



Robotics

Miao Li

Fall 2023, Wuhan University

WeChat: 15527576906

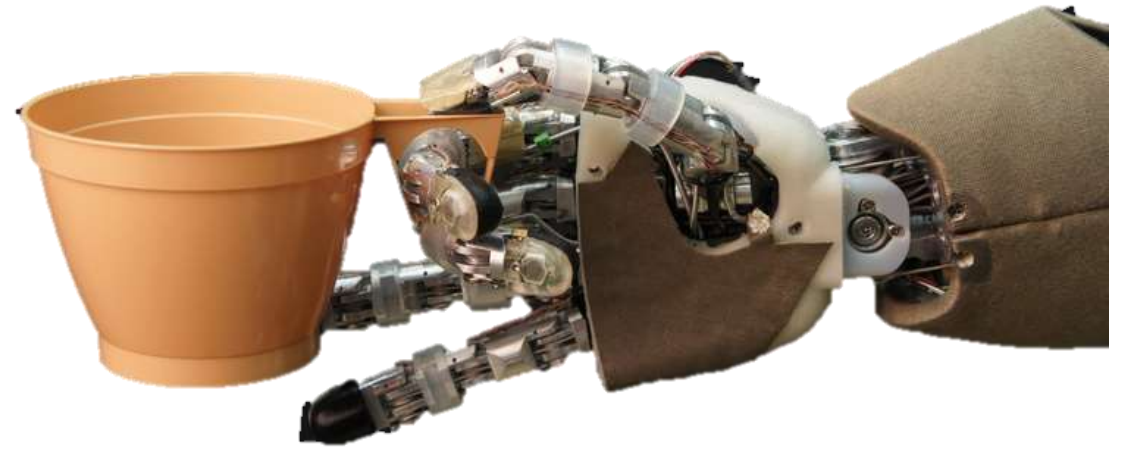
Email: limiao712@gmail.com





Today

- **Design and Modelling of typical robotic arm/hand**
- **DH parameters**
- **Kinematics, IK, Jacobian**
- **Soft hand**
- **Grasp planning**
- **Simulation tool introduction**
- **Group list**





Why robotic hand and grasping?



Grasping and manipulation is an essential skill for humans



Why robotic hand and grasping?



Understanding
human behavior



“..... If there is something that is so natural for humans or animals to do well, but it is so hard for us to control such an engineer system, that most likely means this is the **correct challenge** that deserves our investigation..... ”

----- Russ Tedrake



Why robotic hand and grasping?



(a) Da Vinci Robot

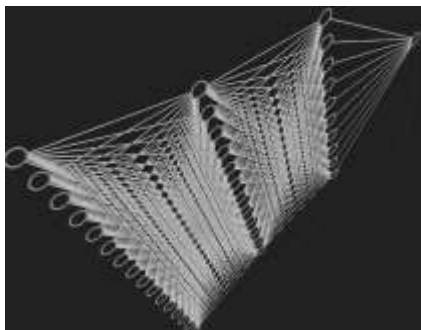


(b) Hand

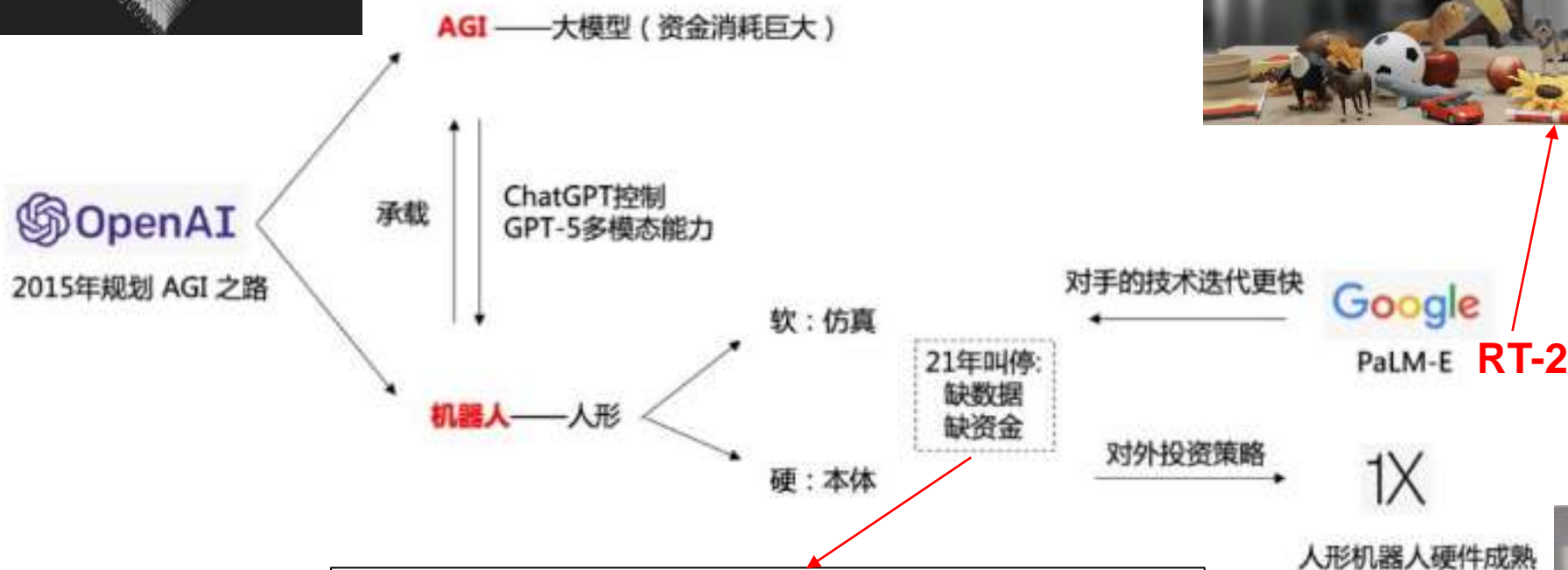


(c) HMI





Why hand?



“So it turns out that we can make a gigantic progress whenever we have access to data. And I kept all of our machinery unsupervised, [using] reinforcement learning — [it] work[s] extremely well. There [are] actually plenty of domains that are very, very rich with data. And ultimately that was holding us back in terms of robotics,” Zaremba said. “The decision [to disband the robotics team] was quite hard for me. But I got the realization some time ago that actually, that’s for the best from the perspective of the company.”



The maker of ChatGPT, OpenAI, invested in 1X, a company that makes humanoid robots designed to do human jobs. The humanoid is named EVE.



Why hand?

• Safety First.



1X tests every EVE in real-world scenarios before they're deployed and ensures that every operator is trained to work with them. EVE's soft, organically-inspired mechanics make them safer from the inside out, so they're ready for your workplace.

• Balanced Performance.



EVE moves like us, so they can meet your needs. 1X engineers EVE for high precision and gentle strength, with wheels and gripper-hands, so they can open your doors, take your elevators, and fit into your work in a natural, intuitive way.

• Smart Behavior.

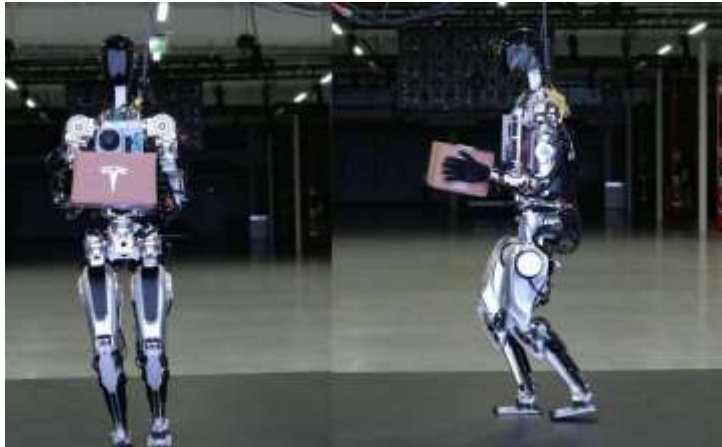


Androids embody artificial intelligence. 1X combines thoughtfully-designed bodies with advanced AI minds, so they can move throughout your space and you can control them from a distance.

1X



Boston Dynamics



Tesla Bot



Sanctuary

https://www.youtube.com/watch?v=YHk7Czt_k6Lg&ab_channel=SanctuaryAI



Why hand?

RODNEY BROOKS

Robots, AI, and other stuff

BLOG MIT ROBUST.AI



Search ...

RECENT POSTS

What Will Transformers Transform?

Predictions Scorecard, 2023 January 01

Where are the crewed eVTOL videos?

My IEEE Spectrum Columns and Articles

No front seat occupants; adventures in autonomous ride services

CATEGORIES

[Dated Predictions](#)

[Essays](#)

[Quick Takes](#)

POST: RESEARCH NEEDED ON ROBOT HANDS

JANUARY 30, 2017 — QUICK TAKES

Research Needed on Robot Hands

rodneybrooks.com/research-needed-on-robot-hands/

This is a short piece I wrote for a workshop on what are good things to work on in robotics research.

One measure of success of robots is how many of them get deployed doing real work in the real world. One way to get more robots deployed is to reduce the friction that comes up during typical deployments. For intelligent robots in factories there are many sources of friction, some sociological, some financial, some concerning takt time, some concerning PLCs and other automation, **but perhaps the most friction that can be attributed to a lack of relevant research results is the problem of getting a gripper suitable for a particular task.**

Today in factories the most commonly used grippers are either a set of custom configured suction cups to pick up a very particular object, or one of a myriad of parallel jaw grippers varying over a large number of parameters, and custom fingers, again carefully selected for a particular object. In both cases just one grasp is used for that particular object. Getting the right gripper for initial deployment can be a weeks long source of friction, and then changing the gripper when new objects are to be handled is another source of friction. Furthermore, grip failure can be a major source of run time errors.

Human hands just work. Give them an object from a very wide class of objects and they grip that object, usually with a wide variety of possible grips. They sense when the grip is failing and adjust. They work reliably and quickly.



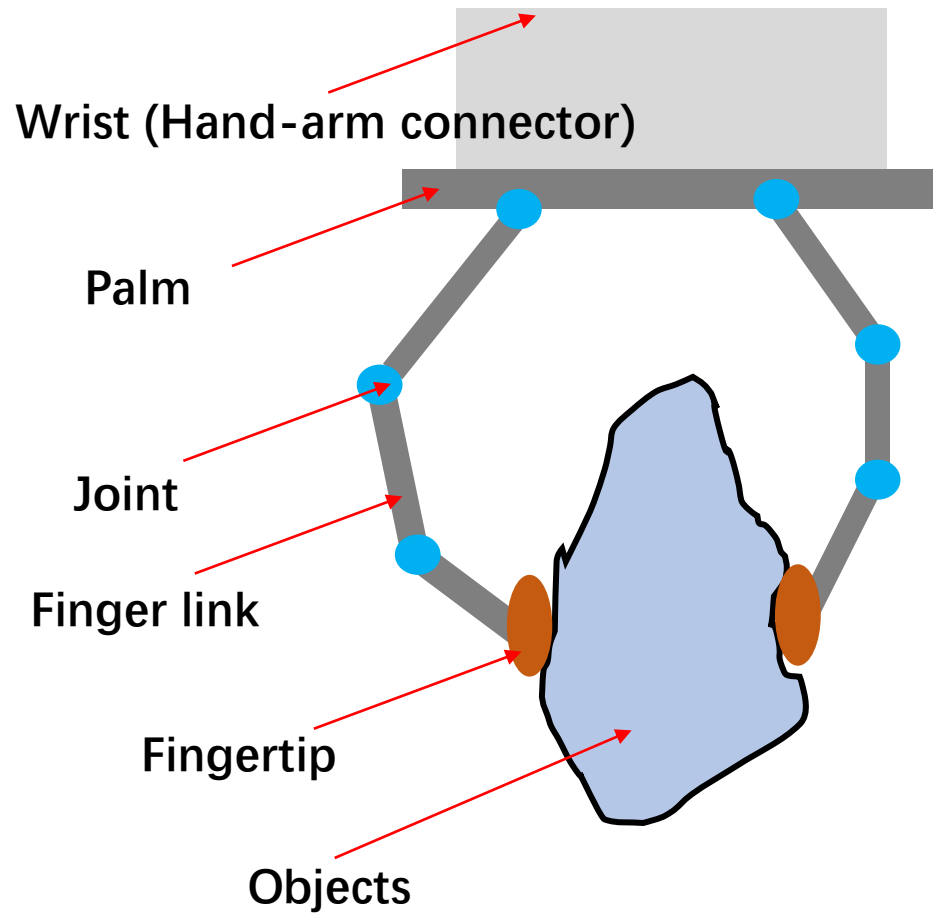
My strawman is that we will need concurrent progress in at least five areas, each feeding off the other, in order to come up with truly useful and general robot hands:

- new (low cost) mechanisms for both kinematics and force control
- materials to act as a skin (grasp properties and longevity)
- long life sensors that can be embedded in the skin and mechanism
- algorithms to dynamically adjust grasps based on sensing
- learning mechanisms on visual/3D data to inform hands for pregrasp

<https://rodneybrooks.com/research-needed-on-robot-hands/>



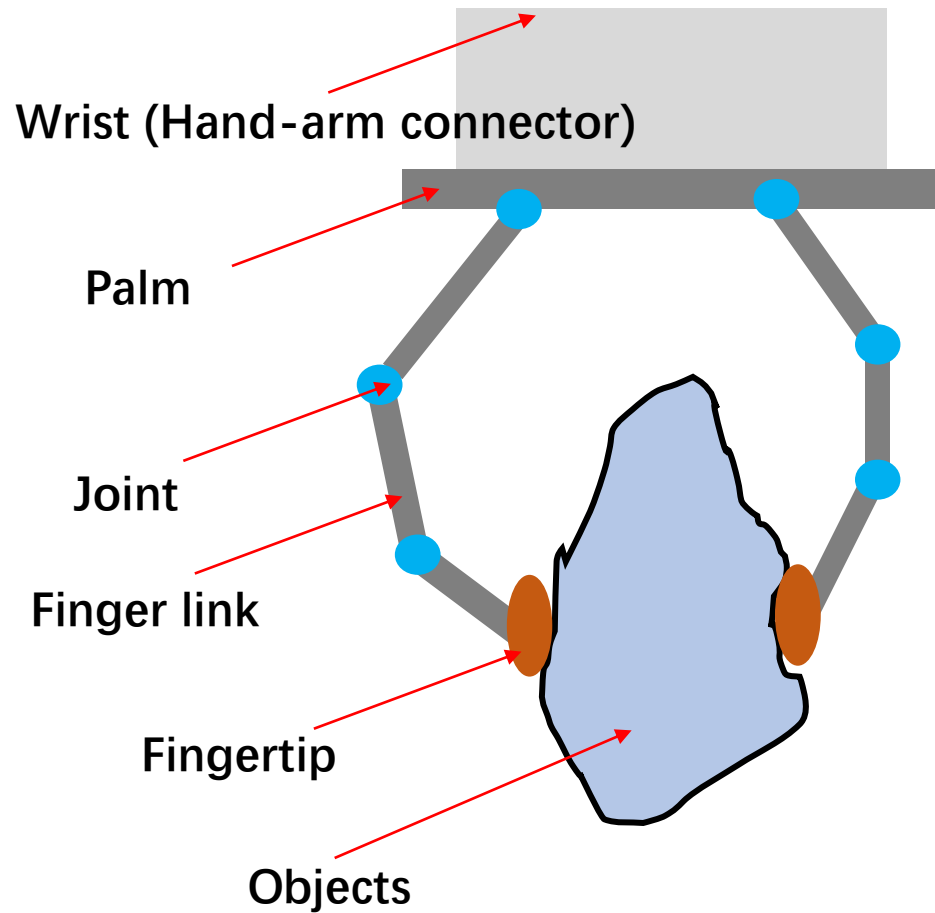
A robotic grasping system



How to design, model, and control a robotic hand?



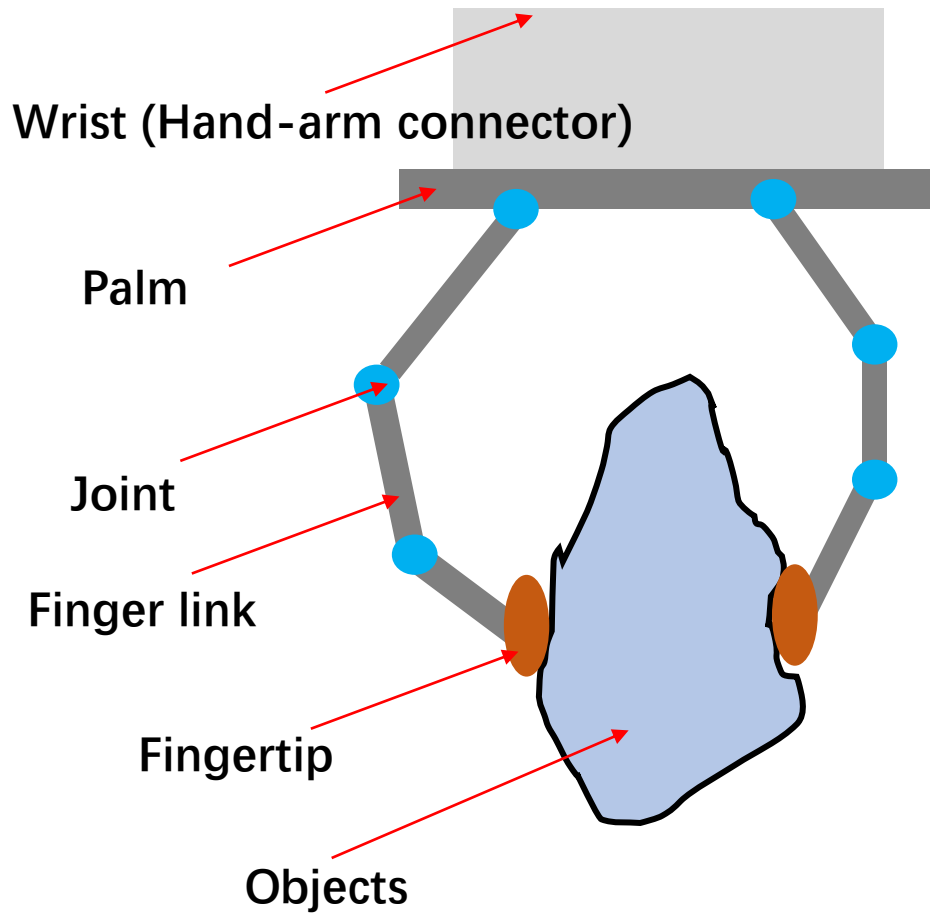
A robotic grasping system



- How to design the hardware including the palm, finger link, finger joint and fingertip?
- How to integrate a proper sensor for a hand, especially a tactile sensor?
- How to choose the configuration of the fingers, including the degree of freedoms for each finger?
- How to model the kinematics of the finger and the hand?
- How to model the contact between the fingertips and the object?
- How to plan a suitable/optimal grasp for different objects?
- How to control the hand for a given task ?
- How to dexterously manipulate an object?



A robotic grasping system



Journals & Magazines > IEEE Robotics & Automation Ma... > Volume: 28 Issue: 2

A Novel Soft Robotic Hand Design With Human-Inspired Soft Palm: Achieving a Great Diversity of Grasps

Publisher: IEEE

[Cite This](#)

[PDF](#)

nature communications

[Explore content](#)

[About the journal](#)

[Publish with us](#)

[nature](#) > [nature communications](#) > [articles](#) > [article](#)

Article | [Open Access](#) | Published: 14 December 2021

Integrated linkage-driven dexterous anthropomorphic robotic hand

[Uikyum Kim](#), [Dawoon Jung](#), [Heeyoen Jeong](#), [Jongwoo Park](#), [Hyun-Mok Jung](#), [Joono Cheong](#), [Hyouk Ryeol Choi](#), [Hyunmin Do](#) & [Chanhun Park](#)

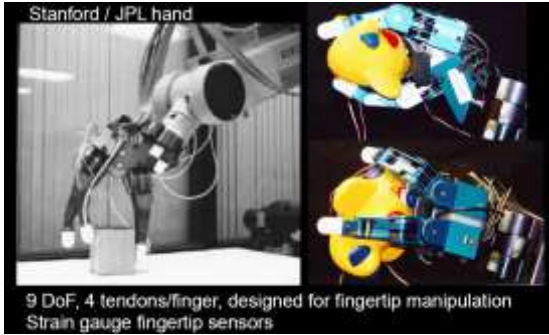
[Nature Communications](#) 12, Article number: 7177 (2021) | [View this article](#)

28k Accesses | 34 Citations | 289 Altmetric | [Metrics](#)

Still a hot topic in robotics



A robotic grasping system



9 DoF, 4 tendons/finger, designed for fingertip manipulation
Strain gauge fingertip sensors



Utah/MIT Hand



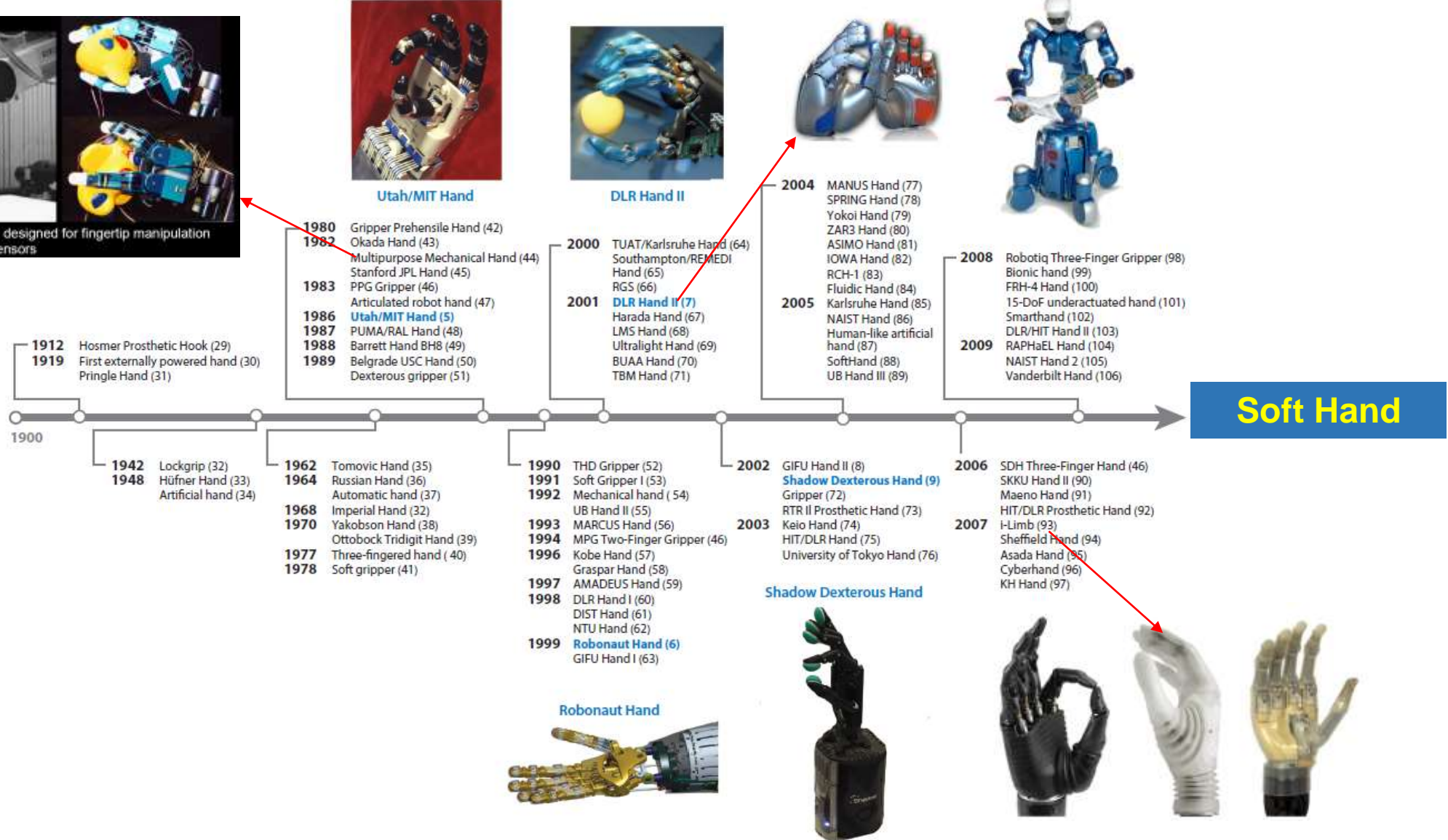
DLR Hand II



- 2004 MANUS Hand (77)
- SPRING Hand (78)
- Yokoi Hand (79)
- ZAR3 Hand (80)
- ASIMO Hand (81)
- IOWA Hand (82)
- RCH-1 (83)
- Fluidic Hand (84)
- 2005 Karlsruhe Hand (85)
- NAIST Hand (86)
- Human-like artificial hand (87)
- SoftHand (88)
- UB Hand III (89)



- 2008 Robotiq Three-Finger Gripper (98)
- Bionic hand (99)
- FRH-4 Hand (100)
- 15-DoF underactuated hand (101)
- Smarthand (102)
- DLR/HIT Hand II (103)
- RAPHaEL Hand (104)
- NAIST Hand 2 (105)
- Vanderbilt Hand (106)



Soft Hand



Robonaut Hand

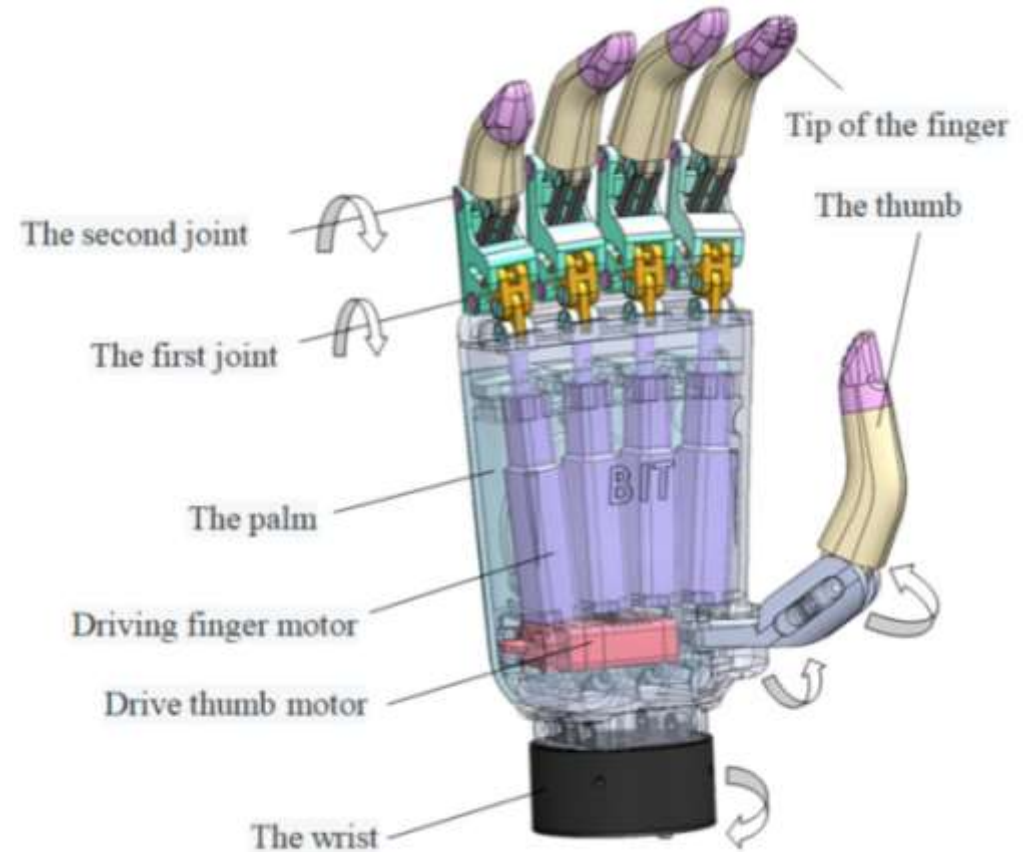
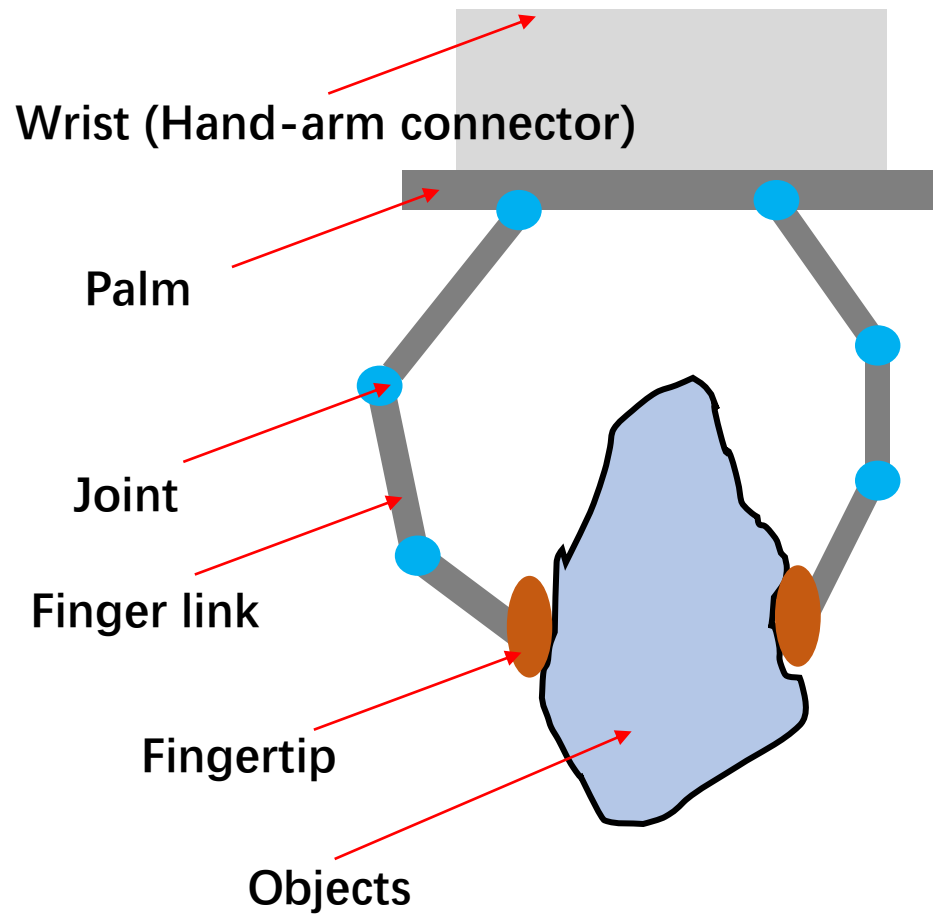


Shadow Dexterous Hand



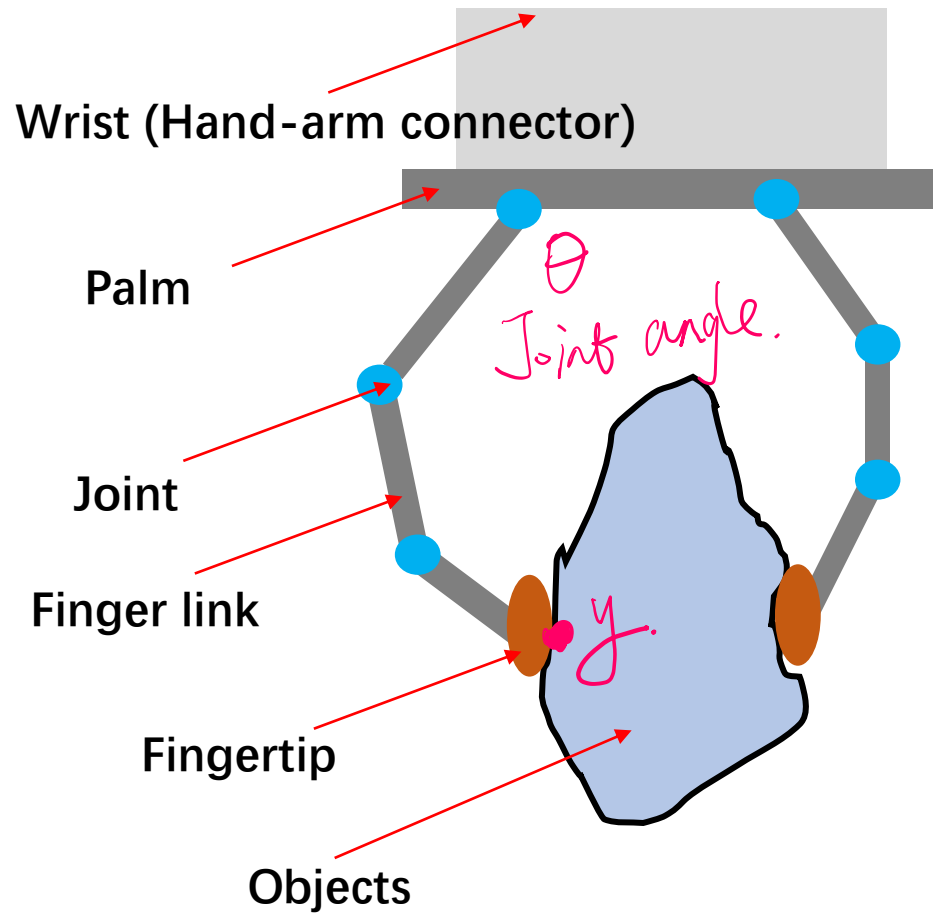


A robotic grasping system





Hand Kinematics



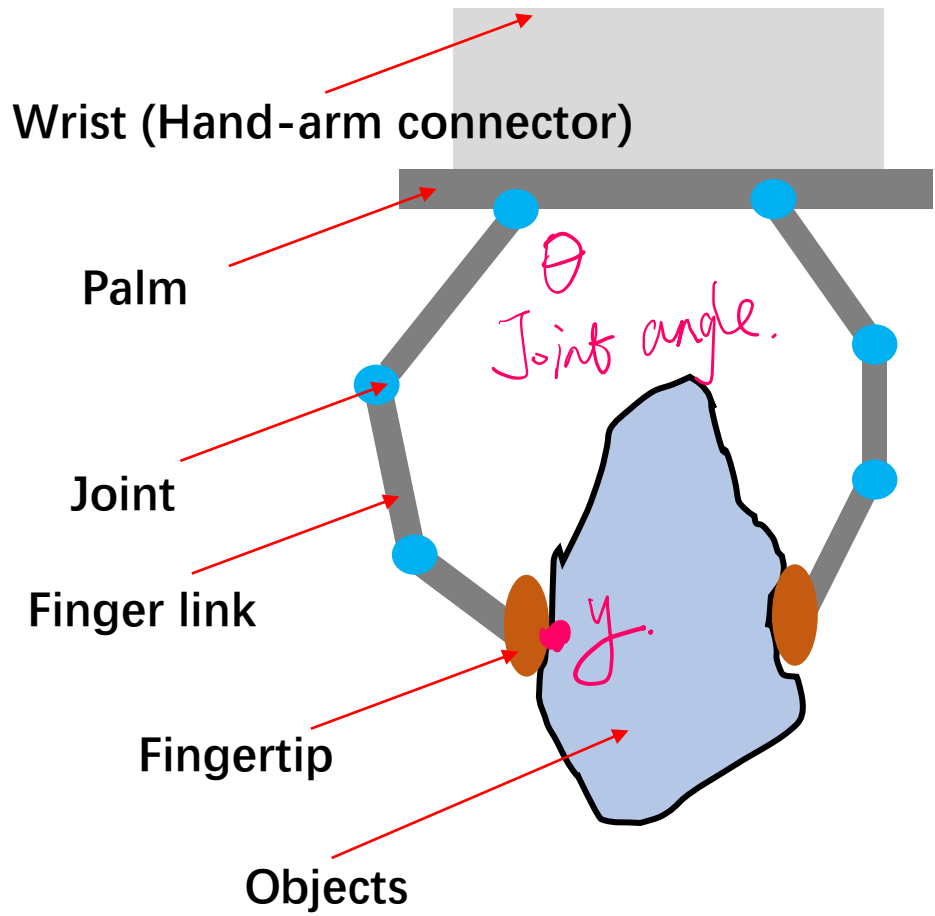
$$y = f(\theta)$$

↓ ↓

pos of fingertip Joint Angle.



Hand Kinematics



$$y = f(\theta)$$

pos of fingertip } Joint Angle.

★ How to get this mapping?

We need a coordinate frame!



Hand Kinematics

We are interested in **two** kinematics topics

Forward Kinematics (angles to position)

| | |
|---------------------|--|
| What you are given: | The length of each link The angle of each joint |
| What you can find: | The position of any point (i.e. it's (x, y, z) coordinates) |

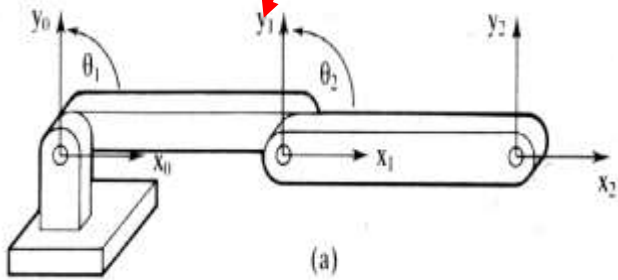
Inverse Kinematics (position to angles)

| | |
|---------------------|--|
| What you are given: | The length of each link The position of some point on the robot |
| What you can find: | The angles of each joint needed to obtain that position |

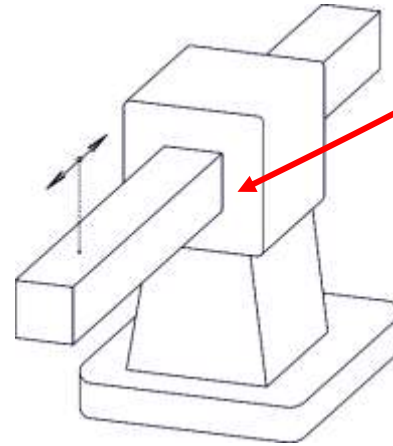


Hand Kinematics- Joints

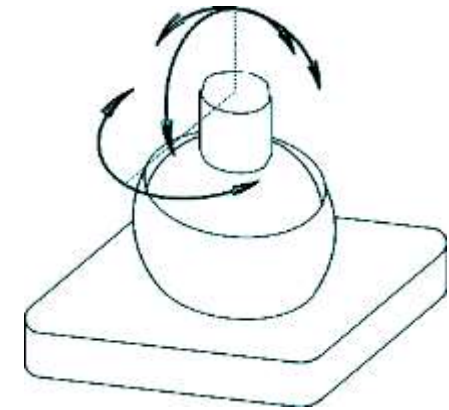
$$q_i = \begin{cases} \theta_i & : \text{ joint } i \text{ revolute} \\ d_i & : \text{ joint } i \text{ prismatic} \end{cases}$$



Revolute Joint
1 DOF



Prismatic Joint
1 DOF (linear)

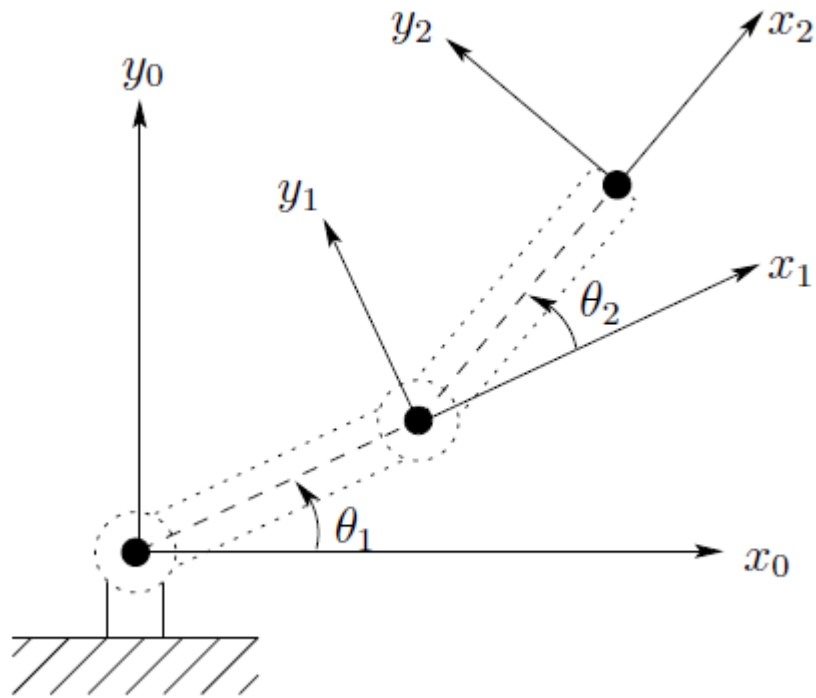


Spherical Joint
3 DOF





Hand Kinematics- An 2D example



Position

$$\begin{aligned}x &= x_2 = \alpha_1 \cos \theta_1 + \alpha_2 \cos(\theta_1 + \theta_2) \\y &= y_2 = \alpha_1 \sin \theta_1 + \alpha_2 \sin(\theta_1 + \theta_2),\end{aligned}$$

Orientation Matrix

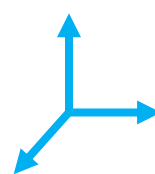
$$\begin{aligned}x_2 \cdot x_0 &= \cos(\theta_1 + \theta_2); & x_2 \cdot y_0 &= \sin(\theta_1 + \theta_2) \\y_2 \cdot x_0 &= \sin(\theta_1 + \theta_2); & y_2 \cdot y_0 &= \cos(\theta_1 + \theta_2)\end{aligned}$$



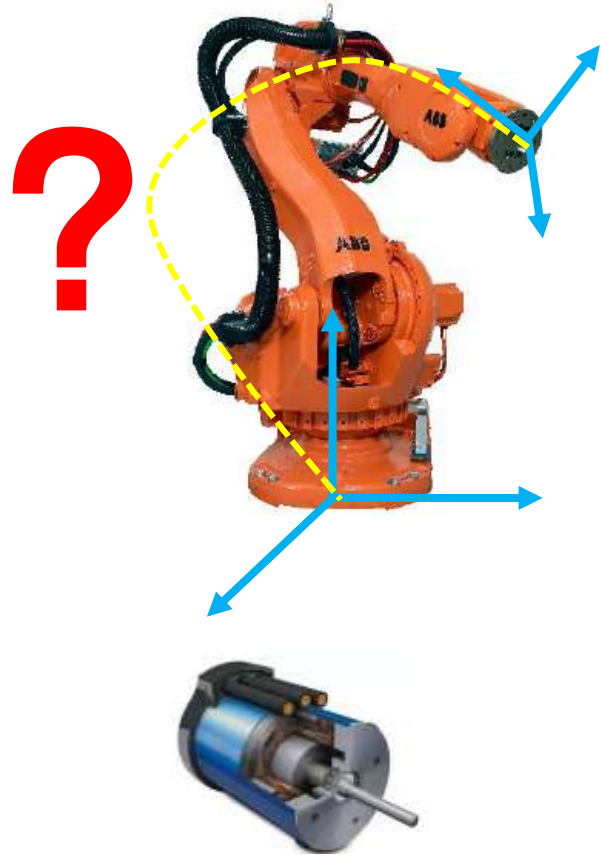
$$\begin{bmatrix} x_2 \cdot x_0 & y_2 \cdot x_0 \\ x_2 \cdot y_0 & y_2 \cdot y_0 \end{bmatrix} = \begin{bmatrix} \cos(\theta_1 + \theta_2) & -\sin(\theta_1 + \theta_2) \\ \sin(\theta_1 + \theta_2) & \cos(\theta_1 + \theta_2) \end{bmatrix}$$



Hand Kinematics- FK



$$H = \begin{bmatrix} R & d \\ \mathbf{0} & 1 \end{bmatrix}; R \in SO(3).$$

$$q_i = \begin{cases} \theta_i & : \text{joint } i \text{ revolute} \\ d_i & : \text{joint } i \text{ prismatic} \end{cases}$$


$$A_i = A_i(q_i). \text{--- } o_i x_i y_i z_i \text{ with respect to } o_{i-1} x_{i-1} y_{i-1} z_{i-1}$$

$$T_j^i = A_{i+1} A_{i+2} \dots A_{j-1} A_j \text{ if } i < j$$

$$T_j^i = I \text{ if } i = j$$

$$T_j^i = (T_i^j)^{-1} \text{ if } j > i.$$

$o_j x_j y_j z_j$ with respect to $o_i x_i y_i z_i$

$$H = \begin{bmatrix} R_n^0 & o_n^0 \\ 0 & 1 \end{bmatrix} \quad H = T_n^0 = A_1(q_1) \dots A_n(q_n)$$

$$A_i = \begin{bmatrix} R_i^{i-1} & o_i^{i-1} \\ 0 & 1 \end{bmatrix}$$

All about FK ?



Hand Kinematics DH Parameters

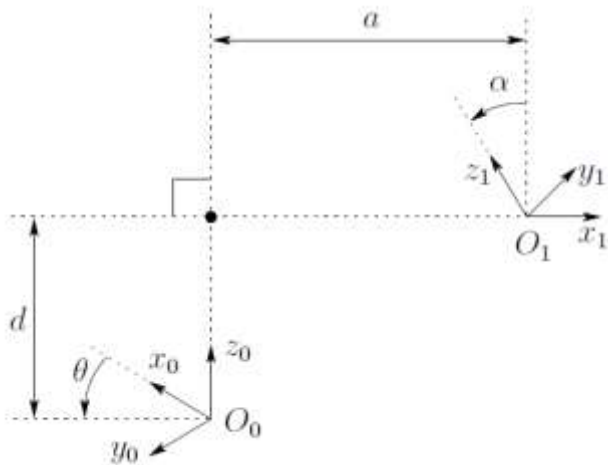
$$A_i = \begin{bmatrix} R_i^{i-1} & O_i^{i-1} \\ 0 & 1 \end{bmatrix}$$

?

How many parameters?

~~6~~

4



(DH1) The axis x_1 is perpendicular to the axis z_0

(DH2) The axis x_1 intersects the axis z_0

1. Denavit, Jacques; Hartenberg, Richard Scheunemann (1955). "A kinematic notation for lower-pair mechanisms based on matrices". *Trans ASME J. Appl. Mech.* **23**: 215–221.
2. Hartenberg, Richard Scheunemann; Denavit, Jacques (1965). *Kinematic synthesis of linkages*. McGraw-Hill series in mechanical engineering. New York: McGraw-Hill. p. 435. [Archived](#) from the original on 2013-09-28. Retrieved 2012-01-13.



Hand Kinematics- FK

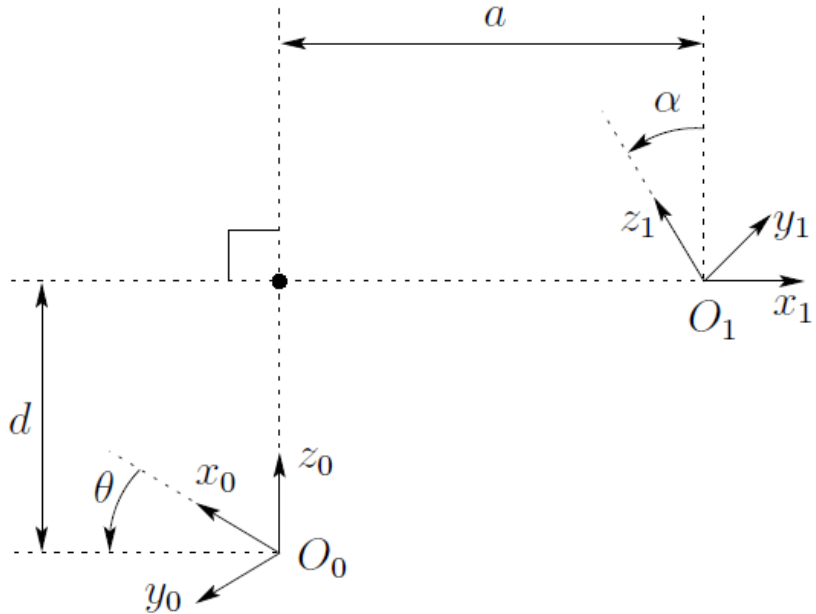


Exist only 4 params (Spong 1989)

(DH1) The axis x_1 is perpendicular to the axis z_0

(DH2) The axis x_1 intersects the axis z_0

$$A = Rot_{z,\theta} Trans_{z,d} Trans_{x,a} Rot_{x,\alpha}$$

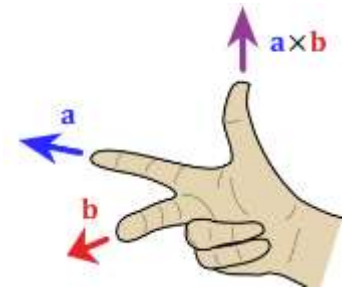
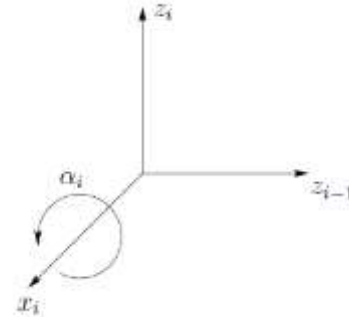
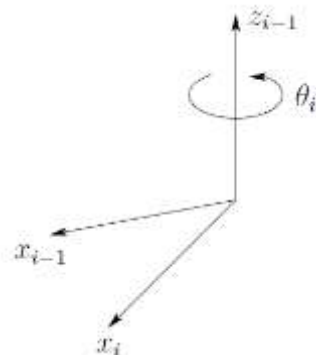


θ_i = the angle between x_{i-1} and x_i measured about z_{i-1}

d_i = distance along z_{i-1} from o_{i-1} to the intersection of the x_i and z_{i-1} axes. d_i is variable if joint i is prismatic.

a_i = distance along x_i from o_i to the intersection of the x_i and z_{i-1} axes.

α_i = the angle between z_{i-1} and z_i measured about x_i



right-hand rule



Hand Kinematics- FK



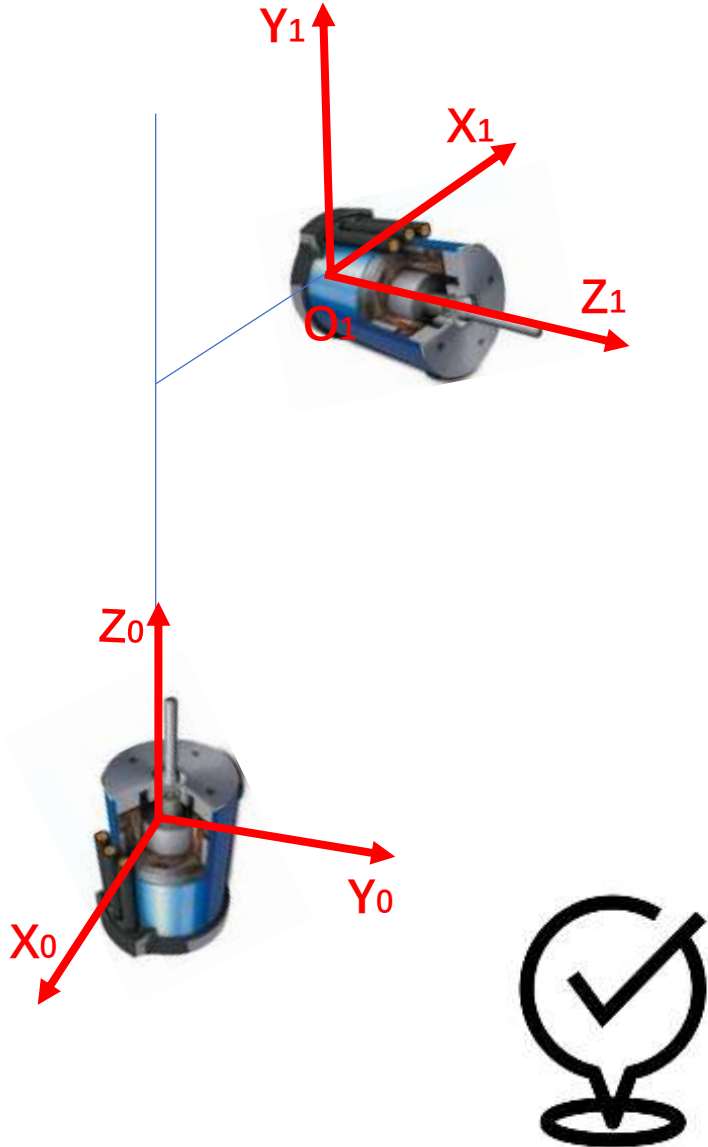
$$A_i = \begin{bmatrix} R_i^{i-1} & O_i^{i-1} \\ 0 & 1 \end{bmatrix} \quad ?$$

$$A = Rot_{z,\theta} Trans_{z,d} Trans_{x,a} Rot_{x,\alpha}$$

$$\begin{aligned} A_i &= Rot_{z,\theta_i} Trans_{z,d_i} Trans_{x,a_i} Rot_{x,\alpha_i} \\ &= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & 0 \\ s_{\theta_i} & c_{\theta_i} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_{\alpha_i} & -s_{\alpha_i} & 0 \\ 0 & s_{\alpha_i} & c_{\alpha_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}s_{\alpha_i} & a_ics_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}s_{\alpha_i} & a_iss_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$



Hand Kinematics- DH param procedure



1 Step 1: Locate and label the joint axes z_0, \dots, z_{n-1} .

2 Step 2: Establish the base frame. Set the origin anywhere on the z_0 -axis. The x_0 and y_0 axes are chosen conveniently to form a right-hand frame.

For $i = 1, \dots, n - 1$, perform Steps 3 to 5.

3 Step 3: Locate the origin o_i where the common normal to z_i and z_{i-1} intersects z_i . If z_i intersects z_{i-1} locate o_i at this intersection. If z_i and z_{i-1} are parallel, locate o_i in any convenient position along z_i .

4 Step 4: Establish x_i along the common normal between z_{i-1} and z_i through o_i , or in the direction normal to the $z_{i-1} - z_i$ plane if z_{i-1} and z_i intersect.

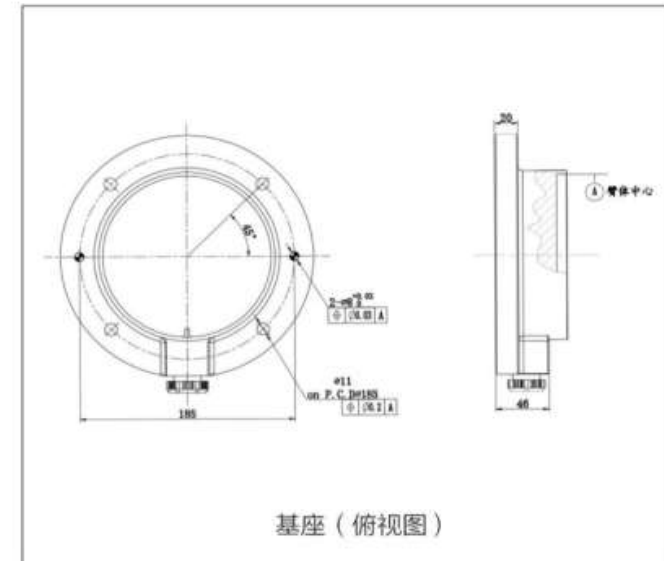
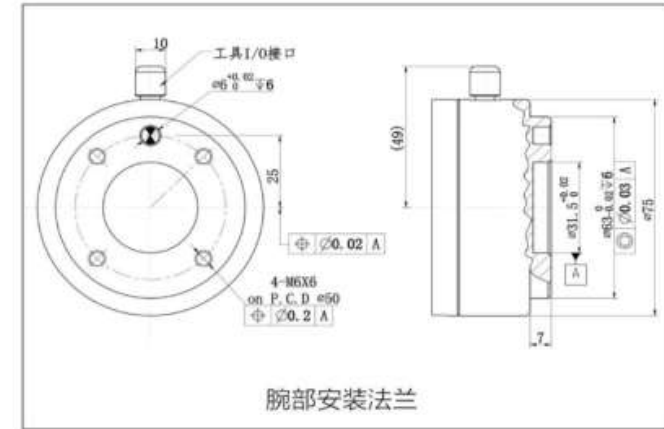
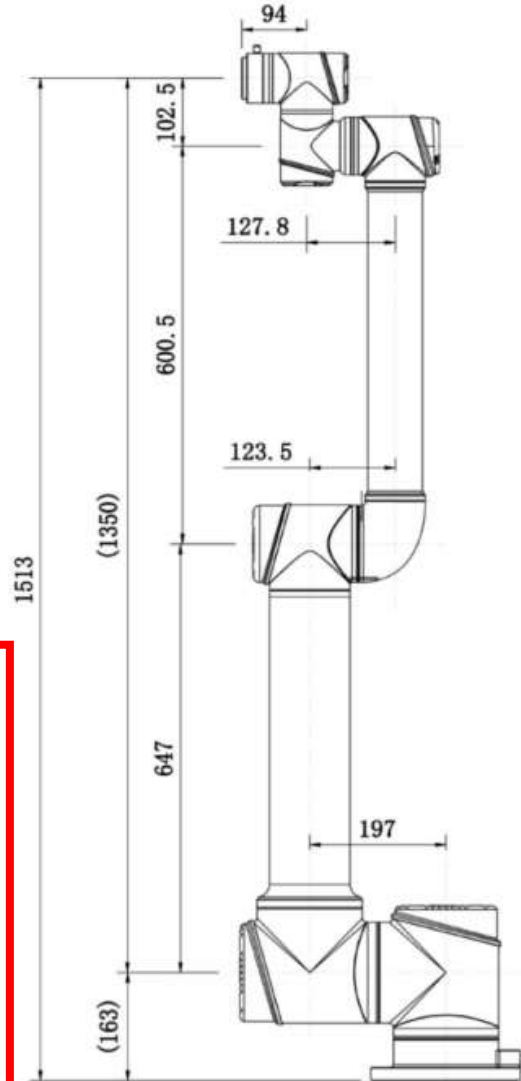
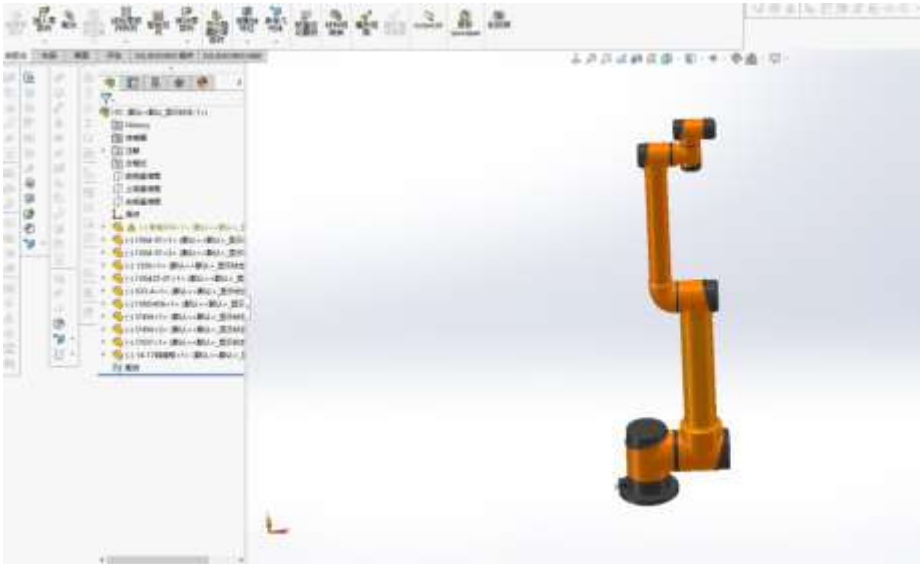
5 Step 5: Establish y_i to complete a right-hand frame.

6 Step 6: Establish the end-effector frame $o_n x_n y_n z_n$. Assuming the n -th joint is revolute, set $z_n = a$ along the direction z_{n-1} . Establish the origin o_n conveniently along z_n , preferably at the center of the gripper or at the tip of any tool that the manipulator may be carrying. Set $y_n = s$ in the direction of the gripper closure and set $x_n = n$ as $s \times a$. If the tool is not a simple gripper set x_n and y_n conveniently to form a right-hand frame.

7 Step 7: Create a table of link parameters $a_i, d_i, \alpha_i, \theta_i$.



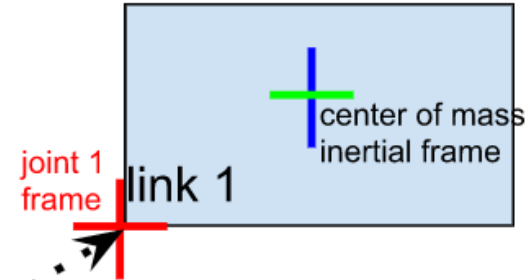
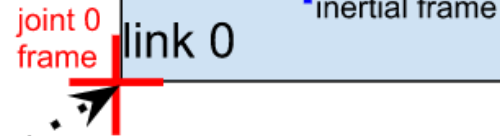
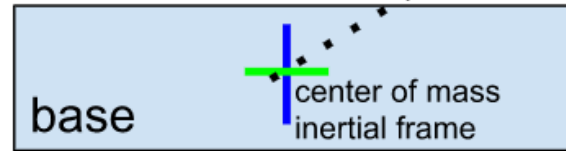
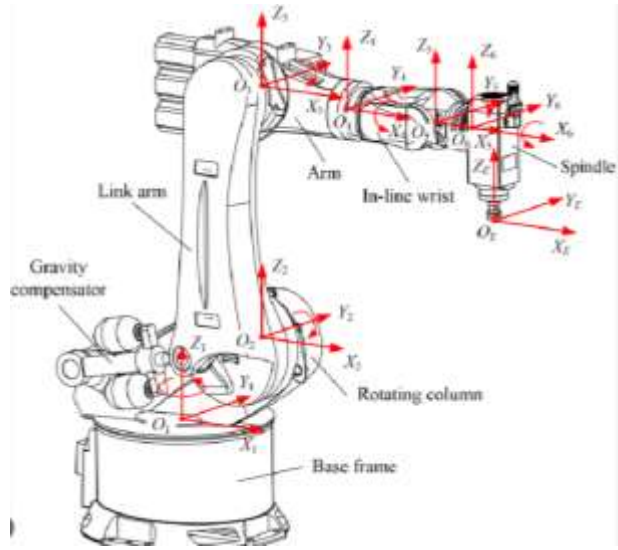
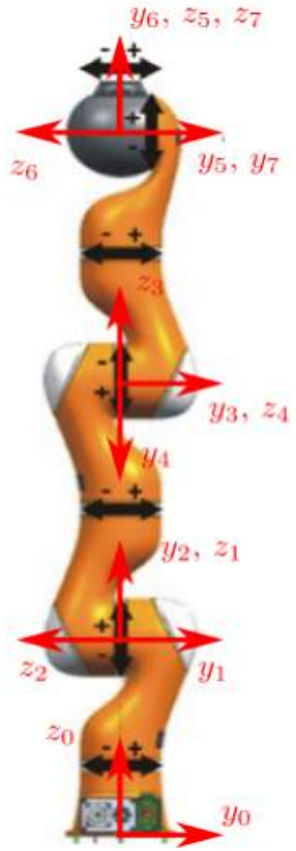
Hand Kinematics- DH in practice



| DH参数 | d(m) | $\theta(^{\circ})$ | a(m) | $\alpha(^{\circ})$ |
|------|--------|--------------------|---------|--------------------|
| 关节1 | 0 | 0 | 0 | 0 |
| 关节2 | 0.163 | 90 | 0 | 90 |
| 关节3 | 0 | 0 | -0.647 | 0 |
| 关节4 | 0 | 0 | -0.6005 | 0 |
| 关节5 | 0.2013 | 0 | 0 | 90 |
| 关节6 | 0.1025 | 0 | 0 | -90 |



Hand Kinematics- FK in Practice



<https://docs.google.com/document/d/10sXEhzFRSnvFcl3XxNGhnD4N2SedqwdAvK3dsihxVUA/edit>



Hand Kinematics- FK in PyBullet



Bullet Real-Time Physics Simulation

Home of Bullet and PyBullet: physics simulation for games, visual effects, robotics and reinforcement learning.

RECENT POSTS

- Kubric: a scalable dataset generator
- NeuroSim: Augmenting Differentiable Simulators with Neural Networks
- PyBullet in a colab
- Learning Agile Robotic Locomotion Skills by Imitating Animals
- Assistive gym @ ICRA, 2020

SELECTED VIDEO

[Top 5 AI ...](#)

KUBRIC: A SCALABLE DATASET GENERATOR

by Typhontaur
MARCH 21, 2022

Kubric is an open-source Python framework that interfaces with PyBullet and Blender to generate photo-realistic scenes, with rich annotations, and seamlessly scales to large jobs distributed over thousands of machines, and generating TBs of data. Kubric can generate semi-realistic synthetic multi-object videos with rich annotations such as instance segmentation masks, depth maps, and optical flow.

You can find the [paper](#) here, or access the [source code](#) from github.

LAST TOPICS POSTS

- [Bullet 6 Bsp](#)
New engine: Ancestor Ancestor Ancestor ...
by Typhontaur
August 25, 2023 8:37 pm
btMultiBodyDynamicsWorld and soft bodies
Hi, This might sound like a stupid question... In the case one decides to use the ...
by glaezy
August 10, 2023 10:10 am
Unstable behavior on shapes like cylinder or box
I will suggest you several ways to solve shape instability: You can increase the collision ...
by sharase23
July 6, 2023 9:02 am
Compiling for ARM/Raspberry Pi
Optionally, you can install the compiled Bullet library on your Ubuntu machine or the Raspberry ...
by sharase23
July 6, 2023 8:58 am
Issue moving objects that are constrained together:
Instead of moving the kinematic object abruptly from -5.f to 1.f, consider implementing a smooth ...
by sharase23
July 6, 2023 8:57 am

<https://docs.google.com/document/d/10sXEhzFRSnvFcl3XxNGhnD4N2SedqwdAvK3dsihxVUA/edit>

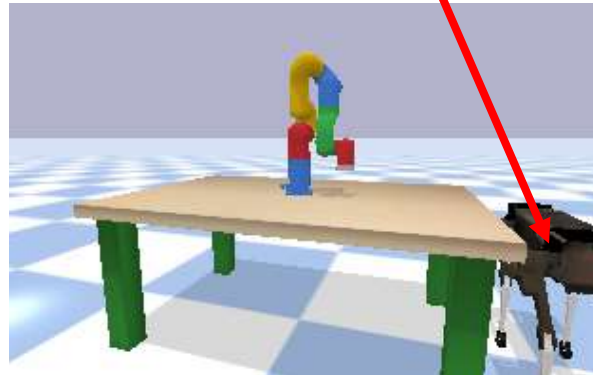
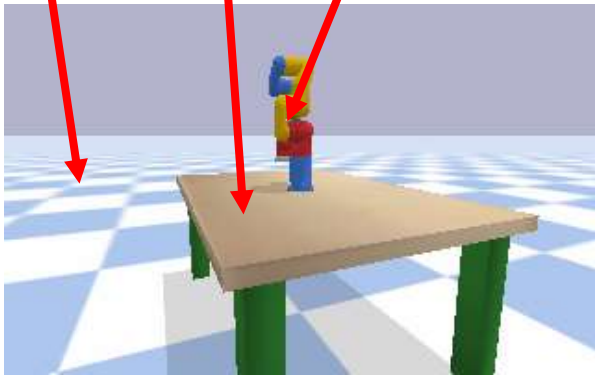


Hand Kinematics- FK in Practice



```
p.resetSimulation()
p.configureDebugVisualizer(p.COV_ENABLE_GUI)
useFixedBase = True
flags = p.URDF_INITIALIZE_SAT_FEATURES

plane_pos = [0, 0, -0.625]
plane = p.loadURDF("plane.urdf", plane_pos, flags = flags, useFixedBase=useFixedBase)
table_pos = [0, 0, -0.625]
table = p.loadURDF("table/table.urdf", table_pos, flags = flags, useFixedBase=useFixedBase)
xarm = p.loadURDF("xarm/xarm6_robot.urdf", flags = flags, useFixedBase=useFixedBase)
xarm = p.loadURDF("laikago/laikago_toes.urdf", [1, 0, -0.15], [0, 0.5, 0.5, 0], flags = flags, useFixedBase=useFixedBase)
```



<https://docs.google.com/document/d/10sXEhzFRSnvFcl3XxNGhnD4N2SedqwdAvK3dsihxVUA/edit>



Hand Kinematics- FK in Practice



getLinkState(s)

You can also query the Cartesian world position and orientation for the center of mass of each link using `getLinkState`. It will also report the local inertial frame of the center of mass to the URDF link frame, to make it easier to compute the graphics/visualization frame.

getLinkState input parameters

| | | | |
|----------|--------------------------|-----|--|
| required | bodyUniqueId | int | body unique id as returned by loadURDF etc |
| required | linkIndex | int | link index |
| optional | computeLinkVelocity | int | If set to 1, the Cartesian world velocity will be computed and returned. |
| optional | computeForwardKinematics | int | if set to 1 (or True), the Cartesian world position/orientation will be recomputed using forward kinematics. |
| optional | physicsClientId | int | if you are connected to multiple servers, you can pick one. |

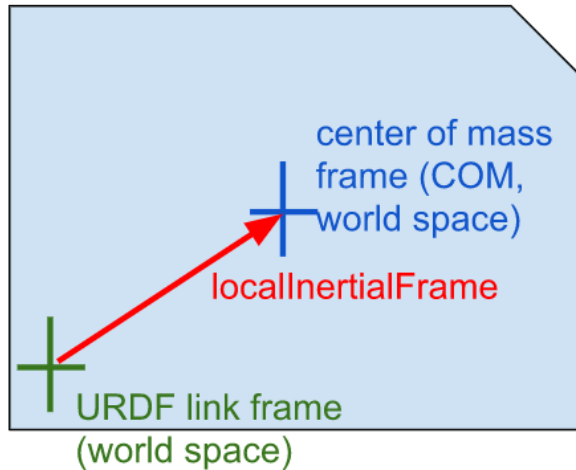
<https://docs.google.com/document/d/10sXEhzFRSnvFcl3XxNGhnD4N2SedqwdAvK3dsihxVUA/edit>



Hand Kinematics- FK in Practice



getLinkState return values



| | | |
|-------------------------------|------------------------|---|
| linkWorldPosition | vec3, list of 3 floats | Cartesian position of center of mass |
| linkWorldOrientation | vec4, list of 4 floats | Cartesian orientation of center of mass, in quaternion [x,y,z,w] |
| localInertialFramePosition | vec3, list of 3 floats | local position offset of inertial frame (center of mass) expressed in the URDF link frame |
| localInertialFrameOrientation | vec4, list of 4 floats | local orientation (quaternion [x,y,z,w]) offset of the inertial frame expressed in URDF link frame. |
| worldLinkFramePosition | vec3, list of 3 floats | world position of the URDF link frame |
| worldLinkFrameOrientation | vec4, list of 4 floats | world orientation of the URDF link frame |
| worldLinkLinearVelocity | vec3, list of 3 floats | Cartesian world velocity. Only returned if computeLinkVelocity non-zero. |
| worldLinkAngularVelocity | vec3, list of 3 floats | Cartesian world velocity. Only returned if computeLinkVelocity non-zero. |

<https://docs.google.com/document/d/10sXEhzFRSnvFcl3XxNGhnD4N2SedqwdAvK3dsi hxVUA/edit>



Hand Kinematics- FK in Practice

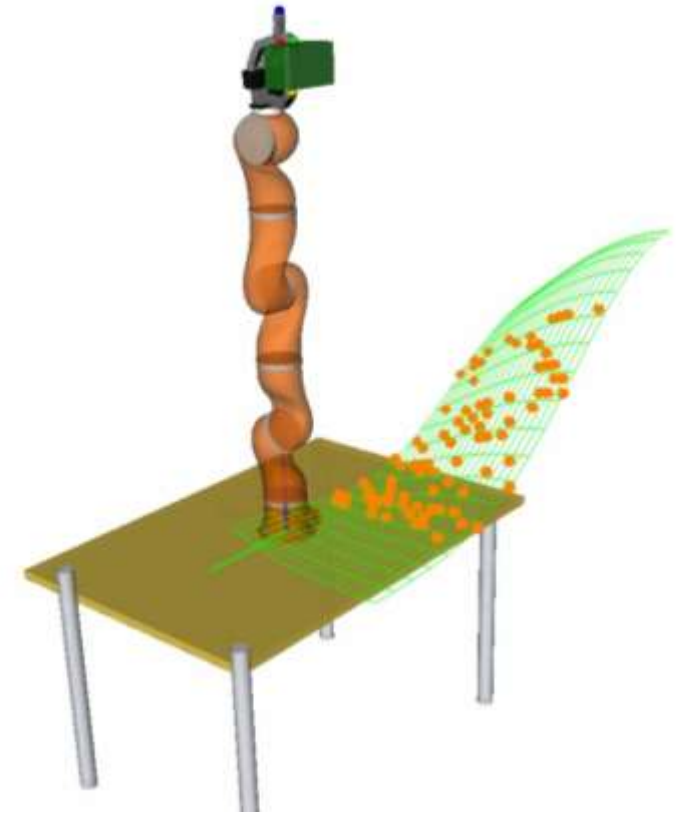


PyBullet Quickstart Guide

[Erwin Coumans](#), [Yunfei Bai](#), 2016-2023

Visit [desktop doc](#), [forums](#), github [discussions](#) and star [Bullet!](#)

| | | | |
|--|-----------|---|-----------|
| Introduction | 2 | Synthetic Camera Rendering | 48 |
| Hello PyBullet World | 3 | computeView/ProjectionMatrix | 48 |
| connect, disconnect, bullet_client | 3 | getCameraImage | 49 |
| setGravity | 7 | getVisualShapeData | 52 |
| loadURDF, loadSDF, loadMJCF | 8 | changeVisualShape, loadTexture | 53 |
| saveState, saveBullet, restoreState | 11 | Collision Detection Queries | 53 |
| createCollisionShape/VisualShape | 12 | getOverlappingObjects, getAABB | 53 |
| createMultiBody | 15 | getContactPoints, getClosestPoints | 54 |
| stepSimulation, performCollisionDetection | 17 | rayTest, rayTestBatch | 56 |
| getBasePositionAndOrientation | 18 | getCollisionShapeData | 57 |
| resetBasePositionAndOrientation | 19 | Enable/Disable Collisions | 58 |
| Transforms: Position and Orientation | 19 | Inverse Dynamics, Kinematics | 61 |
| Controlling a robot | 22 | calculateInverseDynamics(2) | 61 |
| Base, Joints, Links | 22 | calculateJacobian, MassMatrix | 61 |
| getNumJoints, getJointInfo | 22 | calculateInverseKinematics(2) | 63 |
| setJointMotorControl2/Array | 24 | Reinforcement Learning Gym Envs | 66 |
| getJointState(s), resetJointState | 28 | Environments and Data | 66 |
| enableJointForceTorqueSensor | 30 | Stable Baselines & ARS, ES,... | 70 |
| getLinkState(s) | 31 | Virtual Reality | 73 |
| getBaseVelocity, resetBaseVelocity | 33 | getVREvents, setVRCameraState | 73 |
| applyExternalForce/Torque | 33 | Debug GUI, Lines, Text, Parameters | 75 |
| getNumBodies, getBodyInfo, getBodyUniqueId, removeBody | 34 | addUserDebugLine, Points, Text, Parameter | 75 |
| createConstraint, removeConstraint, changeConstraint | 34 | addUserData (get, sync, remove) | 79 |
| getNumConstraints, getConstraintUniqueId | 36 | configureDebugVisualizer | 80 |
| getConstraintInfo/State | 36 | get/resetDebugVisualizerCamera | 80 |
| getDynamicsInfo/changeDynamics | 37 | getKeyboardEvents, getMouseEvents | 82 |
| setTimeStep | 40 | Plugins | 83 |
| setPhysicsEngineParameter | 40 | loadPlugin, executePluginCommand | 83 |
| resetSimulation | 43 | Build and install PyBullet | 84 |
| startStateLogging/stopStateLogging | 43 | Support, Tips, Citation | 87 |
| Deformables and Cloth (FEM, PBD) | 45 | | |
| loadSoftBody/loadURDF | 46 | | |
| createSoftBodyAnchor | 47 | | |



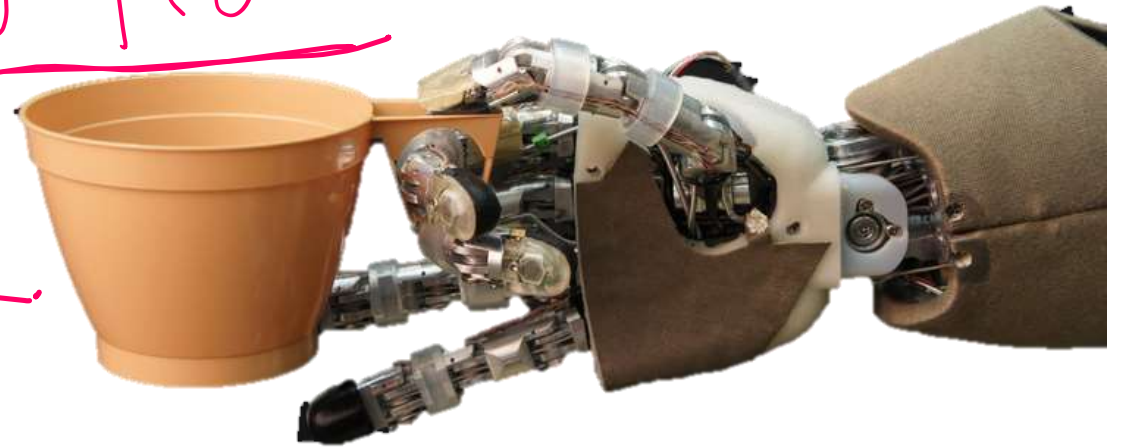


Today

- Design and Modelling of typical robotic arm/hand
- DH parameters
- Kinematics, IK, Jacobian
- Soft hand
- Grasp planning
- Simulation tool introduction
- Group list

Handwritten notes in red ink:

- An arrow points from the text "Kinematics, IK, Jacobian" to the forward kinematics equation: $\dot{y} = f(\dot{\theta})$
- A checkmark is placed next to the inverse kinematics equation: $\theta = f^{-1}(y)$





Robotic Hand Dynamics

STABILITY OF VARIABLE GRASP STIFFNESS CONTROL

In this part, we address the control stability of varying stiffness in robotic grasping using a two-fingered examples, as shown in Fig. B.1(a). To this end, a detailed formulation for the dynamics of the hand-object system, the contact constraints and the soft fingertip formulation on the grasping object are provided. The overall dynamics of the system to grasp the object stably is derived from the proof of contact constraints.

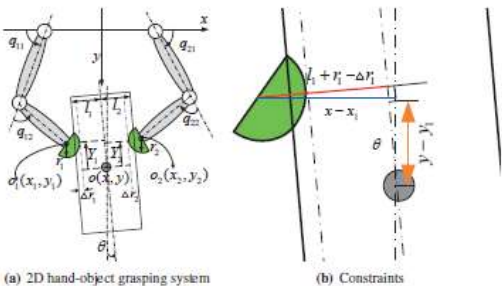


Figure B.1: (a) The hand-object system in 2D. Each finger has 2 DOFs with soft fingertips. (b) The constraint that the fingertip should keep contact with the object's surface.

B.1 DYNAMICS

In this part, we first consider the contact model of the fingertips and the constraints involved in the hand-object system. Then the overall dynamics of the system is formulated. The notations in this section are explained in Fig. B.1 to simplify the understanding of the grasping dynamics.

B.1.1 CONTACT MODEL OF SOFT FINGERTIP

$$f = c_1(\Delta r)^2 + c_2 \frac{d}{dt} \Delta r \quad (\text{B.1})$$

where c_1 and c_2 are positive constant parameters which depend on the material of the fingertip, and Δr is the deformation at the fingertip. The fingertip should keep contact with the object surface, as shown in Fig. B.1(b), which can be expressed as follows

$$l_1 + r_1 - \Delta r_1 = (x - x_1) \cos \theta - (y - y_1) \sin \theta \quad (\text{B.2})$$

$$l_2 + r_2 - \Delta r_2 = -(x - x_2) \cos \theta + (y - y_2) \sin \theta \quad (\text{B.3})$$

B.1.2 ROLLING CONSTRAINTS

$$Y_i = (x_i - x) \sin \theta + (y_i - y) \cos \theta \quad (\text{B.5})$$

$$q_{11} + q_{12} + \phi_1 = \pi + \theta \quad (\text{B.6})$$

$$q_{21} + q_{22} + \phi_2 = \pi - \theta \quad (\text{B.7})$$

B.1.3 OVERALL DYNAMICS

The total kinetic energy for the overall system can be described as follows

$$K = \sum_{i=1,2} \frac{1}{2} \dot{\mathbf{q}}_i^T H_i \dot{\mathbf{q}}_i + \frac{1}{2} M (\dot{x}^2 + \dot{y}^2) + \frac{1}{2} I \dot{\theta}^2 \quad (\text{B.8})$$

where $\mathbf{q}_i = [q_{i1}, q_{i2}]^T$ is the vector of finger joints and $H_i \in \mathbb{R}^{2 \times 2}$ is the inertia matrix for each finger, M and I are the mass and inertia matrix of the object respectively.

The total potential energy from deformation can be given as:

$$P = \sum_{i=1,2} \int_0^{\Delta r_i} c_2 \eta^2 d\eta \quad (\text{B.9})$$

Then from the Hamilton's principle, we have

$$\int_{t_0}^{t_1} [\delta(K - P) - \sum_{i=1,2} c_2 \Delta r_i \frac{\partial \Delta r_i}{\partial X^T} \delta X + \sum_{i=1,2} \lambda_i [(r_i - \Delta r_i) \frac{\partial \phi_i}{\partial X^T} + \frac{\partial Y_i}{\partial X^T} \delta X + \sum_{i=1,2} \mathbf{u}_i^T \delta \mathbf{q}_i] dt = 0 \quad (\text{B.10})$$

where $X = [\mathbf{q}_1^T, \mathbf{q}_2^T, x, y, \theta]^T$.

Then we have the overall dynamics for the object-hand system as follows

$$H_s(\mathbf{q}) \ddot{\mathbf{q}}_s + (\frac{1}{2} \dot{H}_s + S_s) \dot{\mathbf{q}}_s + f_s \frac{\partial \Delta r_i}{\partial \mathbf{q}_i^T} - \lambda_i [(r_i - \Delta r_i) \frac{\partial \phi_i}{\partial \mathbf{q}_i^T} + \frac{\partial Y_i}{\partial \mathbf{q}_i^T}] = \mathbf{u}_s \quad (\text{B.11})$$

$$M \ddot{x} + \sum_{i=1,2} [f_i \frac{\partial \Delta r_i}{\partial x} - \lambda_i \frac{\partial Y_i}{\partial x}] = 0 \quad (\text{B.12})$$

$$M \ddot{y} + \sum_{i=1,2} [f_i \frac{\partial \Delta r_i}{\partial y} - \lambda_i \frac{\partial Y_i}{\partial y}] = 0 \quad (\text{B.13})$$

$$I \ddot{\theta} + \sum_{i=1,2} [f_i \frac{\partial \Delta r_i}{\partial \theta} - \lambda_i ((r_i - \Delta r_i) \frac{\partial \phi_i}{\partial \theta} + \frac{\partial Y_i}{\partial \theta})] = 0 \quad (\text{B.14})$$

With the identities in section B.5, the overall dynamics can be simplified as

$$M \ddot{x} - (f_1 - f_2) \cos \theta + (\lambda_1 + \lambda_2) \sin \theta = 0 \quad (\text{B.15})$$

$$M \ddot{y} + (f_1 - f_2) \sin \theta + (\lambda_1 + \lambda_2) \cos \theta = 0 \quad (\text{B.16})$$

$$I \ddot{\theta} - f_1 Y_1 + f_2 Y_2 + l_1 \lambda_1 - l_2 \lambda_2 = 0 \quad (\text{B.17})$$

$$I \ddot{\theta} - f_1 Y_1 + f_2 Y_2 + l_1 \lambda_1 - l_2 \lambda_2 = 0 \quad (\text{B.18})$$

B.2 VARIABLE GRASP STIFFNESS CONTROL

Motivated by the analysis of the overall system dynamics, the following control law is adopted for each finger to achieve stable grasp

$$\mathbf{u}_i = -D_i \dot{\mathbf{q}}_i + k J_i^T (\mathbf{x}_i - \mathbf{x}_d) \quad (\text{B.19})$$

$$\mathbf{x}_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix} \quad \mathbf{x}_d = \frac{1}{2} \begin{bmatrix} x_1 + x_2 \\ y_1 + y_2 \end{bmatrix} \quad (\text{B.20})$$

where D_i is a diagonal positive definite matrix representing the damping gain, $k \in \mathbb{R}^+$ represents the variable stiffness for each fingertip (k is the same value for the two-finger grasp to ensure force balance).

B.3 STABILITY PROOF-1

Taking the sum of inner product of Eq. (B.15) with $\dot{\mathbf{q}}_1$, $i = 1, 2$, Eq. (B.16) with \dot{x} , Eq. (B.17) with \dot{y} , Eq. (B.18) with $\dot{\theta}$, we have

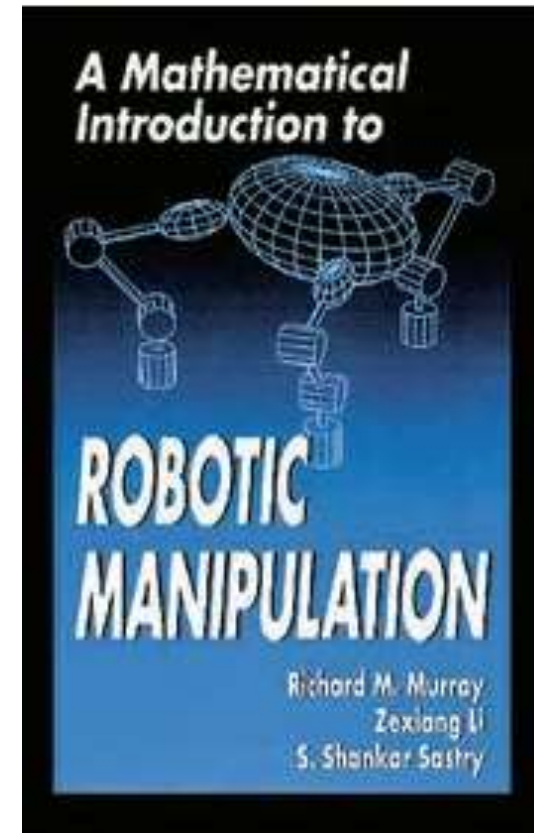
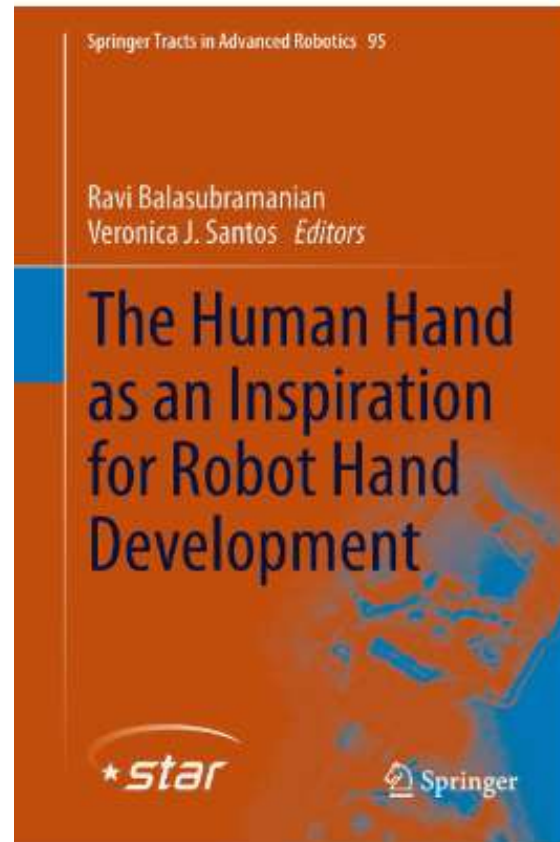
$$\frac{d}{dt} E = - \sum_{i=1,2} (\dot{\mathbf{q}}_i^T D_i \dot{\mathbf{q}}_i + c_2 \Delta r_i^2) + \frac{k}{4} (\mathbf{x}_1 - \mathbf{x}_2)^T (\mathbf{x}_1 - \mathbf{x}_2) \quad (\text{B.21})$$

$$E = K + V + P \quad (\text{B.22})$$

Complicated with many assumptions

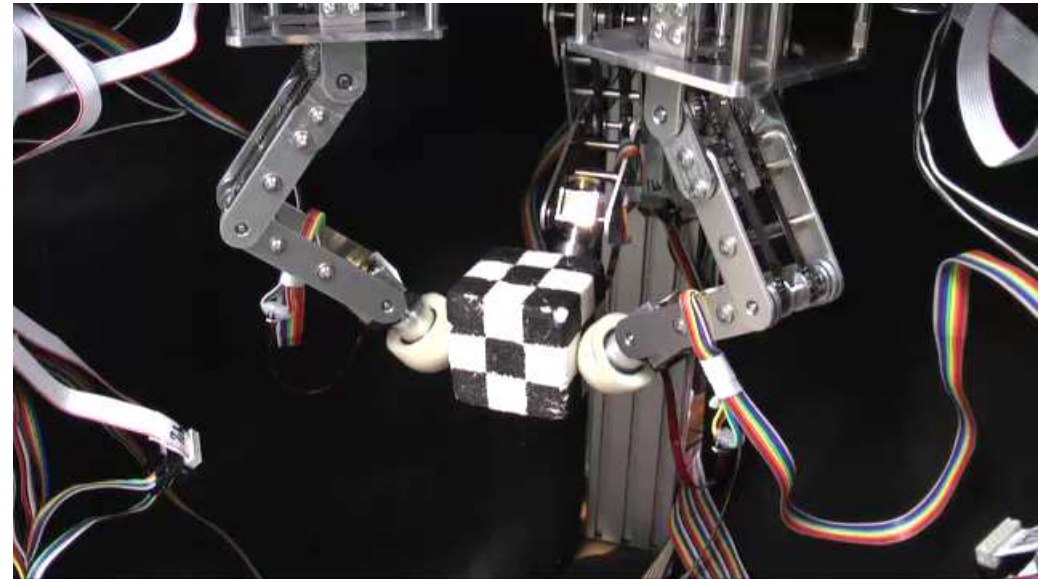
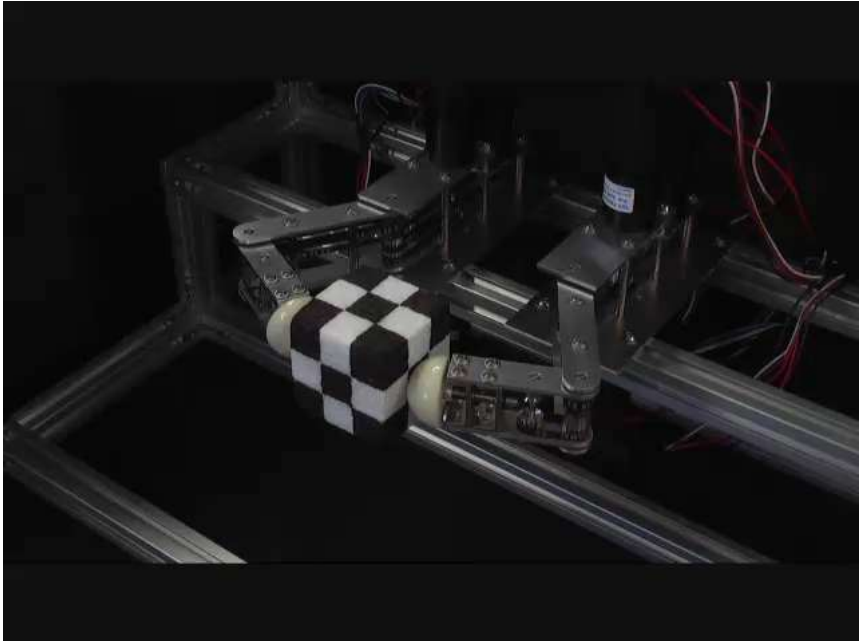
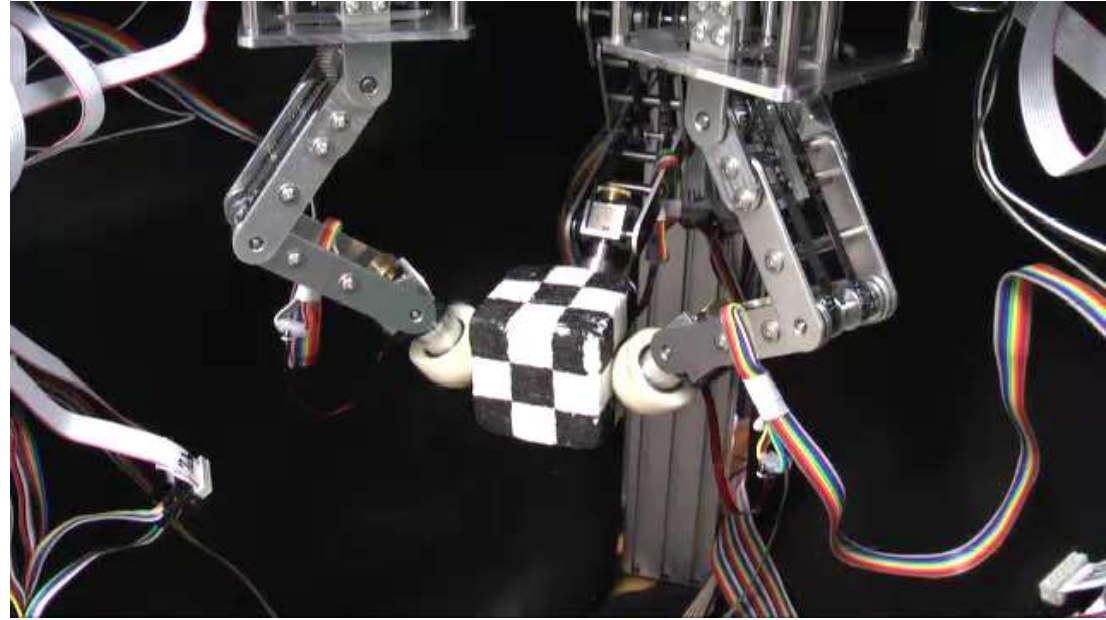
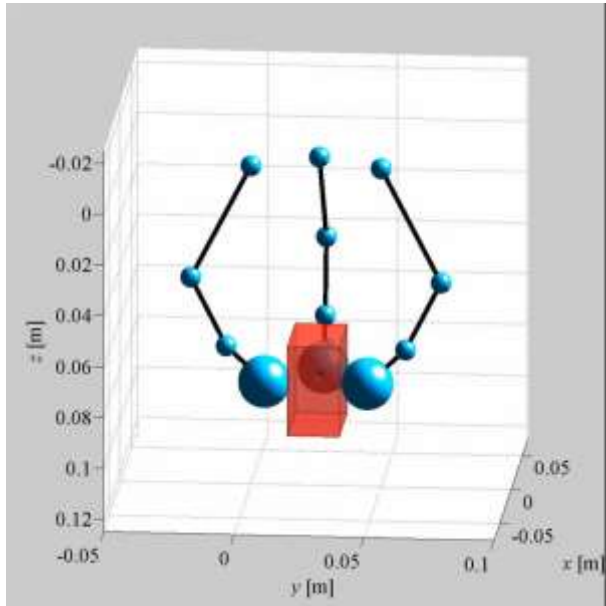


Robotic Hand Dynamics





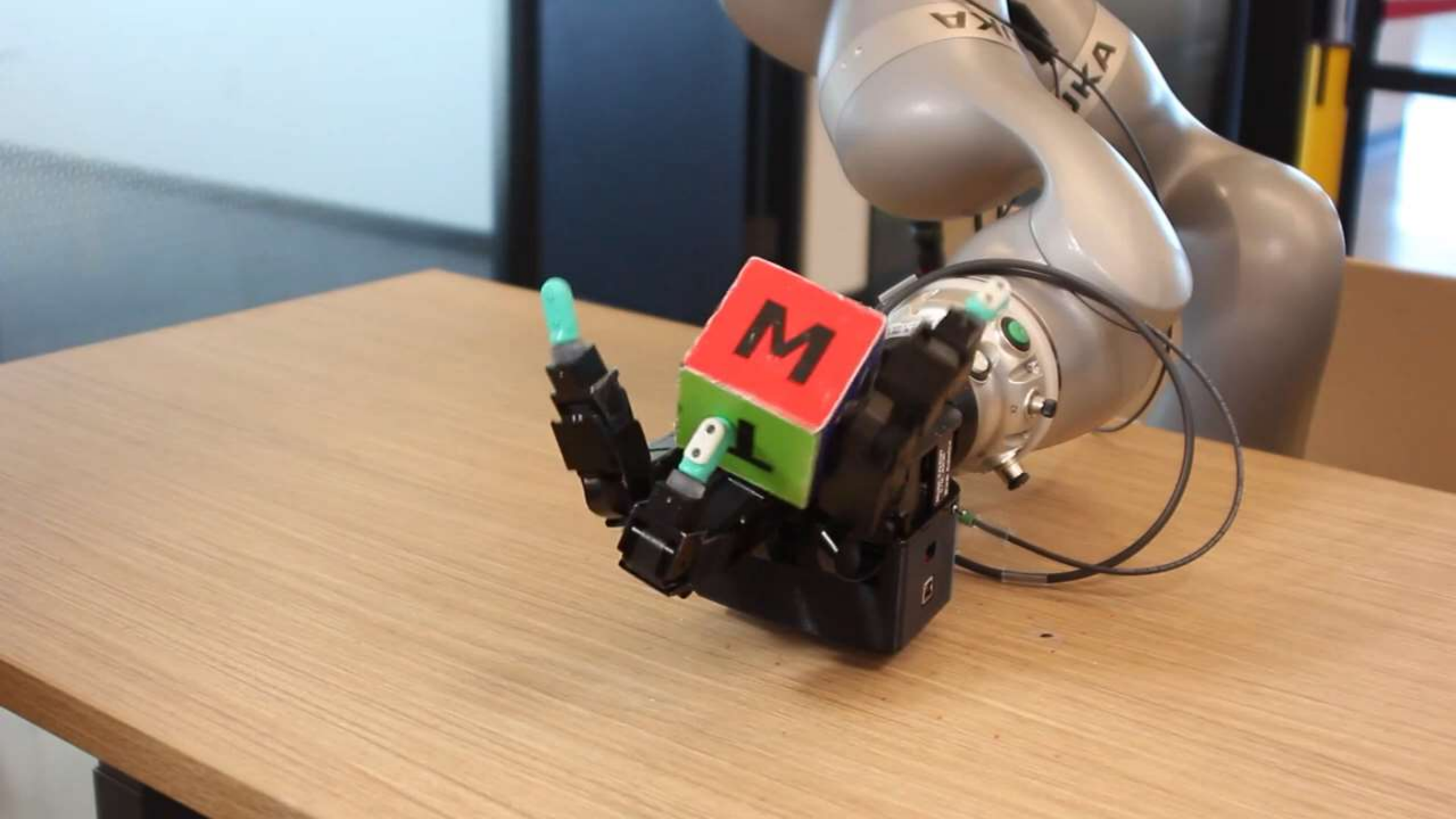
Robotic Hand Dynamics





Opening thoughts on robot hands

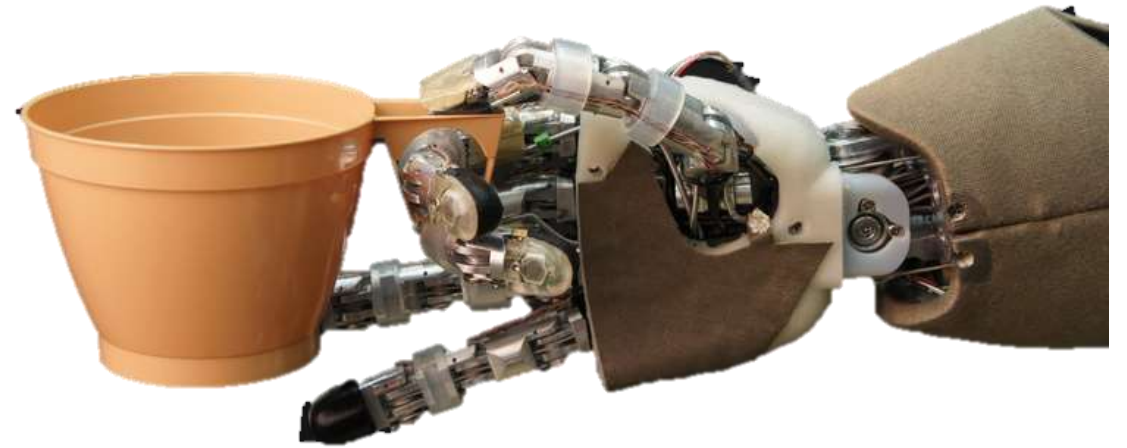
- We have had high degree of freedom robot hands in humanlike form since the 80's. What is missing?
- There have been many exciting new ideas about hand design throughout the past decades.
- Yet we still do not have highly dexterous robotic hand.
- What are the gaps?
- How can we close them?





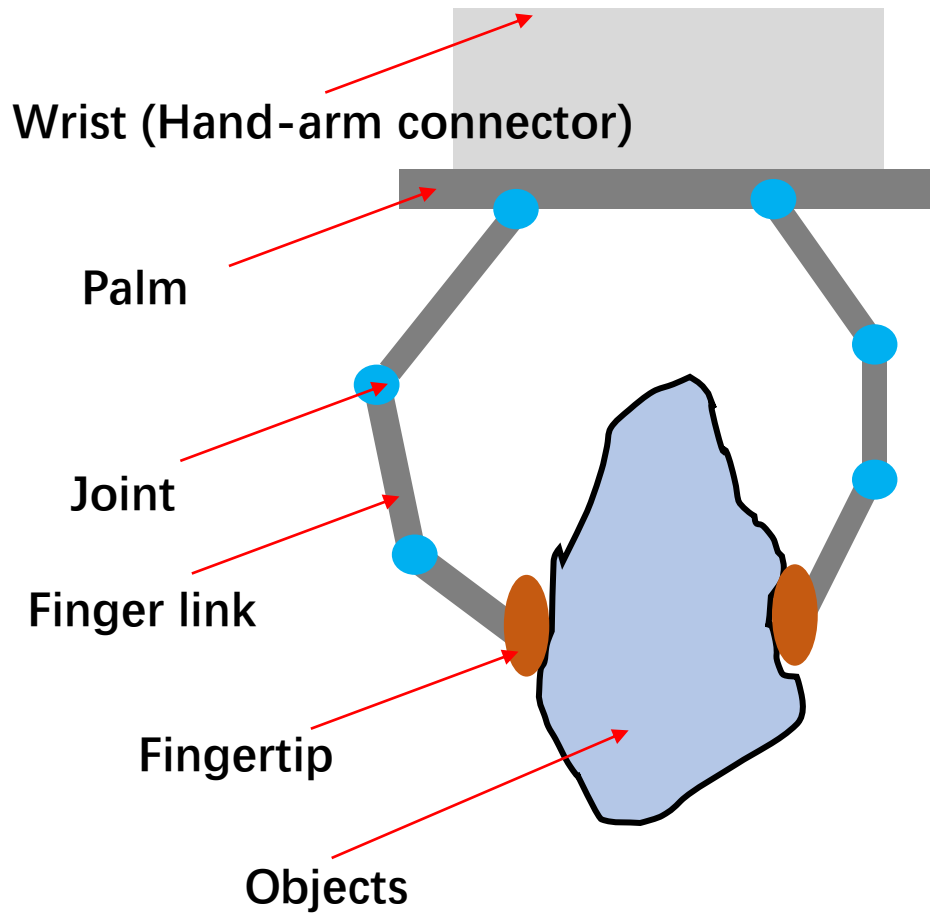
Today

- Design and Modelling of typical robotic arm/hand
- DH parameters
- Kinematics, IK, Jacobian
- **Soft hand**
- Grasp planning
- Simulation tool introduction
- Group list





A robotic grasping system



- **Complex hands = Complicated!**

- **Difficult to control**

- **Expensive**

- **Fragile**

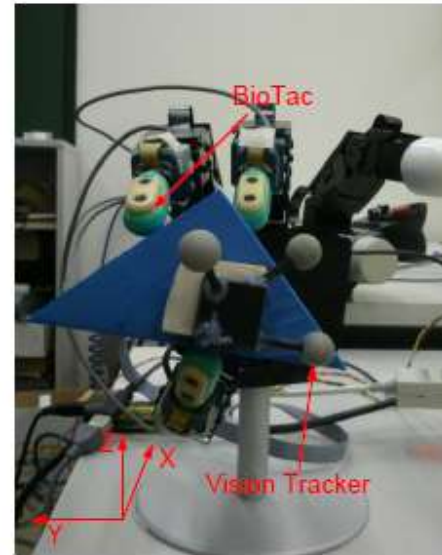
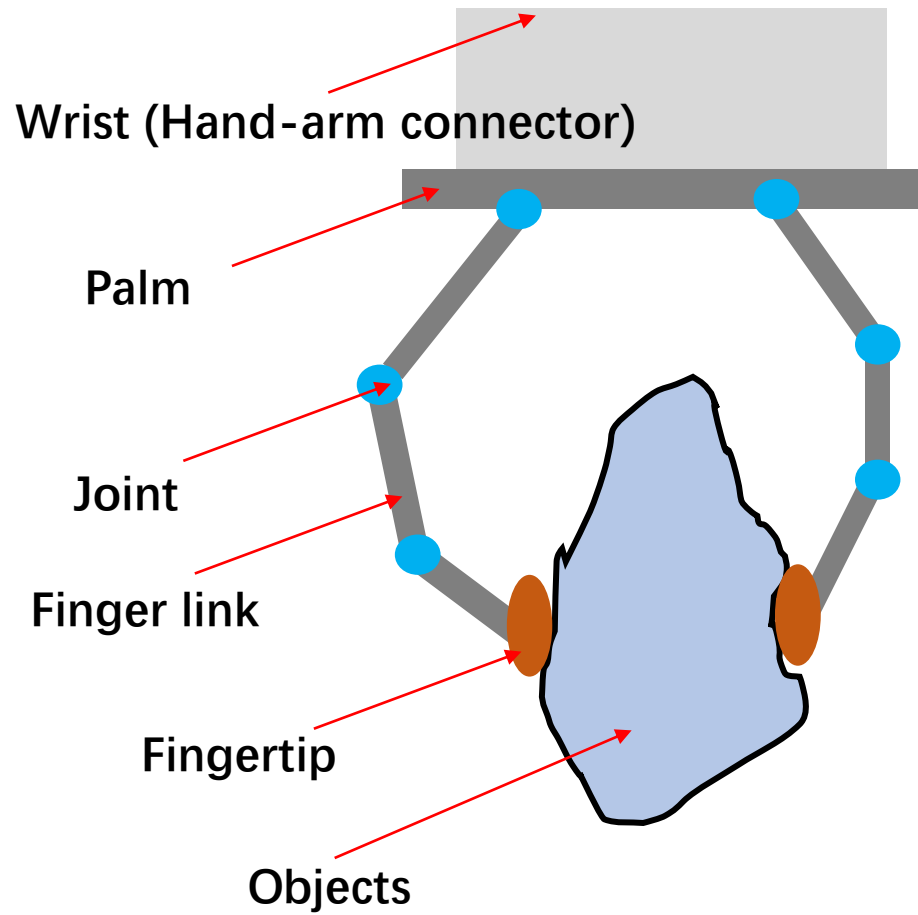


Utah/MIT hand
robonaut.jsc.nasa.gov

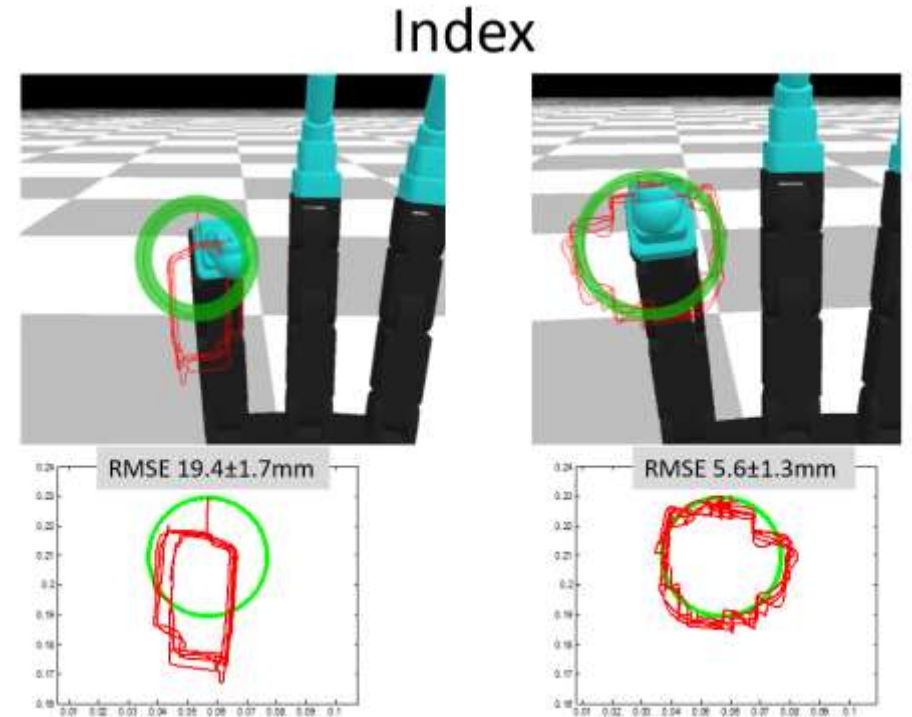
They don't work reliably!



A robotic grasping system



(a)

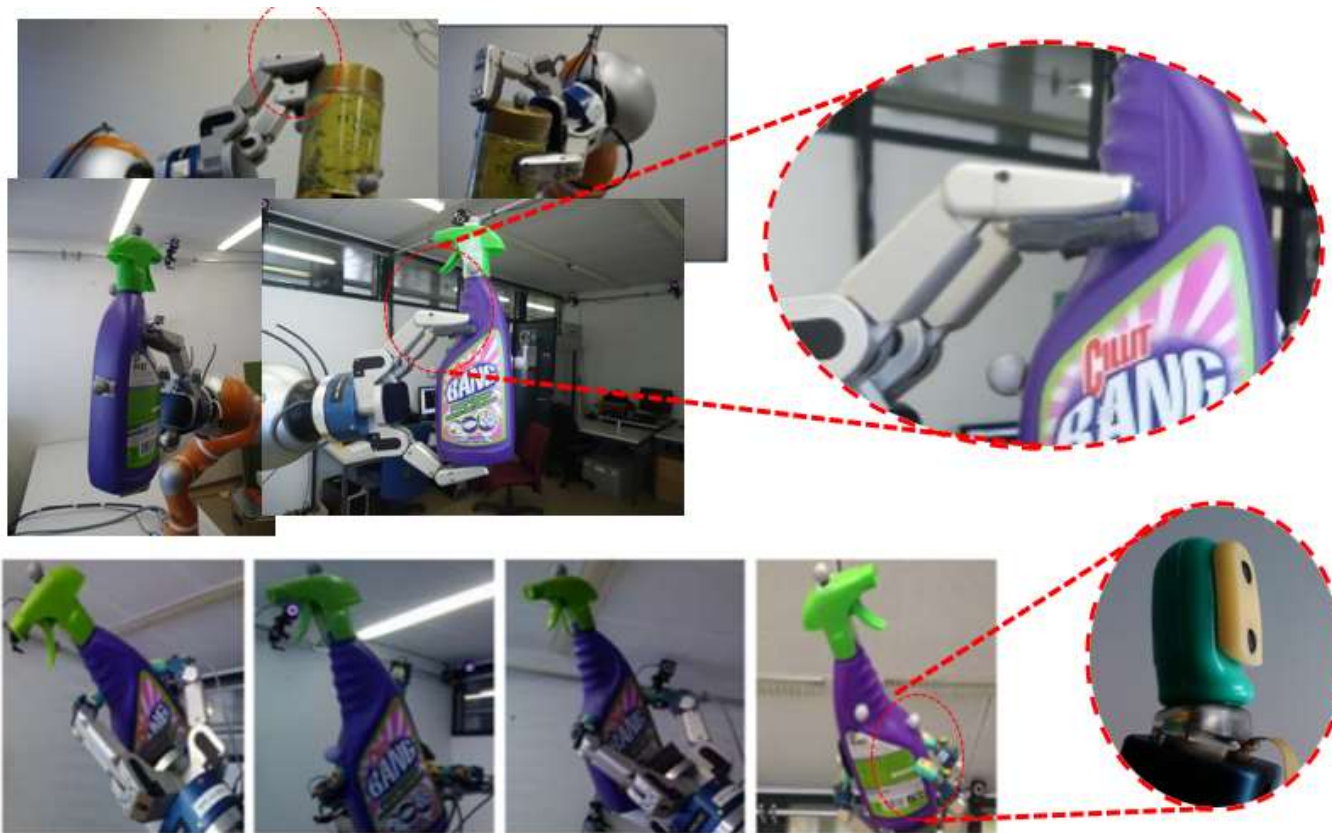
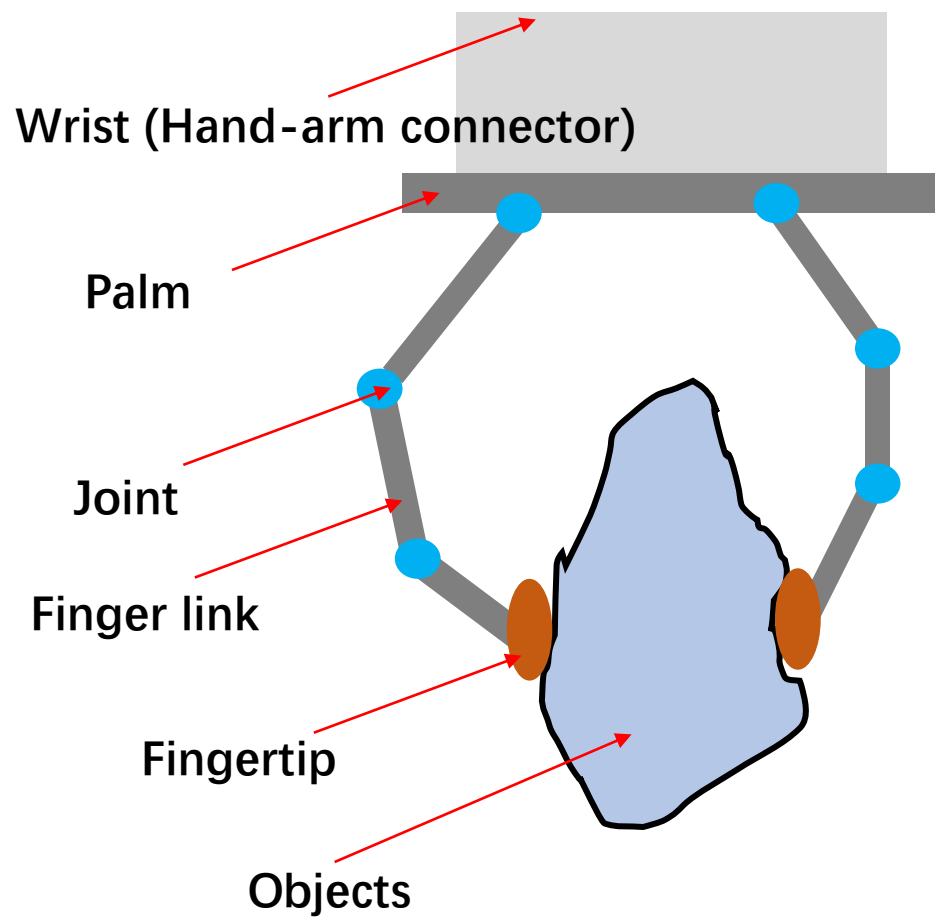


How to deal with “poor” sensing?

- Errors in positioning, finger placement
- Can't control contact forces

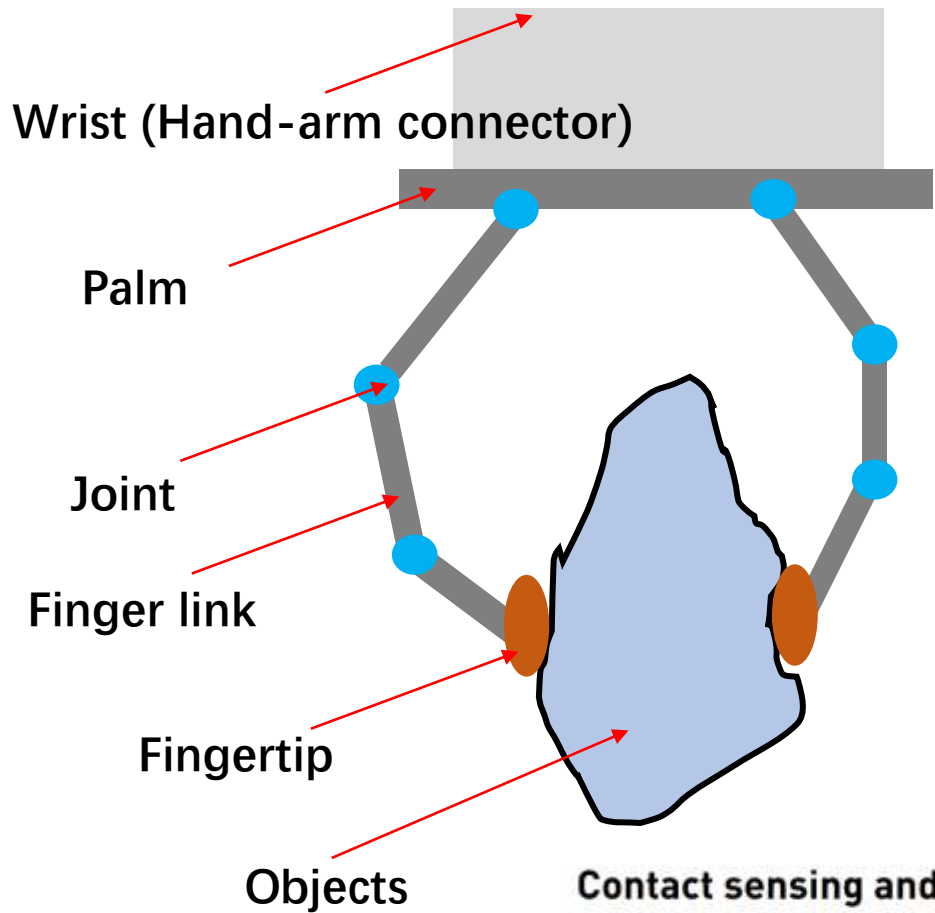


A robotic grasping system





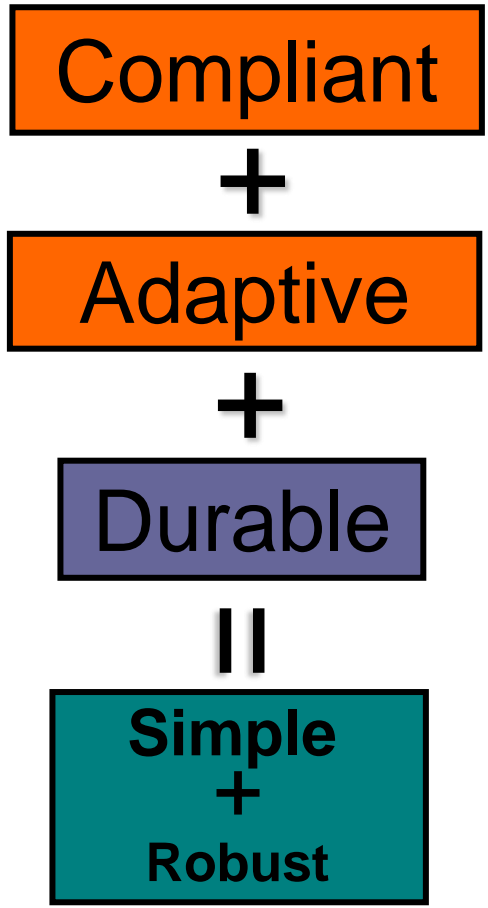
A **new** robotic grasping system



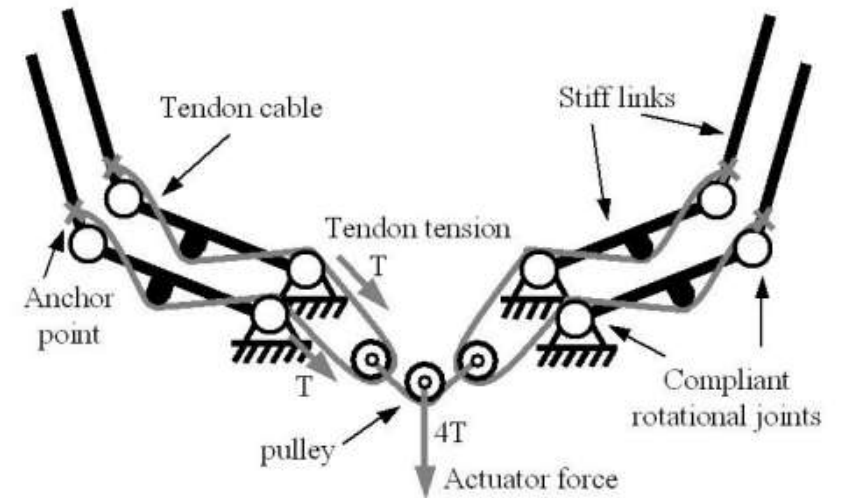
Contact sensing and grasping performance of compliant hands

Citation

Dollar, Aaron M., Leif P. Jentoft, Jason H. Gao, and Robert D. Howe. 2009. "Contact Sensing and Grasping Performance of Compliant Hands." *Auton Robot* 28 (1) [August 26]: 65-75. doi:10.1007/s10514-009-9144-9.

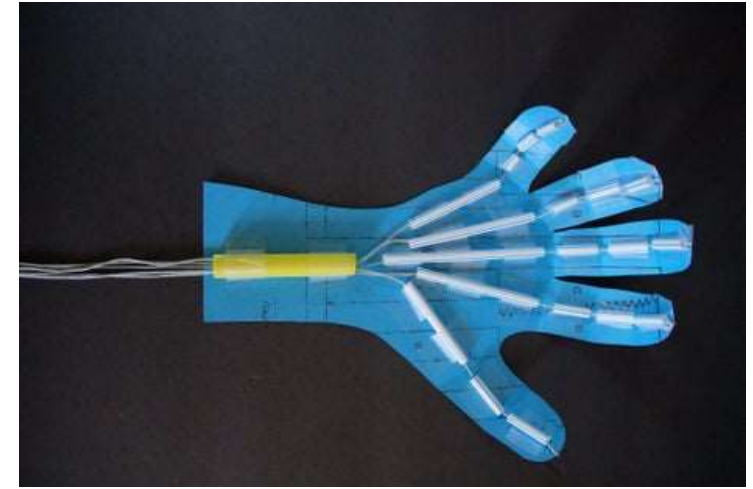


- 3D printing
- Soft hand technology
- Anyone can make a hand?





A toy example





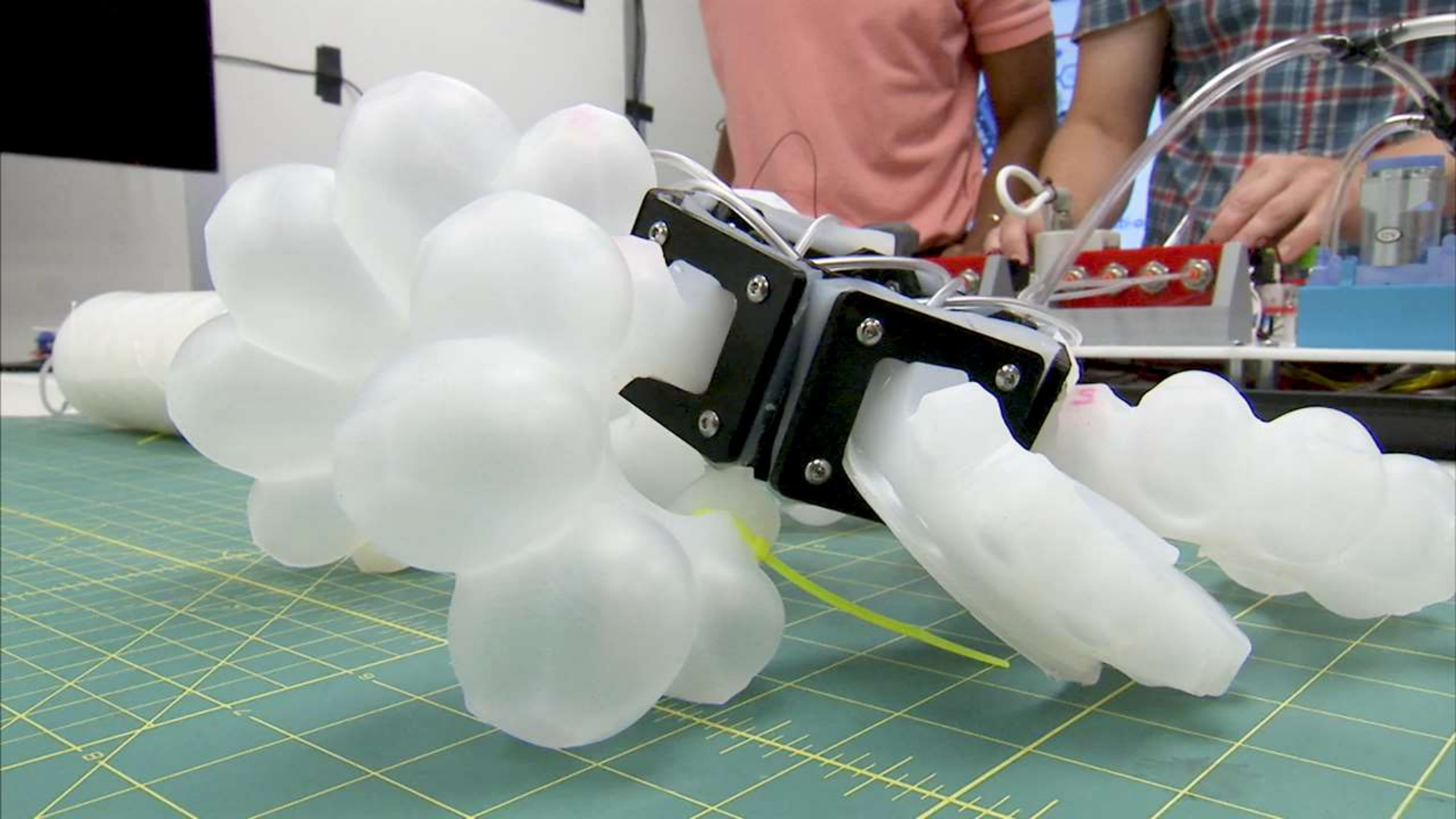
Video
(2 mins)

Video
(7 mins)

Soft Robotics

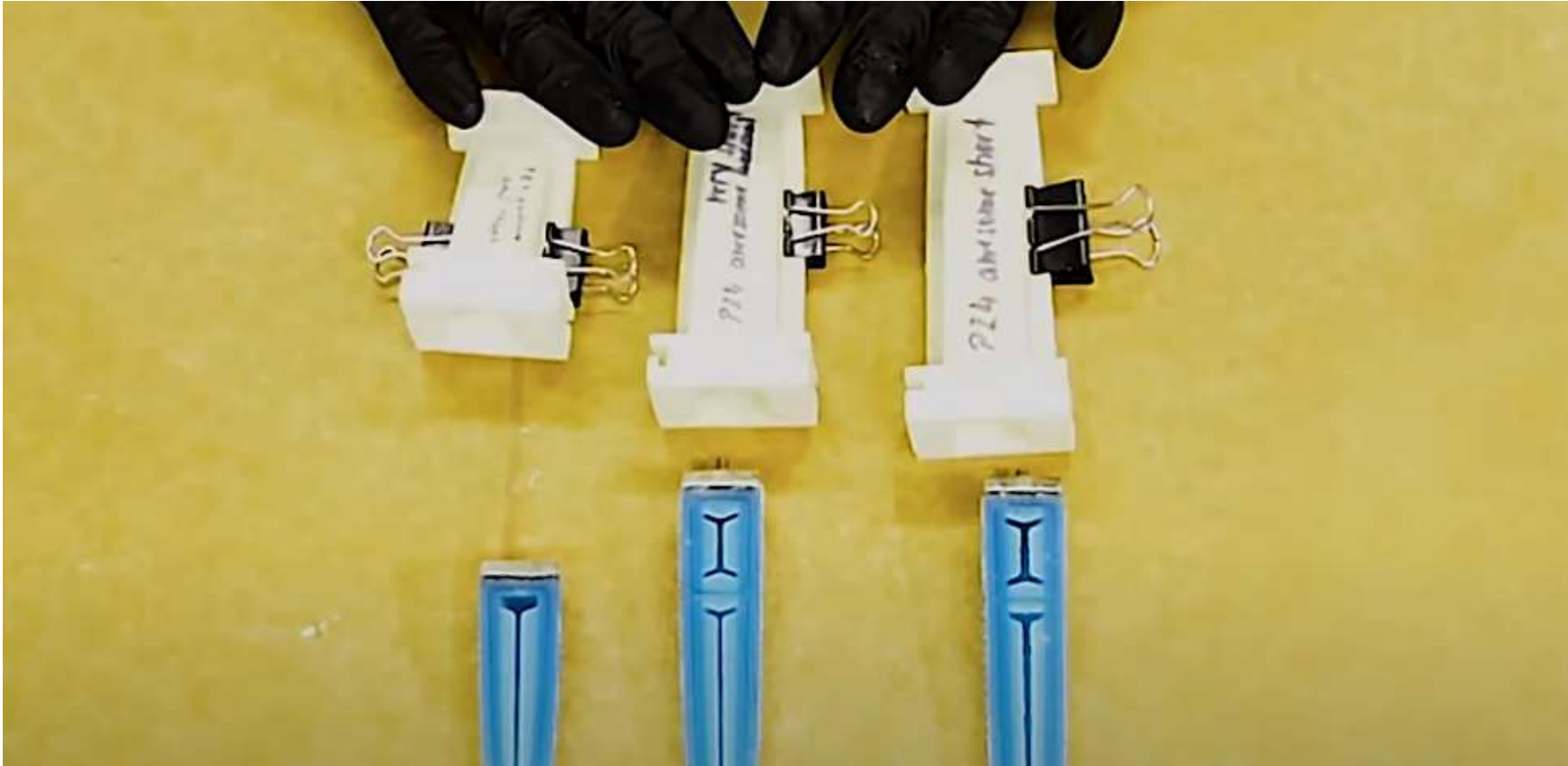
Soft robotic gripper







Soft Hands

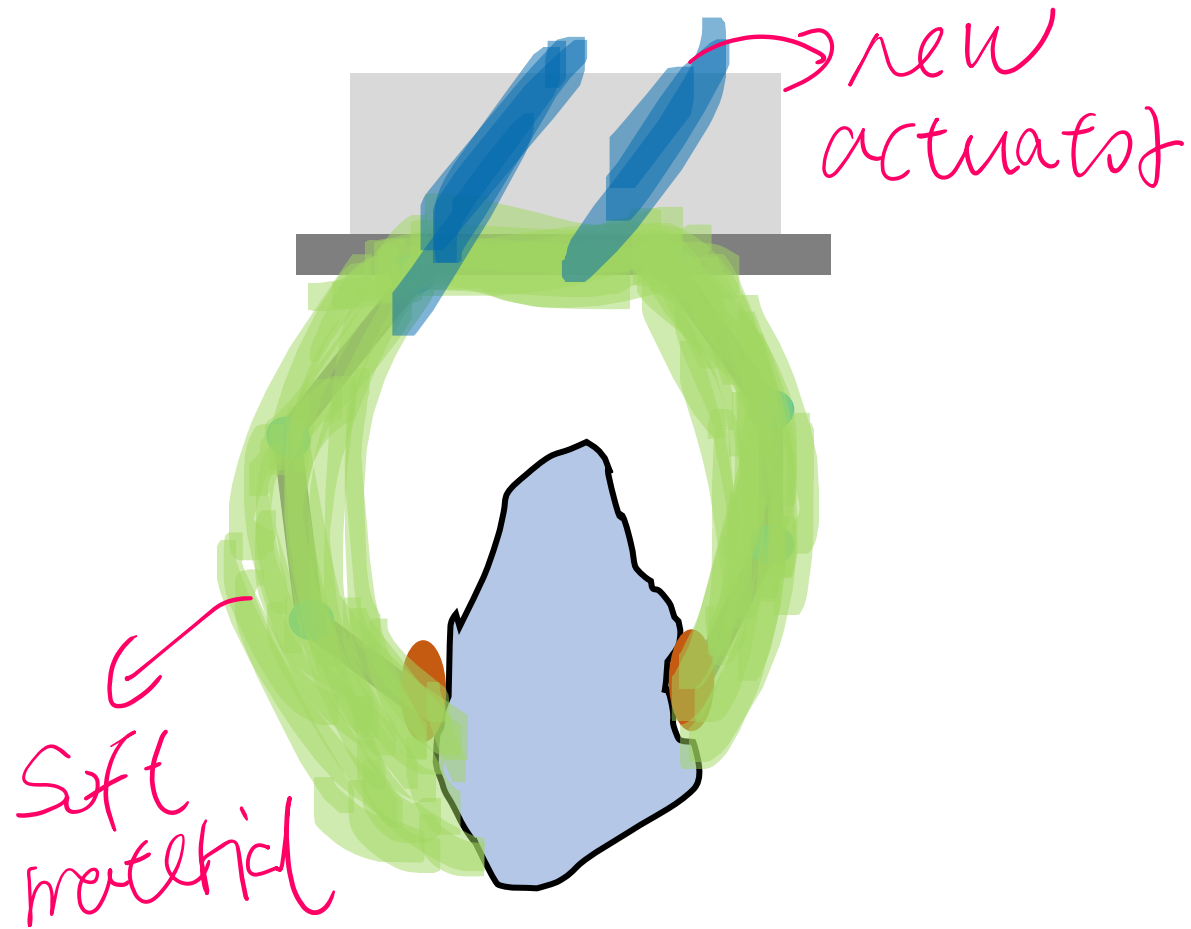
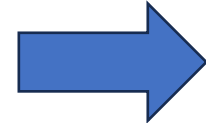


How to build Soft Robotics Fingers, Pulpes and Gloves

https://www.youtube.com/watch?v=HE4MGYLkXjk&ab_channel=RBOTUBerlin



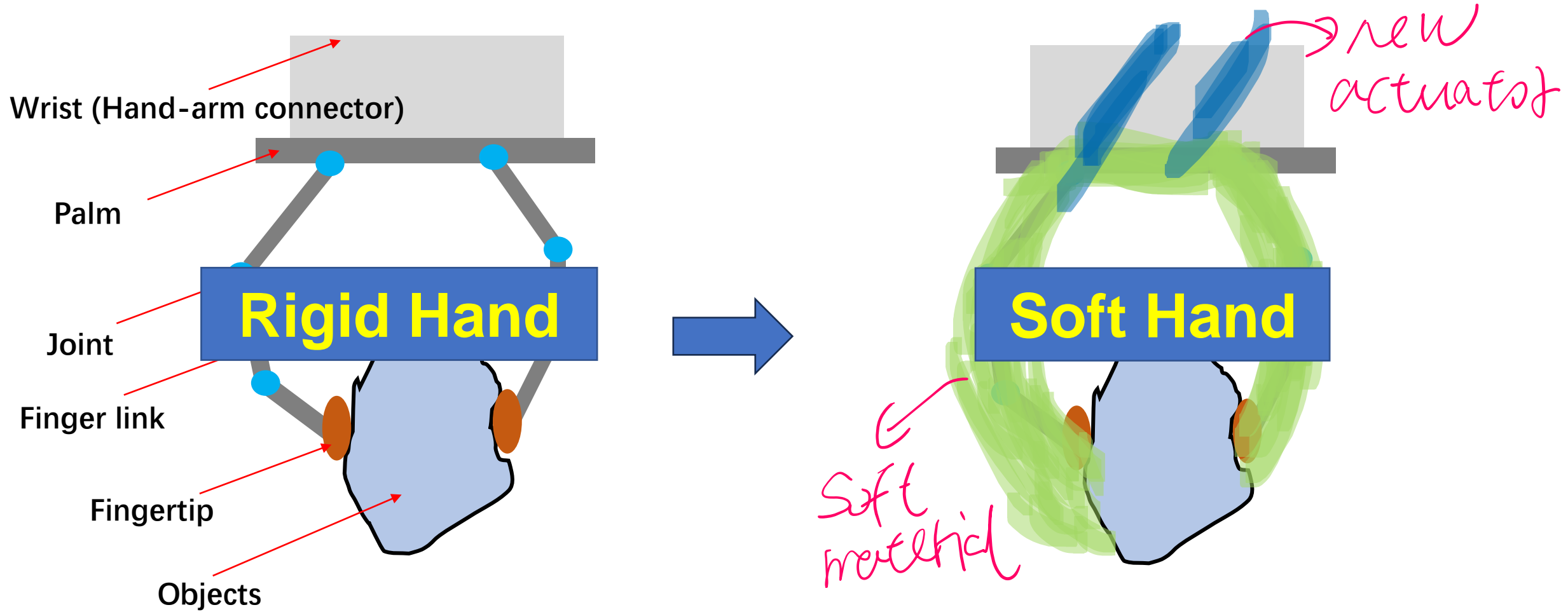
A robotic grasping system



Deformation












A robotic grasping system



How to design and control the deformation?



Soft Hands

| | | | | |
|---|---|---|---|---|
| Soft Exo-suit  | Octopus-inspired robot  | Universal gripper  | Fluid-drive origami-inspired artificial muscles  | |
|  |  |  |  |  |
| X-RHex | Soft griper | Origami robot | Rehabilitation glove | Octobot |
| Mostly stiff Few selective compliant elements | | | | Entirely soft |
| <i>Soft robots with different degree of stiffness</i> | | | | |

How to design and control the deformation?



Soft Hands

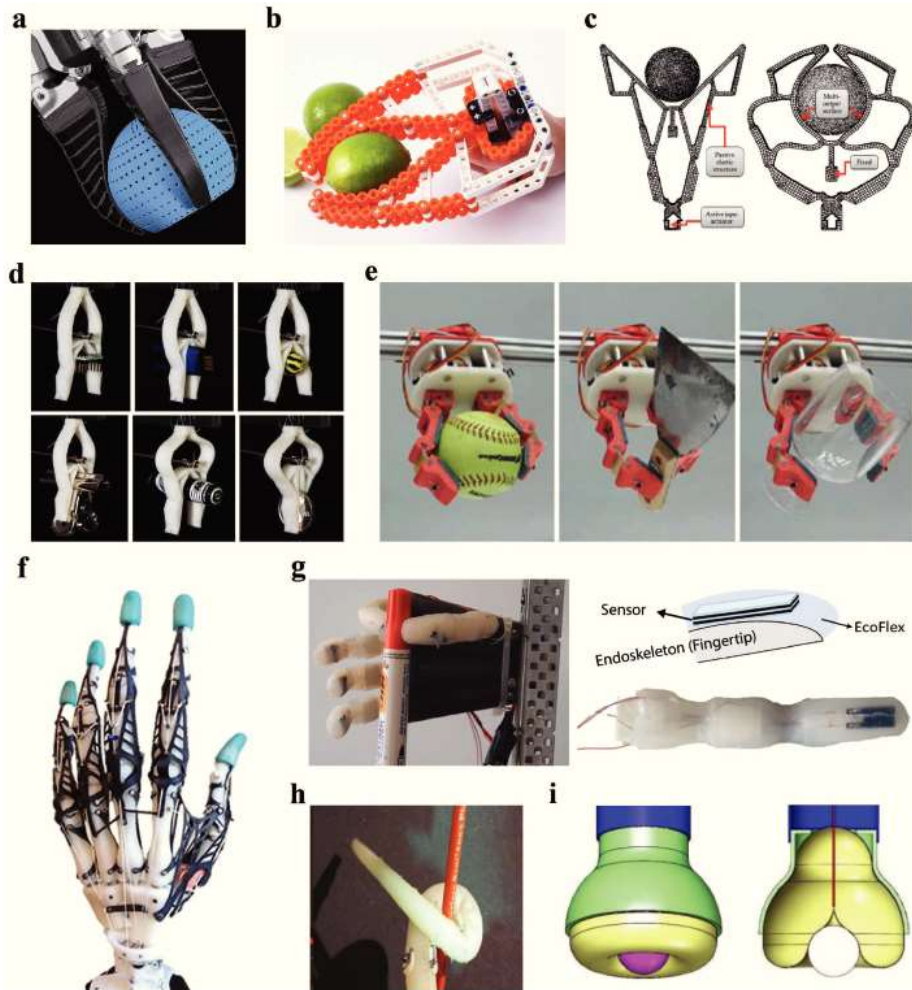
REVIEW

Soft Grippers

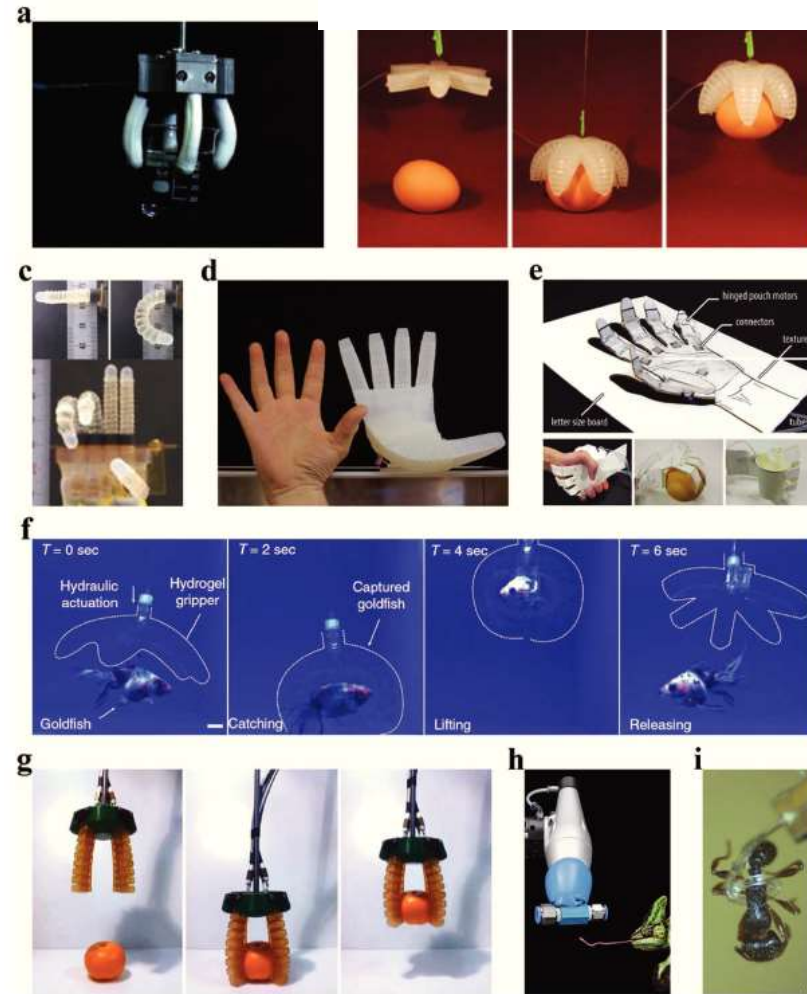
ADVANCED MATERIALS
www.ademat.de

Soft Robotic Grippers

Jun Shintake, Vito Cacucciolo, Dario Floreano, and Herbert Shea*



Soft grippers using passive structure with external motors



Soft grippers using fluidic elastomer actuators



Soft Hands

REVIEW

Soft Grippers

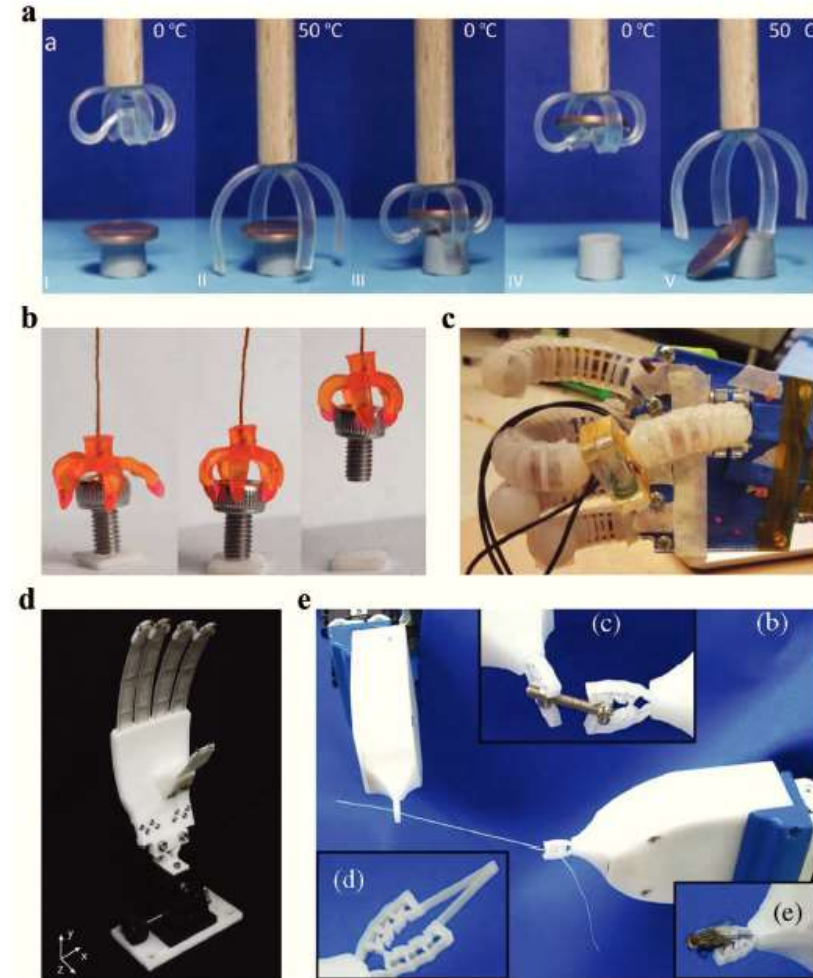
ADVANCED MATERIALS
www.ademat.de

Soft Robotic Grippers

*Jun Shintake, Vito Cacucciolo, Dario Floreano, and Herbert Shea**



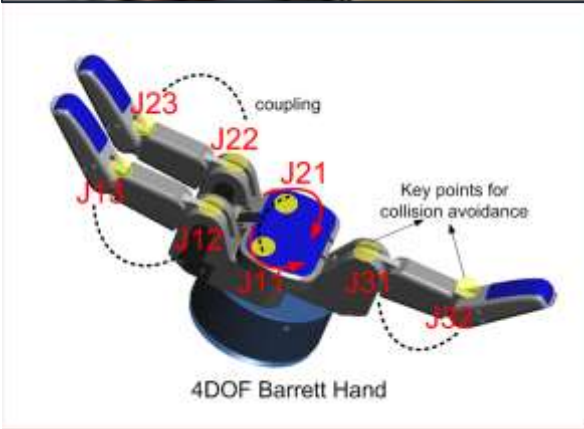
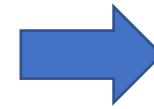
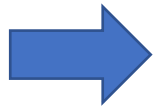
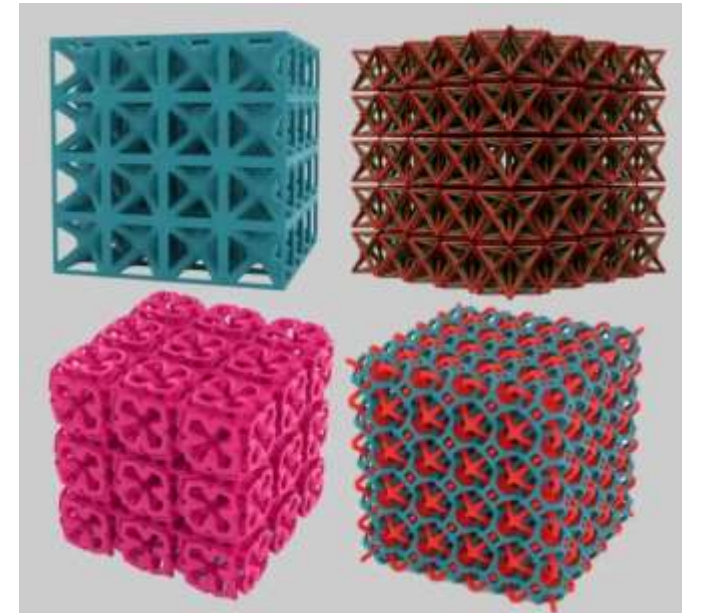
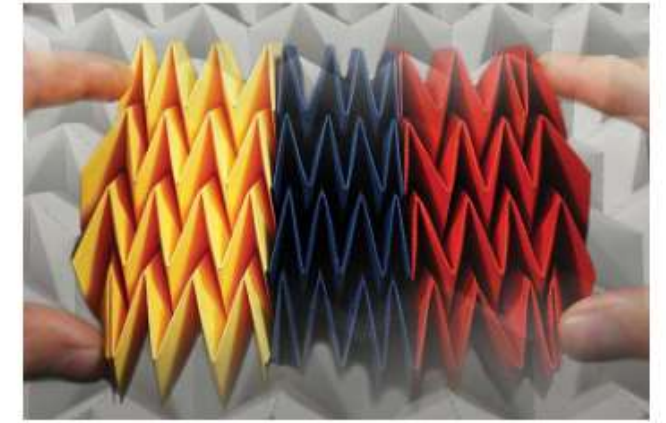
Soft grippers using dielectric elastomer actuators



Soft grippers using shape memory materials



New Material



Steel → Rubber... → Meta Material

Video
(3 mins)

Grasping without Squeezing: Shear Adhesion Gripper with Fibrillar Thin Film

E.W. Hawkes, D.L. Christensen, A.K. Han, H. Jiang, and M.R. Cutkosky
Stanford University



 Biomimetic
Dexterous
Manipulation
Laboratory

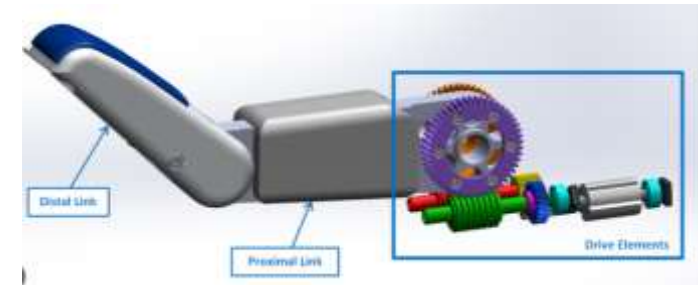
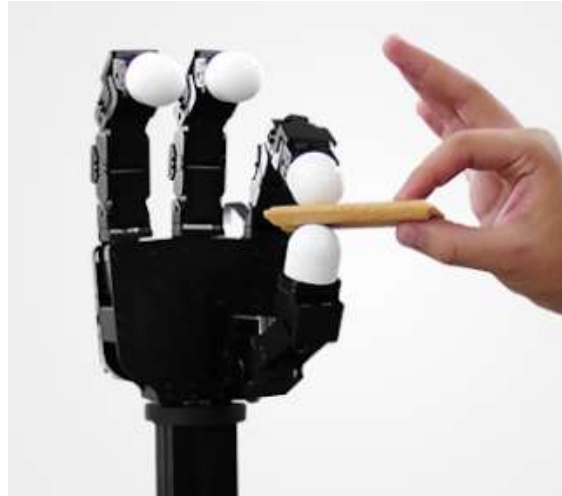
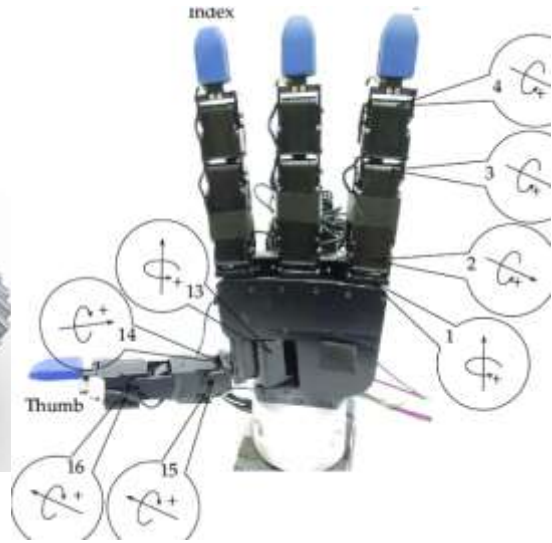
Video
(1.5 mins)

Design, Fabrication, and Evaluation of Tendon-Driven Foam Manipulators

Jonathan P. King, Dominik Bauer, Cornelia Schlagenhauf,
Kai-Hung Chang, Daniele Moro, Nancy Pollard, and Stelian Coros

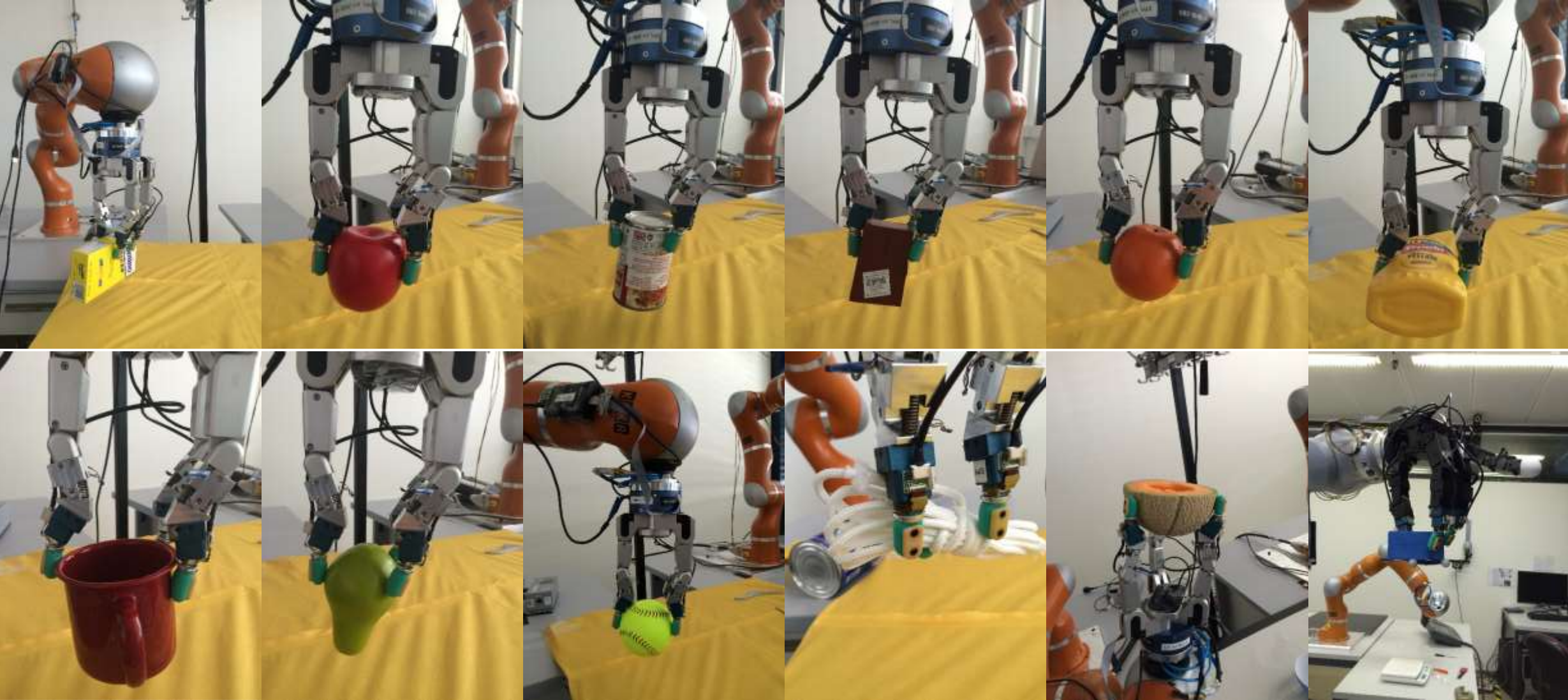


Motion Transmission

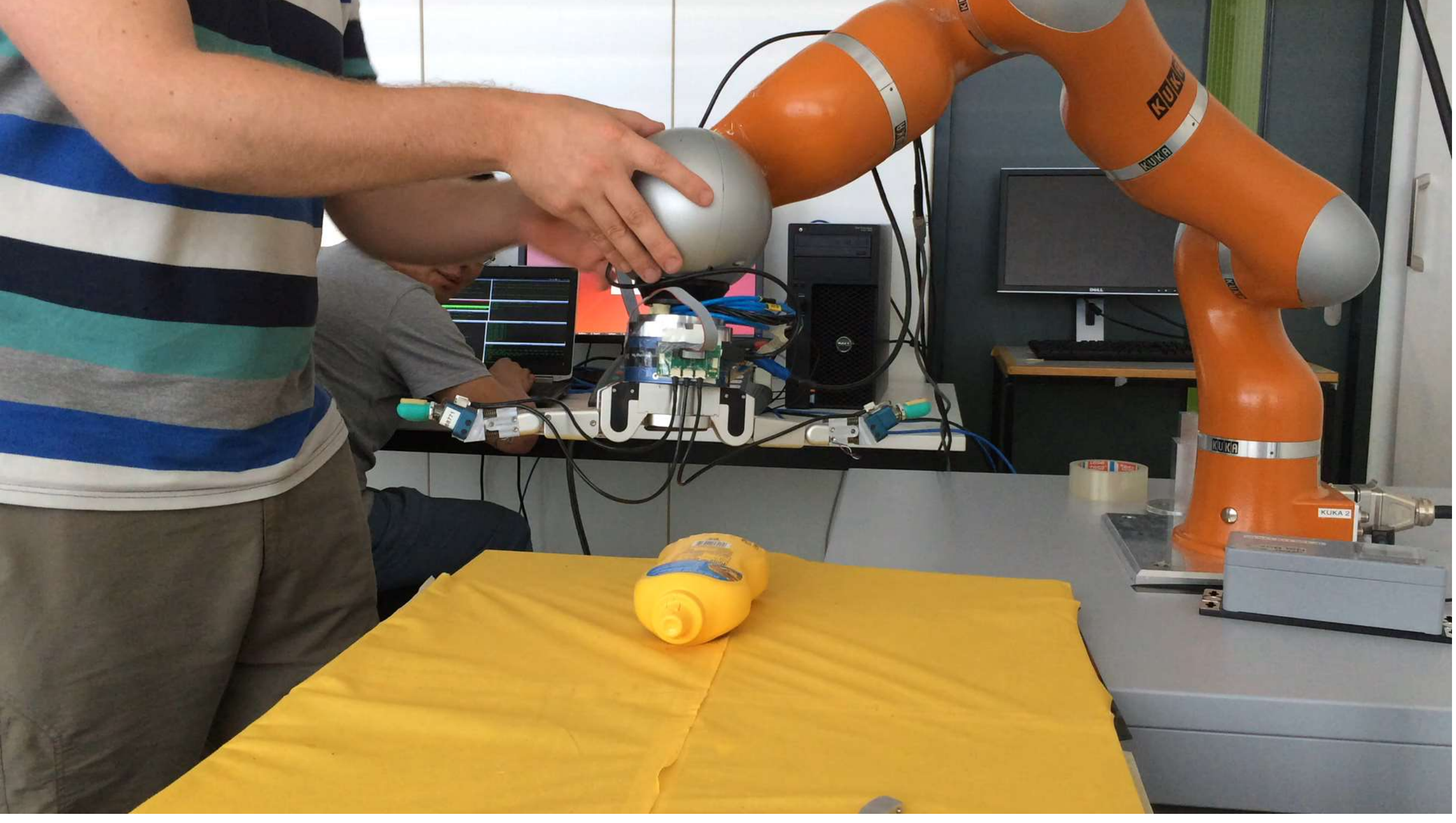


No(few) Deformation!

Back-drivable is already good enough for rigid hand!

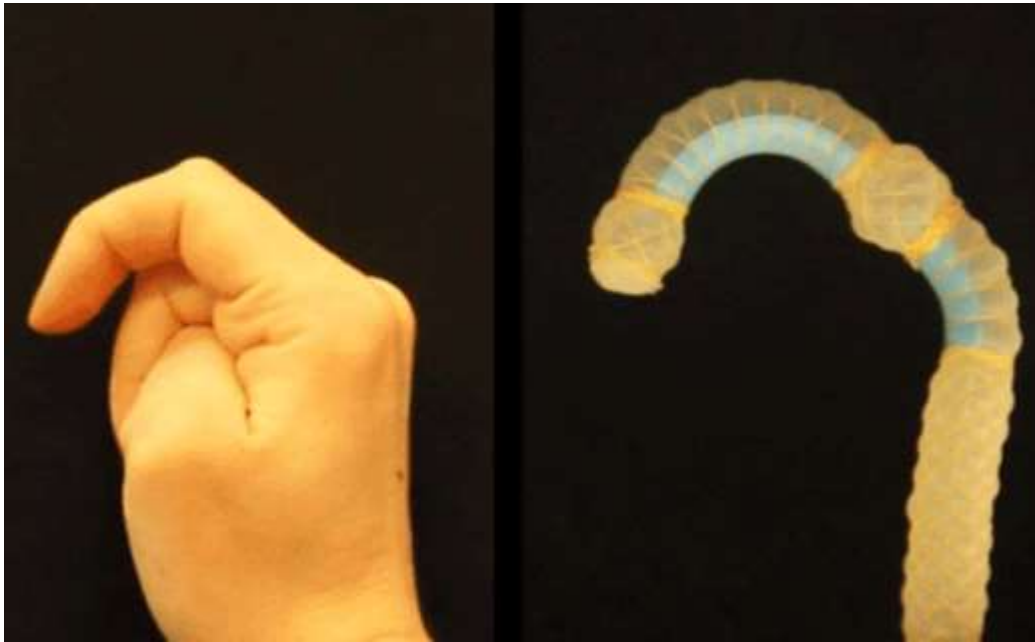


Add some deformation to right hand!





Motion Transmission



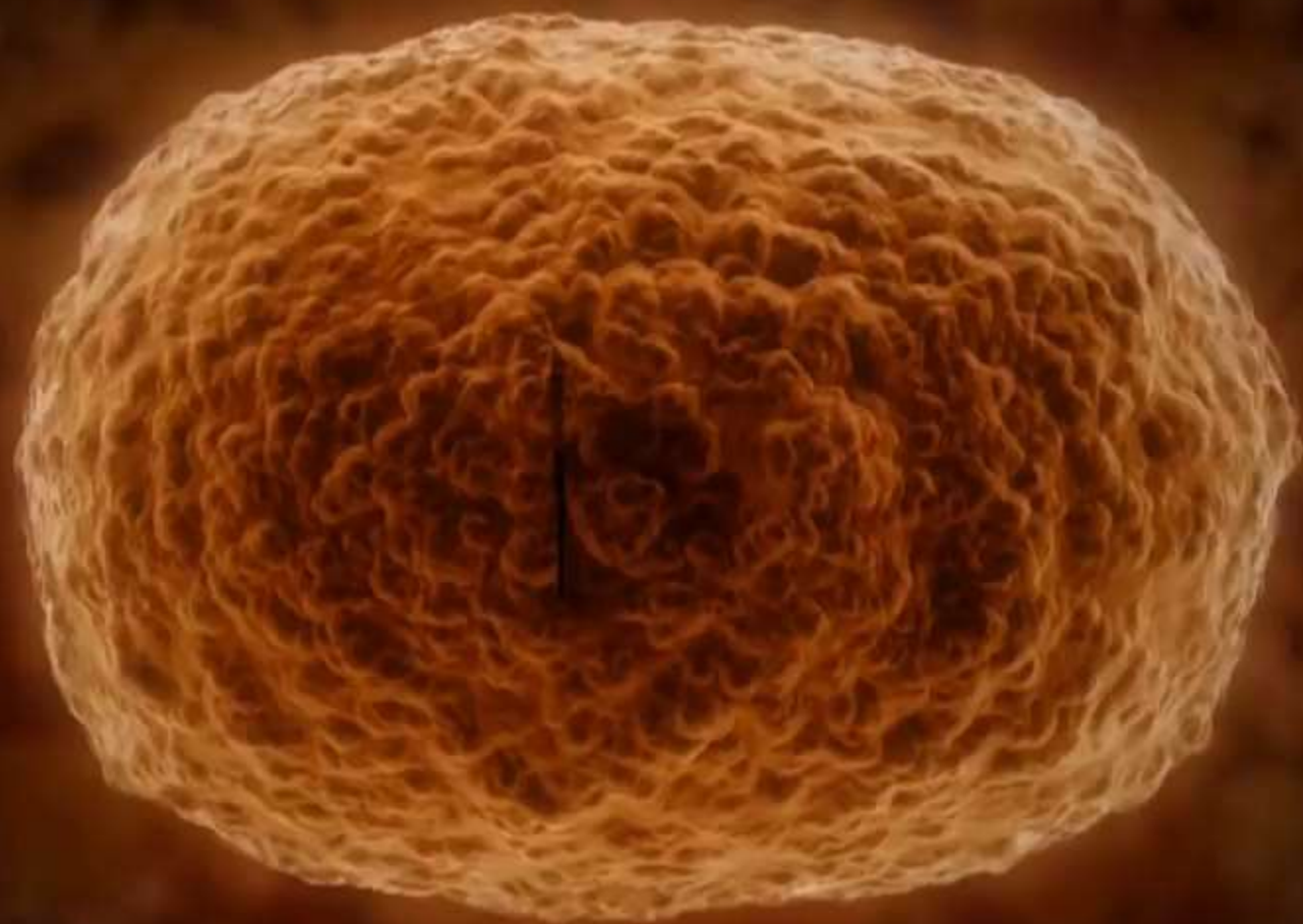
Deformation is motion!
Shape is function!

Video
(1 mins)

HOW DOES A ROBOT PICK UP A RASPBERRY WITHOUT SQUISHING IT?



Video
(2 mins)



Video
(1 mins)

**But this gripper
relies on simple
inflation to
entangle objects
and doesn't
require any
sensing, planning,
or feedback
control.**



**Video
(1 mins)**





Actuator

- electric motors;
- pneumatic actuators;
- hydraulic actuators;
- shape memory alloys (SMA).

| Proprieties class | Power density ρ [W/Kg] | σ_{max} [MPa] | ϵ_{max} | E [GPa] | Efficiency |
|-------------------|-----------------------------|----------------------|------------------|----------------------|------------|
| DC motors | 100 | 0.1 | 0.4 | * | 0.6–0.8 |
| Pneumatic | 400 | 0.5–0.9 | 1 | $5-9 \times 10^{-4}$ | 0.4–0.5 |
| Hydraulic | 2,000 | 20–70 | 1 | 2–3 | 0.9–0.98 |
| SMA | 1,000 | 100–700 | 0.07 | 30–90 | 0.01–0.02 |
| Human muscle | 500 | 0.1–0.4 | 0.3–0.7 | 0.005–0.09 | 0.2–0.25 |

FIGURE 2.11: Actuator Performance Indices: Power density ρ = Power per unit of weight, σ_{max} = Maximum force exerted by the actuators per area, ϵ_{max} = Maximum run per length, E Actuator stiffness. Maximum stress and strain are indexes specifically designed for linear actuators. Units are expressed as follow: W Watt, Kg kilogram, MPa Mega Pascal, GPa Giga Pascal.

*Depending on the gearhead



Actuator



(a) McKibben actuators



(b) Shadow Hand finger set-up

FIGURE 2.12: Pneumatic actuators for robotic hands

Video
(5 mins)





Actuator

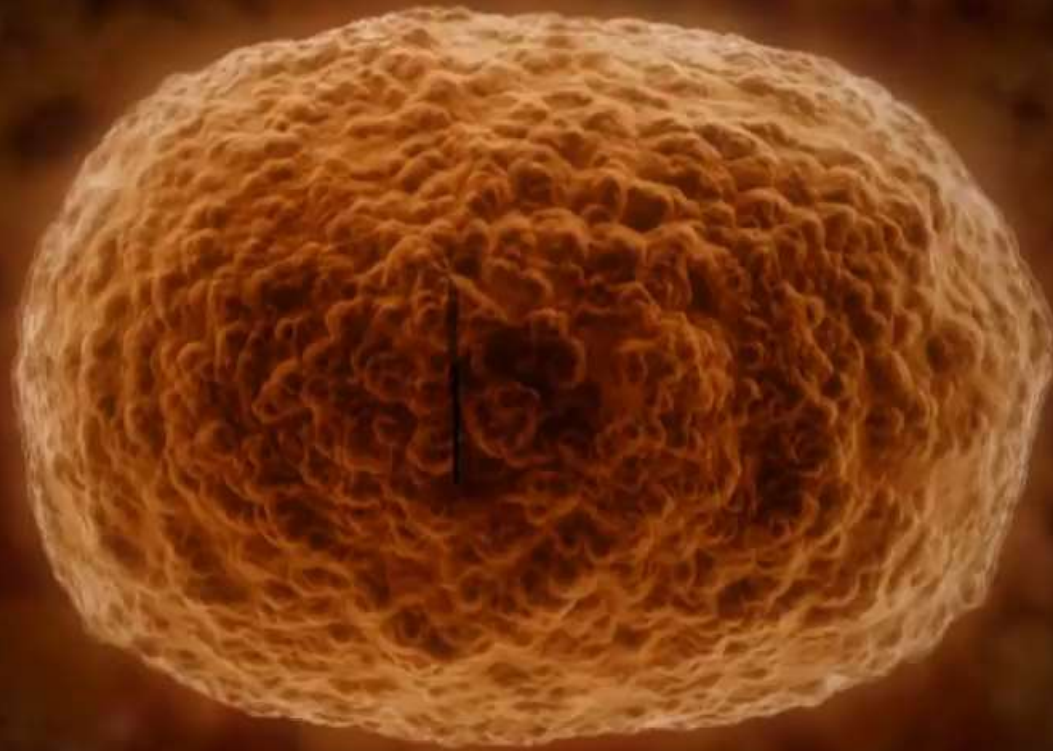
Video
(3 mins)





Actuator

Video
(2 mins)





Sensors

- motor position sensors; ✓

- finger joints position sensors; ✓

- ✓ motor torque sensors;

- ✓ joint torque sensors;

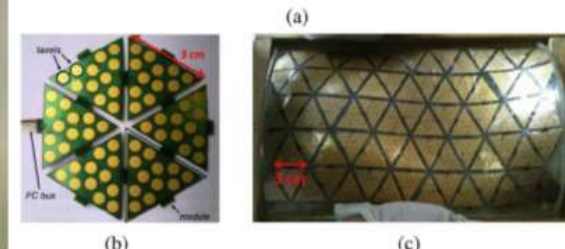
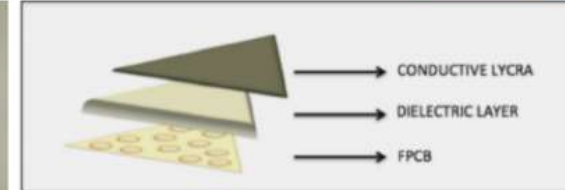
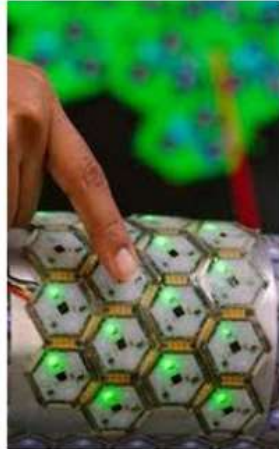
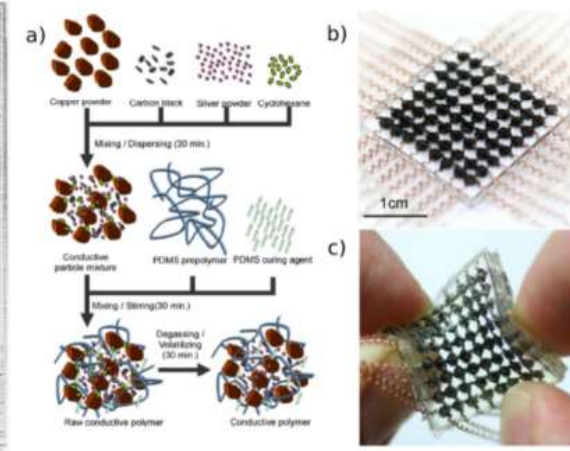
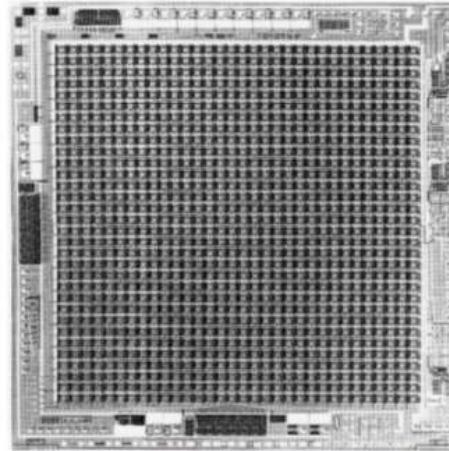
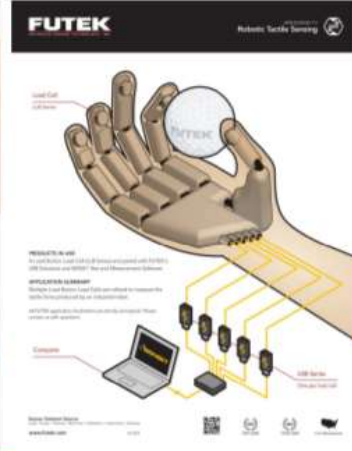
- tactile sensors;

- temperature sensors; (★)

- in hand camera.



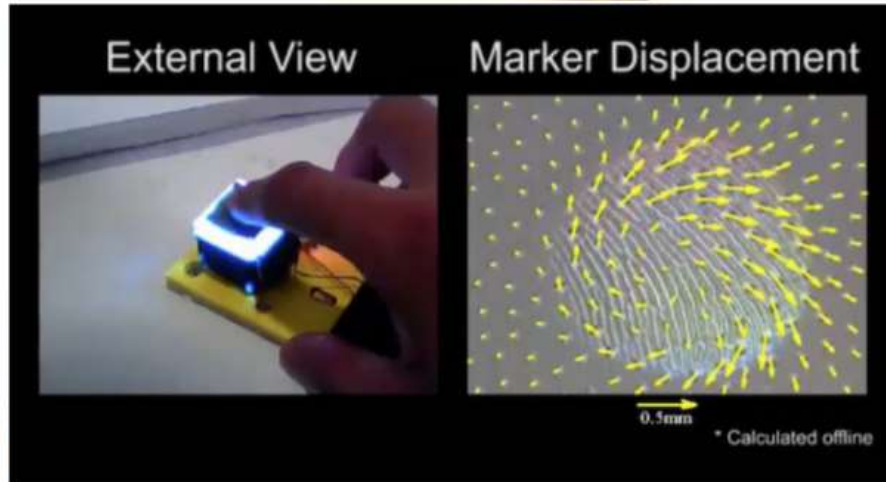
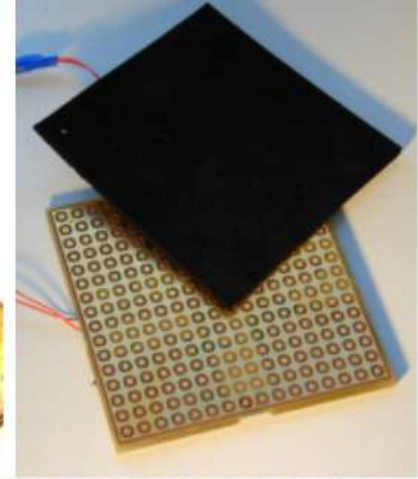
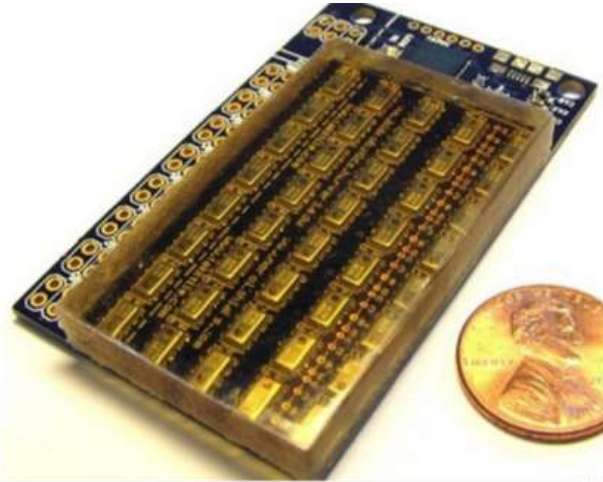
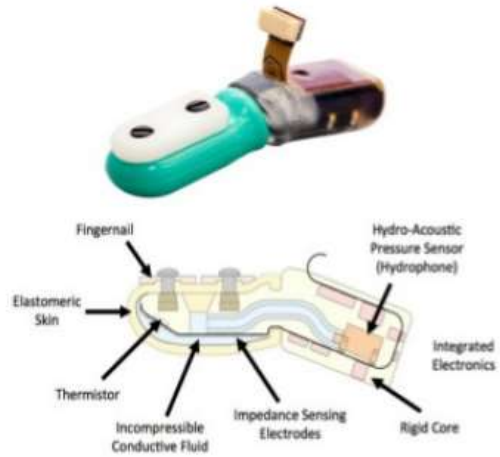
Sensors



Still a hot topic

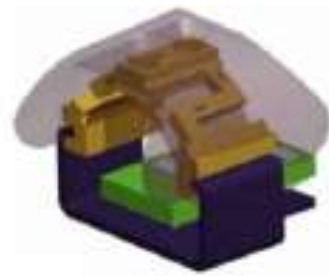


Sensors





Sensors



(a) Force/Torque sensor [35]



(b) Tendon tension sensor [35]



(c) Single axis joint sensor [36]

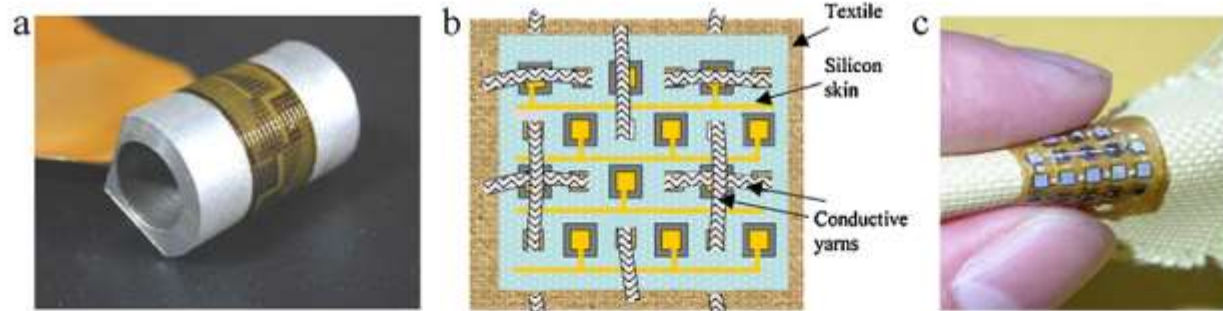


(d) Double axis joint sensor [36]

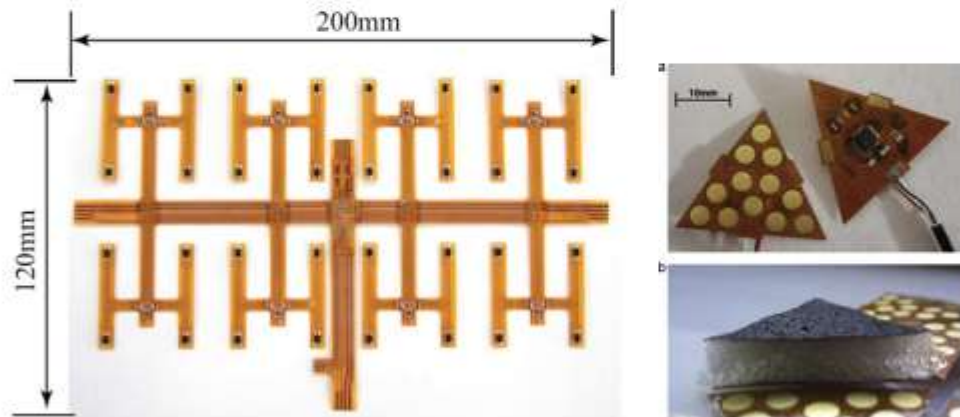
FIGURE 2.15: Force/Torque integrated sensors



Sensors



(a) Strain gauges tactile sensor [37]



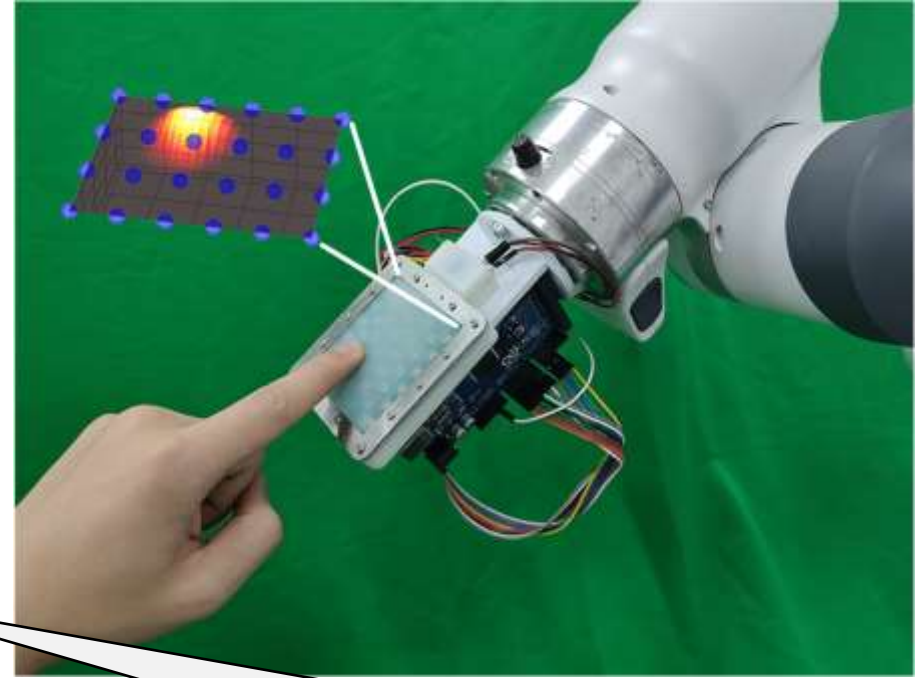
(b) Led based tactile sensor [38]

(c) Capacitive tactile sensor [39]



FIGURE 2.16: Tactile sensors

柔性触觉传感器



In case you need some tactile sensor for your design

A biomimetic tactile palm for robotic object manipulation

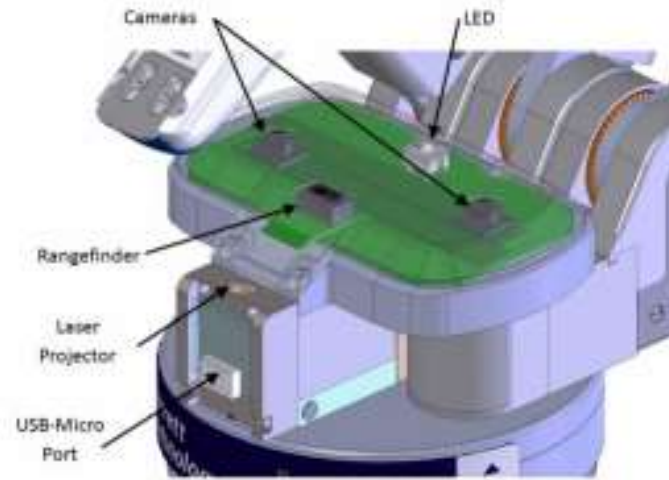
Ziwei Lei^{1,3}, Xutian Deng¹, Yi Wang¹, Xiaohui Xiao², Dong Han³, Fei Chen^{4,*}, Miao Li^{1,2,*}



Sensors



(a) Barrett Hand



(b) In Hand Cam Integration

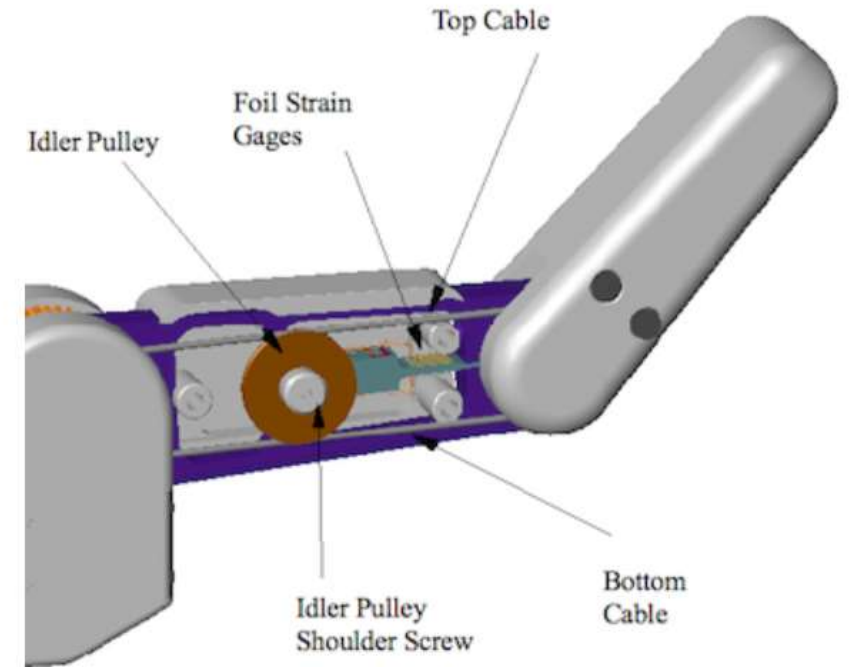
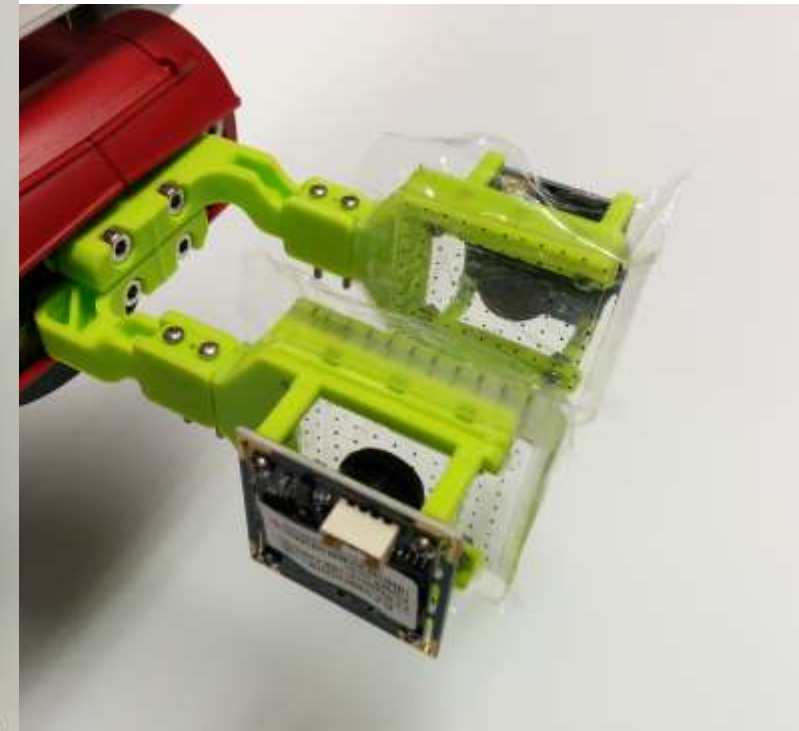
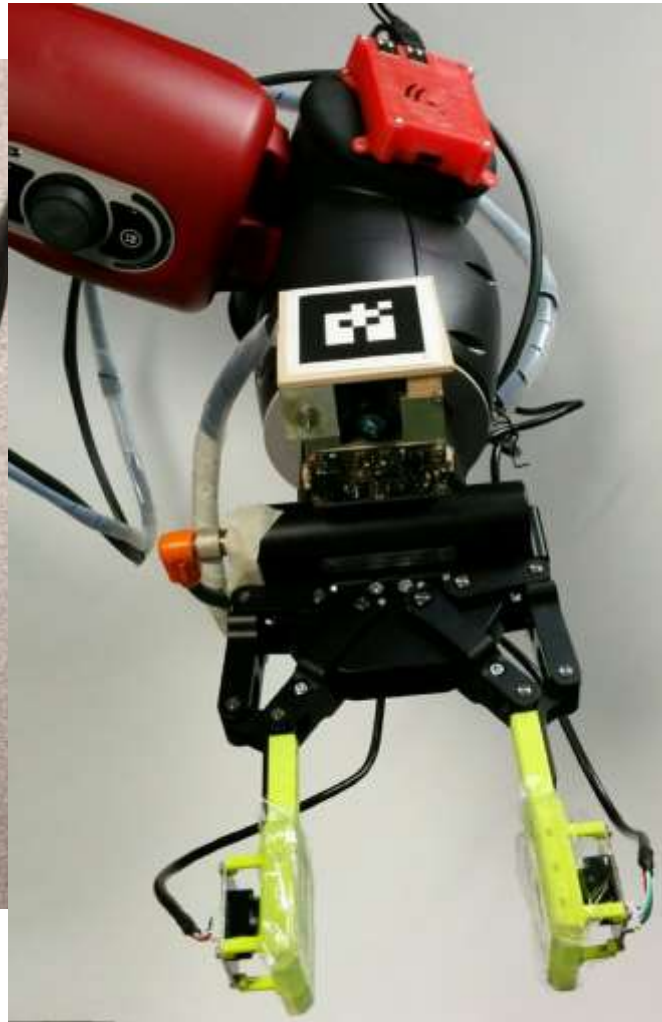


FIGURE 2.17: Patented Hand-Eye integration [32]



Sensors



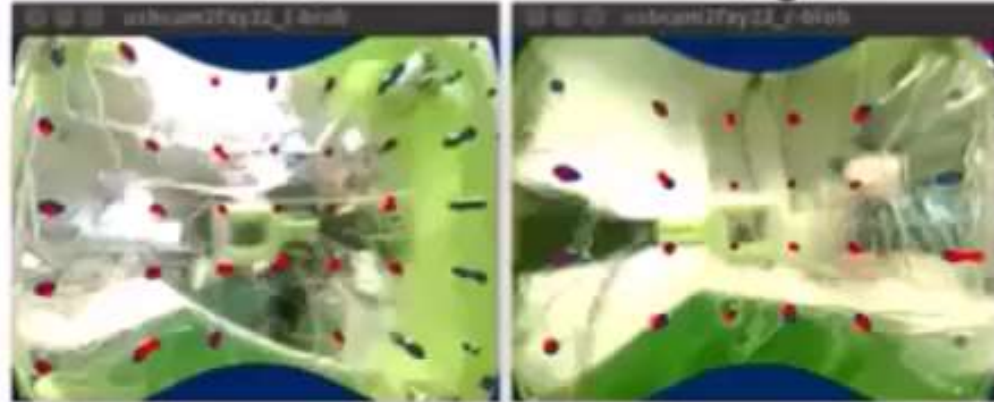


Sensors

**Finger Vision: Tactile Sensing
Feeling With Your Eyes**



View of FingerVision
Left Right



:Force :Slip :Object



Sensors

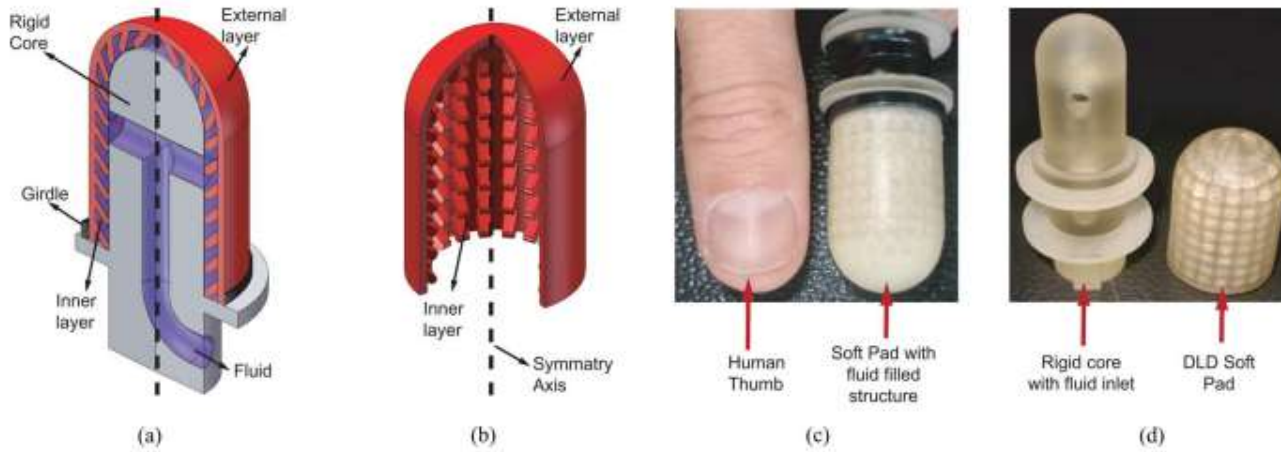


FIGURE 2.18: Fluid-filled soft-pad concept and prototype. (a) 3-D model. (b) Longitudinal cross section. (c) Prototype comparison with human-thumb dimensions. (d) Rigid core with fluid inlet and soft pad.

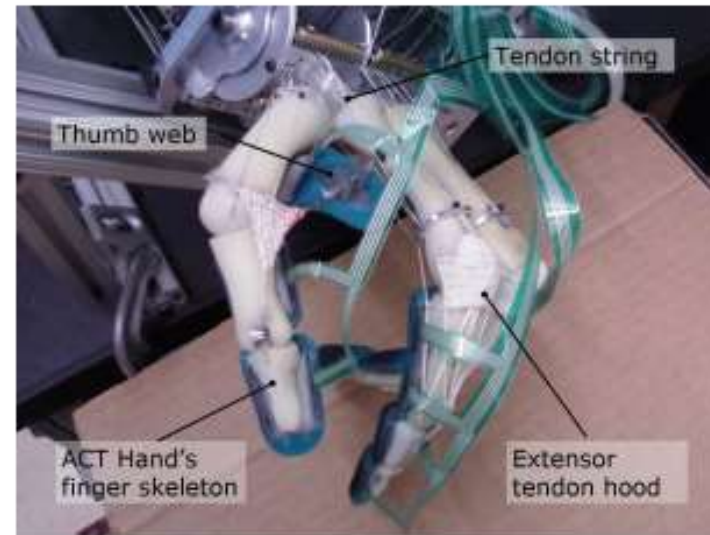




Sensors



(a) Sensorized artificial skin concept

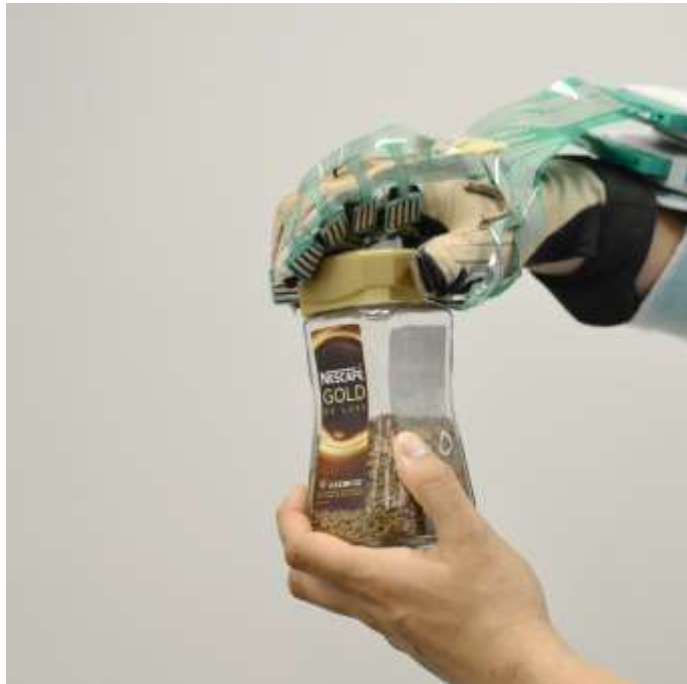


(b) Integration of the skin within the robotic hand

FIGURE 2.19: Artificial skin with integrated tactile sensors



Sensors



Learning Object Exploration from Human Demonstration

Dr. Sahar El Khoury
Ravin Luis De Souza
Miao Li
Prof. Aude Billard

Learning Algorithms and Systems Laboratory
EPFL



Sensors

**Video
(2 mins)**

Dexterous Bimanual Object Exploration with Whole-hand Tactile Sensing

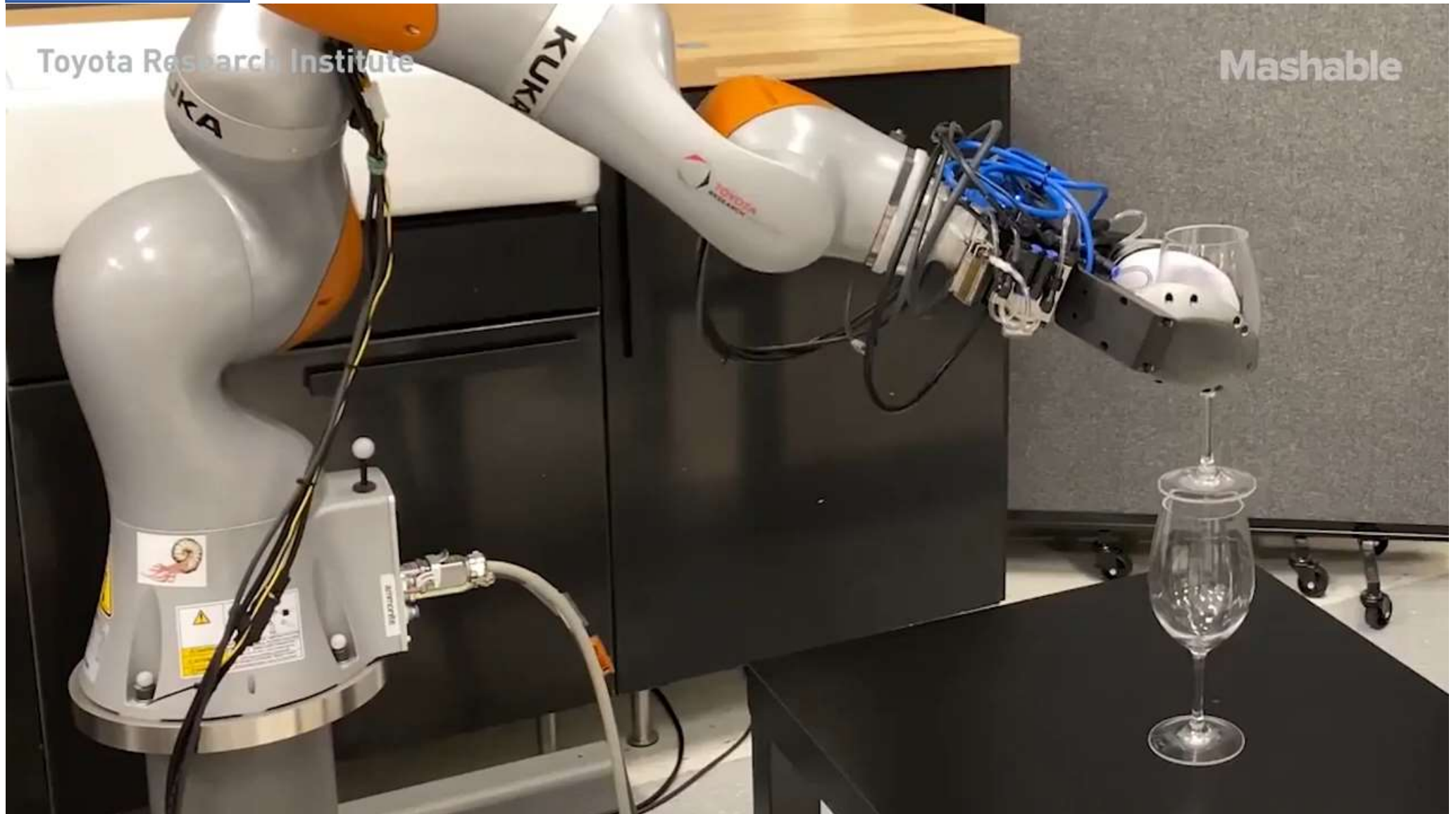
**Nicolas Sommer, Miao Li
Aude Billard**

**Learning Algorithms and Systems Laboratory
EPFL**



Video
(1 mins)

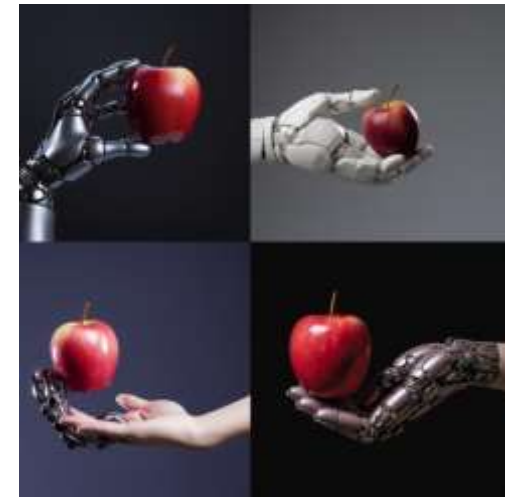
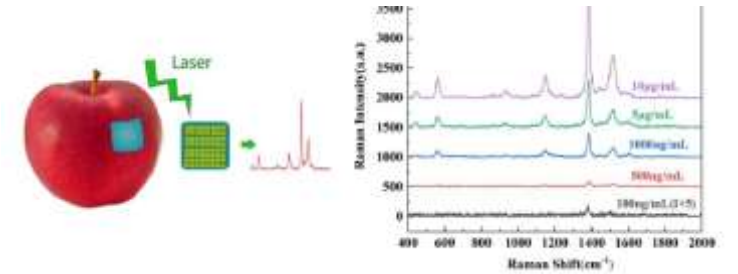
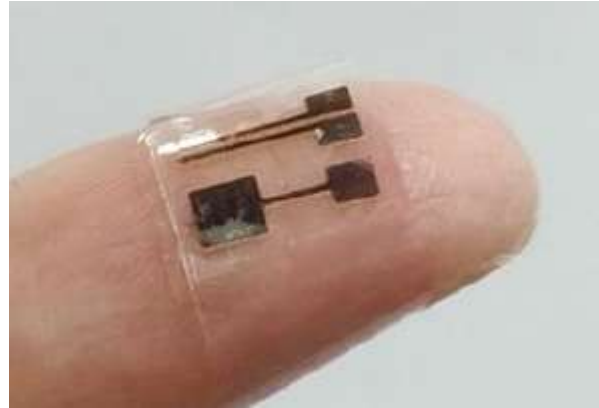
Sensors





Sensors

Attach the device to the palm of your hand



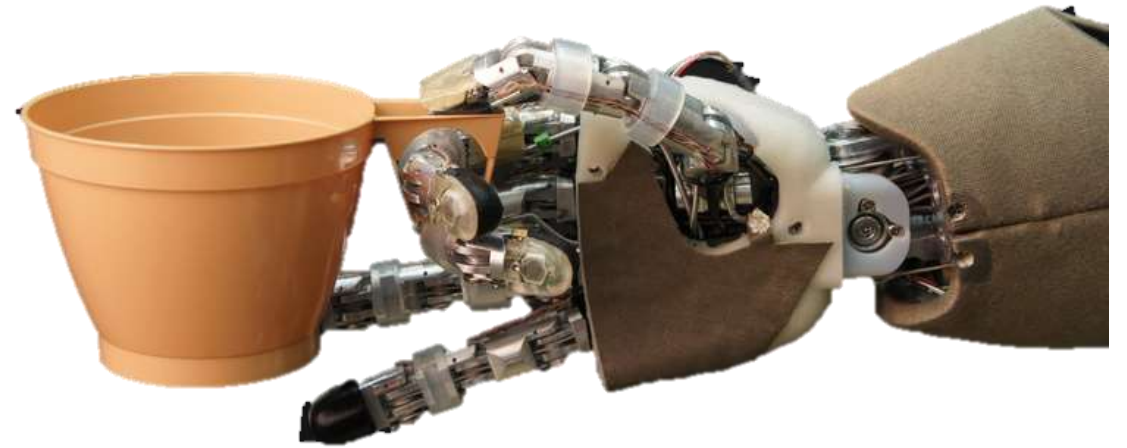
Can we attach this sensor to something else?

Please join us if you are interested in this project!



Today

- Design and Modelling of typical robotic arm/hand
- DH parameters
- Kinematics, IK, Jacobian
- Soft hand
- Grasp planning and control
- Simulation tool introduction
- Group list



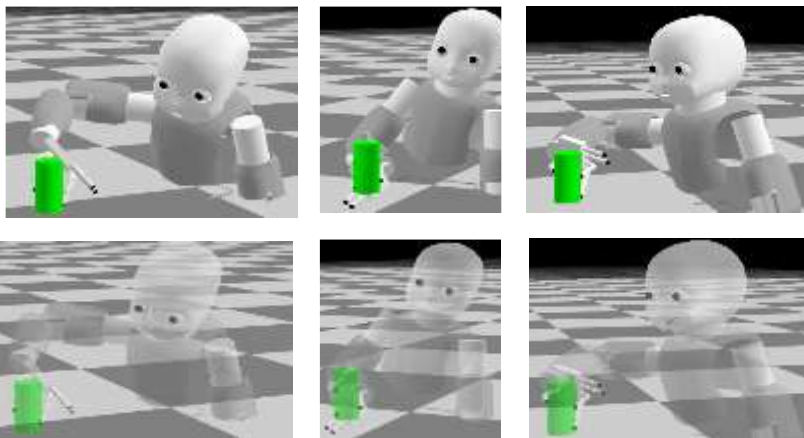


Goal for this course

- Design: soft hand design **x1** ✓
- Perception: vision, point cloud, tactile, force/torque **x1**
- Planning: sampling-based, optimization-based, learning-based **x3**
- Control: feedback, multi-modal **x2** ⇒ DL ↓
tFJ
- Learning: imitation learning, RL **x2**
- Simulation tool (pybullet, matlab, OpenRAVE, Issac Nvidia, Gazebo)
- **How to get a robot moving!**



Grasp planning



Grasp Planning as Optimization

| | | |
|--|---|--------------------|
| $\arg \min_{\theta, \mathbf{H}, \mathbf{p}, \mathbf{n}}$ | $f(\theta, \mathbf{H}, \mathbf{p}, \mathbf{n})$ | objective function |
| subject to: | $h_i(\theta_i, \mathbf{H}) - \mathbf{p}_i = 0$ | hand constraints |
| | $l_i(\theta_i, \mathbf{H}) - \mathbf{n}_i = 0$ | |
| | $g(\mathbf{p}_i) = 0$ | object constraints |
| | $\nabla g(\mathbf{p}_i) \times \mathbf{n}_i = 0$ | |
| | $\nabla g(\mathbf{p}_i) \cdot \mathbf{n}_i < 0$ | |
| | $Q_{\text{task}}(\theta, \mathbf{H}, \mathbf{p}, \mathbf{n}) \in \mathcal{G}_{\text{task}}$ | task constraints |

where $\theta = (\theta^{1T}, \dots, \theta^{NT})^T$ is the vector of the generalized joint positions for N fingers. $\mathbf{H} \in SE(3)$ represents the hand position and orientation. $\mathbf{p} = (\mathbf{p}^{1T}, \dots, \mathbf{p}^{NT})^T$ and $\mathbf{n} = (\mathbf{n}^{1T}, \dots, \mathbf{n}^{NT})^T$ is the vector of fingertip positions and normal directions. h_i and l_i are functions derived from hand forward kinematics, to compute fingertip positions and normal direction. g is the implicit representation of the object surface and $\nabla g(\mathbf{p}^i)$ is the outward normal direction of object surface at \mathbf{p}^i . Q_{task} represents the task constraints and $\mathcal{G}_{\text{task}}$ contains all the grasp that suitable for the task.



Grasp control

Robotic Hands

STABILITY OF VARIABLE GRASP STIFFNESS CONTROL

In this paper, we address the control stability of reaching stiffness in robotic grasping by a variable stiffness control, as shown in Fig. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100.

Complicated with many assumptions

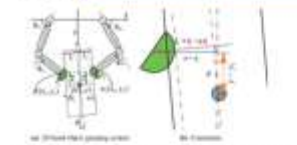


Fig. 1. A two-link robotic hand system with two degrees of freedom and a gripper. The gripper is used for grasping a rigid object with a contact force.

1.1 DYNAMICS

In this paper, we first consider the control model of the gripper and the contact force in the hand-grasper system. Then the overall dynamics of the system is derived. The equations in this section are represented in Fig. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100.

1.1 CONTACT MODEL OF CONTACT FORCE

$$F = -k(x - x_0) - b(\dot{x} - \dot{x}_0) \quad (1)$$

where x_0 and \dot{x}_0 are positive constant parameters which depend on the material of the gripper, and so on the deformation of the gripper. The gripper should be in contact with the object surface, as shown in Fig. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100.

$$\ddot{x} + b\dot{x} + kx = b\dot{x}_0 + kx_0 \quad (2)$$

1.1.1 ROBUSTNESS ANALYSIS

With the dynamic in section 1.1, the overall dynamics can be simplified as follows:

$$\ddot{x} + b\dot{x} + kx = b\dot{x}_0 + kx_0 \quad (3)$$

1.1.2 OVERALL DYNAMICS

The total kinetic energy of the overall system can be described as follows:

$$E = \sum_{i=1}^n \frac{1}{2} m_i \dot{x}_i^2 + \frac{1}{2} k x^2 + \frac{1}{2} b \dot{x}^2 \quad (4)$$

where $m_i = 1, 2, \dots, n$ is the mass of the gripper and $E = 0$ is the reference energy. The mass and inertia matrix of the gripper respectively.

$$M = \sum_{i=1}^n m_i \dot{x}_i^2 + \frac{1}{2} k x^2 + \frac{1}{2} b \dot{x}^2 \quad (5)$$

Then we have the overall dynamics for the gripper based system as follows:

$$M \ddot{x} + b \dot{x} + kx = b \dot{x}_0 + kx_0 \quad (6)$$

$$\ddot{x} + b\dot{x} + kx = b\dot{x}_0 + kx_0 \quad (7)$$

$$\ddot{x} + b\dot{x} + kx = b\dot{x}_0 + kx_0 \quad (8)$$

$$\ddot{x} + b\dot{x} + kx = b\dot{x}_0 + kx_0 \quad (9)$$

With the dynamic in section 1.1, the overall dynamics can be simplified as follows:

$$\ddot{x} + b\dot{x} + kx = b\dot{x}_0 + kx_0 \quad (10)$$

$$\ddot{x} + b\dot{x} + kx = b\dot{x}_0 + kx_0 \quad (11)$$

$$\ddot{x} + b\dot{x} + kx = b\dot{x}_0 + kx_0 \quad (12)$$

1.2 VARIABLE GRASP STIFFNESS CONTROL

Based on the analysis of the overall system dynamics, the following control law is designed for each finger in order to achieve grasp:

$$\ddot{x} + b\dot{x} + kx = b\dot{x}_0 + kx_0 \quad (13)$$


where \ddot{x}_0 is a diagonal positive definite matrix representing the grasping gain. \ddot{x}_0 is designed as the variable stiffness for each finger i in the same ratio for the overall gripper system (see Fig. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100).

1.3 STABILITY PROOF 1

Using the rate of energy method of Fig. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100.

$$\dot{E} = \sum_{i=1}^n m_i \dot{x}_i \ddot{x}_i + kx \dot{x} + b \dot{x}^2 \quad (14)$$

$$\dot{E} = \sum_{i=1}^n m_i \dot{x}_i \ddot{x}_i + kx \dot{x} + b \dot{x}^2 \quad (15)$$




Surprisingly Robust In-Hand Manipulation: An Empirical Study

Aditya Bhatt* Adrian Sieler* Steffen Puhlmann Oliver Brock

Presented at RSS 2021

* Both authors contributed equally to this work.

Model-based → Learning-based

ICRA

May 30 to June 5, 2021

Xi'an China



Soft Robotics for Delicate and Dexterous Manipulation

Robert Wood
Harvard University

ICRA 2021 Plenary Talk



https://www.youtube.com/watch?v=gskX0cwCTZQ&ab_channel=IEEERoboticsandAutomationSociety

IEEE 2021
ICRA
May 30 to June 5, 2021
Xi'an China



Soft Robotics for Delicate and Dexterous Manipulation

Robert Wood
Harvard University

ICRA 2021 Plenary Talk
June 1, 2021

