Affordance Prediction with Vision via Task-Oriented Grasp Quality Metrics

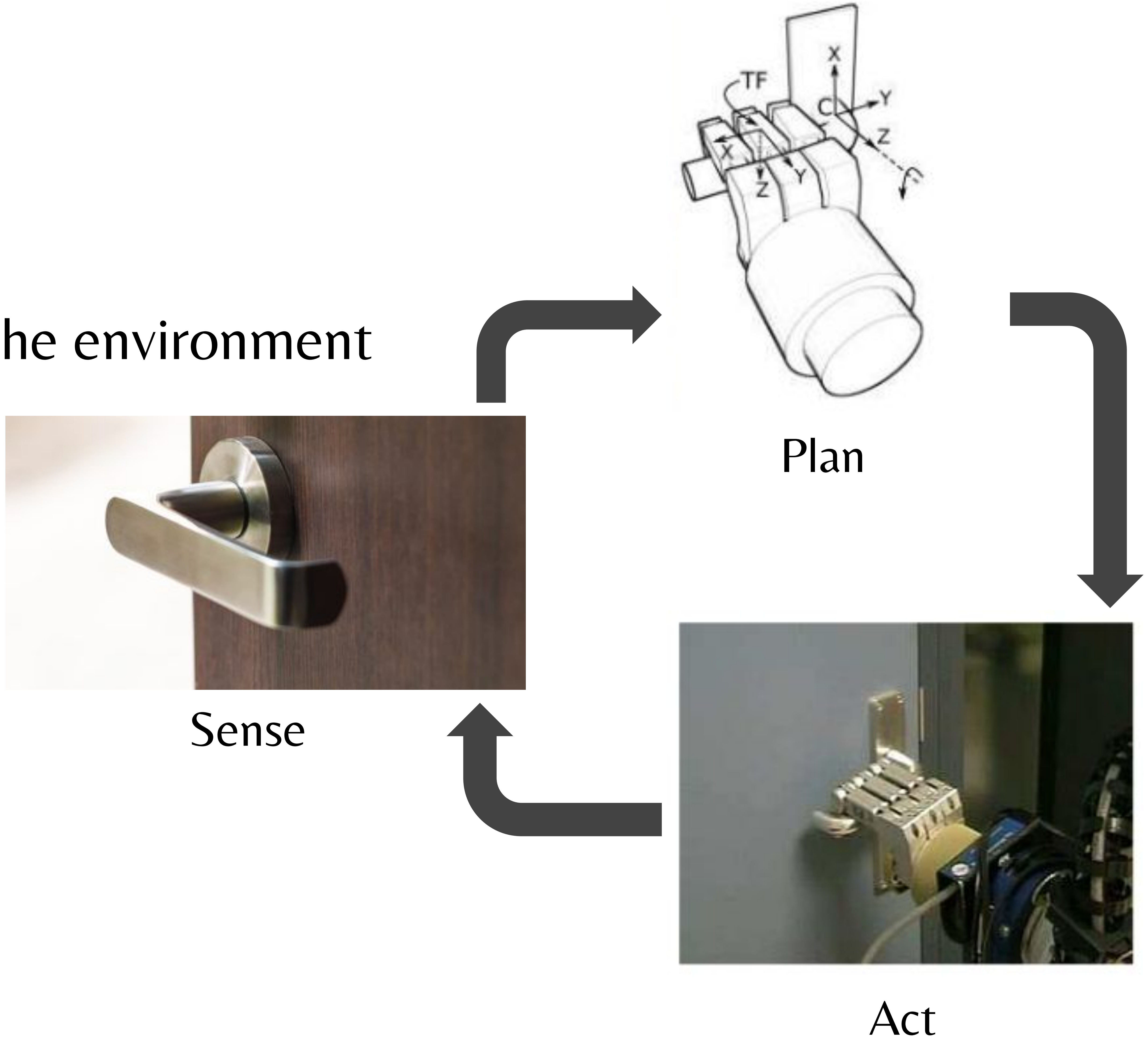
Luca Cavalli
luca3.cavalli@mail.polimi.it
Computer Science and Engineering

POLITECNICO
MILANO 1863

HONOURS PROGRAMME
HP-SR
in Information Technology

Autonomous Robots

- **Sense**: acquire and model data about the environment
- **Plan**: select the course of action
- **Act**: perform each planned action



Affordances

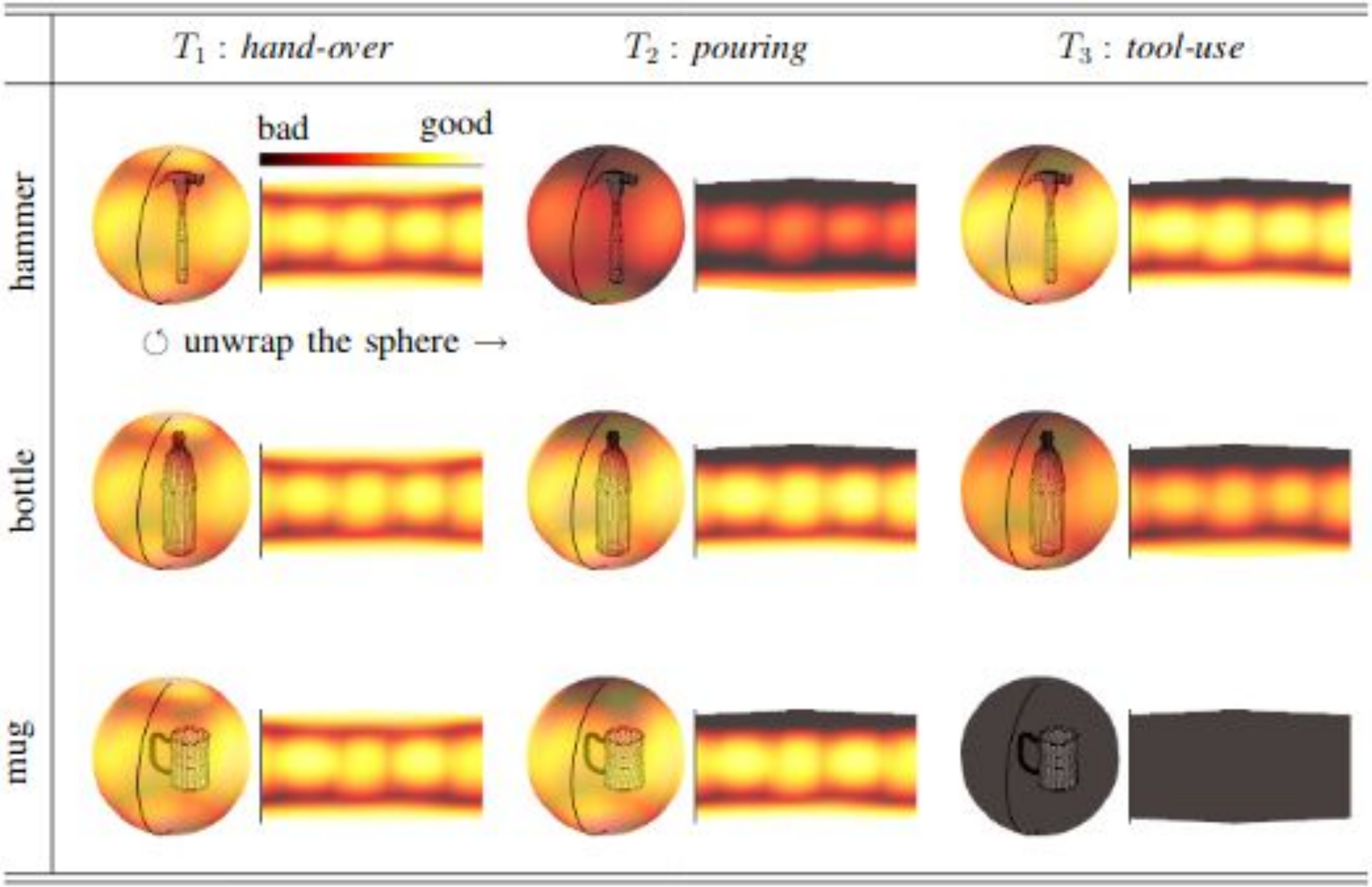
Affordances

Emergent **properties** embodied in the **relations** between an animal and its environment directly connected with the **possibility of action** of the animal with the environment

Michaels, C. Affordances: Four points of debate. ECOLOGICAL PSYCHOLOGY 15 (04 2003), 135–148.

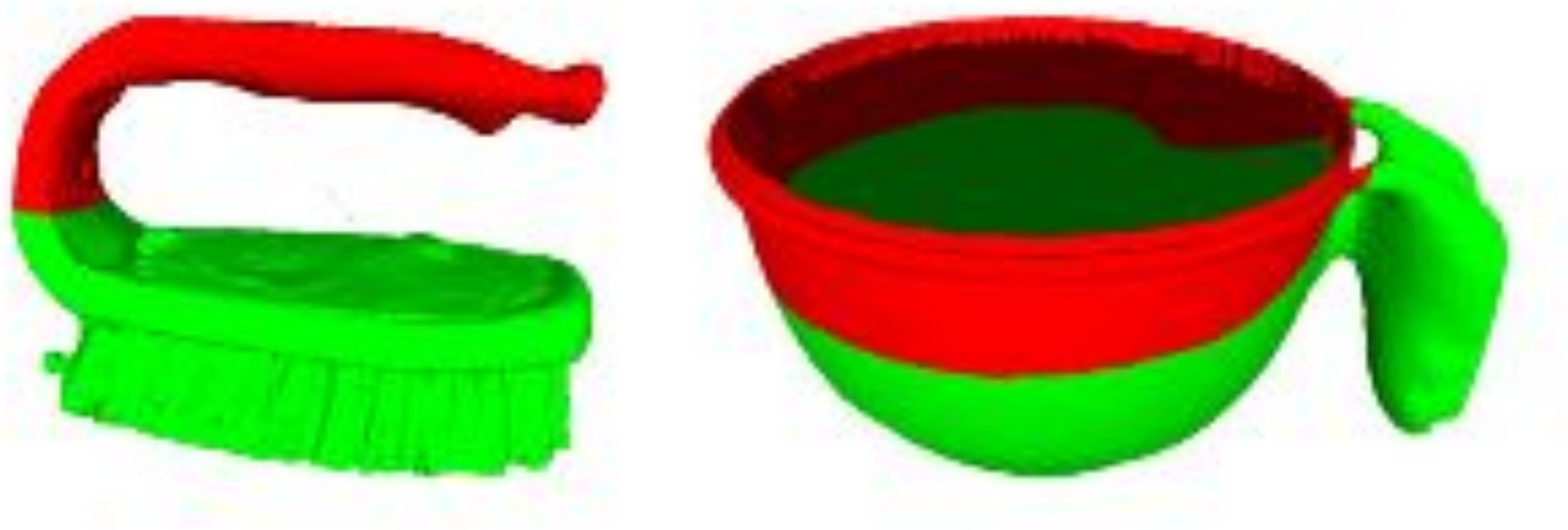
Task-Oriented Grasping

Affordance Learning



Song et al. Learning task constraints for robot grasping using graphical models. IROS (2010)

Affordance Learning



Detry et al. Task-oriented grasping with semantic and geometric scene understanding. IROS (2017)

State of the Art Limitations

State of the Art Limitations

- Categorical expression of tasks
  - Task definition intrinsic into the dataset associated to its label
  - Not easily extensible number of different tasks
  - Not possible to fine tune the task definition

State of the Art Limitations

- Categorical expression of tasks
  - Task definition intrinsic into the dataset associated to its label
  - Not easily extensible number of different tasks
  - Not possible to fine tune the task definition

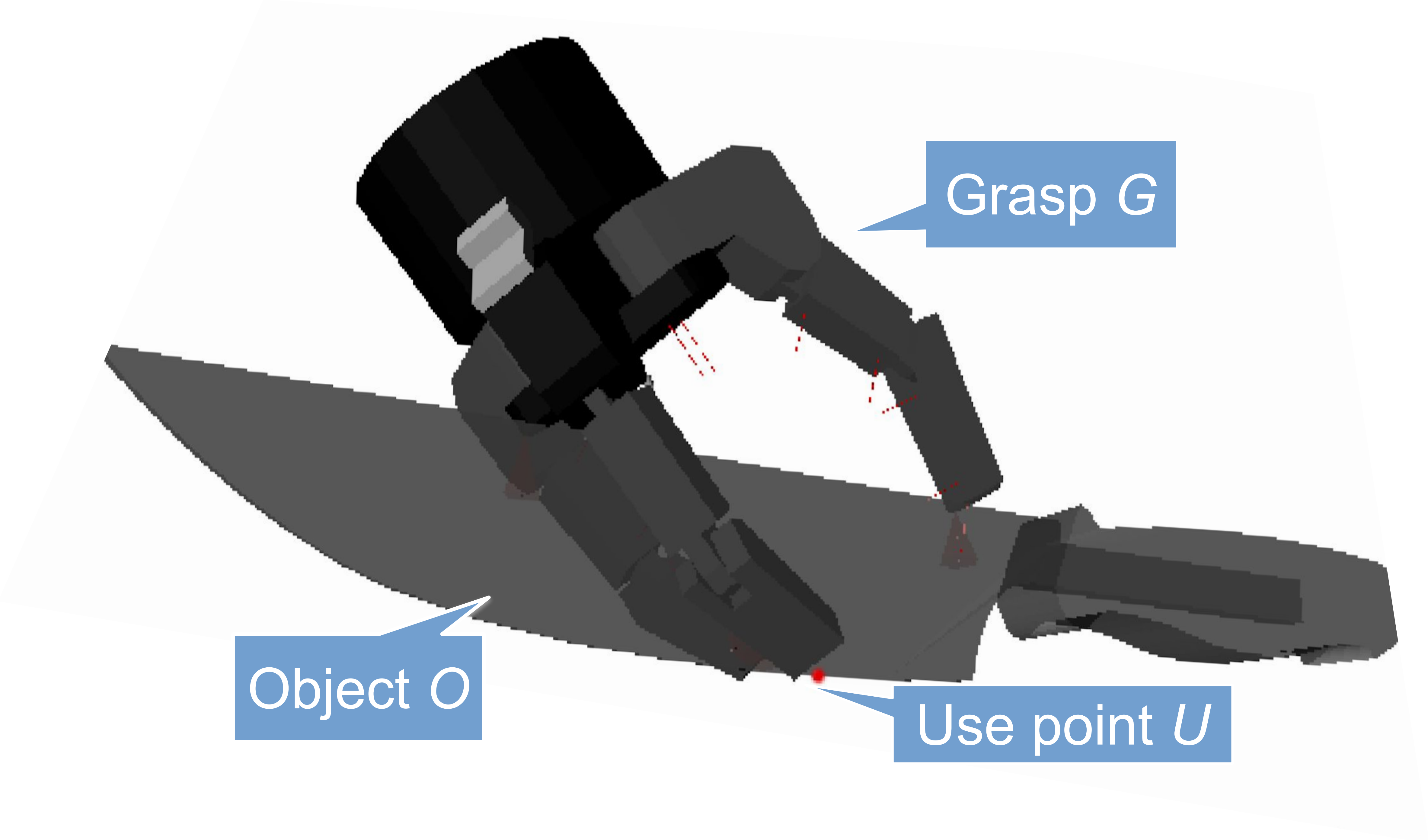
- Human labeling
  - Slow process, prevents scaling of dataset size
  - Biased towards human hand affordances
  - No guarantee on optimality

Proposed Approach

- ***Affordance function*** for T with object O, grasp G, and use point U:

$$F_T : (O, G, U) \mapsto \mathbb{R}$$

The higher the more suited (O, G, U) are for task T, e.g., for cutting



Proposed Approach

- ***Affordance function*** for T with object O, grasp G, and use point U:

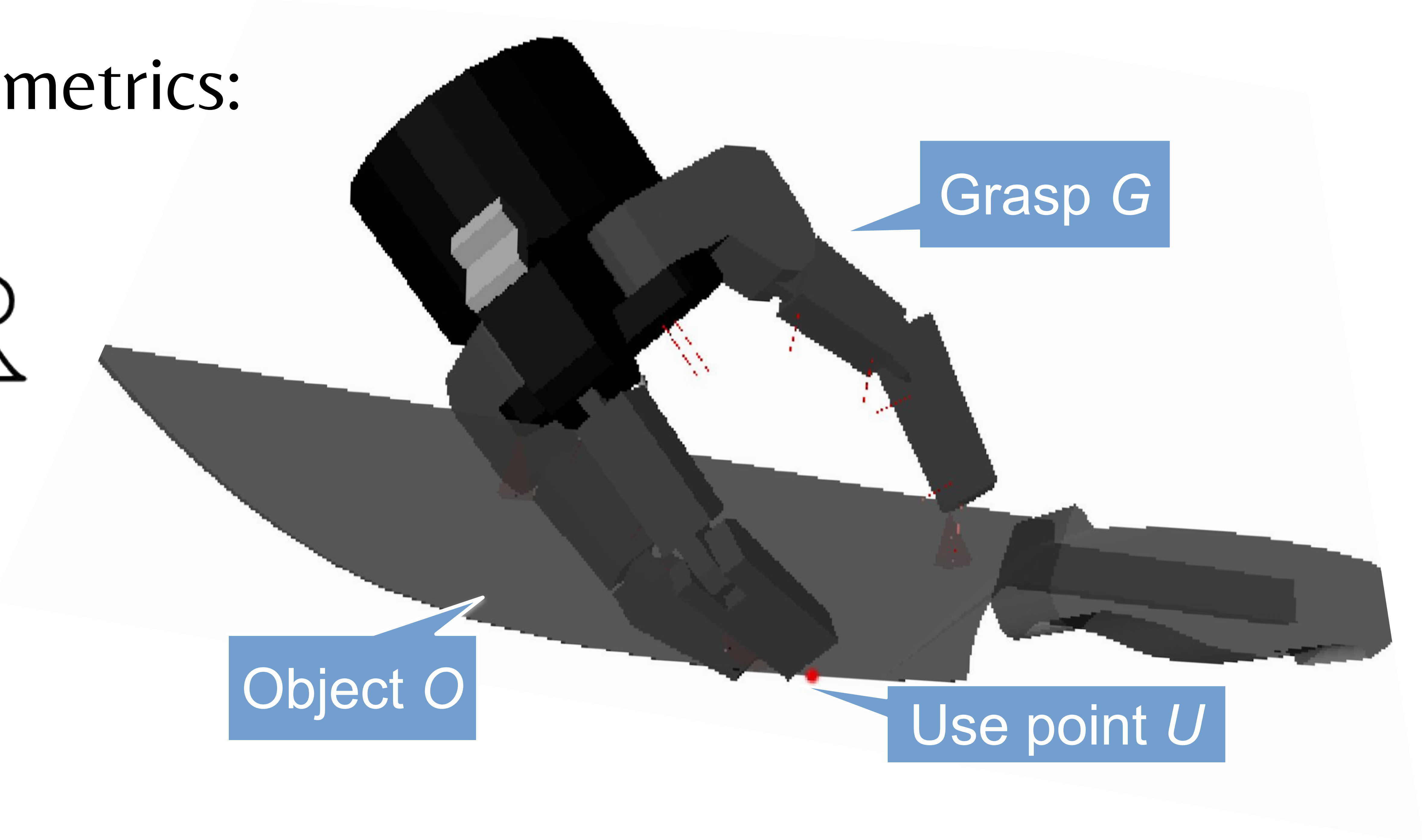
$$F_T : (O, G, U) \mapsto \mathbb{R}$$

The higher the more suited (O, G, U) are for task T, e.g., for cutting

- ***Approximated*** as a function of base grasp metrics:

$$\tilde{F}_T : \phi \in \mathbb{R}^n \mapsto \mathbb{R}$$

Base grasp metrics inferred with vision



Selected Metrics

State of the Art Metrics:

- Grasp Robustness

Geometrical Metrics:

- Rotational Inertia
- Momentum discharge efficiency
- Use local geometry

Optimization Metrics:

- Hand effort on impact
- Hand effort on hold
- Force transmitted to use

Selected Metrics

State of the Art Metrics:
- Grasp Robustness

Geometrical Metrics:
- Rotational Inertia
- Momentum discharge efficiency
- Use local geometry

Optimization Metrics:
- Hand effort on impact
- Hand effort on hold
- Force transmitted to use

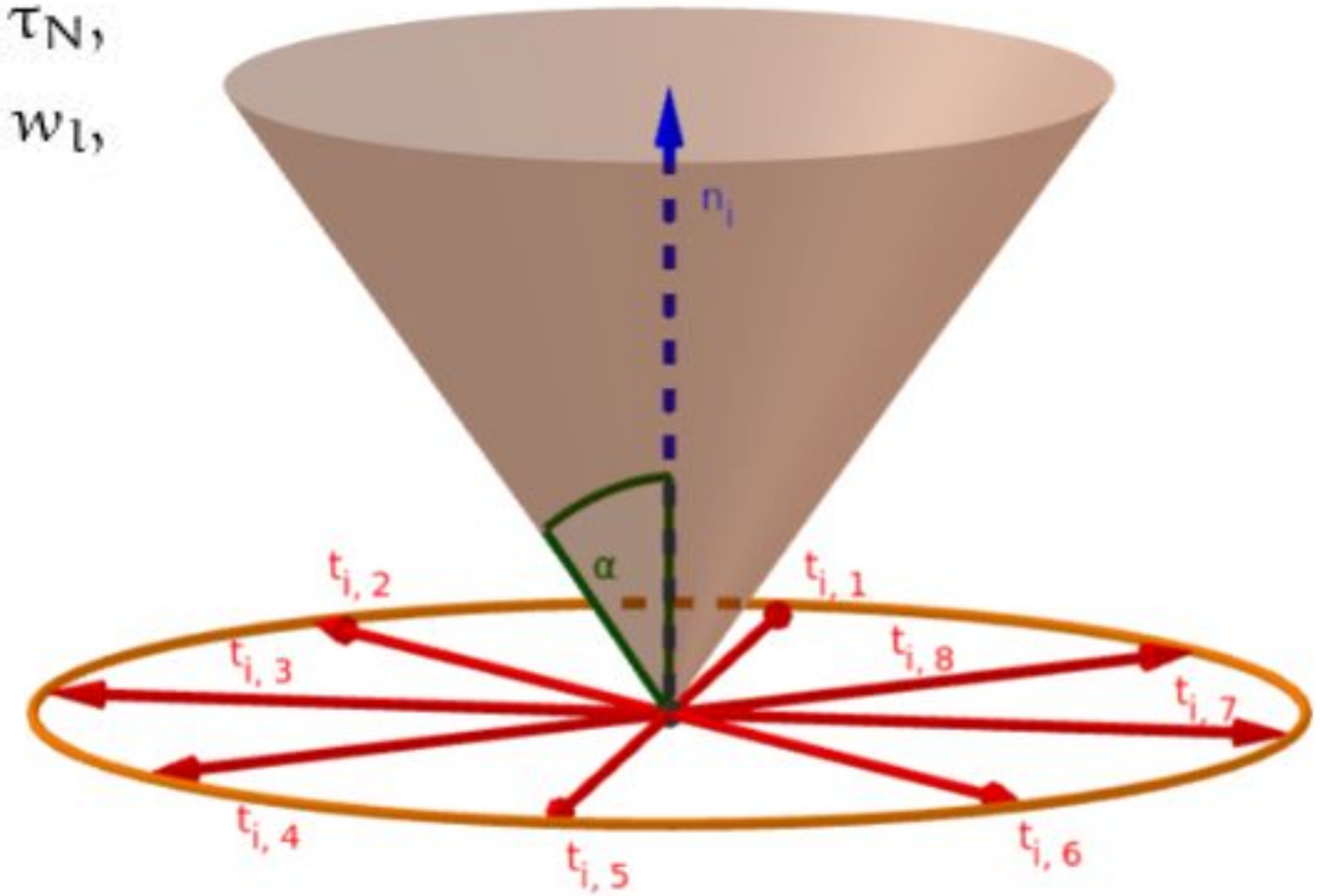
$$\sum_{j=1}^{8} t_{i,j} \leqslant \mu_s n_i, \qquad i = 1 \dots n_c$$

$$\sum_{i=1}^{n_c} M_i \begin{bmatrix} t_{i,1} \\ \vdots \\ t_{i,8} \\ n_i \end{bmatrix} + \sum_{j=1}^{k} \bar{W}_j + \sum_{l=1}^{v} W_l w_l = 0,$$

$$0 \leqslant t_{i,j}, \qquad i = 1 \dots n_c, j = 1 \dots 8$$

$$0 \leqslant n_i \leqslant \tau_N, \qquad i = 1 \dots n_c$$

$$0 \leqslant w_l, \qquad l = 1 \dots v$$



Selected Affordance Functions

Beating

function $\tilde{F}_{beat}(\epsilon, \delta, I, E_i, E_h)$
  if $(\epsilon < \tau_\epsilon \; || \; \delta < \tau_\delta \; || \; \sum_{i=1}^{6} E_h[i] == \infty)$ **then**
    return $-\infty$
  else
    return $\frac{I}{E_i}$
  end if
end function

Cutting

function $\tilde{F}_{cut}(\epsilon, U_\tau, U_g)$
  if $(\epsilon < \tau_\epsilon \; || \; U_g < \tau_{U_g})$ **then**
    return $-\infty$
  else
    return U_τ
  end if
end function

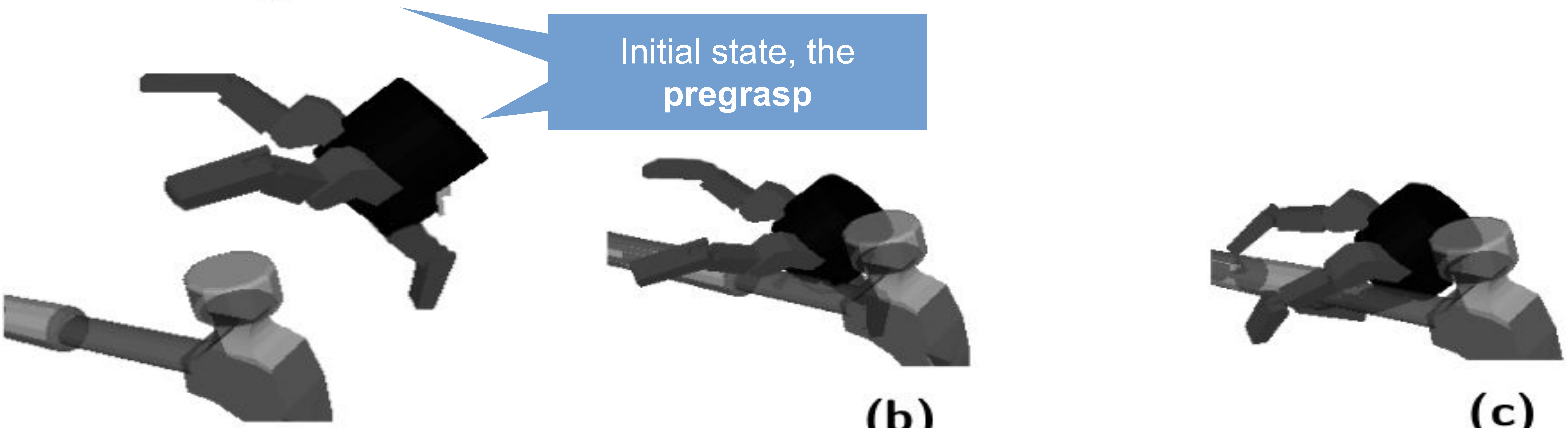
Picking

function $\tilde{F}_{pick}(E_h)$
  return $-\sum_{i=1}^{6} E_h[i]$
end function

Pregrasps

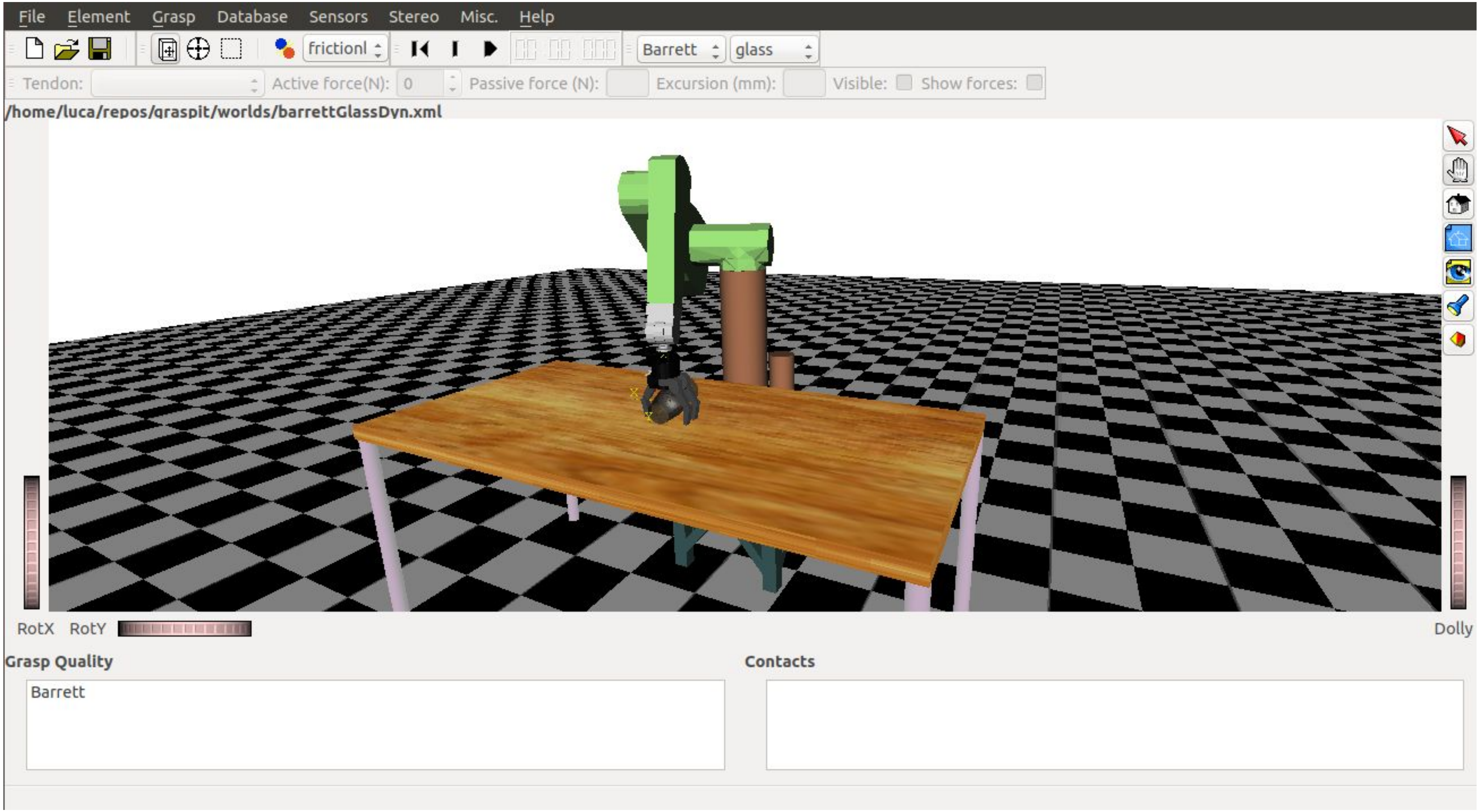
Decouple grasp from object through a fixed **grasping policy** GP:

$$GP(p_0, O) \mapsto \mathcal{G}(O)$$



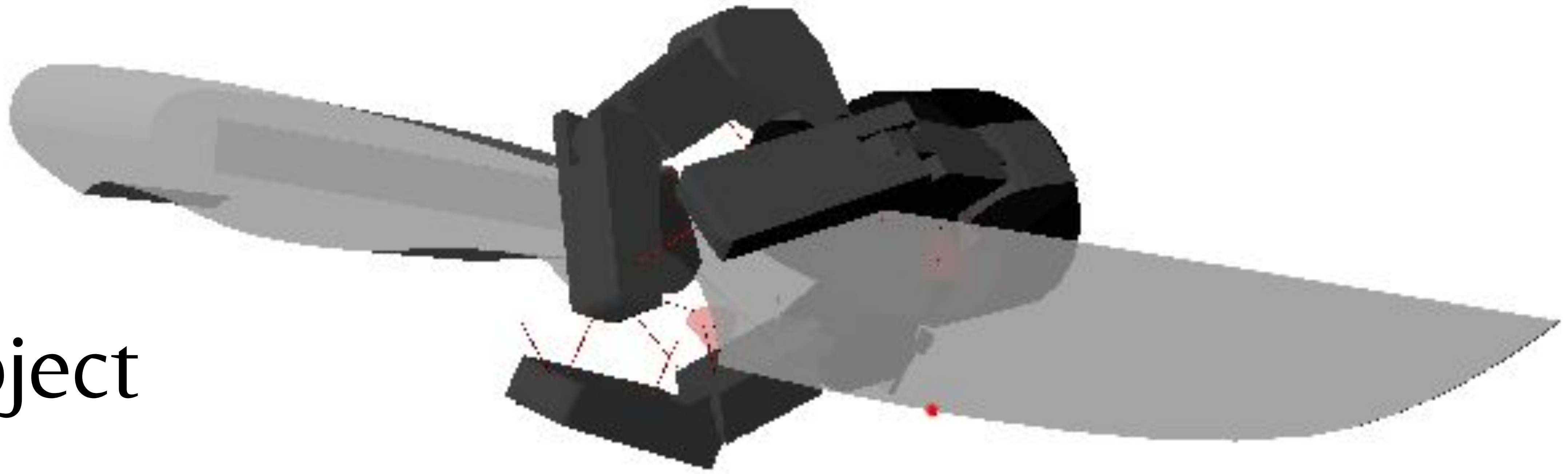
(a) (b) (c)

Simulating physics: GraspIt!



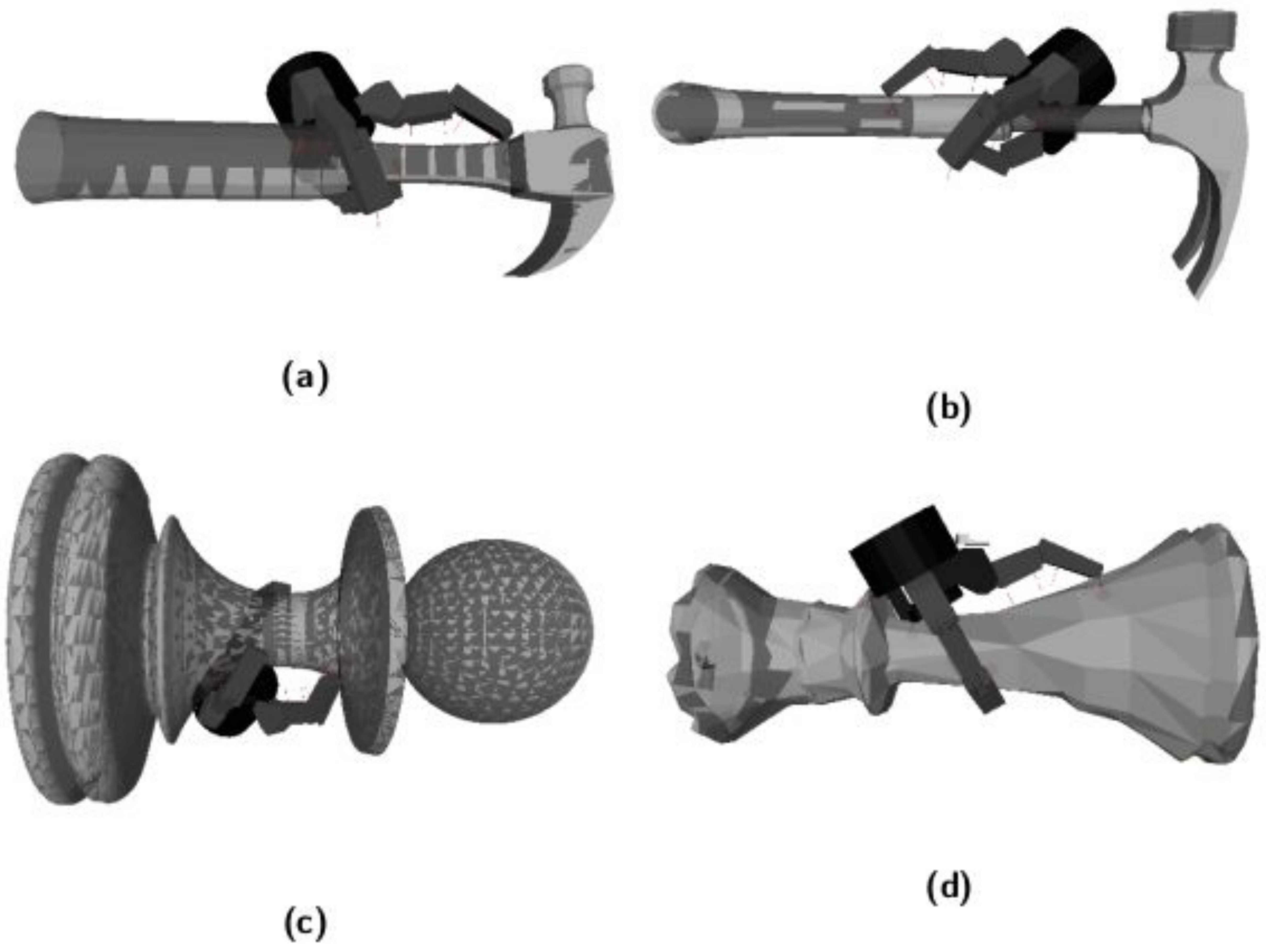
Data Collection

- Load objects from the Princeton Shape Benchmark
- Extract a *random* pregrasp and use location
- Simulate the random grasp on the target object
- Evaluate the metrics on the simulated grasp
- Log the pregrasp, use location and metric results

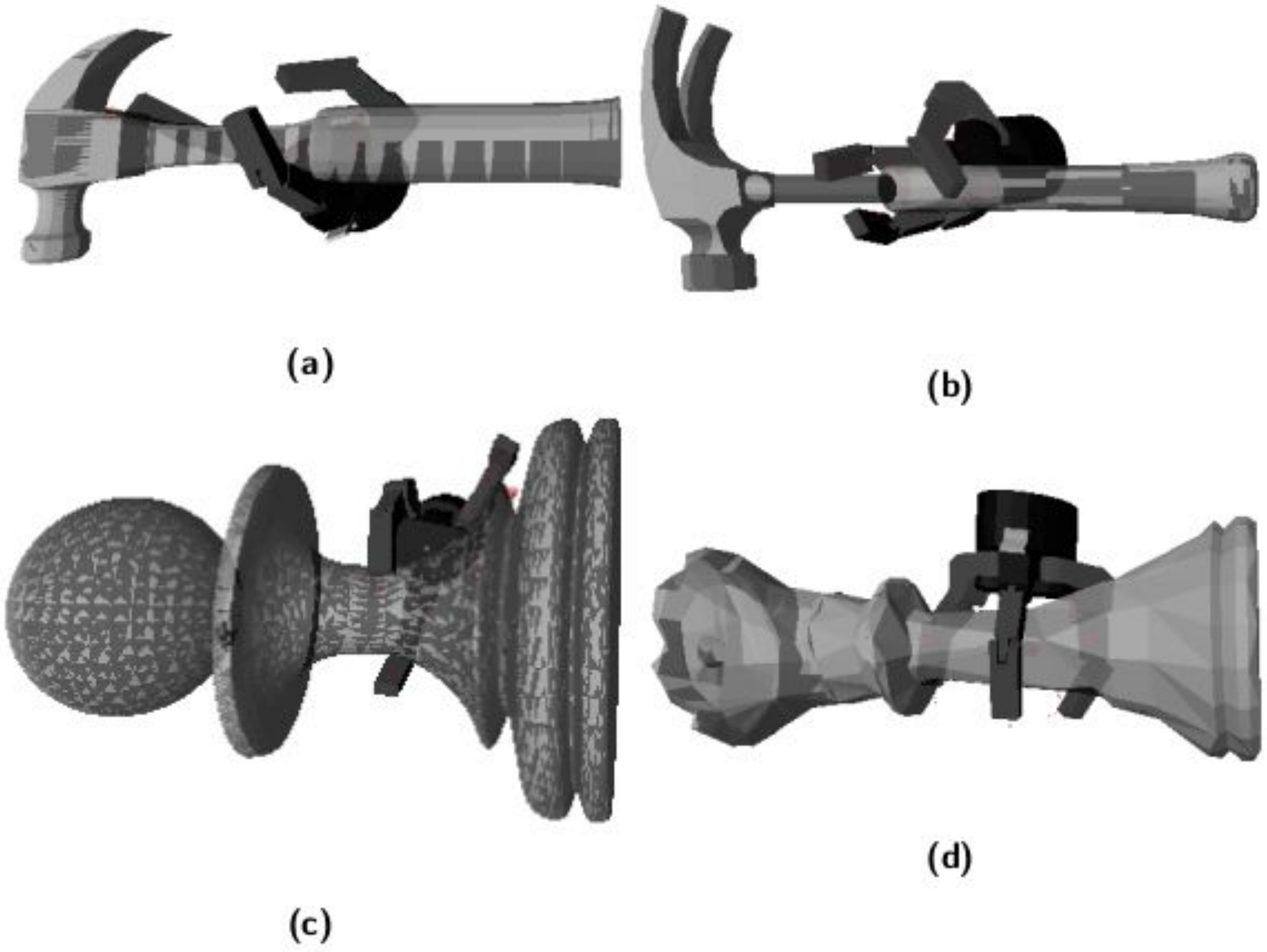


Round	core days	Samples[M]	GGS[M]	UGG[M]	UGG/obj[K]
1	350	400	20	1.25	56.82
2	280	97	91.3	5.7	259

Dataset Best: Picking



First data collection round



Second data collection round

Dataset Best: Picking

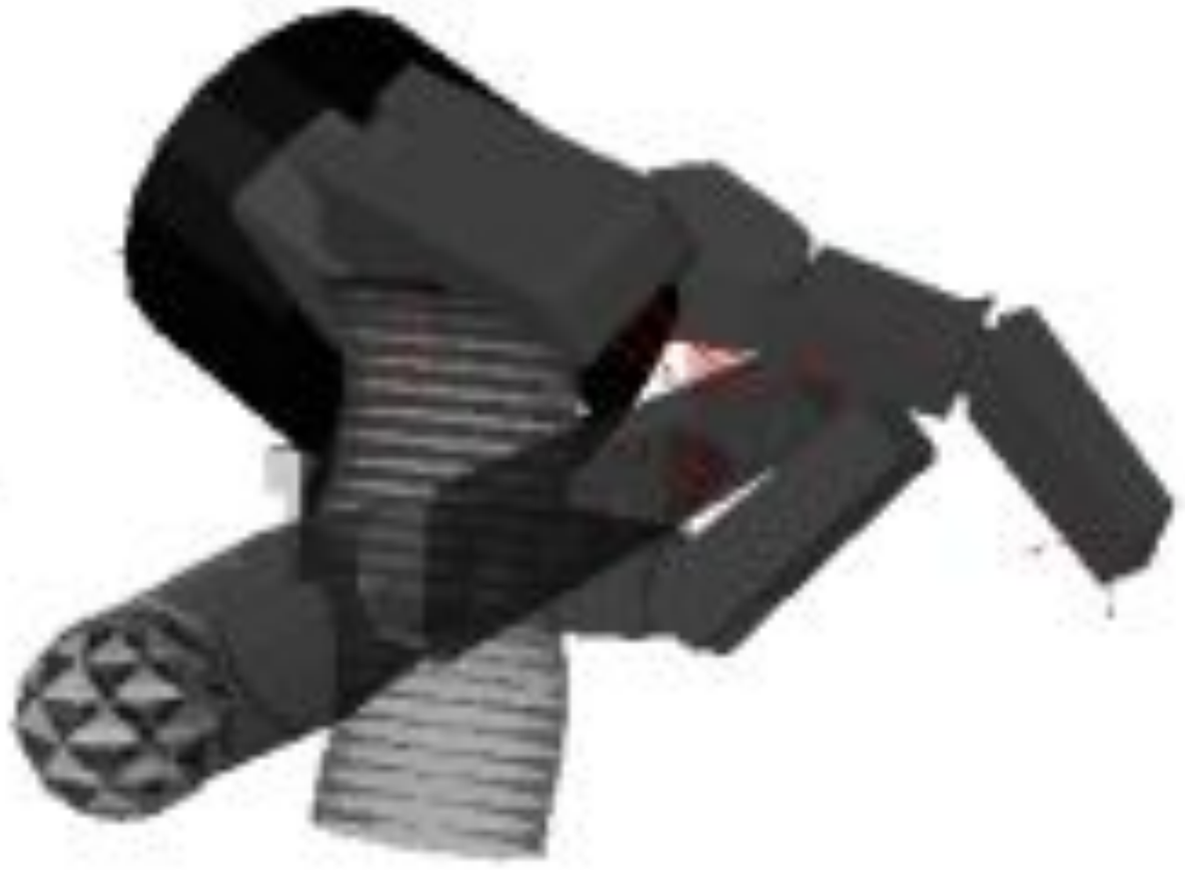
First data collection round

(c)

(d)

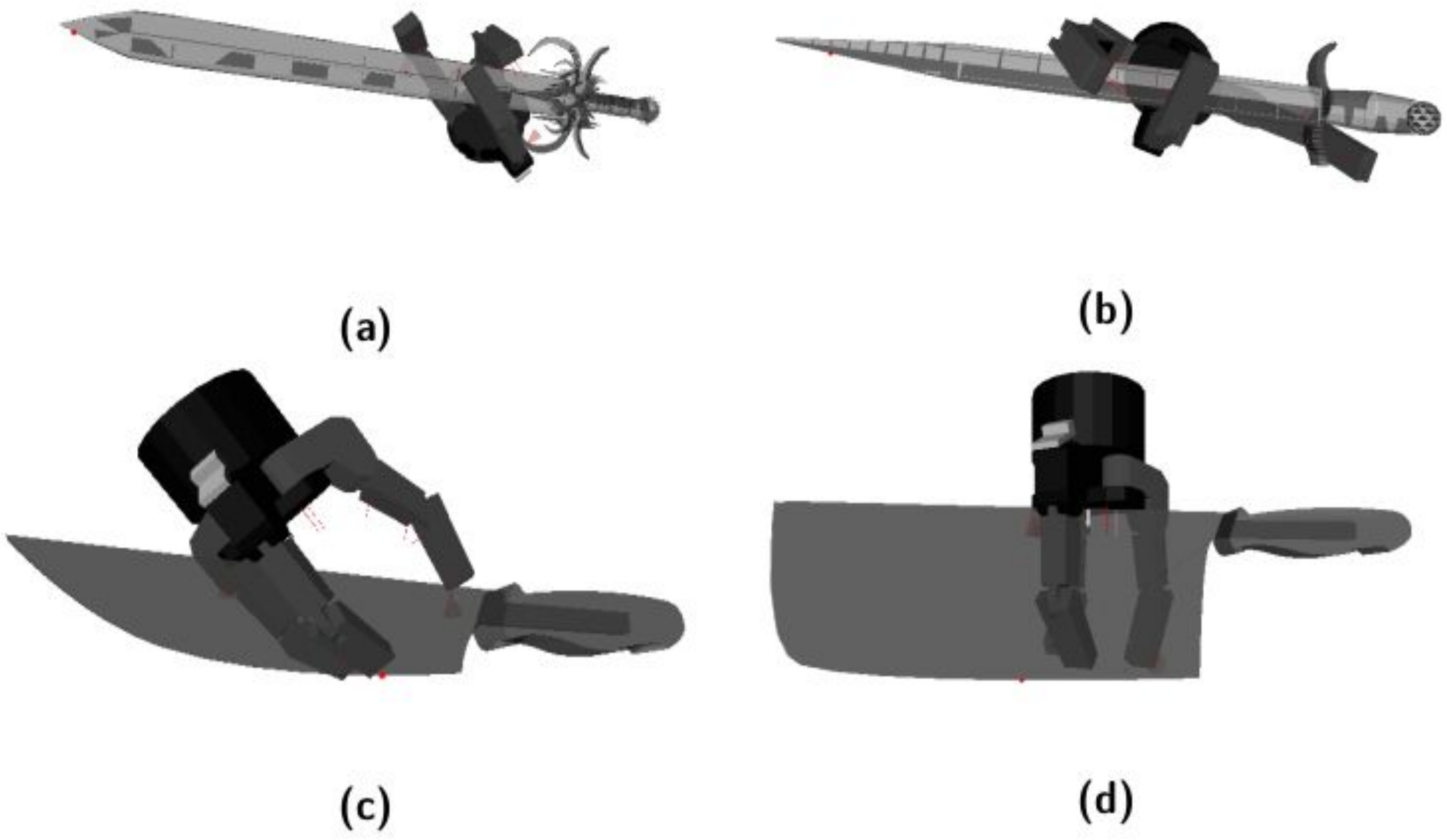
Second data collection round

(e)

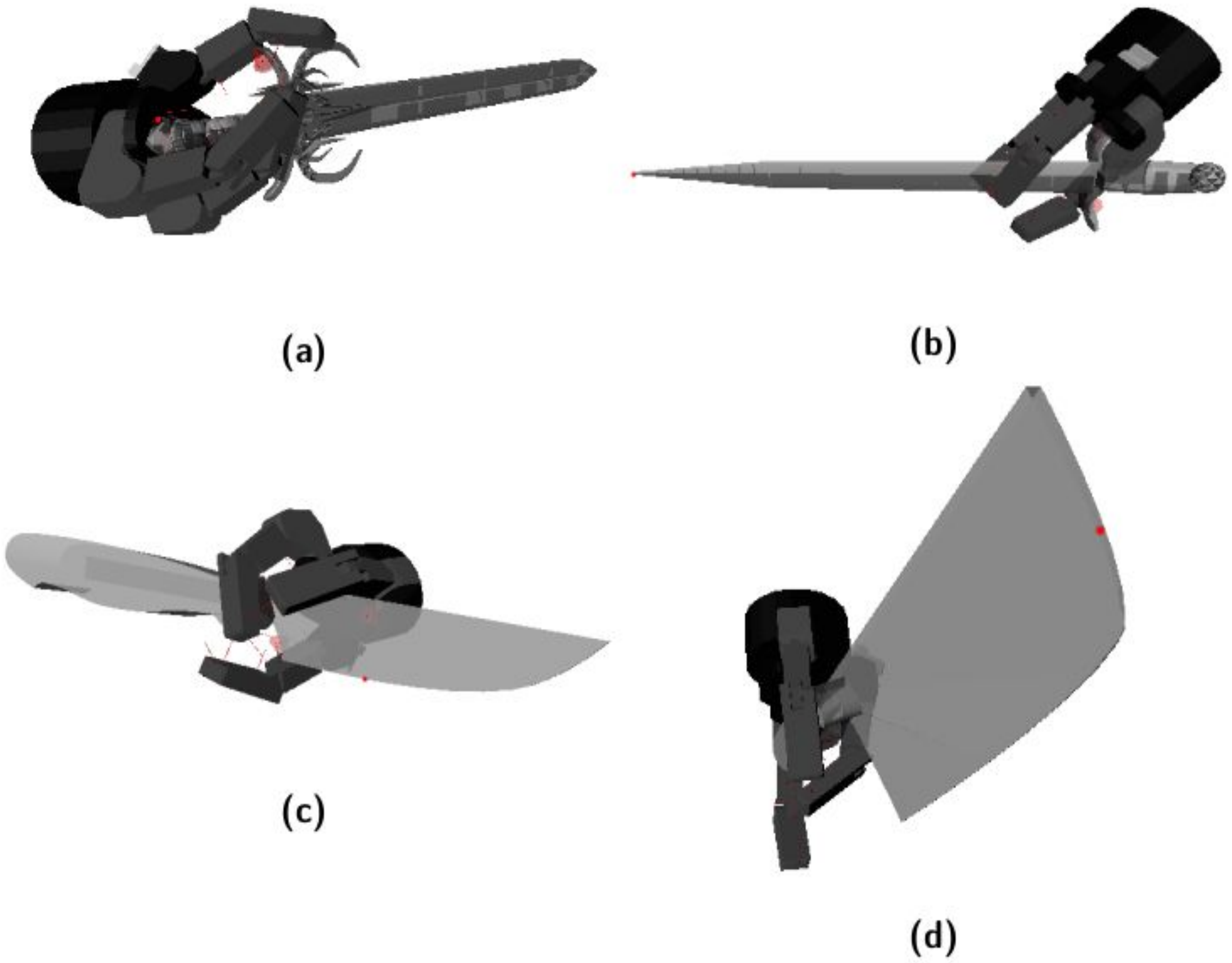


(f)

Dataset Best: Cutting

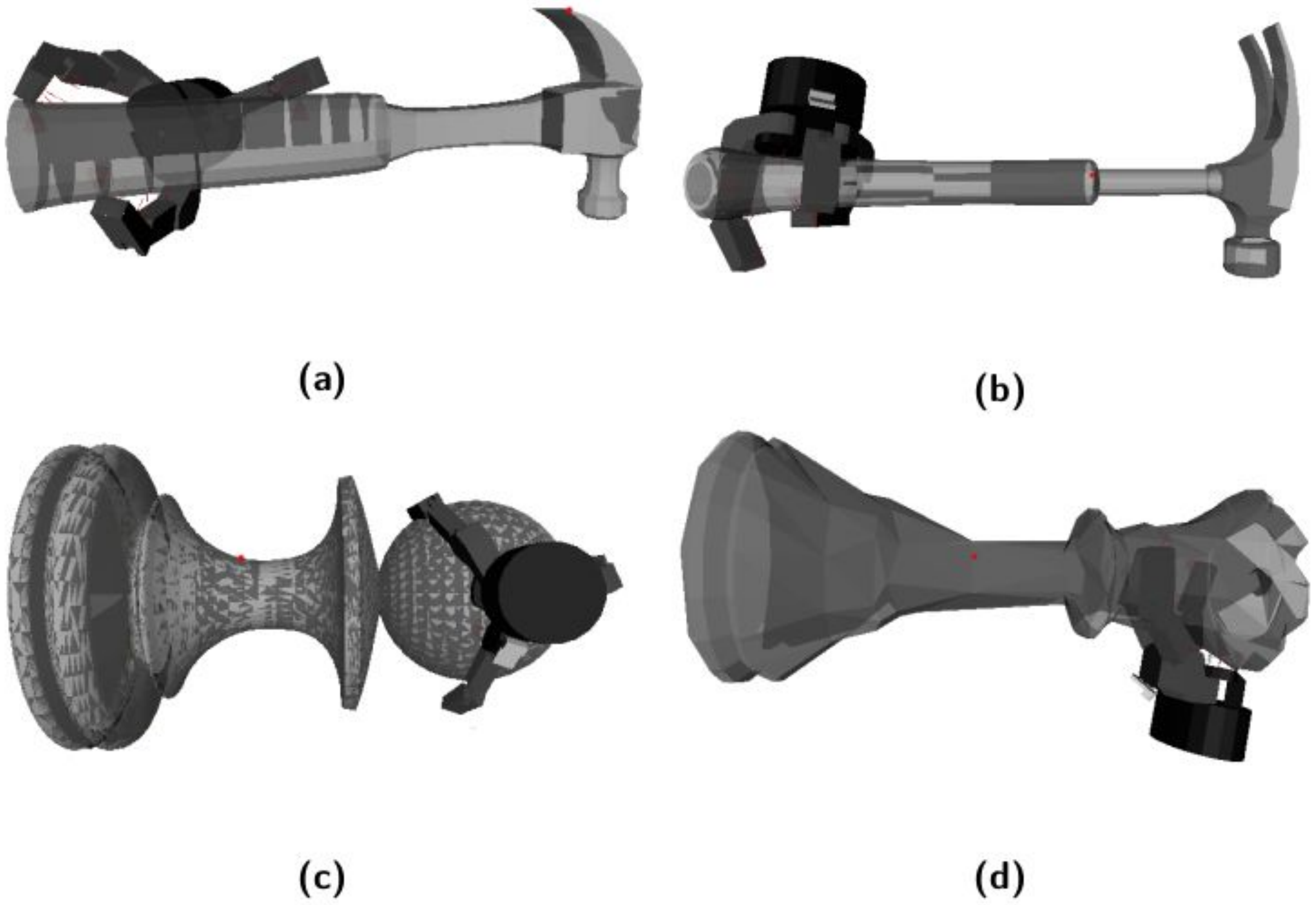


First data collection round

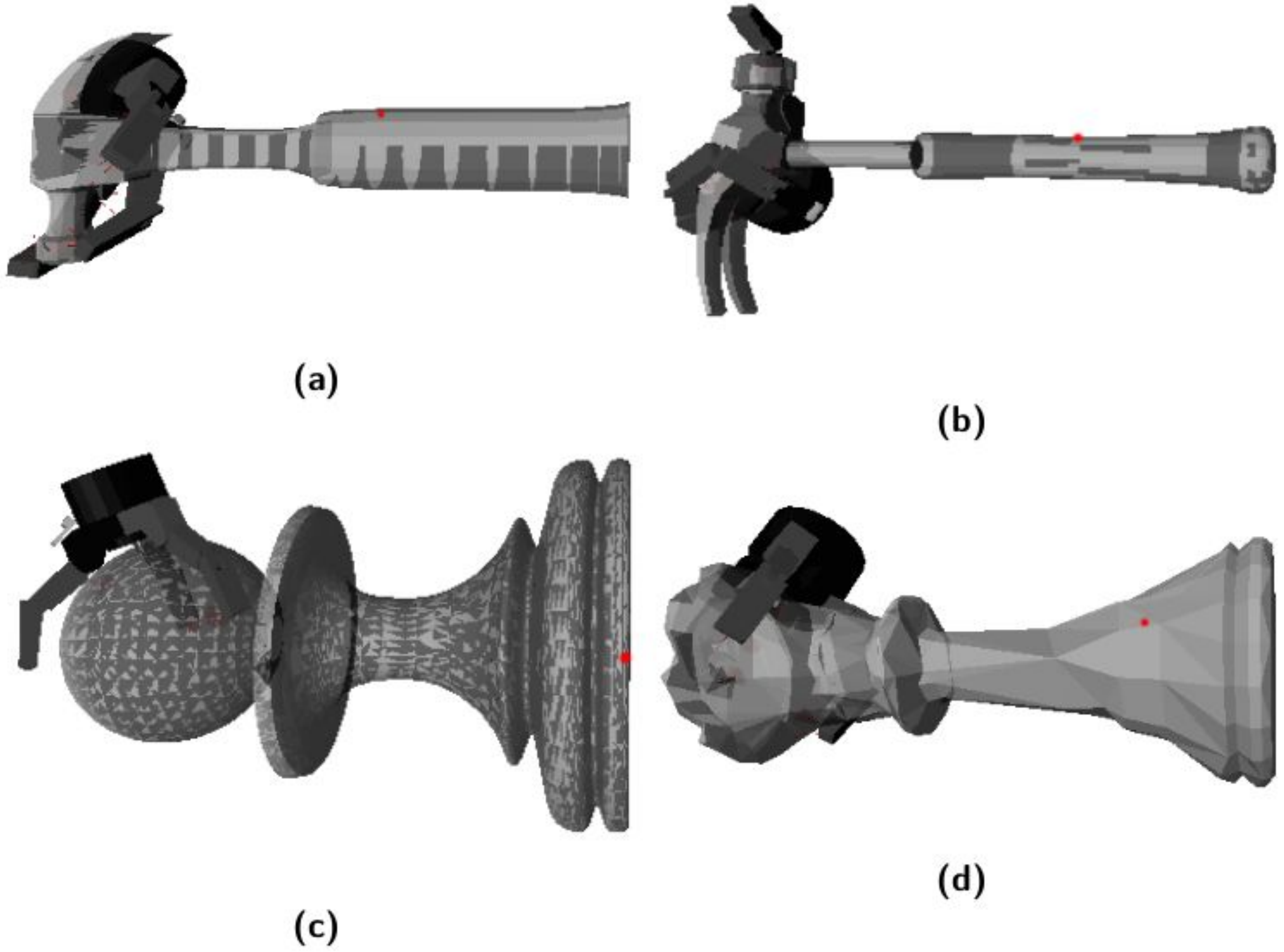


Second data collection round

Dataset Best: Beating

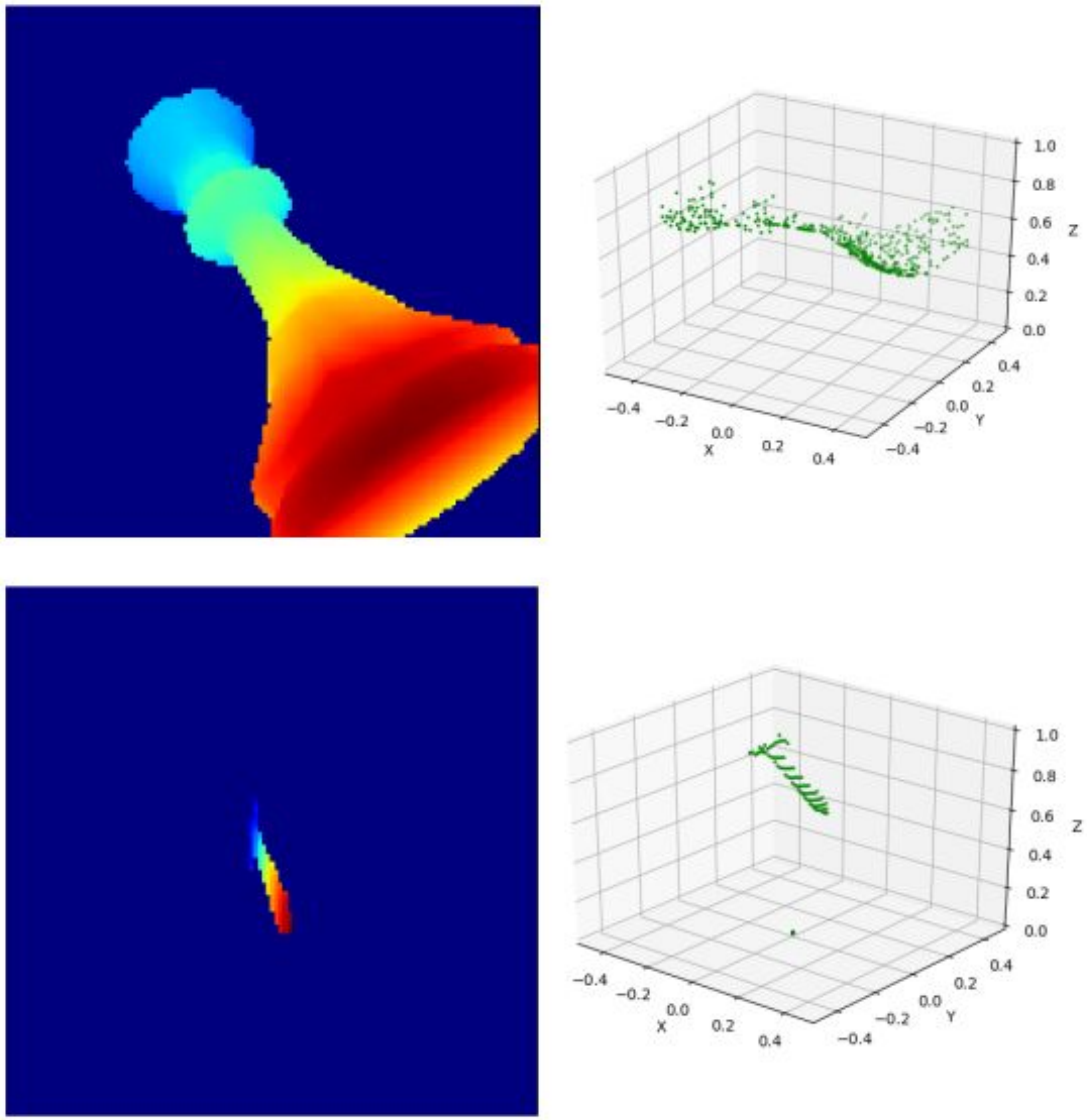


First data collection round

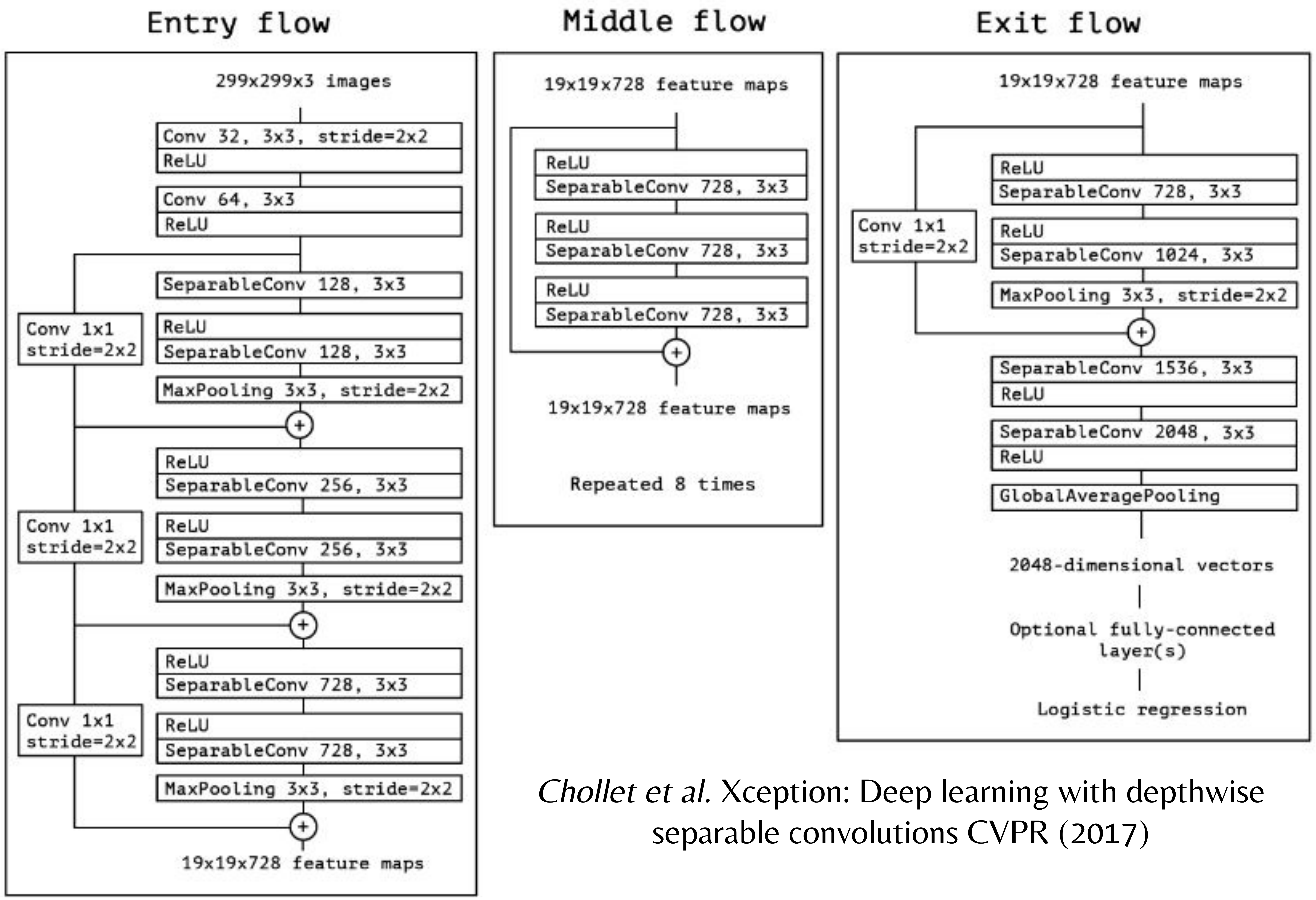


Second data collection round

Simulating vision

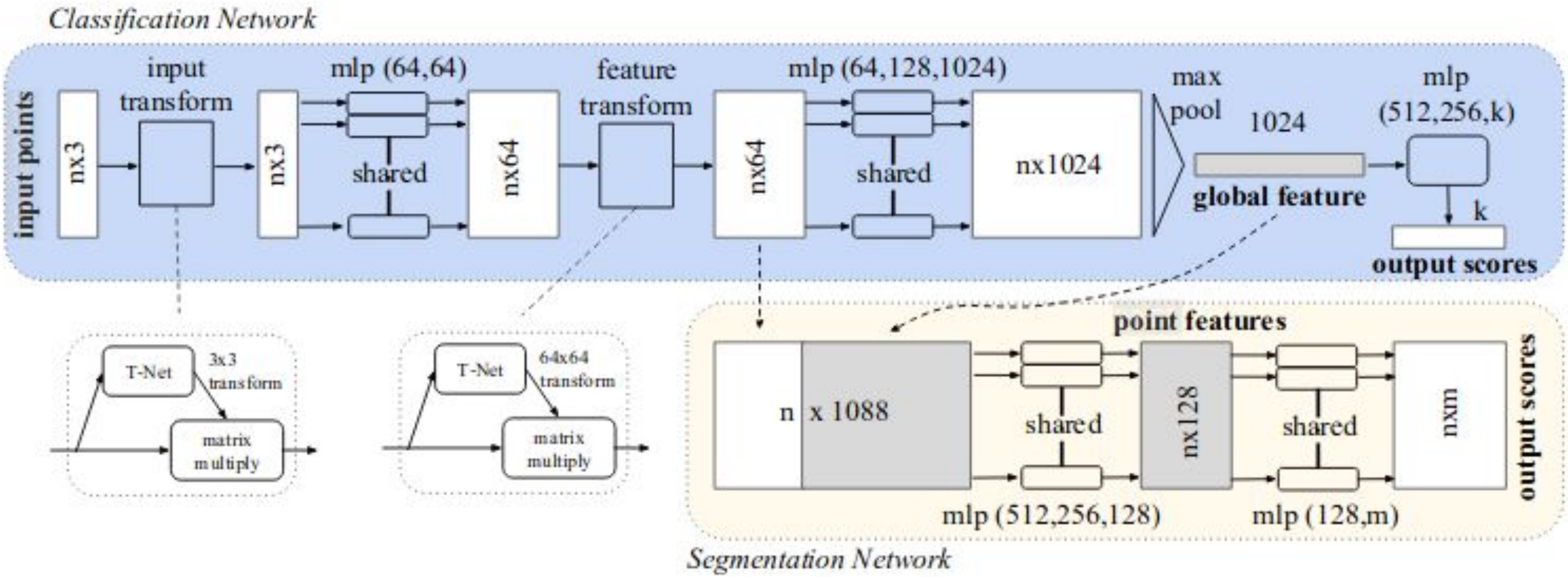


Learning Models: Xception



Chollet et al. Xception: Deep learning with depthwise separable convolutions CVPR (2017)

Learning Models: PointNet



Qi et al. Pointnet: Deep learning on point sets for 3d classificatino and segmentation CVPR (2017)

Learning Models: Local PointNet

- PointNet captures local geometrical patterns with no explicit notion of locality
- Bias PointNet towards capturing hierarchical local geometrical patterns like Convolutional Neural Networks do, while preserving point cloud processing

Learning Models: Local PointNet

- PointNet captures local geometrical patterns with no explicit notion of locality
- Bias PointNet towards capturing hierarchical local geometrical patterns like Convolutional Neural Networks do, while preserving point cloud processing
- Use the neighborhood of pixels in the depth image as a feature for locality of the resulting points
- Use the PointNet to compute point embeddings

Learning Models: Local PointNet

- PointNet captures local geometrical patterns with no explicit notion of locality
- Bias PointNet towards capturing hierarchical local geometrical patterns like Convolutional Neural Networks do, while preserving point cloud processing
- Use the neighborhood of pixels in the depth image as a feature for locality of the resulting points
- Use the PointNet to compute point embeddings
- Preserve the pixel neighborhood of point embeddings from the original depth image
- Use Xception to process point embeddings to the final output

Learning Task Separation

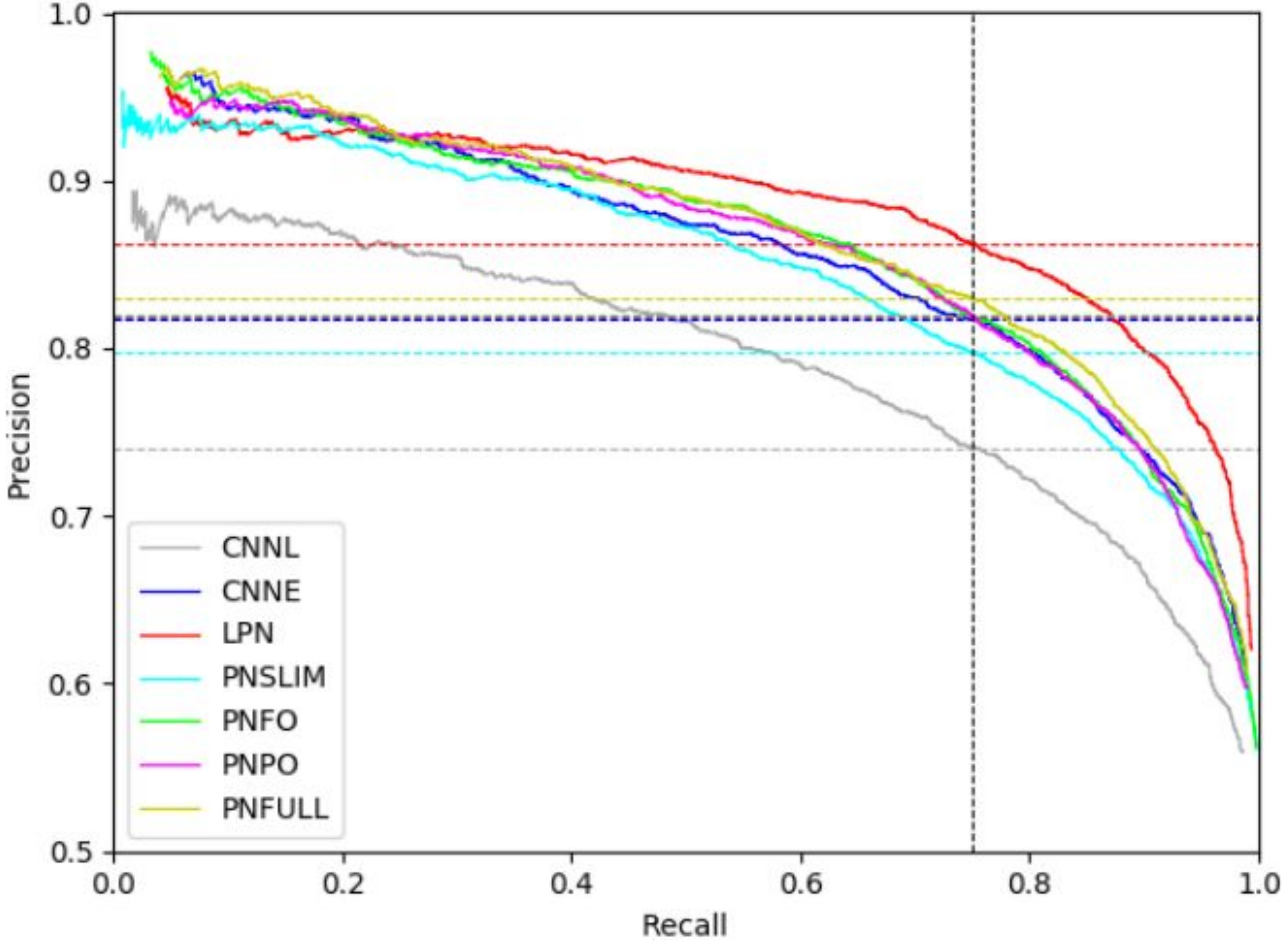
function $\mathcal{M}^{\Phi}(g, u)$
  if $\mathcal{M}_{C}^{\Phi}(g) < \tau_C$ **then**
    return v_{fail}
  else
    return $\mathcal{M}_{R}^{\Phi}(g, u)$
  end if
end function

Evaluation model built in two steps:

- First filter stable grasps only with a general classifier
- Then infer the specific metrics only from stable grasps with a specialized regressor

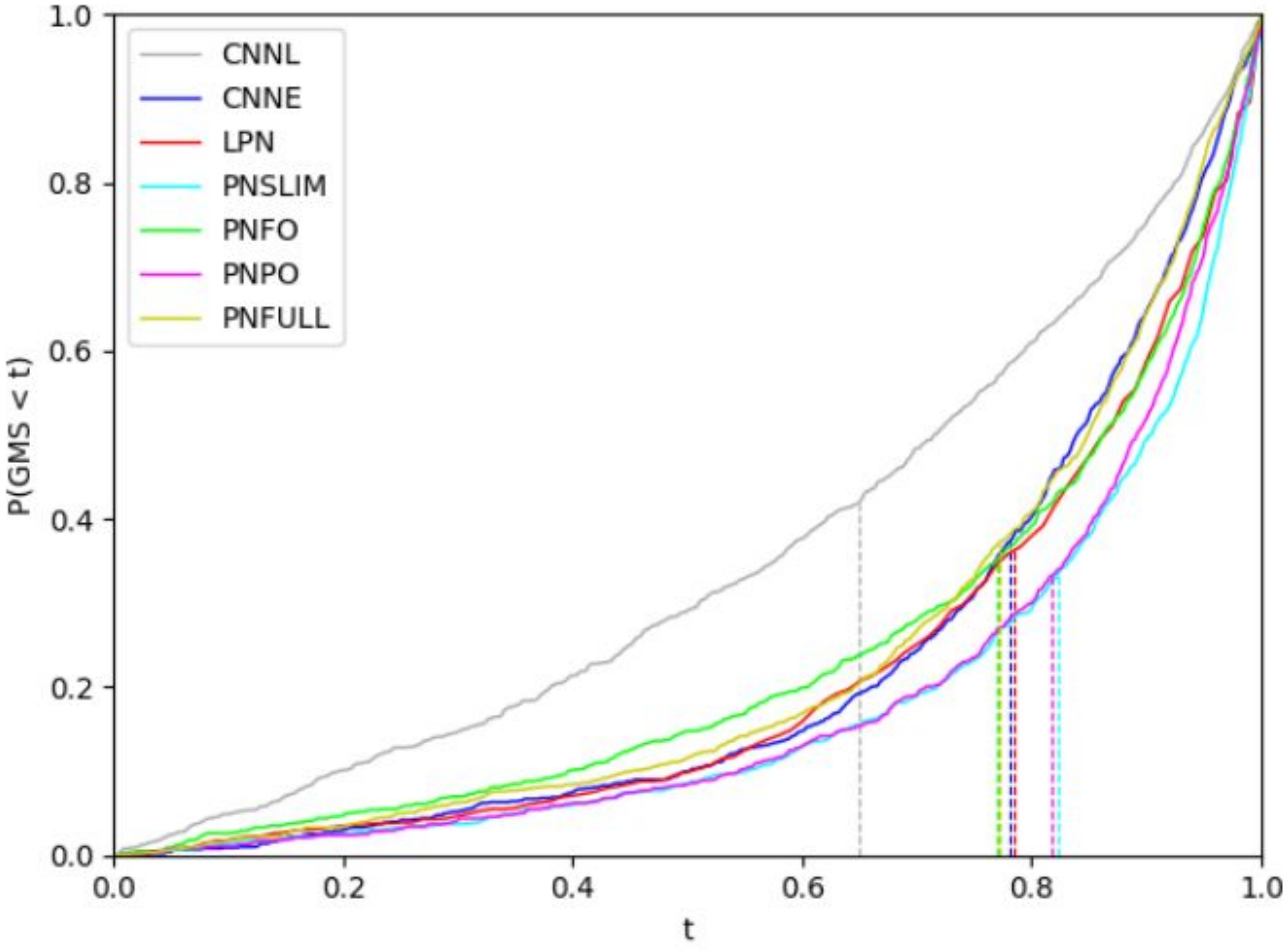
Benchmarking: Classification

Model code	Cross entropy	Precision
LPN	**0.3842**	**0.856**
PNFULL	0.4398	0.829
PNFO	0.4403	0.820
PNPO	0.4628	0.819
CNNE	0.4400	0.818
PNSLIM	0.4696	0.800
CNNL	0.5537	0.741

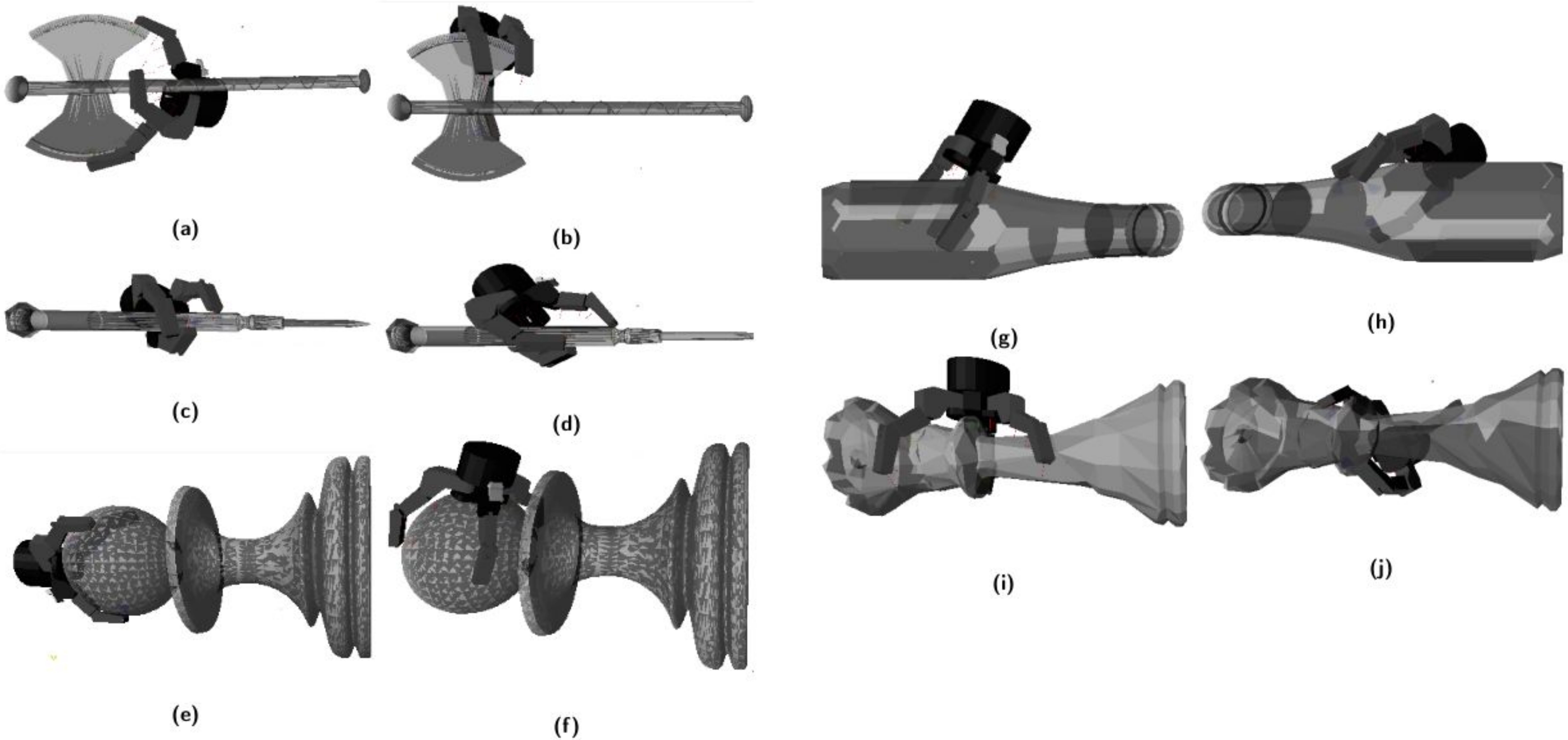


Benchmarking: Regression

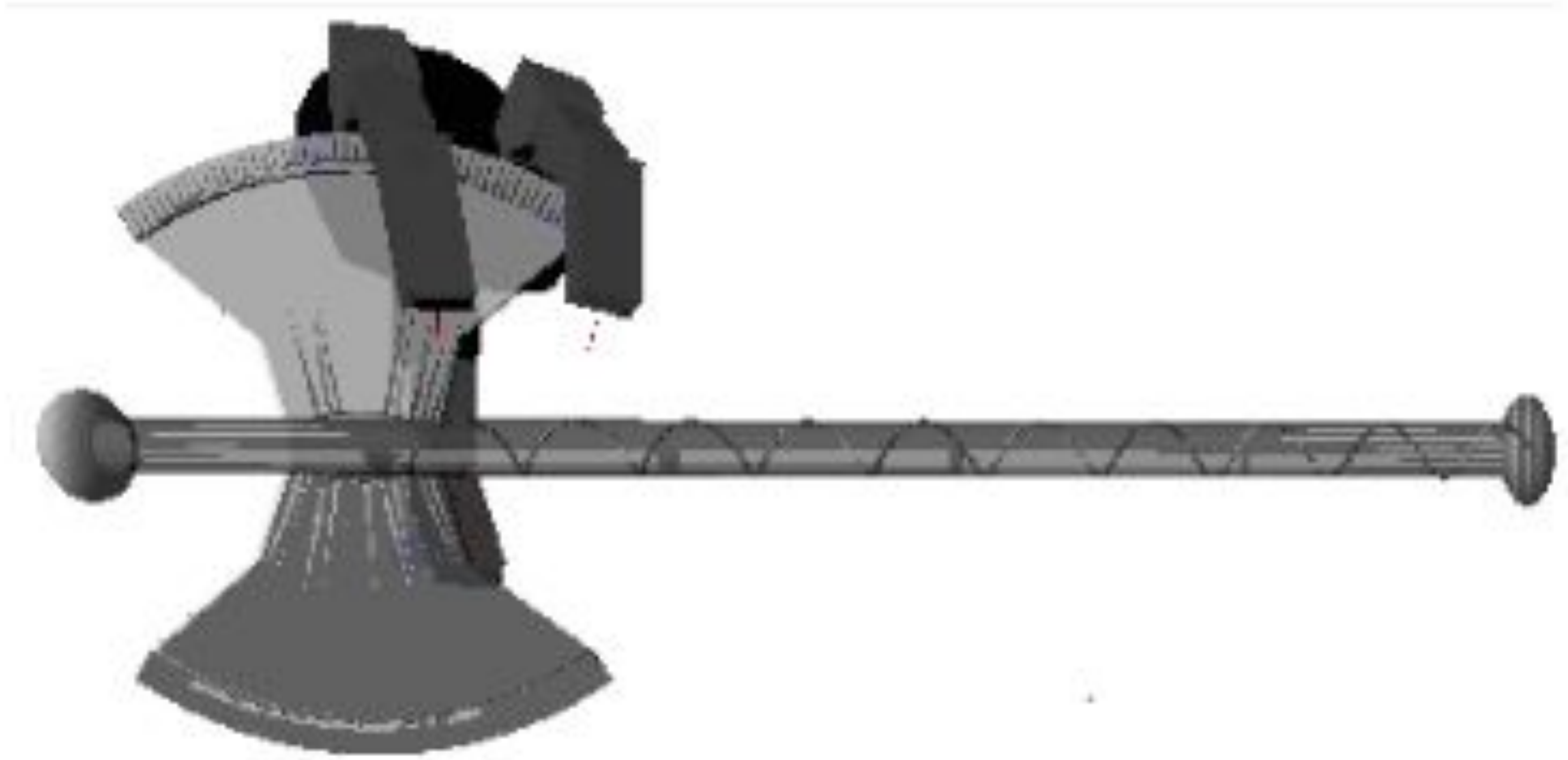
Model code	MSE	CA	E[GMS]
PNPO	**0.033**	**0.73**	0.818
PNSLIM	0.034	0.72	**0.824**
PNFULL	0.034	0.71	0.771
PNFO	0.035	0.72	0.770
LPN	0.037	0.69	0.785
CNNE	0.038	0.69	0.781
CNNL	0.050	0.58	0.649

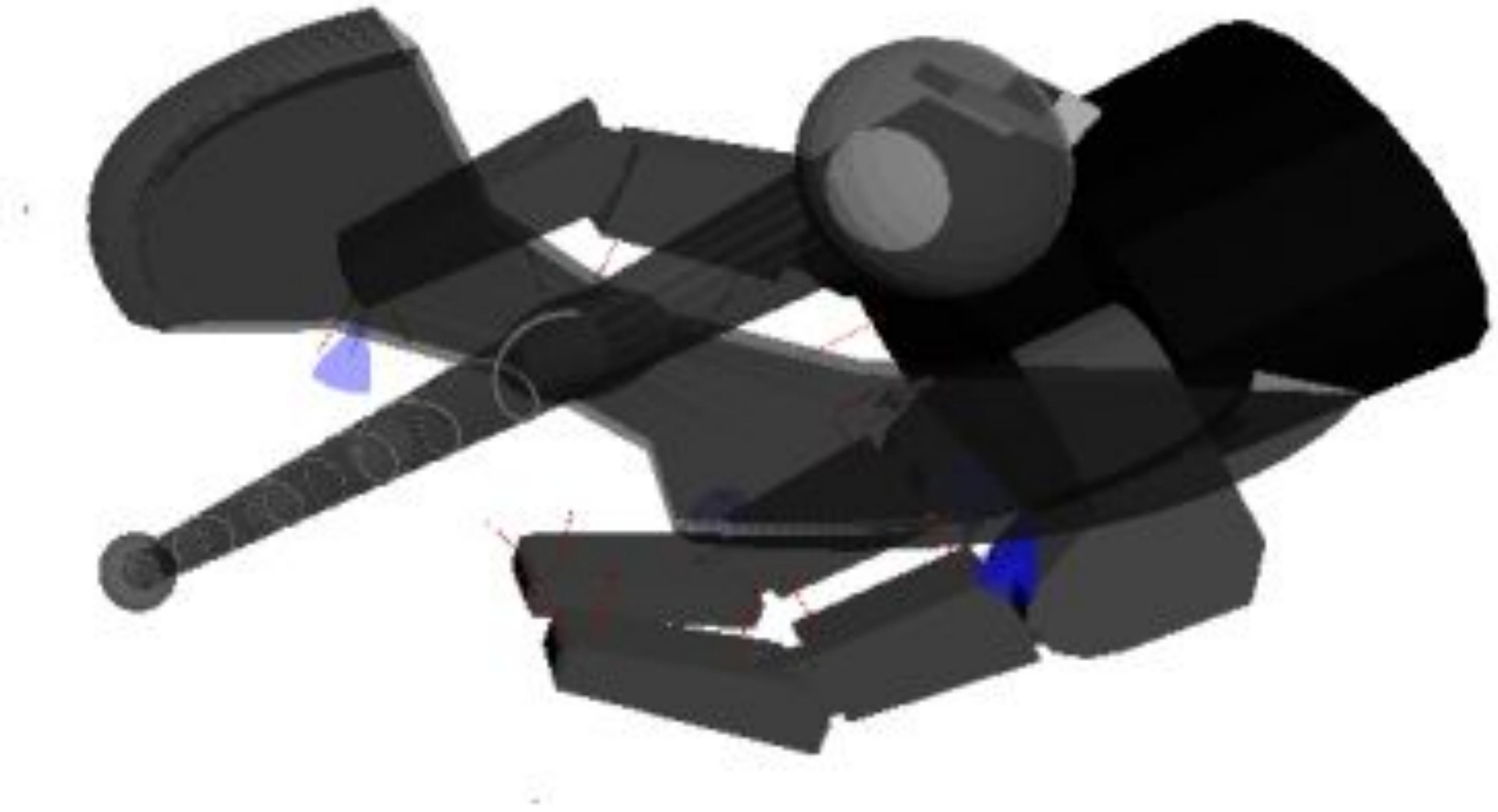


Picking grasps from Vision



Picking grasps from Vision





Presentations

2nd IWCMAR @
International Conference on
Robotics and Automation

This work produced:

- The substance of my MSc thesis work
- An early (submitted on 1st May) peer-reviewed accepted presentation at the *Second International Workshop of Computational Models of Affordance for Robotics* (IWCMAR) held in Montreal at ICRA 2019
- An accepted long abstract in the journal *Frontiers in Neurorobotics* about Computational Models of Affordance for Robotics, planning a submission by the 13th of October

frontiers
in Neurorobotics

Thank you!