TO KNOW OR TO SEE: FEW- AND ZERO-SHOT OBJECT PERCEPTION FOR ROBOTIC MANIPULATION

Prof. Dr. Rudolph Triebel

Department Perception and Cognition, Inst. for Robotics and Mechatronics (DLR)

Chair of Intelligent Robot Perception at Karlsruhe Institute of Technology (KIT)

Karlsruhe Institute of Technology

Motivation

Motivation

Motivation

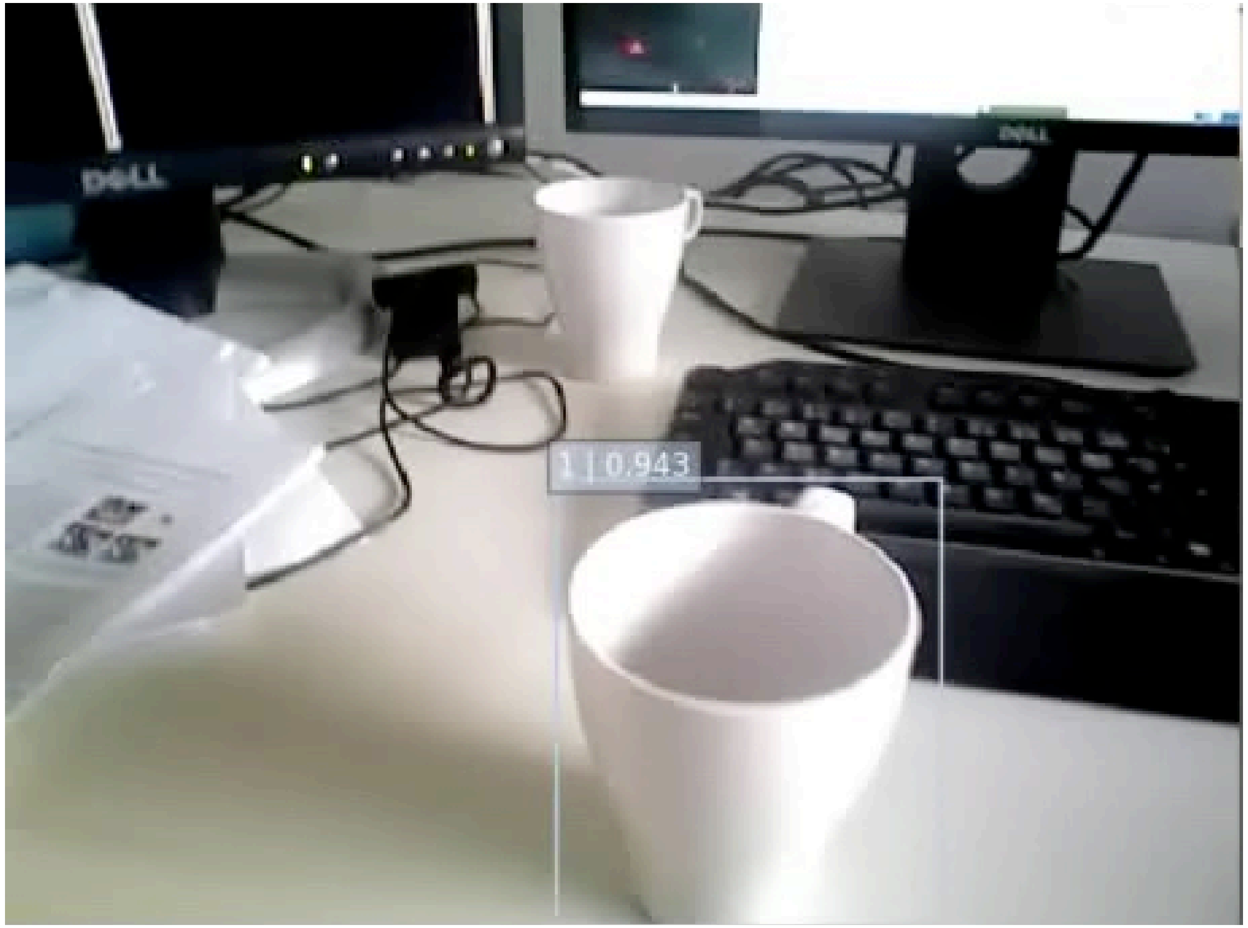
Motivation

Perception for Robotic Manipulation

3 main perception tasks for manipulation:

Object detection

- often involves segmentation
- adds *semantic* information
- requires the appearance (color, texture) of objects



Perception for Robotic Manipulation

3 main perception tasks for manipulation:

Object detection

↓

Object pose estimation

- retrieves the exact position and orientation in the camera (robot) frame
- requires the exact geometry of objects

Frame-by-frame

Sundermeyer, Marton, Durner, Brucker, Triebel: “Implicit 3D Orientation Learning for 6D Object Detection from RGB Images”, European Conf. on Computer Vision *(ECCV) 2018*

Tracking

Stoiber, Pfanne, Strobl, Triebel, Albu-Schäffer, “A Sparse Gaussian Approach to Region-Based 6DoF Object Tracking”, Asian Conference on Computer Vision *(ACCV) 2020*

Perception for Robotic Manipulation

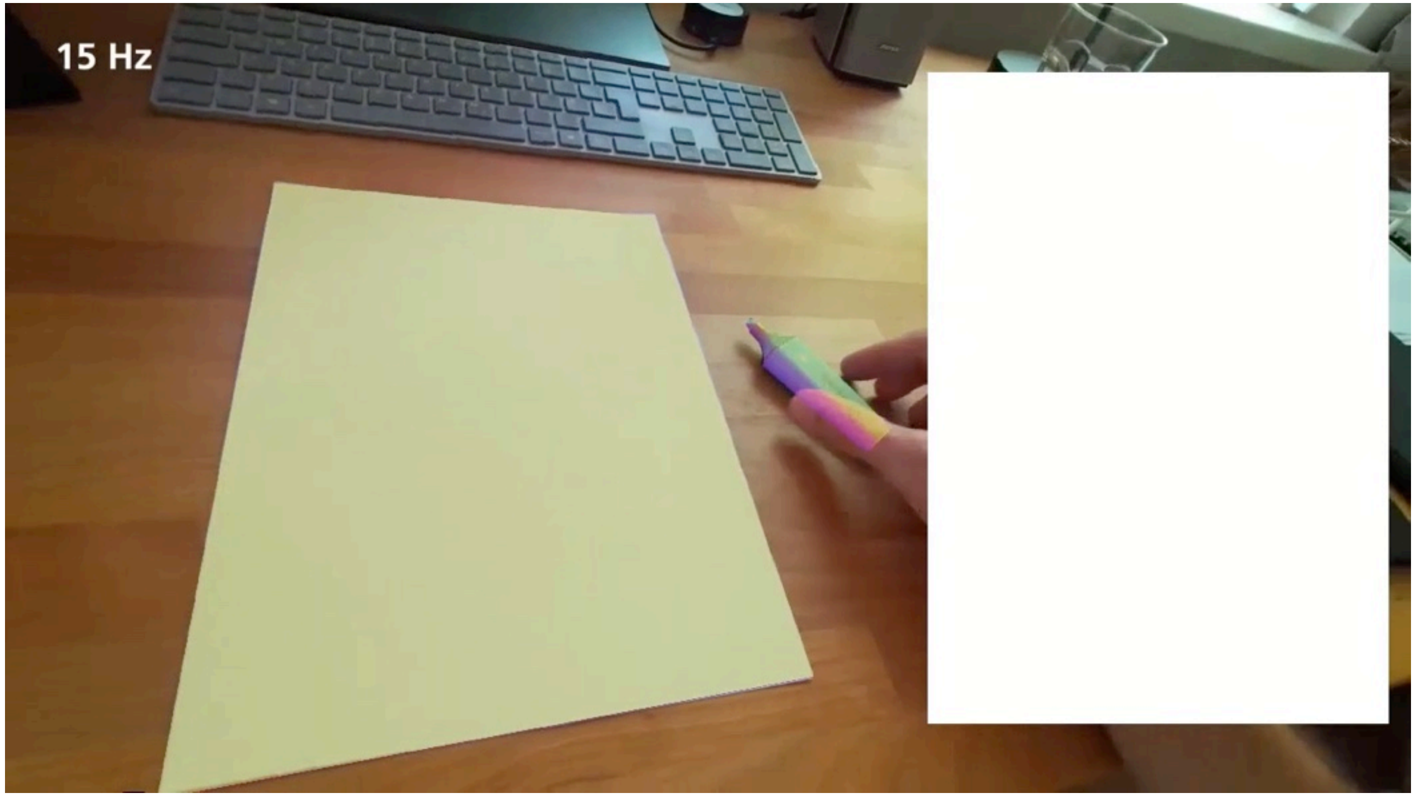
3 main perception tasks

Object detection

↓

Object pose estimation

- retrieves the exact position and orientation in the camera (robot) frame
- requires the exact geometry of objects



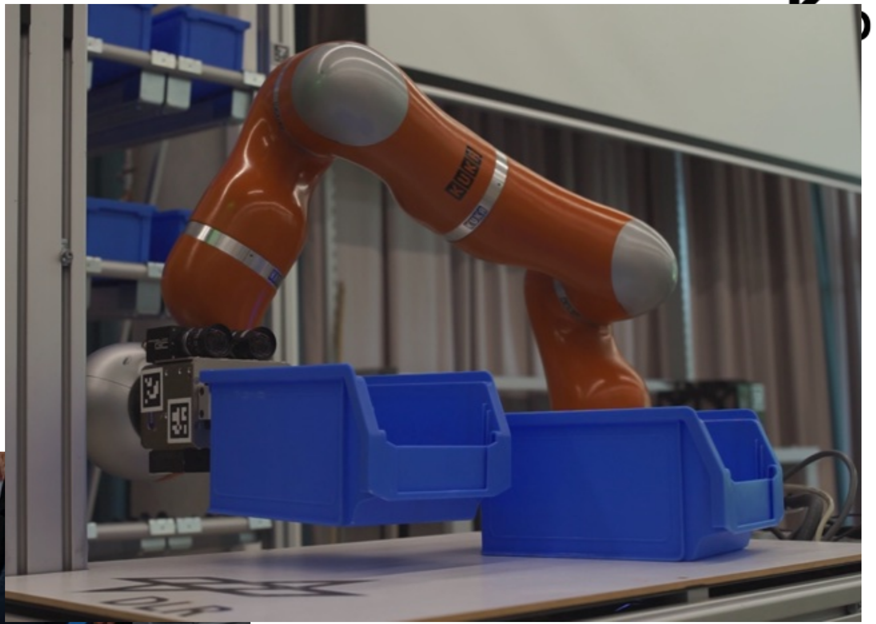
Stoiber, Elsayed, Reichert, Steidle, Lee, Triebel, “Fusing Visual Appearance and Geometry for Multi-Modal 6DoF Object Tracking”, IROS *2023*

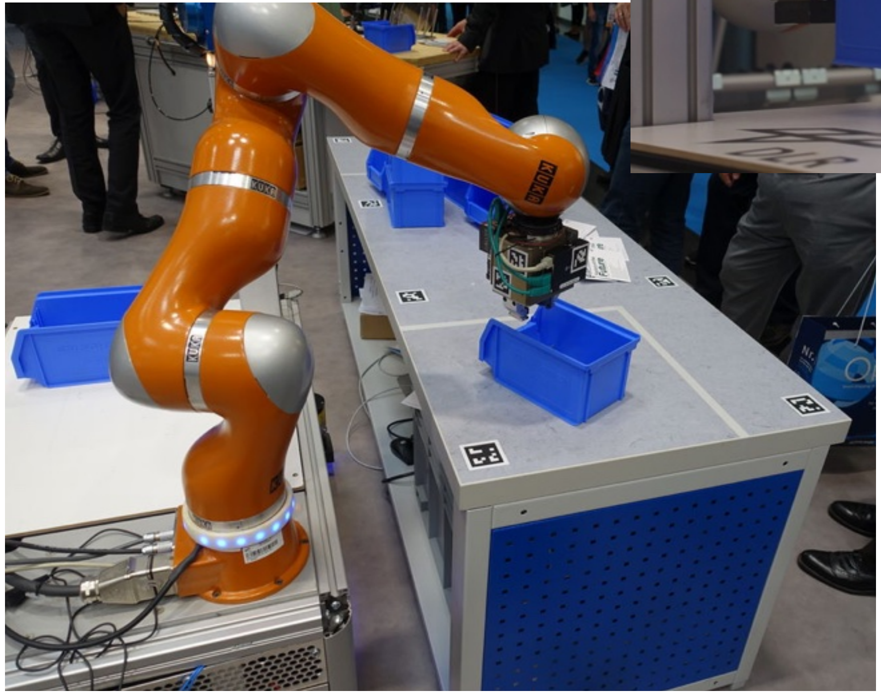
Perception for Robotic Manipulation

3 main perception tasks for manipulation:

Object detection → **Object pose estimation** → **Grasp detection**

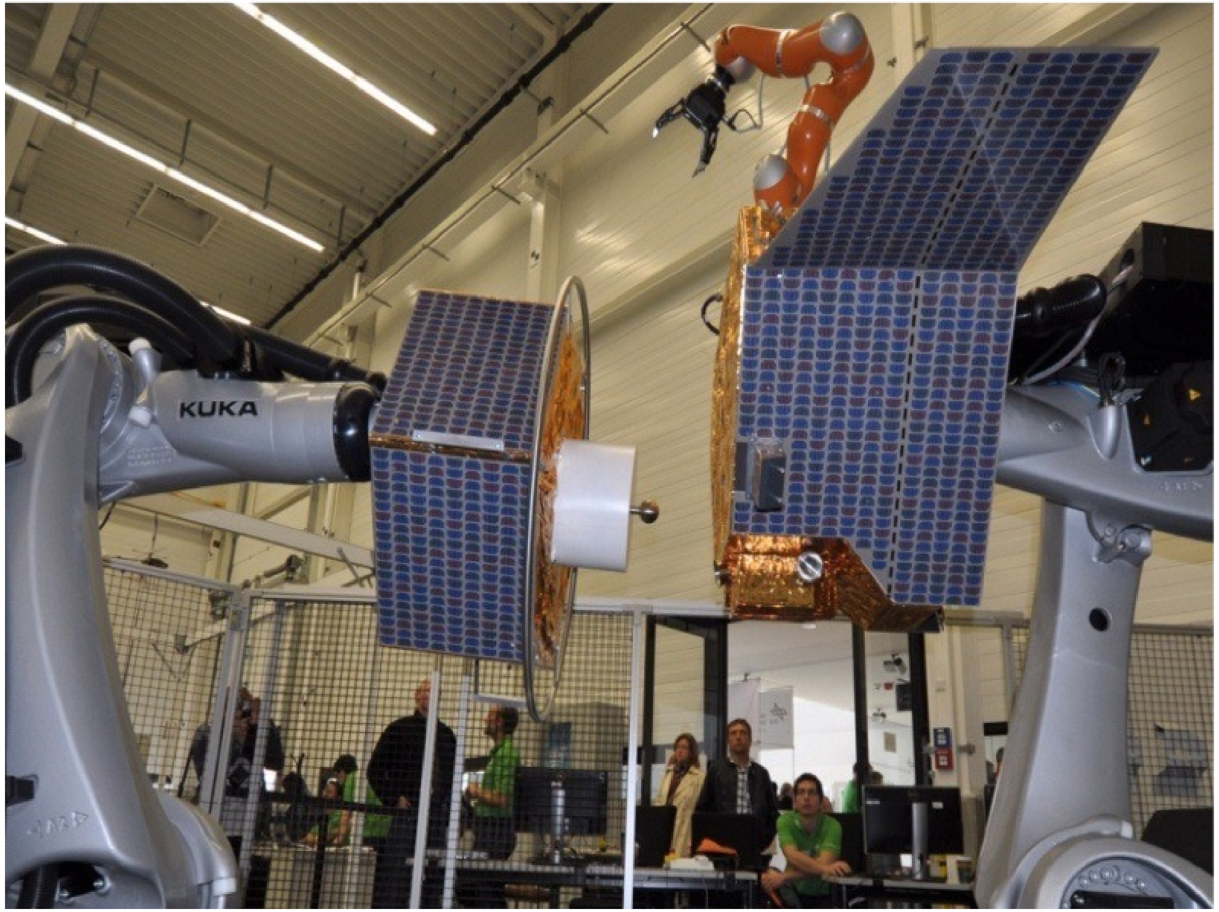
- finds the pose of the robotic gripper for a grasp
- requires the exact geometry and kinematics of the gripper



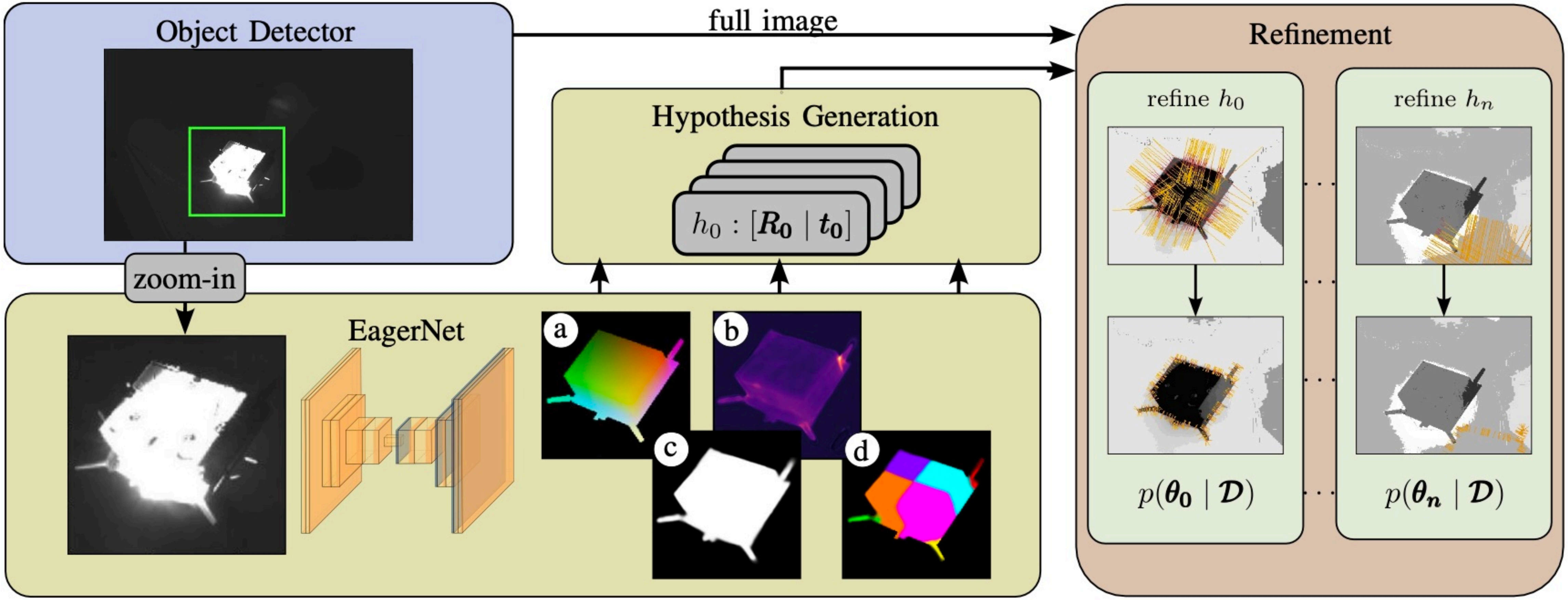


Example: Satellite Pose Estimation for On-Orbit Servicing

- Aim: find the relative 6DoF pose between servicer and target
- Challenges:
  - very difficult lighting conditions
  - inaccurate 3D models
  - insufficient training data

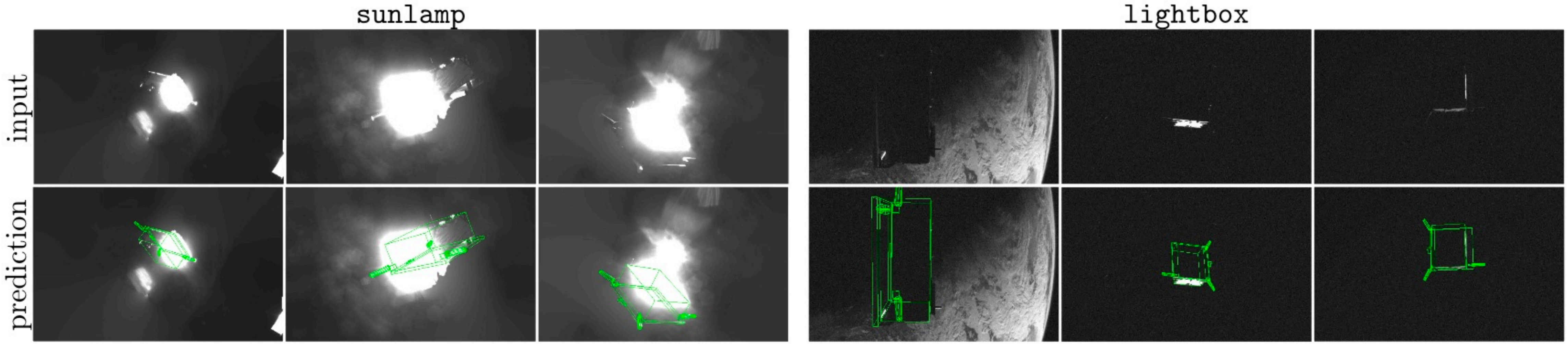


Example: Satellite Pose Estimation for On-Orbit Servicing



Ulmer, Durner, Sundermeyer, Stoiber, Triebel, "6D Object Pose Estimation from Approximate 3D Models for Orbital Robotics", IROS *2023*

Example: Satellite Pose Estimation for On-Orbit Servicing



Qualitative Results on SPEED+ benchmark data set

- State-of-the-art performance on SPEED+ data set (post-mortem)
- In category lightbox, EagerNet is best in all categories
- In sunlamp, best in rotation error
- 3D model not necessarily perfect

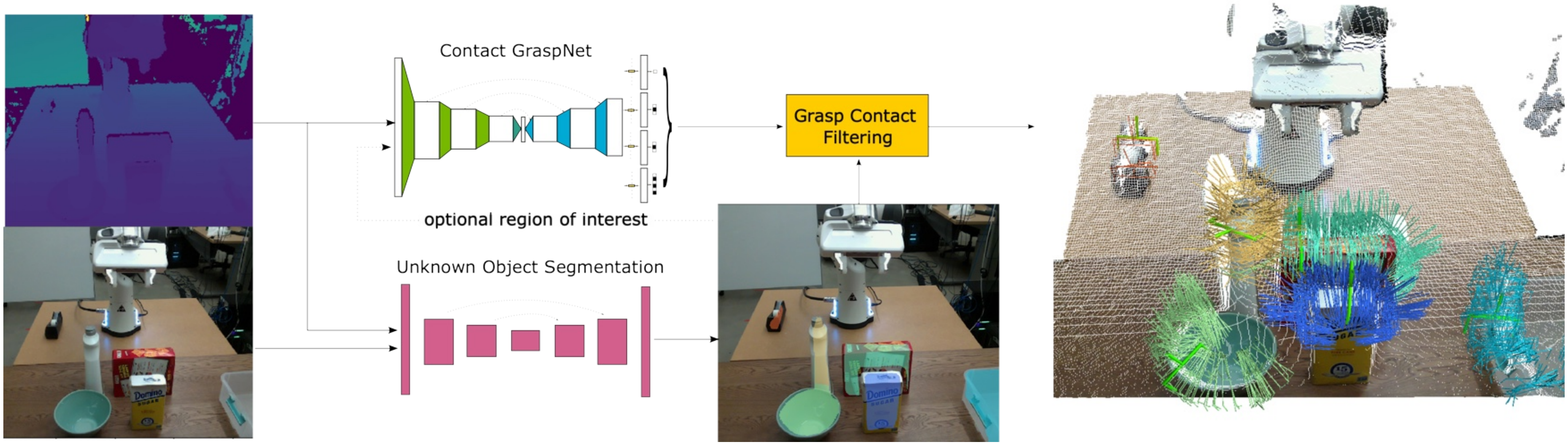
	lightbox			sunlamp			
	e_t	e_R	e_{pose}	e_t	e_R	e_{pose}	μ
lava1302 [6]	0.0464	0.1163	0.1627	**0.0069**	0.0476	**0.0545**	0.1086
prow	0.0196	0.0944	0.1140	0.0133	0.0840	0.0972	0.1056
VPU [5]	0.0215	0.0799	0.1014	0.0118	0.0493	0.0612	0.0813
TangoUnchained	0.0161	0.0519	0.0679	0.0150	0.0750	0.0900	0.0790
haoranhuang_njust	0.0138	0.0515	0.0652	0.0110	0.0479	0.0589	0.0621
EagerNet (ours)	**0.0085**	**0.0305**	**0.0390**	0.0126	**0.0465**	0.0590	**0.0490**

Ulmer, Durner, Sundermeyer, Stoiber, Triebel, “6D Object Pose Estimation from Approximate 3D Models for Orbital Robotics”, IROS *2023*

What is Known and What is Seen

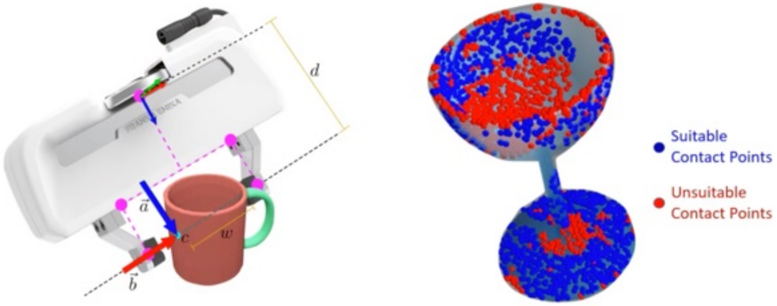
- Known vs. unknown objects:
  - Objects contained in training data?
- Seen vs. unseen objects:
  - CAD Model of object given during inference?
- Zero-shot vs. few-shot
  - How many samples are required for (re-)training?
- Model-based vs. model-free
  - CAD model given beforehand?

Example: Learning to Grasp Unknown Objects



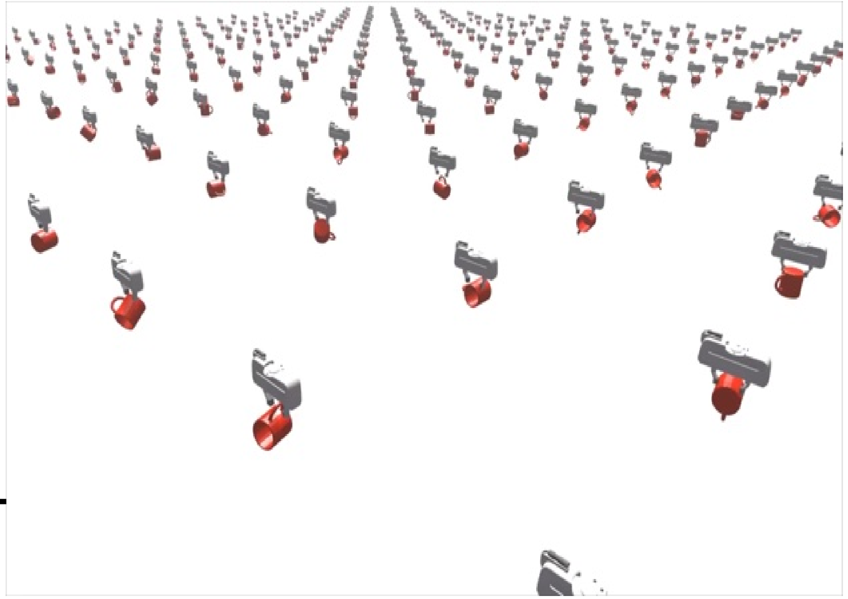
Main ideas:

- Use a representation of grasp contact points for 2-finger robotic grippers
- Train a network to predict feasible contact points from a large simulated training data set
- Combine this with unknown-object segmentation to mask out objects

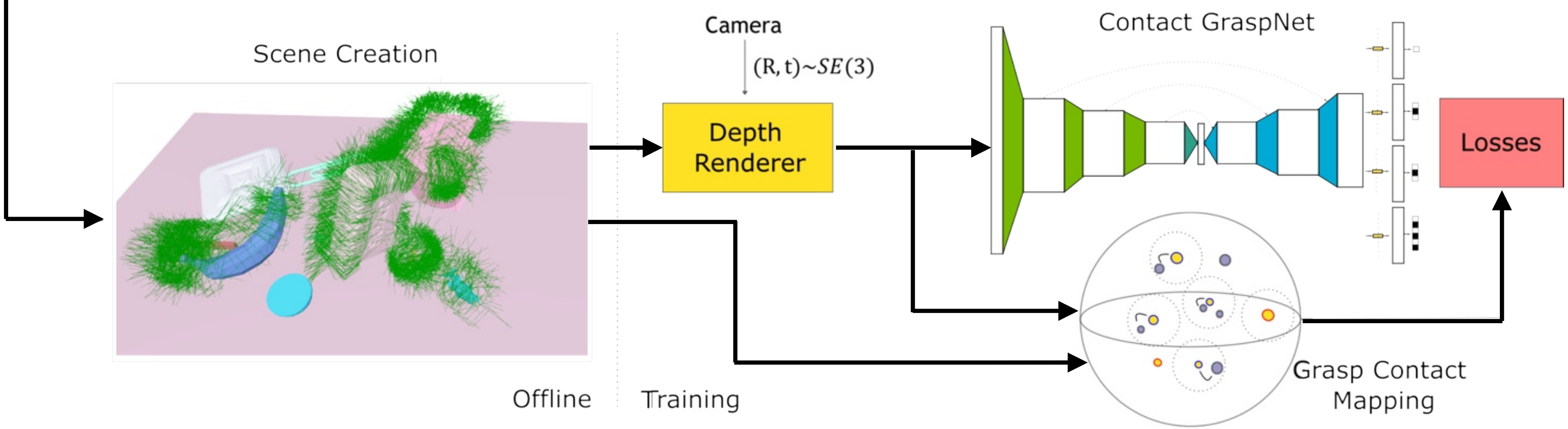


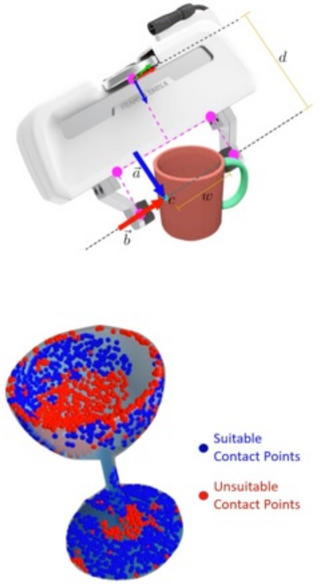
Sundermeyer, Mousavian, Triebel, Fox: “Contact-GraspNet: Efficient 6-DoF Grasp Generation in Cluttered Scenes”, IEEE Intern. Conf. *on Robotics and Automation (ICRA) 2021*

Example: Learning to Grasp Unknown Objects

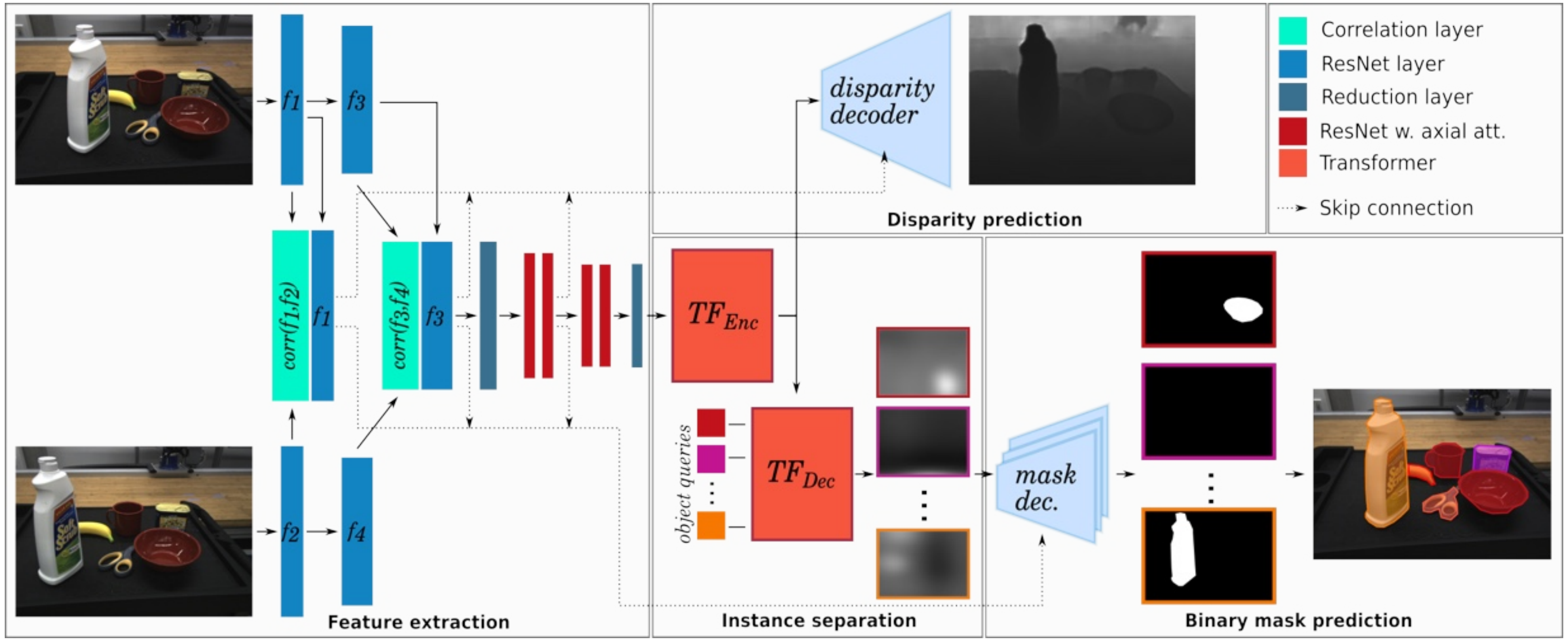


- generate 17.7 m grasps from physics simulation (ACRONYM)
- generate synthetic scenes with objects and stable grasps
- learn a model that maps grasps to contact points





Example: Learning to Grasp Unknown Objects

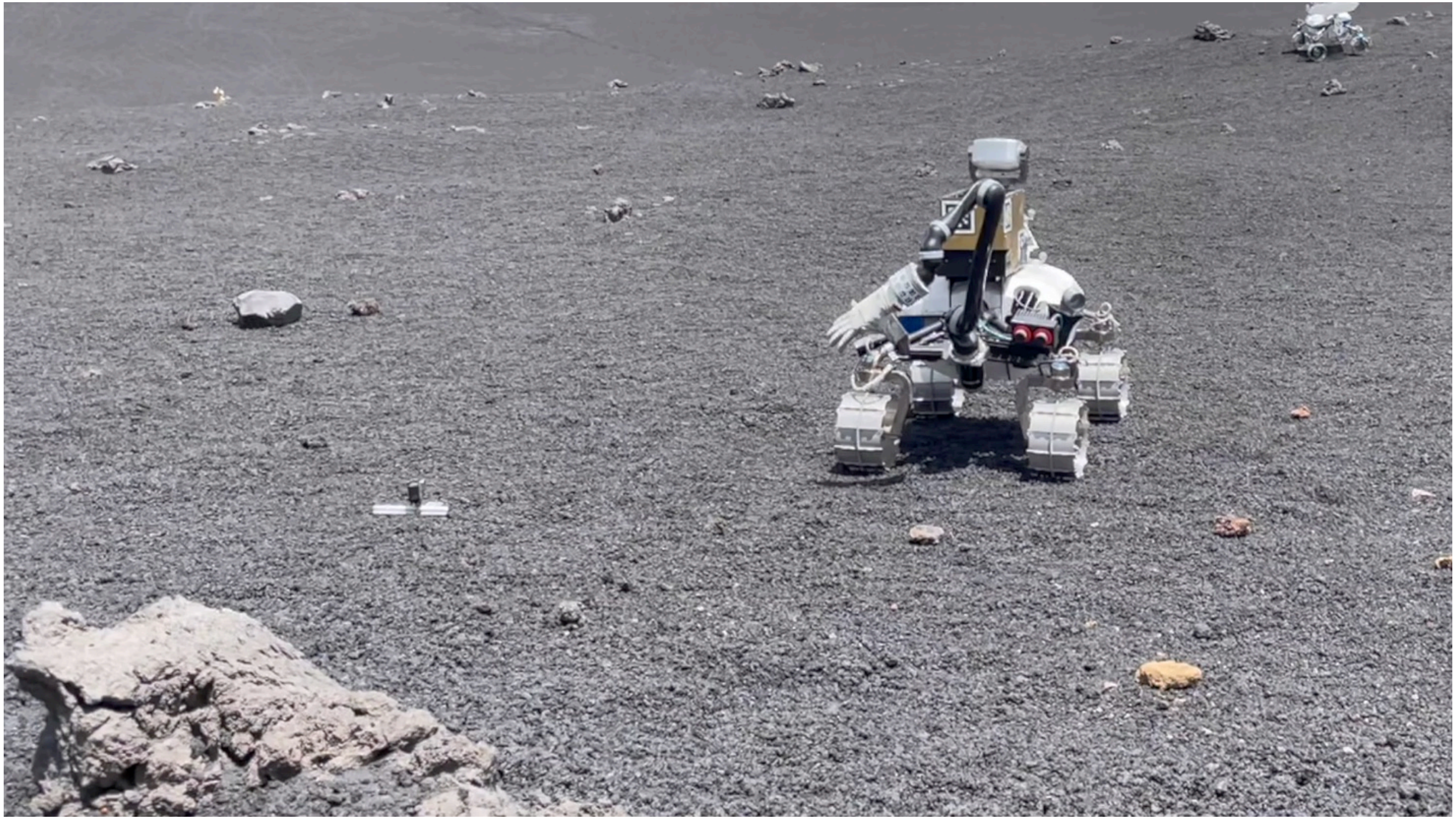


Learning to segment unknown objects from stereo images using INSTR

- ContactGraspNet operates on entire scenes, not objects
- To manipulate objects, we need to segment them
  - ➡ INSTR for stereo-based object segmentation
- Then, segments are overlaid with the detected grasps

Durner, Boerdijk, Sundermeyer, Friedl, Marton, Triebel: “Unknown Object Segmentation from Stereo Images ”, IEEE Intern. Conference on Intelligent Robots and Systems (IROS), *2021*

Grasping Unknown Objects in the Wild

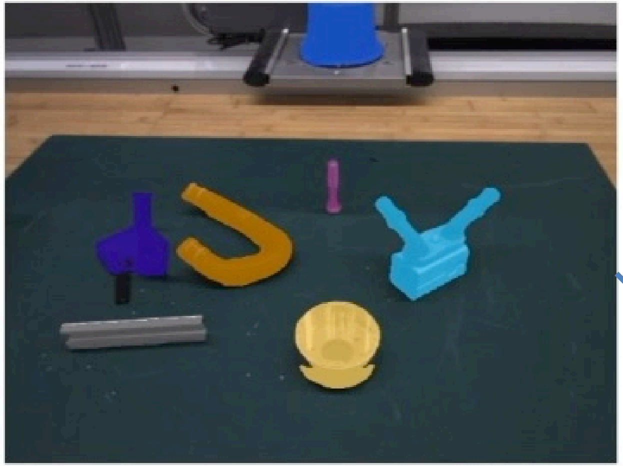


Durner, Boerdijk, Sundermeyer, Friedl, Marton, Triebel: "Unknown Object Segmentation from Stereo Images ", IEEE Intern. Conference on Intelligent Robots and Systems (IROS), *2021*

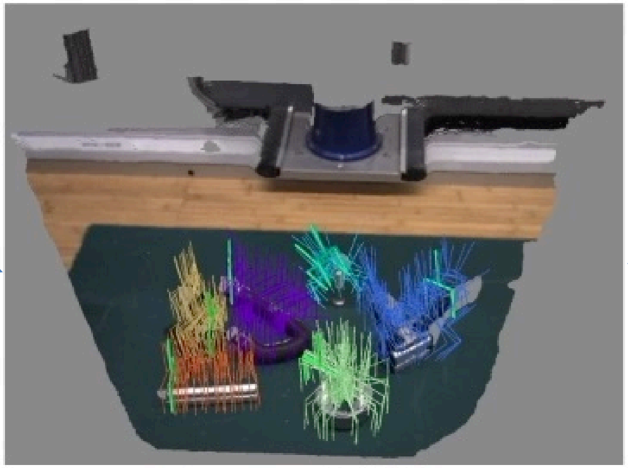
Grasping Unknown Objects of Difficult Shapes

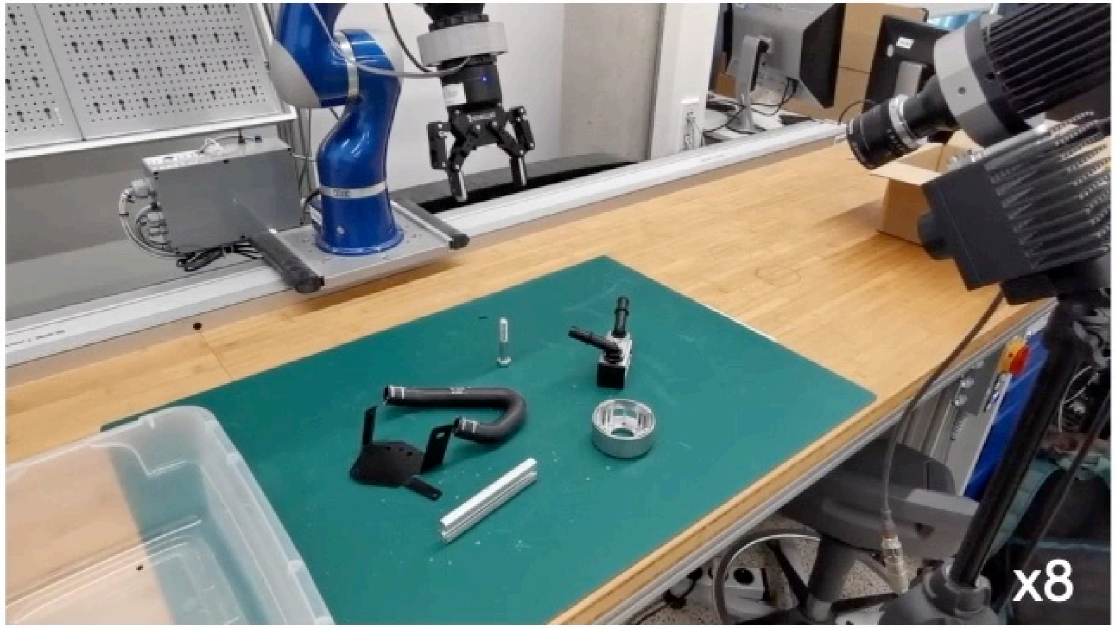
Image + Point cloud

Segmentation



Grasp detection

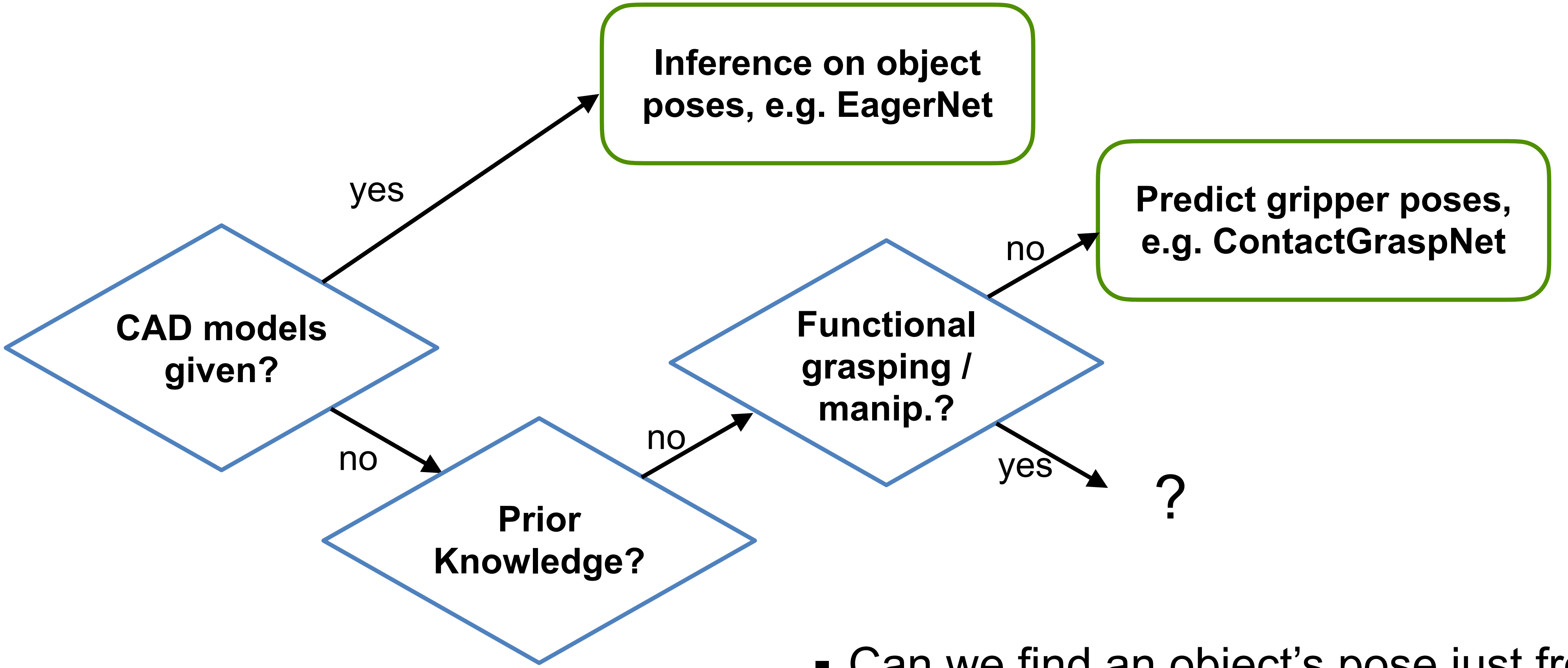




Grasp execution

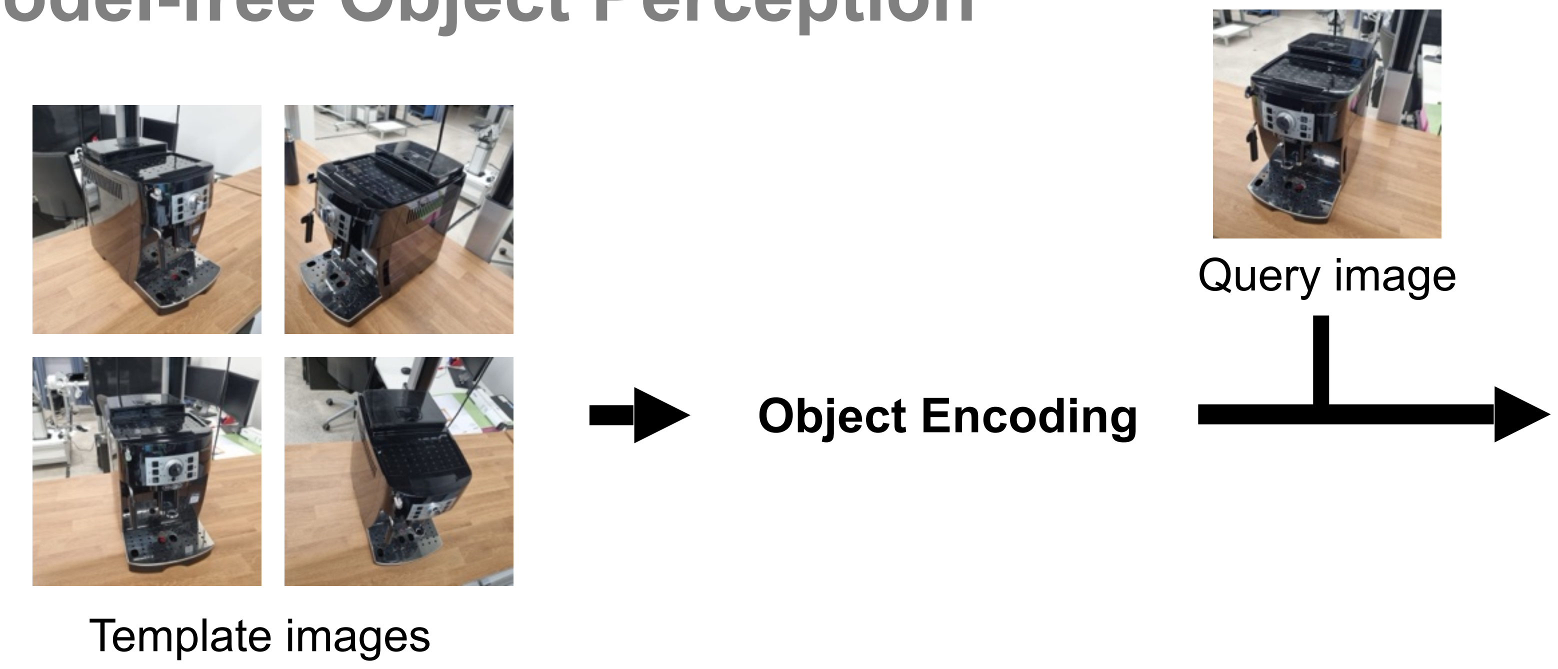
Sundermeyer, Mousavian, Triebel, Fox: “Contact-GraspNet: Efficient 6-DoF Grasp Generation in Cluttered Scenes”, Intern. Conf. *on Robotics and Automation (ICRA) 2021*

Perception Based on Known or Unseen Objects



- Can we find an object's pose just from images?
- Can we do this for new objects without retraining?

Model-free Object Perception



- **Detection**
- **Segmentation**
- **6DoF Pose Estimation**
- **novel view synthesis**
- **…**

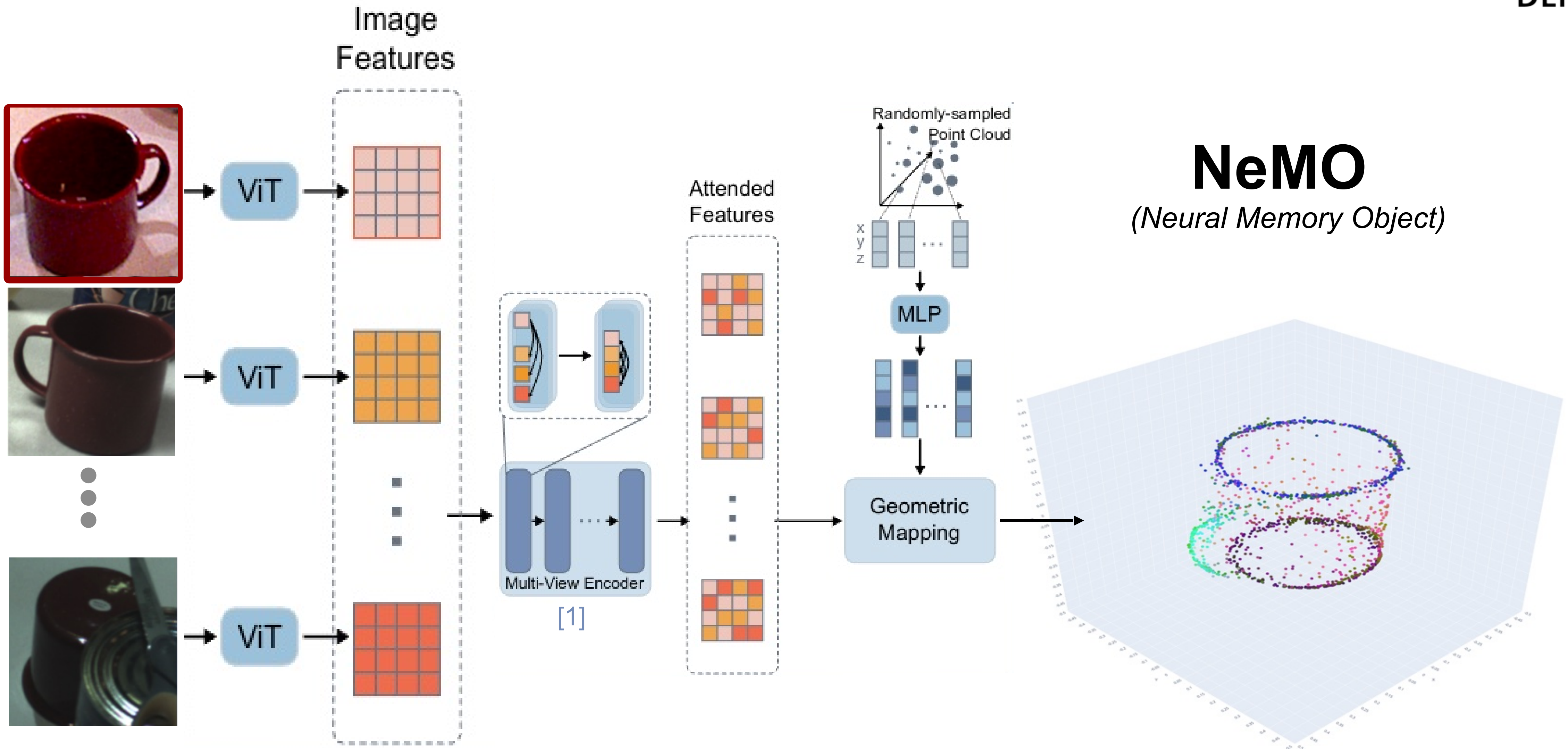
- Collect a set of template images
- Generate an object representation
- Perform inference based on a query image

→ No CAD Model needed

→ Train a single network once, separate object information from network weights

→ No retraining / finetuning

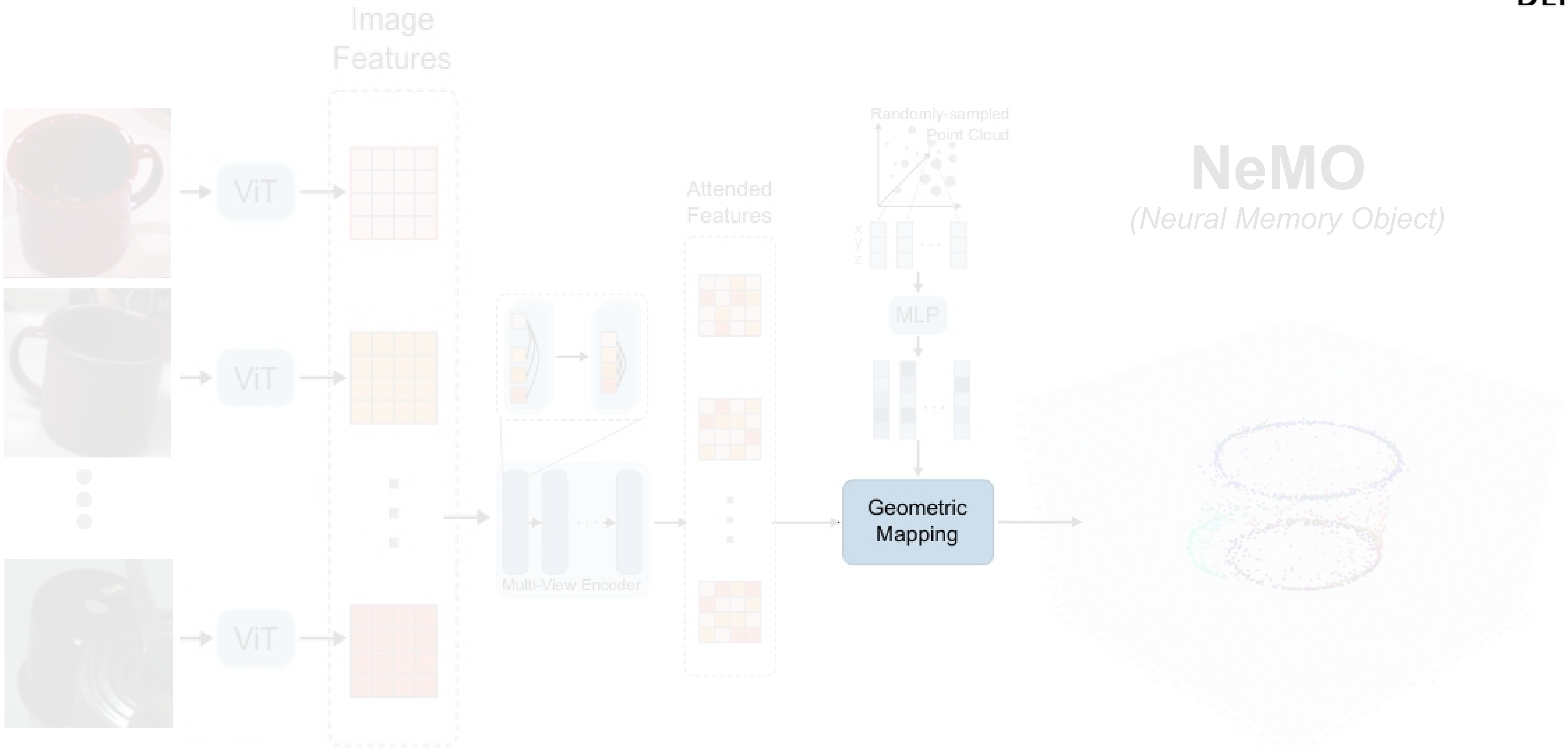
Encoder



[1] Jiang, Jiang, Zhao, Huang, „LEAP: Liberate Sparse-View 3D Modeling from Camera Poses“, in Intern. Conf. on Representation Learning (ICLR), 2024

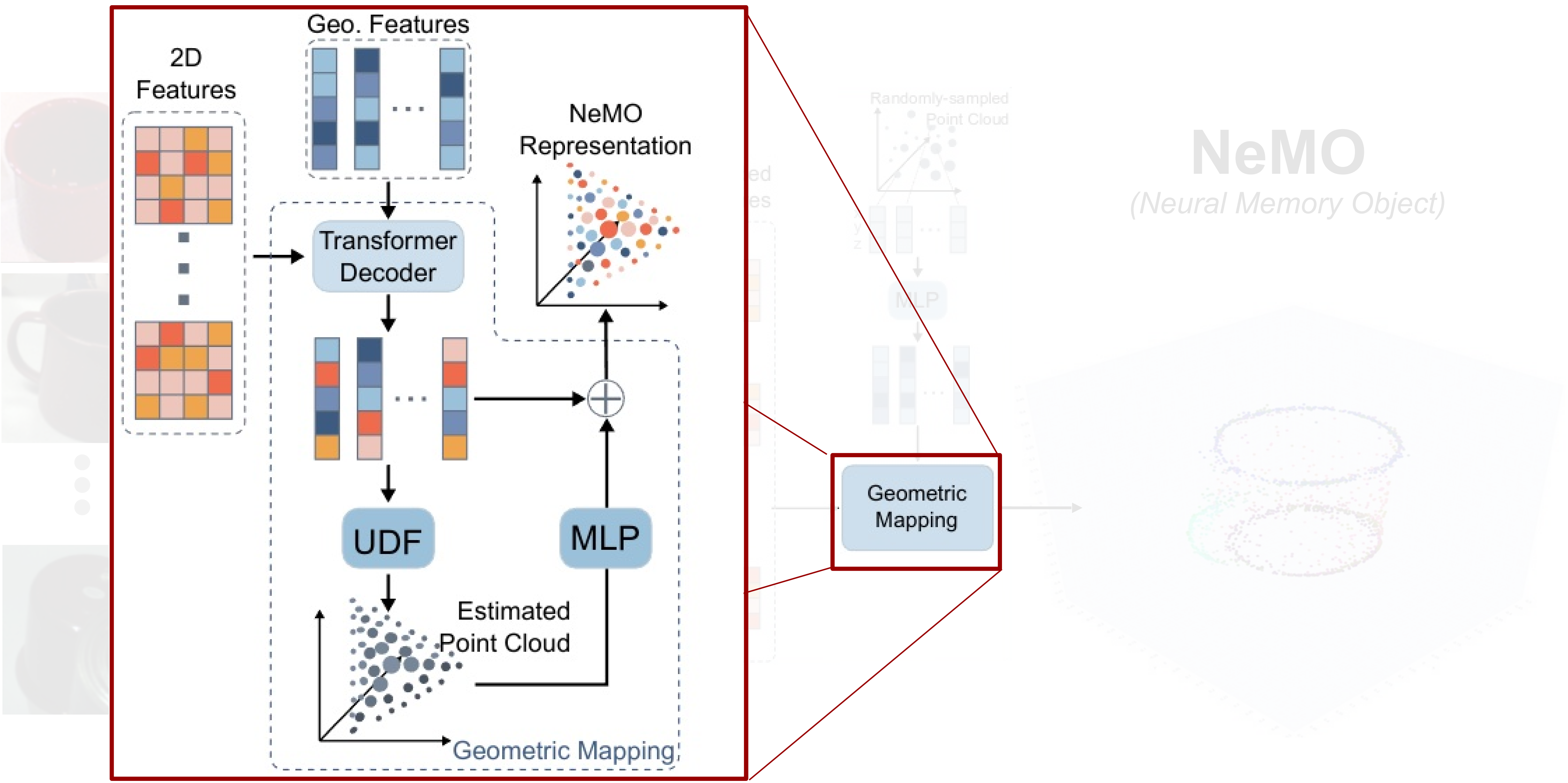
Jung, Klüpfel, Triebel, Durner: “Representation of Template Views for Few-Shot Perception”, *International Conference on 3D Vision* (3DV) 2026 (to appear)

Encoder



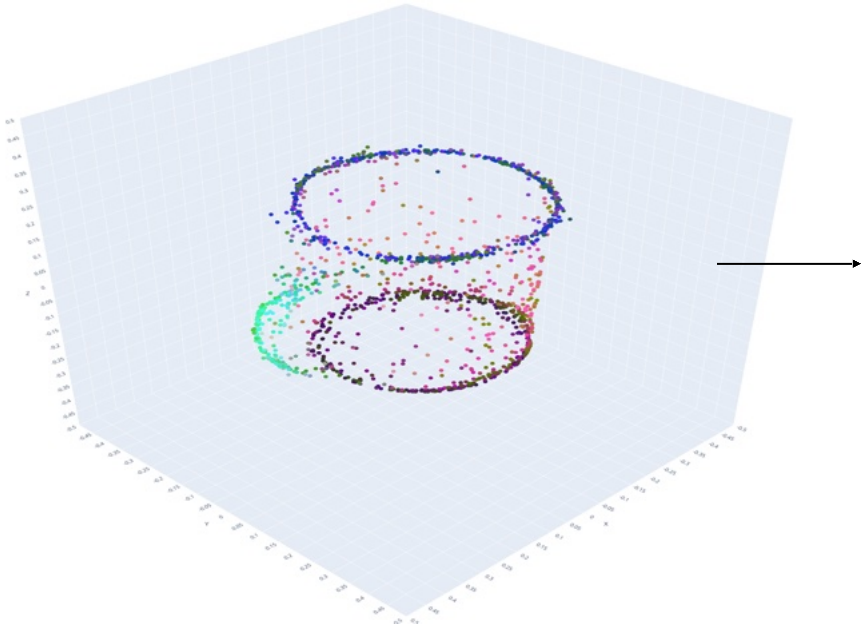
Jung, Klüpfel, Triebel, Durner: “Representation of Template Views for Few-Shot Perception”, *International Conference on 3D Vision* (3DV) 2026 (to appear)

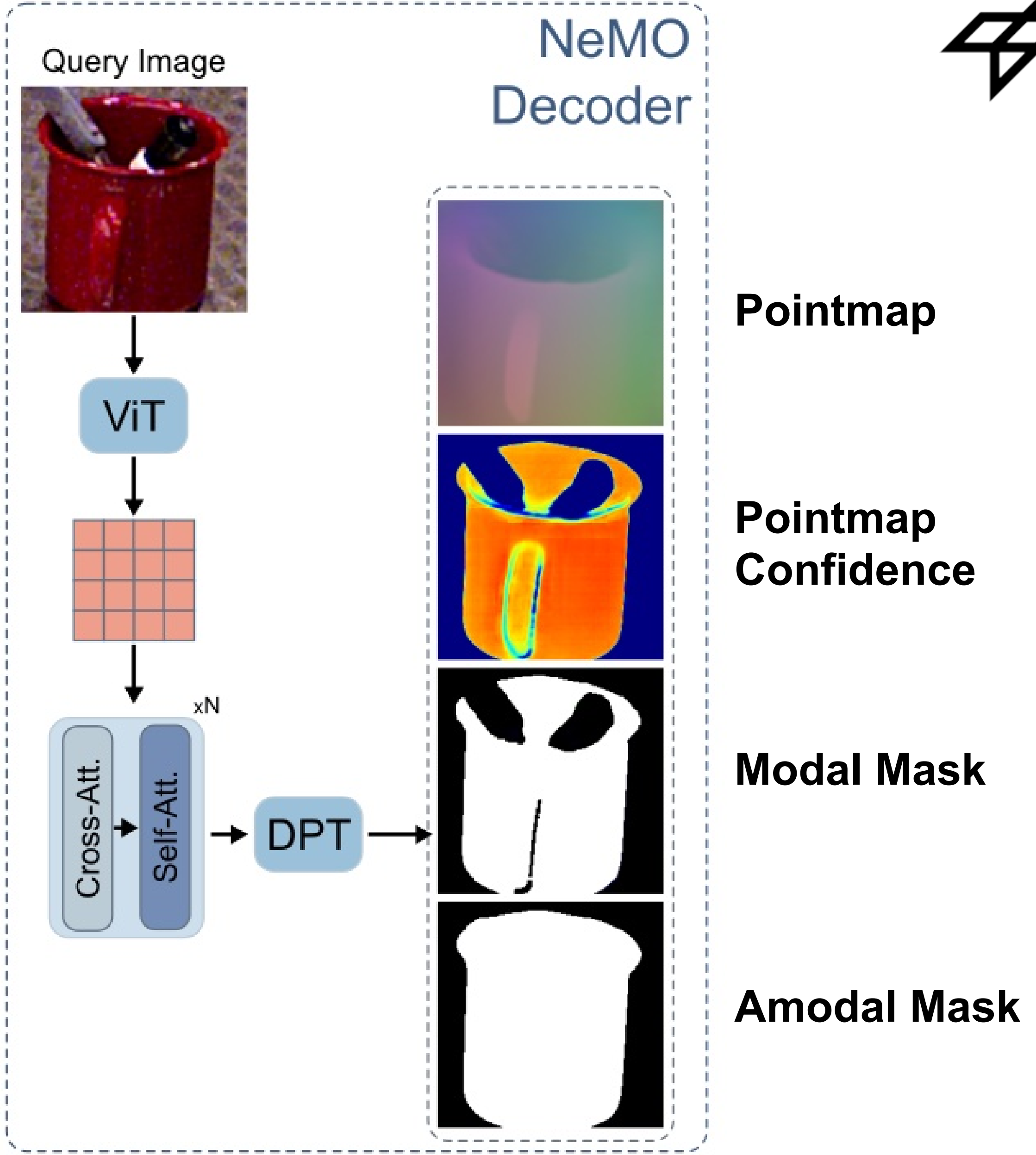
Encoder



Jung, Klüpfel, Triebel, Durner: “Representation of Template Views for Few-Shot Perception”, *International Conference on 3D Vision* (3DV) 2026 (to appear)

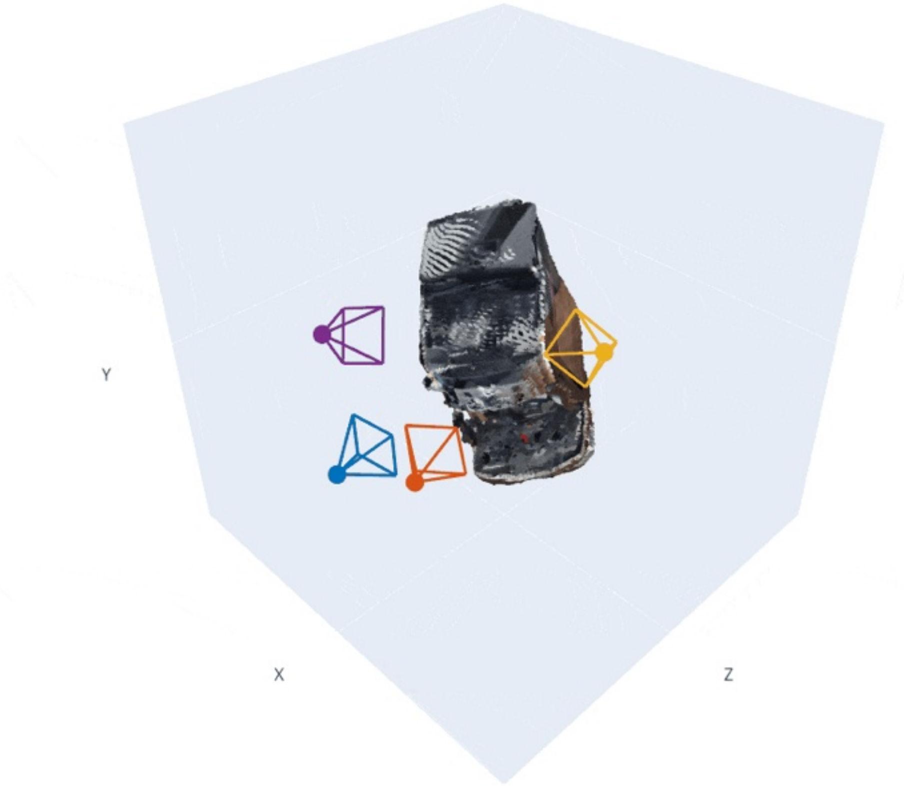
Decoder

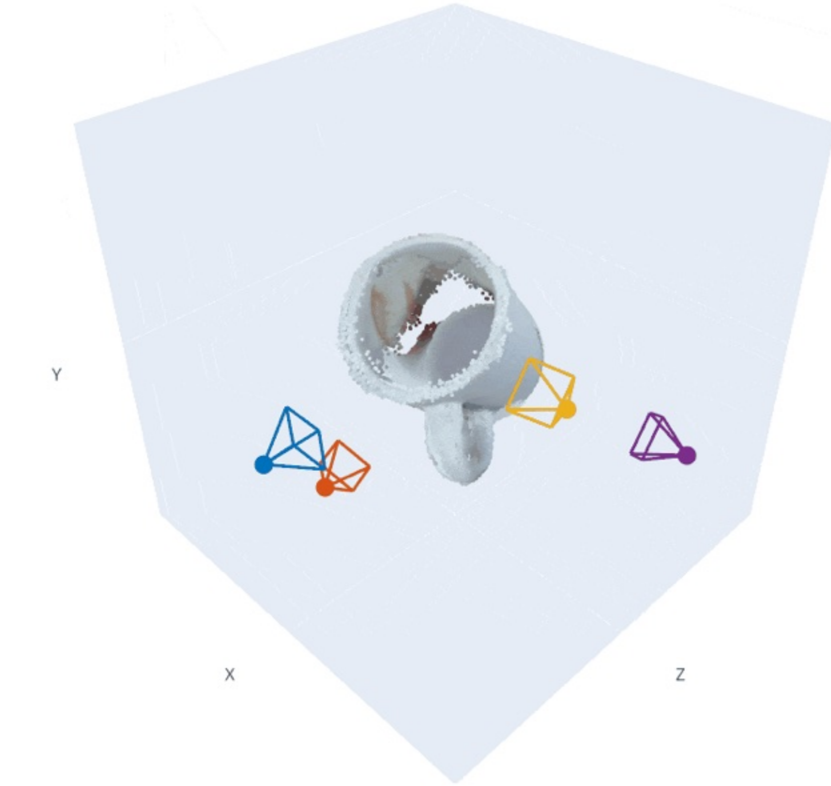




Jung, Klüpfel, Triebel, Durner: “Representation of Template Views for Few-Shot Perception”, *International Conference on 3D Vision* (3DV) 2026 (to appear)

Qualitative Examples





Jung, Klüpfel, Triebel, Durner: “Representation of Template Views for Few-Shot Perception”, *International Conference on 3D Vision* (3DV) 2026 (to appear)

Results: Model-Free Detection and Pose Estimation

Method	HOPEv2	HANDAL
CNOS (SAM) - Static onboarding [41]	0.345	–
dounseen-SAM-CTL [16]	0.380	–
GFreeDet-FastSAM [33]	0.364	0.255
GFreeDet-SAM [33]	<u>0.384</u>	<u>0.264</u>
Ours	**0.411**	**0.273**

Detection results

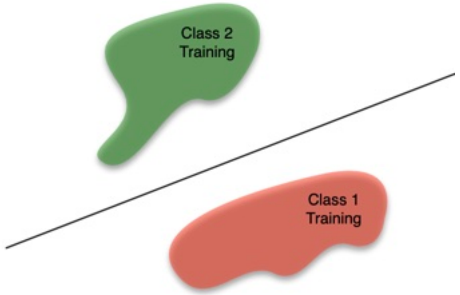
Method	Detections	HOPEv2	HANDAL
OPFormer†	CNOS [41]	**0.335**	0.204
Ours	CNOS [41]	0.307	–
Ours	GFreeDet-FastSAM [33]	<u>0.329</u>	<u>0.213</u>
Ours	NeMO	0.302	**0.235**

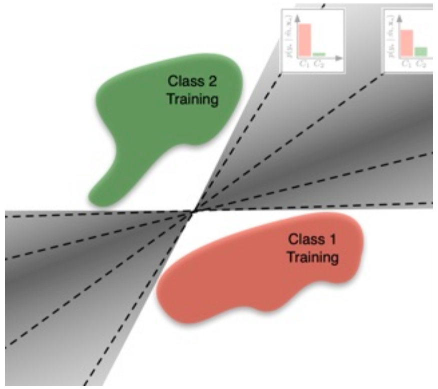
Pose estimation results

Qualitative results

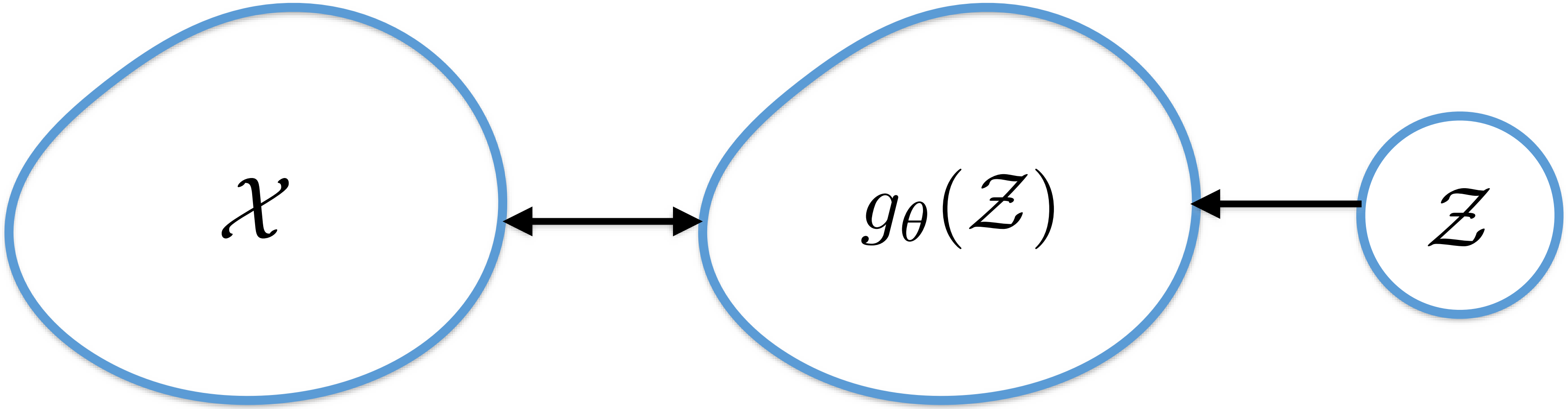
Jung, Klüpfel, Triebel, Durner: "Representation of Template Views for Few-Shot Perception", *International Conference on 3D Vision* (3DV) 2026 (to appear)

Discriminative vs Generative Models





- Discriminative models learn to distinguish known classes
- They have some difficulties detecting OOD data
- Generative models learn a function g that maps from a latent space $\mathcal{Z}$ to the data space $\mathcal{X}$
- They can generate samples from the latent space $\mathcal{Z}$ and apply g



Generative Models: Advantages and Challenges

DLR

Advantages

- good test of representing high-dimensional data
- useful for reinforcement learning
- can be trained with missing data, e.g. semi-supervised learning
- work with **multi-modal output**, e.g. predicting the next frame in a video

Challenges

- need good hyper parameters during training (architecture, training objective, regularisation, …)
- need a similarity between generated and observed data, e.g.:
  - invert generator
  - learn similarity (e.g. discriminator)
- how to find a good dimensionality of the latent space?

Recent Generative Models

- Generative Adversarial Networks (GANs)
- Variational Autoencoders
- Generative Pretrained Transformers (GPT)
- Autoregressive Models
- Normalising Flow
- **Diffusion models**
- VLMs, VLAs
- …

Diffusion-based Zero-Shot Instance Segmentation

Object Conditions

Input Image

Ulmer, Boerdijk, Triebel, Durner: “Conditional Latent Diffusion Models for Zero-Shot Instance Segmentation”, *International Conference on Computer Vision* (ICCV) 2025

Diffusion-based Zero-Shot Instance Segmentation

- Idea: Sequentially generate instance segmentations with diffusion
- Guide sampling by conditioning the reverse process on target objects

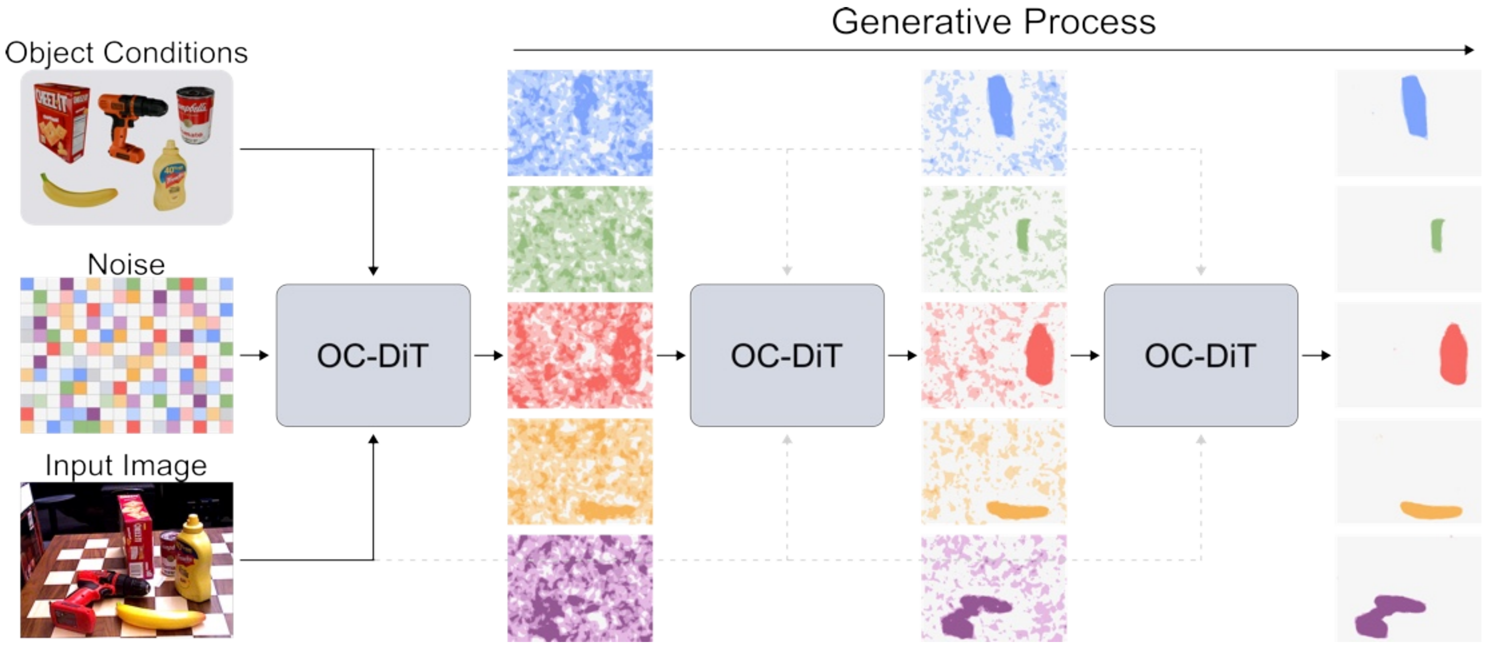
Object Conditions

Input Image

Ulmer, Boerdijk, Triebel, Durner: “Conditional Latent Diffusion Models for Zero-Shot Instance Segmentation”, *International Conference on Computer Vision* (ICCV) 2025

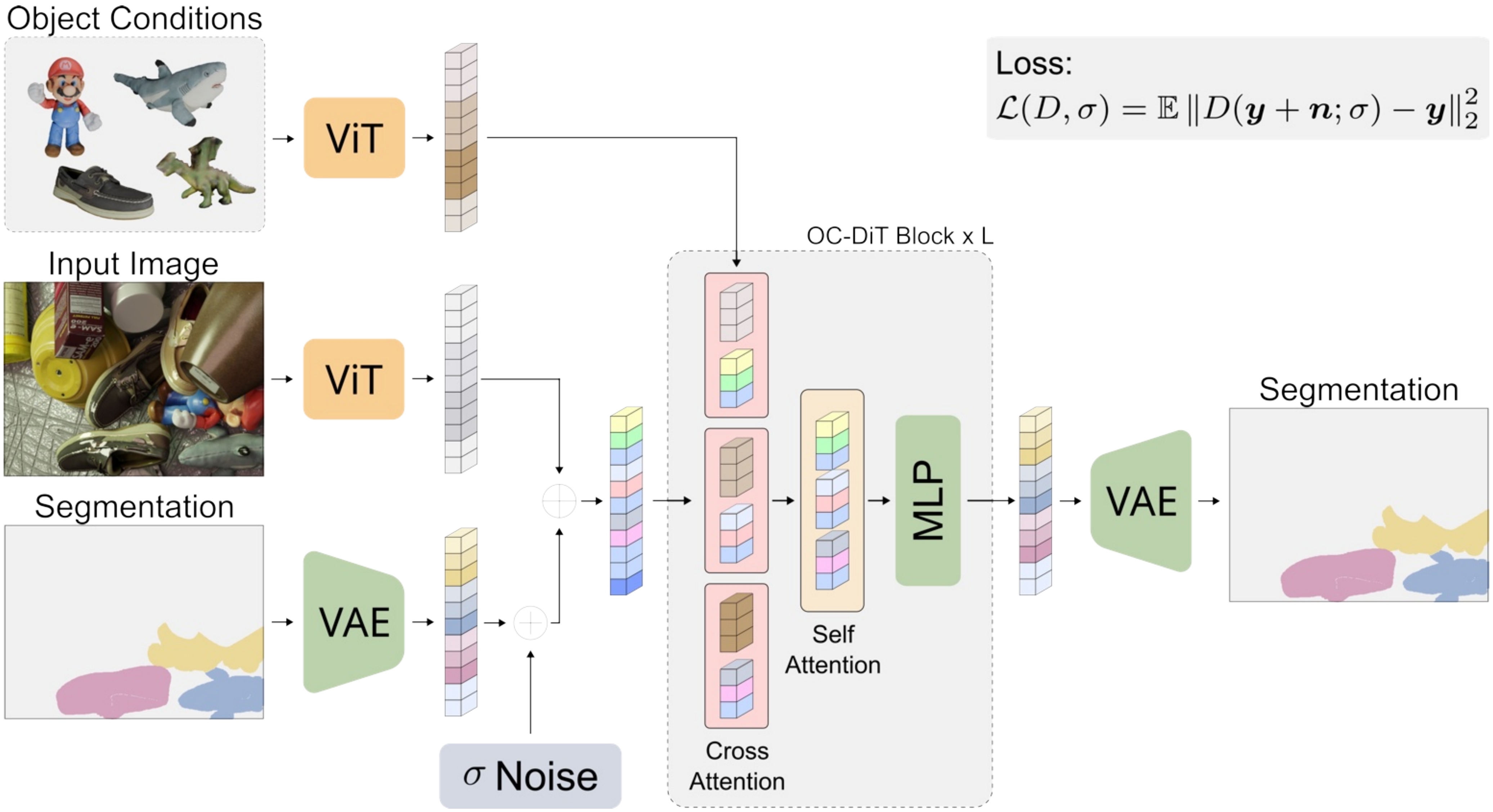
Diffusion-based Zero-Shot Instance Segmentation

- Idea: Sequentially generate instance segmentations with diffusion
- Guide sampling by conditioning the reverse process on target objects
- Latent Diffusion: Use VAE to shape latent space statistics
- Why Diffusion? Strong scaling, effective against object ambiguities



Ulmer, Boerdijk, Triebel, Durner: "Conditional Latent Diffusion Models for Zero-Shot Instance Segmentation", *International Conference on Computer Vision* (ICCV) 2025

Conditional Latent Diffusion: Architecture



Loss:

$$\mathcal{L}(D,\sigma) = \mathbb{E}\,\|D(\boldsymbol{y}+\boldsymbol{n};\sigma) - \boldsymbol{y}\|_2^2$$

Ulmer, Boerdijk, Triebel, Durner: “Conditional Latent Diffusion Models for Zero-Shot Instance Segmentation”, *International Conference on Computer Vision* (ICCV) 2025

Results: Model-based 2D Segmentation of Unseen Objects

- Evaluation on BOP benchmark data set on model-based 2D segmentation
- At test time RGB images are received of objects that are not in training data
- We did not fine-tune on target data
- Conditioned on GTH objects

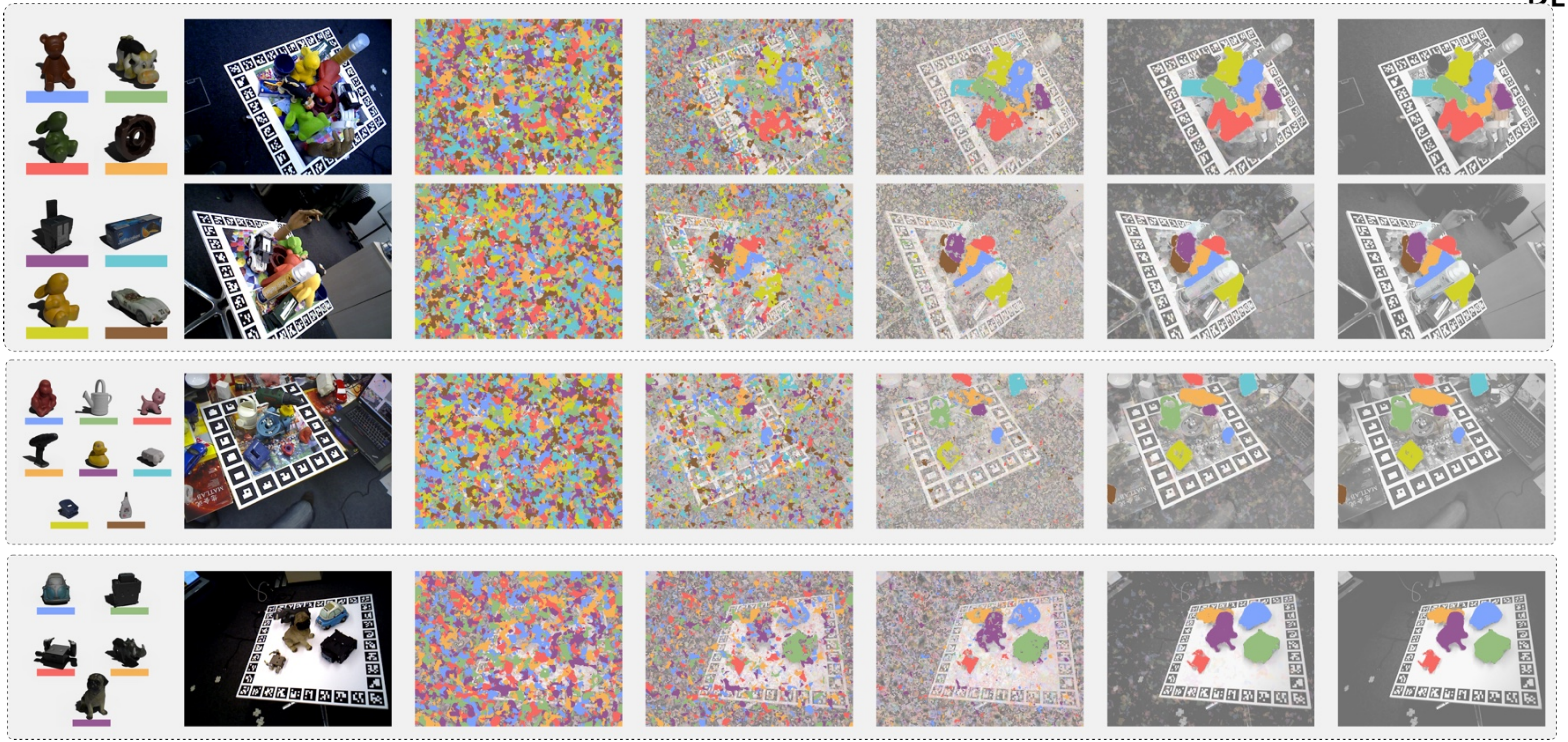
Average Precision

AP	YCBV	TUDL	LMO	HB
CNOS [29]	59.9	48.0	39.7	51.1
SAM6D [24]	60.5	56.9	46.0	59.3
NIDS [27]	65.0	55.6	43.9	62.0
MUSE	67.2	56.5	**47.8**	59.7
LDSeg	64.7	58.7	**47.8**	**62.2**
Ours `coarse`	68.6	32.5	29.6	52.4
Ours `refined`	**71.7**	**59.4**	40.1	61.5

- AP metric: mean of precision values at Intersection over Union (IoU) thresholds ranging from 50% to 95% with a step size of 5%
- Strong results on VXBV, TUDL, HB
- The `refined` model uses bounding boxes from `coarse` and is trained with samples that contain false positives

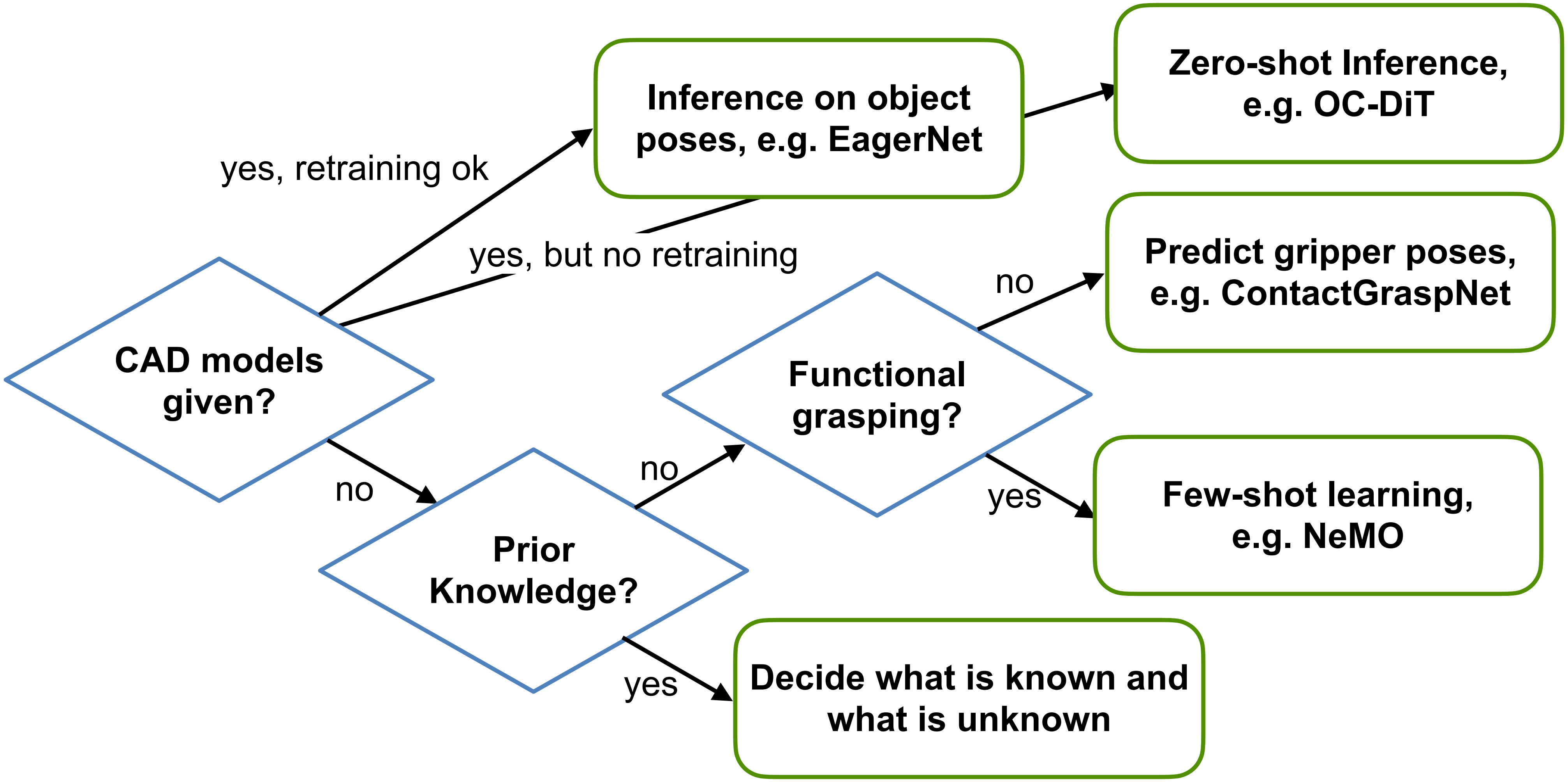
Ulmer, Boerdijk, Triebel, Durner: “Conditional Latent Diffusion Models for Zero-Shot Instance Segmentation”, *International Conference on Computer Vision* (ICCV) 2025

Qualitative Results



Ulmer, Boerdijk, Triebel, Durner: “Conditional Latent Diffusion Models for Zero-Shot Instance Segmentation”, *International Conference on Computer Vision* (ICCV) 2025

Perception Based on Known or Unseen Objects



Olov Anderson, Justus Piater: „Something unexpected always happens“

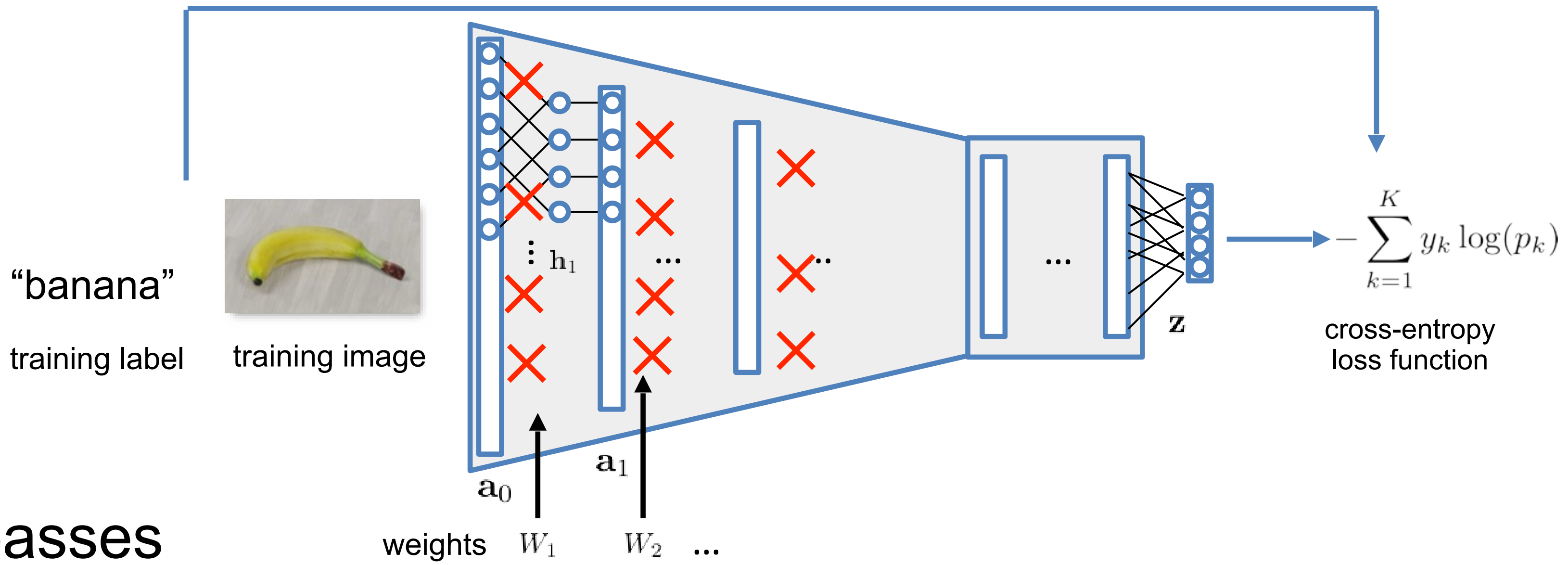
Epistemic Uncertainty from a Neural Network

Several techniques exist:

- predictive entropy
- **MC-dropout in inference**

Idea:

Run several forward passes with differently “dropped out” weights



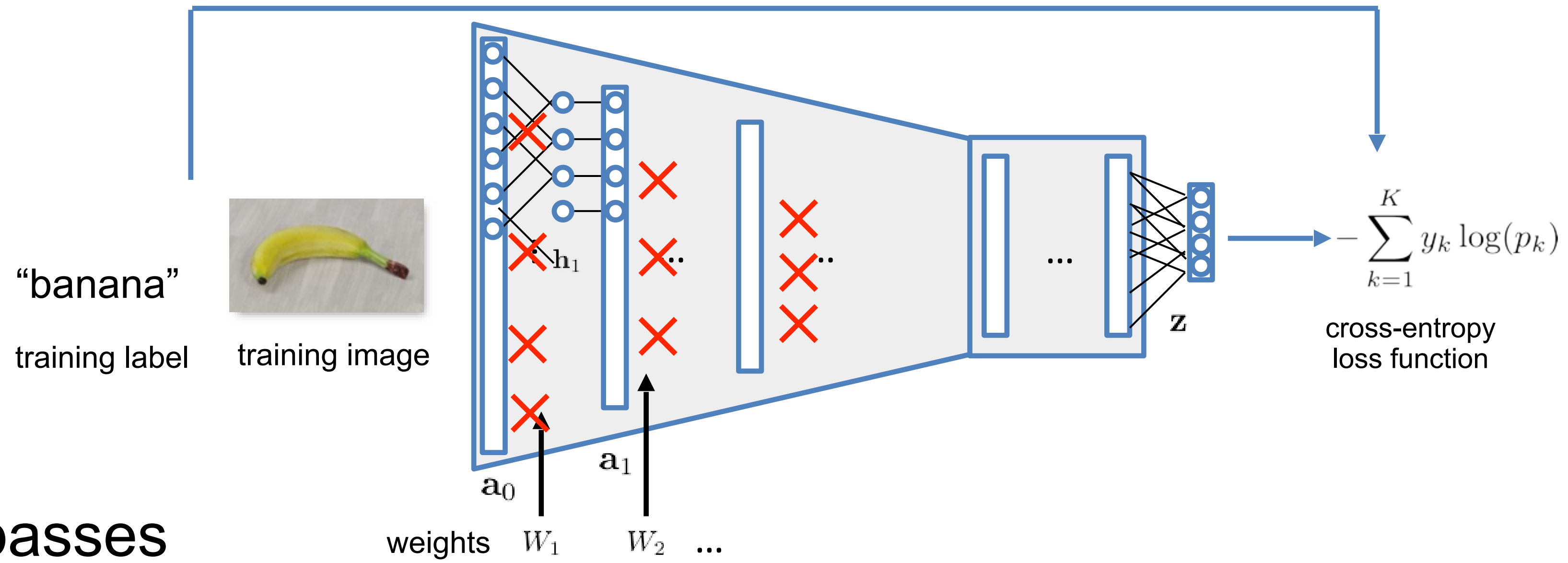
The Epistemic Uncertainty from a Neural Network

Several techniques exist:

- predictive entropy
- **MC-dropout in inference**

Idea:

Run several forward passes with differently “dropped out” weights



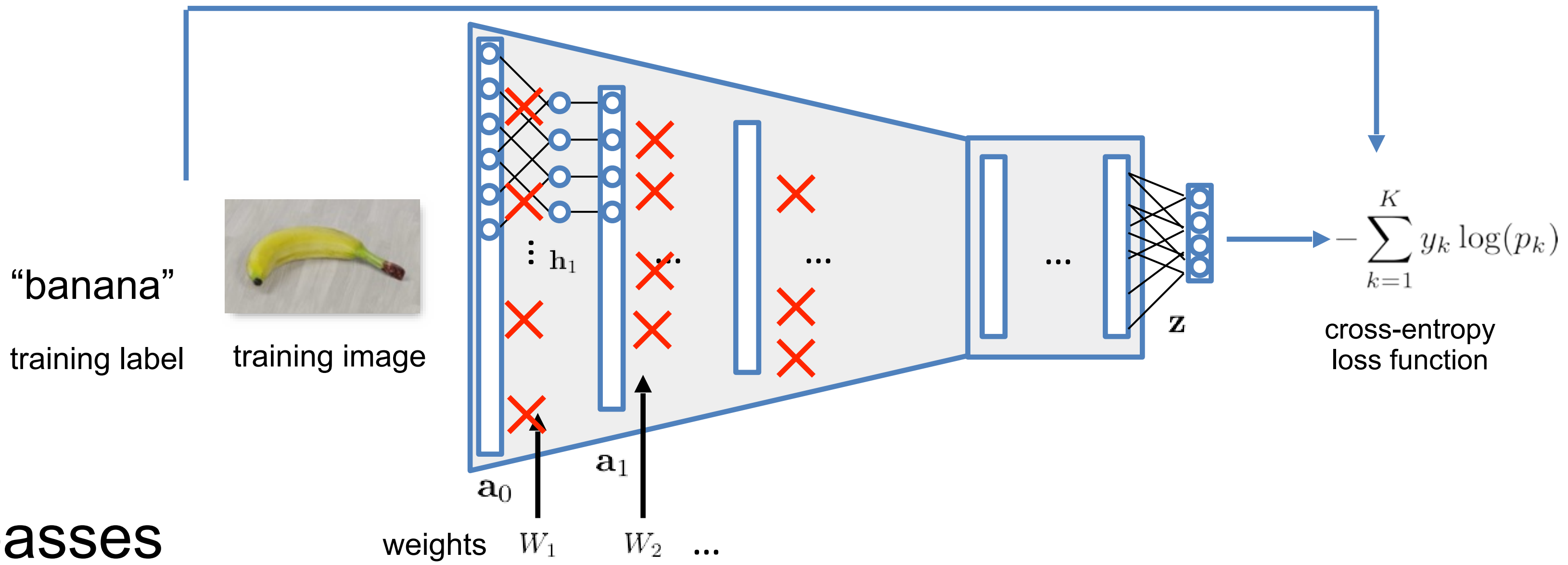
Epistemic Uncertainty from a Neural Network

Several techniques exist:

- predictive entropy
- **MC-dropout in inference**

Idea:

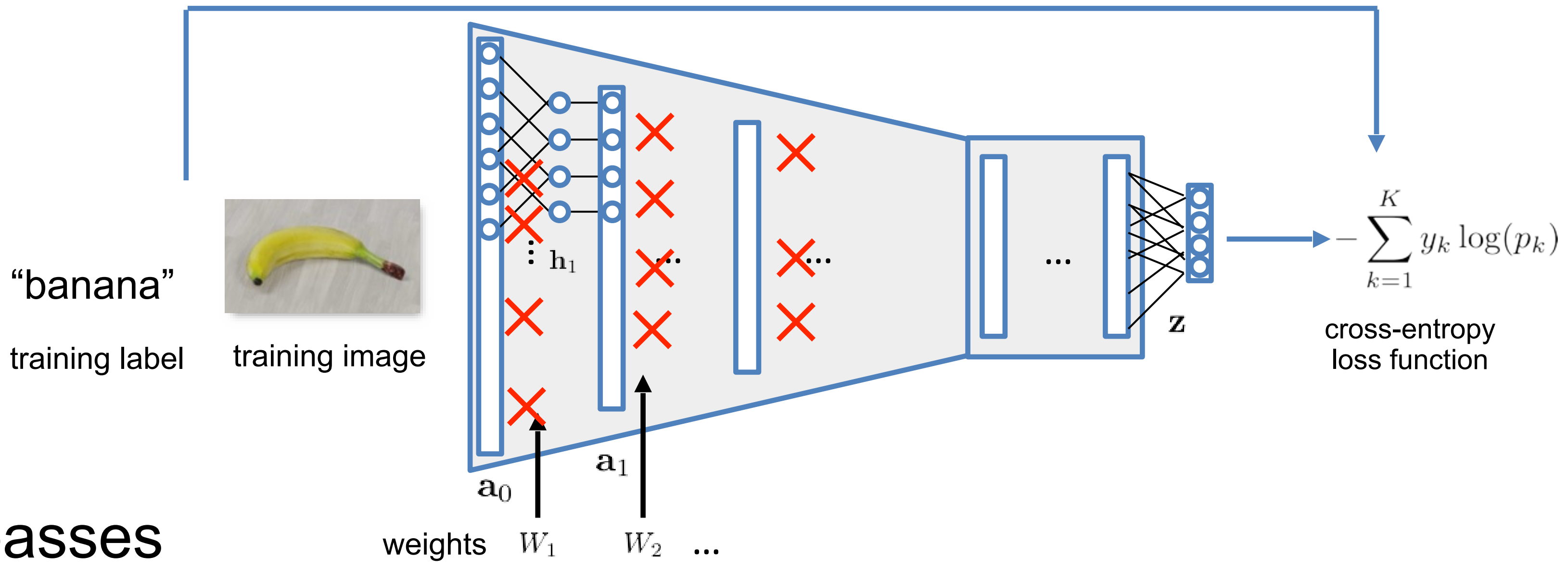
Run several forward passes with differently “dropped out” weights



Epistemic Uncertainty from a Neural Network

Several techniques exist:

- predictive entropy
- **MC-dropout in inference**



Idea:

Run several forward passes with differently "dropped out" weights

Use statistics (mean, variance) over these samples to estimate pred. dist.

Problem: Tends to be overconfident

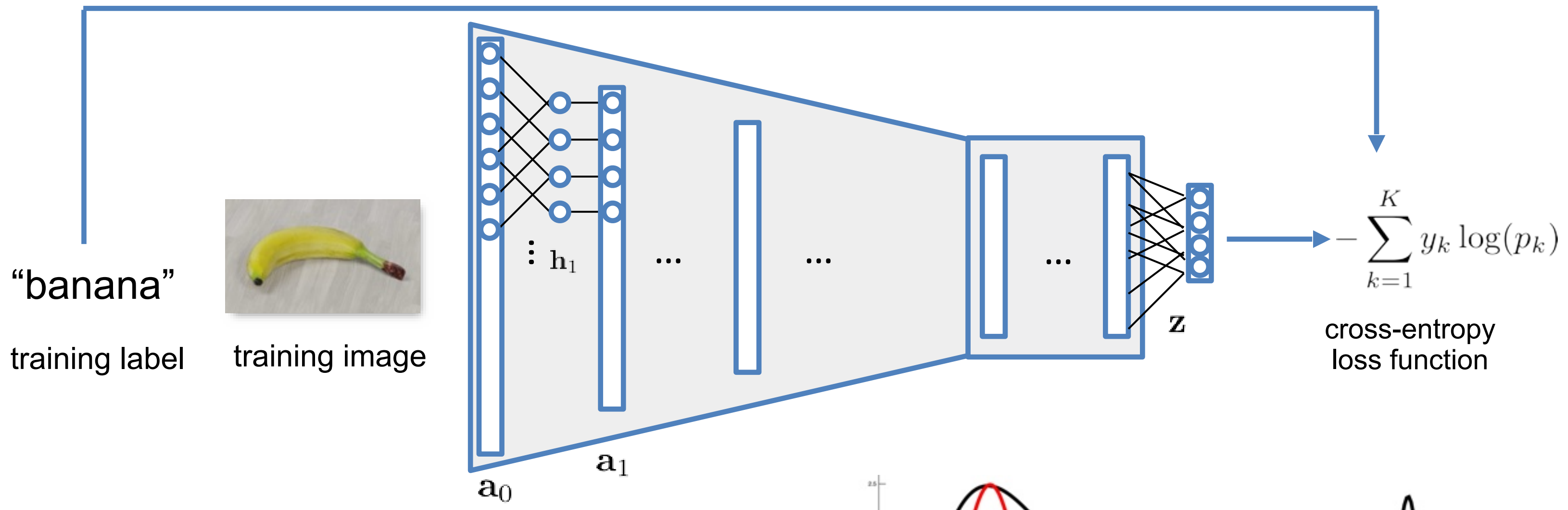
Bayesian Neural Networks

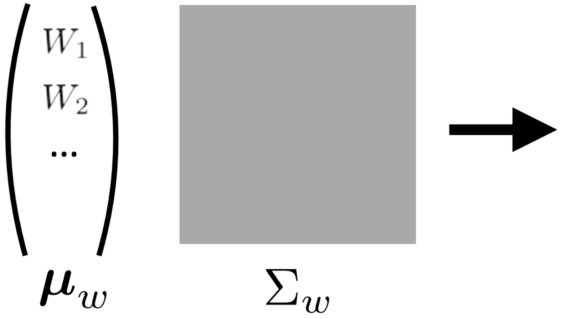
Main idea:

- use Laplace-Approximation to define a posterior
- run Monte-Carlo integration for inference

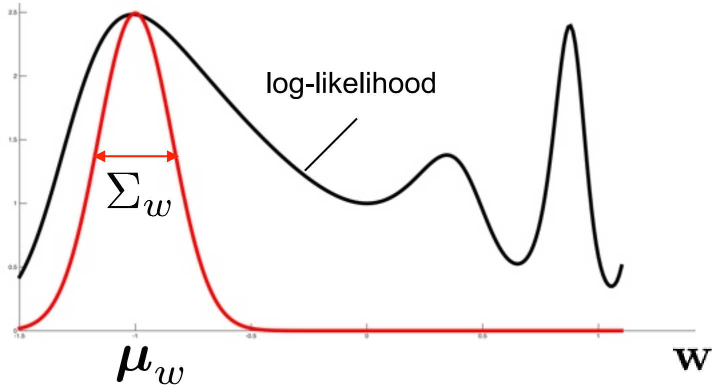
Problem:

- Inversion of H

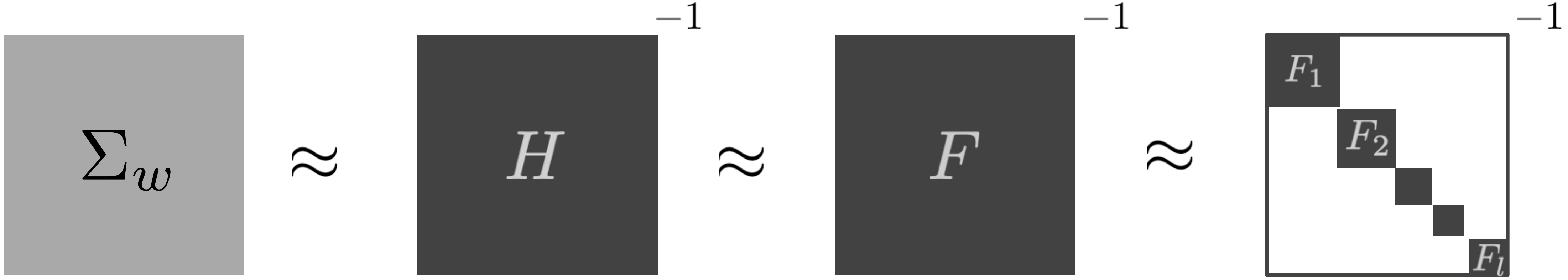




$$\mathcal{N}(\mathbf{w} \mid \boldsymbol{\mu}_w, \Sigma_w)$$
$$\Sigma_w \approx H^{-1}$$



Approximating the Hessian Matrix



Fisher information matrix

$$F = \mathbb{E}[\delta\boldsymbol{\theta}\delta\boldsymbol{\theta}^T]$$

Assume uncorrelated layers

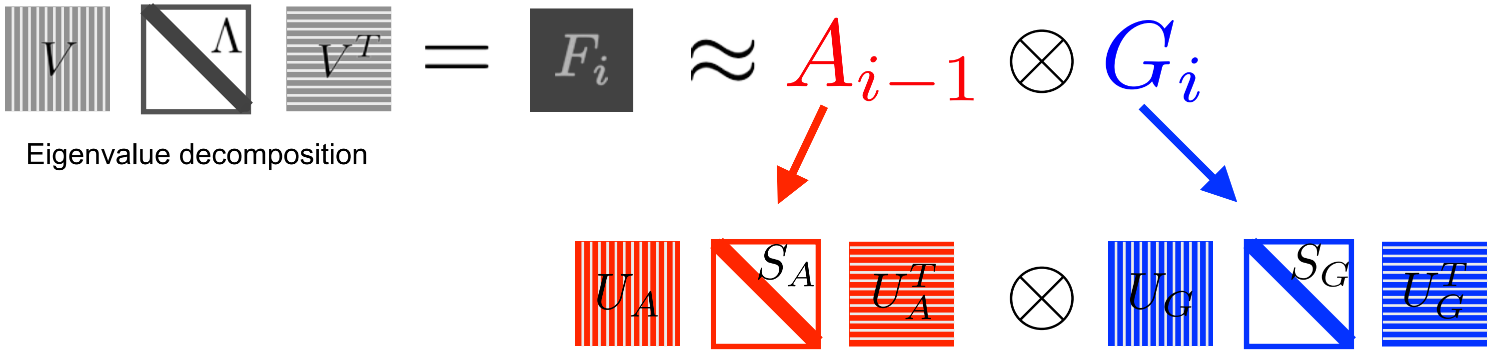
$$F_i = \mathbb{E}[\blacksquare \otimes \blacksquare] \approx \mathbb{E}[\blacksquare] \otimes \mathbb{E}[\blacksquare] = A_{i-1} \otimes G_i$$

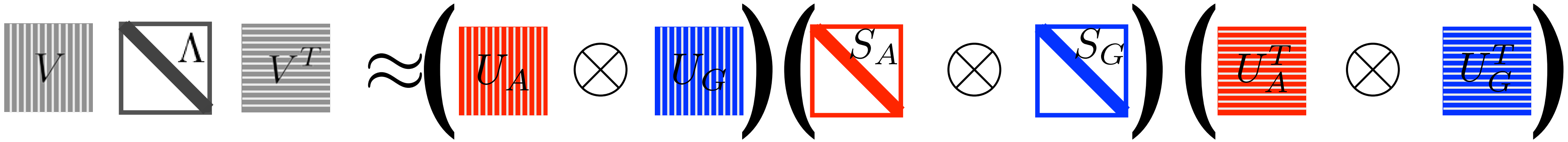
$$\mathbf{a}_{i-1}\mathbf{a}_{i-1}^T \otimes \mathbf{g}_i\mathbf{g}_i^T$$

“Kronecker Factorisation”

Lee, Humt, Feng, Triebel: “Estimating Model Uncertainty of Neural Networks in Sparse Information Form”, *Intern. Conf. on Machine Learning* (ICML) 2020

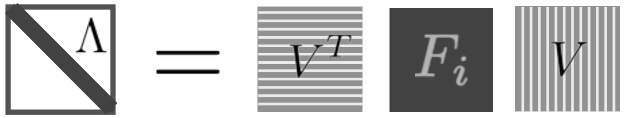
Approximating the Hessian Matrix





$$\Rightarrow (U_A \otimes U_G) \approx V$$

$$F \approx (U_A \otimes U_G)\Lambda(U_A \otimes U_G)^T$$



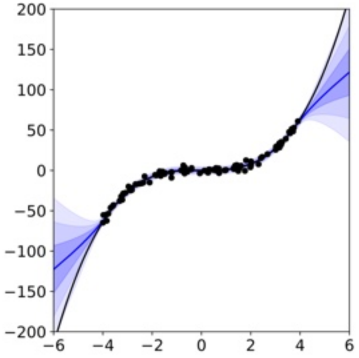
$$\Rightarrow \Lambda_{jj} = \mathbb{E}[(V^T \delta\boldsymbol{\theta})_j^2]$$

A further improvement can be made by setting the diagonals to the exact diagonals of F:

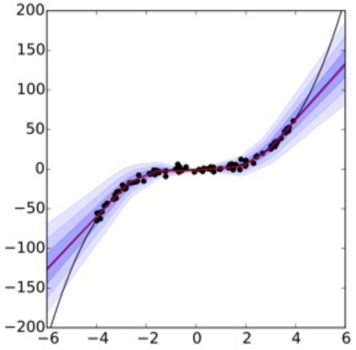
$$F \approx (U_A \otimes U_G)\Lambda(U_A \otimes U_G)^T + D$$

Lee, Humt, Feng, Triebel: "Estimating Model Uncertainty of Neural Networks in Sparse Information Form", *Intern. Conf. on Machine Learning* (ICML) 2020

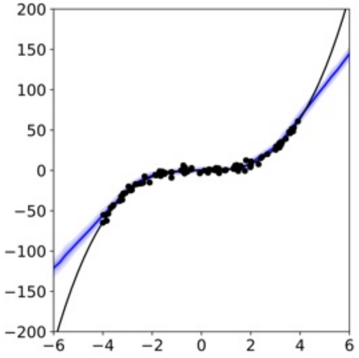
BNNs in Practice: Knowing When We Don't Know



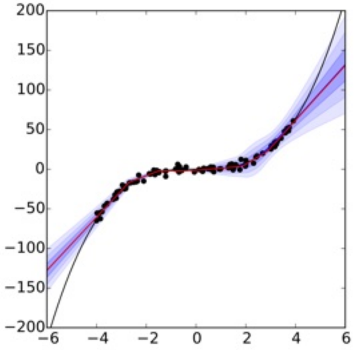
HMC ("ground truth")



Classical KFac [1]



Bayes-by-backprop [2]



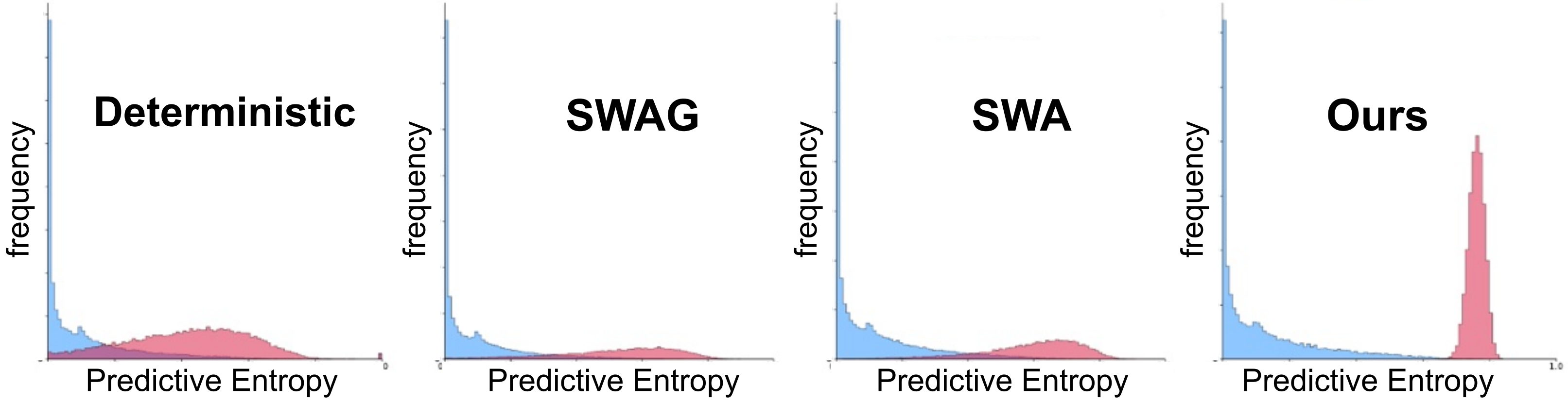
Information-based (ours)

Over- and under-confidence in a toy regression problem

[1] Ritter et al., ICLR 2018
[2] Blundell et al., ICML 2015

Training Data:
- ImageNet with 1000 classes
- 14 million images

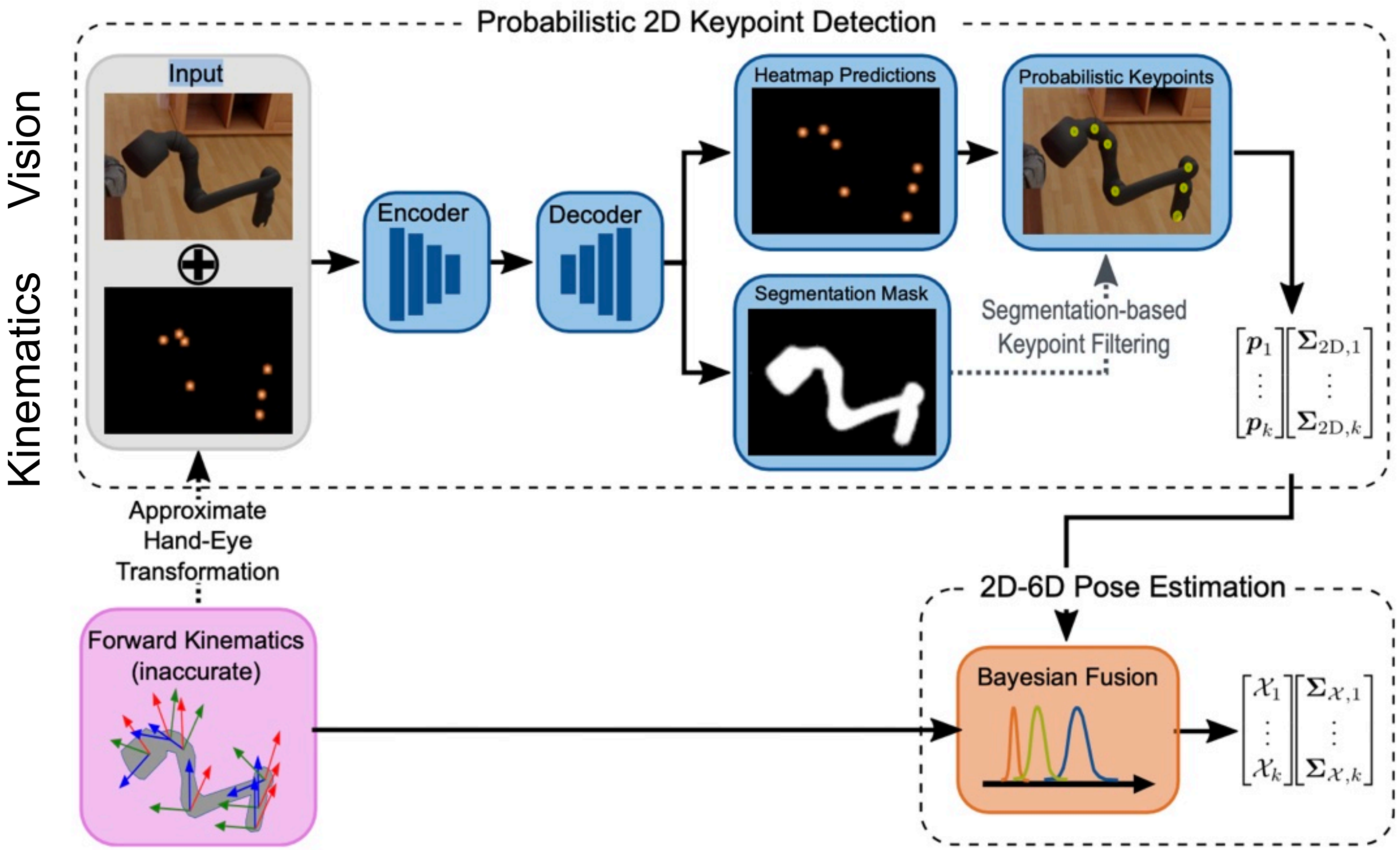
Test Data:
- artistic impressions, paintings



known classes
OOD data

Lee, Humt, Feng, Triebel: "Estimating Model Uncertainty of Neural Networks in Sparse Information Form", *Intern. Conf. on Machine Learning* (ICML) 2020

Example: Fusing Kinematics with Vision



- Learn a network to predict 2D key points from kinematics
- Input is an image and an erroneous set of key points (=joint locations in 2D)
- Output is a mask of the arm and corrected key points
- Uncertainty is estimated using a Bayesian NN
- Fusion of kinematics and vision using an EKF

"Seeing Through Uncertainty: Robot Pose Estimation Based on Imperfect Prior Kinematic Knowledge", Klüpfel, Burkhard, Reichert, Durner, Triebel, in: IEEE Transactions on Robotics (41), 2025

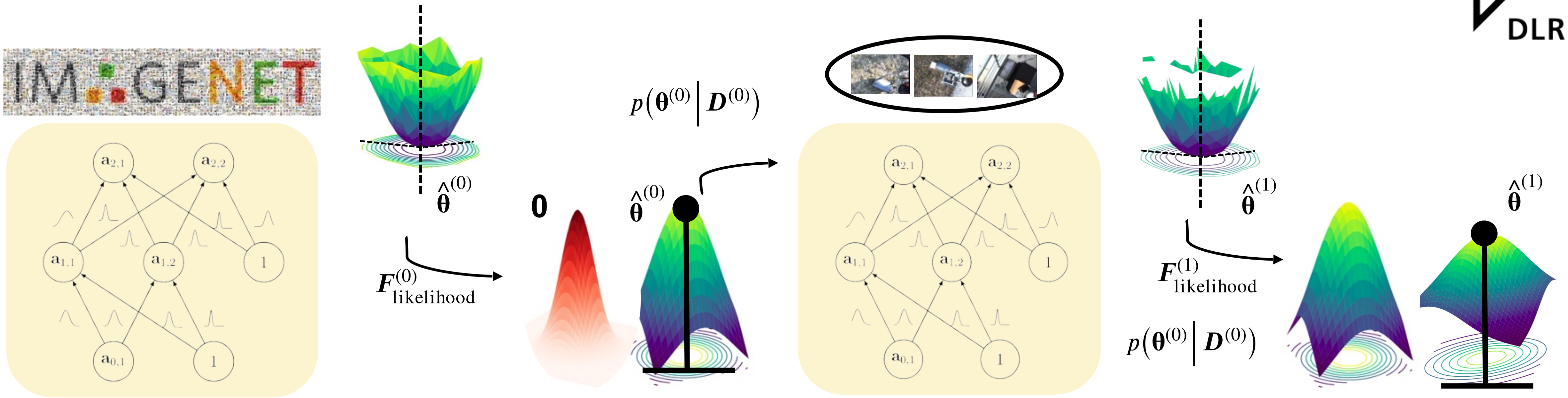
Results: Kinematics and Vision

Method	*Panda* PCK (@px) ↑					*Jaco2* PCK (@px) ↑					*neoDavid* PCK (@px) ↑				
	1	3	5	10	50	1	3	5	10	50	1	3	5	10	50
Ours	**0.068**	**0.491**	**0.793**	**0.944**	**0.992**	0.020	**0.189**	**0.428**	**0.786**	**0.98**	0.009	0.117	**0.277**	**0.428**	**0.727**
Ours w/ o seg.	0.05	0.436	0.742	0.922	0.991	**0.021**	0.176	0.425	0.768	0.957	**0.016**	0.103	0.212	0.32	0.569
Ours w/ o (PK + seg.)	0.05	0.404	0.686	0.822	0.844	0.016	0.118	0.25	0.427	0.66	0.012	**0.152**	0.260	0.418	0.610
DREAM w/ PK	0.057	0.398	0.679	0.871	0.969	0.012	0.087	0.244	0.569	0.828					
DREAM	0.041	0.35	0.631	0.766	0.789	0.004	0.019	0.047	0.12	0.224					

- Evaluated on three different robot arms
- Overall, Bayesian fusion of predicted kinematics and corrected vision worked best
- On the challenging neoDavid arm, the fused method outperforms others at larger levels of thresholds, for key point detection

Results on neoDavid

How to find a good prior?



At task 0 (broad data-set and large architecture)

- Maximum a posteriori estimation

$$\hat{\boldsymbol{\theta}}^{(0)} \in \arg\max p(\boldsymbol{\theta}^{(0)} \mid D^{(0)})$$

- Laplace Approximation

$$p(\boldsymbol{\theta}^{(0)} \mid D^{(0)}) \approx \mathcal{N}\left(\boldsymbol{\theta}^{(0)} \mid \hat{\boldsymbol{\theta}}^{(0)}, (F^{(0)})^{-1}\right)$$

- Kronecker-factorized information matrix

$$\boldsymbol{F}^{(0)} = \boldsymbol{F}^{(0)}_{\text{likelihood}} + \boldsymbol{F}^{(0)}_{\text{prior}} = \boldsymbol{L}^{(0)} \otimes \boldsymbol{R}^{(0)} + \gamma \boldsymbol{I}\,.$$

At task 1 (our robotic data of interest):

- Prior learned on task 0

$$\pi(\boldsymbol{\theta}^{(1)}) = \mathcal{N}\left(\hat{\boldsymbol{\theta}}^{(0)}, (F^{(0)})^{-1}\right)$$

- Posterior update on task 1

$$p(\boldsymbol{\theta}^{(1)} \mid D^{(1)}) \approx \mathcal{N}\left(\hat{\boldsymbol{\theta}}^{(1)}, (F^{(1)} + F^{(0)})^{-1}\right)$$

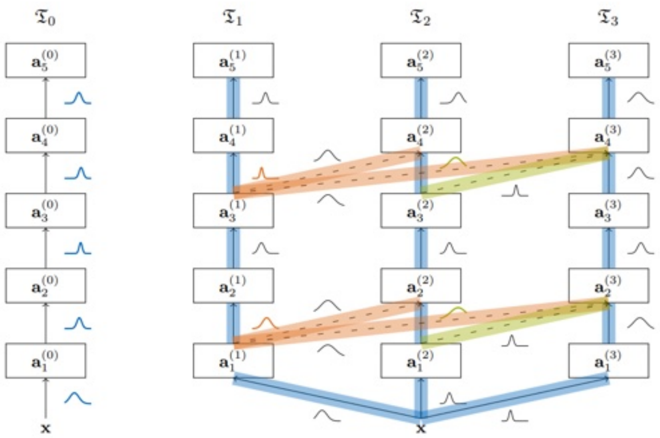
- Kronecker-factorized information matrix

$$\boldsymbol{F}^{(1)} = \boldsymbol{L}^{(1)} \otimes \boldsymbol{R}^{(1)} + \boldsymbol{L}^{(0)} \otimes \boldsymbol{R}^{(0)} + \gamma \boldsymbol{I}\,.$$

Learning Expressive Priors for Generalization and Uncertainty Estimation in Neural Networks. D. Schnaus*, J. Lee*, D. Cremers, R. Triebel. ICML 2023.

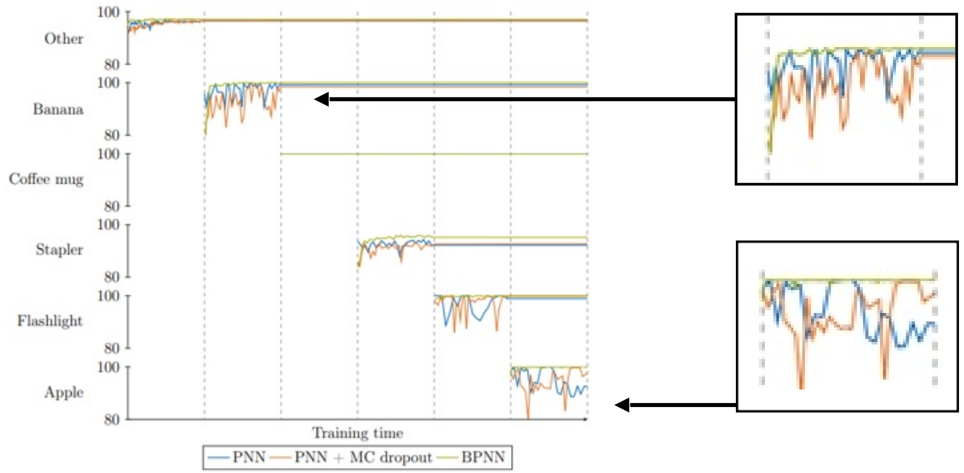
Continual learning: Bayesian Progressive Neural Networks

Extensions to continual learning



Bayesian interpretation of progressive neural networks (Rusu et al,2016).

Evaluation within a robotic benchmark

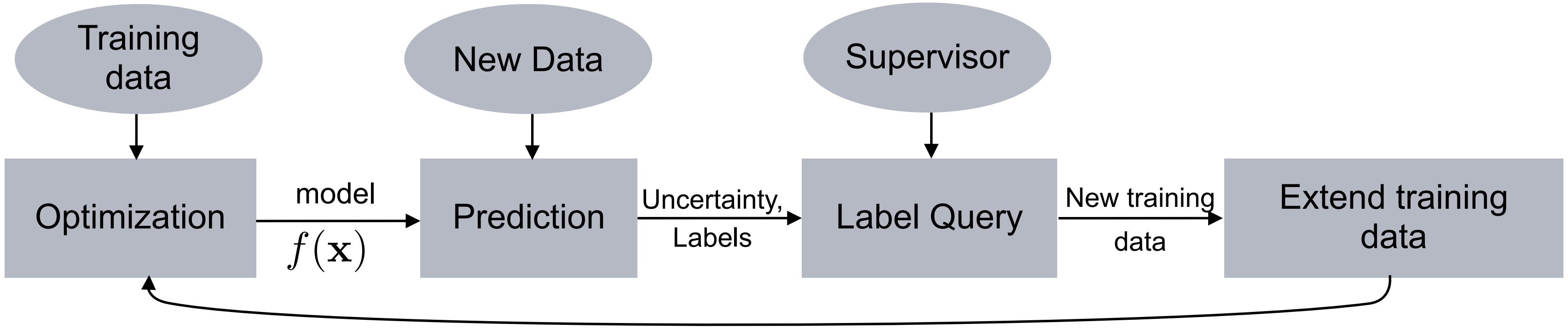


	OTHER	BANANA	COFFEE MUG	STAPLER	FLASHLIGHT	APPLE	AVERAGE
PNN (WEIGHT DECAY 10^{-3})	95.7 ± 0.4	98.5 ± 2.3	**100.0 ± 0.0**	91.7 ± 1.0	93.8 ± 9.1	93.3 ± 4.7	95.5 ± 2.9
PNN (WEIGHT DECAY 10^{-5})	**96.7 ± 0.3**	99.0 ± 1.2	**100.0 ± 0.0**	90.7 ± 2.3	99.7 ± 0.4	94.1 ± 3.2	96.7 ± 1.2
MC DROPOUT	96.3 ± 0.1	**99.3 ± 0.9**	**100.0 ± 0.0**	92.7 ± 0.8	99.8 ± 0.4	90.4 ± 5.1	96.4 ± 1.2
ZERO MEAN & ISOTROPIC	96.3 ± 0.3	95.5 ± 4.2	**100.0 ± 0.1**	91.9 ± 1.7	**100.0 ± 0.1**	87.7 ± 7.8	95.2 ± 2.4
ISOTROPIC	96.1 ± 0.3	98.7 ± 1.0	**100.0 ± 0.0**	93.1 ± 0.6	**100.0 ± 0.0**	87.8 ± 6.6	96.0 ± 1.4
LEARNED	96.2 ± 0.2	98.4 ± 1.6	**100.0 ± 0.0**	**93.9 ± 0.9**	**100.0 ± 0.0**	**95.1 ± 4.7**	**97.3 ± 1.2**

Natural extensions to continual learning for application scenarios in robotics.

Learning Expressive Priors for Generalization and Uncertainty Estimation in Neural Networks. D. Schnaus*, J. Lee*, D. Cremers, R. Triebel. ICML 2023.
Persistent Anytime Learning of Objects from Unseen Classes. M. Denninger and R. Triebel. IROS 2018. Best Cognitive Robotics Paper Finalist

Active Learning with a Humanoid



- Epistemic uncertainty can be used to query a human supervisor in case of OOD data
- New classes can be learned using the Bayesian Progressive Neural Network approach by adding new branches
- Learning can be done comparably fast

Conclusions

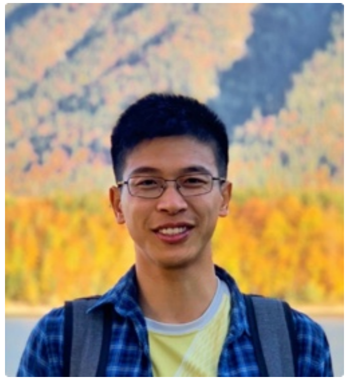
- Current methods in robot perception for manipulation require less geometric knowledge about the objects, but rather rely on image data
- Generative AI methods such as diffusion models are powerful tools, e.g. for semantic segmentation, although still costly
- Bayesian Neural Networks are useful to get epistemic uncertainty.
- This can be used for fusion with kinematics or for active learning.

Thank you!


Wout Boerdijk


Anne Reichert


Maximilian Durner


Jianxiang Feng


Matthias Humt


Sebastian Jung

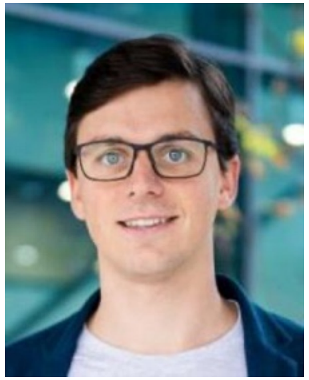

Leonard Klüpfel


Jongseok Lee


Martin Sundermeyer


Manuel Stoiber


Maximilian Ulmer


Lukas Burkhard