (GraspIt!)
2021	DDGC [57]	Barrett	A (GraspIt!)
2021	Contact-Consistency [46]	MANO	A (ObMan)

Table 1: Modern vision-based grasp prediction for multi-finger hands relies on datasets created by human capture or analytic synthesis. Human capture is expensive and does not address the need for robotic grasp datasets. Analytic synthesis is only practical under significant limiting assumptions that exclude key grasp types [15, 39].

grippers [19, 23, 24, 66, 84]. However, when the number of degrees of freedom becomes larger, as in the case of multi-finger grippers, black box optimization over a grasping metric (whether analytic or simulation-based) becomes infeasible. Simplifying assumptions can be made to reduce the dimensionality of the search space, but they often reduce the plausibility of generated grasps.

To address these shortcomings, we propose Grasp'D, a grasp synthesis pipeline based on *differentiable simulation* which can generate contact-rich grasps that realistically conform to object surface geometry without any simplifying assumptions. A metric based on differentiable simulation admits gradient-based optimization, which is sample-efficient, even in high-dimensional spaces, and affords all the benefits of simulation-based metrics, i.e., physical plausibility, scalability, and extendability. Differentiable grasping simulation, however, also presents new challenges. Non-smooth object geometry (e.g., at the edges or corners of a cube) results in discontinuities in the contact forces and, subsequently, our grasping metric, complicating gradient-based optimization. Adding to that, if the hand and the object are not touching, small perturbations to the hand pose do not generate any additional force, resulting in vanishing gradients. Finally, the optimization landscape is rugged, making optimization challenging. Once the hand is touching the object, small changes to the hand pose may result in large changes to contact forces (and our metric).

We address these challenges as follows: (1) At the start of each optimization, we simulate contact between the hand and a smoothed, padded version of the object surface that gradually resolves to the true, detailed surface geometry, using a coarse-to-fine approach. This smoothing softens discontinuities in surface normals, allowing gradient-based optimization to smoothly move from one continuous surface area to another. This is enabled by our signed-distance function (SDF) approach to collision detection, which lets us freely recover a rounded object surface as the radius r level set of the SDF. (2) We allow gradients to *leak* through force computations for contact points that are not yet in the collision, introducing a biased gradient that can be followed to create new contacts. The intuition behind this choice is similar to the one for using LeakyReLU activations to prevent the phenomenon of "dying neurons" in deep neural networks [58]. (3) Inspired by Contact-Invariant Optimization (CIO) [64, 65], we relax the problem formulation by introducing additional force variables that allow physics violations to be treated as a cost rather than a constraint. In effect, this decomposes the problem into finding contact forces that solve the task (of keeping the object

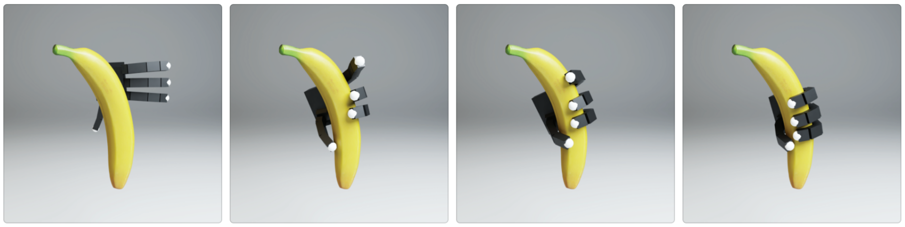

Fig. 2: Our method can synthesize grasps for both human and robotic hands, such as the four-finger Allegro hand in this figure. After hand initialization, we run gradient-based optimization to iteratively improve the grasp, in terms of stability and contact area. We include additional examples in Appendix B.

stably in place) and finding a hand pose that provides those forces. We evaluate our method on synthetic object models from ShapeNet [11] and object meshes reconstructed from the YCB RGB-D dataset [10]. Experimental results show that our method generates contact-rich grasps with physical realism and with favorable performance against an existing analytic method [39].

Figure 1 displays example grasps generated by our method side-by-side with grasps from [39]. Because we do not make assumptions about contact locations or reduce the dimensionality of the search space, our method can discover contact-rich grasps that are more stable and more plausible than the fingertip-only grasps usually discovered by analytic synthesis. The same procedure works equally for robotic hands. Figure 2 displays snapshots of an optimization trajectory for an Allegro hand. As optimization progresses and our simulated metric decreases, the grasp becomes increasingly stable, plausible, and high-contact.

Summary of contributions:

1. We propose a differentiable simulation-based protocol for generating synthetic grasps from visual data. Unlike other simulation-based approaches, our method can scale to tens of thousands of dense contacts, and discover plausible, contact-rich grasps, without any simplifying assumptions.
2. We address challenges arising from the differentiable nature of our scheme, using a coarse-to-fine SDF collision detection approach, defining leaky gradients for contact points that are not yet in collision, and integrating physics violations as additional terms to our cost function.
3. We show that our method finds grasps with better stability, lower interpenetration, and higher contact area when compared to analytic grasp synthesis baselines, and justify our design choices through extensive evaluations.

2 Related Work

Grasp synthesis. Although analytic metrics have been successfully applied to parallel-jaw gripper grasp synthesis (based on grasp wrench space analysis [27, 35, 63], robust grasp wrench space analysis [60, 86], or caging [62, 74]), more recent works [19, 24, 48, 66] have focused on simulation-based synthesis. While they are more computationally costly, simulation-based metrics for parallel-jaw grasps better align with human judgement [48] and with real world performance [17, 24, 61, 66]. In contrast to parallel-jaw grippers, multi-finger grasp synthesis is still largely analytic, with many recent works in multi-finger robotic grasping [56, 57, 78], grasp affordance prediction [49], and hand-object pose estimation [21, 39, 46] relying on datasets of analytically synthesized grasps (see Table 1). Notably, [21, 39, 46, 49, 56, 57] all use datasets synthesized with the GraspIt! [63] simulator, which is widely used for both multi-finger robotic and human grasp synthesis. The ObMan dataset [39] for hand-object pose estimation (also used in [46, 49]) is constructed by performing grasp synthesis with the MANO hand [75] in the GraspIt! Eigengrasp planner, and rendering the synthesized grasps against realistic backgrounds. The GraspIt! Eigengrasp planner optimizes analytic metrics based on grasp wrench space analysis. Dimensionality reduction [13] in the hand joint space, or using pre-specified contact locations for each hand link can be used to make the problem more tractable, but this limits the space of discoverable grasps and requires careful tuning. Our approach can successfully operate in the full grasp space, eschewing such simplifying assumptions while excelling in terms of physical fidelity over analytic synthesis for multi-finger grippers.

Human grasp capture. To estimate human grasps from visual inputs, existing methods train models on large-scale datasets [7, 9, 31, 38, 83]. Collecting these datasets puts humans in a lab with precise, calibrated cameras, lidar, and special gloves for accurately capturing human grasp poses. A human in the loop may also be needed for collecting annotations. All these requirements make the data collection process expensive and laborious. In addition, the captured grasps are only appropriate for human hands and not for robotic ones (which are important for many applications [1, 12]). Some works [8, 52] aim to transfer human grasps to robotic hands by matching contact patterns, but these suffer from important limitations, since the same contacts may not be achievable by human and robotic hands, given differences in their morphology and articulation constraints (e.g., see Fig. 8 of [52]). Our method provides a procedural way of generating high quality grasps for any type of hand -- human or robotic.

Vision-based grasp prediction. Whereas grasp synthesis is useful for generating grasps when full object geometry is available (i.e., a mesh or complete SDF is given), practical scenarios require predicting grasps from visual input. GAN-Hand [15] learns to predict human grasp affordances (as poses of a MANO [75] hand model) from input RGBD images using GANs. Since analytic synthesized datasets do not include many high-contact grasps, the authors also released the YCB Affordance dataset of 367 fine-grained grasps of the YCB object set [10], created by manually setting MANO hand joint angles in the GraspIt! simulator's GUI. Rather than predicting joint angles, Grasping Field [49] takes an implicit approach to grasp representation by learning to jointly predict signed distances for the MANO hand and the object to be grasped. For parallel-jaw grippers, most recent works [47, 61, 66, 79] learn from simulation-based datasets (e.g., [24, 48]). In contrast, multi-finger grasp prediction systems are still trained on either analytically synthesized datasets or datasets of captured human grasps (see Table 1). [21, 39, 46, 49, 56, 57] all use analytically synthesized datasets from the GraspIt! simulator [63], whereas [7, 9, 81] use datasets of captured human grasps. [36, 46] use captured human grasps to train a contact model, then refine grasps at test-time by optimizing hand pose to match predicted contacts. The higher quality training data generated by our grasp synthesis pipeline can lead to improved performance for any of these vision-based grasping prediction systems. Our system can also be used directly for vision-based grasp prediction, by running simulations with reconstructed objects (see Section 4.3).

Differentiable Grasping. We know of two works that have created differentiable grasp metrics in order to take advantage of gradient-based optimization for multi-finger grasp synthesis. [54] formulates a differentiable version of the epsilon metric [27] and uses it to synthesize grasps with the shadow robotic hand. They formulate the epsilon metric computation as a semidefinite programming (SDP) problem. Sensitivity analysis on this problem can then provide the gradient of the solution with respect to the problem parameters, including gripper pose. They manually label 45 potential contact points on the gripper. In contrast, we are able to scale to tens of thousands of contact points. Since the gripper may not yet be in contact with the object, they use an exponential weighting of points. Liu et al. [55] formulate a differentiable force closure metric and use gradient-based optimization to synthesize grasps with the MANO [75] hand model. Their formulation assumes zero friction and that the magnitude of all contact forces is uniform across contact points (although an error term allows both of these constraints to be slightly violated). Our method requires neither of these assumptions: the user can specify varying friction coefficients, and contact forces at different points are free to vary realistically. Their optimization problem involves finding a hand pose and a subset of candidate contact points on the hand that minimize an energy function. They find that the algorithm performs better with a smaller number of contact points and candidates. Selecting 3 contact points from the 773 candidate vertices of the MANO hand, it takes about 40 minutes to find 5 acceptable grasps. In contrast, our method is able to scale to tens of thousands of contact points while synthesizing an acceptable grasp in about 5 minutes. Notably, both of these prior works aim to take an analytic metric (the epsilon metric [27]) and make a differentiable variant. In contrast, we are presenting a differentiable simulation-based metric, which prior work on parallel-jaw grippers suggests will have greater physical fidelity [17, 24, 66] and better match human judgements [48] than analytic metrics.

Differentiable Physics. There has been significant progress in the development of differentiable physics engines [28, 33, 40, 41, 43, 44, 73, 87, 88]. However, certain limitations in recent approaches render them inadequate. Brax [28] and the Tiny Differentiable Simulator [41] only support collision primitives and cannot model general collisions between objects. Nimblephysics [87] supports mesh-to-mesh collision, but cannot handle cases where the gradient of contact normals with respect to position is zero (e.g., on a mesh face). While its analytic computation of gradients is fast, Nimblephysics requires manually writing forward and backward passes in C++, and only runs on CPU. Our work presents a new class of differentiable physics simulators to addresses many of these shortcomings.

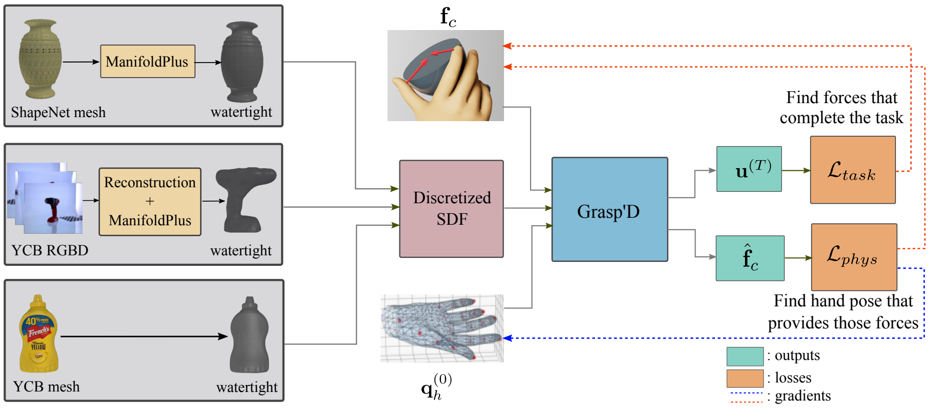

Fig. 3: Method overview. Grasp'D takes as input the discretized-SDF of an object (computed from a mesh or reconstructed from RGB-D) and synthesizes a stable grasp that can hold the object static as we vary the object's initial velocity. We optimize jointly over a hand pose $\mathbf{u}^{(T)}$ and the stabilizing forces $\hat{\mathbf{f}}_c$ provided by its contacts.

Further, Grasp'D supports GPU parallelism, enabling us to scale to tens of thousands of contacts, effectively approximating surface contacts.

3 Grasp'D: Differentiable Contact-rich Grasp Synthesis

We present a method for solving the grasp synthesis problem (Figure 3). From an input object and hand model (represented respectively by a signed-distance function and an articulation chain with mesh links), we generate a physically-plausible stable grasp, as a base pose and joint angles of the hand. This is achieved by iterative gradient-based optimization over a metric computed by differentiable simulation. The final grasp is dependent on the pose initialization of the hand, so different grasps can be recovered by sampling different starting poses. We detail our method below, but first outline the challenges that motivate our design.

Non-smooth object geometry. When optimizing the location of contacts between a hand and a sphere, the gradient of contact normals with respect to contact positions is well-defined and continuous, allowing gradient-based optimization to smoothly adjust contact positions along the sphere surface. But most objects are not perfectly smooth. Discontinuities in surface normals (e.g., at the edges or corners of a cube) result in discontinuities in contact normals and their gradients with respect to contact positions. Gradient-based optimization cannot effectively optimize across these discontinuities (e.g., cannot follow the gradient to move contact locations from one face of a cube to another). We address this with a coarse-to-fine smoothing approach, optimizing against a smoothed and padded version of the object surface that gradually resolves to the true surface as optimization continues (see Section 3.2).

Contact sparsity. Of all possible contacts between the hand and object, only a sparse subset is active at any given time. If a particular point on the hand is inactive (not in contact with the object), then an infinitesimal perturbation of the hand pose will not change its status (make it touch the object). The gradient of the force applied by any inactive contact (with respect to hand pose) will be exactly zero. This means that gradient-based optimization can not effectively create new contacts, since contacts that are not already active do not contribute to the gradient. We address this by allowing gradient to *leak* through the force computations of inactive contacts (see Section 3.3).

Rugged optimization landscape. When many contacts are active (i.e., hand touching the object), small changes to hand pose may result in large changes to contact forces and, subsequently, large changes to our grasp metric. This makes gradient-based optimization challenging. We address this with a problem relaxation inspired by Contact-Invariant Optimization [64, 65] (see Section 3.4).

3.1 Rigid body dynamics

In the interest of speed and simplicity, we limit ourselves to simple rigid body dynamics. Let $\mathbf{q}$ and $\mathbf{u}$ be the joint and spatial coordinates, respectively, with first and second time derivatives $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$, $\dot{\mathbf{u}}$, $\ddot{\mathbf{u}}$. Let $\mathbf{M}$ be the mass matrix. The kinematic map $\mathbf{H}$ maps joint coordinate time derivatives to spatial velocities as $\dot{\mathbf{q}} = \mathbf{H}(\mathbf{q})\mathbf{u}$, and is related to contact and external forces ($\mathbf{f}_{\mathrm{c}}$ and $\mathbf{f}_{\mathrm{ext}}$) through the following motion equation: $\mathbf{H}\mathbf{M}\mathbf{H}^{\top}\ddot{\mathbf{q}} = \mathbf{f}_{\mathrm{c}} + \mathbf{f}_{\mathrm{ext}}$, which yields the semi-implicit Euler update used for discrete time stepping [5]:

$$\dot{\mathbf{q}}^{(t+1)} \leftarrow \dot{\mathbf{q}}^{(t)} + \Delta t \mathbf{M}^{-1}(\mathbf{f}_{\mathrm{c}} + \mathbf{f}_{\mathrm{ext}}) \quad (1)$$

$$\mathbf{q}^{(t+1)} \leftarrow \mathbf{q}^{(t)} + \Delta t \dot{\mathbf{q}}^{(t+1)}. \quad (2)$$

3.2 Object model with coarse-to-fine surface smoothing

SDF representation. For the purpose of collision detection, the hand is represented by a set of surface points $\mathbf{X}_{\mathrm{h}}$, and the object to grasp is represented by its Signed Distance Function (SDF), $\phi(\mathbf{x})$ (similar to [4, 29, 59]). The SDF maps a spatial position $\mathbf{x} \in \mathbb{R}^3$ to its distance to the closest point on the surface of the object, with a negative or positive sign for interior and exterior points, respectively [68]. The object surface can be recovered as the zero level-set of the SDF: $\{\mathbf{x} | \phi(\mathbf{x}) = 0\}$. The gradient of the SDF $\nabla\phi(\mathbf{x})$ is always of unit magnitude, corresponds to the surface normal for $\mathbf{x}$ on the object surface, and yields the closest point on the object as $\mathbf{x} - \phi(\mathbf{x})\nabla\phi(\mathbf{x})$. SDF representations are well-suited to differentiable collision detection [59], since contact forces can be written in terms of a penetration depth (ϕ) and normal direction ($\nabla\phi$), for which gradients can be computed as $\nabla\phi$ and $\nabla^2\phi$, respectively.

Whereas primitive objects (e.g., a sphere or box) admit an analytic SDF, this is not the case for complex objects, for which an SDF representation is not readily available. We model the object to be grasped by a discretized SDF which we extract from ground truth meshes (easier to come by for most object sets [10, 11]), yielding a 3D grid. Given a query point $\mathbf{x}$, to compute $\phi(\mathbf{x})$ based on the grid, we first convert $\mathbf{x}$ to local shape coordinates (where the object is in canonical pose: unrotated and centered at the origin), yielding $\mathbf{x}_{\mathrm{local}}$. If $\mathbf{x}_{\mathrm{local}}$ falls within the bounds of the grid, we map it to grid indices and compute $\phi(\mathbf{x}_{\mathrm{local}})$ by tri-linear interpolation of neighbouring grid cells. If $\mathbf{x}_{\mathrm{local}}$ falls outside the grid, we clamp it to the grid bounds, yielding $\mathbf{x}_{\mathrm{clamp}}$, and compute $\phi(\mathbf{x}) := \phi(\mathbf{x}_{\mathrm{clamp}}) + \|\mathbf{x} - \mathbf{x}_{\mathrm{clamp}}\|$.

Coarse-to-fine smoothing. To successfully optimize contact locations over non-smooth object geometry we employ surface smoothing in a coarse-to-fine way. At the start of each optimization, we define the object surface *not* as the zero level-set of the SDF, but as the radius r level-set: $\{\mathbf{x}|\phi(\mathbf{x}) = r > 0\}$, which gives a smoothed and padded version of the original surface. As optimization continues, we decrease r on a linear schedule until it reaches 0, yielding the original surface. This coarse-to-fine smoothing allows gradient-based optimization to effectively move contact points across discontinuities and prevents the optimization from quickly overfitting to local geometric features. We set r to approximately 10cm at the start of each optimization. Details are in Appendix A.2.

3.3 Contact dynamics with leaky gradient

Contact forces. We use a primal (penalty-based) formulation of contact forces, which allows us to compute derivatives with autodiff [3] and keep a consistent memory footprint. For a given point $\mathbf{x} \in \mathbf{X}_\mathrm{h}$, the resultant contact force is

$$\begin{aligned}\mathbf{f}_\mathrm{c} &= \mathbf{f}_n + \mathbf{f}_t && (3)\\ \mathbf{f}_n &= k_n \min(\phi(\mathbf{x}),\ 0)\nabla\phi(\mathbf{x}) && (4)\\ \mathbf{f}_t &= -\min(k_f\|\mathbf{v}_t\|,\ \mu\|\mathbf{f}_n\|)\mathbf{v}_t, && (5)\end{aligned}$$

where $\mathbf{f}_n$ is the normal component, proportional to penetration depth $\phi(\mathbf{x})$, and $\mathbf{f}_t$ is the frictional component, computed using a Coulomb friction model. k_n and k_f are the normal and frictional stiffness coefficients, respectively, μ is the friction coefficient, and $\mathbf{v}_t$ is the component of relative velocity between hand and object at the contact point $\mathbf{x}$ that is tangent to the contact normal $\nabla\phi(\mathbf{x})$.

Leaky gradients. At any one time, most possible hand-object contacts are inactive -- a property we refer to as *contact sparsity.* Since an infinitesimal perturbation to hand pose will not activate these contacts (i.e., will not make them touch the object), the gradient of their contact forces with respect to hand pose is zero, i.e., $\partial\mathbf{f}_\mathrm{c}/\partial\mathbf{q} = \partial\mathbf{f}_\mathrm{c}/\partial\dot{\mathbf{q}} = \partial\mathbf{f}_\mathrm{c}/\partial\ddot{\mathbf{q}} = 0$. When the hand is not touching the object, all contacts are inactive and gradient-based optimization can get stuck in a plateau. We work around this by computing a *leaky* gradient for the normal force term. From equation (4), we have $\frac{\partial\|\mathbf{f}_n\|}{\partial\mathbf{q}} = 0$ if $\phi(\mathbf{x}) \geq 0$ but we instead set

$$\frac{\partial\|\mathbf{f}_n\|}{\partial\mathbf{q}} := \begin{cases} k_n\frac{\partial\phi}{\partial\mathbf{q}} & \text{if } \phi(\mathbf{x}) < 0\\ \alpha k_n\frac{\partial\phi}{\partial\mathbf{q}} & \text{otherwise}\end{cases}, \tag{6}$$

where $\alpha \in [0, 1]$ controls how much gradient leaks through the minimum. We set $\alpha = 0.1$ in our experiments.

3.4 Grasping metric and problem relaxation

Simulation setup. To compute the grasp metric, we simulate the rigid-body interaction between a hand and an object. The hand is kinematic (does not react to contact forces), while the object is dynamic (thus subject to contact forces). The simulator state is given by the configuration vector $\mathbf{q}$ and its first and second time derivatives $\dot{\mathbf{q}}, \ddot{\mathbf{q}}$. $\mathbf{q}$ is composed of hand and object components $\mathbf{q} = (\mathbf{q}_\mathrm{h}, \mathbf{q}_\mathrm{o})$ with corresponding spatial coordinates $\mathbf{u} = (\mathbf{u}_\mathrm{h}, \mathbf{u}_\mathrm{obj})$. The object is always initialized with the same configuration $\mathbf{q}_{\mathrm{o}}^{(0)}$: unrotated and untranslated at the origin. Given a state $\mathbf{q}^{(t)}$, following equations (1) and (2), our simulator uses a semi-implicit Euler update scheme to compute subsequent state $\mathbf{q}^{(t+1)}$.

Computing the grasp metric by simulation. To measure the quality of a candidate grasp $\mathbf{q}_{\mathrm{h}}$, we test its ability to withstand forces applied to the object. Given an initial state $\mathbf{q}^{(0)} = (\mathbf{q}_{\mathrm{h}}, \mathbf{q}_{\mathrm{o}}^{(0)})$, we apply an initial velocity $\dot{\mathbf{q}}_{\mathrm{o}}^{(0)}$ to the object. The hand is kept static, with $\dot{\mathbf{u}}_{\mathrm{h}} = 0$. We run forward simulation to compute the object's final velocity $\dot{\mathbf{u}}_{\mathrm{o}}^{(T)}$. A stable grasp will produce contact forces that resist the object velocity, so lower $\|\dot{\mathbf{u}}_{\mathrm{o}}^{(T)}\|$ indicates a more stable grasp. In fact, a stable grasp should be able to resist object velocities in *any* direction, so we perform multiple simulations with different initial velocities and average the results. This suggests the following basic grasp metric: for each set of M simulations, indexed by $m = \{1, \ldots, M\}$, we set a different initial object velocity, run the simulation, and record $L_m = \|\dot{\mathbf{u}}_{\mathrm{o}}^{(T)}\|$. Then, averaging, we have

$$\mathcal{L}_{\mathrm{grasp}} = \sum_{m=1}^{M} \frac{L_m}{M}. \tag{7}$$

Since $\mathcal{L}_{\mathrm{grasp}}$ is a differentiable function of the output of a differentiable simulation, it is itself differentiable with respect to $\mathbf{q}_{\mathrm{h}}$, and we can compute loss gradients $\partial\mathcal{L}_{\mathrm{grasp}}/\partial\mathbf{q}_{\mathrm{h}}$ and use gradient-based optimization to find stable grasps.

Unfortunately, in practice, this basic procedure does not succeed. As explained at the beginning of Section 3, the grasp optimization landscape is extremely rugged, with sharp and narrow ridges, peaks, and valleys. Our leaky contact force gradients (see Section 3.3) provide some help in escaping plateaus, but once the hand is in contact with the object, small changes in hand configuration still cause large jumps in contact forces by making/breaking contacts and shifting contact normals. However, differentiability alone does not resolve this issue.

Problem relaxation. Inspired by Contact-Invariant Optimization [64, 65] we relax the problem making it more forgiving to gradient-based optimization. Specifically, we introduce additional *desired* or *prescribed* contact force variables. This allows us to model physics violations as a cost rather than a constraint. For each surface point on the hand $\mathbf{x}^i \in \mathbf{X}_{\mathrm{h}}$, we introduce a 6-dimensional vector $\widehat{\mathbf{f}}_{\mathrm{c}}^{i}$ representing the desired hand-object contact wrench arising from contact at $\mathbf{x}^i$.

Our overall loss now has two components. The task loss $\mathcal{L}_{\mathrm{task}}(\widehat{\mathbf{f}}_{\mathrm{c}})$ measures whether the prescribed forces $\widehat{\mathbf{f}}_{\mathrm{c}}$ successfully resist initial object velocities. This is computed identically to the previous $\mathcal{L}_{\mathrm{grasp}}$, except that instead of computing contact forces according to equations (3), (4) and (5), contact forces are simply set equal to $\widehat{\mathbf{f}}_{\mathrm{c}}$. The physics violation loss $\mathcal{L}_{\mathrm{phys}}(\mathbf{q}_{\mathrm{h}}, \widehat{\mathbf{f}}_{\mathrm{c}})$ measures whether the hand configuration $\mathbf{q}_{\mathrm{h}}$ actually provides the desired forces $\widehat{\mathbf{f}}_{\mathrm{c}}$. It is computed as

$$\mathcal{L}_{\mathrm{phys}}(\mathbf{q}_{\mathrm{h}}, \widehat{\mathbf{f}}_{\mathrm{c}}) = \|f_{\mathrm{c}}(\mathbf{q}_{\mathrm{h}}) - \widehat{\mathbf{f}}_{\mathrm{c}}\|, \tag{8}$$

where $f_{\mathrm{c}}(\mathbf{q}_{\mathrm{h}})$ is the contact force arising from the hand pose $\mathbf{q}_{\mathrm{h}}$ according to equations (3), (4) and (5).

Intuitively, minimizing these losses corresponds to finding a set of desired forces (as close as possible to the actual contact forces arising from the current hand configuration) that complete the task, and finding a hand configuration that provides those forces. We expect problem formulations derived from and inspired by Contact-Invariant Optimization [13, 14] to be a fruitful area of research as they are made newly attractive by advances in differentiable simulation.

Additional heuristic losses. We include some additional losses that improve the plausibility of resulting grasps. Most hand models have defined joint range limits. Let $\mathbf{q}_\mathrm{h}^\mathrm{low}$ and $\mathbf{q}_\mathrm{h}^\mathrm{up}$ be the lower and upper joint limits respectively. $\mathcal{L}_\mathrm{range}$ encourages hand joints to be near the middle of their ranges. $\mathcal{L}_\mathrm{limit}$ penalizes hand joints outside of their range. $\mathcal{L}_\mathrm{inter}$ penalizes self intersections of the hand.

$$\mathcal{L}_\mathrm{range}(\mathbf{q}_\mathrm{h}) = \|\mathbf{q}_\mathrm{h} - \frac{\mathbf{q}_\mathrm{h}^\mathrm{up} + \mathbf{q}_\mathrm{h}^\mathrm{low}}{2}\| \tag{9}$$

$$\mathcal{L}_\mathrm{limit}(\mathbf{q}_\mathrm{h}) = \max(\mathbf{q}_\mathrm{h} - \mathbf{q}_\mathrm{h}^\mathrm{up}, 0) + \max(\mathbf{q}_\mathrm{h}^\mathrm{low} - \mathbf{q}_\mathrm{h}, 0) \tag{10}$$

$$\mathcal{L}_\mathrm{inter}(\mathbf{q}_\mathrm{h}) = \|\mathbf{f}_\mathrm{link}\|. \tag{11}$$

The hand is kinematic, so it is not subject to contact forces. However, we still compute forces arising from contact between the hand links, for use in this loss term, as $\mathbf{f}_\mathrm{link}$. We ignore contacts between neighbouring links in the chain. For the purpose of computing $\mathbf{f}_\mathrm{link}$, we represent each hand link as both a point set and an SDF and compute $\mathbf{f}_\mathrm{link}$ according to equations (3), (4), and (5).

3.5 Optimization

We use the Modified Differential Multiplier Method [70], treating $\mathcal{L}_\mathrm{task} < C_\mathrm{task}$ and $\mathcal{L}_\mathrm{limit} < C_\mathrm{limit}$ as constraints, while minimizing $\mathcal{L}_\mathrm{phys}$, $\mathcal{L}_\mathrm{limit}$ and $\mathcal{L}_\mathrm{inter}$. We update our parameters $\widehat{\mathbf{f}}_\mathrm{c}$ and $\mathbf{q}_\mathrm{h}$ using the Adamax [50] optimizer. Details of learning rates, C_task and C_limit can be found in Appendix A.7.

4 Experiments

Our evaluations and analysis of Grasp'D answer the following questions:

1. How well does Grasp'D perform compared to analytic methods? (Section 4.2)
2. Can Grasp'D generalize to objects reconstructed from real-world RGBD images? (Section 4.3)
3. How much do coarse-to-fine SDF collision and the problem relaxation contribute to final performance? (Section 4.4)

4.1 Experimental setup

For each experiment, we synthesize grasps following the procedure described in Section 3. We compute the metric with $M = 3$ simulations: each setting a different initial velocity on the hand: $(0, 0, 0)$, $(0.01, 0.01, 0.01)$ or $(-0.01, -0.01, -0.01)\mathrm{m/s}$. Each simulation is run for a single timestep of length 1×10^{-5}s.

Evaluation metrics. We follow [39] and use contact area (CA), intersection volume (IV), and the ratio between contact area and intersection volume ($\frac{\mathrm{CA}}{\mathrm{IV}}$). We compute evaluation metrics that measure grasp stability and contact area.

Method	**CA**$\uparrow$	**IV**$\downarrow$	$\frac{\textbf{CA}}{\textbf{IV}}\uparrow$	$\epsilon\uparrow$	**Vol** $\uparrow$	**SD** $\downarrow$
Scale (Unit)	cm^2	cm^3	cm^{-1}	$\times10^{-1}$	$\times10^{1}$	cm
ObMan [39] (top2)	9.4	1.28	7.37	4.70	1.36	1.95
ObMan [39] (top5)	7.8	**1.05**	7.37	4.52	1.36	2.22
Grasp'D (top2)	**43.0**	5.70	**7.55**	5.01	1.44	**0.59**
Grasp'D (top5)	41.4	5.48	**7.55**	**5.02**	**1.46**	1.04

Table 2: Experimental results. We synthesize MANO hand grasps for ShapeNet objects. Our grasps achieve over 4× denser contact (as measured by contact surface area - CA) than an analytic synthesis baseline [39], leading to significantly higher grasp stability (4× lower simulation displacement - SD). Higher contact does result in higher interpenetration, but we keep a similar ratio of contact area to interpenetration volume.

In addition, we measure the contact area each grasp creates and the volume of hand-object interpenetration. We compute two analytic measures of stability -- the Ferrari-Canny (epsilon ϵ) [27] and the volume metric (Vol) -- and one simulated measure: the simulation displacement (SD) metric introduced in [39].

Hand parameterization. We use a differentiable PyTorch layer [39] to compute the 773 vertices of the MANO hand [75] model. The input is a set of weights for principal components extracted from the MANO dataset of human scans [75]. We find that this PCA parameterization provides a useful prior for human-like hand poses. We use the maximum number of principal components (44).

4.2 Grasp synthesis with ShapeNet models

We compare to baseline grasps from the ObMan [39] dataset, which generates grasps with the GraspIt! [63] simulator using an analytic metric. We report these metrics over the top-2 and top-5 grasps per scaled object, with ranking decided by simulation displacement for our method and by ObMan's heuristic measure (detailed in Appendix C.2 of [39]) for theirs. Further details in Appendix A.6.

Data. We evaluate our approach to grasp synthesis by generating grasps with the MANO human hand [75] model for 57 ShapeNet [11] objects that span 8 categories (bottles, bowls, cameras, cans, cellphones, jars, knives, remote controls), and are each considered at 5 different scales (as in ObMan). See the Appendix A for details of mesh pre-processing, initialization, simulation, and optimization.

Results. Results are presented in Table 2. Grasps generated by our method (both top-2 and top-5) have a contact area of around 42cm^2. This is higher than the $\sim$ 20cm^2 area achieved with fingertip only grasps [7] and about 4× higher than grasps from the ObMan dataset (top-2 or top-5). These contact-rich grasps achieve modest improvements in analytic measures of stability, and a significant reduction in simulation displacement ($\sim$ 3× for top-2 grasps). Visualizations of our generated grasps in Figure 1 confirm that these grasps achieve larger areas of contact by closely conforming to object surface geometry, whereas the analytically generated grasps largely make use of fingertip contact only. These higher contact grasps have accordingly higher interpenetration, but the ratio between contact area and intersection volume is similar to the ObMan baseline.

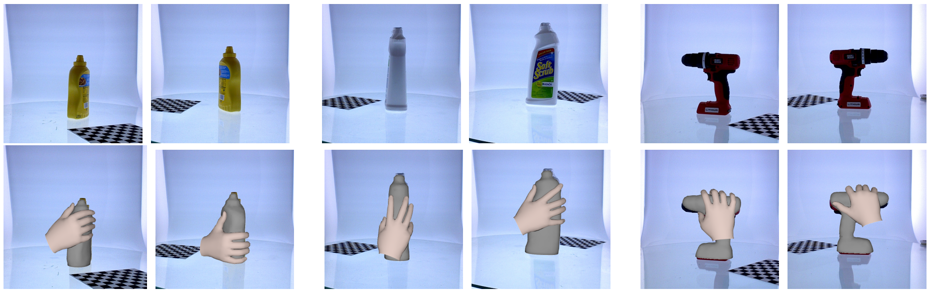

Fig. 4: Grasp synthesis from RGB-D. We use RGB-D captures from the YCB dataset [10] (top row) to reconstruct object models from which we synthesize grasps (bottom row). Our method can synthesize plausible grasps not just from ground truth object models, but also from imperfect reconstructions.

4.3 Grasp synthesis from RGB-D input of unknown objects

Setting. One possible application of our method is to direct grasp prediction from RGB-D images by simulation on reconstructed object models. Currently, our method is too slow to be used online (about 5 minutes per grasp), but as simulation speeds increase and recent works in implicit fields push reconstruction accuracy higher and higher, we believe that grasp prediction by simulation models will become increasingly viable. To validate the plausibility of using our method with reconstructed object models, we present results from running our system on meshes reconstructed from RGB-D inputs. We synthesize grasps based on RGB-D (with camera pose) inputs from the YCB object dataset [10]. In addition to reconstructed meshes, the YCB dataset provides the original RGB-D captures the meshes are based on. Each object was captured from 5 different cameras at 120 different angles for a total of 600 images. To confirm that our method can work with reconstructions done under more realistic assumptions, we limit our reconstructions to using 5 different angles from 3 cameras (2.5% of captures).

Data. For a subset of the YCB objects, we generate Poisson surface reconstructions and use our method to synthesize MANO hand grasps. Since the inputs are from cameras with a known pose, the object reconstruction is in the world frame. Details in the Appendix A.4.

Results. Our results confirm the viability of using simulation to synthesize grasps on reconstructed object models. Qualitative results are presented in Figure 4; additional results can be found in Appendix D. Although synthesis does not perform as well as with ground-truth models, plausible human grasps are discovered for many objects and the grasps appear well-aligned with the real-world object poses. Future work could take advantage of learning-based reconstruction methods to achieve grasp synthesis with fewer input images.

4.4 Ablation study

We investigate the impact of our coarse-to-fine smoothing (Section 3.2), leaky contact force gradients (Section 3.3), and relaxed problem formulation (Section 3.4). We generate MANO hand grasps on 21 objects from the YCB dataset [10]. *Grasp'D w/o coarse-to-fine* does not pad or smooth the object. *Grasp'D w/o problem relaxation* attempts to solve the problem without intro-

Method	**CA**↑	**IV**↓	$\frac{\textbf{CA}}{\textbf{IV}}$↑	ϵ ↑	**Vol** ↑	**SD** ↓
Scale/Unit	cm^2	cm^3	cm^{-1}	$\times 10^{-1}$	$\times 10^{1}$	cm
Grasp'D	42.6	2.83	15.1	2.38	20.6	0.41
Grasp'D w/o coarse-to-fine	43.2	2.84	15.2	2.37	20.7	0.55
Grasp'D w/o problem relaxation	6.1	0.40	15.2	0.52	4.0	3.82

Table 3: Ablation study. We validate our design choices with an ablation study. Our relaxed problem formulation has a large positive impact on all metrics. The quantitative impact of coarse-to-fine smoothing is more limited, but we observe a qualitative difference in grasps generated with and without smoothing.

ducing additional force variables or a relaxed objective. This amounts to the ‘‘basic procedure’’ described in Section 3.4, i.e., directly optimize over hand pose to minimize $\mathcal{L}_{\text{grasp}}$ and the heuristic losses.

Results. We adopt the same data as in Section 4.2. Table 3 presents the results. Our relaxed problem formulation is key to our method's success, and without it, performance greatly degrades by all measures, with discovered grasps creating very little contact (low contact area and intersection volume). Coarse-to-fine smoothing has a modest impact, with all metrics comparable with or without smoothing, except for simulation displacement, which is about 25% higher without smoothing. We did not include a variant without leaky gradient, since this variant would never make contact with the object (if the hand is not touching the object at initialization, there will be no gradient to follow and optimization will immediately be stuck in a plateau).

5 Conclusions

We presented a simulation-based grasp synthesis pipeline capable of generating large datasets of plausible, high-contact grasps. By being differentiable, our simulator is amenable to gradient-based optimization, allowing us to produce high-quality grasps, even for multi-finger grippers, while scaling to thousands of dense contacts. Our experiments have shown that we outperform the existing classical grasping algorithm both quantitatively and qualitatively. Our approach is compatible with PyTorch and can be easily integrated into existing pipelines. More importantly, the produced grasps can directly benefit any vision pipeline that learns grasp prediction from synthetic data.

Acknowledgements. DT was supported in part by a Vector research grant. The authors appreciate the support of NSERC, Vector Institute and Samsung AI. AG was also supported by NSERC Discovery Grant, NSERC Exploration Grant, CIFAR AI Chair, XSeed Discovery Grant from University of Toronto.

Bibliography

[1] Allshire, A., Mittal, M., Lodaya, V., Makoviychuk, V., Makoviichuk, D., Widmaier, F., Wüthrich, M., Bauer, S., Handa, A., Garg, A.: Transferring dexterous manipulation from gpu simulation to a remote real-world trifinger. arXiv preprint arXiv:2108.09779 (2021) 5

[2] Baek, S., Kim, K.I., Kim, T.K.: Pushing the envelope for rgb-based dense 3d hand pose estimation via neural rendering. In: CVPR (2019) 2

[3] Baydin, A.G., Pearlmutter, B.A., Radul, A.A., Siskind, J.M.: Automatic differentiation in machine learning: a survey. JMLR (2018) 9

[4] Bender, J., Duriez, C., Jaillet, F., Zachmann, G.: Continuous collision detection between points and signed distance fields. In: Workshop on Virtual Reality Interaction and Physical Simulation (2014) 8

[5] Bender, J., Erleben, K., Trinkle, J.: Interactive simulation of rigid body dynamics in computer graphics. Computer Graphics Forum (2014) 8

[6] Boukhayma, A., Bem, R.d., Torr, P.H.: 3d hand shape and pose from images in the wild. In: CVPR (2019) 2

[7] Brahmbhatt, S., Ham, C., Kemp, C.C., Hays, J.: Contactdb: Analyzing and predicting grasp contact via thermal imaging. In: CVPR (2019) 2, 3, 5, 12

[8] Brahmbhatt, S., Handa, A., Hays, J., Fox, D.: Contactgrasp: Functional multi-finger grasp synthesis from contact. In: IROS (2019) 5

[9] Brahmbhatt, S., Tang, C., Twigg, C.D., Kemp, C.C., Hays, J.: Contactpose: A dataset of grasps with object contact and hand pose. In: ECCV (2020) 3, 5

[10] Calli, B., Singh, A., Bruce, J., Walsman, A., Konolige, K., Srinivasa, S., Abbeel, P., Dollar, A.M.: Yale-cmu-berkeley dataset for robotic manipulation research. International Journal of Robotics Research (2017) 4, 5, 8, 13

[11] Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015) 4, 8, 12

[12] Chen, T., Xu, J., Agrawal, P.: A system for general in-hand object re-orientation. In: CoRL (2022) 5

[13] Ciocarlie, M., Goldfeder, C., Allen, P.: Dexterous grasping via eigengrasps: A low-dimensional approach to a high-complexity problem. In: RSS (2007) 5, 11

[14] Ciocarlie, M.T., Allen, P.K.: Hand posture subspaces for dexterous robotic grasping. International Journal of Robotics Research (2009) 11

[15] Corona, E., Pumarola, A., Alenya, G., Moreno-Noguer, F., Rogez, G.: Ganhand: Predicting human grasp affordances in multi-object scenes. In: CVPR (2020) 3, 5

[16] Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR (2005) 2

[17] Danielczuk, M., Xu, J., Mahler, J., Matl, M., Chentanez, N., Goldberg, K.: Reach: Reducing false negatives in robot grasp planning with a robust efficient area contact hypothesis model. In: International Symposium of Robotic Research (2019) 4, 6

[18] De Giorgio, A., Romero, M., Onori, M., Wang, L.: Human-machine collaboration in virtual reality for adaptive production engineering. Procedia Manufacturing (2017) 2

[19] Depierre, A., Dellandréa, E., Chen, L.: Jacquard: A large scale dataset for robotic grasp detection. In: IROS (2018) 3, 4

[20] Do, T.T., Nguyen, A., Reid, I.: Affordancenet: An end-to-end deep learning approach for object affordance detection. In: ICRA (2018) 2

[21] Doosti, B., Naha, S., Mirbagheri, M., Crandall, D.J.: Hope-net: A graph-based model for hand-object pose estimation. In: CVPR (2020) 3, 5

[22] Duan, K., Bai, S., Xie, L., Qi, H., Huang, Q., Tian, Q.: Centernet: Keypoint triplets for object detection. In: ICCV (2019) 2

[23] Eppner, C., Mousavian, A., Fox, D.: A billion ways to grasp: An evaluation of grasp sampling schemes on a dense, physics-based grasp data set. arXiv preprint arXiv:1912.05604 (2019) 3

[24] Eppner, C., Mousavian, A., Fox, D.: Acronym: A large-scale grasp dataset based on simulation. In: ICRA (2021) 2, 3, 4, 5, 6

[25] Fang, K., Zhu, Y., Garg, A., Kurenkov, A., Mehta, V., Fei-Fei, L., Savarese, S.: Learning task-oriented grasping for tool manipulation from simulated self-supervision. International Journal of Robotics Research (IJRR) (2019) 2

[26] Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. TPAMI (2009) 2

[27] Ferrari, C., Canny, J.F.: Planning optimal grasps. In: ICRA (1992) 2, 4, 6, 12

[28] Freeman, C.D., Frey, E., Raichuk, A., Girgin, S., Mordatch, I., Bachem, O.: Brax - a differentiable physics engine for large scale rigid body simulation (2021), http://github.com/google/brax 6

[29] Fuhrmann, A., Sobotka, G., Groß, C.: Distance fields for rapid collision detection in physically based modeling. In: Proceedings of GraphiCon (2003) 8

[30] Gammieri, L., Schumann, M., Pelliccia, L., Di Gironimo, G., Klimant, P.: Coupling of a redundant manipulator with a virtual reality environment to enhance human-robot cooperation. Procedia Cirp (2017) 2

[31] Garcia-Hernando, G., Yuan, S., Baek, S., Kim, T.K.: First-person hand action benchmark with rgb-d videos and 3d hand pose annotations. In: CVPR (2018) 5

[32] Ge, L., Ren, Z., Li, Y., Xue, Z., Wang, Y., Cai, J., Yuan, J.: 3d hand shape and pose estimation from a single rgb image. In: CVPR (2019) 2

[33] Geilinger, M., Hahn, D., Zehnder, J., Bächer, M., Thomaszewski, B., Coros, S.: Add: Analytically differentiable dynamics for multi-body systems with frictional contact. ACM Transactions on Graphics (2020) 6

[34] Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014) 2

[35] Goldfeder, C., Ciocarlie, M., Dang, H., Allen, P.K.: The columbia grasp database. In: ICRA (2009) 4

[36] Grady, P., Tang, C., Twigg, C.D., Vo, M., Brahmbhatt, S., Kemp, C.C.: Contactopt: Optimizing contact to improve grasps. In: CVPR (2021) 6

[37] Hamer, H., Schindler, K., Koller-Meier, E., Van Gool, L.: Tracking a hand manipulating an object. In: ICCV (2009) 2

[38] Hampali, S., Rad, M., Oberweger, M., Lepetit, V.: Honnotate: A method for 3d annotation of hand and object poses. In: CVPR (2020) 5

[39] Hasson, Y., Varol, G., Tzionas, D., Kalevatykh, I., Black, M.J., Laptev, I., Schmid, C.: Learning joint reconstruction of hands and manipulated objects. In: CVPR (2019) 1, 3, 4, 5, 11, 12

[40] Heiden, E., Macklin, M., Narang, Y.S., Fox, D., Garg, A., Ramos, F.: DiSECt: A Differentiable Simulation Engine for Autonomous Robotic Cutting. In: RSS (2021) 6

[41] Heiden, E., Millard, D., Coumans, E., Sheng, Y., Sukhatme, G.S.: NeuralSim: Augmenting differentiable simulators with neural networks. In: ICRA (2021) 6

[42] Heumer, G., Amor, H.B., Weber, M., Jung, B.: Grasp recognition with uncalibrated data gloves-a comparison of classification methods. In: IEEE Virtual Reality Conference (2007) 2

[43] Hu, Y., Anderson, L., Li, T.M., Sun, Q., Carr, N., Ragan-Kelley, J., Durand, F.: Difftaichi: Differentiable programming for physical simulation. In: ICLR (2020) 6

[44] Hu, Y., Liu, J., Spielberg, A., Tenenbaum, J.B., Freeman, W.T., Wu, J., Rus, D., Matusik, W.: Chainqueen: A real-time differentiable physical simulator for soft robotics. In: ICRA (2019) 6

[45] Huang, D.A., Ma, M., Ma, W.C., Kitani, K.M.: How do we use our hands? discovering a diverse set of common grasps. In: CVPR (2015) 2

[46] Jiang, H., Liu, S., Wang, J., Wang, X.: Hand-object contact consistency reasoning for human grasps generation. In: ICCV (2021) 3, 5, 6

[47] Jiang, Z., Zhu, Y., Svetlik, M., Fang, K., Zhu, Y.: Synergies between affordance and geometry: 6-dof grasp detection via implicit representations. arXiv preprint arXiv:2104.01542 (2021) 5

[48] Kappler, D., Bohg, J., Schaal, S.: Leveraging big data for grasp planning. In: ICRA (2015) 2, 4, 5, 6

[49] Karunratanakul, K., Yang, J., Zhang, Y., Black, M.J., Muandet, K., Tang, S.: Grasping field: Learning implicit representations for human grasps. In: 3DV (2020) 3, 5

[50] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2014) 11

[51] Kokic, M., Kragic, D., Bohg, J.: Learning task-oriented grasping from human activity datasets. IEEE Robotics and Automation Letters (2020) 2

[52] Lakshmipathy, A., Bauer, D., Bauer, C., Pollard, N.S.: Contact transfer: A direct, user-driven method for human to robot transfer of grasps and manipulations. In: 2022 International Conference on Robotics and Automation (ICRA). pp. 6195--6201. IEEE (2022) 5

[53] Lau, M., Dev, K., Shi, W., Dorsey, J., Rushmeier, H.: Tactile mesh saliency. ACM Transactions on Graphics (2016) 2

[54] Liu, M., Pan, Z., Xu, K., Ganguly, K., Manocha, D.: Deep differentiable grasp planner for high-dof grippers. arXiv preprint arXiv:2002.01530 (2020) 6

[55] Liu, T., Liu, Z., Jiao, Z., Zhu, Y., Zhu, S.C.: Synthesizing diverse and physically stable grasps with arbitrary hand structures using differentiable force closure estimator. IEEE Robotics and Automation Letters (2021) 6

[56] Lundell, J., Corona, E., Le, T.N., Verdoja, F., Weinzaepfel, P., Rogez, G., Moreno-Noguer, F., Kyrki, V.: Multi-fingan: Generative coarse-to-fine sampling of multi-finger grasps. arXiv preprint arXiv:2012.09696 (2020) 3, 5

[57] Lundell, J., Verdoja, F., Kyrki, V.: Ddgc: Generative deep dexterous grasping in clutter. IEEE Robotics and Automation Letters (2021) 3, 5

[58] Maas, A.L., Hannun, A.Y., Ng, A.Y., et al.: Rectifier nonlinearities improve neural network acoustic models. In: ICML (2013) 3

[59] Macklin, M., Erleben, K., Müller, M., Chentanez, N., Jeschke, S., Corse, Z.: Local optimization for robust signed distance field collision. ACM on Computer Graphics and Interactive Techniques (2020) 8

[60] Mahler, J., Liang, J., Niyaz, S., Laskey, M., Doan, R., Liu, X., Ojea, J.A., Goldberg, K.: Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. arXiv preprint arXiv:1703.09312 (2017) 4

[61] Mahler, J., Matl, M., Satish, V., Danielczuk, M., DeRose, B., McKinley, S., Goldberg, K.: Learning ambidextrous robot grasping policies. Science Robotics (2019) 5

[62] Mahler, J., Pokorny, F.T., McCarthy, Z., van der Stappen, A.F., Goldberg, K.: Energy-bounded caging: Formal definition and 2-d energy lower bound algorithm based on weighted alpha shapes. IEEE Robotics and Automation Letters (2016) 4

[63] Miller, A.T., Allen, P.K.: Graspit! a versatile simulator for robotic grasping. IEEE Robotics & Automation Magazine (2004) 1, 3, 4, 5, 12

[64] Mordatch, I., Popović, Z., Todorov, E.: Contact-invariant optimization for hand manipulation. In: ACM SIGGRAPH/Eurographics symposium on computer animation (2012) 3, 8, 10

[65] Mordatch, I., Todorov, E., Popović, Z.: Discovery of complex behaviors through contact-invariant optimization. ACM Transactions on Graphics (2012) 3, 8, 10

[66] Mousavian, A., Eppner, C., Fox, D.: 6-dof graspnet: Variational grasp generation for object manipulation. In: ICCV (2019) 3, 4, 5, 6

[67] Nakamura, Y.C., Troniak, D.M., Rodriguez, A., Mason, M.T., Pollard, N.S.: The complexities of grasping in the wild. In: International Conference on Humanoid Robotics (2017) 2

[68] Osher, S., Fedkiw, R.: Level Set Methods and Dynamic Implicit Surfaces, vol. 153. Springer Science & Business Media (2006) 8

[69] Pirk, S., Krs, V., Hu, K., Rajasekaran, S.D., Kang, H., Yoshiyasu, Y., Benes, B., Guibas, L.J.: Understanding and exploiting object interaction landscapes. ACM Transactions on Graphics (2017) 2

[70] Platt, J.C., Barr, A.H.: Constrained differential optimization. In: NeurIPS (1987) 11

[71] Pokorny, F.T., Kragic, D.: Classical grasp quality evaluation: New algorithms and theory. In: IROS (2013) 3

[72] Porzi, L., Bulo, S.R., Penate-Sanchez, A., Ricci, E., Moreno-Noguer, F.: Learning depth-aware deep representations for robotic perception. IEEE Robotics and Automation Letters (2016) 2

[73] Qiao, Y.L., Liang, J., Koltun, V., Lin, M.C.: Efficient differentiable simulation of articulated bodies. In: ICML (2021) 6

[74] Rodriguez, A., Mason, M.T., Ferry, S.: From caging to grasping. International Journal of Robotics Research (2012) 4

[75] Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. ACM Transactions on Graphics (2017) 5, 6, 12

[76] Roy, A., Todorovic, S.: A multi-scale cnn for affordance segmentation in rgb images. In: ECCV (2016) 2

[77] Saxena, A., Driemeyer, J., Kearns, J., Ng, A.: Robotic grasping of novel objects. Advances in neural information processing systems **19** (2006) 2

[78] Shao, L., Ferreira, F., Jorda, M., Nambiar, V., Luo, J., Solowjow, E., Ojea, J.A., Khatib, O., Bohg, J.: Unigrasp: Learning a unified model to grasp with multifingered robotic hands. IEEE Robotics and Automation Letters (2020) 3, 5

[79] Sundermeyer, M., Mousavian, A., Triebel, R., Fox, D.: Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes. In: ICRA (2021) 5

[80] Supančič, J.S., Rogez, G., Yang, Y., Shotton, J., Ramanan, D.: Depth-based hand pose estimation: methods, data, and challenges. IJCV (2018) 2

[81] Taheri, O., Ghorbani, N., Black, M.J., Tzionas, D.: Grab: A dataset of whole-body human grasping of objects. In: ECCV (2020) 3, 5

[82] Turpin, D., Wang, L., Tsogkas, S., Dickinson, S., Garg, A.: GIFT: Generalizable Interaction-aware Functional Tool Affordances without Labels. In: Robotics: Systems and Science (RSS) (2021) 2

[83] Tzionas, D., Ballan, L., Srikantha, A., Aponte, P., Pollefeys, M., Gall, J.: Capturing hands in action using discriminative salient points and physics simulation. IJCV (2016) 5

[84] Veres, M., Moussa, M., Taylor, G.W.: An integrated simulator and dataset that combines grasping and vision for deep learning. arXiv preprint arXiv:1702.02103 (2017) 3

[85] Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: CVPR (2001) 2

[86] Weisz, J., Allen, P.K.: Pose error robust grasping from contact wrench space metrics. In: ICRA (2012) 4

[87] Werling, K., Omens, D., Lee, J., Exarchos, I., Liu, C.K.: Fast and feature-complete differentiable physics for articulated rigid bodies with contact. arXiv preprint arXiv:2103.16021 (2021) 6

[88] Xu, J., Makoviychuk, V., Narang, Y., Ramos, F., Matusik, W., Garg, A., Macklin, M.: Accelerated Policy Learning with Parallel Differentiable Simulation. In: International Conference on Learning Representations (ICLR) (2022) 6

[89] Yang, Y., Fermuller, C., Li, Y., Aloimonos, Y.: Grasp type revisited: A modern perspective on a classical feature for vision. In: CVPR (2015) 2

[90] Zhou, Y., Hauser, K.: 6dof grasp planning by optimizing a deep learning scoring function. In: RSS (2017) 2

[91] Zimmermann, C., Brox, T.: Learning to estimate 3d hand pose from single rgb images. In: ICCV (2017) 2