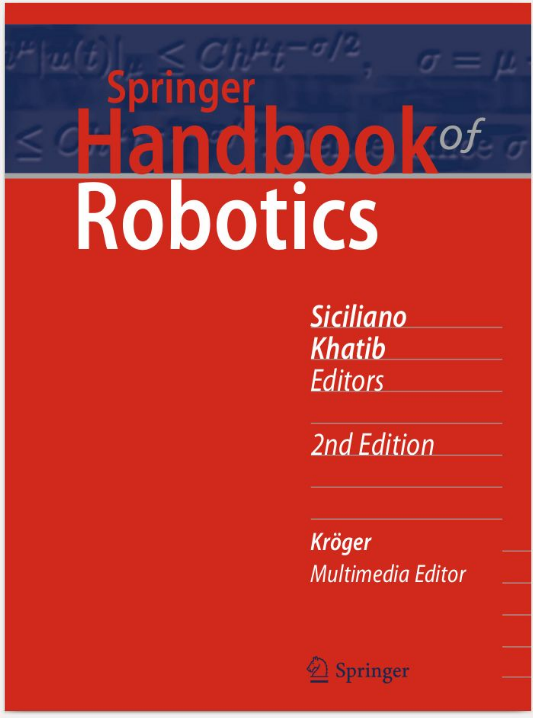
Humanoid robots - Manipulation and Grasping

Doc. Mgr. Matěj Hoffmann, Ph.D.

Manipulation

- “Prehensile manipulation” - grasping. (CZ: prehensile ~ “chápavý”)
- “Nonprehensile manipulation” - everything else you can do with your hands (manus in latin)
    - pushing
    - rolling
    - throwing
    - catching
    - tapping
    - etc.



Multimedia Contents

Part D Manipulation and Interfaces

Ed. by Makoto Kaneko

36 Motion for Manipulation Tasks
James Kuffner, Pittsburgh, USA
Jing Xiao, Charlotte, USA

37 Contact Modeling and Manipulation
Imin Kao, Stony Brook, USA
Kevin M. Lynch, Evanston, USA
Joel W. Burdick, Pasadena, USA

38 Grasping
Domenico Prattichizzo, Siena, Italy
Jeffrey C. Trinkle, Troy, USA

39 Cooperative Manipulation
Fabrizio Caccavale, Potenza, Italy
Masaru Uchiyama, Sendai, Japan

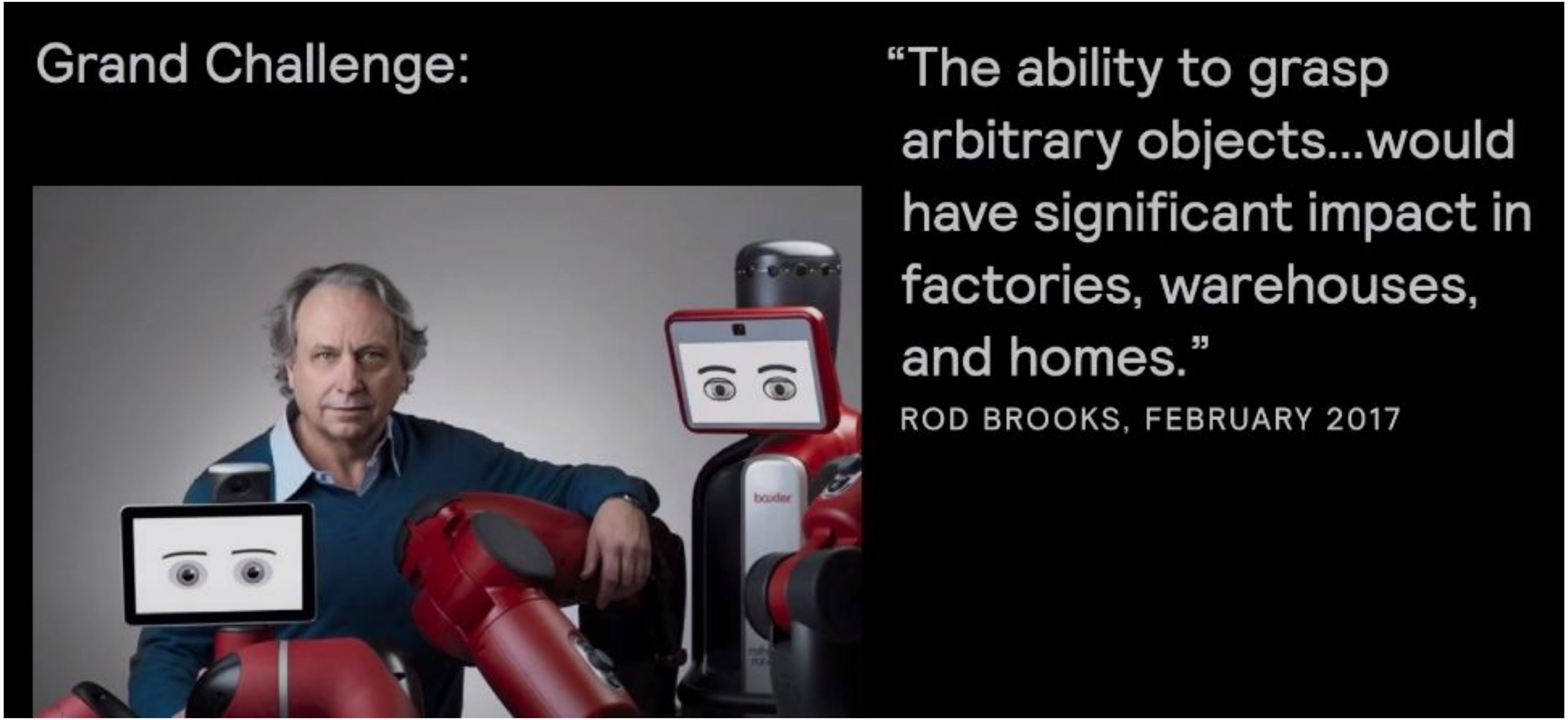
40 Mobility and Manipulation
Oliver Brock, Berlin, Germany
Jaeheung Park, Suwon, Korea
Marc Toussaint, Stuttgart, Germany

41 Active Manipulation for Perception
Anna Petrovskaya, Stanford, USA
Kaijen Hsiao, Palo Alto, USA

42 Haptics
Blake Hannaford, Seattle, USA
Allison M. Okamura, Stanford, USA

43 Telerobotics
Günter Niemeyer, Glendale, USA
Carsten Preusche, Wessling, Germany
Stefano Stramigioli, Enschede, The Netherlands
Dongjun Lee, Seoul, Korea

44 Networked Robots
Dezhen Song, College Station, USA
Ken Goldberg, Berkeley, USA
Nak-Young Chong, Ishikawa, Japan



Slide taken from Ken Goldberg - The New Wave in Robot Grasping: https://youtu.be/ATDrSWZXuwk

Universal picking challenge

Pictures from Ken Goldberg - The New Wave in Robot Grasping: https://youtu.be/ATDrSWZXuwk

Contact joints

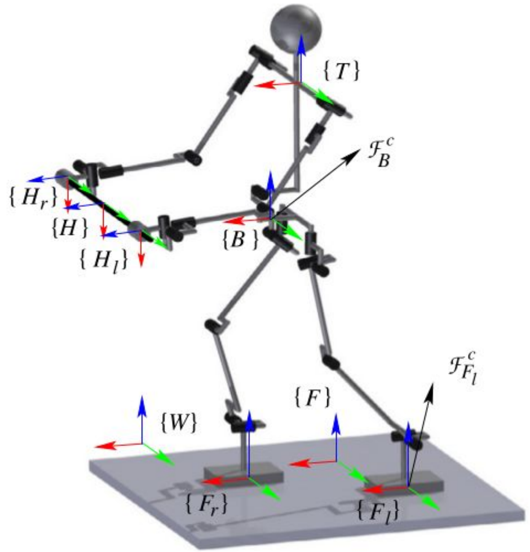


FIGURE 2.11 Closed loops are formed via contact joints at the feet and hands. Contact coordinate frames $\{k\}$, $k \in \{e_r, e_l\}$, $e \in \{H, F\}$ are fixed at the center of pressure (CoP) to the common loop-closure link (floor F and rod H). The z-axes at the feet (shown in blue color) point in a way s.t. the *reaction* force at the contact is always nonnegative. The contact constraints in the vertical direction at the feet are unilateral while those in the angular tangential directions are bilateral, with bounds. All contact constraints at the hands are bilateral.

Section 2.9 in Nenchev, D. N., Konno, A., & Tsujita, T. (2018). Humanoid robots: Modeling and control. Butterworth-Heinemann.

Contact joints

2.9.3 Kinematic Models of Frictionless Contact Joints

Denote by $\bar{\mathcal{V}}_k^m \in \Re^{\eta_k}$ the first-order instantaneous motion components along the unconstrained-motion directions at contact joint k. These components determine the contact joint twist, i.e.

$$\mathcal{V}_k = {}^k\mathbb{B}_m \bar{\mathcal{V}}_k^m. \tag{2.62}$$

Here ${}^k\mathbb{B}_m \in \Re^{6\times\eta_k}$ is a transform that comprises orthonormal basis vectors for the twist components in the unconstrained motion directions.[2] There is a complementary transform s.t. ${}^k\mathbb{B}_m \oplus {}^k\mathbb{B}_c = \boldsymbol{E}_6$ ($\oplus$ denotes the direct sum operator):

$$\mathcal{V}_k = {}^k\mathbb{B}_c \bar{\mathcal{V}}_k^c. \tag{2.63}$$

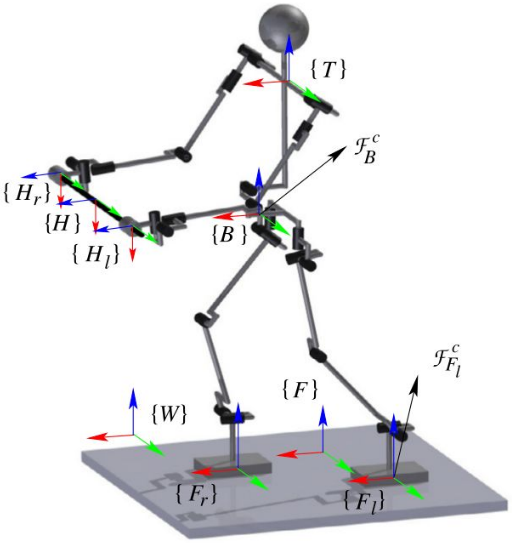
Here $\bar{\mathcal{V}}_k^c$ comprises first-order instantaneous motion components in the constrained motion directions. In the above notations (and throughout this text), the overbar notation signifies a restricted quantity, i.e.

$$\bar{\mathcal{V}}_k^m = N({}^k\mathbb{B}_c)\mathcal{V}_k = {}^k\mathbb{B}_m^T \mathcal{V}_k, \tag{2.64}$$

$$\bar{\mathcal{V}}_k^c = N({}^k\mathbb{B}_m)\mathcal{V}_k = {}^k\mathbb{B}_c^T \mathcal{V}_k. \tag{2.65}$$

These relations imply that

$$\begin{bmatrix} \bar{\mathcal{V}}_k^c \\ \bar{\mathcal{V}}_k^m \end{bmatrix} = \begin{bmatrix} {}^k\mathbb{B}_c^T \\ {}^k\mathbb{B}_m^T \end{bmatrix} \mathcal{V}_k, \quad \bar{\mathcal{V}}_k^c \perp \bar{\mathcal{V}}_k^m. \tag{2.66}$$



In the example in Fig. 2.11, the frictionless cylindrical contact joints at the hands determine

$${}^{H_j}\mathbb{B}_m = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad \bar{\mathcal{V}}_{H_j}^m = \begin{bmatrix} v_y \\ \omega_y \end{bmatrix}. \tag{2.68}$$

The frictionless plane-contact joints at the feet, on the other hand, are modeled with

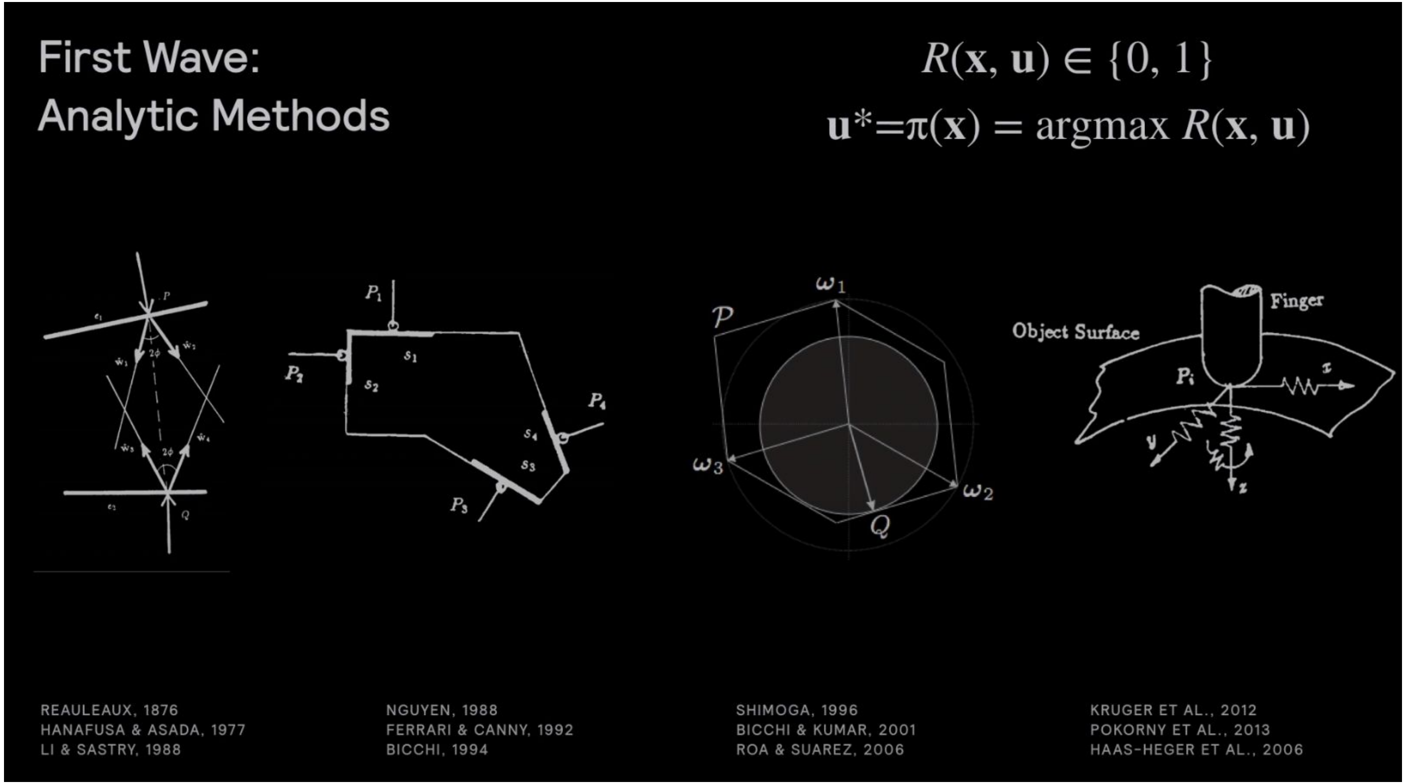
$${}^{F_j}\mathbb{B}_m = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \bar{\mathcal{V}}_{F_j}^m = \begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix}. \tag{2.69}$$

Section 2.9 in Nenchev, D. N., Konno, A., & Tsujita, T. (2018). Humanoid robots: Modeling and control. Butterworth-Heinemann.

Outline

1. Contact kinematics
    a. Form closure
2. Contact forces and friction
    a. Force closure
3. Grasp quality metrics
4. Sampling-based and data-driven grasp planning

“First wave” - grasping from first principles



Ken Goldberg - The New Wave in Robot Grasping: https://youtu.be/ATDrSWZXuwk

Contact kinematics

- study of how two or more rigid bodies can move relative to each other while respecting the *impenetrability constraint*.
- motion at a contact
  - breaking
  - sliding
  - rolling (sticking)

Analysis of single contact

Consider two rigid bodies whose configurations are given by the local coordinate column vectors q_1 and q_2, respectively. Writing the composite configuration as $q = (q_1, q_2)$, we define a distance function $d(q)$ between the bodies that is positive when they are separated, zero when they are touching, and negative when they are in penetration. When $d(q) > 0$, there are no constraints on the motions of the bodies; each is free to move with six degrees of freedom. When the bodies are in contact ($d(q) = 0$), we look at the time derivatives $\dot{d}$, $\ddot{d}$, etc., to determine whether the bodies stay in contact or break apart as they follow a particular trajectory $q(t)$. This can be determined by the following table of possibilities:

d	$\dot{d}$	$\ddot{d}$	$\cdots$	
> 0				no contact
< 0				infeasible (penetration)
$= 0$	> 0			in contact, but breaking free
$= 0$	< 0			infeasible (penetration)
$= 0$	$= 0$	> 0		in contact, but breaking free
$= 0$	$= 0$	< 0		infeasible (penetration)
etc.				

The contact is maintained only if all time derivatives are zero.

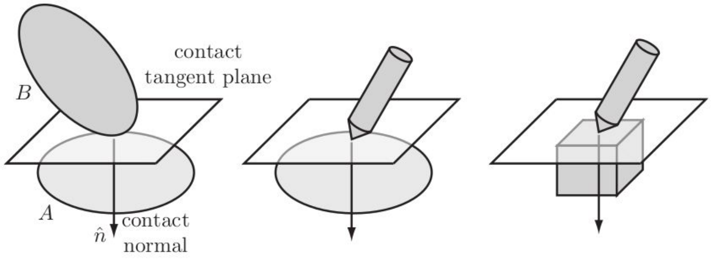


Figure 12.2: (Left) The bodies A and B in single-point contact define a contact tangent plane and a contact normal vector $\hat{n}$ perpendicular to the tangent plane. By default, the positive direction of the normal is chosen into body A. Since contact curvature is not addressed in this chapter, the contact places the same restrictions on the motions of the rigid bodies in the middle and right panels.

12.1.1 in Lynch, K. M., & Park, F. C. (2017). Modern robotics. Cambridge University Press.
https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-1-1-first-order-analysis-of-a-single-contact/

First-order analysis

Now let's assume that the two bodies are initially in contact ($d = 0$) at a single point. The first two time derivatives of d are written

$$\dot{d} = \frac{\partial d}{\partial q}\dot{q}, \tag{12.1}$$

$$\cancel{\ddot{d} = \dot{q}^{\mathrm{T}}\frac{\partial^2 d}{\partial q^2}\dot{q} + \frac{\partial d}{\partial q}\ddot{q}.} \tag{12.2}$$

The terms $\partial d/\partial q$ and $\partial^2 d/\partial q^2$ carry information about the local contact geometry. The gradient vector $\partial d/\partial q$ corresponds to the separation direction in q space associated with the **contact normal** (Figure 12.2). The matrix $\partial^2 d/\partial q^2$ encodes information about the relative curvature of the bodies at the contact point.

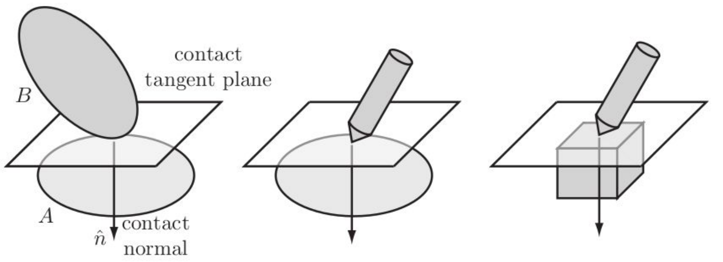
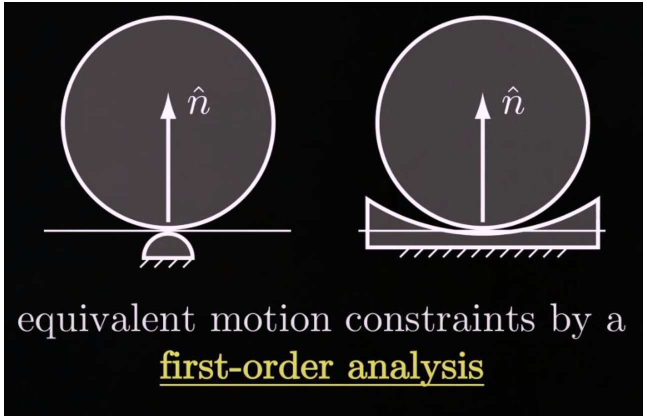
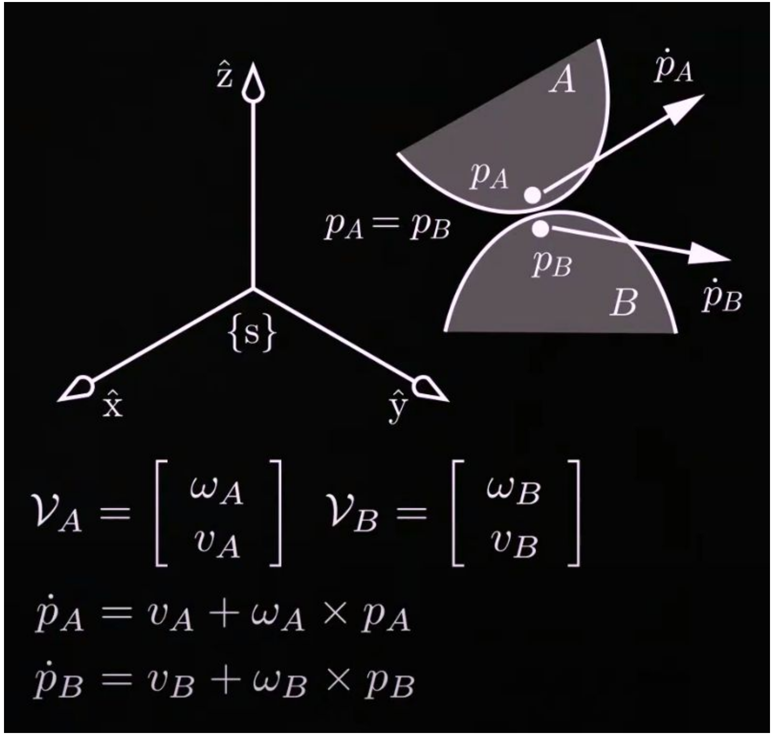


Figure 12.2: (Left) The bodies A and B in single-point contact define a contact tangent plane and a contact normal vector $\hat{n}$ perpendicular to the tangent plane. By default, the positive direction of the normal is chosen into body A. Since contact curvature is not addressed in this chapter, the contact places the same restrictions on the motions of the rigid bodies in the middle and right panels.



12.1.1 in Lynch, K. M., & Park, F. C. (2017). Modern robotics. Cambridge University Press.
https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-1-1-first-order-analysis-of-a-single-contact/

Twists of contact points



12.1.2 in Lynch, K. M., & Park, F. C. (2017). Modern robotics. Cambridge University Press.
https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-1-2-contact-types-rolling-sliding-and-breaking

Contact types

first-order rolling (~ sticking) contact

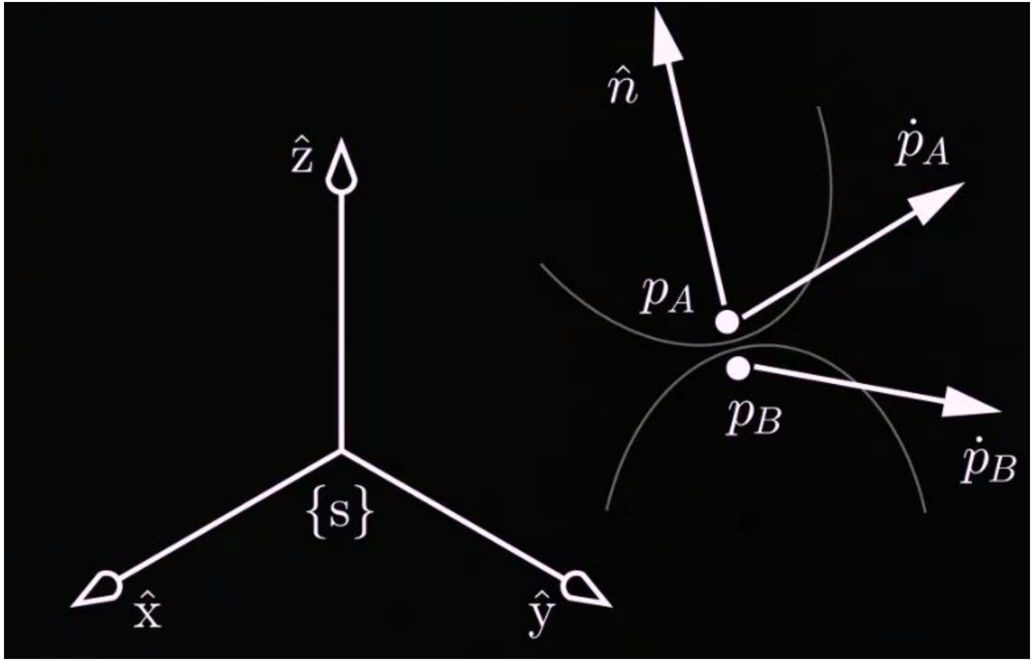
$$\dot{p}_A - \dot{p}_B = 0$$

impenetrability constraint

$$\hat{n}^T(\dot{p}_A - \dot{p}_B) \geq 0$$

first-order roll-slide

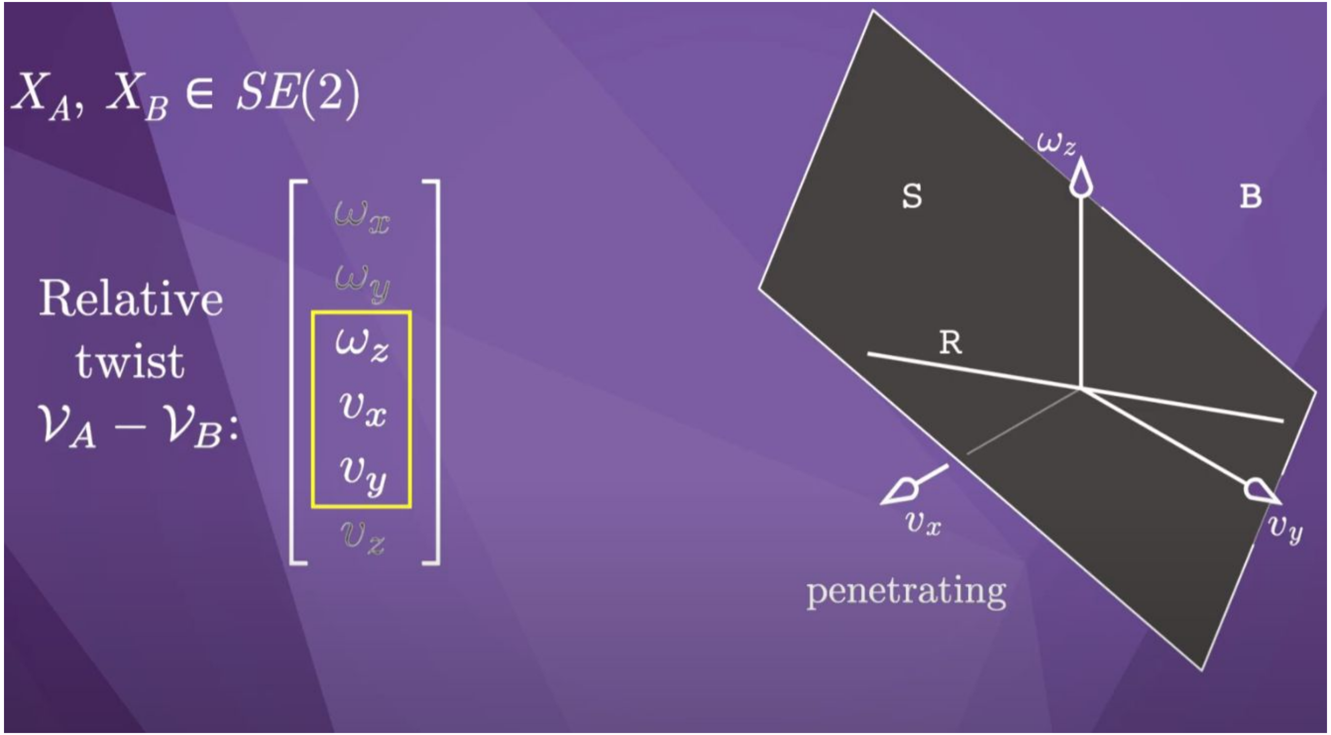
$$\hat{n}^T(\dot{p}_A - \dot{p}_B) = 0$$



$$\mathbf{a} \cdot \mathbf{b} = \|\mathbf{a}\| \, \|\mathbf{b}\| \cos\theta,$$

12.1.2 in Lynch, K. M., & Park, F. C. (2017). Modern robotics. Cambridge University Press.
https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-1-2-contact-types-rolling-sliding-and-breaking

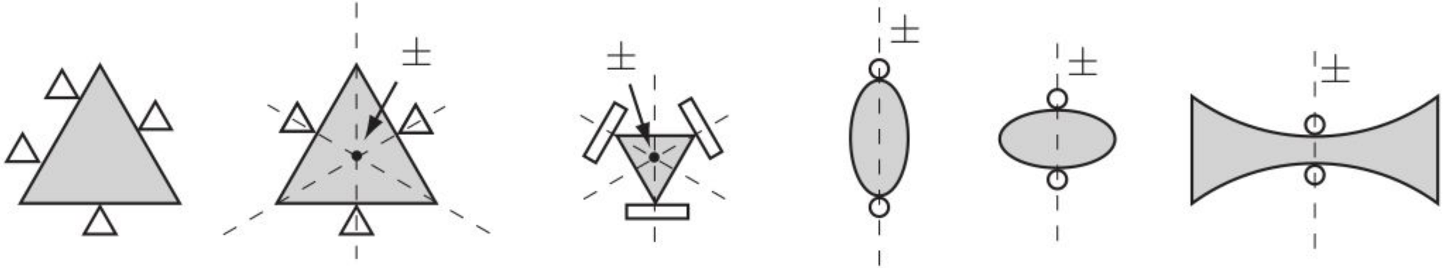
Contact types



https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-1-2-contact-types-rolling-sliding-and-breaking

Form closure

- if a set of stationary constraints prevents all motion of the body.
- i.e. the only twist is the zero twist.

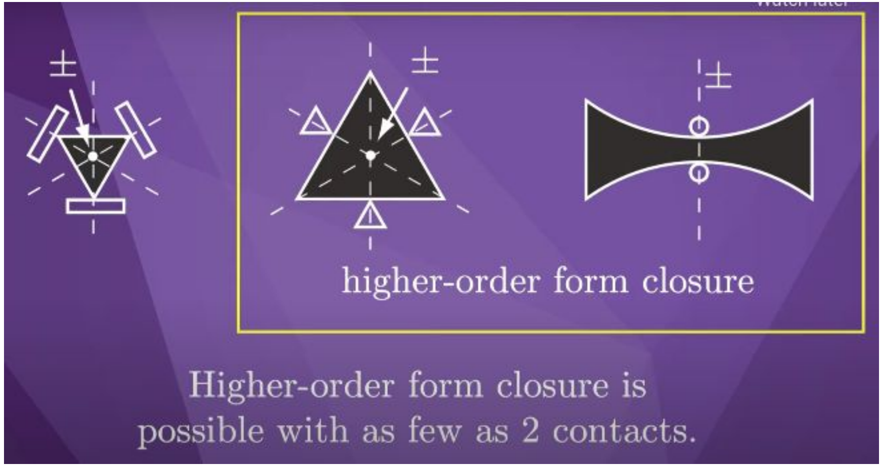


(a) (b) (c) (d) (e) (f)

		(a)	(b)	(c)	(d)	(e)	(f)
Form closure	1st order analysis	✅	×	×	×	×	×
	2nd order analysis		✅	×	×	×	✅

12.1.7 in Lynch, K. M., & Park, F. C. (2017). Modern robotics. Cambridge University Press.
https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-1-7-form-closure

- If an object is in form closure by first-order analysis, then it is also in form closure by a higher-order analysis.
- If a first-order analysis concludes only sliding and rolling contacts (no breaking), a higher-order analysis may conclude form closure.



12.1.7 in Lynch, K. M., & Park, F. C. (2017). Modern robotics. Cambridge University Press.
https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-1-7-form-closure

Form closure

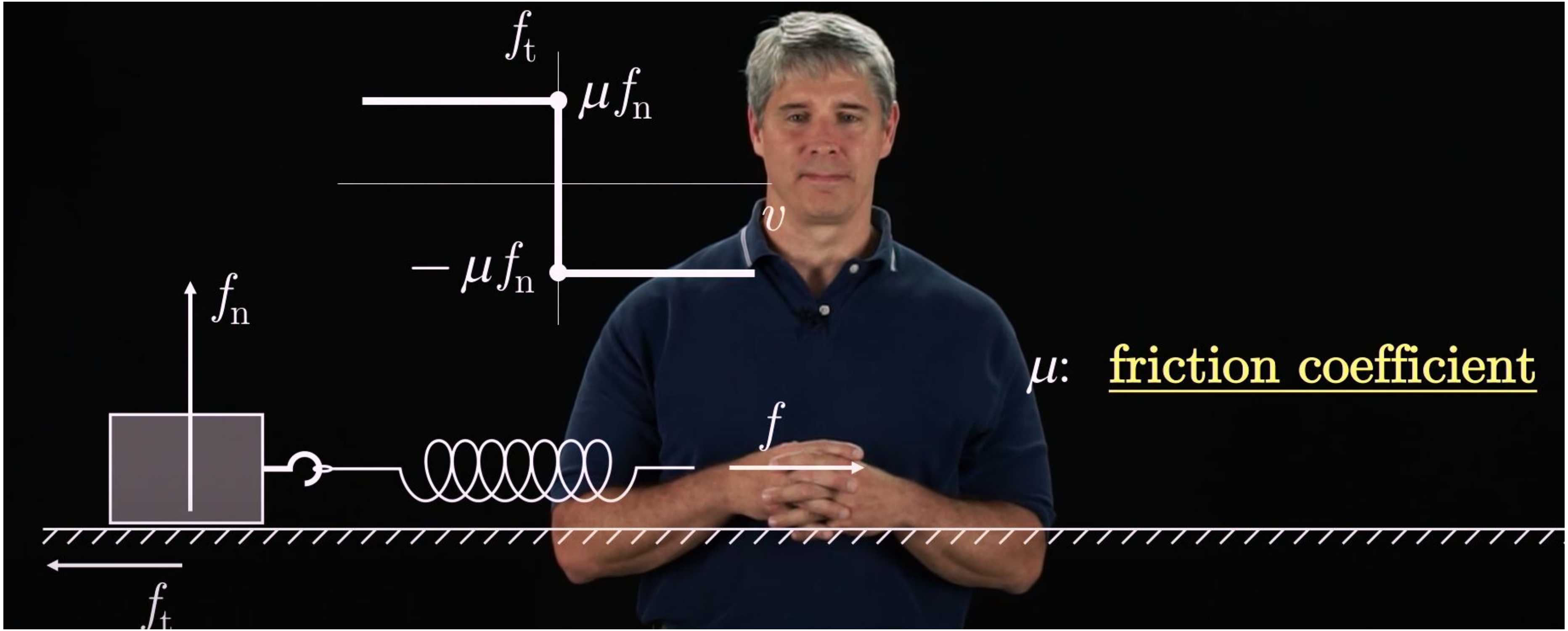
- Form-closure requires:
  - At least 4 point contacts for a planar body.
  - At least 7 point contacts for a spatial body.

Question: are we grasping like that?

Grasping vs. design of fixtures.

12.1.7 in Lynch, K. M., & Park, F. C. (2017). Modern robotics. Cambridge University Press.
https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-1-7-form-closure

Contacts with friction – Coulomb model



This model is reasonable for hard, dry, materials.

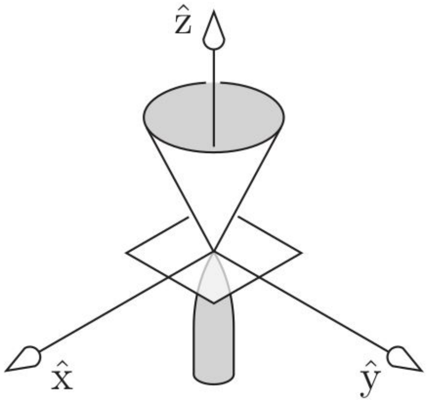
https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-2-1-friction/

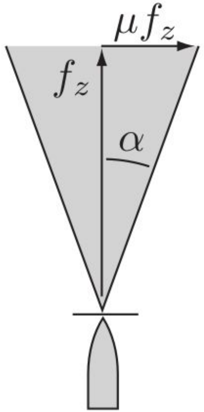
Friction cone

For a contact normal in the $+\hat{z}$-direction, the set of forces that can be transmitted through the contact satisfies

$$\sqrt{f_x^2 + f_y^2} \leq \mu f_z, \qquad f_z \geq 0. \tag{12.16}$$

- What happens to the friction cone if
  - I press harder?
  - The friction coefficient changes?

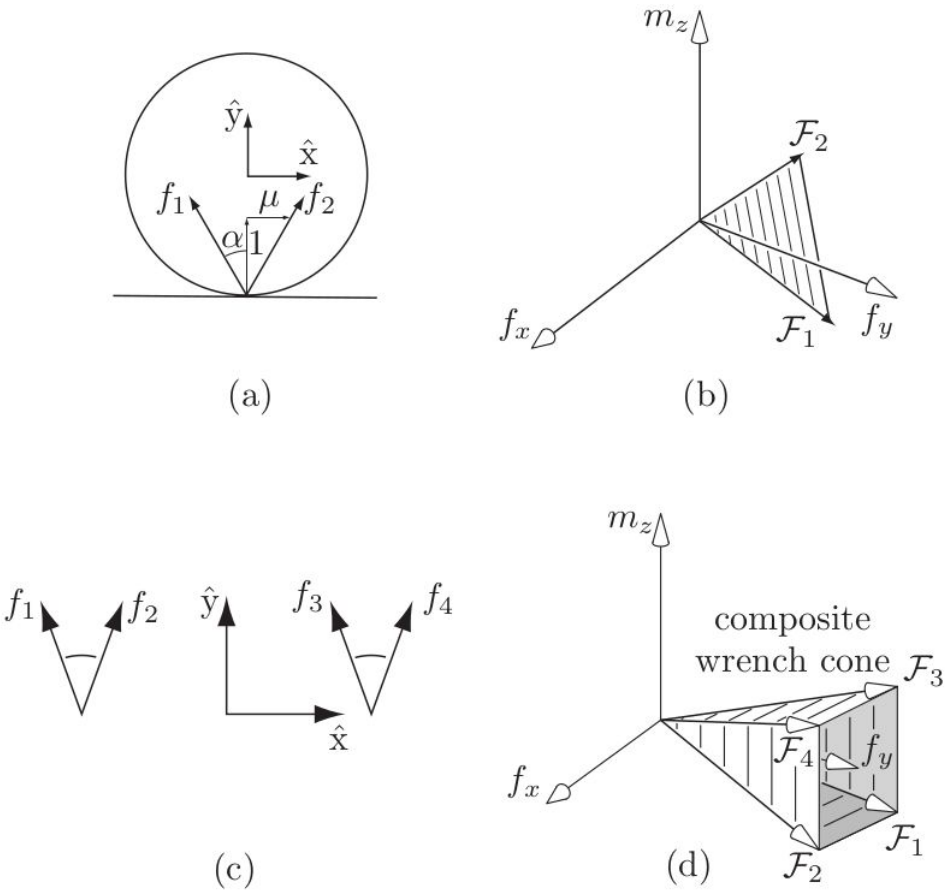




12.2.1 in Lynch, K. M., & Park, F. C. (2017). Modern robotics. Cambridge University Press.
https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-2-1-friction/

Wrench cone

- Not only forces but also moments/torques can be transmitted through contacts with friction.
- Note that every contact provides more than 1 force "basis" vector.



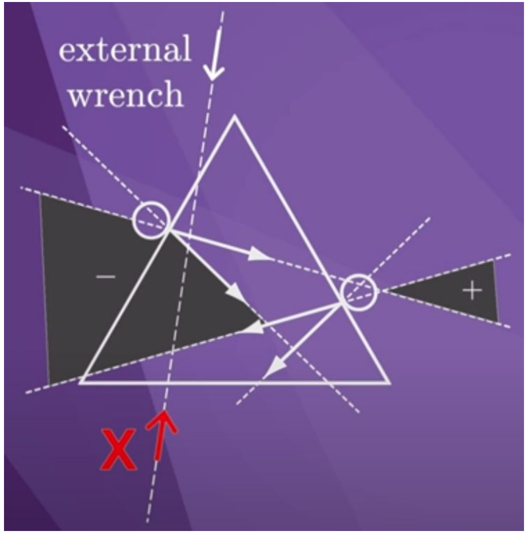
12.2.1 in Lynch, K. M., & Park, F. C. (2017). Modern robotics. Cambridge University Press.
https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-2-1-friction/

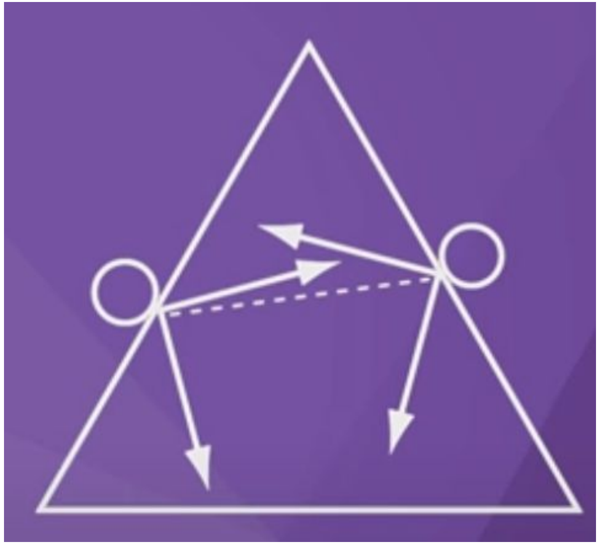
Force closure

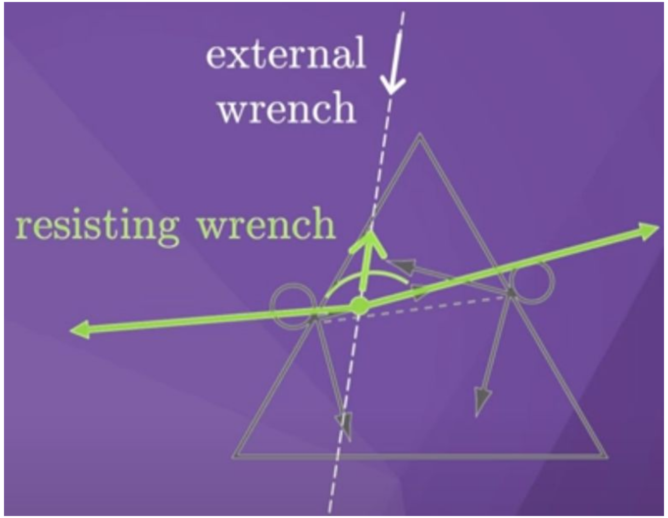
- A grasp is *force closure*
  - If for any external wrench there exist contact wrenches that cancel it.
  - The composite wrench cone contains the entire wrench space, so that any external wrench on the body can be balanced by contact forces.
- Intuition
  - Form closure - object completely immobilized statically/geometrically (no forces applied).
  - Force closure - someone is trying to take the object out of my hand but I can resist any such force or rotation by pushing firmly through my fingers at the appropriate contact locations.

12.2.3 in Lynch, K. M., & Park, F. C. (2017). Modern robotics. Cambridge University Press.
https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-2-3-force-closure/

Force closure







- What has changed?
  - (new contact points)
  - **friction coefficient increased!**
- Now: any wrench can be generated -> force closure.

https://modernrobotics.northwestern.edu/nu-gm-book-resource/12-2-3-force-closure

Intuitions - summary - form vs. force closure

With form closure, the contacts were acting (preventing object's motion) only along the normal. With friction, we get leverage in the orthogonal direction!

Friction always requires contact forces (pushing)!

Friction forces only counteract/resist other forces. That is actually very handy here - resist wrenches that want to take the object away from the grasp...

Each contact is not a single basis like in form closure but through the friction/wrench cone actually a set...

Form and force closure summary

Friction-less force closure ~ first-order form closure.

Form closure requires:

- At least 4 point contacts for a planar body.
- At least 7 point contacts for a spatial body.

Force closure with friction possible with as few as:

- 2 contacts for a planar body.
- 3 contacts for a spatial body.
    - 2 soft fingers - yes!

Now, how do we choose a grasp?

Prerequisite: evaluate alternative grasps (grasp proposals).

Grasp quality measure.

Grasp wrench space - “minimum ball”.

(employed in GraspIt! simulator)

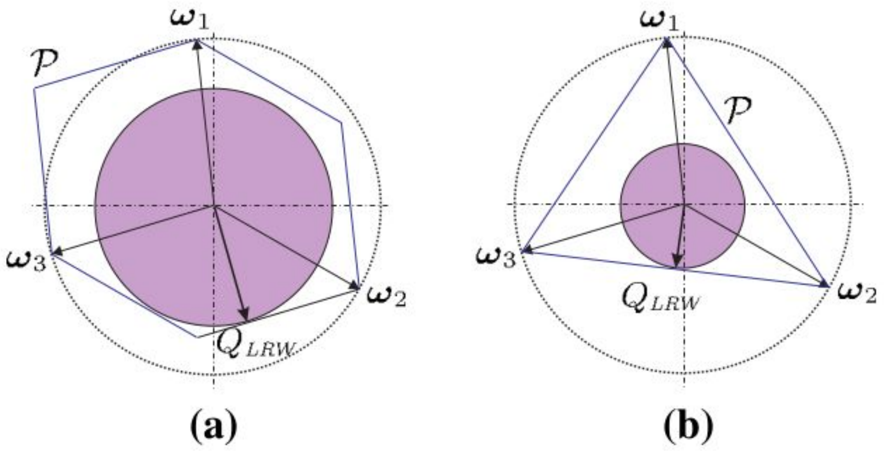


Fig. 5 Qualitative 2-dimensional example of the grasp quality using 3 fingers and **a** a limit in the module of each force; **b** a limit in the sum of the modules of the applied forces

Prattichizzo, D., & Trinkle, J. C. (2016). Grasping. In Springer handbook of robotics (pp. 955-988). Springer, Cham.

Grasp quality measures

Table 2 Grasp quality measures

Group	Subgroup	Quality index	Criterion
Measures related to the position of the contact points on the object	Based on algebraic properties of G	Minimum singular value of G	Maximize
		Volume of the ellipsoid in the wrench space	Maximize
		Grasp isotropy index	Maximize
	Based on geometric relations	Shape of the grasp polygon[a]	Minimize
		Area of the grasp polygon	Maximize
		Distance between the centroid C and the center of mass CM	Minimize
		Orthogonality	Minimize
		Margin of uncertainty in finger positions[b]	Maximize
		Based on independent contact regions	Maximize
	Considering limitations on the finger forces	Largest-minimum resisted wrench	Maximize
		Volume of the Grasp Wrench Space	Maximize
		Decoupled forces and torques	Maximize
		Normal components of the contact forces	Minimize
		Coplanarity of the normals[a]	Minimize
		Task oriented measures	Maximize
Measures related to hand configuration		Distance to singular configurations	Maximize
		Volume of the manipulability ellipsoid	Maximize
		Uniformity of transformation	Minimize
		Finger joint positions	Minimize
		Similar flexion values	Minimize
		Task compatibility index	Maximize
		Safety margin	Maximize
Other measures		Biomechanical fatigue	Minimize
		Deviation in object pose	Minimize

[a] Applicable only to 2D and 3D planar grasps
[b] Applicable only to 2D grasps

Roa, M. A., & Suárez, R. (2015). Grasp quality measures: review and performance. *Autonomous robots*, *38*(1), 65-88.

Sampling based grasp planning revisited

- Sampling approach
  - Choose candidate contacts.
  - Evaluate resulting grasp.

- Instead of choosing contact locations, sample location to place *preshaped* hand, and simulate where contacts happen after closing fingers.
  - Preshapes for prototypical grasps, e.g. pinch grasp, power grasp, cylindrical grasp.
  - Miller et al. 2003.

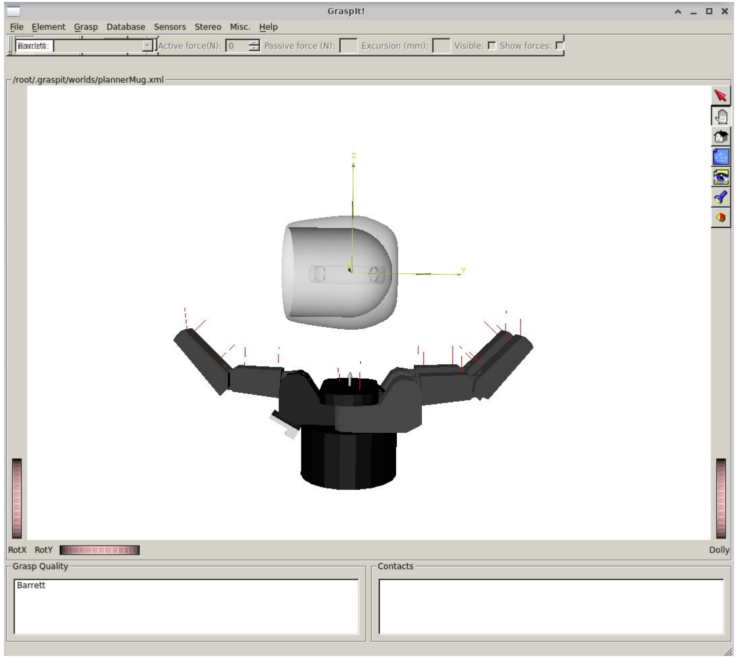
Aalto University
School of Electrical Engineering

Ville Kyrki

Slide from Ville Kyrki, Aalto University. Course: Robotic manipulation. Lectures 8: Friction and grasping.
https://mycourses.aalto.fi/course/view.php?id=32938&section=1

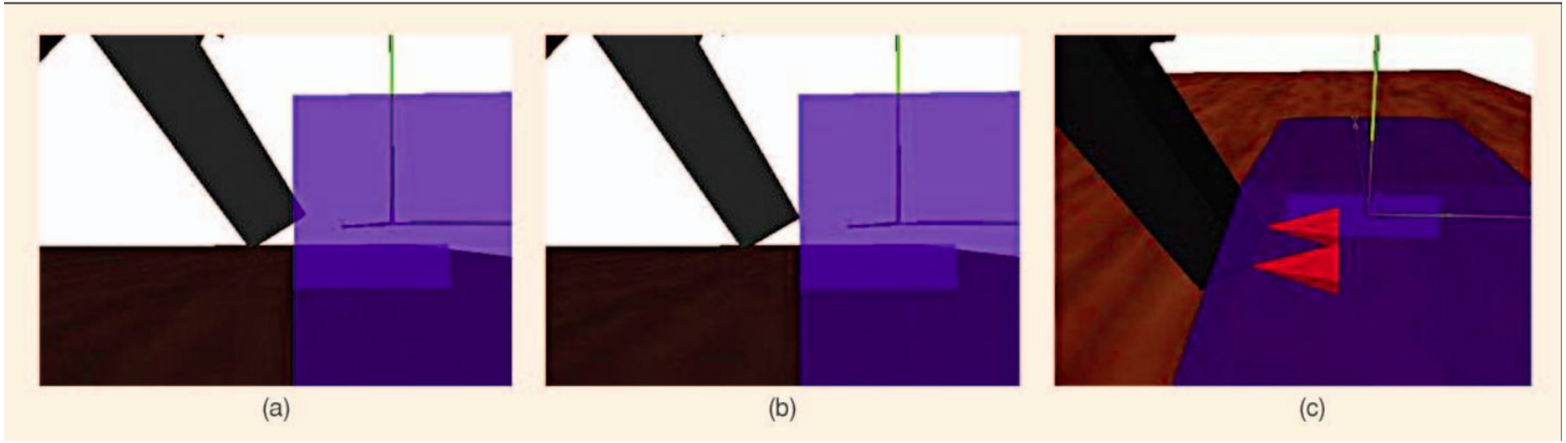
GraspIt! - Overview

- http://graspit-simulator.github.io
    - Miller, A. T., & Allen, P. K. (2004). Graspit: A versatile simulator for robotic grasping. IEEE Robotics and Automation Magazine.
- Used for long time
    - For example as generator of labeled grasps
- Supports different hands or robots
    - Users can define their own
- Supports obstacles
    - Importable as meshes
- Supports materials
    - Different coefficients of friction
- Dynamic simulation can be enabled
    - Bullet



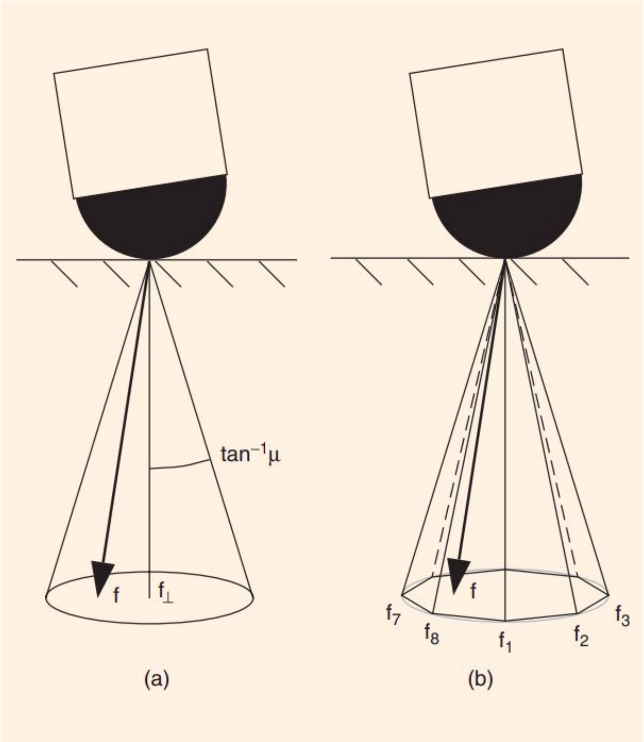
GraspIt! - How it works

- Contact between object and gripper is detected (a)
    - Using collision detection based on trees of bounding boxes
- Joint angle which caused the collision is found and the movement is reverted before collision (b)
- Geometry of the contact is found and friction cones are created (c)



GraspIt! - Friction cones

- Coulomb friction model
  - Force applicable at the contact is in the friction cone
- Friction cone (a)
  - Apex in the contact point
  - Axis along the normal force $\boldsymbol{f}_{\perp}$
  - Half angle $tan^{-1}\mu$
    - μ is the friction coefficient
- During grasp analysis, the cone is approximated with an *m* side pyramid (b)
  - **f** is convex combination of *m* vectors



GraspIt! - Grasp Wrech Space

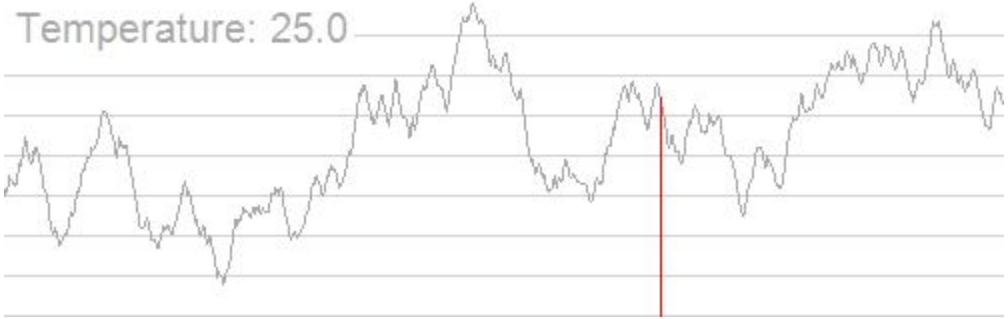
- Wrenches $\boldsymbol{w}_{i,j} = \begin{bmatrix} \boldsymbol{f}_{i,j} \\ \lambda(\boldsymbol{d}_i \times \boldsymbol{f}_{i,j}) \end{bmatrix}$
  - $\boldsymbol{f}_{i,j}$ one of *m* forces from the cone at contact point *i*
  - $\boldsymbol{d}_i$ vector from the torque origin
  - λ force to torque multiplicator
- GWS - space of wrenches applicable to the object given limit on normal force
  - Computed as convex hull of wrenches
- $\boldsymbol{W}_{L1} = ConvexHull\left(\bigcup_{i=1}^{n}(\boldsymbol{w}_{i,j}, \ldots, \boldsymbol{w}_{i,m})\right)$
  - Used in GraspIt!
- $\boldsymbol{W}_{L\infty} = ConvexHull\left(\bigoplus_{i=1}^{n}(\boldsymbol{w}_{i,j}, \ldots, \boldsymbol{w}_{i,m})\right)$
  - Minkowski sum
- For 3D object the GWS is 6D -> three coordinates need to be fixed for visualization

GraspIt! - Metrics

- Task wrench space
  - Space of wrenches which needs to be applied to carry out the given task
    - 6D ball when we assume that disturbances can come from any direction
- 1) Epsilon-quality
  - Radius of the biggest 6D ball in the torque origin which can fit into unit GWS
  - The closer to 1, the better quality
- 2) Volume of $\boldsymbol{W}_{L1}$
  - The bigger, the better

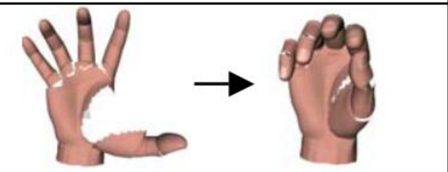
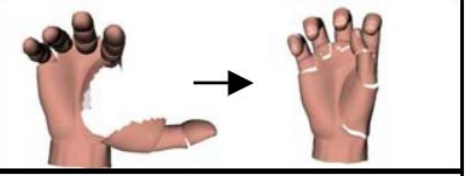
GraspIt! - Simulated Annealing

- Used to find global extrema
- Randomly computes a neighbor of current states and probabilistically decides if to change state or not
- Use parameter "Temperature T"
  - Decreases in time
  - If T = 0, it is basic hill climbing algorithm
- Used in GraspIt! to sample possible grasps



GraspIt! - Eigengrasps

- [Ciocarlie *et al.* ,2007. Dimensionality reduction for hand-independent dexterous robotic grasping. IEEE International Conference on Intelligent Robots and Systems.]()
- Reduction of DOF of hands
  - Based on results from robotics and neuroscience
    - Majority of grasps lacks individual finger movements
- For example, human hand needs only 2 eigengrasps

Human	20	Thumb rotation Thumb flexion MCP flexion Index abduction		Thumb flexion MCP extension PIP flexion	
Barrett	4	Spread angle opening		Finger flexion	

GraspIt! - Interface

- ROS interface https://github.com/graspit-simulator/graspit_interface
  - Publishes topics and services based on GraspIt! API
- Python client https://github.com/graspit-simulator/graspit_commander
  - Access the services with Python
  - Minimal knowledge of ROS needed
    - Only datatypes - Point, Quaternion, etc.

```
In [ ]: from graspit_commander import GraspitCommander
```

```
In [ ]: GraspitCommander.clearWorld()
        GraspitCommander.importRobot("BarrettBH8_280")
        GraspitCommander.importGraspableBody("my_object.ply")
        plan = GraspitCommander.planGrasps(max_steps=70000)
```

Problems in practice?

On the side of object:

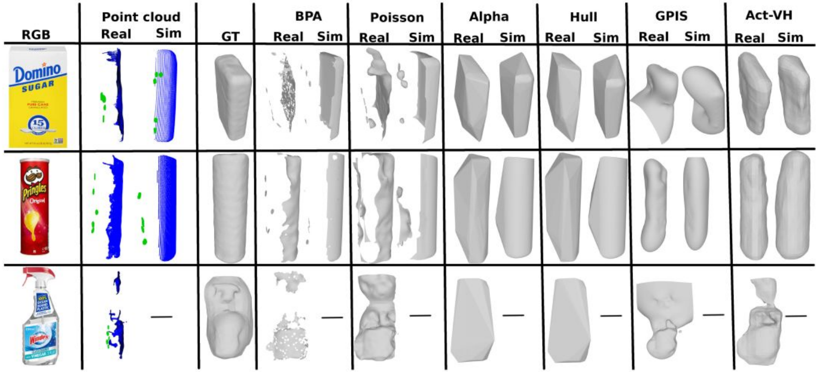
- shape estimation uncertainty
- pose estimation uncertainty
- friction estimation uncertainty
- rigidity assumption
- highly simplified contact model vs. reality

On gripper side:

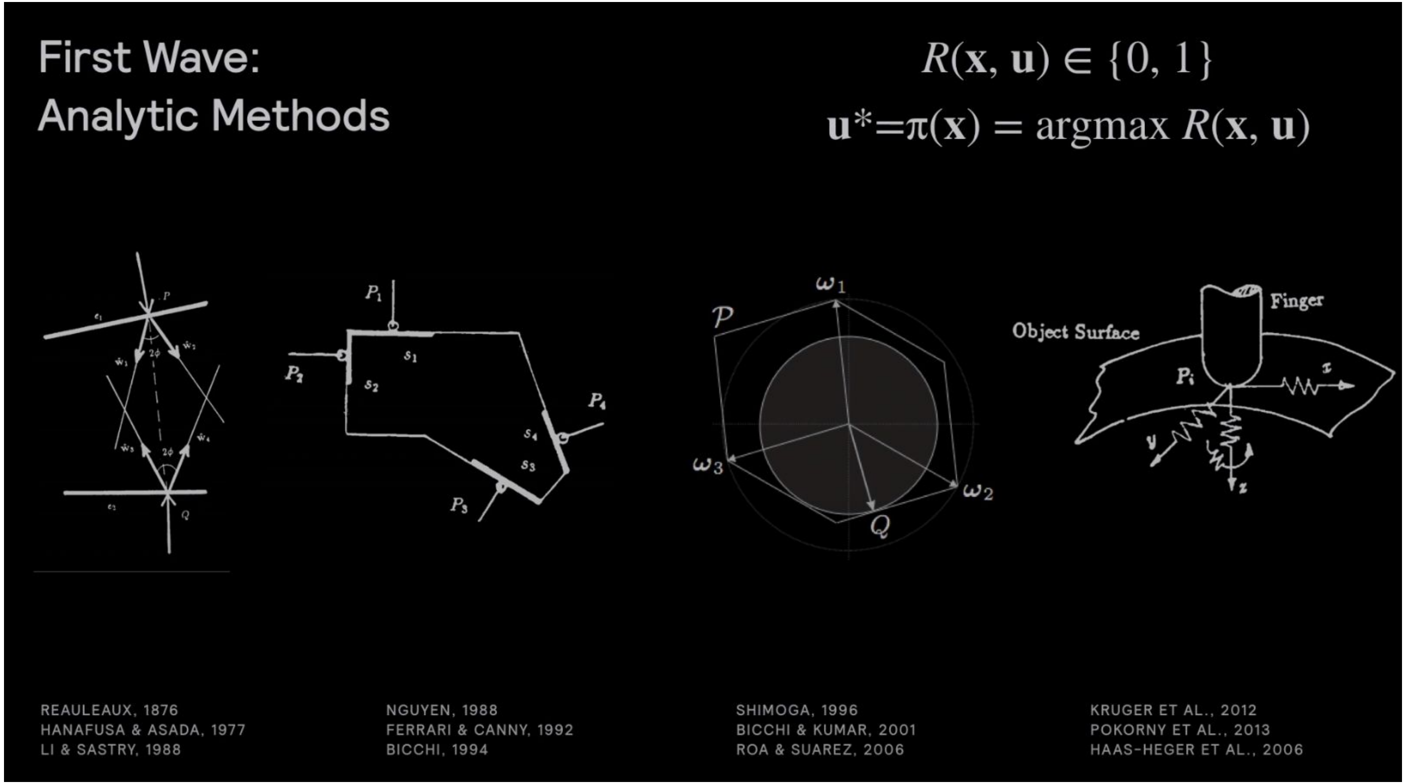
- kinematic constraints

Plus:

- planned vs. actual placement of gripper jaws / fingers
- task compatibility



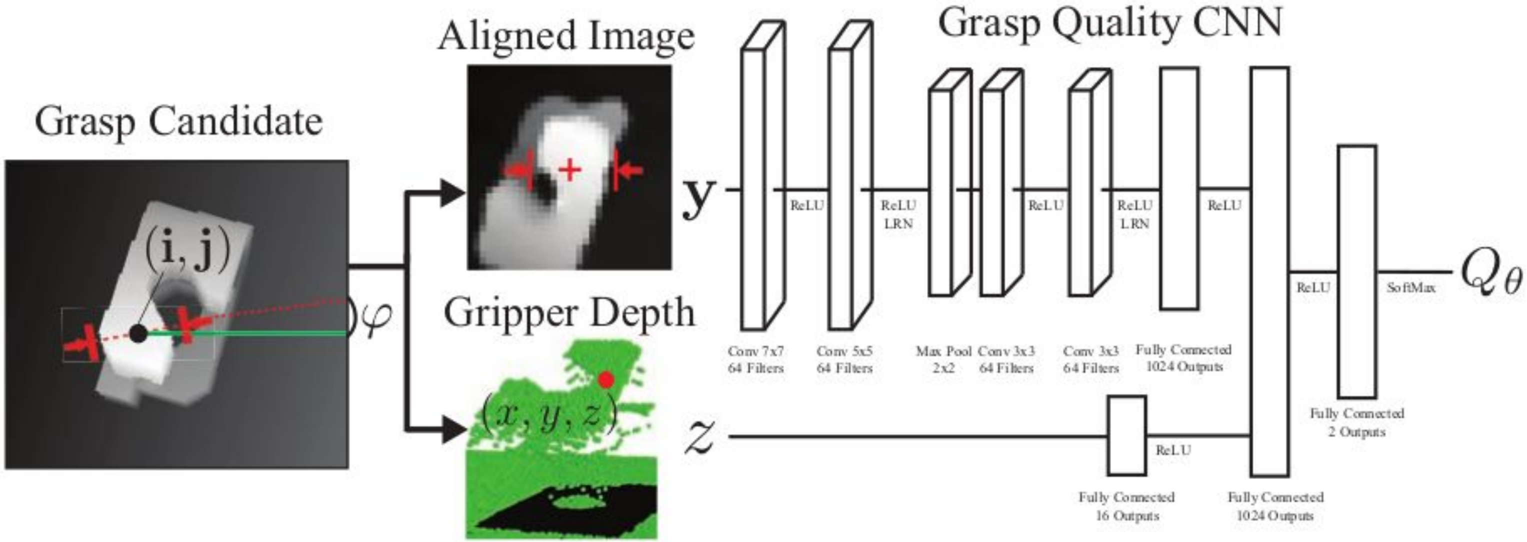
“First wave” - great theory but there is uncertainty everywhere



Ken Goldberg - The New Wave in Robot Grasping: https://youtu.be/ATDrSWZXuwk

Grasping as a learning problem

- ~ Data-driven grasping.
- Train a neural network to do the grasp evaluation.



Mahler, J., Liang, J., Niyaz, S., Aubry, M., Laskey, M., Doan, R., ... & Goldberg, K. (2018). Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics.

Dex-Net

https://youtu.be/r-0PKne9e_w

- Overview in a talk: Ken Goldberg - The New Wave in Robot Grasping: https://youtu.be/ATDrSWZXuwk
- Mahler, J., Liang, J., Niyaz, S., Aubry, M., Laskey, M., Doan, R., ... & Goldberg, K. (2018). Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics.
- Mahler, J., Matl, M., Satish, V., Danielczuk, M., DeRose, B., McKinley, S., & Goldberg, K. (2019). Learning ambidextrous robot grasping policies. Science Robotics, 4(26), eaau4984.

Dex-Net 2.0

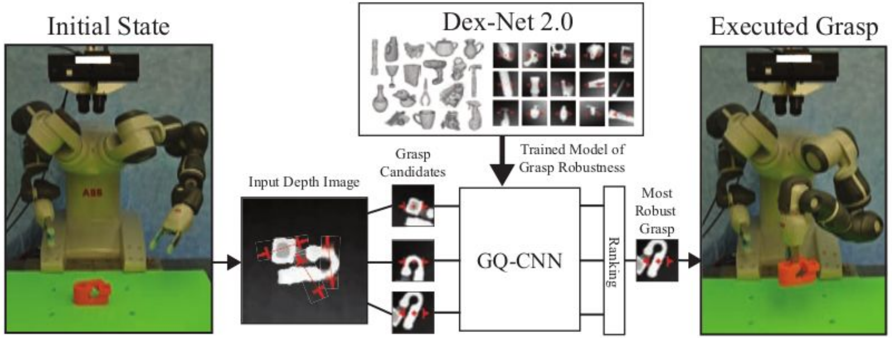


Fig. 1: Dex-Net 2.0 Architecture. (Center) The Grasp Quality Convolutional Neural Network (GQ-CNN) is trained offline to predict the robustness candidate grasps from depth images using a dataset of 6.7 million synthetic point clouds, grasps, and associated robust grasp metrics computed with Dex-Net 1.0. (Left) When an object is presented to the robot, a depth camera returns a 3D point cloud, where pairs of antipodal points identify a set of several hundred grasp candidates. (Right) The GQ-CNN rapidly determines the most robust grasp candidate, which is executed with the ABB YuMi robot.

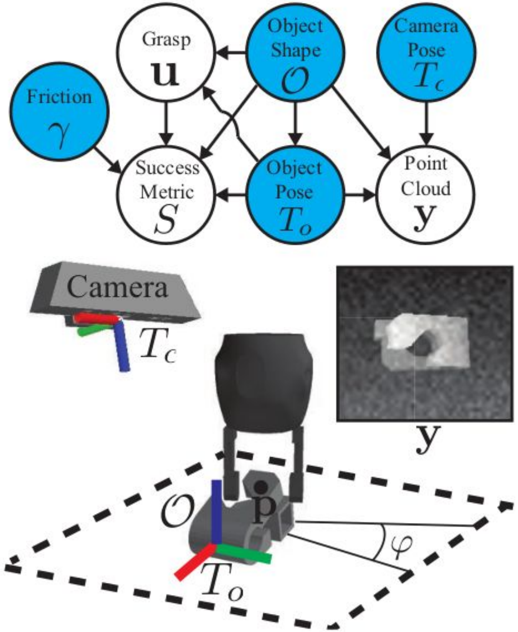


Fig. 2: Graphical model for robust parallel-jaw grasping of objects on a table surface based on point clouds. Blue nodes are variables included in the state representation. Object shapes $\mathcal{O}$ are uniformly distributed over a discrete set of object models and object poses T_o are distributed over the object's stable poses and a bounded region of a planar surface. Grasps $\mathbf{u} = (\mathbf{p}, \varphi)$ are sampled uniformly from the object surface using antipodality constraints. Given the coefficient of friction γ we evaluate an analytic success metric S for a grasp on an object. A synthetic 2.5D point cloud $\mathbf{y}$ is generated from 3D meshes based on the camera pose T_c, object shape, and pose and corrupted with multiplicative and Gaussian Process noise.

Mahler, J., Liang, J., Niyaz, S., Aubry, M., Laskey, M., Doan, R., ... & Goldberg, K. (2018). Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics.

Dex-Net 2.0

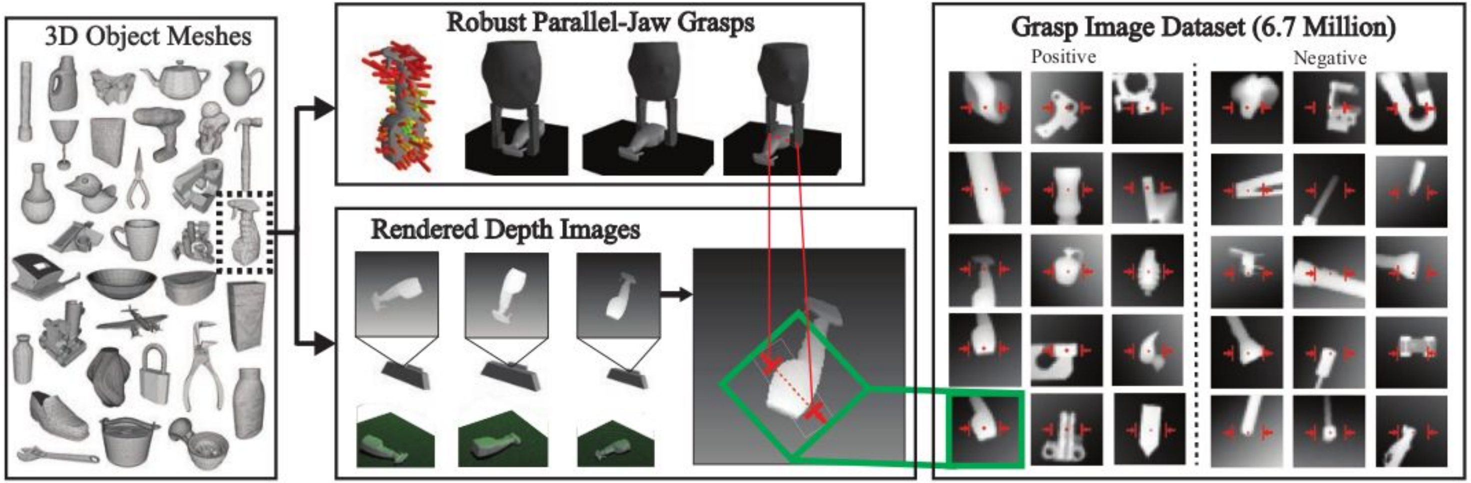


Fig. 3: Dex-Net 2.0 pipeline for training dataset generation. (Left) The database contains 1,500 3D object mesh models. (Top) For each object, we sample hundreds of parallel-jaw grasps to cover the surface and evaluate robust analytic grasp metrics using sampling. For each stable pose of the object we associate a set of grasps that are perpendicular to the table and collision-free for a given gripper model. (Bottom) We also render point clouds of each object in each stable pose, with the planar object pose and camera pose sampled uniformly at random. Every grasp for a given stable pose is associated with a pixel location and orientation in the rendered image. (Right) Each image is rotated, translated, cropped, and scaled to align the grasp pixel location with the image center and the grasp axis with the middle row of the image, creating a 32×32 grasp image. The full dataset contains over 6.7 million grasp images.

Mahler, J., Liang, J., Niyaz, S., Aubry, M., Laskey, M., Doan, R., ... & Goldberg, K. (2018). Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics.

Dex-Net 4.0

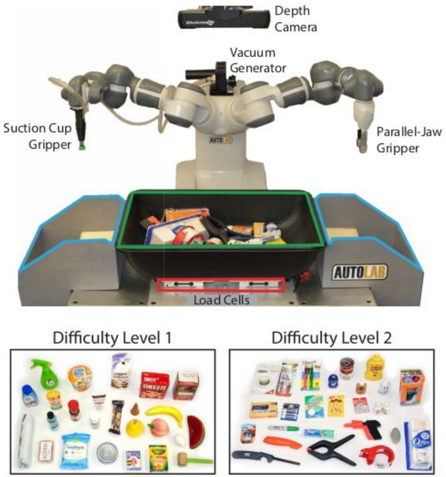


Fig. 2. Physical benchmark for evaluating UP policies. (**Top**) The robot plans a grasp to iteratively transport each object from the picking bin (green) to a receptacle (blue) using either a suction-cup or a parallel-jaw gripper. Grasp planning is based on 3D point clouds from an overhead Photoneo PhoXi S industrial depth camera. (**Bottom**) Performance is evaluated on two datasets of novel test objects not used in training. (**Left-Bottom**) Level 1 objects consist of prismatic and circular solids (e.g., boxes and cylinders) spanning groceries, toys, and medicine. (**Right-Bottom**) Level 2 objects are more challenging, including common objects with clear plastic and varied geometry, such as products with cardboard blisterpack packaging.

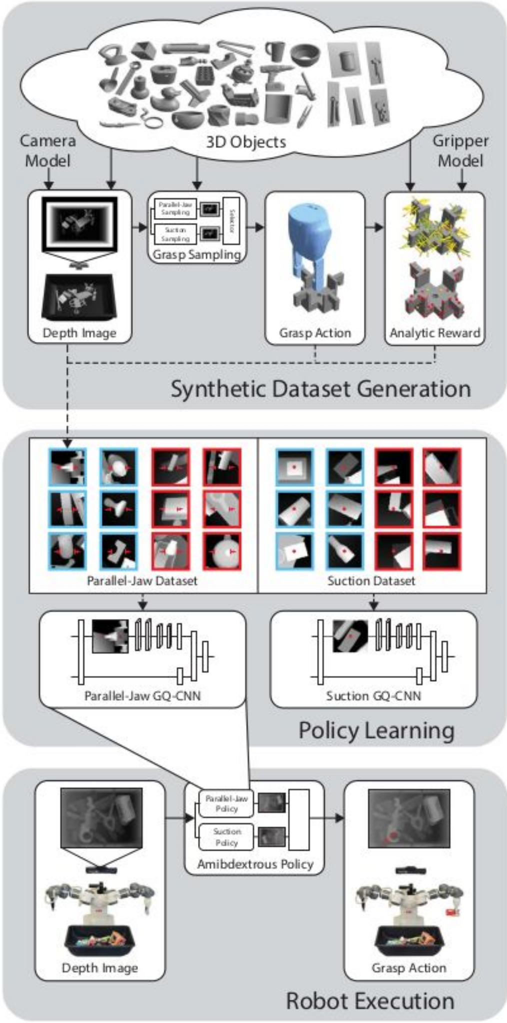


Fig. 1. Learning ambidextrous grasping policies for UP. (**Top**) Synthetic training datasets of depth images, grasps, and rewards are generated from a set of 3D computer-aided design (CAD) models using analytic models based on physics and domain randomization. Specifically, a data collection policy proposes actions given a simulated heap of objects, and the synthetic training environment evaluates rewards. Reward is computed consistently across grippers by considering the ability of a grasp to resist a given wrench (force and torque) based on the grasp wrench space, or the set of wrenches that the grasp can resist through contact. (**Middle**) For each gripper, a policy is trained by optimizing a deep GQ-CNN to predict the probability of grasp success given a point cloud over a large training dataset containing millions of synthetic examples from the training environment. Data points are labeled as successes (blue) or failures (red) according to the analytic reward metric. (**Bottom**) The ambidextrous policy is deployed on the real robot to select a gripper by maximizing grasp quality using a separate GQ-CNN for each gripper.

Mahler, J., Matl, M., Satish, V., Danielczuk, M., DeRose, B., McKinley, S., & Goldberg, K. (2019). Learning ambidextrous robot grasping policies. Science Robotics, 4(26), eaau4984.

GPD - Overview

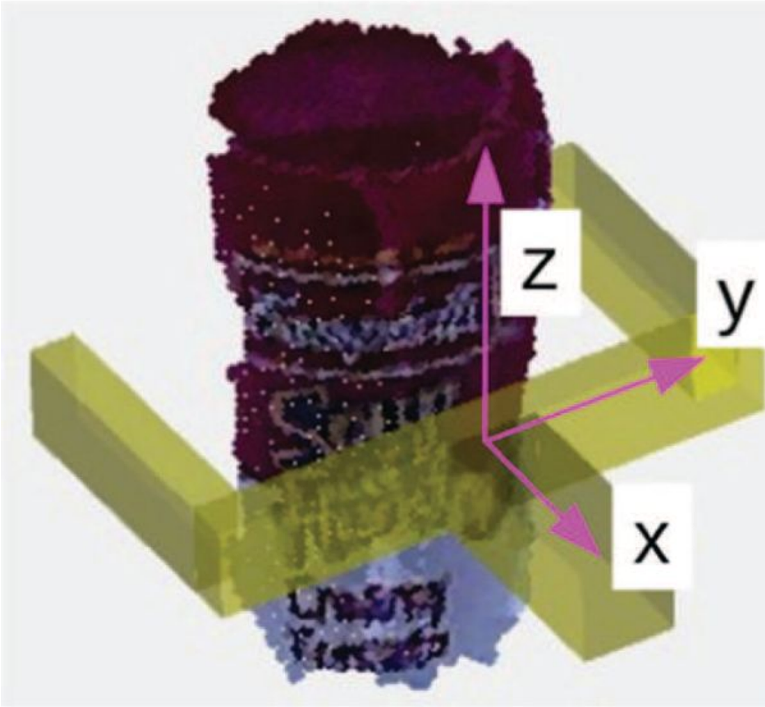
- https://github.com/atenpas/gpd
  - ten Pas *et al.*, 2017. Grasp Pose Detection in Point Clouds. International Journal of Robotics Research.
- based on point clouds
  - even a single view
- machine learning
- no physical properties needed
  - Materials, etc.
- faster than GraspIt!
- works in cluttered environments
- assumes only two-finger grippers

GPD - Point Clouds

- point clouds from RGB-D cameras
  - one view is sufficient
  - basic pre-processing is needed
    - denoising, downsampling, outliers removal
- only information in Region of Interest (ROI) is considered
  - segmented object
  - or only given region in point cloud, *e.g.*, workspace

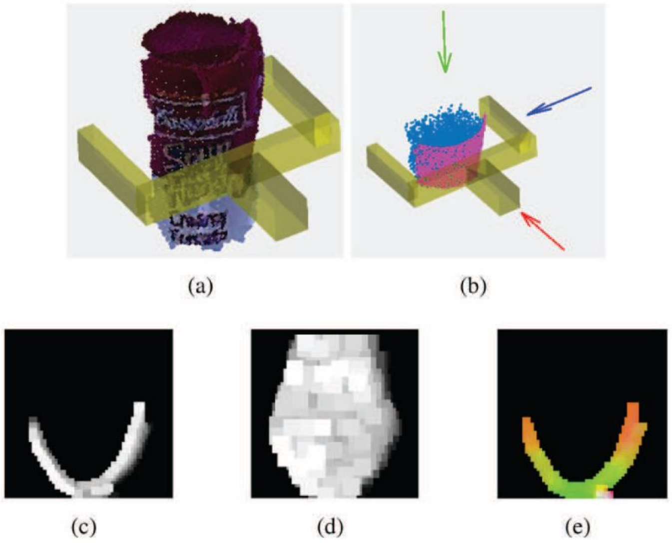
GPD - Grasps sampling

- candidates sampled uniformly randomly over the point cloud
- two conditions:
  - the body of the hand is not in collision with the point cloud
  - the closing region of the hand contains at least one point from the point cloud
- for each candidate, reference frame **F** of the hand is computed
- Grid search in grid $G = Y \times Z$ is performed. *Y* and *Z* contains values along *y* and *z* axis of **F.**
  - corresponding rotation and translation for each grid point applied to the hand
- rotated hand pushed along negative *x* axis until contact with point cloud occurs
  - last point before contact added to set of possible grasp if any point from the point cloud is in the closing region of the hand



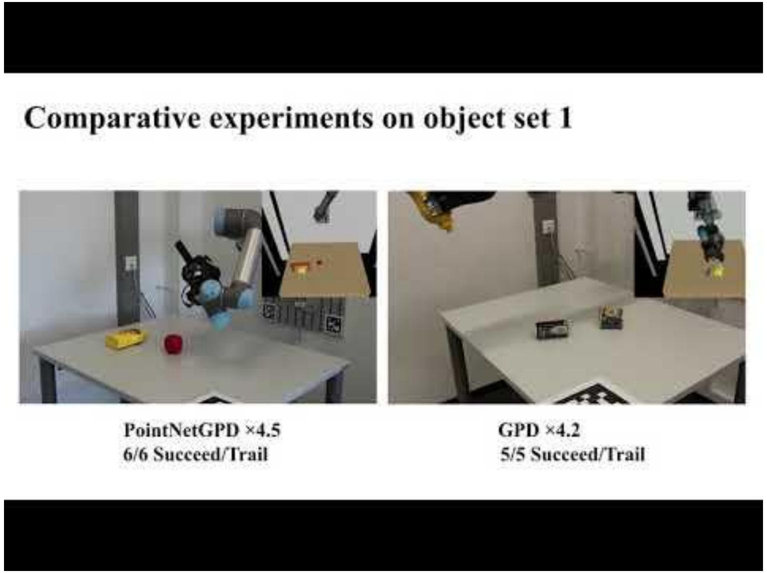
GPD - Grasp Classification

- four-layer CNN
  - Binary classification - grasp/no grasp
- trained from 300 thousand (sampled from 1.5 million) labeled grasps for 55 objects (~ labeled using ~ force closure)
- points in closing region (b) are voxelized (MxMxM voxels)
- input to CNN are heightmaps (c, d) of voxels projected to planes orthogonal to axes of the hand (b) and surface normals (e)



Others - PointNetGPD

- https://github.com/lianghongzhuo/PointNetGPD
    - Liang *et al.*, 2018. PointNetGPD: Detecting Grasp Configurations from Point Sets, IEEE International Conference on Robotics and Automation.
- same grasp sampling as GPD
- fewer parameters in CNN than GPD -> less prone to overfitting
- no hand-crafted features needed for training
- works with more sparse point clouds
- provides dataset with 350k real point clouds
- grasp with probability, not only binary



IPALM

Interactive Perception-Action-Learning for Modelling Objects

Chist-ERA (2019-2022)

Imperial College London

École des Ponts ParisTech

Institut de Robòtica i Informàtica Industrial

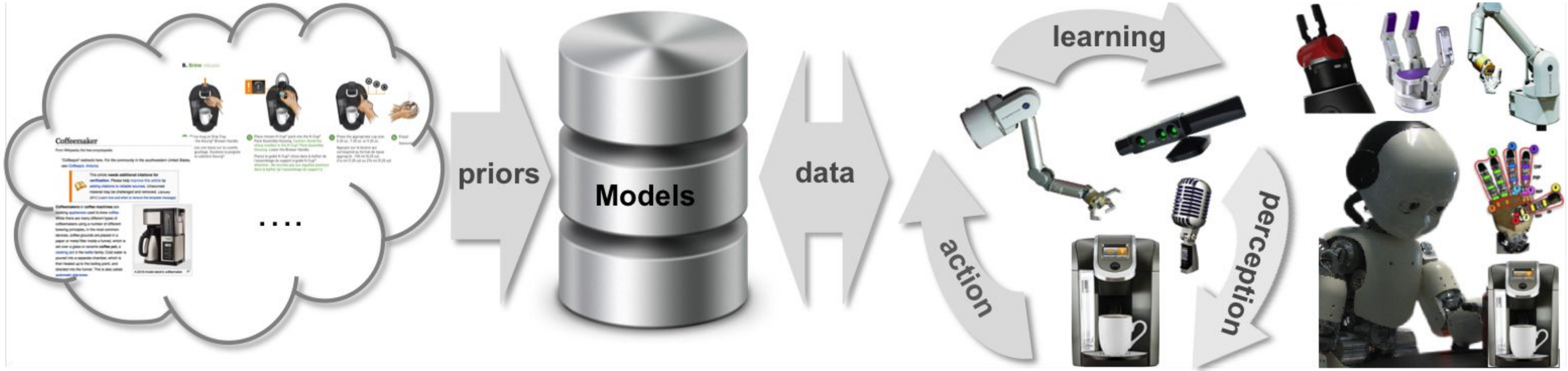
Aalto University

Czech Technical University in Prague

IPALM Consortium

- Imperial College London, UK, Krystian Mikolajczyk, Yiannis Demiris
  AI reasoning, vision, human-robot Interaction, developmental robotics
- ENPC ParisTech, France, Vincent Lepetit
  Object modeling from vision
- IRI, Spain, Francesc Moreno Noguer
  DL for non-rigid objects from appearance and depth
- Aalto University, Finland, Ville Kyrki
  Perception, learning and manipulation of objects
- Czech Technical University in Prague, Czech Republic, Matej Hoffmann
  Haptic object exploration, embodied perception

IPALM Interactive Perception-Action-Learning for Modelling Objects



What:

- Automatic digitization of objects and their physical properties by exploratory manipulations.
- Learning physical properties of objects from: vision, touch, audio and text.
- A benchmark and a database of objects models with a variety across properties

Imperial College London

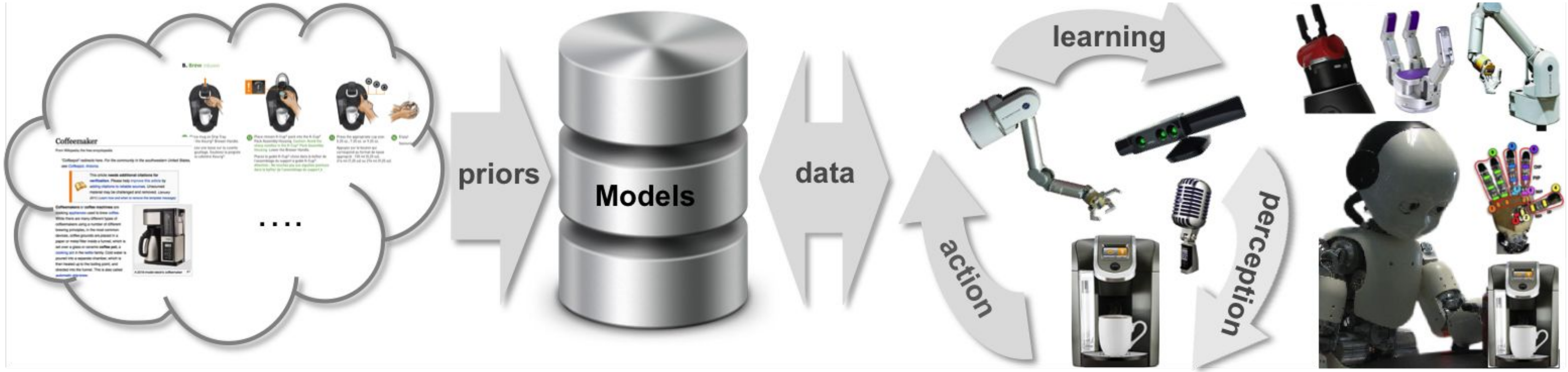
École des Ponts ParisTech

Institut de Robòtica i Informàtica Industrial

Aalto University

Czech Technical University in Prague

IPALM Interactive Perception-Action-Learning for Modelling Objects

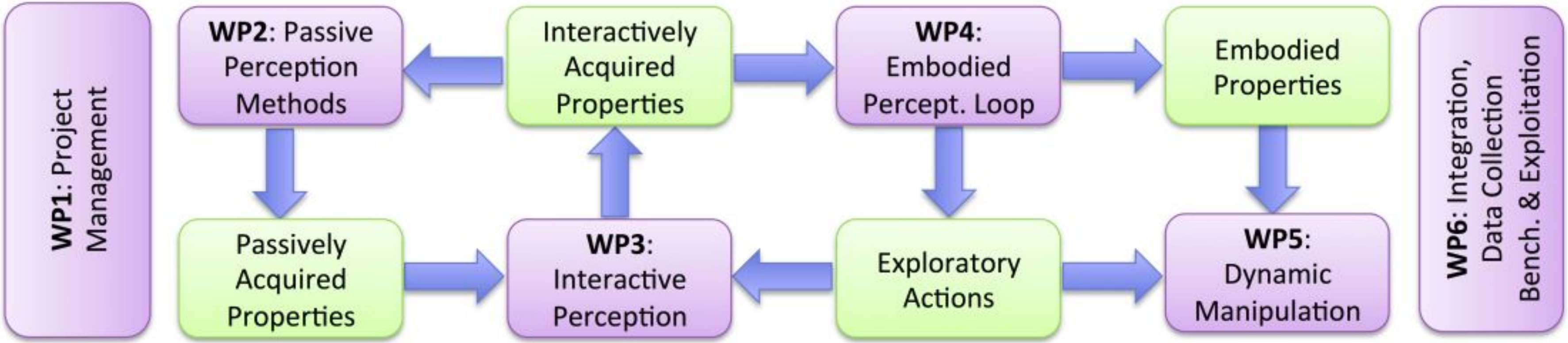


How:

- Vision and language resources provides priors and category level models for object recognition and manipulation
- Instance modelling based on a perception-action-learning loop
- New knowledge from instances is then used to refine category-level models

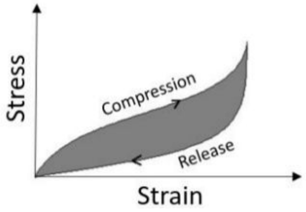
Imperial College London
École des Ponts ParisTech
Institut de Robòtica i Informàtica Industrial
Aalto University
Czech Technical University in Prague

How?


WP1: Project Management
WP2: Passive Perception Methods
Interactively Acquired Properties
WP4: Embodied Percept. Loop
Embodied Properties
WP6: Integration, Data Collection Bench. & Exploitation
Passively Acquired Properties
WP3: Interactive Perception
Exploratory Actions
WP5: Dynamic Manipulation

Measuring visco-elastic properties of soft objects

- empirical assessment of the feasibility of haptic online soft object discrimination
- elasticity and viscoelasticity estimation from compression and release cycles
- evaluation of 2-finger grippers with force feedback and F/T sensor
- analysis of effects of precycling, compression speed and gripper surface area
- dataset and code publicly available

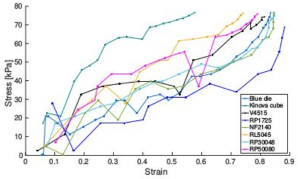


(a) Cubes and dice

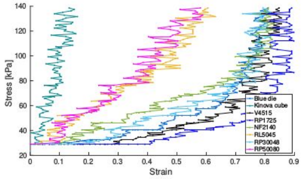
(b) Foams

(c) Mixed set

(a) Robotiq 2F-85



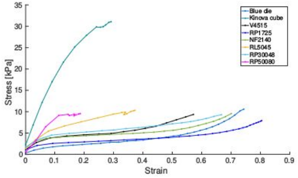
(a) Robotiq 2F-85



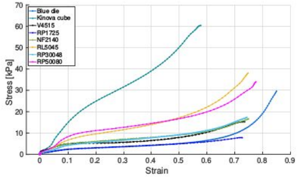
(b) OnRobot RG6

(b) OnRobot RG6

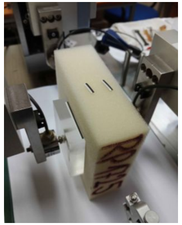
(c) Robotiq FT300



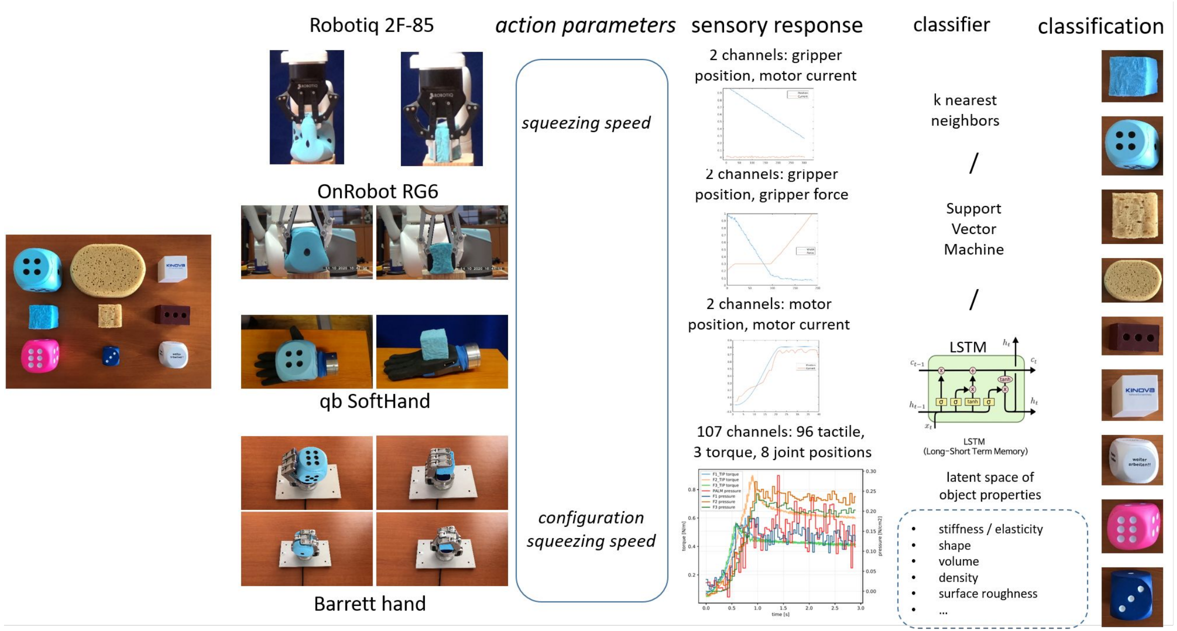
(c) Force/Torque sensor



(d) Professional setup



Single-grasp deformable object classification



Single-grasp deformable object classification: the effect of gripper morphology, sensing modalities, and action parameters

Michal Pliska[1], Michal Mareš[1], Pavel Stoudek[1], Zdeněk Straka[1], Karla Štěpánová[2], Matěj Hoffmann[1]

[1] Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague
[2] Czech Institute of Informatics, Robotics, and Cybernetics, Czech Technical University in Prague

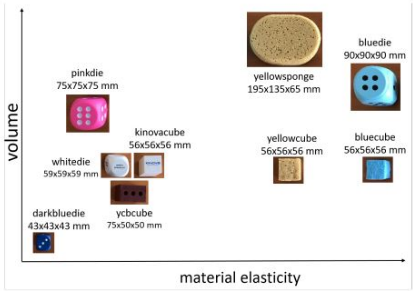
Pliska, M., Mares, M., Straka, Z., Stoudek, P., Stepanova, K. and Hoffmann, M. (2022), 'Single-grasp deformable object classification: the effect of gripper morphology, sensing modalities, and action parameters'. [under review]

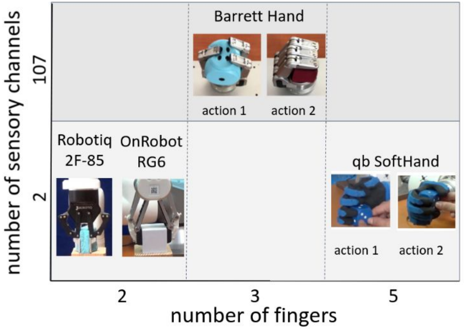
FACULTY OF ELECTRICAL ENGINEERING CTU IN PRAGUE

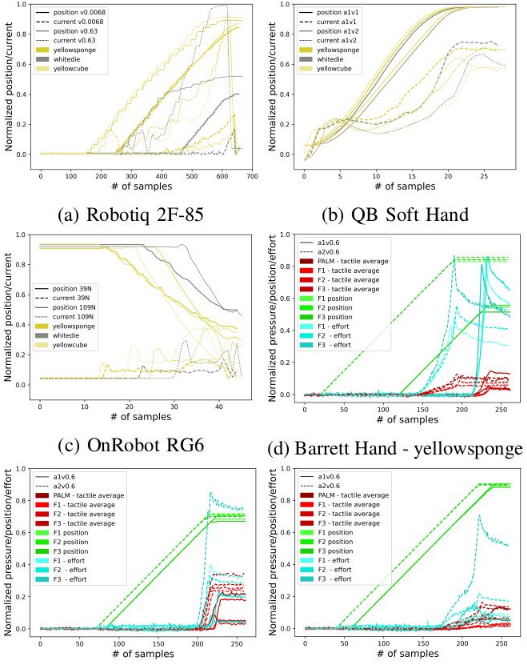
CZECH INSTITUTE OF INFORMATICS ROBOTICS AND CYBERNETICS CTU IN PRAGUE

Visual vs. haptic perception

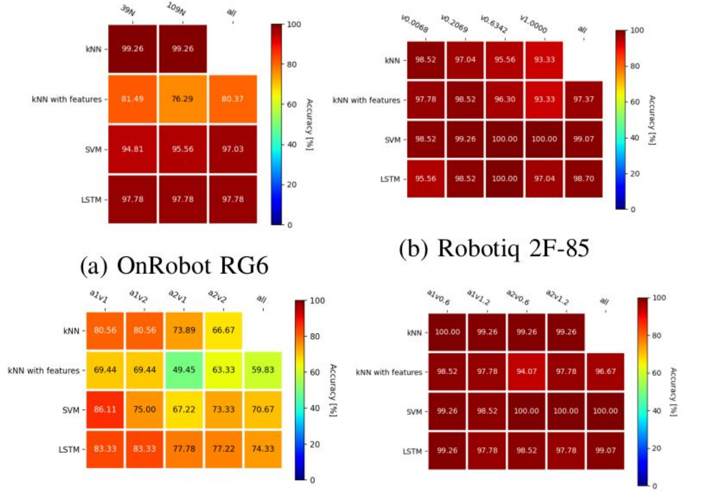
- Haptic - relating to the sense of touch ~ *tactile and proprioceptive information*
- (Compared to vision,) haptic perception is importantly **active** and **embodiment-dependent**





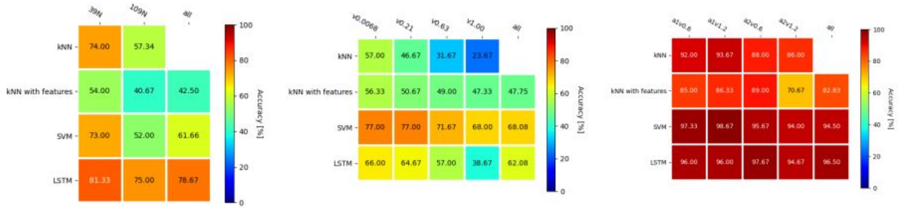


(a) Robotiq 2F-85 (b) QB Soft Hand
(c) OnRobot RG6 (d) Barrett Hand - yellowsponge
(e) Barrett Hand - whitedie (f) Barrett Hand - yellowcube



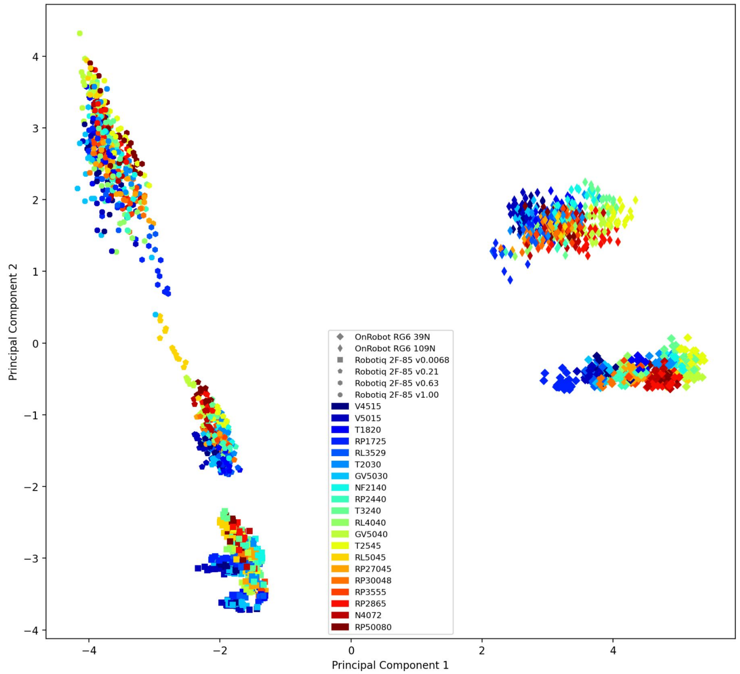
(a) OnRobot RG6 (b) Robotiq 2F-85 (c) qb Soft Hand (d) Barrett Hand

Fig. 4: Objects set – classification accuracy. (a) OnRobot RG6. (b) Robotiq 2F-85. (c) QB Soft Hand (d) Barrett Hand.



(a) OnRobot RG6 (b) Robotiq 2F-85 (c) Barrett Hand

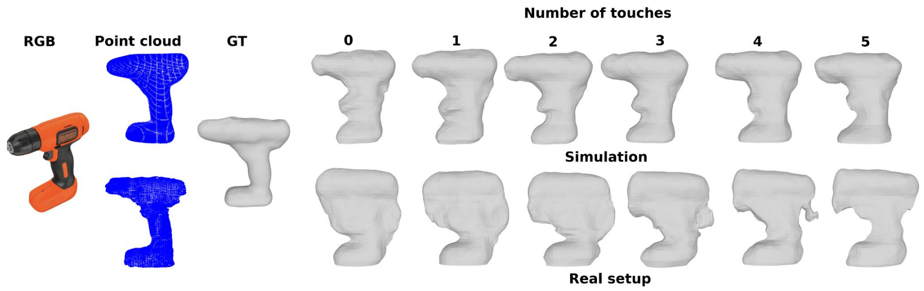
Fig. 5: Polyurethane foams set – classification accuracy. (a) OnRobot RG6 (b) Robotiq 2F-85 (c) Barrett Hand.

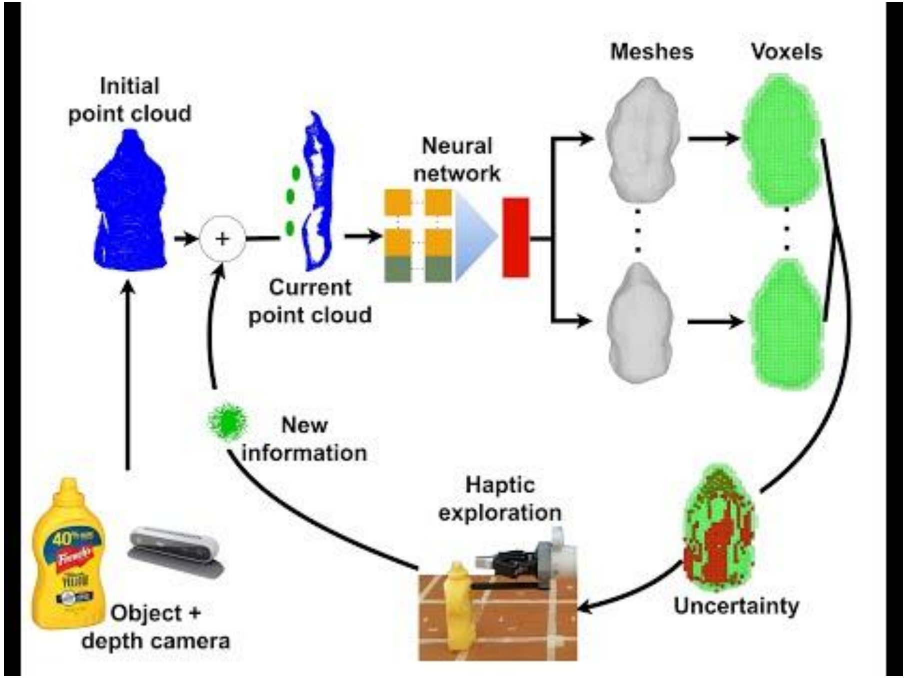


Pliska, M., Mares, M., Straka, Z., Stoudek, P., Stepanova, K. and Hoffmann, M. (2022), 'Single-grasp deformable object classification: the effect of gripper morphology, sensing modalities, and action parameters'. [under review]

Active visuo-haptic shape completion

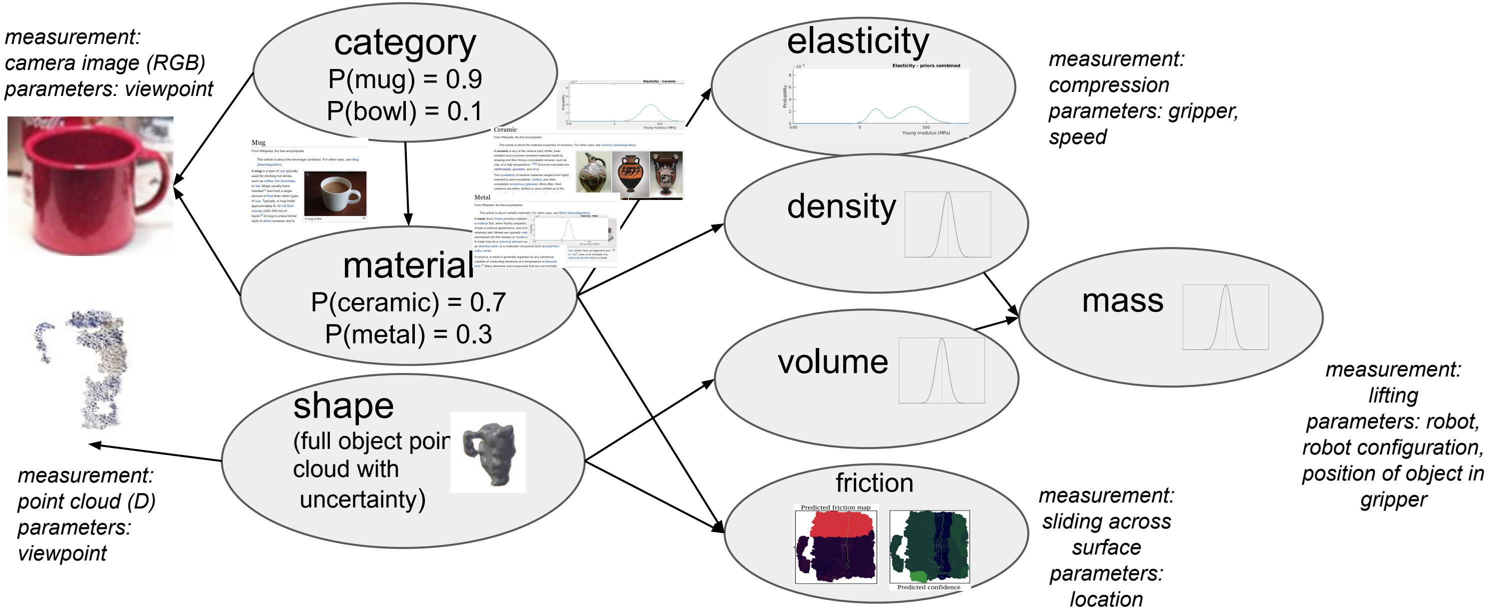
- computes where to touch objects based on reconstruction uncertainty from single input point cloud
- experimental evaluation of reconstruction accuracy against five baselines
- higher grasp success rates than the baseline





Rustler, L., Lundell, J., Behrens, J. K., Kyrki, V., & Hoffmann, M. (2022). 'Active Visuo-Haptic Object Shape Completion'. *IEEE Robotics and Automation Letters* ***7*** *(2), 5254-5261*. https://youtu.be/iZF4ph4zMEA

Embodied Perception-Action-Learning Loop



Exploratory action selection

calculate uncertainty of object attributes → calculate expected information gain of different measurements → perform most promising measurement → Bayes update of network with measurement

Kružliak, A. (2021), 'Exploratory action selection to learn object properties through robot manipulation', Bachelor thesis, Faculty of Electrical Engineering, Czech Technical University in Prague. [link to thesis page][pdf][**received Dean's Award**]

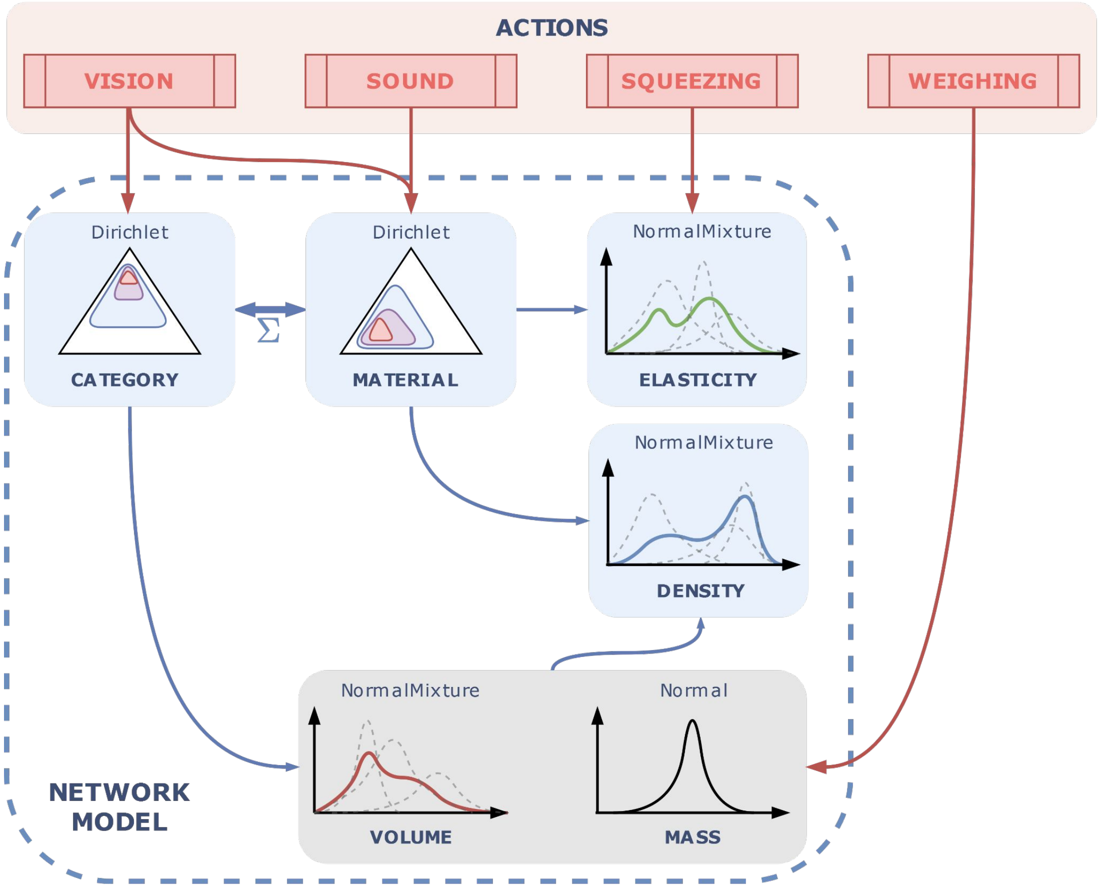
Inference from the Bayesian Network

CATEGORY and **MATERIAL** weights are adjusted based on direct probability observations or observations from other mixture nodes.

The inference is sample-based using MCMC (Markov Chain Monte Carlo) methods using PyMC3.

Network handling utilizing NetworkX back-end.

PyMC3 NetworkX Network Analysis in Python



Resources

- Books / book sections
    - Chapter 12: Grasping and manipulation in Lynch, K. M., & Park, F. C. (2017). Modern robotics. Cambridge University Press.
    - Sections 2.9 and 6.2 in Nenchev, D. N., Konno, A., & Tsujita, T. (2018). Humanoid robots: Modeling and control. Butterworth-Heinemann.
    - Kao, I., Lynch, K. M., & Burdick, J. W. (2016). Contact modeling and manipulation. In Springer Handbook of Robotics (pp. 931-954). Springer, Cham.
    - Prattichizzo, D., & Trinkle, J. C. (2016). Grasping. In Springer handbook of robotics (pp. 955-988). Springer, Cham.
- Online resources
    - https://modernrobotics.northwestern.edu/nu-gm-book-resource/grasping-and-manipulation/ - **video lectures by Kevin Lynch** (covering Lynch, K. M., & Park, F. C. (2017). Modern robotics.)
    - Lecture slides by Ville Kyrki: Robotic manipulation: Lectures 7 and 8. https://mycourses.aalto.fi/course/view.php?id=32938&section=1
    - GraspIt! Simulator: https://graspit-simulator.github.io/
    - iCub Gazebo grasping benchmark: https://robotology.github.io/icub-gazebo-grasping-sandbox/
    - MIT RoboSeminar - Ken Goldberg - The New Wave in Robot Grasping: https://youtu.be/ATDrSWZXuwk
- Articles
    - Kleeberger, K., Bormann, R., Kraus, W., & Huber, M. F. (2020). A survey on learning-based robotic grasping. *Current Robotics Reports*, *1*(4), 239-249.