Domain Adaptation at the Edge using Hyperdimensional Computing

Tutorial presented by Aswin Raghavan

IEEE 43nd International Conference on Consumer Electronics

January 11, 2025

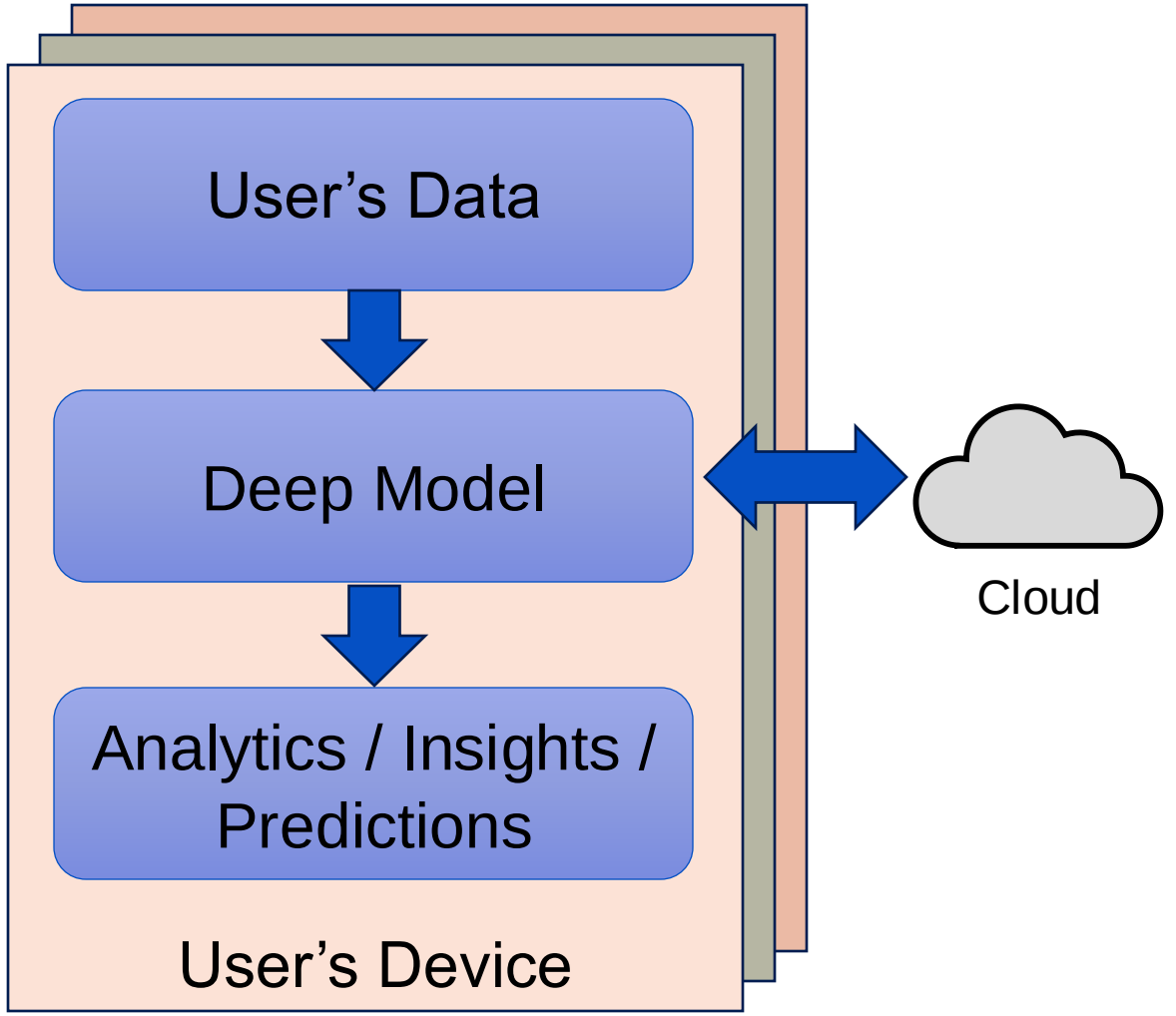
Tutorial Handouts

- All the material for this tutorial is hosted on github
  - Repo: https://github.com/aswin-raghavan/icce-tutorial-2025-domain-adaptation
  - There, find these slides
  - If you want to run the notebooks, find the Conda environment as well

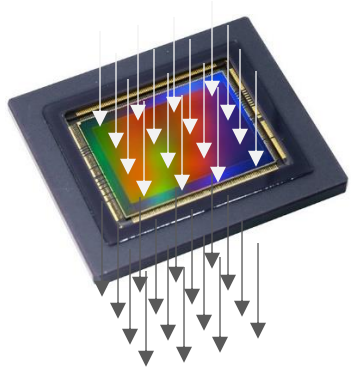
- This is a hands-on tutorial
- AI/ML expertise is not required
- Please ask questions if terminology is unclear

Tutorial Scope

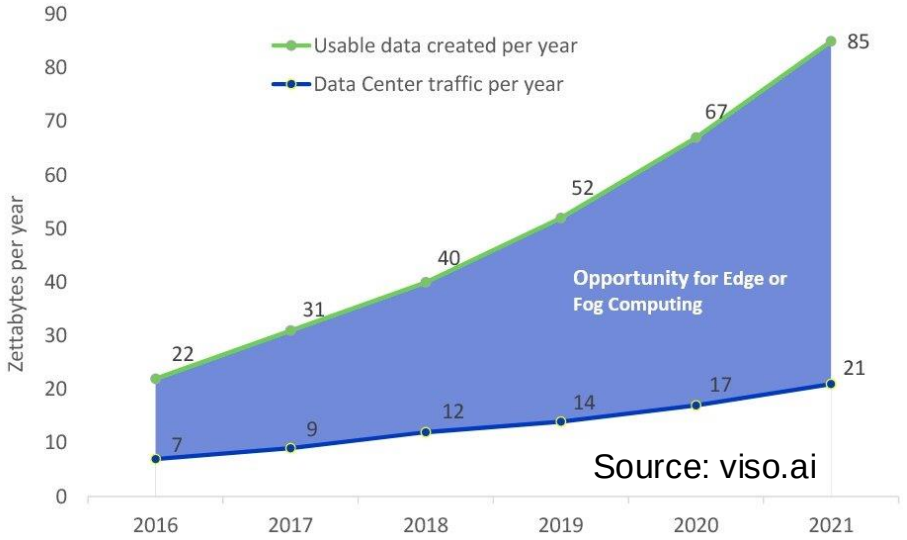
- Suppose that you are building an app that uses Artificial Intelligence (AI) / ML
  - Many powerful **Deep Models** are available today that are **Pre-Trained**
  - But these models are too big and slow to adapt on edge or IoT devices
- **Domain**: The input to your model e.g. text, image, depth map, other sensors.
- **Adaptation**: Make the model work well in your application specific domain and user
- at the **Edge:** adapt the model **efficiently** to work **efficiently** across different user's devices (low Size, Weight, Power)
- **Personalization**: adapt the model to work for each user's private data



Why do we need Edge Intelligence?



High sensor throughput (8kx8k, 120Hz, 16-bit, ~100Gb/s): communication of all sensor data to cloud is not practical



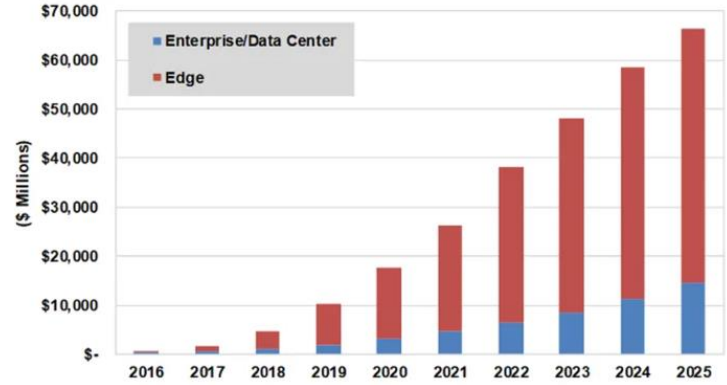
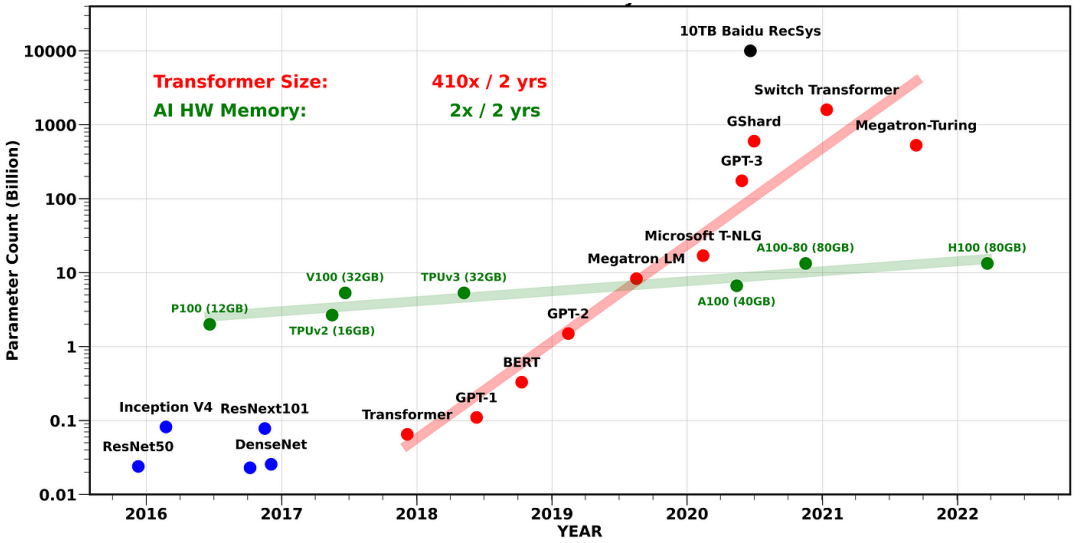


Fig. 1: Deep learning chipset revenue by market sector. Source: Tractica.

- Most of the data is produced at the edge
- Data deluge due to high throughput sensors
- Chipset market dominated by edge
- But AI complexity out-growing hardware

We need better methods to adapt large powerful AI models to edge devices of all shapes and sizes



Why Domain Adaptation for Personalized Edge Intelligence?

Compared to train-once and then deploy, with the original training performed on Cloud

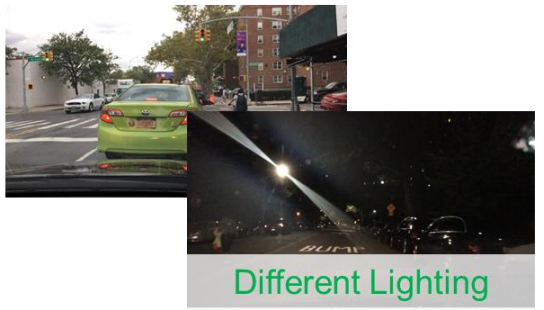
- **Domain Shift**: Operating with different sensor package, for example
- **Concept Drift**: Each user has a different target concept or mental model
- **Dynamic World**: Operating in rapidly changing environment

Challenges

- **Complexity** - AI system complexity is growing – large models, data/power hungry
- **Cloud reliance –** Cost and delay from the data movement between cloud and edge
- **Memory wall –** Von Neumann architectures require large amounts of data movement
- **Cost of data annotation –** At-the-edge labelling is limited to few samples

Changes in sensor modality

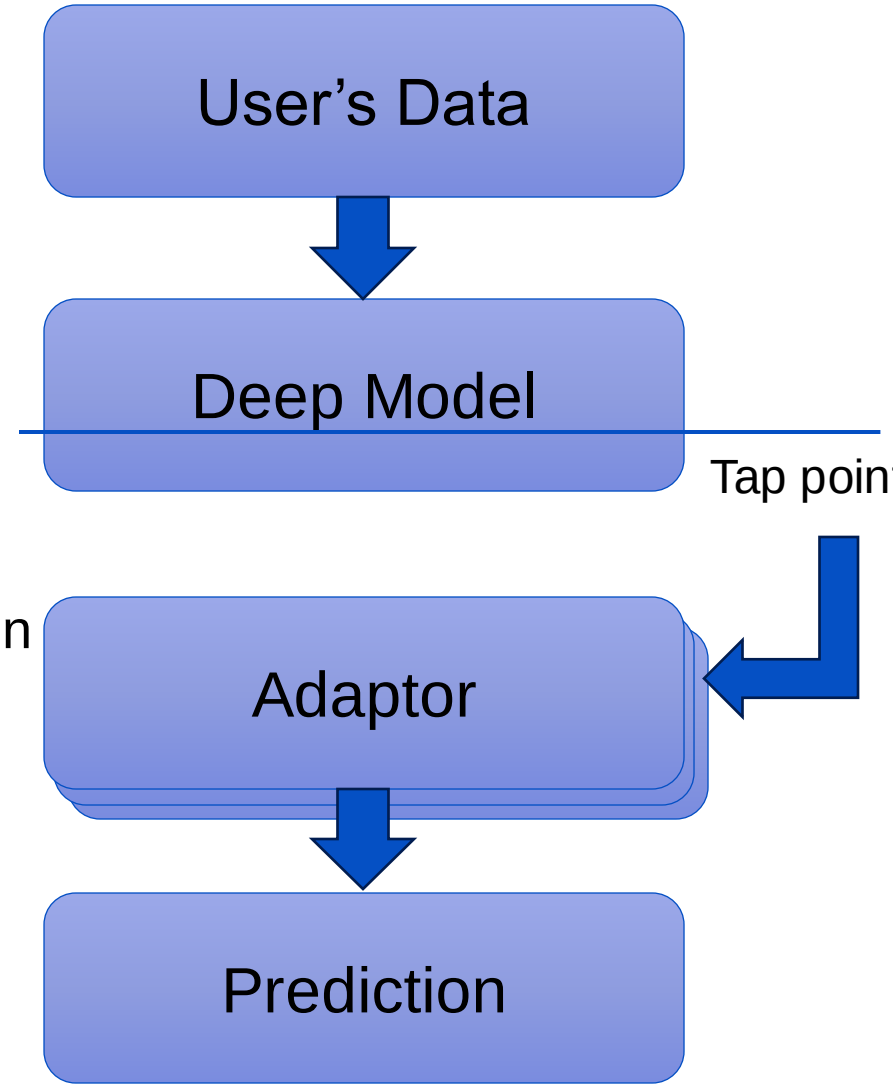
Changes in task space



Changes in environmental factors

Tutorial Outline

1. Notebooks on Hyperdimensional Computing (HD)
   - Introduction to HD computing
   - Training HD encoders
   - Domain Adaptation to depth images
2. Slides on Case studies
   - Domain Adaptation in object detection
   - Domain Adaptation in video activity recognition
   - Application to retrieval in Retrieval Augmented Generation
3. Optimize the tap point for the adaptor
   - When do I need an adaptor? OOD Detection using HD
   - Where do I put the tap? Some theory



About Me

- PhD in Computer Science, Oregon State University (2017)
- Since 2017, Scientist / Principal Investigator at SRI International
  - Edge Intelligence: Low Precision Neural Networks, Domain Adaptation, Continual Learning
  - Decision Making: Lifelong Reinforcement Learning, Multi-Agent Reinforcement Learning
  - Human-AI Integration: Visual Storytelling, Collaborative Problem Solving, Coactive Design
- Free time: play music, long-distance run, woodworking, crochet.
- https://scholar.google.com/citations?user=Ss2KBccAAAAJ

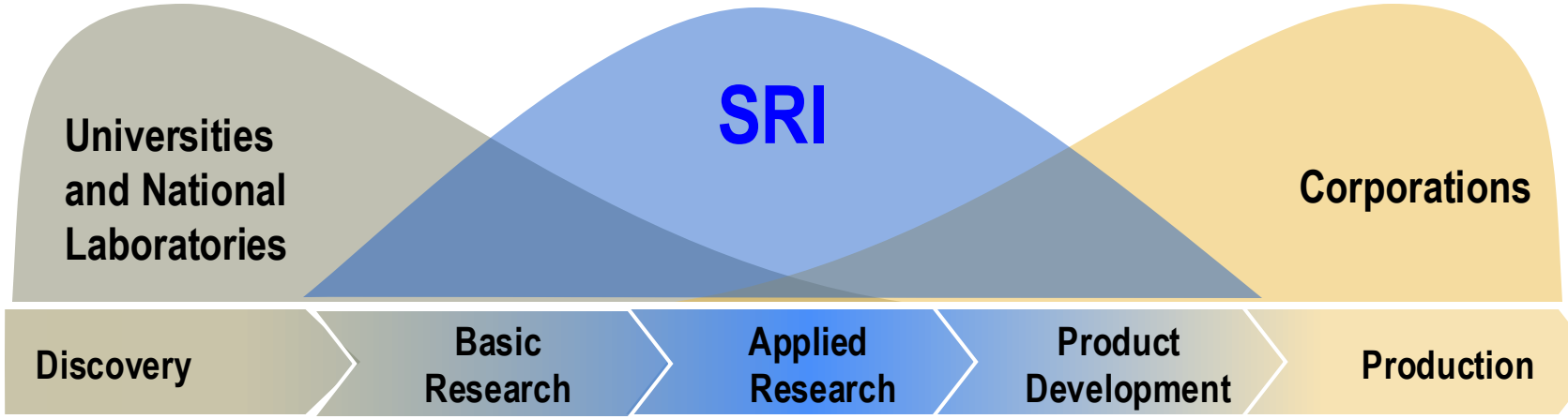
About SRI: From Research to Commercialization Across the Sciences

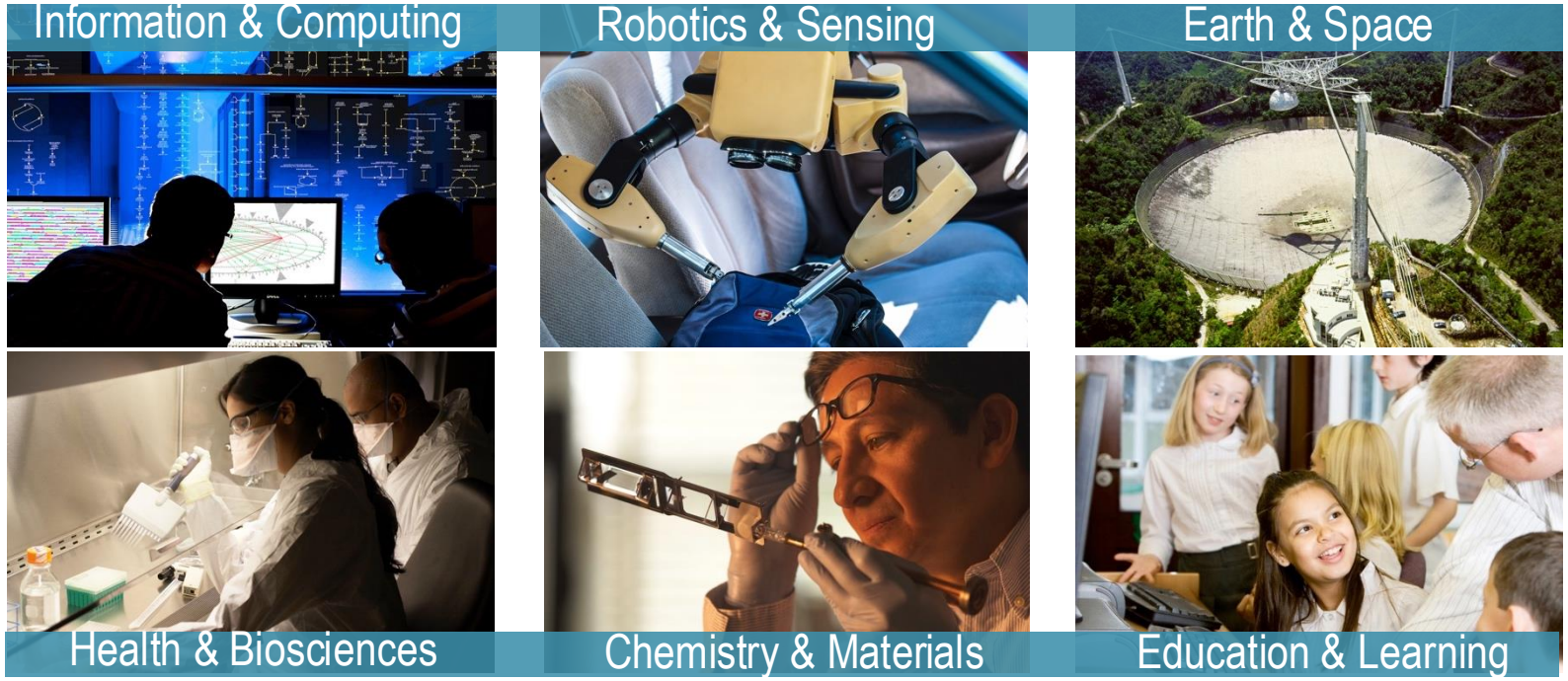
14
Locations in US

1,600+
Staff members

50+
Spin-off companies

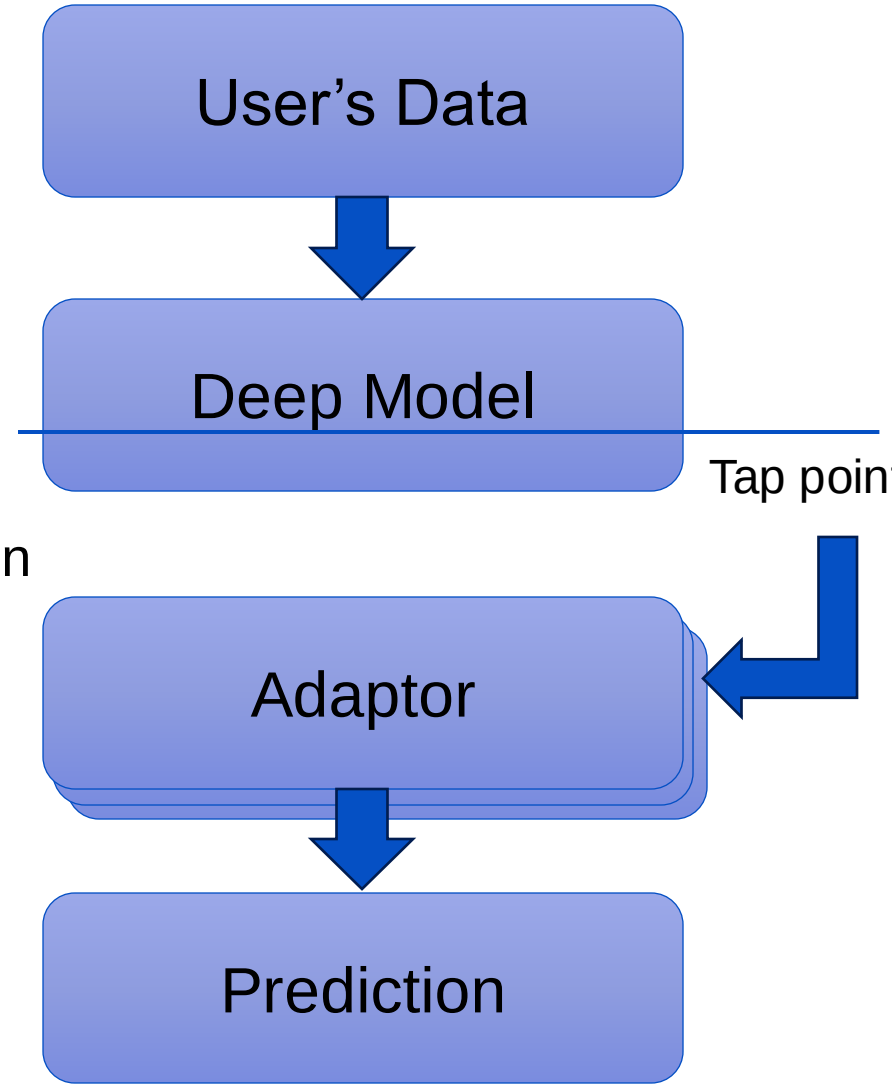
100+
Active licenses





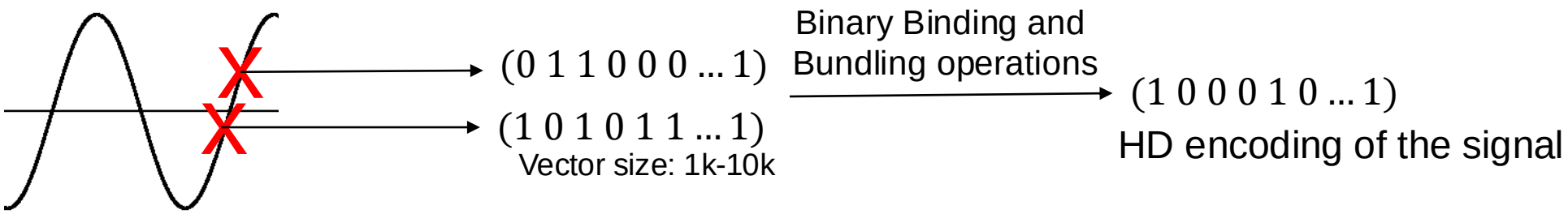
Tutorial Outline

1. **Notebooks on Hyperdimensional Computing (HD)**
   - **Introduction to HD computing**
   - Domain Adaptation to depth images
2. Slides on Case studies
   - Domain Adaptation in object detection
   - Domain Adaptation in video activity recognition
   - Application to retrieval in Retrieval Augmented Generation
3. Optimize the tap point for the adaptor
   - When do I need an adaptor? OOD Detection using HD
   - Where do I put the tap? Some theory



Introduction to Hyperdimensional (HD) Computing

What is HD computing: *Random Kernel mapping to high dimensional vector (e.g., binary or ternary) that supports efficient processing*



Efficient Operations:

Bind (XOR): $a = b * c, 1 * a = 1 - a, \quad 0 * a = a$

$a = (b * c) * d = b * (c * d)$ (associative)

Unbind: $a * (a * c) = (a * a) * c = c$

$(a * b + c * d) * a = b$ + noise[1,2]

Bundle (Avg. then threshold):

$a = \mathbf{1}(0.5b + 0.5c \geq 0.5)$

Hamming Distance: $h(a, b) = ||b - a||_1$ for binary HD vectors

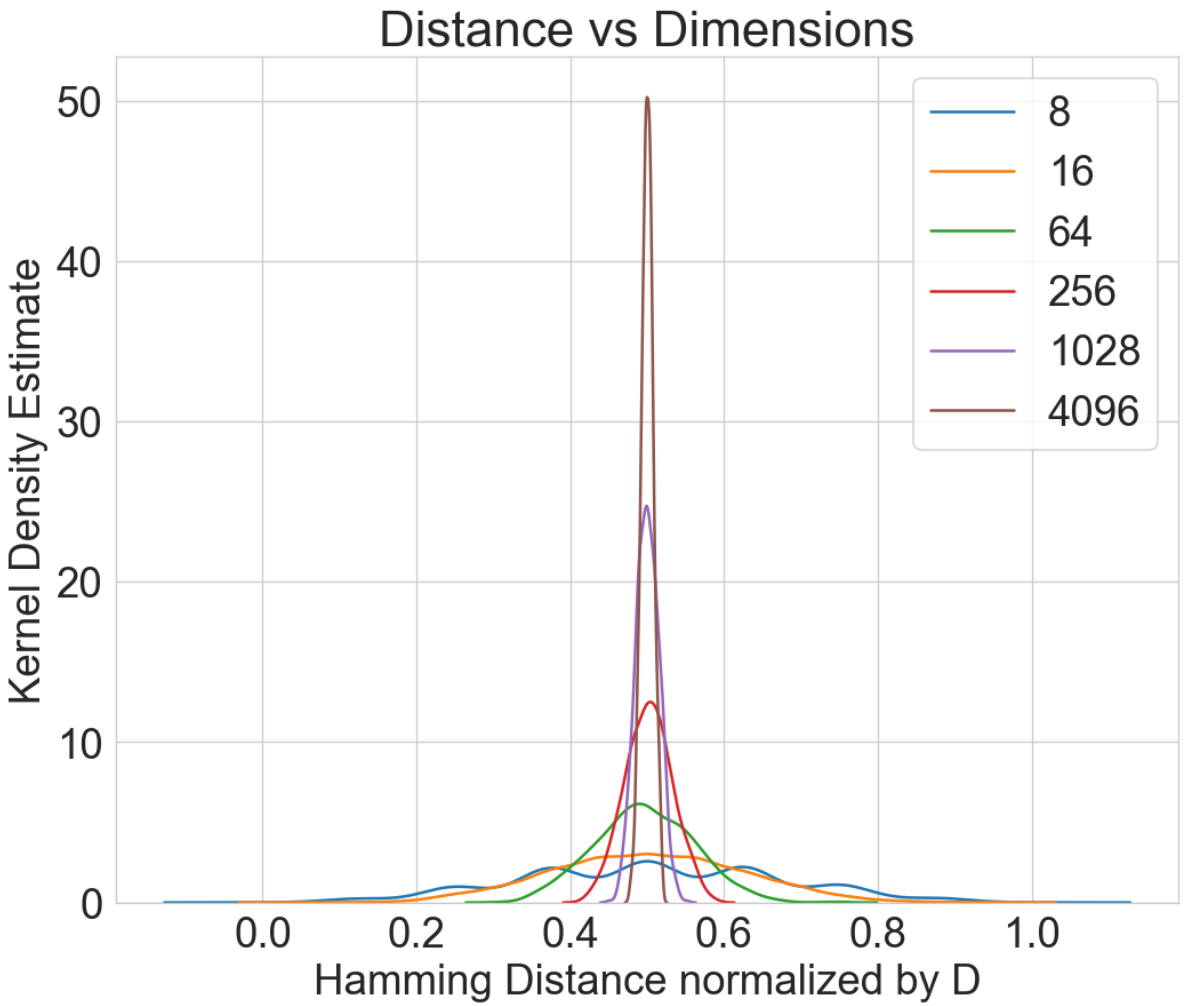
More generally, cosine distance can be used

1. A. Thomas, S. Dasgupta, T. Rosing, *Theoretical Foundations of Hyperdimensional Computing*, Dec 15, 2020.
2. Kenny Schlegel, Peer Neubert, Peter Protzel, *A comparison of Vector Symbolic Architectures*, Jan, 2020.
3. S. Levy and R.W. Gayler, *Vector Symbolic Architectures: A New Building Material for Artificial General Intelligence*, Dec, 2007.

Notebook: Introduction to HD computing

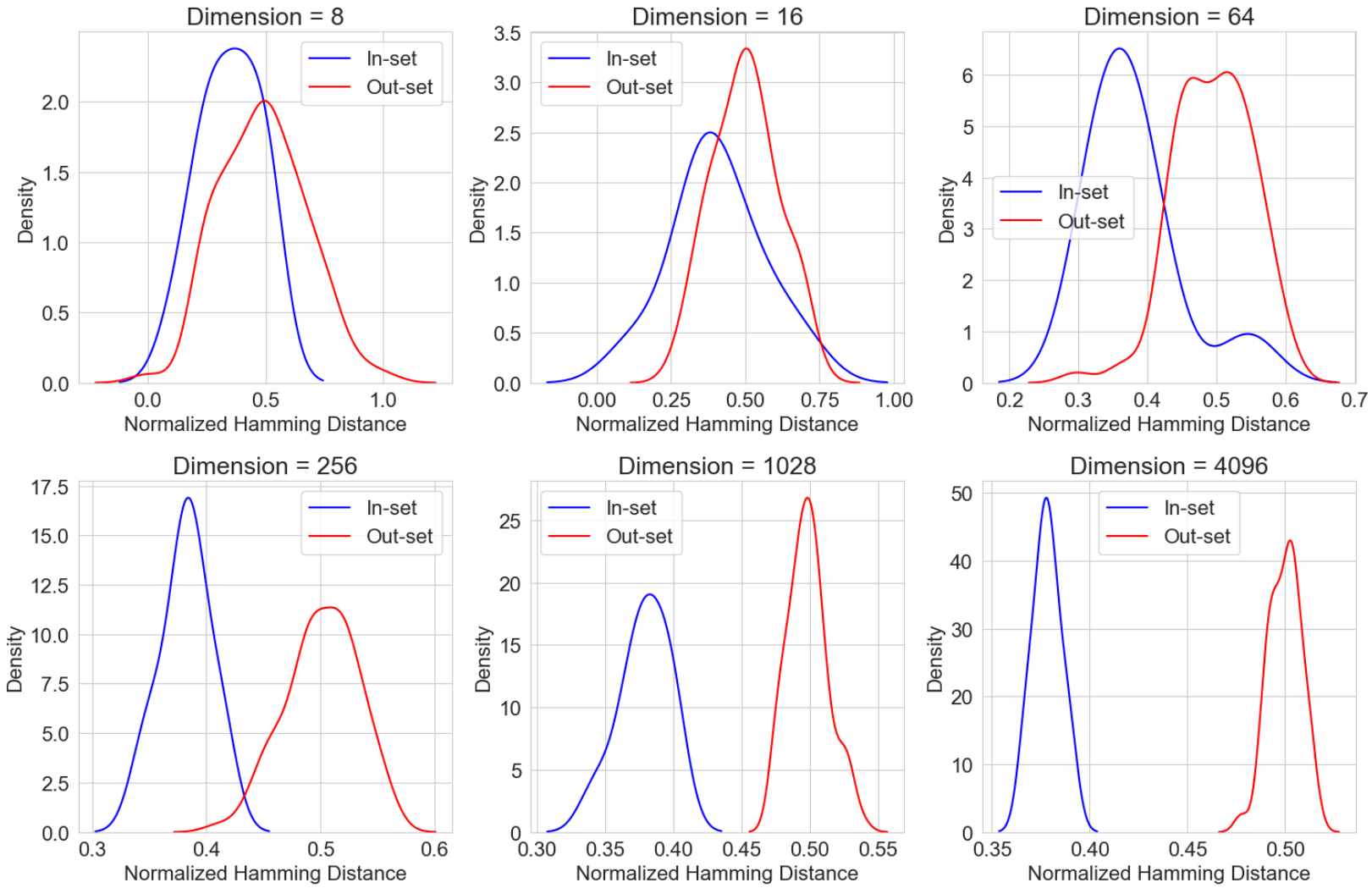
https://github.com/aswin-raghavan/icce-tutorial-2025-domain-adaptation

Recap from Notebook



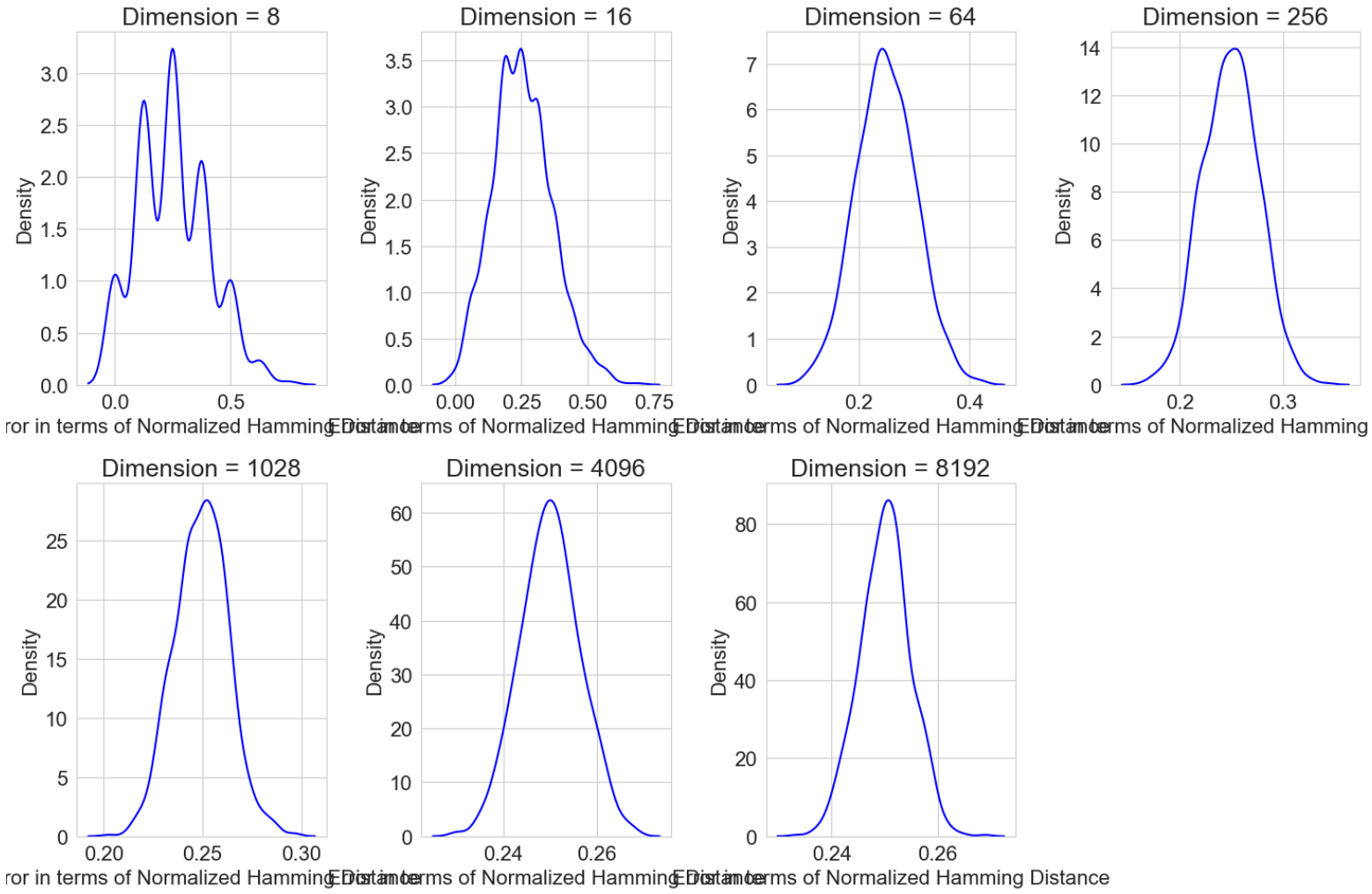
As the number of dimensions increases, the relative hamming distance concentrates around 0.5.

Recap from Notebook



Bundling any subset of HD vectors has discriminative power at small sample sizes

Recap from Notebook



Recovering individual terms from the result of a linear combination of HD vectors (bind/bundle) using unbinding produces bounded error as dims increases

Bundling to Produce Class Exemplars for Classification

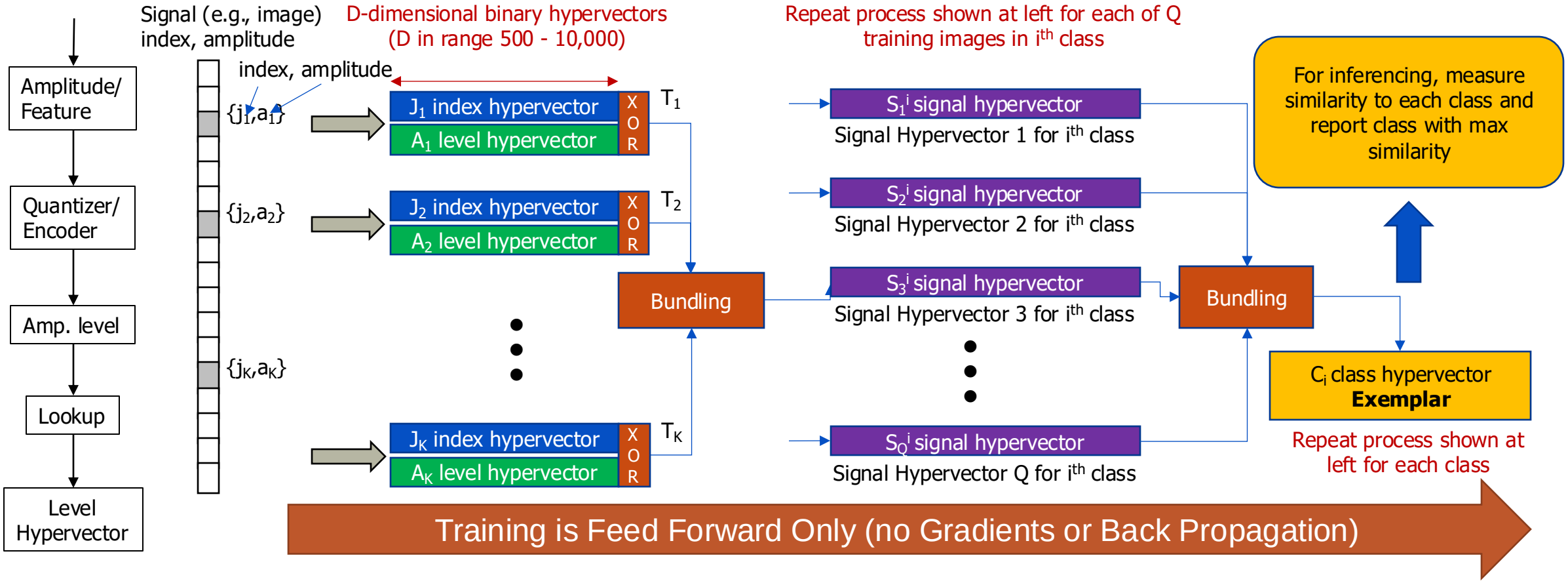
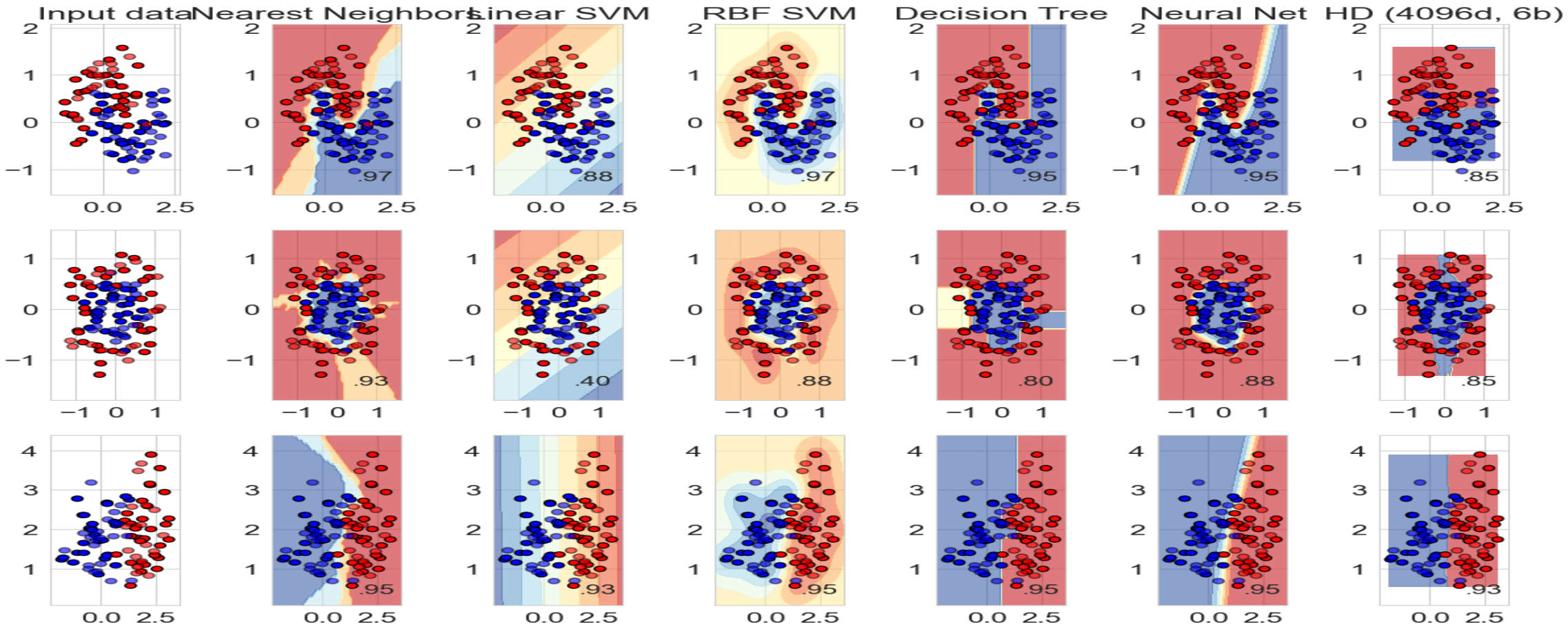


Figure based on Imani, Mohsen, et al. "VoiceHD: Hyperdimensional computing for efficient speech recognition." 2017 IEEE International Conference on Rebooting Computing (ICRC). IEEE, 2017.

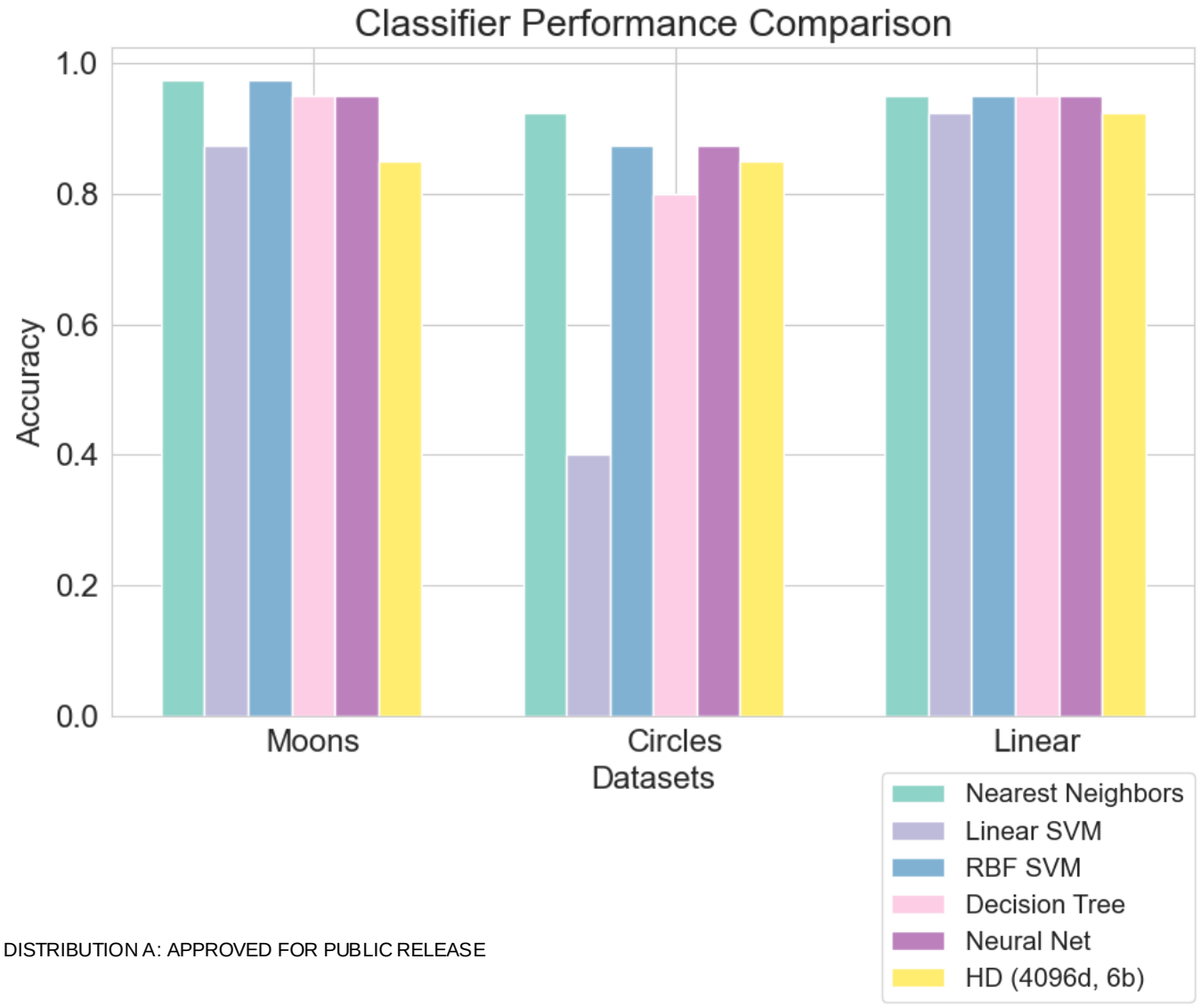
Training Workflow for HD

1. $E[y] = [\mathbf{0}]^D$ for each class y
2. For each example (X, y): (Only one iteration through the data)
   a. $T = [\mathbf{0}]^D$
   b. For each $x_i \in X$: (e.g., dims in input signal or pixel in image)
      i. V_i = encode-to-HD(x_i)
      ii. P_i = encode-to-HD(i)
      iii. $T \mathrel{+}= (V_i \oplus P_i)$ (XOR)
   c. Normalize T: $T = \frac{T}{len(X)}$ (e.g., dims of input signal or by number of pixels)
   d. Threshold t: $t[idx] = 1 \; if \; T[idx] \geq 0.5 \; else \; 0$
   e. Add to exemplar for class y: $E[y] \mathrel{+}= t$ (Online step: on-the-fly training)
3. Normalize and Threshold exemplar: $\mathbf{1}(\frac{E[y]}{(\#\, examples\, of\, y)} \geq 0.5)$

Recap from Notebook: Fit HD Classifiers on 2D Datasets



Recap from Notebook



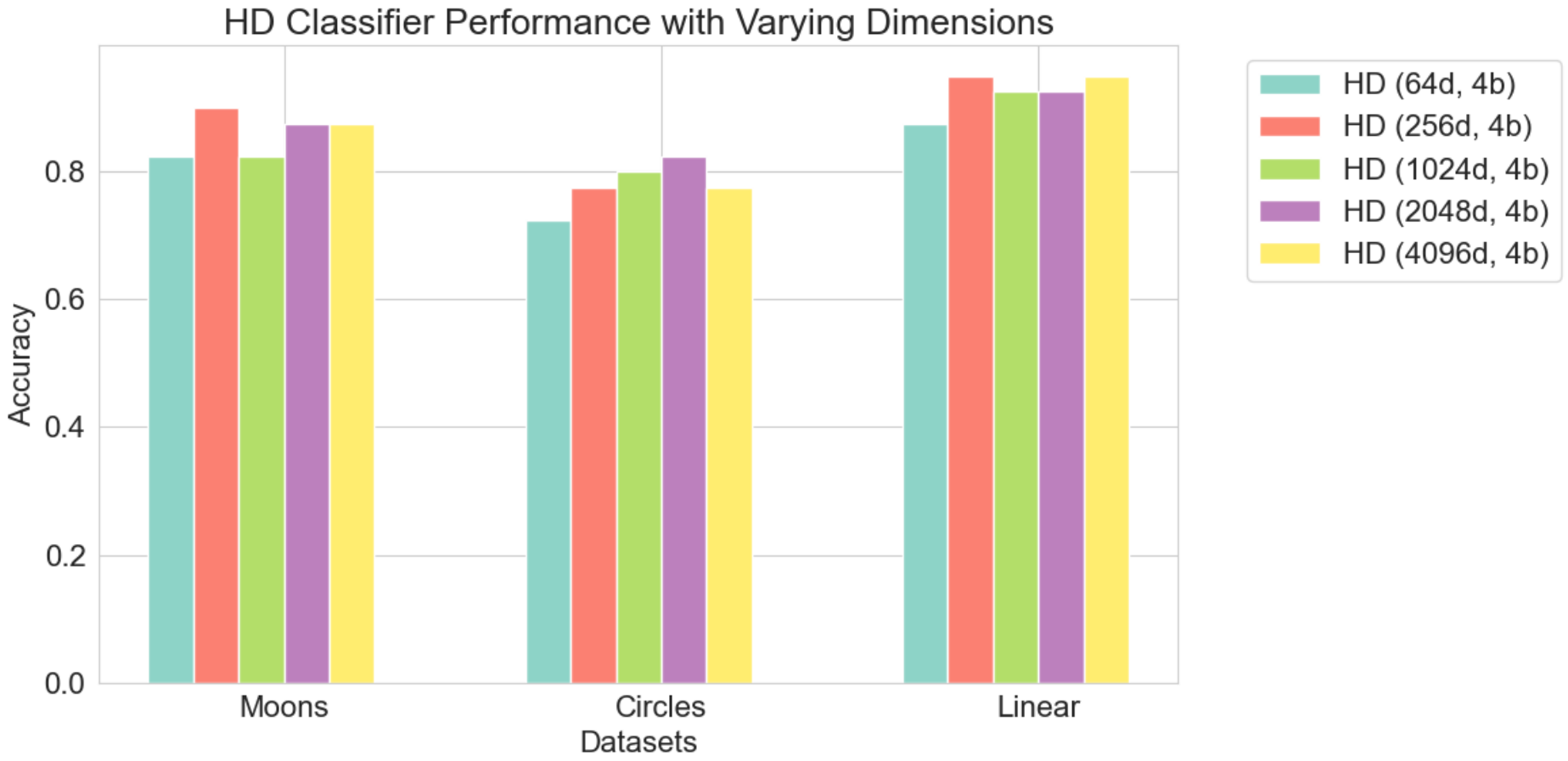
Pros:

- **Hardware Efficient**: Uses simple binary operations (XOR, majority voting) making it ideal for edge computing and resource-constrained environments
- **Competitive Performance**: Achieves 85-95% accuracy, comparable to traditional ML methods like Neural Networks and SVMs
- **Flexible Training**: Works well with varying amounts of data (20-500 samples) and supports online/streaming training

Cons:

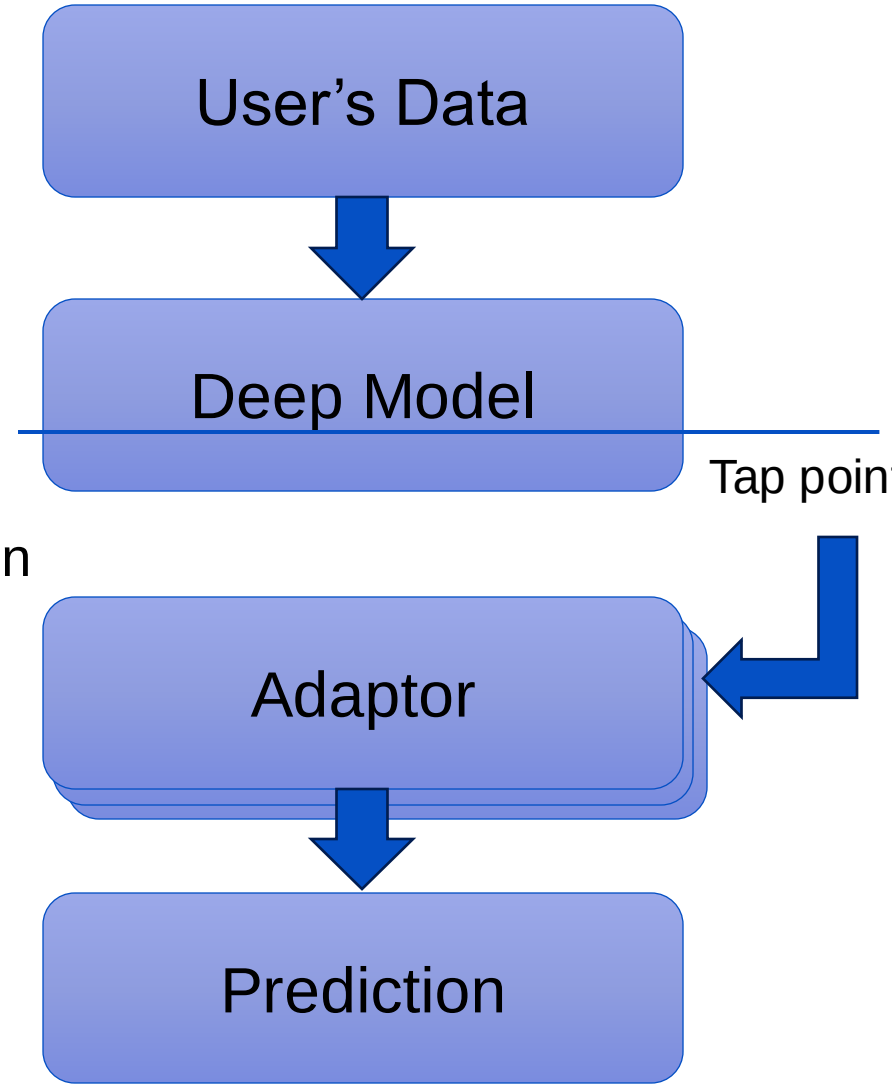
- **Limited Architecture**: Cannot automatically learn hierarchical features like deep learning, requiring feature engineering or deep feature extractors
- **Parameter Sensitive**: Performance heavily depends on tuning hypervector dimensions, quantization bits, and training sample size
- **Dataset Dependent**: Performance varies significantly across different types of datasets (e.g., weaker on circular patterns)

Recap from Notebook



Tutorial Outline

1. **Notebooks on Hyperdimensional Computing (HD)**
   - Introduction to HD computing
   - **Domain Adaptation to new classes and new modality**
2. Slides on Case studies
   - Domain Adaptation in object detection
   - Domain Adaptation in video activity recognition
   - Application to retrieval in Retrieval Augmented Generation
3. Optimize the tap point for the adaptor
   - When do I need an adaptor? OOD Detection using HD
   - Where do I put the tap? Some theory



Using CLIP

- CLIP embeds images and text to 512 dimensional vector
- Given image X_{img} and text X_{text}
- $f_X = CLIP(X_{img})$
- $f_i = CLIP(y_{text})$
- Similarity $s_i = \cos(f_X, f_i)$ for $i = 1, \ldots, n$ (these are the confidences per class)
- CLIP is trained to have high s_i for an image and its corresponding caption

```
!pip install transformers
from transformers import CLIPProcessor, CLIPModel
clip_model = CLIPModel.from_pretrained("openai/clip-vit-base-patch32")
clip_processor = CLIPProcessor.from_pretrained("openai/clip-vit-base-patch32")
```

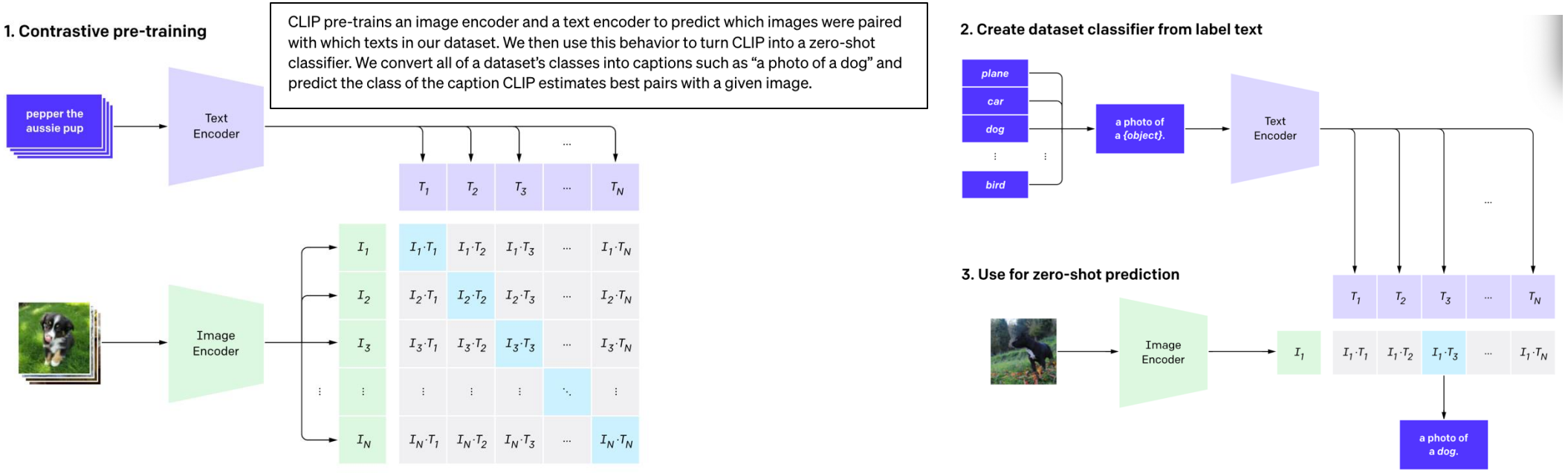
```
inputs = clip_processor(text=[f"a photo of a {label}" for label in labels], images=img, return_tensors="pt", padding=True)
outputs = clip_model(**inputs)
```

```
# 512 dim embedding
print(outputs.image_embeds, outputs.text_embeds)
# Cosine similarity
print(outputs.logits_per_image)
```

Background: CLIP Vision-Language Model

- CLIP is a deep model that can encode images and/or text to a vector space that captures semantic similarity
- In the training domain, CLIP will produce correlated text and image embeddings – zero-shot will work

- In a different domain, the text and visual embeddings may not be similar
- We will see how to align the embeddings in HD space, aligning the new domain images with the corresponding captions



https://huggingface.co/sentence-transformers/clip-ViT-B-32

Domain Adaptation using HD on top of CLIP Vision-Language Model

Given few examples (X, y) in the user's domain / new domain

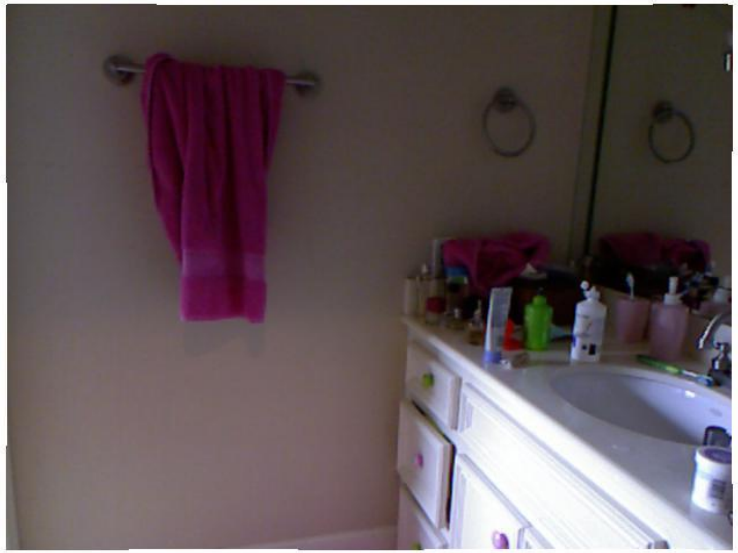
1. $E[y] = [\mathbf{0}]^D$ for each class y
2. For each example (X, y): (Only one iteration through the data)
   a. $T = [\mathbf{0}]^D$
   b. $\phi_{img} = CLIP(X)$
   c. $\phi_{text} = CLIP(y)$
   d. $\phi = \phi_{img} \oplus \phi_{text}$ (XOR)
   e. For each $\phi_i \in \phi$: (512 dims CLIP embedding)
      i. V_i = encode-to-HD(ϕ_i)
      ii. P_i = encode-to-HD(i)
      iii. $T \mathrel{+}= (V_i \oplus P_i)$ (XOR)
   f. Normalize T: $T = \frac{T}{512}$
   g. Threshold t: $t[idx] = 1 \; if \; T[idx] \geq 0.5 \; else \; 0$
   h. Add to exemplar for class y: $E[y] \mathrel{+}= t$ (Online step: on-the-fly training)
3. Normalize and Threshold exemplar: $\mathbf{1}(\frac{E[y]}{(\#\, examples\; of\; y)} \geq 0.5)$

https://github.com/aswin-raghavan/icce-tutorial-2025-domain-adaptation

Notebook: Domain adaptation from RGB to Depth images

See images folder in the repo

- Notebook: multimodal_domain_adaptation_using_HD.ipynb
- Classification problem based on the NYU_Depth_V2 dataset
- Paired RGB and LIDAR depth images
- Each image is labelled by the type of indoor room
  - 10 labels e.g. kitchen, dining room, bathroom etc.

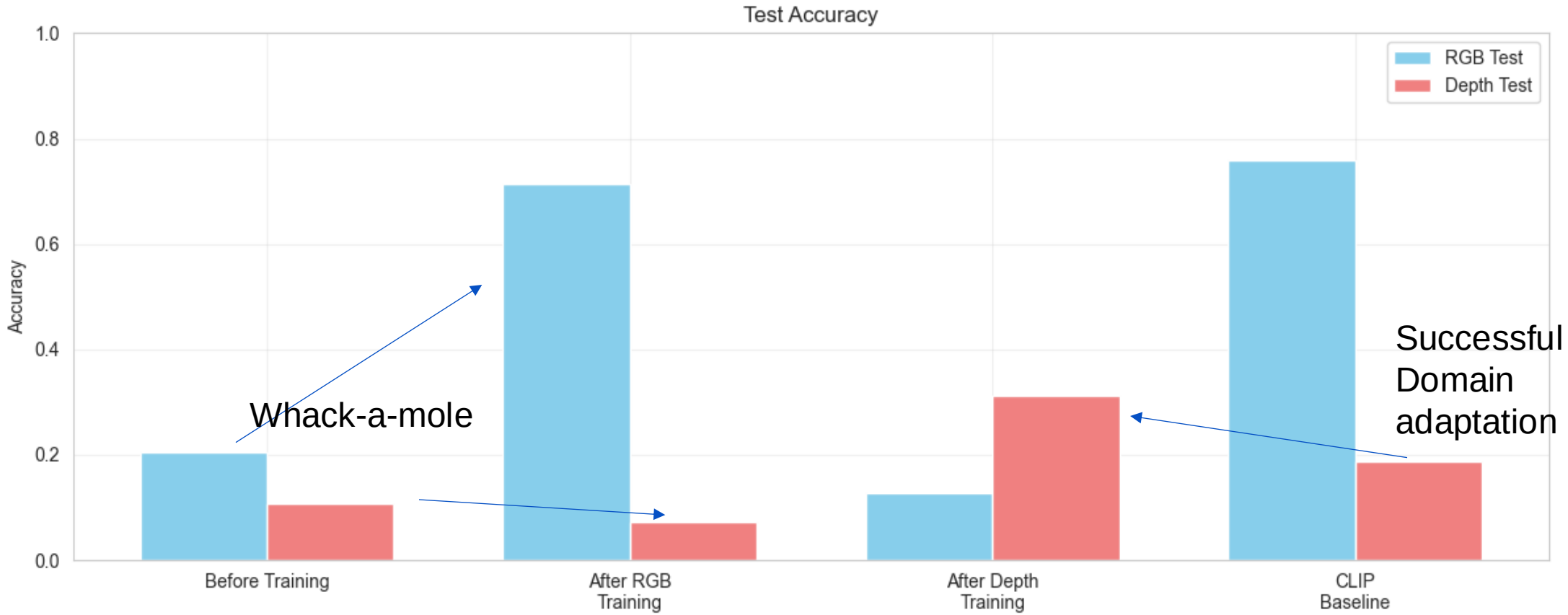


CLIP prediction: “bathroom”

Let us try domain adaptation using HD

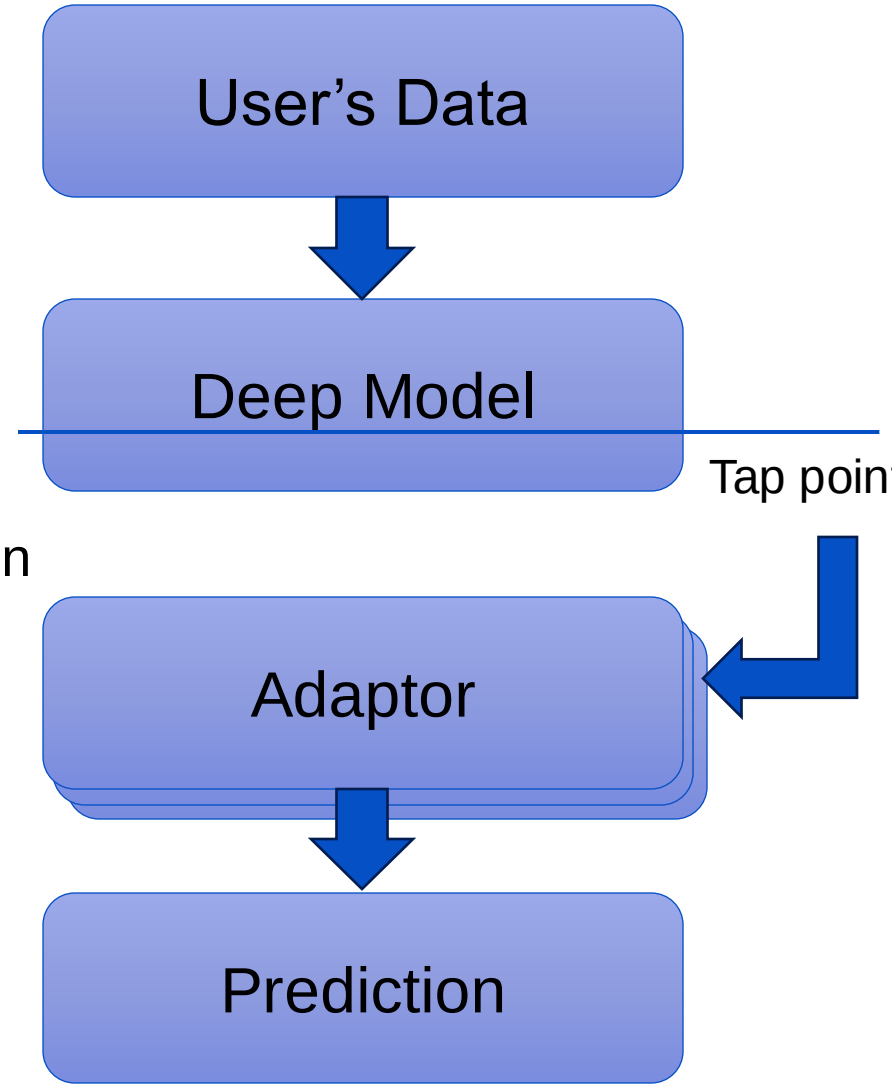
CLIP prediction (top-3):
“basement”, “classroom”, “office”

Recap from Notebook

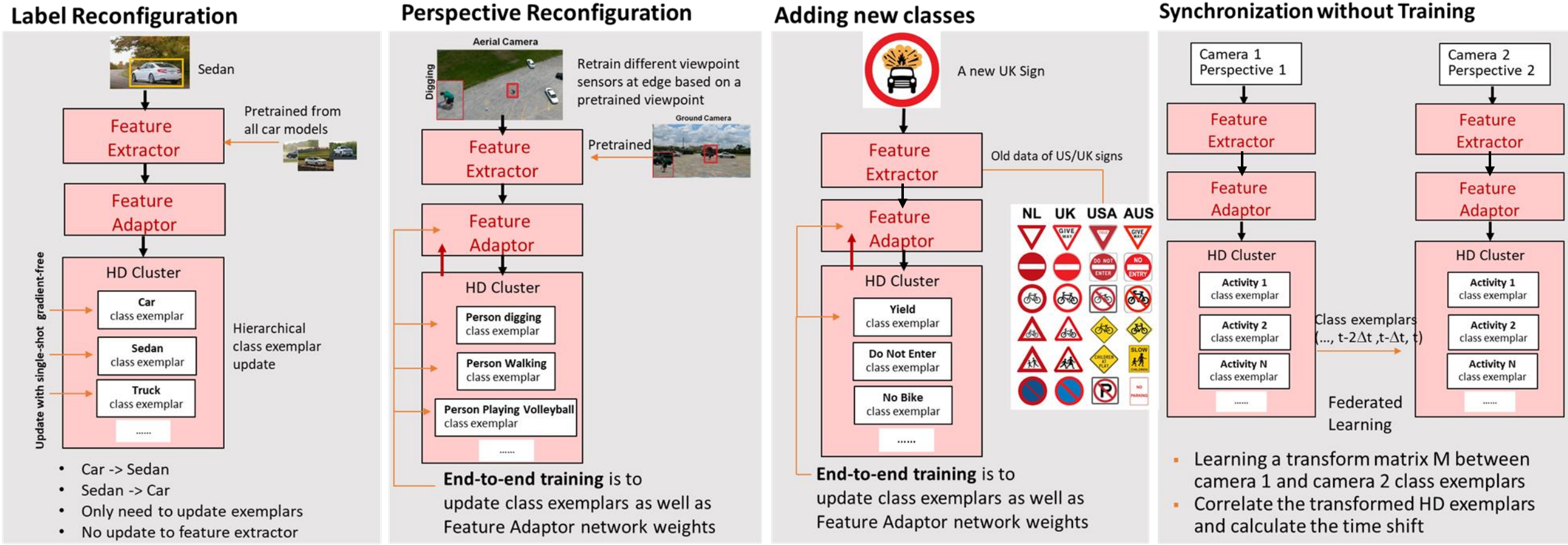


Tutorial Outline

1. Notebooks on Hyperdimensional Computing (HD)
   - Introduction to HD computing
   - Domain Adaptation to depth images
2. **Slides on Case studies**
   - **Domain Adaptation in object detection**
   - Domain Adaptation in video activity recognition
   - Application to retrieval in Retrieval Augmented Generation
3. Optimize the tap point for the adaptor
   - When do I need an adaptor? OOD Detection using HD
   - Where do I put the tap? Some theory

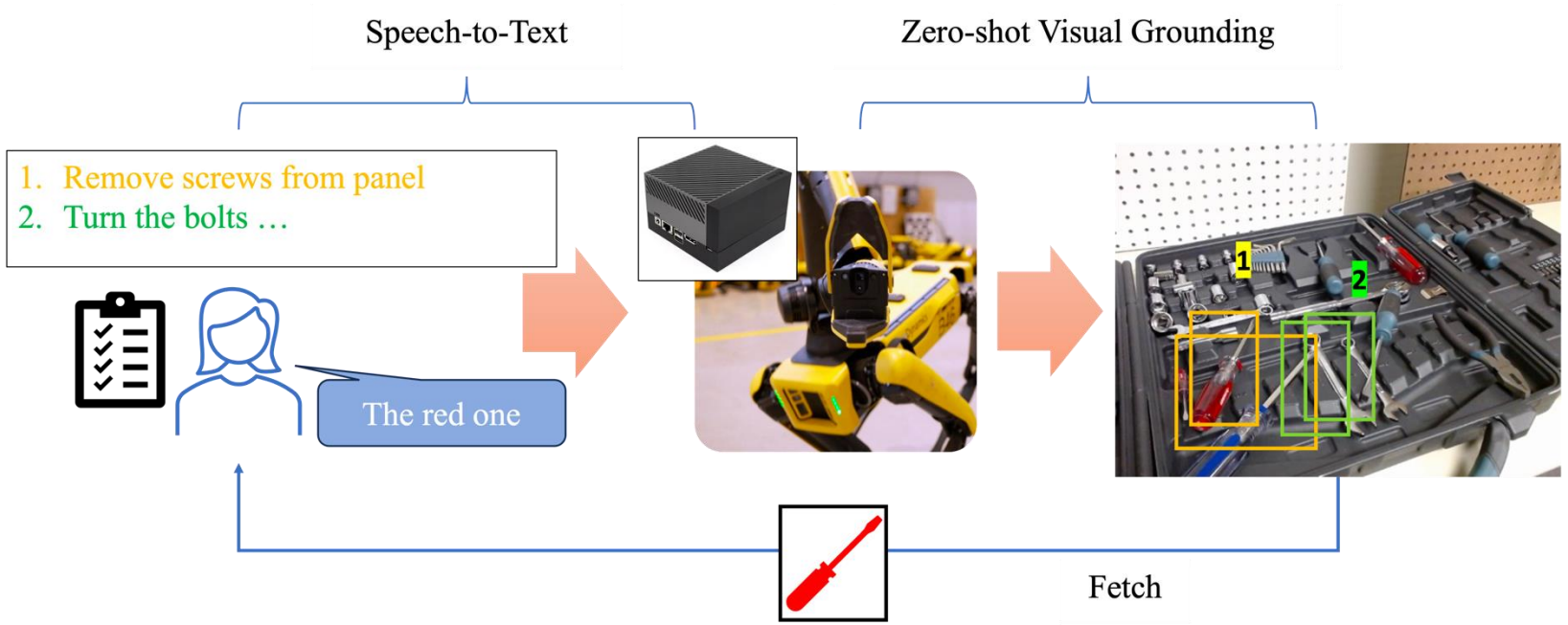


Case Studies: Online Adaptation Scenarios



(a) (b) (c) (d)

Tools Detection Problem

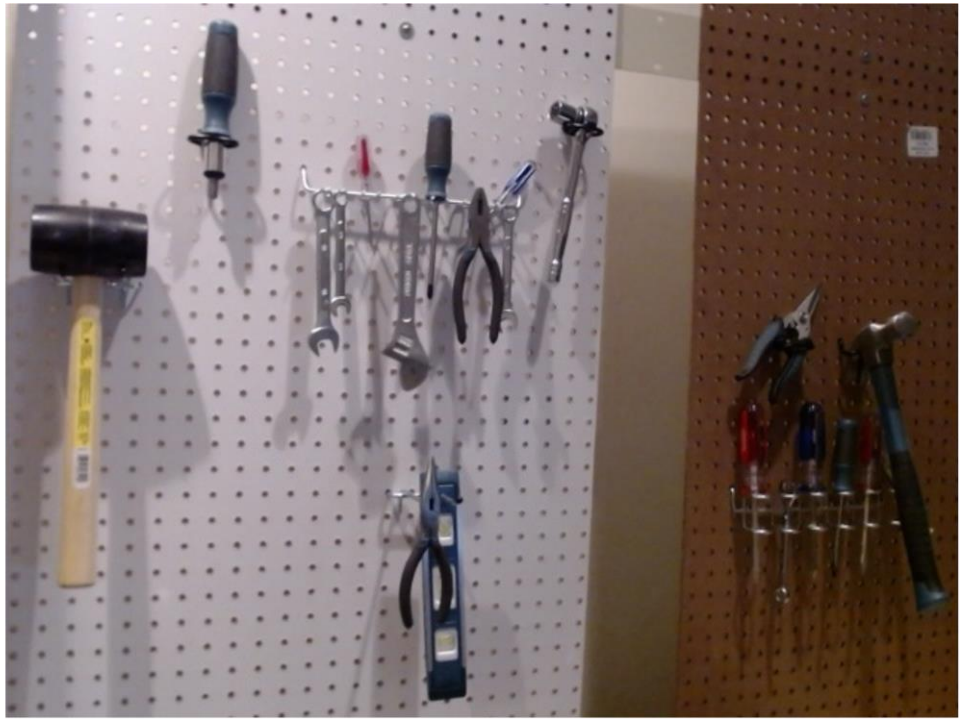


General purpose object detectors do not have different tools in their databases. Adaptation:

1. Add new classes: Need to adapt to add new labels for different tools
2. Adapt to understand the purposes of different tools
3. Adapt to depth images of tools

Lab Setup

- Voice capture, speech-to-text, and object detection runs on a NVIDIA Jetson Orin
- Test laptop for debug



Adaptation of Object Detectors using HD

Input: pre-trained two-stage object detector to fixed object classes

- 2-Stage detection: Stage-1: put boxes around all objects. Stage-2: classification of each box
- E.g., trained to detect 80 MSCOCO classes

Zero-shot Visual Grounding: detect any object (open-world) from text query

- Subclass or superclass
- Identify an object from its function description
- Objects with similar purposes
- Objects by parts

Few-shot learning few descriptive captions

- Descriptive names for objects
- Split a previous class into two
- Add a new class

Live System Demonstration at AAAI-24 Demonstration Program
"Zero-shot object detection dataset on target categories" Jun Hu, Phil Miler, Aswin Raghavan et. al. in the proceedings of AAAI-24

Tools Detection Problem: Add new classes

- As shown in the first notebook, HD needs good feature representations for the objects of interest.
- Consider a two-stage object detection approach: first stage proposes boxes in the image, potentially around the objects of interest.
- First, we retrain the object proposal network for tools e.g., using the ALET dataset.

- In order to improve the generalization between domains, we incorporate text (from audio) as an additional input to the classifier, e.g. combining the appearance of objects (e.g., say a shovel) with its purpose (e.g., to dig a hole) can help generalize to new tools with similar appearance on a part of the tool (e.g., a spade can also dig a hole).We use a vision and language model called CLIP. CLIP can embed images and text in a vector space such that an image and its caption will have high cosine similarity.

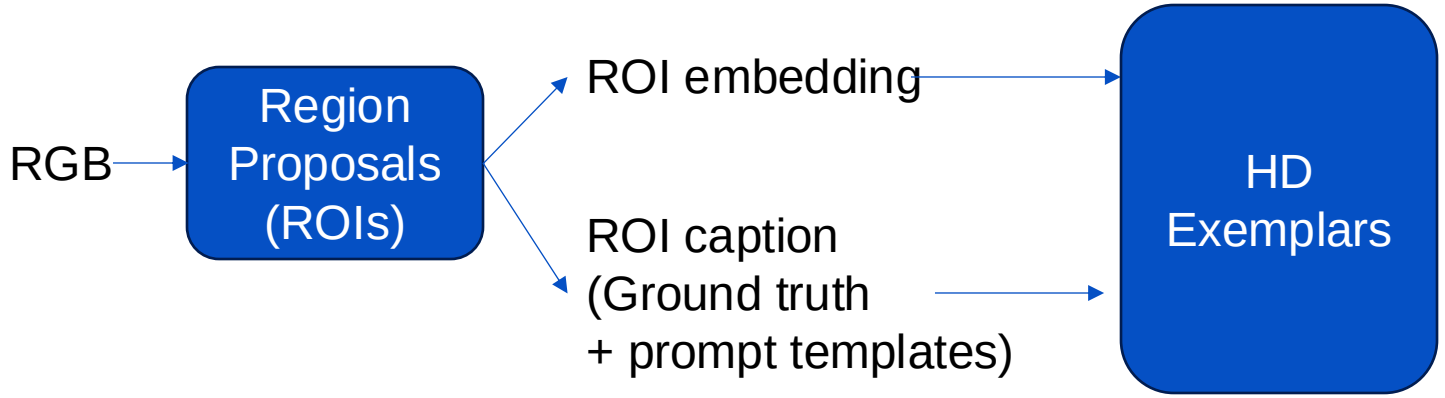
- Our method applies CLIP to each proposal (the cropped image). Each proposal is captioned with either the name of the tool, or the purpose of the tool, or “background” for ROIs with insufficient overlap with ground truth boxes.

https://github.com/metu-kovan/METU-ALET

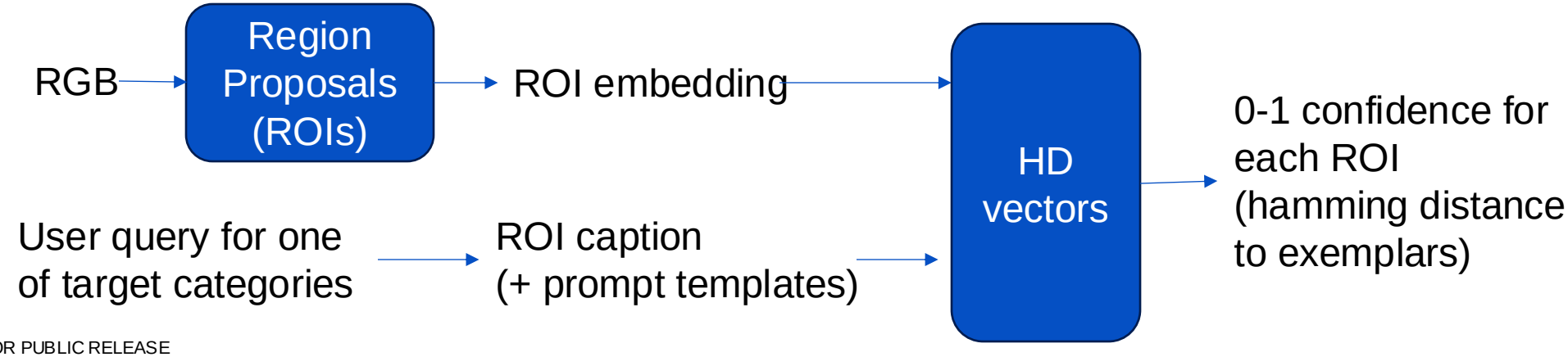
Zero Shot Object Detection and Visual Grounding

ROI Region of Interest

Pre-Training on object detection dataset on source categories



Zero-shot object detection dataset on target categories



Pseudocode

Feature extraction with CLIP
Then Encode to HD

XOR Bind

Bundle

Algorithm 1: Training / Inference in Target Domain

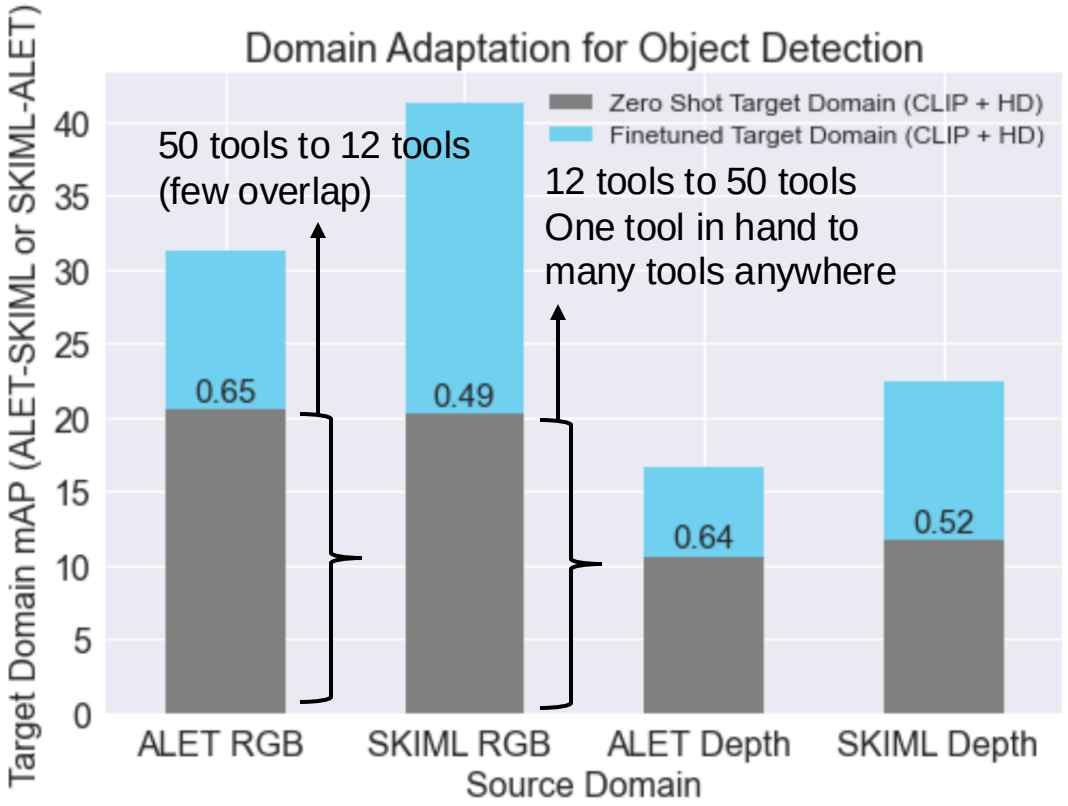
Input: RGB or depth images $\mathcal{I}$
Input: (In training) Labelled bounding boxes $\mathcal{B}$
Parameter: Size of HD vectors D
Output: (Training) HD Exemplars E for source categories

Initialize training
$e \leftarrow \mathbf{0}^D$ for each $e \in E$
# Loop if training
for each training image I **do**
  Boxes $B \leftarrow$ RegionProposals(I)
  for each box $b \in B$ **do**
    $\phi_b \leftarrow$ CLIP(b); $v_b \leftarrow$ EncodeHD(ϕ_b)
    # In Training (inference caption comes from user)
    caption $\leftarrow b$.label from $\mathcal{B}$ else "background"
    $\phi_Q \leftarrow$ CLIP(caption); $v_Q \leftarrow$ EncodeHD(ϕ_Q)
    $V \leftarrow v_q \oplus v_b$
    # In Training
    E[label] $\leftarrow E$[label] + V; n[label] $\leftarrow n$[label] + 1
    # In inference
    $d \leftarrow \frac{1}{D}||V - e||_1$ for each exemplar $e \in E$
    Label b with caption if $\max(Softmax(\text{1-d})) > \epsilon$
  end for
end for
# In training
$E[i] \leftarrow \mathbf{1}_D(\frac{E[i]}{n[i]} > 0.5)$ for each label i
return E

Adaptation of Object Detectors using HD

Live System Demonstration at AAAI-24 Demonstration Program
“Zero-shot object detection dataset on target categories”
Jun Hu, Phil Miler, Aswin Raghavan et. al. to appear in the proceedings of AAAI-24

Zero Shot Object Detection and Visual Grounding



Zero-shot mAP	Vision	Vision+Language
ALET to SKIML	9.5	20.5
SKIML to ALET	5	20

Eval. purpose queries	mAP@0.5	@0.5:0.95
Exemplars from tool name	27.61	18.43
Exemplars from purposes	43.57	28.83

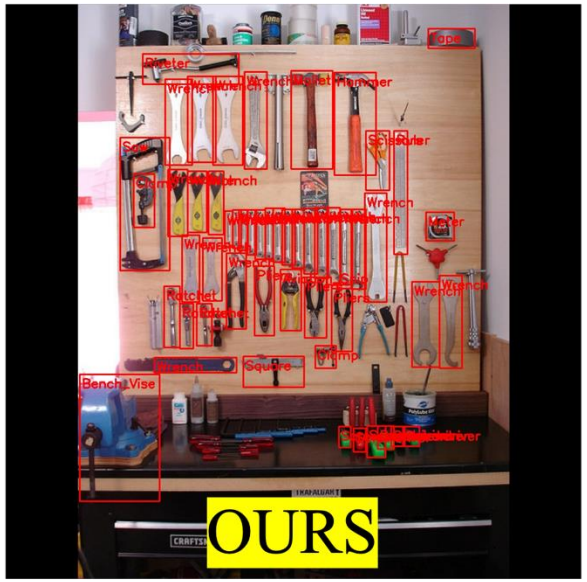
The HD-based method is able to perform reasonably fast adaptation directly on the edge device (Orin)

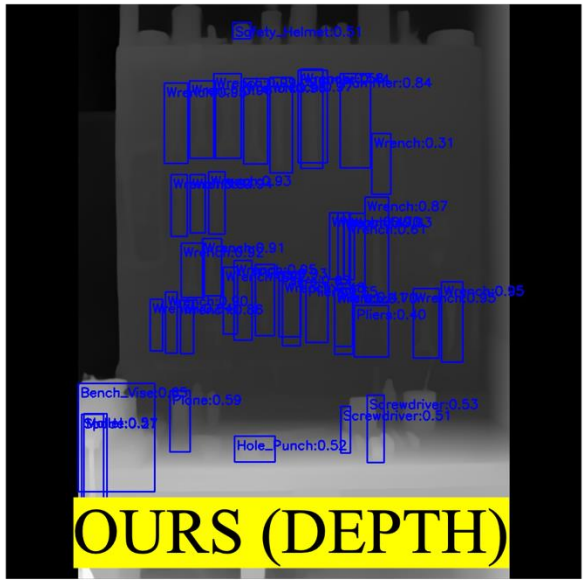
Runtime (s/image)	Region Proposals	CLIP	HD
A5000 GPU	4.38	2.30	0.52
NVIDIA Orin	18.08	6.85	1.95

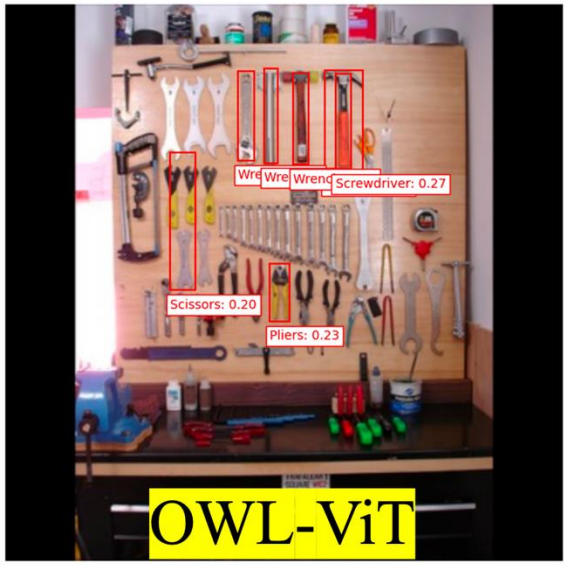
Two types of images RGB/Depth
Two datasets (ALET, SKIML) for source or target
Gray bar: zero-shot, HD is trained using source domain dataset
Blue bar: Only the HD exemplars are adapted using target domain data. Train-test split of 75%-25%.

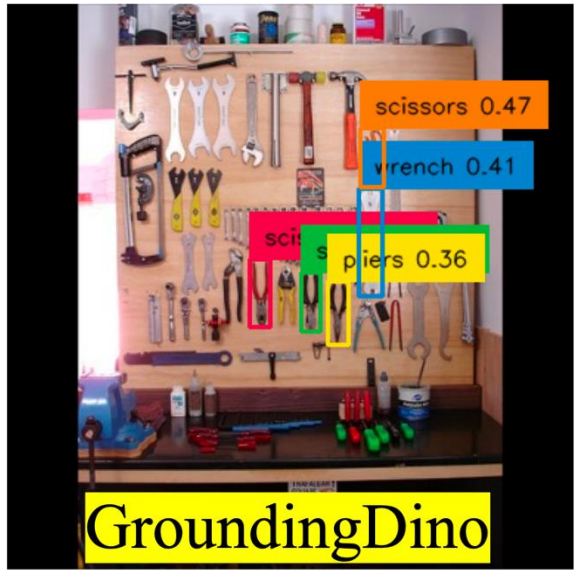
Region Proposals and CLIP need to be run only once per image and user query

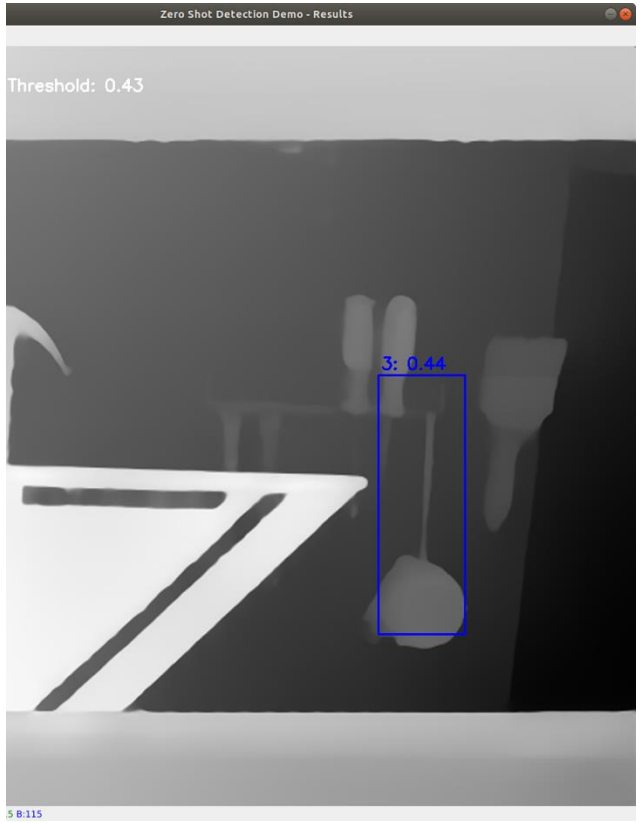
Adaptation of Object Detectors using HD

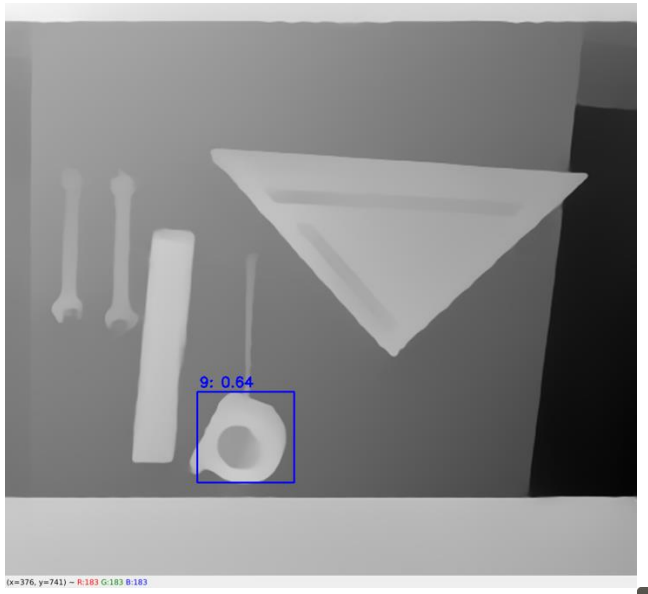


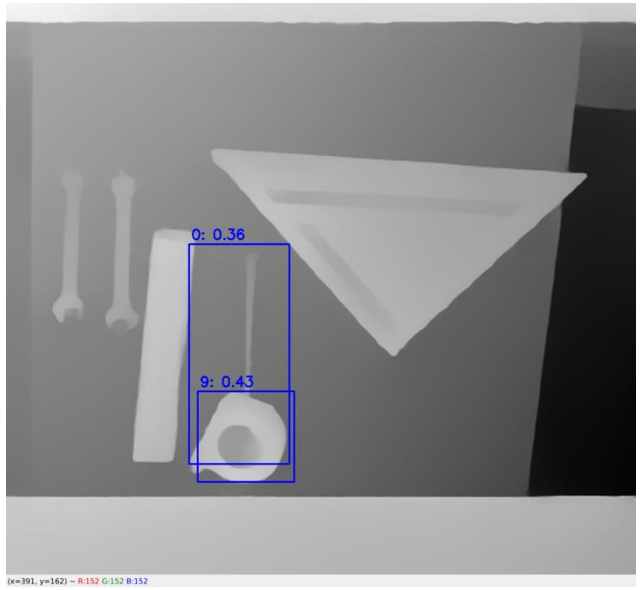


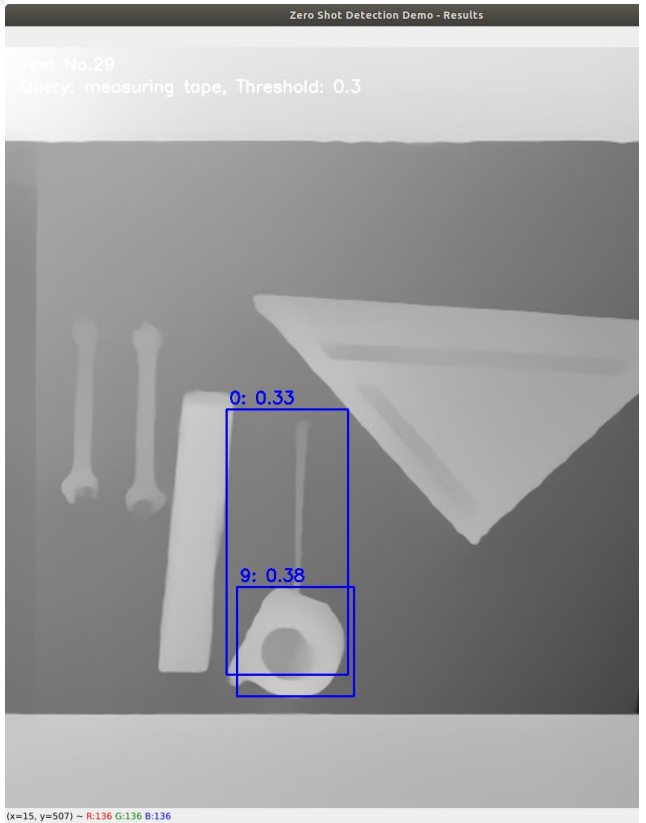


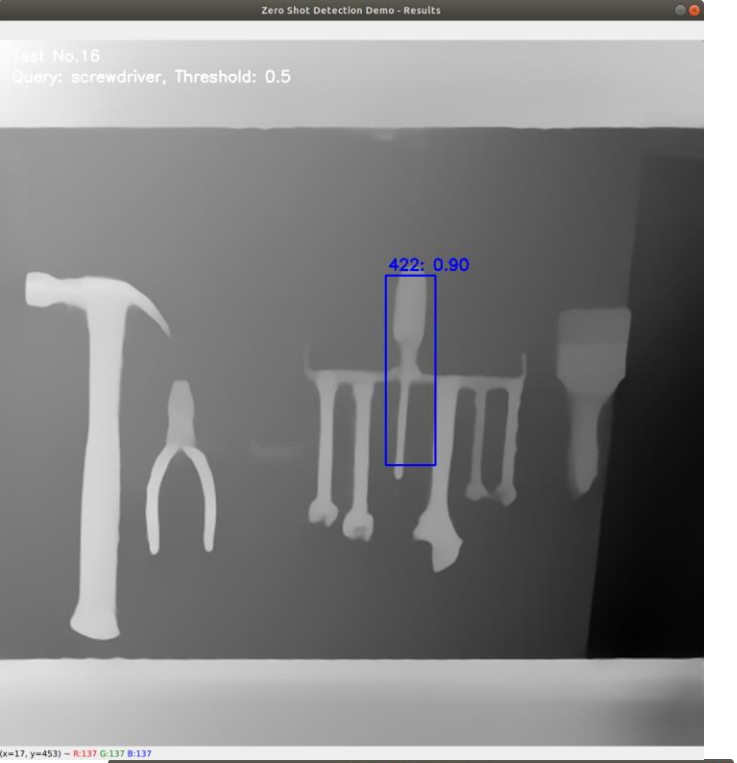


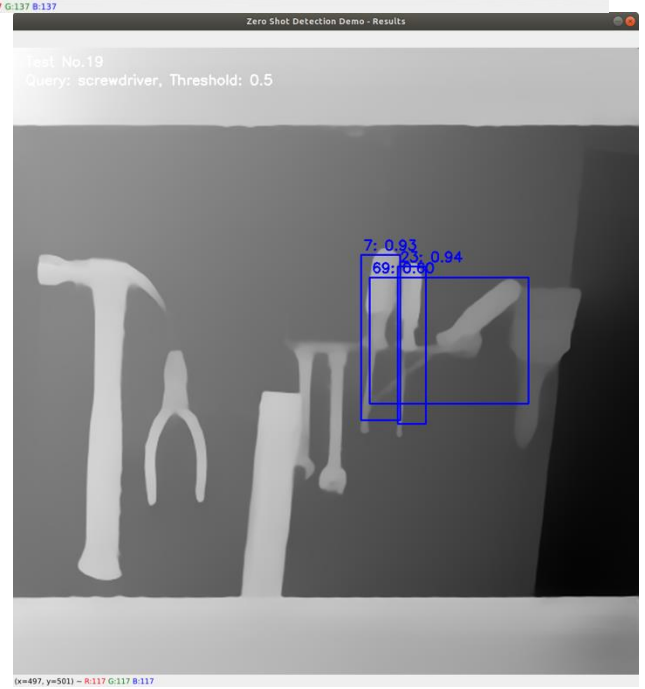


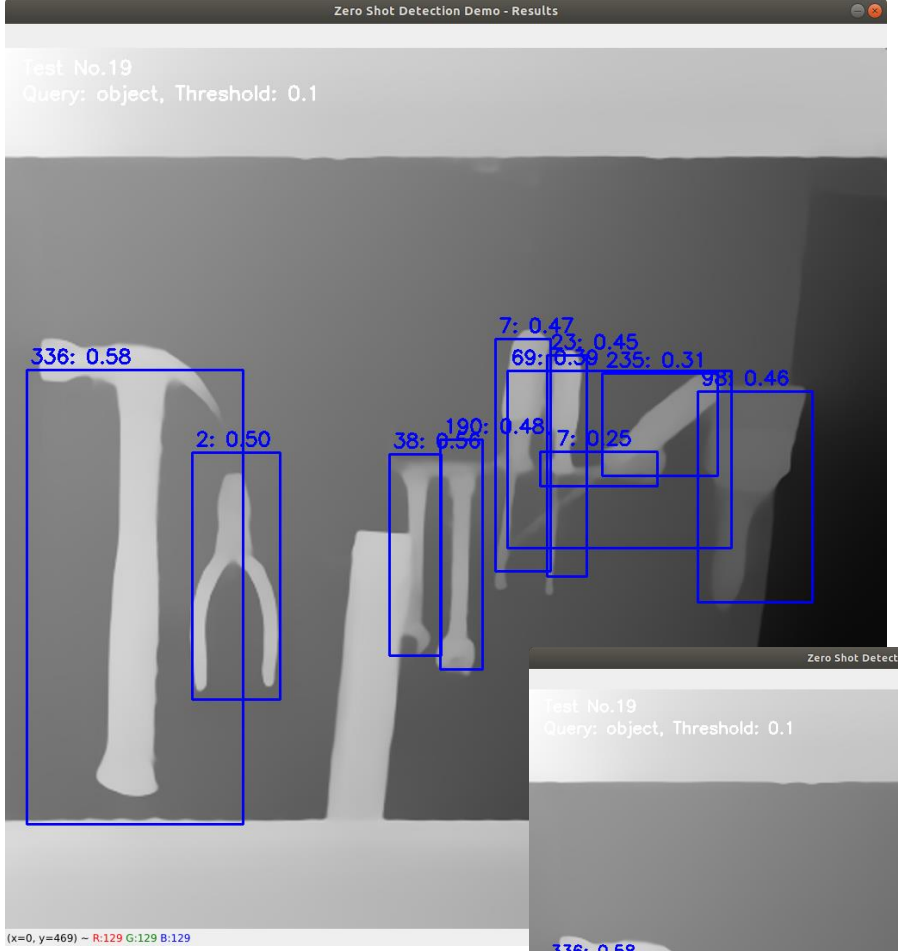


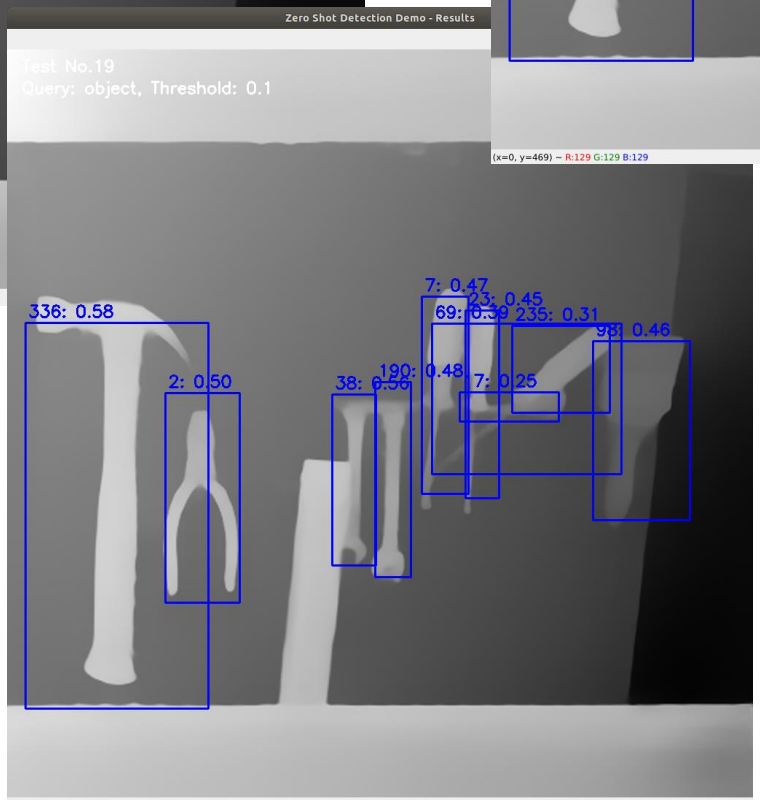


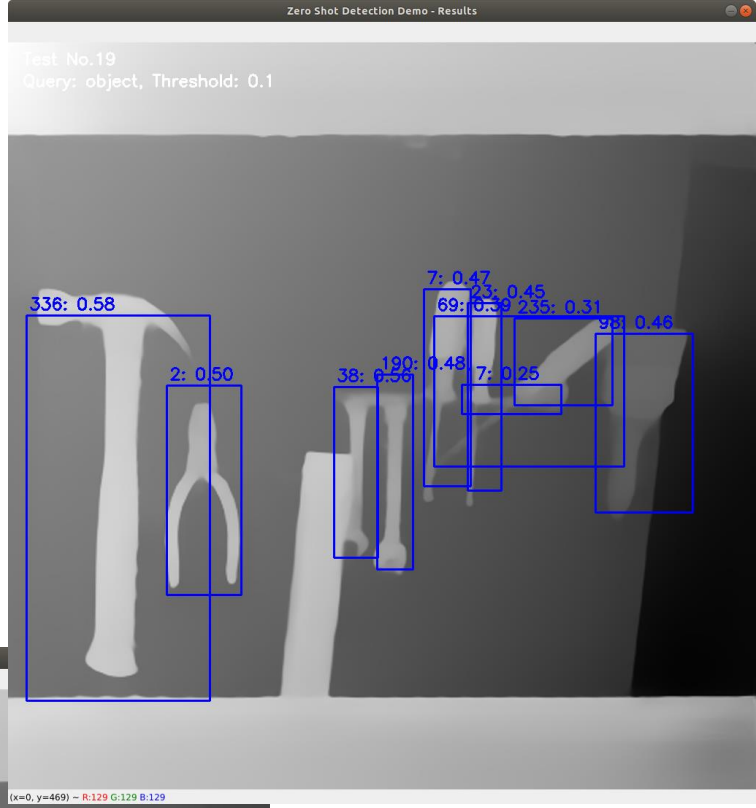


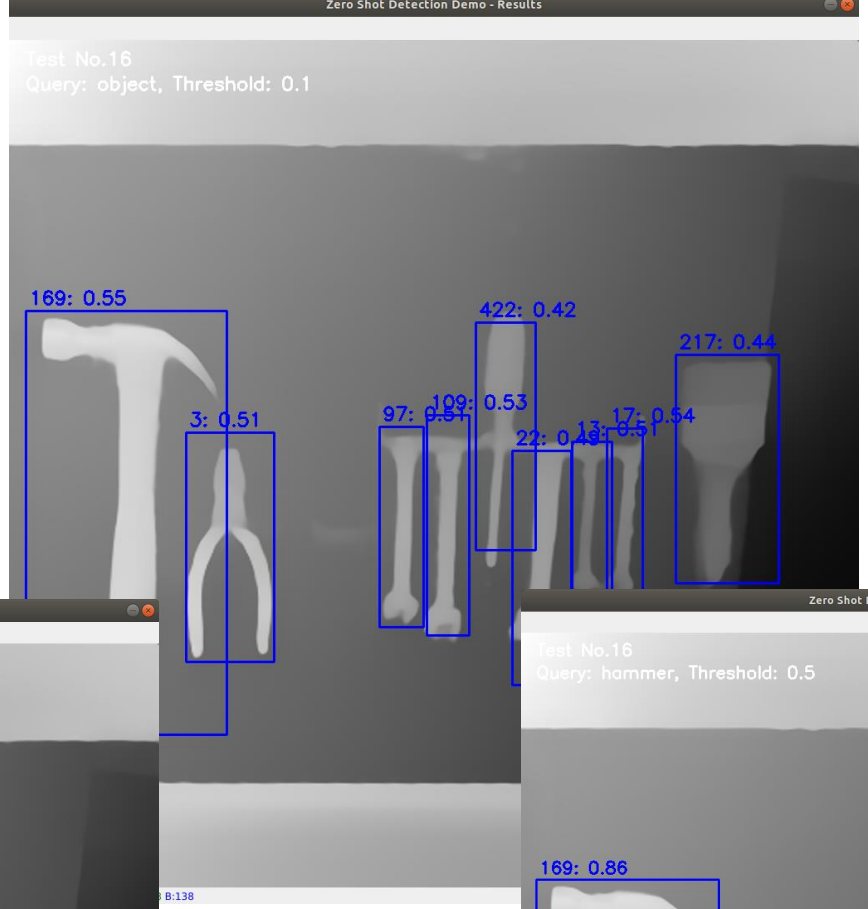

Zero Shot Detection Demo - Results
Test No.16
Query: screwdriver, Threshold: 0.5
422: 0.90
(x=17, y=453) ~ R:137 G:137 B:137

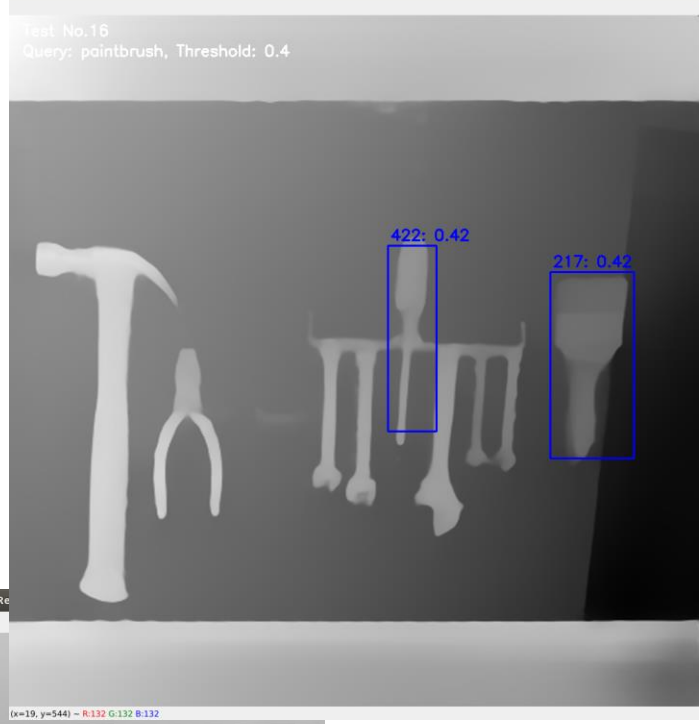

Zero Shot Detection Demo - Results
Test No.19
Query: screwdriver, Threshold: 0.5
7: 0.93
23: 0.94
69: 0.60
(x=497, y=501) ~ R:117 G:117 B:117

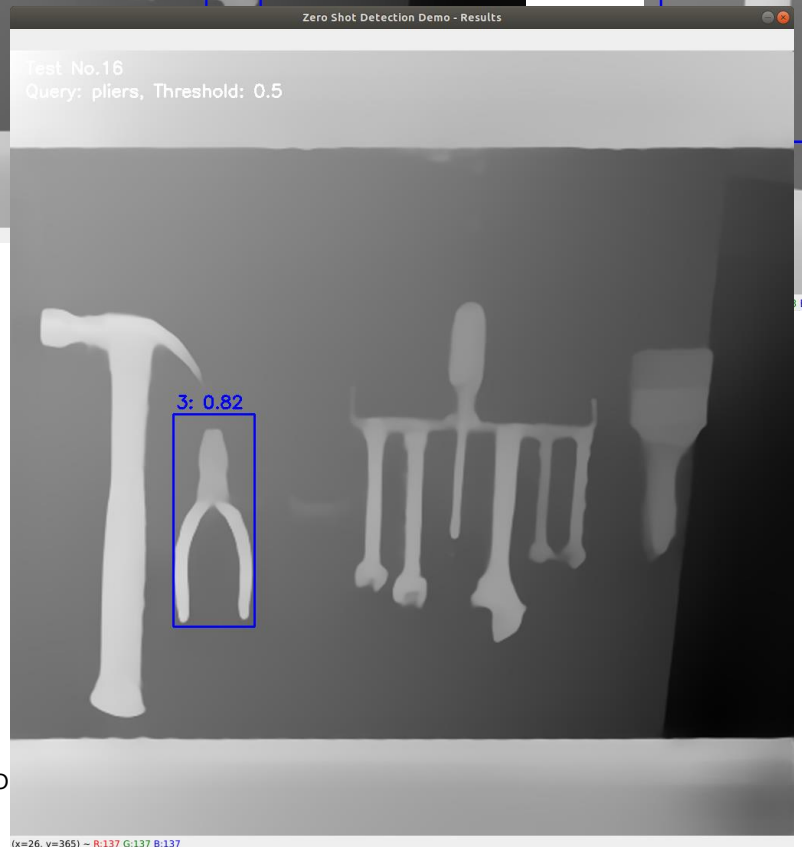

Zero Shot Detection Demo - Results
Test No.19
Query: object, Threshold: 0.1
336: 0.58
2: 0.50
38: 0.56
190: 0.48
7: 0.47
23: 0.45
69: 0.39
235: 0.31
7: 0.25
98: 0.46
(x=0, y=469) ~ R:129 G:129 B:129

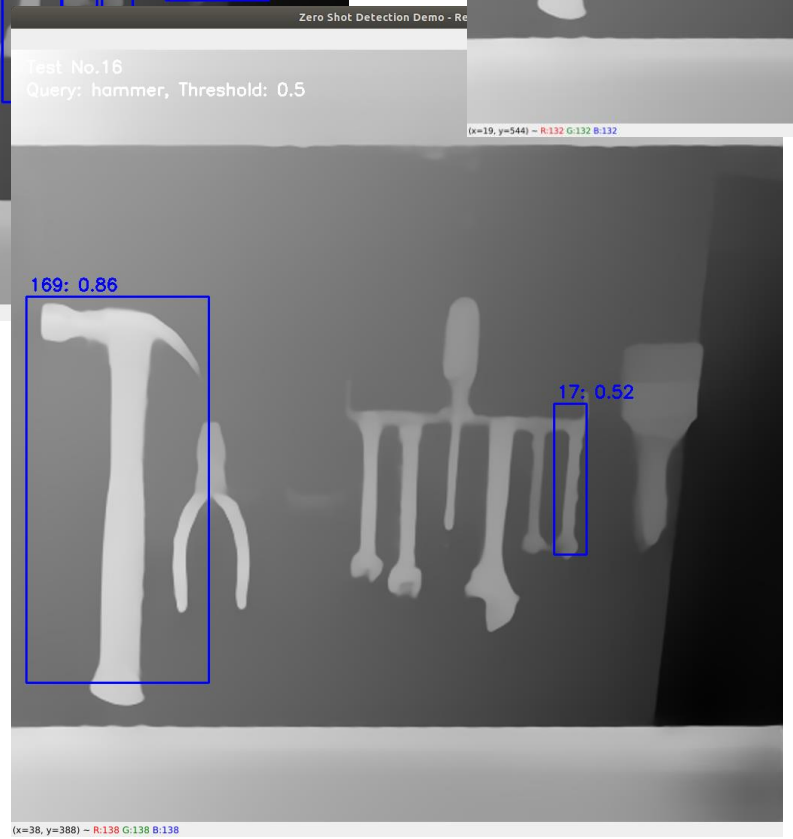

Zero Shot Detection Demo - Results
Test No.19
Query: object, Threshold: 0.1
336: 0.58
2: 0.50
38: 0.56
190: 0.48
7: 0.47
23: 0.45
69: 0.39
235: 0.31
7: 0.25
98: 0.46


Zero Shot Detection Demo - Results
Test No.19
Query: object, Threshold: 0.1
336: 0.58
2: 0.50
38: 0.56
190: 0.48
7: 0.47
23: 0.45
69: 0.39
235: 0.31
7: 0.25
98: 0.46
(x=0, y=469) ~ R:129 G:129 B:129









DISTRIBUTIO

k8s dataset projects obj-detect-hd code test alet

Few Shot Detection Demo - Results

edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

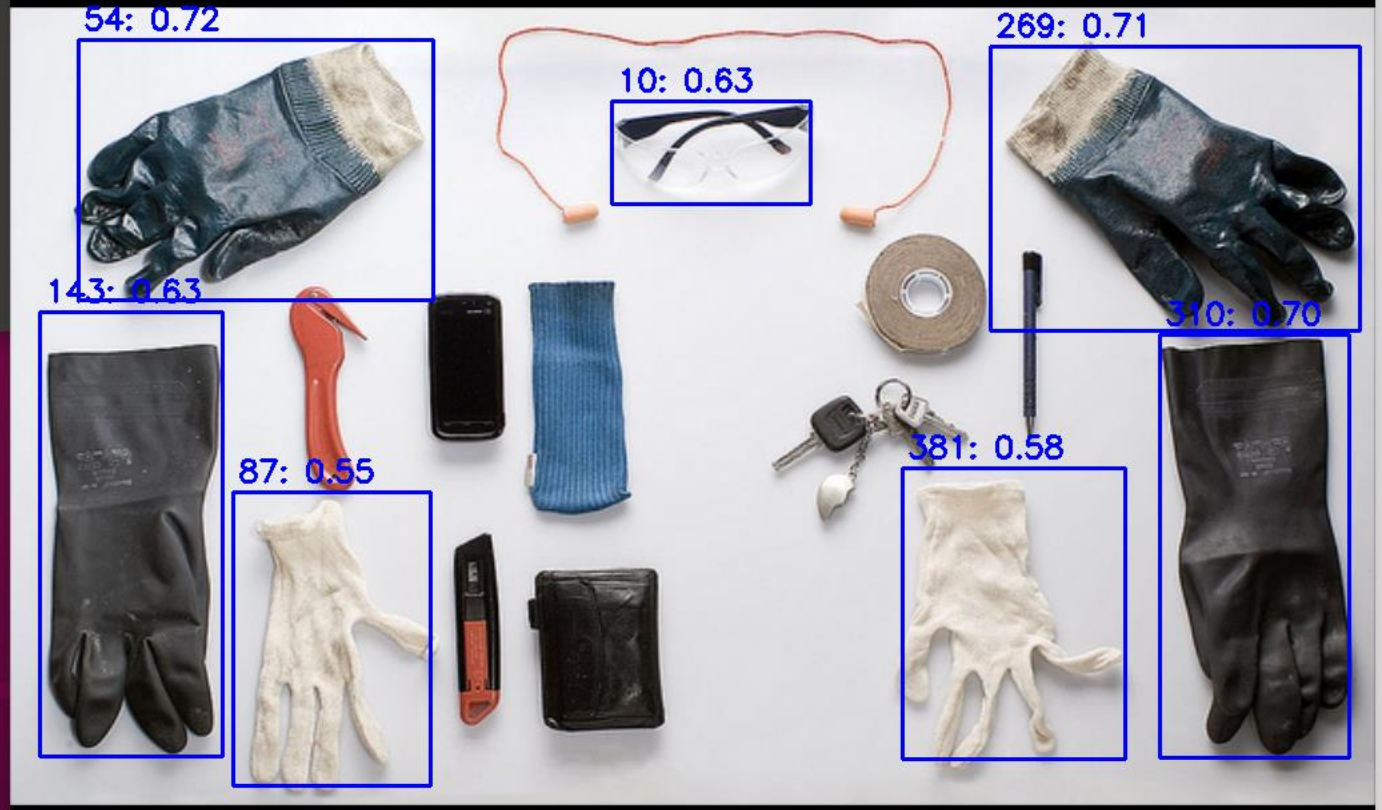
```
2 [305 674 732 753]
Chisel 0.2635070504846152
DObject: rive stakes into the ground at 3 ft intervals
Threshold(0.0-1.0): .4
Object: Drive stakes into the ground at 3 ft intervals
Threshold(0.0-1.0): .39
104 [297  82 775 429]
Mallet 0.3970841351880859
Object:
Threshold(0.0-1.0):
31 [ 463  834  659 1000]
Safety_Headphones 0.5144689715604017
103 [365 619 750 847]
Mallet 0.3672264271997169
9 [138 336 462 522]
Flashlight 0.2402338082308773
2 [470 833 549 929]
Safety_Headphones 0.19514174500886655
1 [469 216 587 385]
Hex_Key 0.10280410612297937
Object:
Threshold(0.0-1.0):
24 [709 443 941 602]
Hammer 0.6116054439643513
19 [674 367 938 498]
Hammer 0.5623510635057481
48 [261 495 279 524]
Safety_Helmet 0.44146990357539934
77 [213 586 308 656]
Mallet 0.35671634941147057
177 [406 376 463 499]
Plane 0.3241173016215959
62 [259 584 345 631]
Chisel 0.12052508791598279
Object: Dig a trench
Threshold(0.0-1.0): .1
24 [709 443 941 602]
Hammer 0.4656193317518804
19 [674 367 938 498]
Hammer 0.4911007953517742
48 [261 495 279 524]
Safety_Helmet 0.3296176061823637
77 [213 586 308 656]
Mallet 0.28516201496300475
177 [406 376 463 499]
Plane 0.272394318687157
62 [259 584 345 631]
Chisel 0.1819847872958973
DiObject: g a trench
Threshold(0.0-1.0): .45
24 [709 443 941 602]
Hammer 0.4656193317518804
19 [674 367 938 498]
Hammer 0.4911007953517742
Object:
```

Activities Terminal Wed 14:41

projects obj-detect-hd code test alet annotations

Few Shot Detection Demo - Results

sri/test2/images/4814649030_6aaf41672f_z.jpg
Query: identify the safety equipment, Threshold: 0.55



edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

```
381 [558 669 776 836]
Gloves 0.5758614050789848
143 [441  22 774 159]
Gloves 0.6325549106083678
87 [576 167 796 315]
Gloves 0.5546945265885779
701 [599 329 744 372]
Knife 0.5196754754027464
3 [381 634 474 731]
Tape 0.37060855505390106
55 [533 616 598 646]
Screwdriver 0.4505480386563291
11 [414 207 581 270]
Pliers 0.4630901862165025
35 [416 304 540 374]
Stapler 0.217631566017123
47 [401 757 543 776]
Pencil 0.28157985677769864
141 [624 388 766 496]
Hole_Punch 0.2823373101252163
idObject: entify the safety equipment
Threshold(0.0-1.0):
54 [237  51 432 317]
Gloves 0.7166437438745759
310 [ 458  862  775 1004]
Gloves 0.7044857453730431
269 [ 242  735  455 1012]
Gloves 0.7069453050600257
10 [283 451 360 600]
Safety_Glass 0.6345570076511964
381 [558 669 776 836]
Gloves 0.5758614050789848
143 [441  22 774 159]
Gloves 0.6325549106083678
87 [576 167 796 315]
Gloves 0.5546945265885779
701 [599 329 744 372]
Knife 0.5196754754027464
Object: identify the safety equipment
Threshold(0.0-1.0): .55
54 [237  51 432 317]
Gloves 0.7166437438745759
310 [ 458  862  775 1004]
Gloves 0.7044857453730431
269 [ 242  735  455 1012]
Gloves 0.7069453050600257
10 [283 451 360 600]
Safety_Glass 0.6345570076511964
381 [558 669 776 836]
Gloves 0.5758614050789848
143 [441  22 774 159]
Gloves 0.6325549106083678
87 [576 167 796 315]
Gloves 0.5546945265885779
Object:
```

```
edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code
File  Edit  View  Search  Terminal  Help
Pencil 0.28157985677769864
141 [624 388 766 496]
Hole_Punch 0.2823373101252163
idObject: entify the safety equipment
Threshold(0.0-1.0):
54 [237  51 432 317]
Gloves 0.7166437438745759
310 [ 458  862  775 1004]
Gloves 0.7044857453730431
269 [ 242  735  455 1012]
Gloves 0.7069453050600257
10 [283 451 360 600]
Safety_Glass 0.6345570076511964
381 [558 669 776 836]
Gloves 0.5758614050789848
143 [441  22 774 159]
Gloves 0.6325549106083678
87 [576 167 796 315]
Gloves 0.5546945265885779
701 [599 329 744 372]
Knife 0.5196754754027464
Object: identify the safety equipment
Threshold(0.0-1.0): .55
54 [237  51 432 317]
Gloves 0.7166437438745759
310 [ 458  862  775 1004]
Gloves 0.7044857453730431
269 [ 242  735  455 1012]
Gloves 0.7069453050600257
10 [283 451 360 600]
Safety_Glass 0.6345570076511964
381 [558 669 776 836]
Gloves 0.5758614050789848
143 [441  22 774 159]
Gloves 0.6325549106083678
87 [576 167 796 315]
Gloves 0.5546945265885779
Object: before proceeding to the next step, ensure that proper safety equipment is
worn
Threshold(0.0-1.0):
54 [237  51 432 317]
Gloves 0.7145001838083638
310 [ 458  862  775 1004]
Gloves 0.669263404402852
269 [ 242  735  455 1012]
Gloves 0.6939395860626993
10 [283 451 360 600]
Safety_Glass 0.5654049394631466
381 [558 669 776 836]
Gloves 0.5983915073032986
143 [441  22 774 159]
Gloves 0.6443316644129229
87 [576 167 796 315]
Gloves 0.5659020134913517
Object:
```

Activities Terminal Wed 14:42

projects | obj-detect-hd | code | test | alet | annotations

Few Shot Detection Demo - Results

sri/test2/images/4814649030_6aaf41672f_z.jpg
Query: cut the red wire after cutting the blue wire, Threshold: 0.19

edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

```
Safety_Glass 0.6345570076511964
381 [558 669 776 836]
Gloves 0.5758614050789848
143 [441  22 774 159]
Gloves 0.6325549106083678
87 [576 167 796 315]
Gloves 0.5546945265885779
Object: before proceeding to the next step, ensure that proper safety equipment is
worn
Threshold(0.0-1.0):
54 [237  51 432 317]
Gloves 0.7145001838083638
310 [ 458  862  775 1004]
Gloves 0.669263404402852
269 [ 242  735  455 1012]
Gloves 0.6939395860626993
10 [283 451 360 600]
Safety_Glass 0.5654049394631466
381 [558 669 776 836]
Gloves 0.5983915073032986
143 [441  22 774 159]
Gloves 0.6443316644129229
87 [576 167 796 315]
Gloves 0.5659020134913517
Object: cut the red wire after cutting the blue wire
Threshold(0.0-1.0):
Object: cut the red wire after cutting the blue wire
Threshold(0.0-1.0): .1
701 [599 329 744 372]
Knife 0.20403184821230663
55 [533 616 598 646]
Screwdriver 0.24971084124137266
11 [414 207 581 270]
Pliers 0.15663059260313067
47 [401 757 543 776]
Pencil 0.11705605720418641
Object: cut the red wire after cutting the blue wirte
Threshold(0.0-1.0):
Object: cut the red wire after cutting the blue wire
Threshold(0.0-1.0): .1
701 [599 329 744 372]
Knife 0.20403184821230663
55 [533 616 598 646]
Screwdriver 0.24971084124137266
11 [414 207 581 270]
Pliers 0.15663059260313067
47 [401 757 543 776]
Pencil 0.11705605720418641
Object: cut the red wire after cutting the blue wire
Threshold(0.0-1.0): .19
701 [599 329 744 372]
Knife 0.20403184821230663
55 [533 616 598 646]
Screwdriver 0.24971084124137266
Object:
```

Activities Terminal Wed 14:52

projects obj-detect-hd code test alet annotations

Few Shot Detection Demo - Results

sri/test2/images/4814649030_6aaf41672f_z.jpg
Query: wear safety equipment on your hands, Threshold: 0.5

edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

```
87 [576 167 796 315]
Gloves 0.1535791132279714
55 [533 616 598 646]
Screwdriver 0.3326045770471377
11 [414 207 581 270]
Pliers 0.2683629009723557
47 [401 757 543 776]
Pencil 0.27433477385080024
3 [381 634 474 731]
Tape 0.3332422755276463
Object: wear safety equipment on your hands
Threshold(0.0-1.0): .1
54 [237  51 432 317]
Gloves 0.7759409934463142
310 [ 458  862  775 1004]
Gloves 0.7446147043083466
269 [ 242  735  455 1012]
Gloves 0.7679156763039303
10 [283 451 360 600]
Safety_Glass 0.4503237311464029
381 [558 669 776 836]
Gloves 0.6212864783928279
143 [441  22 774 159]
Gloves 0.6739176543018468
87 [576 167 796 315]
Gloves 0.682052808371479
701 [599 329 744 372]
Knife 0.2997159854882809
11 [414 207 581 270]
Pliers 0.3237243207312011
3 [381 634 474 731]
Tape 0.2091055449064691
35 [416 304 540 374]
Stapler 0.15404766584209584
55 [533 616 598 646]
Screwdriver 0.27824601626952383
47 [401 757 543 776]
Pencil 0.16562411527929766
141 [624 388 766 496]
Hole_Punch 0.15099426341356678
weObject: ar safety equipment on your hands
Threshold(0.0-1.0):
54 [237  51 432 317]
Gloves 0.7759409934463142
310 [ 458  862  775 1004]
Gloves 0.7446147043083466
269 [ 242  735  455 1012]
Gloves 0.7679156763039303
381 [558 669 776 836]
Gloves 0.6212864783928279
143 [441  22 774 159]
Gloves 0.6739176543018468
87 [576 167 796 315]
Gloves 0.682052808371479
Object:
```

Activities Terminal Wed 14:47

projects obj-detect-hd code test alet annotations

Few Shot Detection Demo - Results

sri/test2/images/4814649030_6aaf41672f_z.jpg
Query: mark 5 cm from the edge, Threshold: 0.3

edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

```
47 [401 757 543 776]
Pencil 0.41182429410903454
381 [558 669 776 836]
Gloves 0.2520971798095153
143 [441  22 774 159]
Gloves 0.23576754760466204
3 [381 634 474 731]
Tape 0.23137591247224615
701 [599 329 744 372]
Knife 0.23009321942225502
87 [576 167 796 315]
Gloves 0.22634026667279955
35 [416 304 540 374]
Stapler 0.18985026598437693
55 [533 616 598 646]
Screwdriver 0.2216140437213359
11 [414 207 581 270]
Pliers 0.17830060352630075
141 [624 388 766 496]
Hole_Punch 0.12725113753980263
Object: mark 5cm from the edge
Threshold(0.0-1.0): .1
701 [599 329 744 372]
Knife 0.24982325466241775
54 [237  51 432 317]
Gloves 0.21938575920503614
310 [ 458  862  775 1004]
Gloves 0.24411249675020677
269 [ 242  735  455 1012]
Gloves 0.2446201637053178
10 [283 451 360 600]
Safety_Glass 0.14320835052157013
381 [558 669 776 836]
Gloves 0.25443218181820937
143 [441  22 774 159]
Gloves 0.18224728706587587
87 [576 167 796 315]
Gloves 0.2059770871108655
3 [381 634 474 731]
Tape 0.1742617543931883
47 [401 757 543 776]
Pencil 0.30726677040165606
35 [416 304 540 374]
Stapler 0.17671930219518348
11 [414 207 581 270]
Pliers 0.17624895872015967
55 [533 616 598 646]
Screwdriver 0.27225155112099597
141 [624 388 766 496]
Hole_Punch 0.2088119450632529
mObject: ark 5 cm from the edge
Threshold(0.0-1.0): .3
47 [401 757 543 776]
Pencil 0.30726677040165606
Object:
```

edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

```
10 [283 451 360 600]
Safety_Glass 0.535460068125702
381 [558 669 776 836]
Gloves 0.5186337885078374
143 [441  22 774 159]
Gloves 0.5177954789969967
87 [576 167 796 315]
Gloves 0.462550074076145
3 [381 634 474 731]
Tape 0.3268202401973064
11 [414 207 581 270]
Pliers 0.2985988924507223
35 [416 304 540 374]
Stapler 0.2630356197871158
55 [533 616 598 646]
Screwdriver 0.25309603075838055
47 [401 757 543 776]
Pencil 0.22093718900906895
141 [624 388 766 496]
Hole_Punch 0.14989545703497661
Object:
Threshold(0.0-1.0): measure 5 cm
Threshold not valid
Threshold(0.0-1.0):
1 [292 161 478 958]
Axe 0.5057317268884385
41 [468 266 624 762]
Axe 0.4963376638264543
43 [686 144 769 837]
Rake 0.27483616996551335
measObject: ure 5 cm
Threshold(0.0-1.0):
6 [695 144 740 945]
Ruler 0.5790007015193939
41 [468 266 624 762]
Axe 0.5447505471651709
Object: measure 5 cm
Threshold(0.0-1.0): .55
6 [695 144 740 945]
Ruler 0.5790007015193939
Object: measure 5 cm from the base of the tree
Threshold(0.0-1.0):
Object: measure 5 cm from the base of the tree
Threshold(0.0-1.0): .2
6 [695 144 740 945]
Ruler 0.385992608285757
1 [292 161 478 958]
Axe 0.26673536713830037
41 [468 266 624 762]
Axe 0.2638736822838174
Object: measure 5 cm from the base of the tree
Threshold(0.0-1.0): .3
6 [695 144 740 945]
Ruler 0.385992608285757
Object:
```

edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

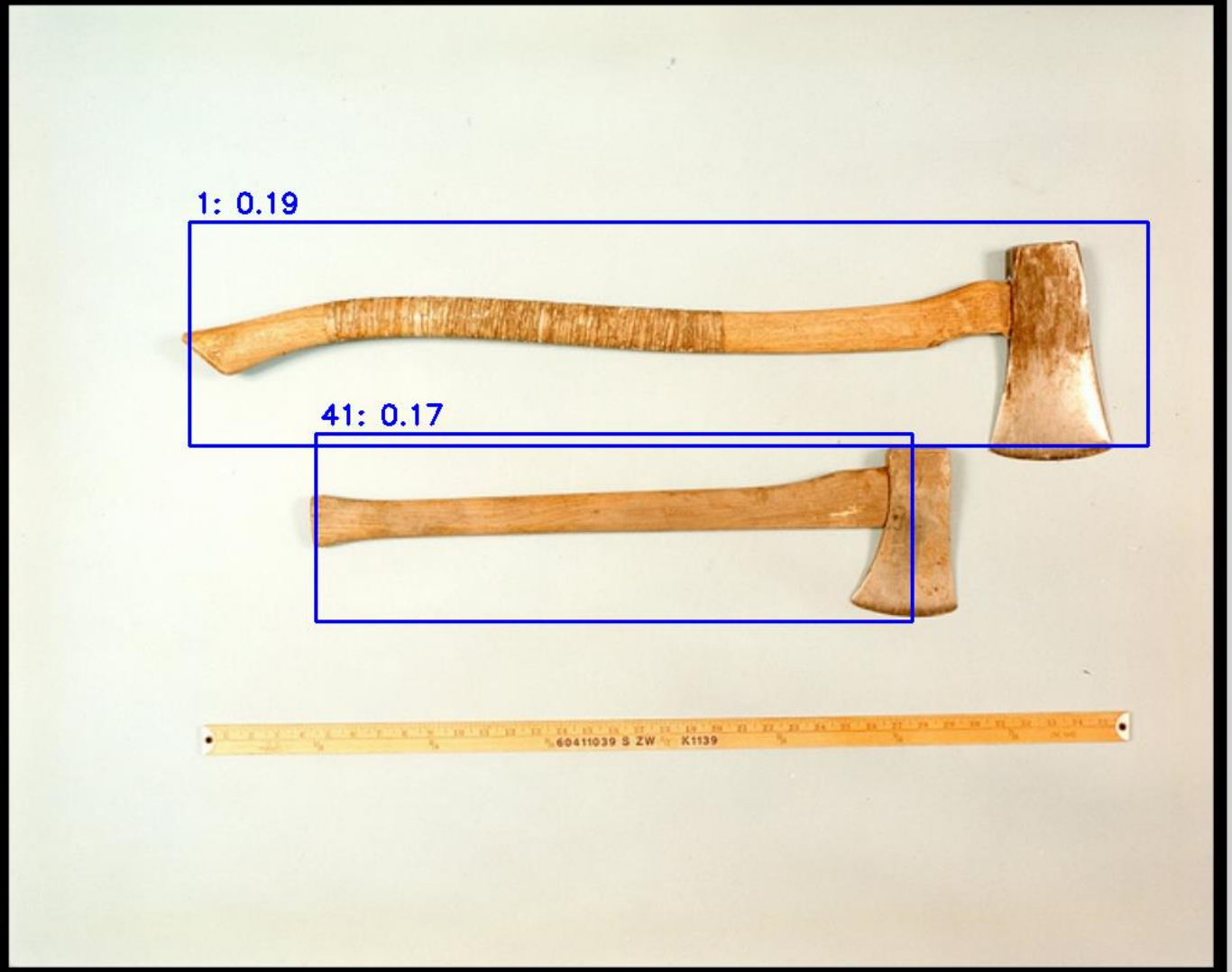
```
701 [599 329 744 372]
Knife 0.32603836196750885
55 [533 616 598 646]
Screwdriver 0.29046512348982256
Object: object
Threshold(0.0-1.0):
54 [237  51 432 317]
Gloves 0.5690896101038708
310 [ 458  862  775 1004]
Gloves 0.5630621212570499
701 [599 329 744 372]
Knife 0.5415042086887007
269 [ 242  735  455 1012]
Gloves 0.5356545480053868
10 [283 451 360 600]
Safety_Glass 0.535460068125702
381 [558 669 776 836]
Gloves 0.5186337885078374
143 [441  22 774 159]
Gloves 0.5177954789969967
87 [576 167 796 315]
Gloves 0.462550074076145
3 [381 634 474 731]
Tape 0.3268202401973064
11 [414 207 581 270]
Pliers 0.2985988924507223
35 [416 304 540 374]
Stapler 0.2630356197871158
55 [533 616 598 646]
Screwdriver 0.25309603075838055
47 [401 757 543 776]
Pencil 0.22093718900906895
141 [624 388 766 496]
Hole_Punch 0.14989545703497661
Object:
Threshold(0.0-1.0): measure 5 cm
Threshold not valid
Threshold(0.0-1.0):
1 [292 161 478 958]
Axe 0.5057317268884385
41 [468 266 624 762]
Axe 0.4963376638264543
43 [686 144 769 837]
Rake 0.27483616996551335
measObject: ure 5 cm
Threshold(0.0-1.0):
6 [695 144 740 945]
Ruler 0.5790007015193939
41 [468 266 624 762]
Axe 0.5447505471651709
Object: measure 5 cm
Threshold(0.0-1.0): .55
6 [695 144 740 945]
Ruler 0.5790007015193939
Object:
```

Activities Terminal Wed 14:55

projects obj-detect-hd code test alet annotations

Few Shot Detection Demo - Results

sri/test2/images/5712533437_29f6355e3a_z.jpg
Query: make a mark 5 cm from the base of the tree, Threshold: 0.15



edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

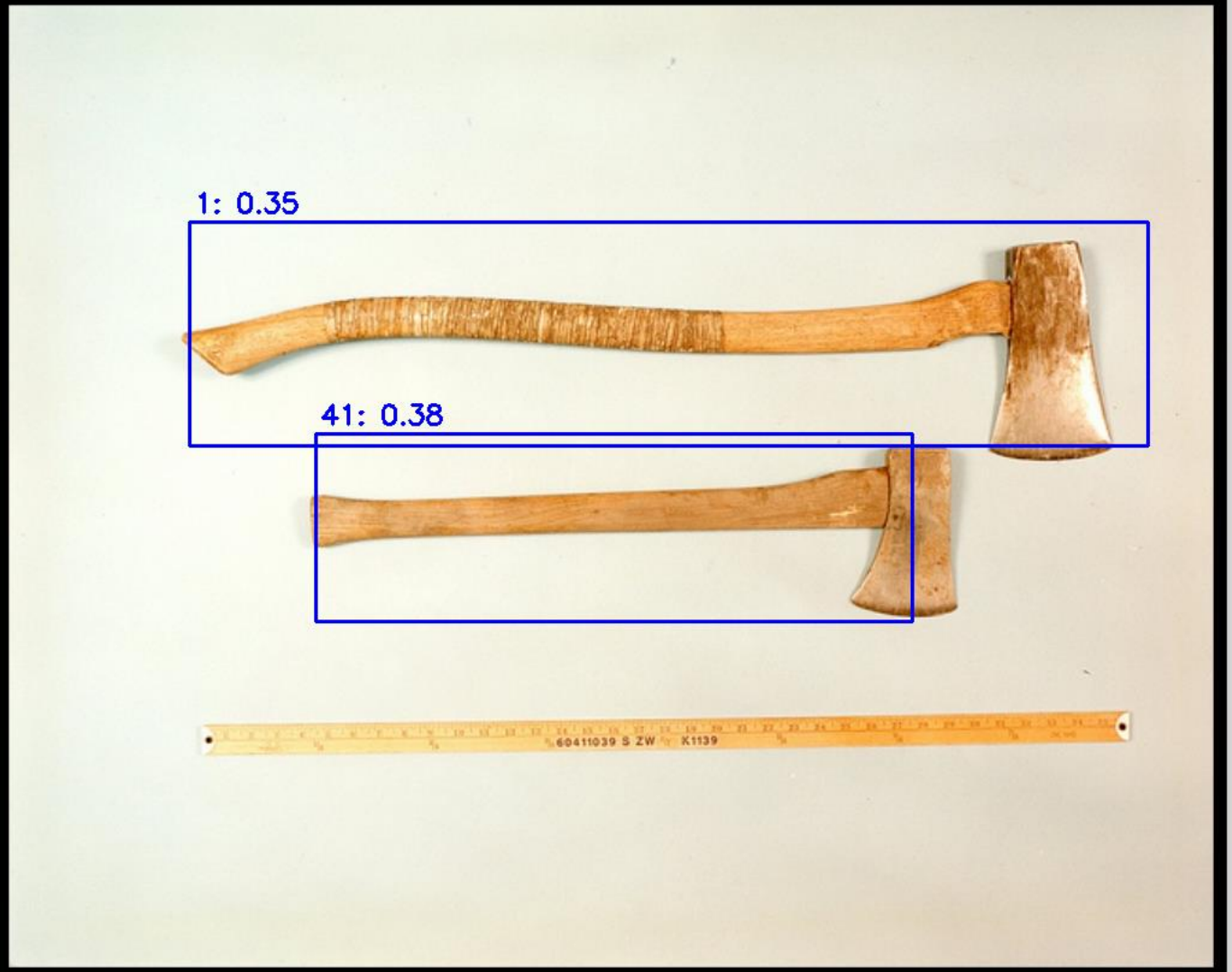
```
55 [533 616 598 646]
Screwdriver 0.25309603075838055
47 [401 757 543 776]
Pencil 0.22093718900906895
141 [624 388 766 496]
Hole_Punch 0.14989545703497661
Object:
Threshold(0.0-1.0): measure 5 cm
Threshold not valid
Threshold(0.0-1.0):
1 [292 161 478 958]
Axe 0.5057317268884385
41 [468 266 624 762]
Axe 0.4963376638264543
43 [686 144 769 837]
Rake 0.27483616996551335
measObject: ure 5 cm
Threshold(0.0-1.0):
6 [695 144 740 945]
Ruler 0.5790007015193939
41 [468 266 624 762]
Axe 0.5447505471651709
Object: measure 5 cm
Threshold(0.0-1.0): .55
6 [695 144 740 945]
Ruler 0.5790007015193939
Object: measure 5 cm from the base of the tree
Threshold(0.0-1.0):
Object: measure 5 cm from the base of the tree
Threshold(0.0-1.0): .2
6 [695 144 740 945]
Ruler 0.385992608285757
1 [292 161 478 958]
Axe 0.26673536713830037
41 [468 266 624 762]
Axe 0.2638736822838174
Object: measure 5 cm from the base of the tree
Threshold(0.0-1.0): .3
6 [695 144 740 945]
Ruler 0.385992608285757
Object: make a mark 5 cm from the base of the tree
Threshold(0.0-1.0): .1
1 [292 161 478 958]
Axe 0.1851405187214813
41 [468 266 624 762]
Axe 0.16555505890676084
6 [695 144 740 945]
Ruler 0.1412002690341673
Object: make a mark 5 cm from the base of the tree
Threshold(0.0-1.0): .15
1 [292 161 478 958]
Axe 0.1851405187214813
41 [468 266 624 762]
Axe 0.16555505890676084
Object:
```

Activities Terminal Wed 14:56

projects obj-detect-hd code test alet annotations

Few Shot Detection Demo - Results

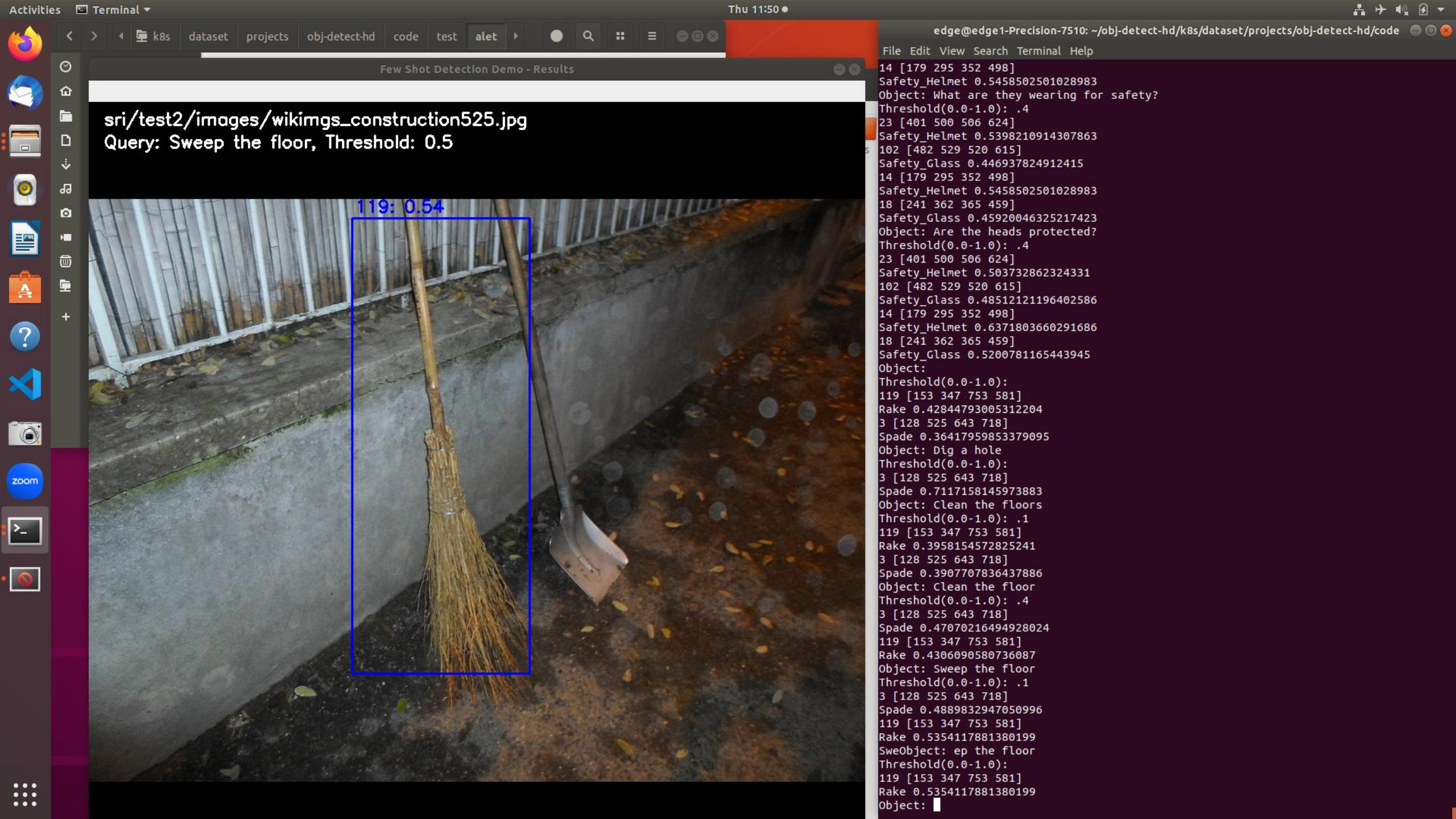
sri/test2/images/5712533437_29f6355e3a_z.jpg
Query: chop the block into pieces not longer than 50 cm, Threshold: 0.3

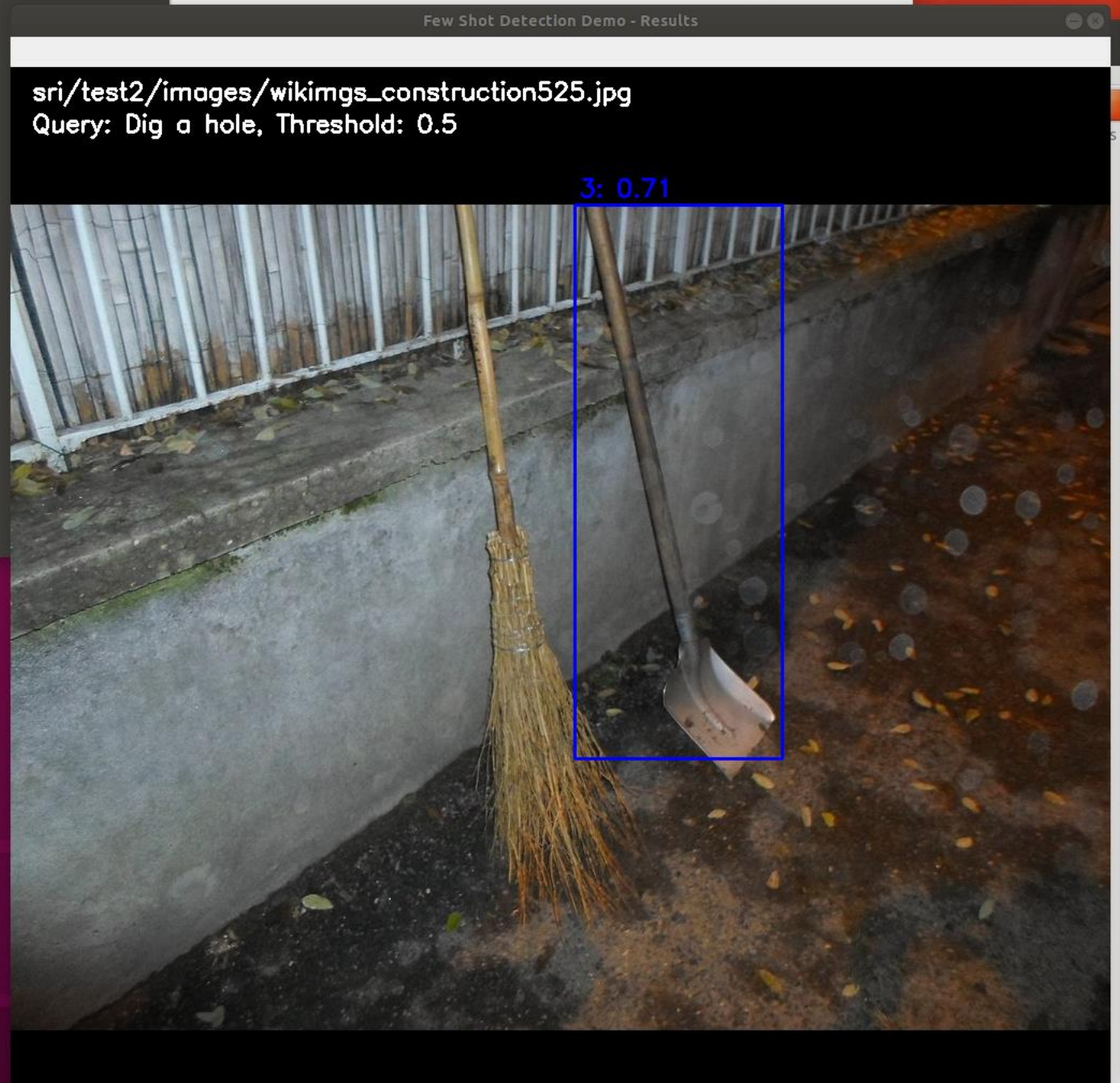


edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

```
43 [686 144 769 837]
Rake 0.27483616996551335
measObject: ure 5 cm
Threshold(0.0-1.0):
6 [695 144 740 945]
Ruler 0.5790007015193939
41 [468 266 624 762]
Axe 0.5447505471651709
Object: measure 5 cm
Threshold(0.0-1.0): .55
6 [695 144 740 945]
Ruler 0.5790007015193939
Object: measure 5 cm from the base of the tree
Threshold(0.0-1.0):
Object: measure 5 cm from the base of the tree
Threshold(0.0-1.0): .2
6 [695 144 740 945]
Ruler 0.385992608285757
1 [292 161 478 958]
Axe 0.26673536713830037
41 [468 266 624 762]
Axe 0.2638736822838174
Object: measure 5 cm from the base of the tree
Threshold(0.0-1.0): .3
6 [695 144 740 945]
Ruler 0.385992608285757
Object: make a mark 5 cm from the base of the tree
Threshold(0.0-1.0): .1
1 [292 161 478 958]
Axe 0.1851405187214813
41 [468 266 624 762]
Axe 0.16555505890676084
6 [695 144 740 945]
Ruler 0.1412002690341673
Object: make a mark 5 cm from the base of the tree
Threshold(0.0-1.0): .15
1 [292 161 478 958]
Axe 0.1851405187214813
41 [468 266 624 762]
Axe 0.16555505890676084
Object: chop the block into pieces not longer than 50 cm
Threshold(0.0-1.0): .1
1 [292 161 478 958]
Axe 0.3471356850857492
41 [468 266 624 762]
Axe 0.38208640624523393
43 [686 144 769 837]
Rake 0.2886066302653635
Object: chop the block into pieces not longer than 50 cm
Threshold(0.0-1.0): .3
1 [292 161 478 958]
Axe 0.3471356850857492
41 [468 266 624 762]
Axe 0.38208640624523393
Object:
```


Activities
Terminal
Thu 11:50
k8s
dataset
projects
obj-detect-hd
code
test
alet
Few Shot Detection Demo - Results
sri/test2/images/wikimgs_construction525.jpg
Query: Sweep the floor, Threshold: 0.5
119: 0.54
edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code
File Edit View Search Terminal Help
14 [179 295 352 498]
Safety_Helmet 0.5458502501028983
Object: What are they wearing for safety?
Threshold(0.0-1.0): .4
23 [401 500 506 624]
Safety_Helmet 0.5398210914307863
102 [482 529 520 615]
Safety_Glass 0.446937824912415
14 [179 295 352 498]
Safety_Helmet 0.5458502501028983
18 [241 362 365 459]
Safety_Glass 0.45920046325217423
Object: Are the heads protected?
Threshold(0.0-1.0): .4
23 [401 500 506 624]
Safety_Helmet 0.503732862324331
102 [482 529 520 615]
Safety_Glass 0.48512121196402586
14 [179 295 352 498]
Safety_Helmet 0.6371803660291686
18 [241 362 365 459]
Safety_Glass 0.5200781165443945
Object:
Threshold(0.0-1.0):
119 [153 347 753 581]
Rake 0.42844793005312204
3 [128 525 643 718]
Spade 0.36417959853379095
Object: Dig a hole
Threshold(0.0-1.0):
3 [128 525 643 718]
Spade 0.7117158145973883
Object: Clean the floors
Threshold(0.0-1.0): .1
119 [153 347 753 581]
Rake 0.3958154572825241
3 [128 525 643 718]
Spade 0.3907707836437886
Object: Clean the floor
Threshold(0.0-1.0): .4
3 [128 525 643 718]
Spade 0.47070216494928024
119 [153 347 753 581]
Rake 0.4306090580736087
Object: Sweep the floor
Threshold(0.0-1.0): .1
3 [128 525 643 718]
Spade 0.4889832947050996
119 [153 347 753 581]
Rake 0.5354117881380199
SweObject: ep the floor
Threshold(0.0-1.0):
119 [153 347 753 581]
Rake 0.5354117881380199
Object:



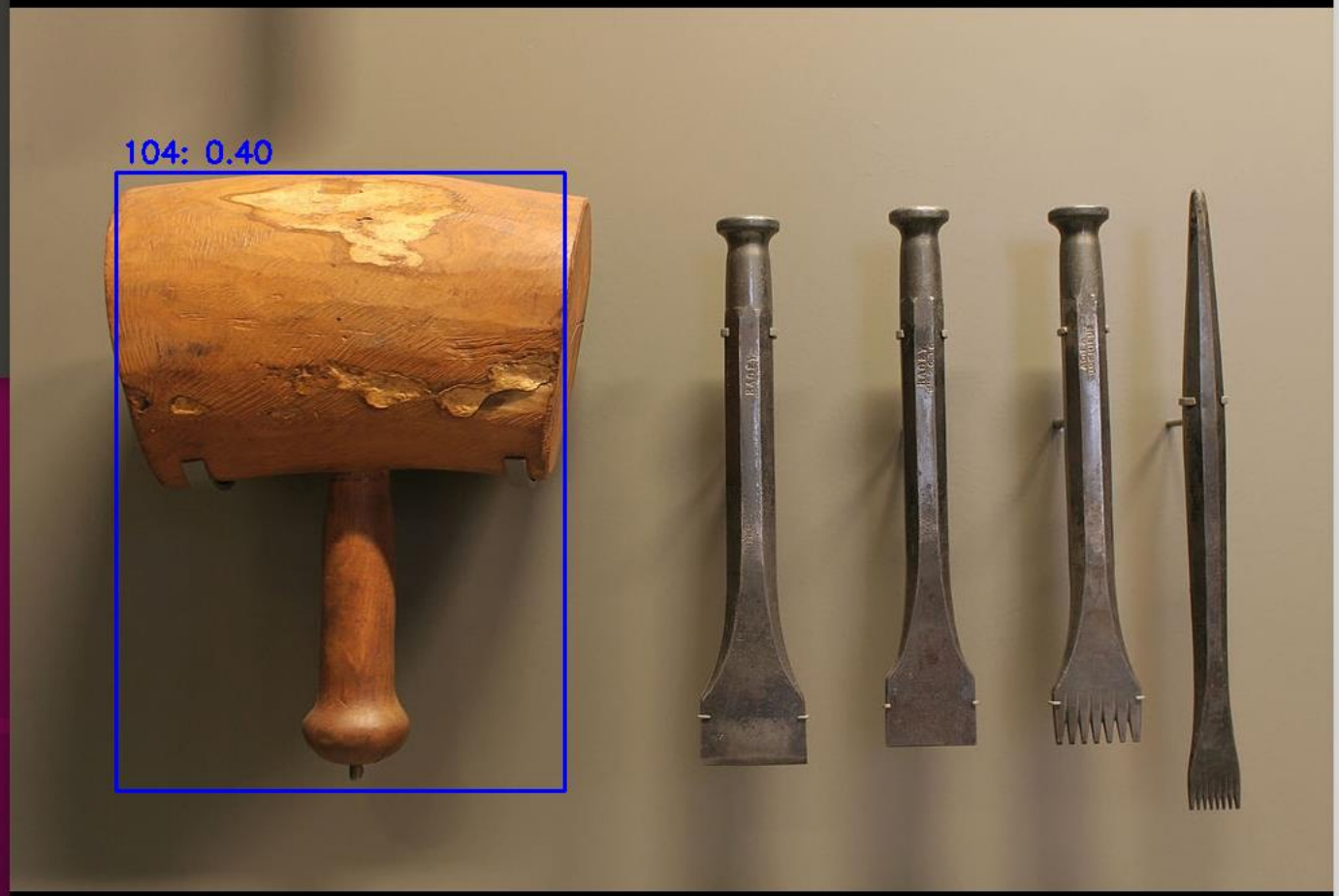

```
edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code
File  Edit  View  Search  Terminal  Help
23 [401 500 506 624]
Safety_Helmet 0.5687023571275274
102 [482 529 520 615]
Safety_Glass 0.5678500948667304
14 [179 295 352 498]
Safety_Helmet 0.5673928719954929
18 [241 362 365 459]
Safety_Glass 0.5499185978185245
64 [713 725 777 844]
Hammer 0.4961679481836516
16 [712 406 813 535]
Meter 0.43988317502158386
17 [583 382 821 677]
Drill 0.40667565457785665
129 [431 416 641 668]
Drill 0.30684897475920875
67 [579 356 660 714]
Drill 0.30444807077721714
Object: What are they wearing for safety?
Threshold(0.0-1.0):
23 [401 500 506 624]
Safety_Helmet 0.5398210914307863
14 [179 295 352 498]
Safety_Helmet 0.5458502501028983
Object: What are they wearing for safety?
Threshold(0.0-1.0): .4
23 [401 500 506 624]
Safety_Helmet 0.5398210914307863
102 [482 529 520 615]
Safety_Glass 0.446937824912415
14 [179 295 352 498]
Safety_Helmet 0.5458502501028983
18 [241 362 365 459]
Safety_Glass 0.45920046325217423
Object: Are the heads protected?
Threshold(0.0-1.0): .4
23 [401 500 506 624]
Safety_Helmet 0.503732862324331
102 [482 529 520 615]
Safety_Glass 0.48512121196402586
14 [179 295 352 498]
Safety_Helmet 0.6371803660291686
18 [241 362 365 459]
Safety_Glass 0.5200781165443945
Object:
Threshold(0.0-1.0):
119 [153 347 753 581]
Rake 0.42844793005312204
3 [128 525 643 718]
Spade 0.36417959853379095
Object: Dig a hole
Threshold(0.0-1.0):
3 [128 525 643 718]
Spade 0.7117158145973883
Object:
```

Activities Terminal Thu 11:40

k8s dataset projects obj-detect-hd code test alet

Few Shot Detection Demo - Results

sri/test2/images/wikimgs_construction157.jpg
Query: Drive stakes into the ground at 3 ft intervals, Threshold: 0.39



edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

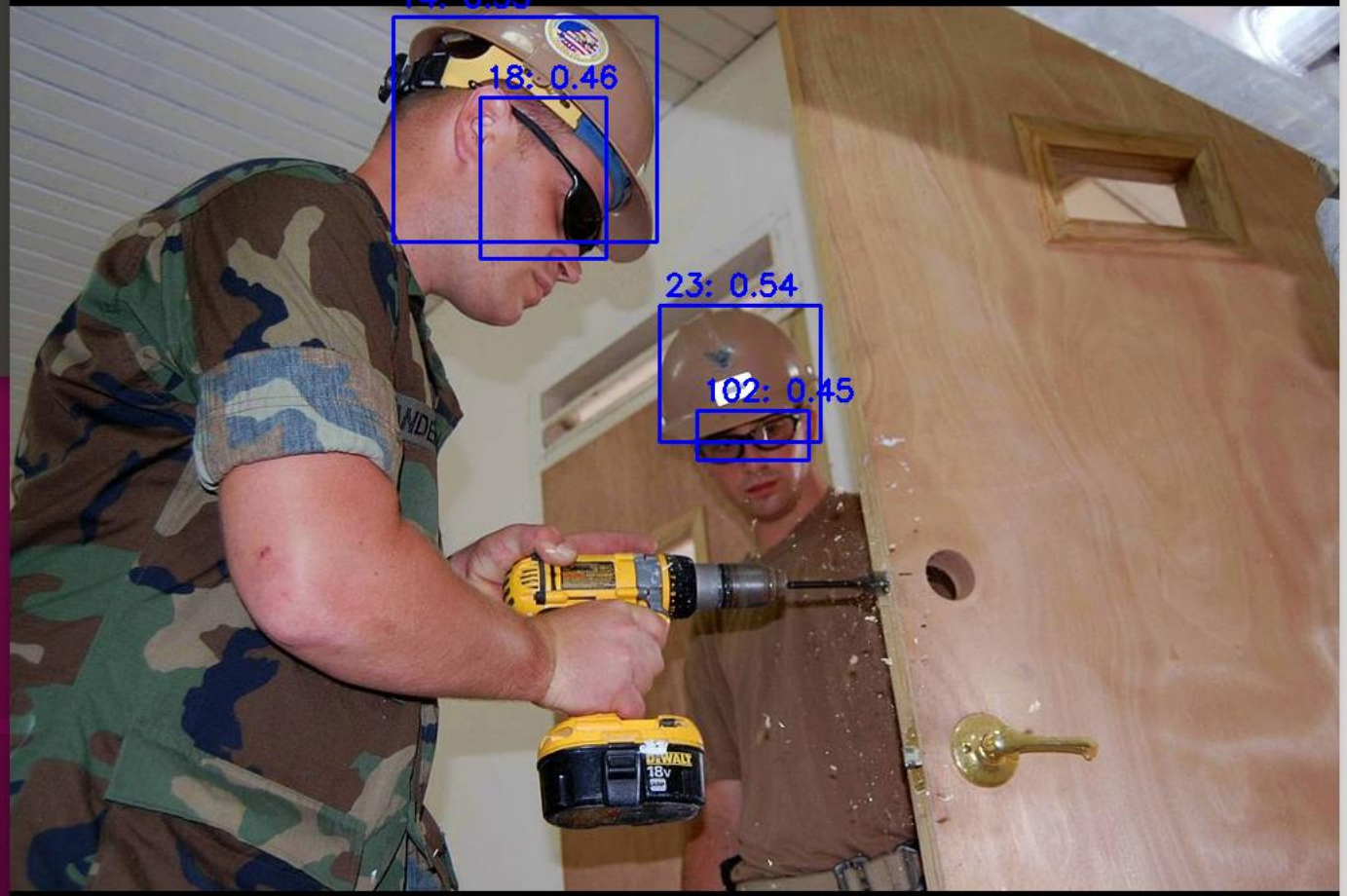
```
27 [404 517 853 555]
Pencil 0.3659864773186031
38 [326 672 822 714]
Pencil 0.33838498087506763
23 [451 799 838 839]
Pencil 0.328865397626121
3 [452 486 853 520]
Pencil 0.3139769733511548
71 [396 739 828 777]
Pencil 0.3353031683695754
56 [370 548 853 585]
Pencil 0.3480936150399423
25 [510 263 853 297]
Pencil 0.3119003797764657
2 [464 456 853 490]
Pencil 0.3014242958789314
9 [563 327 853 360]
Pencil 0.30161830164335435
48 [488 425 843 458]
Pencil 0.33887232779871457
206 [552 362 847 394]
Pencil 0.30892598502842616
Object:
Threshold(0.0-1.0):
104 [297  82 775 429]
Mallet 0.4164882492604964
3 [323 526 745 624]
Chisel 0.30902320164137753
20 [308 804 748 882]
Chisel 0.3062638687859102
2 [305 674 732 753]
Chisel 0.29377860591810817
0 [301 901 813 952]
Chisel 0.2511685176875034
Object: Drive stakes into the ground at 3 ft intervals
Threshold(0.0-1.0):
Object: Drive stakes into the ground at 3 ft intervals
Threshold(0.0-1.0): .1
104 [297  82 775 429]
Mallet 0.3970841351880859
3 [323 526 745 624]
Chisel 0.22750374537918766
20 [308 804 748 882]
Chisel 0.2472235534123024
72 [307 902 813 957]
File 0.2526188262245838
2 [305 674 732 753]
Chisel 0.2635070504846152
DObject: rive stakes into the ground at 3 ft intervals
Threshold(0.0-1.0): .4
Object: Drive stakes into the ground at 3 ft intervals
Threshold(0.0-1.0): .39
104 [297  82 775 429]
Mallet 0.3970841351880859
Object:
```

Activities Terminal Thu 11:47

k8s dataset projects obj-detect-hd code test alet

Few Shot Detection Demo - Results

sri/test2/images/wikimgs_construction488.jpg
Query: What are they wearing for safety?, Threshold: 0.4



edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

```
Object: Make sure that the board is laying flat
Threshold(0.0-1.0): .3
413 [313 519 467 663]
Safety_Helmet 0.4888076208099814
4 [337 209 385 269]
Safety_Helmet 0.3718816323796321
1 [603 413 646 518]
Gloves 0.5116619536070042
28 [660 566 747 627]
Gloves 0.3981751763364499
14 [506 221 537 254]
Gloves 0.34430088027754313
596 [ 777  677  873 1009]
Square 0.34770275001588513
16 [520 198 543 223]
Gloves 0.3133426997356985
180 [686   0 727 570]
Level 0.3583101774078799
Object:
Threshold(0.0-1.0):
23 [401 500 506 624]
Safety_Helmet 0.5687023571275274
102 [482 529 520 615]
Safety_Glass 0.5678500948667304
14 [179 295 352 498]
Safety_Helmet 0.5673928719954929
18 [241 362 365 459]
Safety_Glass 0.5499185978185245
64 [713 725 777 844]
Hammer 0.4961679481836516
16 [712 406 813 535]
Meter 0.43988317502158386
17 [583 382 821 677]
Drill 0.40667565457785665
129 [431 416 641 668]
Drill 0.30684897475920875
67 [579 356 660 714]
Drill 0.30444807077721714
Object: What are they wearing for safety?
Threshold(0.0-1.0):
23 [401 500 506 624]
Safety_Helmet 0.5398210914307863
14 [179 295 352 498]
Safety_Helmet 0.5458502501028983
Object: What are they wearing for safety?
Threshold(0.0-1.0): .4
23 [401 500 506 624]
Safety_Helmet 0.5398210914307863
102 [482 529 520 615]
Safety_Glass 0.446937824912415
14 [179 295 352 498]
Safety_Helmet 0.5458502501028983
18 [241 362 365 459]
Safety_Glass 0.45920046325217423
Object:
```

Activities Terminal Thu 11:55

k8s dataset projects obj-detect-hd code test alet

Few Shot Detection Demo - Results

edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

```
431 [274 555 312 608]
Safety_Helmet 0.5658154388503659
8 [291 361 321 407]
Safety_Helmet 0.5630395911537167
243 [280 137 319 204]
Safety_Helmet 0.5620502106876839
387 [248 744 293 810]
Safety_Helmet 0.5583803653622563
40 [249 423 294 480]
Safety_Helmet 0.5575987286478513
753 [288 752 306 803]
Safety_Glass 0.5505302499917436
51 [229 827 303 980]
Safety_Helmet 0.5455950243877883
189 [281 613 321 655]
Safety_Helmet 0.5406755925864021
499 [536 702 587 726]
Hammer 0.5338433077162869
988 [263 237 287 286]
Safety_Glass 0.5107664070635285
16 [305 558 324 607]
Safety_Glass 0.48965569733436903
180 [731 586 762 622]
Safety_Helmet 0.47668827597874797
23 [294 424 313 478]
Safety_Glass 0.4002675862503443
929 [712 555 732 575]
Safety_Helmet 0.3600288859963268
6 [319 145 331 199]
Safety_Glass 0.3150836543075308
488 [316 558 332 602]
Safety_Glass 0.3130712429354191
57 [316  89 333 139]
Safety_Glass 0.30171695073904614
Object: Protection from accidents
Threshold(0.0-1.0):
2 [275  69 324 142]
Safety_Helmet 0.6643299333790389
106 [225 230 272 293]
Safety_Helmet 0.6475523488511641
0 [288 486 316 524]
Safety_Helmet 0.607105468873278
431 [274 555 312 608]
Safety_Helmet 0.5910870306257916
8 [291 361 321 407]
Safety_Helmet 0.548871155536875
243 [280 137 319 204]
Safety_Helmet 0.6936689860280905
387 [248 744 293 810]
Safety_Helmet 0.5633518565052299
40 [249 423 294 480]
Safety_Helmet 0.6819238514622894
51 [229 827 303 980]
Safety_Helmet 0.5406632367894842
Object:
```

Activities
Terminal
Thu 11:57
k8s
dataset
projects
obj-detect-hd
code
test
alet
Few Shot Detection Demo - Results
sri/test2/images/wikimgs_construction617.jpg
Query: Identify safety equipment, Threshold: 0.5
55: 0.86
161: 0.72
9: 0.84
4: 0.84
1: 0.82
156: 0.62
8: 0.82
2: 0.71
edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code
File Edit View Search Terminal Help
862 [310 564 329 581]
Safety_Helmet 0.46031971182831366
123 [388 79 411 124]
Safety_Glass 0.444742030376696
591 [317 940 371 995]
Safety_Helmet 0.4368169596724729
28 [319 646 330 677]
Safety_Glass 0.35906746598782424
897 [621 352 645 380]
Gloves 0.32447111215992336
27 [325 509 340 528]
Safety_Helmet 0.32031716361497053
19 [516 745 541 851]
Trowel 0.2946461940530264
Object: Protect the brain
Threshold(0.0-1.0):
8 [352 44 398 139]
Safety_Helmet 0.5615290926266319
55 [248 161 302 237]
Safety_Helmet 0.5253784748765815
665 [326 377 366 419]
Safety_Helmet 0.5033988064407057
Object: Identify safety equipment
Threshold(0.0-1.0):
4 [288 525 317 571]
Safety_Helmet 0.8354744735199131
1 [293 634 324 680]
Safety_Helmet 0.8196874003709118
8 [352 44 398 139]
Safety_Helmet 0.8209221420309508
55 [248 161 302 237]
Safety_Helmet 0.8558983699751002
665 [326 377 366 419]
Safety_Helmet 0.8359862824716415
9 [304 420 338 476]
Safety_Helmet 0.8353740925115367
161 [271 261 305 312]
Safety_Helmet 0.7205609557266375
276 [296 278 336 328]
Safety_Helmet 0.7847845709954322
156 [279 854 325 928]
Safety_Helmet 0.6179998529957917
3 [330 465 357 498]
Safety_Helmet 0.8368891755086446
17 [293 175 309 230]
Safety_Glass 0.6025441922066508
371 [312 593 334 614]
Safety_Helmet 0.6103124261920262
2 [703 335 826 438]
Safety_Helmet 0.7083459934485716
862 [310 564 329 581]
Safety_Helmet 0.5850979475027362
27 [325 509 340 528]
Safety_Helmet 0.5136806634238349
Object:

Activities Terminal Thu 12:00

k8s dataset projects obj-detect-hd code test alet

Few Shot Detection Demo - Results

edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

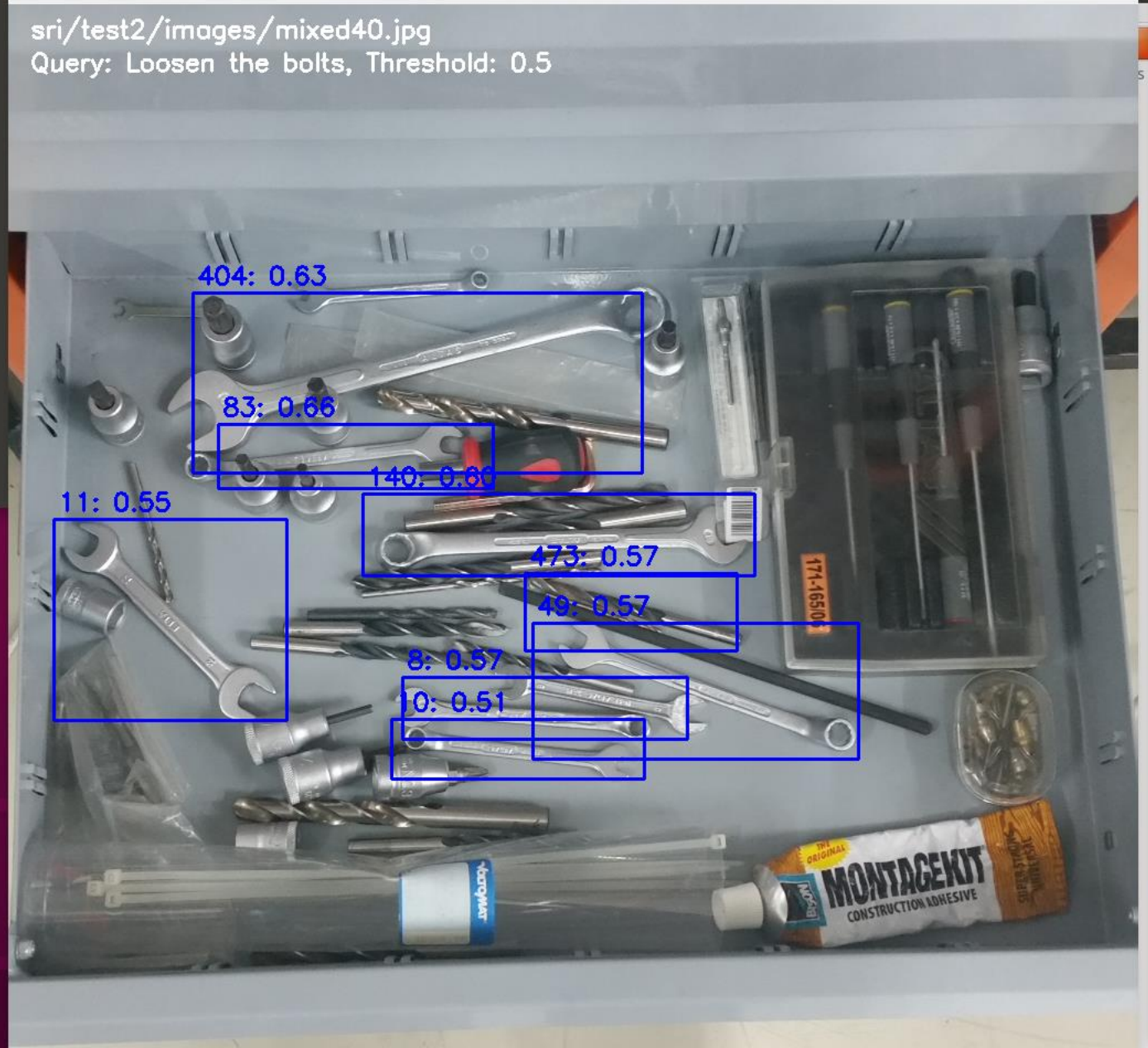
```
24 [524 558 551 583]
Safety_Helmet 0.4968337492089221
66 [498 114 514 129]
Gloves 0.4840930461480117
329 [557 721 575 740]
Safety_Helmet 0.40168938602555687
Object: Check if the people are safe
Threshold(0.0-1.0):
62 [298 308 320 339]
Safety_Helmet 0.560561034990062
8 [343 512 369 542]
Safety_Helmet 0.5474197081278546
12 [361 686 385 714]
Safety_Helmet 0.5404478640643258
26 [253 150 275 175]
Safety_Helmet 0.5395333066359409
17 [319 443 344 471]
Safety_Helmet 0.5455091333443045
6 [460 324 489 352]
Safety_Helmet 0.5506608947463554
24 [524 558 551 583]
Safety_Helmet 0.5746848451405635
66 [498 114 514 129]
Gloves 0.5122791629335849
329 [557 721 575 740]
Safety_Helmet 0.5107607598000494
Object: safety equipment
Threshold(0.0-1.0):
62 [298 308 320 339]
Safety_Helmet 0.5062200753407635
8 [343 512 369 542]
Safety_Helmet 0.547772818346743
12 [361 686 385 714]
Safety_Helmet 0.5886837538726801
Object: Identify safety equipment
Threshold(0.0-1.0):
62 [298 308 320 339]
Safety_Helmet 0.6095845094930152
8 [343 512 369 542]
Safety_Helmet 0.7033028370989707
12 [361 686 385 714]
Safety_Helmet 0.7333886457518379
26 [253 150 275 175]
Safety_Helmet 0.5698392138862766
37 [433 945 460 977]
Safety_Helmet 0.6072965373697803
17 [319 443 344 471]
Safety_Helmet 0.66291894285765
59 [408 108 435 130]
Safety_Helmet 0.5828538106371197
6 [460 324 489 352]
Safety_Helmet 0.5822574627736047
24 [524 558 551 583]
Safety_Helmet 0.600257744614302
Object:
```

Activities
Terminal
Thu 11:48
k8s
dataset
projects
obj-detect-hd
code
test
alet
Few Shot Detection Demo - Results
sri/test2/images/wikimgs_construction488.jpg
Query: Are the heads protected? , Threshold: 0.4
14: 0.64
18: 0.52
23: 0.50
102: 0.49
edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code
File Edit View Search Terminal Help
14 [506 221 537 254]
Gloves 0.34430088027754313
596 [777 677 873 1009]
Square 0.34770275001588513
16 [520 198 543 223]
Gloves 0.3133426997356985
180 [686   0 727 570]
Level 0.3583101774078799
Object:
Threshold(0.0-1.0):
23 [401 500 506 624]
Safety_Helmet 0.5687023571275274
102 [482 529 520 615]
Safety_Glass 0.5678500948667304
14 [179 295 352 498]
Safety_Helmet 0.5673928719954929
18 [241 362 365 459]
Safety_Glass 0.5499185978185245
64 [713 725 777 844]
Hammer 0.4961679481836516
16 [712 406 813 535]
Meter 0.43988317502158386
17 [583 382 821 677]
Drill 0.40667565457785665
129 [431 416 641 668]
Drill 0.30684897475920875
67 [579 356 660 714]
Drill 0.30444807077721714
Object: What are they wearing for safety?
Threshold(0.0-1.0):
23 [401 500 506 624]
Safety_Helmet 0.5398210914307863
14 [179 295 352 498]
Safety_Helmet 0.5458502501028983
Object: What are they wearing for safety?
Threshold(0.0-1.0): .4
23 [401 500 506 624]
Safety_Helmet 0.5398210914307863
102 [482 529 520 615]
Safety_Glass 0.446937824912415
14 [179 295 352 498]
Safety_Helmet 0.5458502501028983
18 [241 362 365 459]
Safety_Glass 0.45920046325217423
Object: Are the heads protected?
Threshold(0.0-1.0): .4
23 [401 500 506 624]
Safety_Helmet 0.503732862324331
102 [482 529 520 615]
Safety_Glass 0.48512121196402586
14 [179 295 352 498]
Safety_Helmet 0.6371803660291686
18 [241 362 365 459]
Safety_Glass 0.5200781165443945
Object:

Activities Terminal Thu 11:21

k8s dataset projects obj-detect-hd code test alet

Few Shot Detection Demo - Results



edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

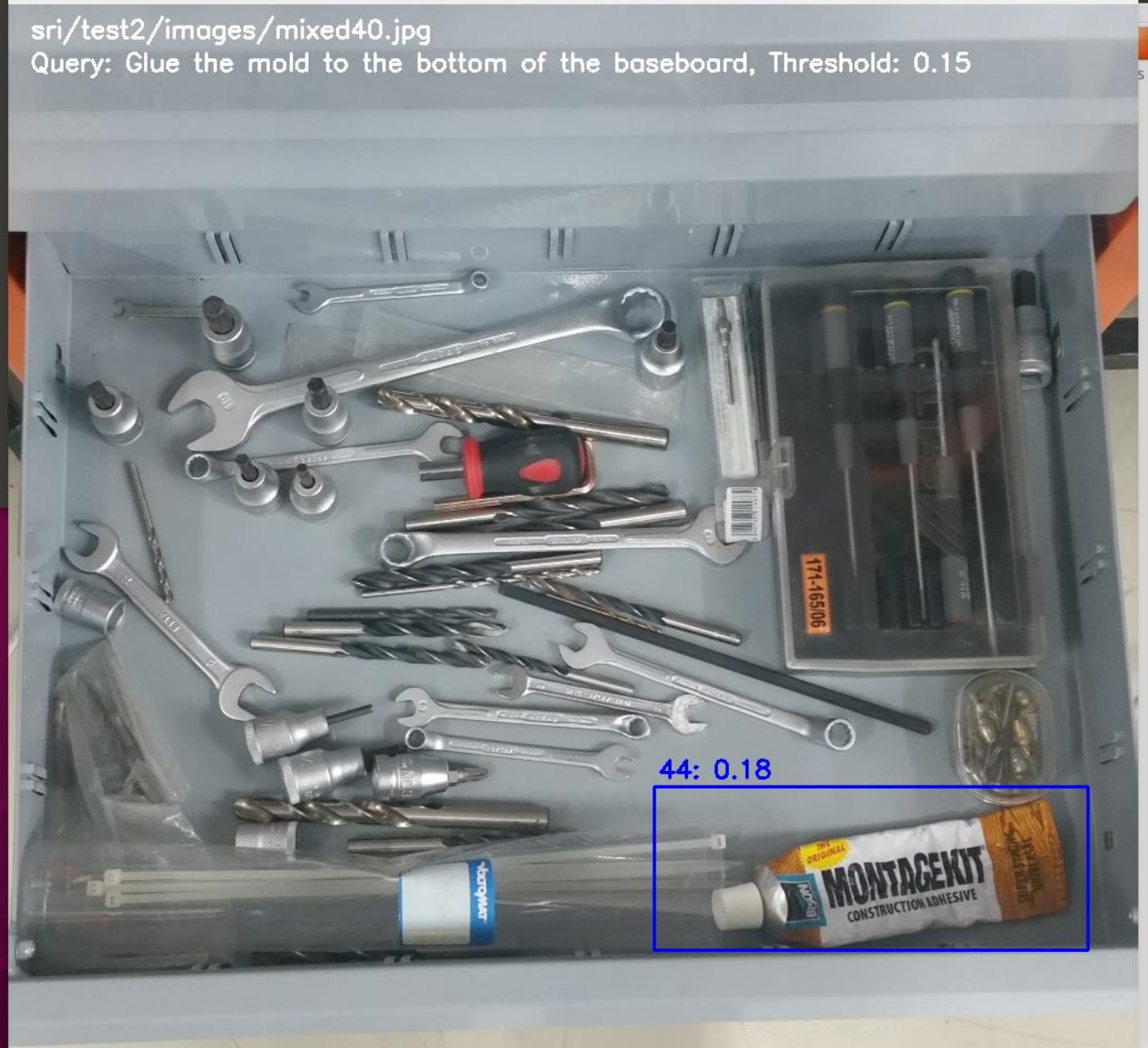
File Edit View Search Terminal Help

```
491 [273 728 514 776]
Screwdriver 0.35410484598485525
823 [246 777 562 850]
Screwdriver 0.3471045278412186
331 [554 242 591 464]
Wrench 0.30216241578779907
44 [708 585 856 978]
Brush 0.26942448888101994
346 [375 345 454 544]
Screwdriver 0.22580779772436915
35 [741  68 836 799]
Screwdriver 0.19565603937230103
Object: Loosen the bolts
Threshold(0.0-1.0):
8 [609 357 665 615]
Wrench 0.5749793451203665
11 [466  41 648 252]
Wrench 0.5502136757921448
83 [380 190 438 439]
Wrench 0.6579746819551111
10 [647 347 701 576]
Wrench 0.5096007107746695
404 [261 167 424 574]
Wrench 0.6349655379912404
473 [515 468 585 660]
Wrench 0.5738522383974936
140 [443 321 517 676]
Wrench 0.5992715553152466
49 [560 475 683 770]
Wrench 0.5686820330334615
Object: Loosen the bolts
Threshold(0.0-1.0): .6
83 [380 190 438 439]
Wrench 0.6579746819551111
404 [261 167 424 574]
Wrench 0.6349655379912404
Object: Loosen the bolts
Threshold(0.0-1.0):
8 [609 357 665 615]
Wrench 0.5749793451203665
11 [466  41 648 252]
Wrench 0.5502136757921448
83 [380 190 438 439]
Wrench 0.6579746819551111
10 [647 347 701 576]
Wrench 0.5096007107746695
404 [261 167 424 574]
Wrench 0.6349655379912404
473 [515 468 585 660]
Wrench 0.5738522383974936
140 [443 321 517 676]
Wrench 0.5992715553152466
49 [560 475 683 770]
Wrench 0.5686820330334615
Object:
```

Activities Terminal Thu 11:22

k8s dataset projects obj-detect-hd code test alet

Few Shot Detection Demo - Results



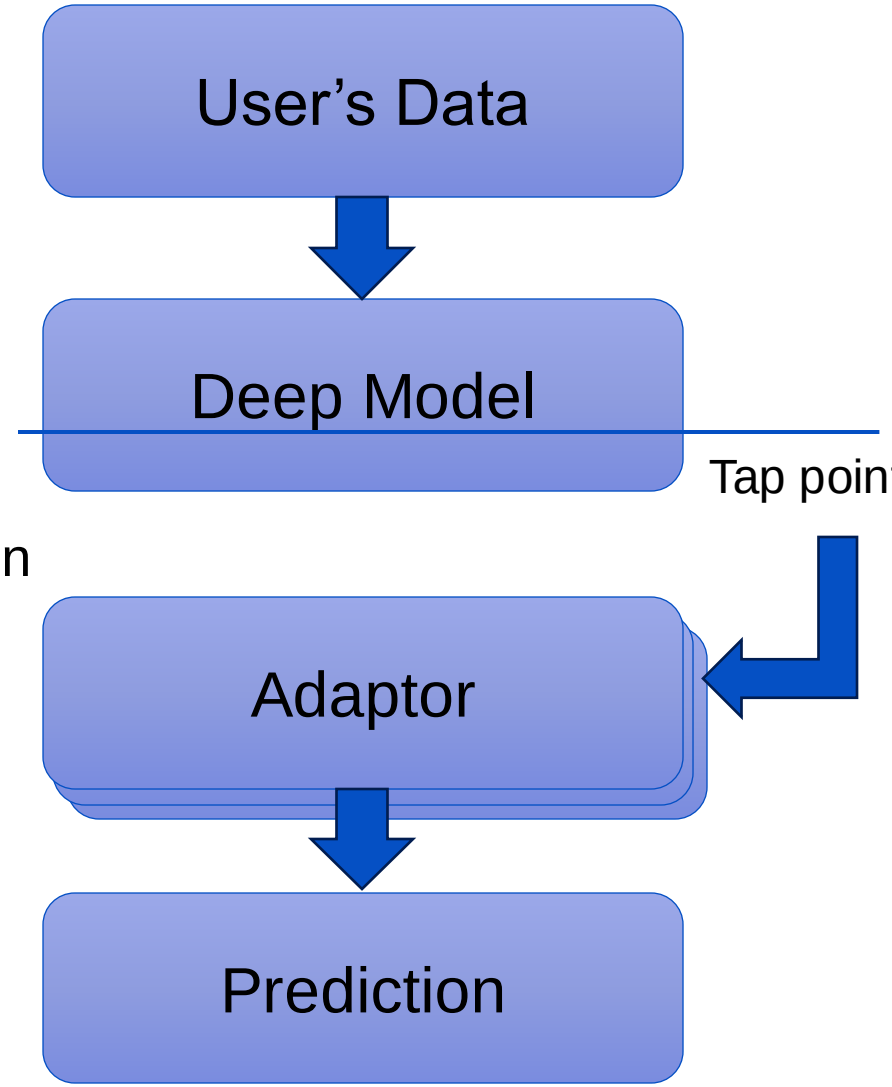
edge@edge1-Precision-7510: ~/obj-detect-hd/k8s/dataset/projects/obj-detect-hd/code

File Edit View Search Terminal Help

```
8 [609 357 665 615]
Wrench 0.5749793451203665
11 [466  41 648 252]
Wrench 0.5502136757921448
83 [380 190 438 439]
Wrench 0.6579746819551111
10 [647 347 701 576]
Wrench 0.5096007107746695
404 [261 167 424 574]
Wrench 0.6349655379912404
473 [515 468 585 660]
Wrench 0.5738522383974936
140 [443 321 517 676]
Wrench 0.5992715553152466
49 [560 475 683 770]
Wrench 0.5686820330334615
Object: Loosen the bolts
Threshold(0.0-1.0): .6
83 [380 190 438 439]
Wrench 0.6579746819551111
404 [261 167 424 574]
Wrench 0.6349655379912404
Object: Loosen the bolts
Threshold(0.0-1.0):
8 [609 357 665 615]
Wrench 0.5749793451203665
11 [466  41 648 252]
Wrench 0.5502136757921448
83 [380 190 438 439]
Wrench 0.6579746819551111
10 [647 347 701 576]
Wrench 0.5096007107746695
404 [261 167 424 574]
Wrench 0.6349655379912404
473 [515 468 585 660]
Wrench 0.5738522383974936
140 [443 321 517 676]
Wrench 0.5992715553152466
49 [560 475 683 770]
Wrench 0.5686820330334615
Object: Glue the mold to the bottom of the baseboard
Threshold(0.0-1.0): .1
473 [515 468 585 660]
Wrench 0.12027464449401798
801 [244 848 573 913]
Screwdriver 0.10207170068272235
491 [273 728 514 776]
Screwdriver 0.10977602304480187
44 [708 585 856 978]
Brush 0.1764872878452598
Object: Glue the mold to the bottom of the baseboard
Threshold(0.0-1.0): .15
44 [708 585 856 978]
Brush 0.1764872878452598
Object:
```

Tutorial Outline

1. Notebooks on Hyperdimensional Computing (HD)
   - Introduction to HD computing
   - Domain Adaptation to depth images
2. **Slides on Case studies**
   - Domain Adaptation in object detection
   - **Domain Adaptation in video activity recognition**
   - Application to retrieval in Retrieval Augmented Generation
3. Optimize the tap point for the adaptor
   - When do I need an adaptor? OOD Detection using HD
   - Where do I put the tap? Some theory



Getting HD to work with Video

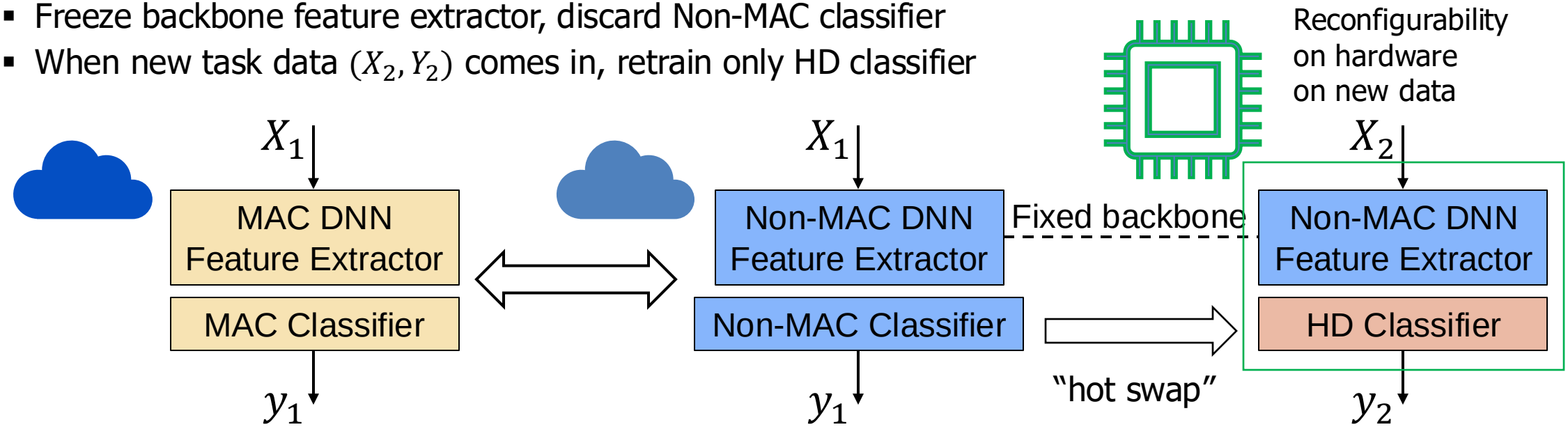
Step 1: Train 32-bit Multiply-Accumulate DNN for Video

Step 2: Train a Non-MAC DNN [1] [2]:

- Train DNN with **powers-of-two weights** and **fully integer data path**
- Combines knowledge distillation and regularization

Step 3: Train HD classifier (reconfigure)

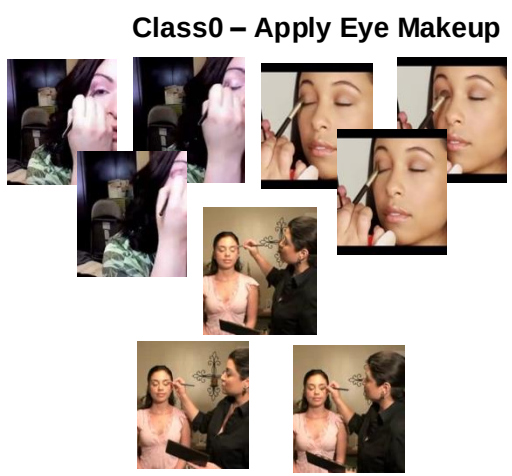
- Freeze backbone feature extractor, discard Non-MAC classifier
- When new task data (X_2, Y_2) comes in, retrain only HD classifier

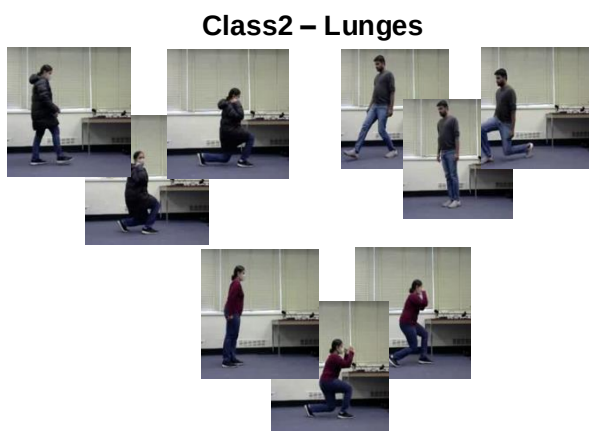


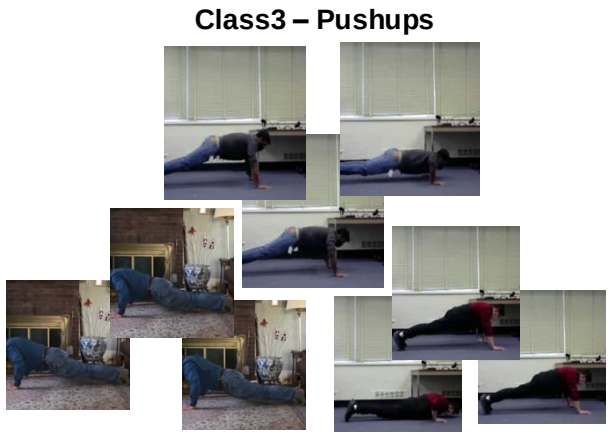
[1] S. Parajuli, **A. Raghavan** et. al., "**Generalized Ternary Connect**: End-to-End Learning and Compression of Multiplication-Free Deep Neural Networks".

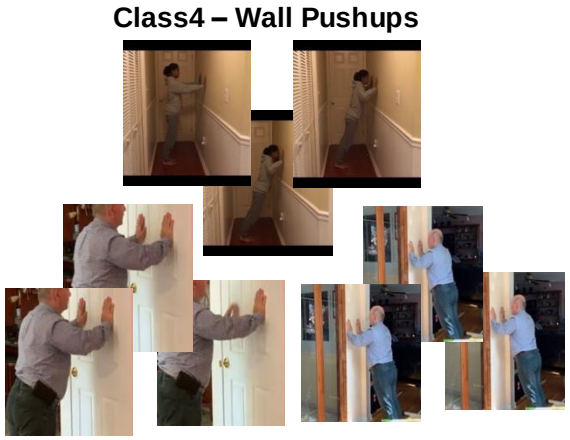
[2] **A. Raghavan**, et. al., "BIT-REGULARIZED OPTIMIZATION OF NEURAL NETS," p. 11, 2018.

Domain Adaptation on FPGA on New Activities







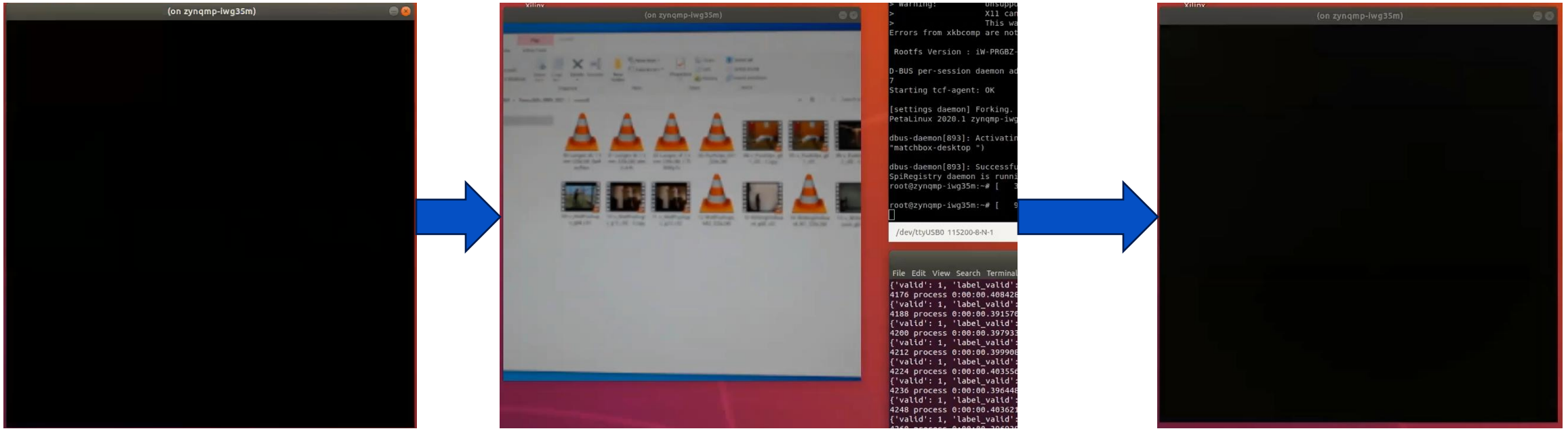


Source domain: UCF101 Videos @25Hz YouTube UCF101 / USB webcam / iPhone videos
Target domain: SRI Videos @30Hz
3 Test Videos per 5 classes
Few-shot adaptation: Adding 6th activity (Writing on board)

Isnardi, Michael, Saurabh Farkya, Indu Kandaswamy, Aswin Raghavan, David Zhang, Gooitzen van der Wal, Joe Zhang, Zachary Daniels, and Michael Piacentino. "**Hyper-Dimensional Analytics of Video Action at the Tactical Edge.**" In *GOMACTech 2021 Conference*, 2021.

Real-Time Few-Shot Adaptation using HD (FPGA)

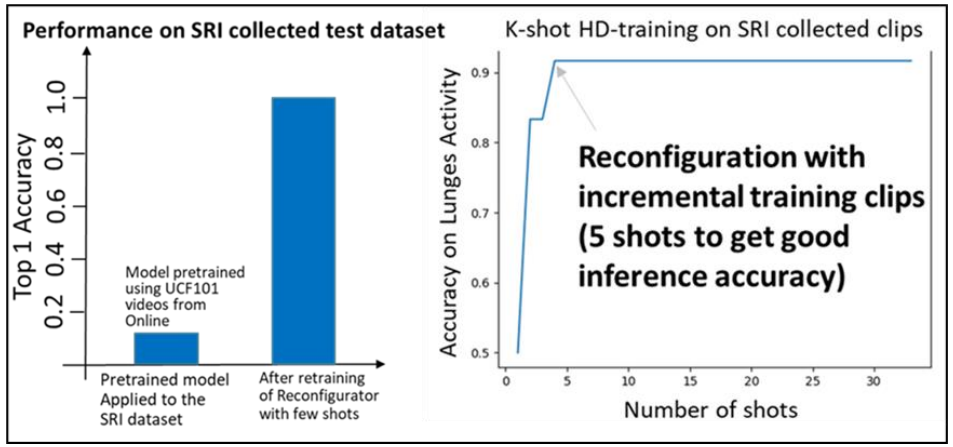
Before Adaptation



After Adaptation

Inference of 6 activities using 5 trained classes			
Activity	**Videos**	**Mean accuracy per activity**	**Overall Accuracy**
Eye makeup	3	100%	80%
Rock climbing	3	100%	
lunges	3	100%	
pushups	3	81%	
Wall pushups	3	98%	
Writing on board (**untrained**)	3	0%	

Few Shot Training – 3 Videos



Inference of 6 activities with added *reconfigured* class			
Activity	**Videos**	**Mean accuracy per activity**	**Overall Accuracy**
Eye makeup	3	100%	87%
Rock climbing	3	100%	
lunges	3	100%	
pushups	3	81%	
Wall pushups	3	41%	
Writing on board (**reconfigured**)	3	97%	

Real-Time Adaptation in Video Activity Recognition

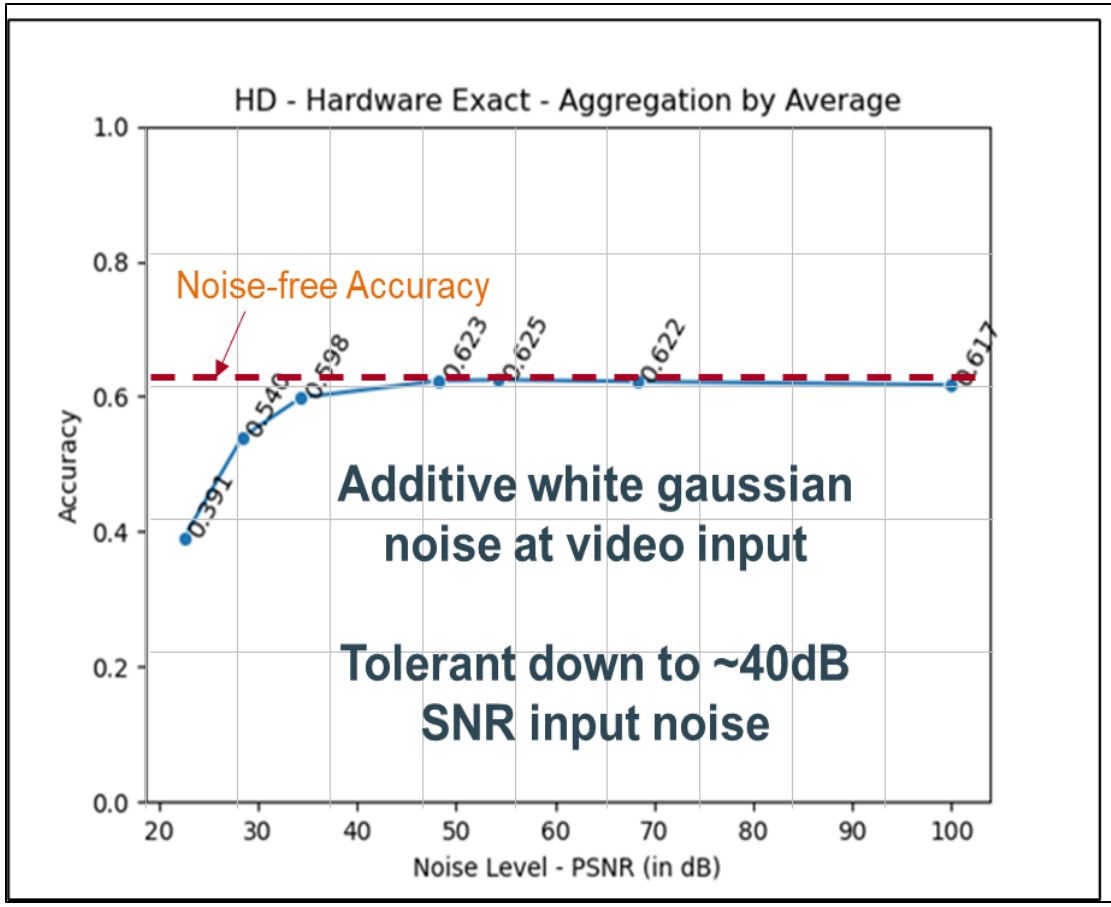
Metric	Result
Application	**Video Activity Classification**
Reference DNN	**Resnet50 + LSTM (LRCN)**
Optimal Dimensionality (>500 goal)	**D = 4096**
Network Size Reduction (10x goal)	**11x with 2.8-bit model**
Power * Latency Improvement (100x goal)	**(5.9x * 4x) in FPGA > 115x in ASIC**
Accuracy vs Noise	**Tolerant down to ~40dB SNR input noise Tolerant up to ~30% HD bit flipping**

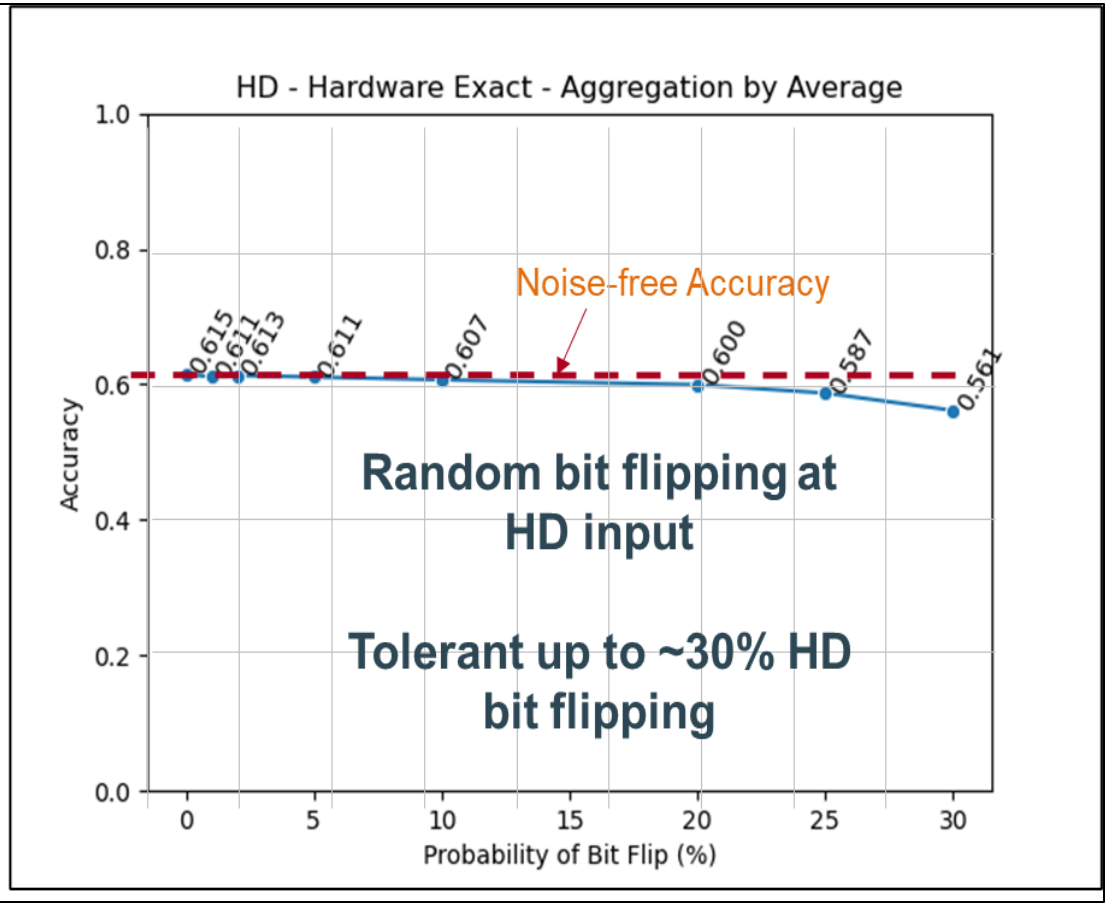
Key takeaways:

- Novel combination of a **Non-MAC Feature Extractor** and a **HD Classifier**.
- It is optimized for **video activity classification** and is suitable for **in-situ retraining at the edge**.
- Retraining a new activity happens with **high accuracy** and **forward-only HD reclassification**.
- Live FPGA demo shows **real-time video activity recognition** and **in-situ retraining.**

Isnardi, Michael, Saurabh Farkya, Indu Kandaswamy, Aswin Raghavan, David Zhang, Gooitzen van der Wal, Joe Zhang, Zachary Daniels, and Michael Piacentino. “**Hyper-Dimensional Analytics of Video Action at the Tactical Edge.”** In *GOMACTech 2021 Conference*, 2021.

Noise Resiliency of HD-based Video Activity Detector

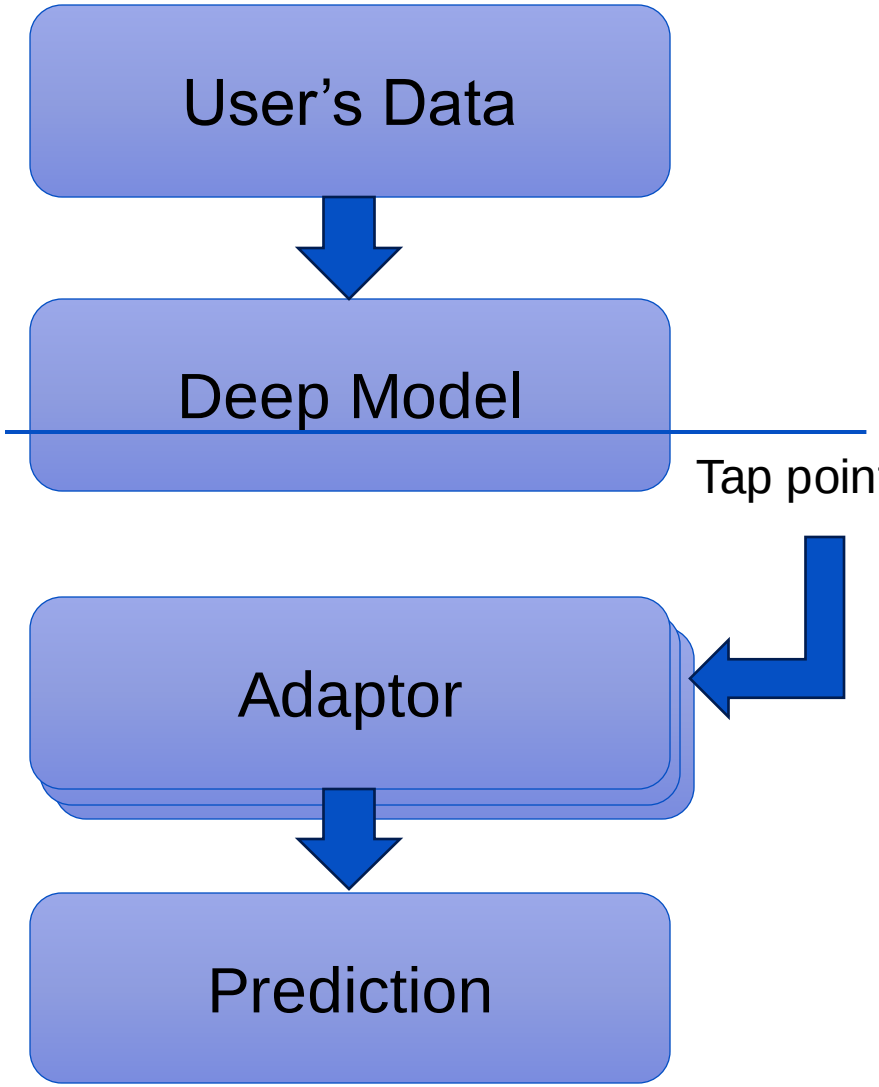




Isnardi, Michael, Saurabh Farkya, Indu Kandaswamy, Aswin Raghavan, David Zhang, Gooitzen van der Wal, Joe Zhang, Zachary Daniels, and Michael Piacentino. "**Hyper-Dimensional Analytics of Video Action at the Tactical Edge.**" In *GOMACTech 2021 Conference*, 2021.

Tutorial Outline

1. Notebooks on Hyperdimensional Computing (HD)
   - Introduction to HD computing
   - Training HD encoders
   - Domain Adaptation to depth images
2. **Slides on Case studies**
   - Domain Adaptation in object detection
   - Domain Adaptation in video activity recognition
   - **Application to retrieval in Retrieval Augmented Generation**
3. Optimize the tap point for the adaptor
   - When do I need an adaptor? OOD Detection using HD
   - Where do I put the tap? Some theory



Motivation and problem setting

Visual Grounding at the Edge Given a domain-specific image, a referring expression (e.g., user query), and a domain-specific reference database, detect (localize) the referred object in the image.

We tackle the problem of **zero-shot, retrieval-augmented**, **hardware-efficient** visual grounding on **novel domain data**.

Leverage RAG for Zero-shot Visual Grounding:

Without training on target domain data but *aided by* a reference database in target domain

Hardware-efficient: Maintain a compact DB and increase efficiency of retrieval

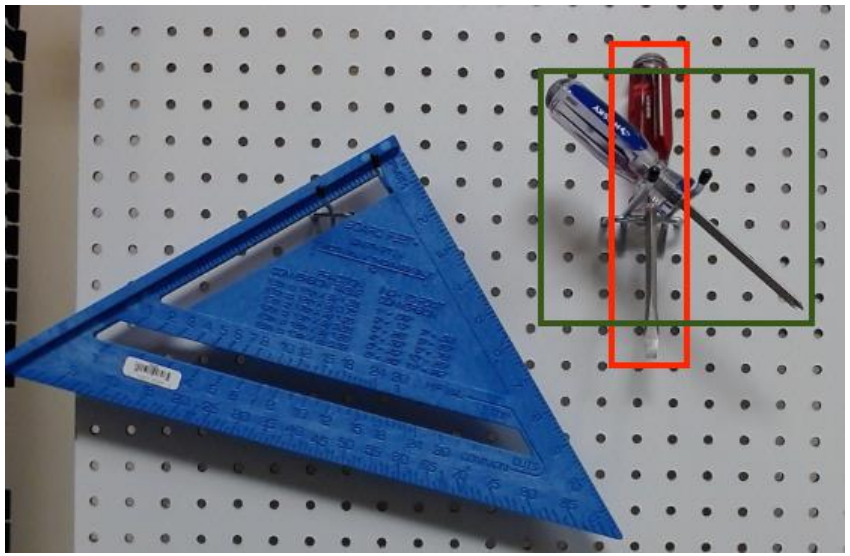
Support local DB at the edge or distributed DB at nearby edge/fog over secure comms.

Motivation and problem setting

Visual grounding at the Edge

Given an image, and a referring expression (e.g., user query), detect the referred object in the image.

You can park up ahead behind the **silver car**, next to that lamp post with the orange sign on it



For flathead screws.
For Philips head screws.

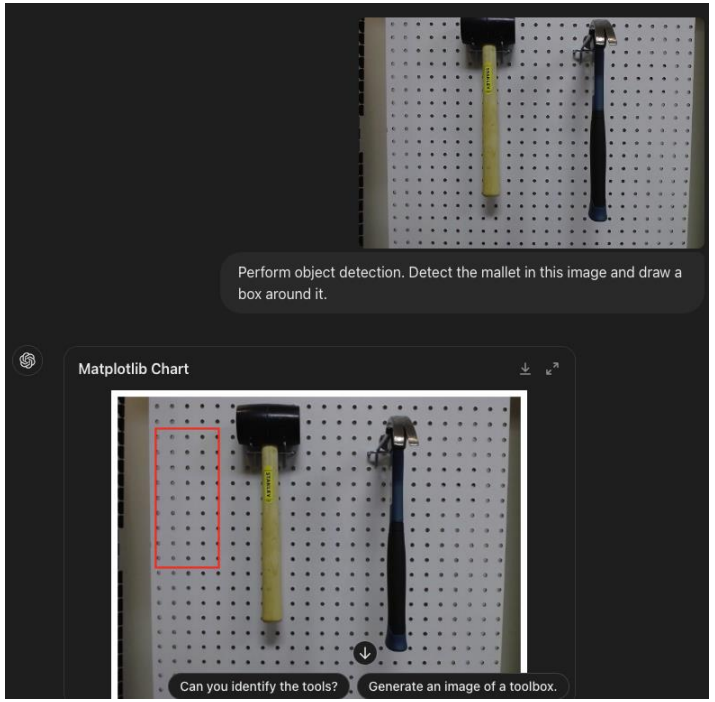
Justyna Kowalczyk, **Kikkan Randall** and **Ingvild Flugstad Østberg** at the Royal Palace Sprint, part of the FIS World Cup 2012/2013, in Stockholm on March 20, 2013. **Kikkan Randall** won the sprint cup.

Captions from Domain-Specific DB

Current approaches

- Current models fail in settings where object categories have not been encountered during training.
- Not designed to run on edge devices due to the use of computationally expensive models (e.g., Zero-shot grounding network [1])
- RAG requires access to large scale databases that usually cannot be stored on edge devices and requires connection to cloud-based servers

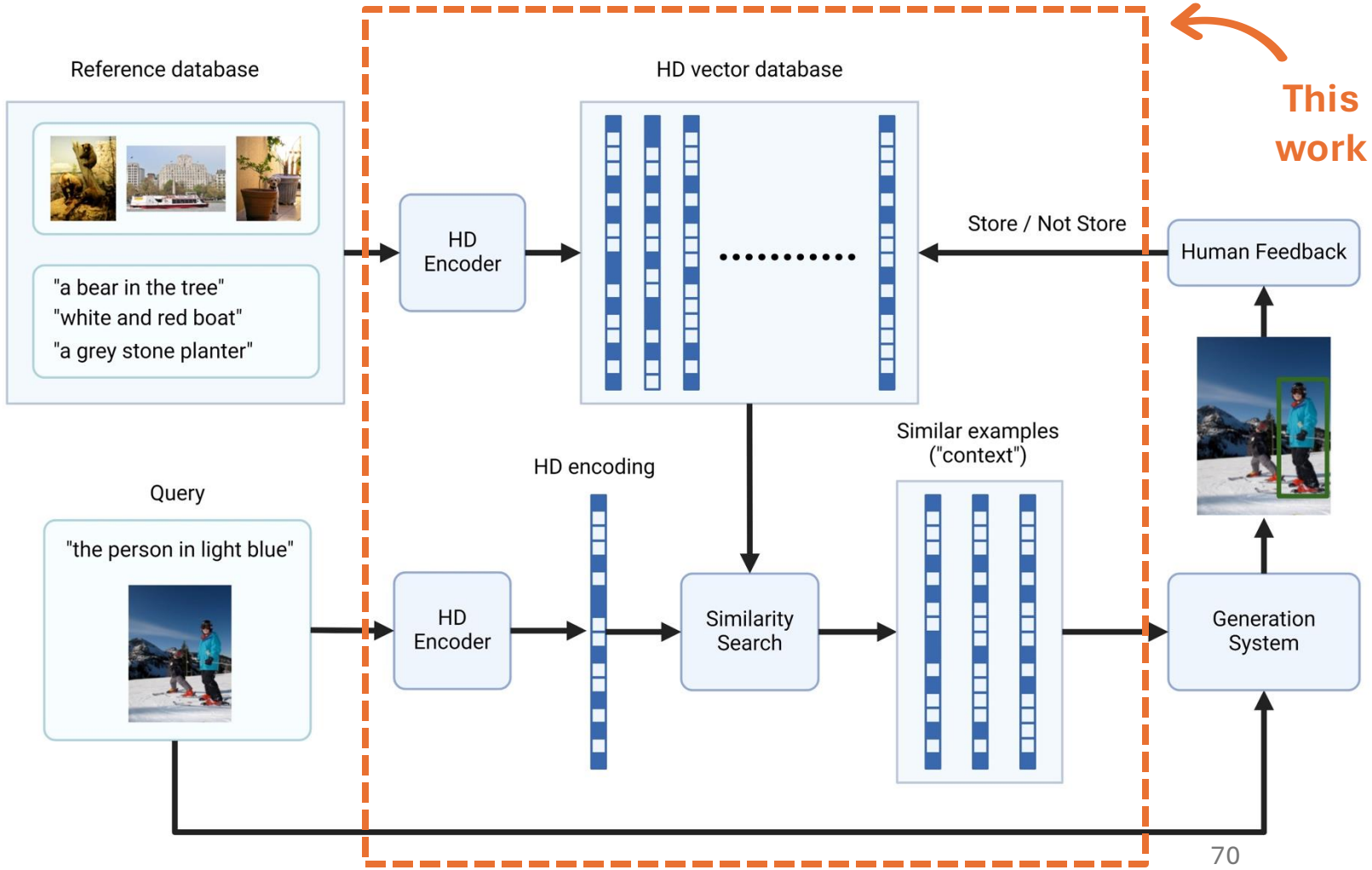
GPT-4o



[1] Sadhu, Arka, Kan Chen, and Ram Nevatia. "Zero-shot grounding of objects from natural language queries." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

Overall pipeline

- Hyperdimensional (HD) computing uses **very high-dimensional binary/ternary vectors**, related to learning with random kernels.
- We can combine embeddings of multiple objects into one HD vector with efficient operations, e.g., bundling
- HD is **more edge friendly**
  - Computation with binary variables is faster and more energy efficient
  - HD vectors can handle sparsity which leads to more compact and efficient storage



Our approach: Learned Multimodal HD Encoding (method #1)

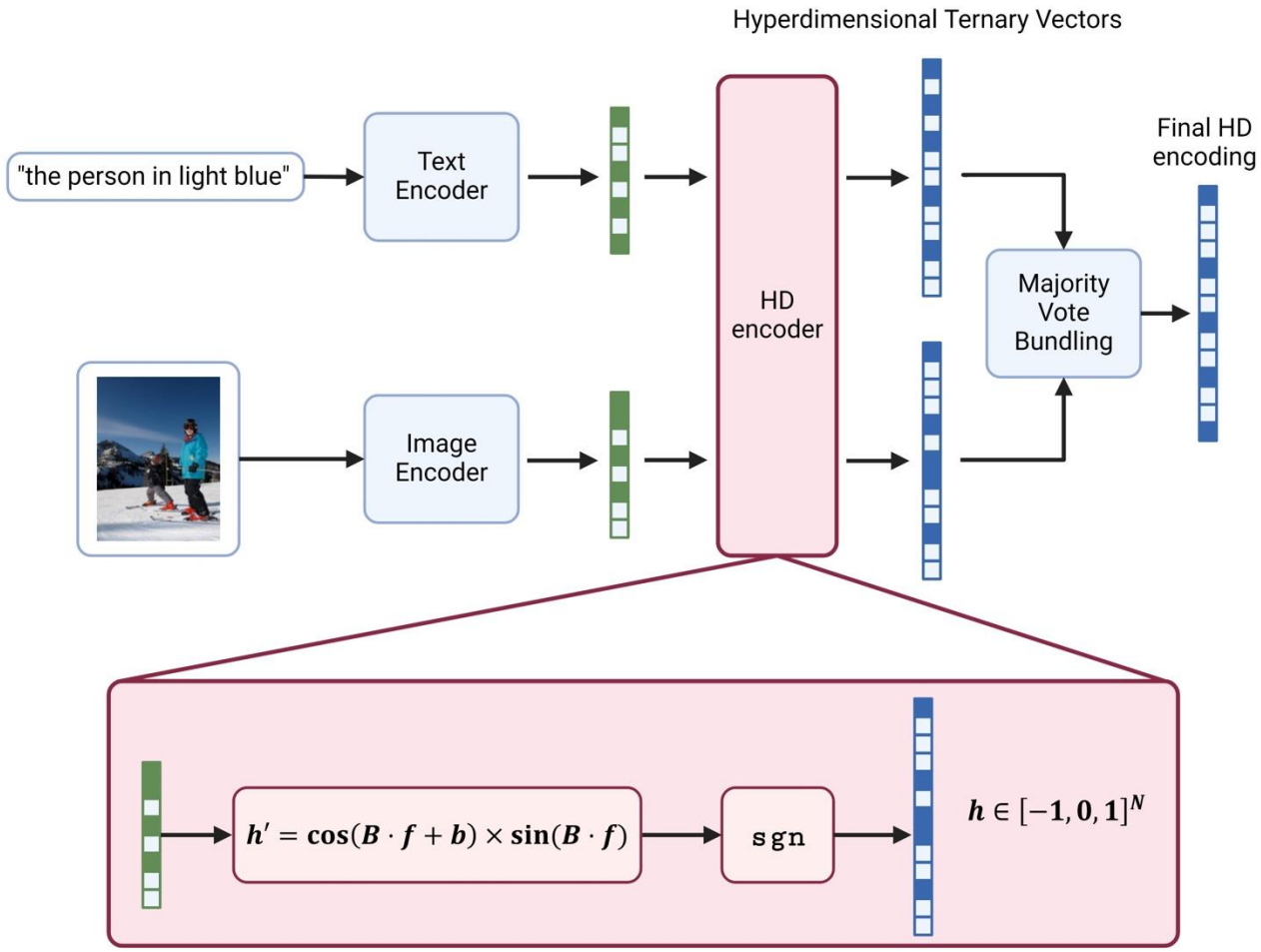
- Multimodal models like CLIP can embed text and image.
- We train a multimodal HD encoder that can retain the CLIP similarities in HD vectors: **sparse ternary vectors**.

- Let $\boldsymbol{f} \in \mathbb{R}^{\mathbf{512}}$ be the CLIP embedding
- Let $\boldsymbol{h} \in [-\mathbf{1}, \mathbf{0}, \mathbf{1}]^{\boldsymbol{N}}$ the encoded HD vector
- $\boldsymbol{N}$ is the dimensionality of the HD (hyperparameter). The parameters of the encoder are:

$$B \in \mathbb{R}^{N \times 512} \sim N(\mathbf{0}, \boldsymbol{I}), \quad b \in \mathbb{R}^{N} \sim U(\mathbf{0}, \mathbf{2}\boldsymbol{\pi})$$

- These parameters are learned by optimizing the similarity matching objective on the reference database (DB).

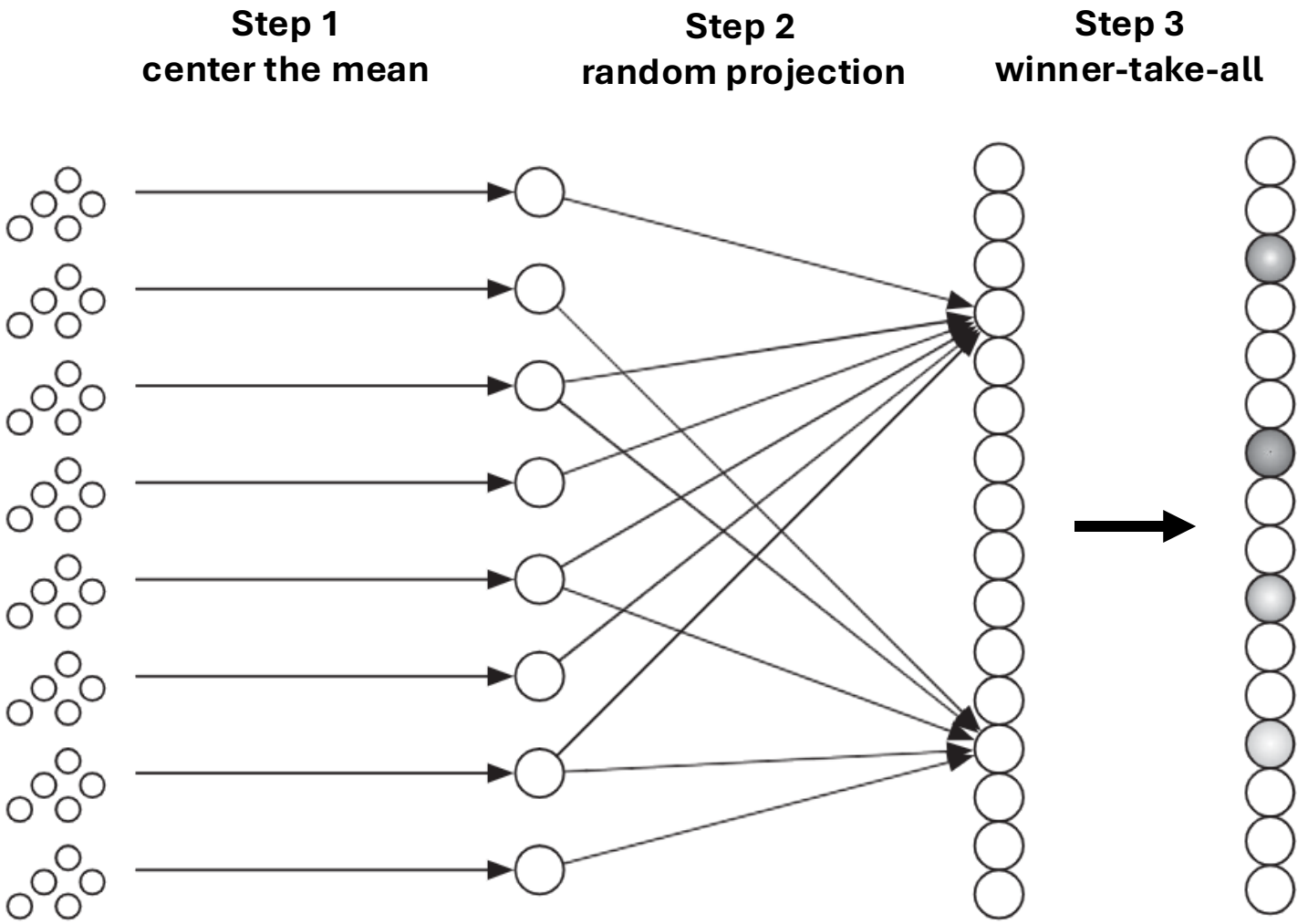
$$\text{minimize}_{\boldsymbol{B},\boldsymbol{b}} \frac{1}{D^2} \sum_{i=1}^{D} \sum_{j=1}^{D} \left(\boldsymbol{f}_i^{\top} \boldsymbol{f}_j - \boldsymbol{h}_i^{\top} \boldsymbol{h}_j \right)^2$$



Our approach: Sparse Encoding to HD (method #2)

Fly LSH algorithm by [Dasgupta et al. (2017)]

- This results in more similar inputs being mapped to highly similar hash codes or HD embeddings
- The winner-take-all nonlinearity only keeps the top k neurons with highest activities
- Sparsity can be controlled by **specifying the percentage of neurons whose activities are kept** at the winner-take-all nonlinearity
- The projection weights can be trained to optimize similarity matching

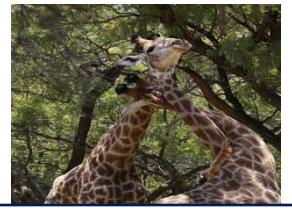


Results: training with the similarity matching objective

CLIP similarity

Image A	*Image B*	*Image C*
1.	0.84587777	0.5185583
0.84587777	1.	0.5079449
0.5185583	0.5079449	1.

Caption: ‘two giraffes eating’

Image A	*Image B*	*Image C*
		

Takeaway

Similarity matching objective works well for both HD encoding methods.

	Image A	*Image B*	*Image C*
	1.	0.7880	0.5060
	0.7880	1.	0.4740
	0.5060	0.4740	1.
Similarity to caption	**0.2700**	**0.2700**	**0.1010**

Image A	*Image B*	*Image C*
1.	0.8012	0.5016
0.8012	1.	0.5056
0.5016	0.5056	1.
0.17	**0.16**	**0.0850**

Results: Evaluating Retrieval

We measure performance using mean Average Precision (mAP).
For a given query, a retrieval is classified as a **hit if it contains the same object category** as that of the query.

For a given query $\boldsymbol{q}$, the average precision is

$$\mathbf{AP} = \frac{\#\ \mathbf{hits}}{\#\ \mathbf{retrieved\ items}}$$

The mean average precision is then

$$\mathbf{mAP} = \frac{\mathbf{1}}{\mathbf{Q}} \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{Q}} \boldsymbol{AP},$$

where Q is the total number of test queries.

Results: Retrieval when DB has Image - Caption

Bundling : $\oplus = \mathrm{sgn}(\sum h_i)$

Binding: $\otimes = -h_1 \cdot h_2$

CLIP Baseline: CLIP embeddings in the database calculated as the *sum of image and phrase embeddings*

DB Encoding
Images $\oplus$ captions

Image $\oplus$ caption	0.4046
Image only	0.3914
Caption only	0.3844

DB Encoding
Images $\otimes$ captions

Image $\otimes$ caption	0.3732
Image only	0.1498
Caption only	0.0811

Query

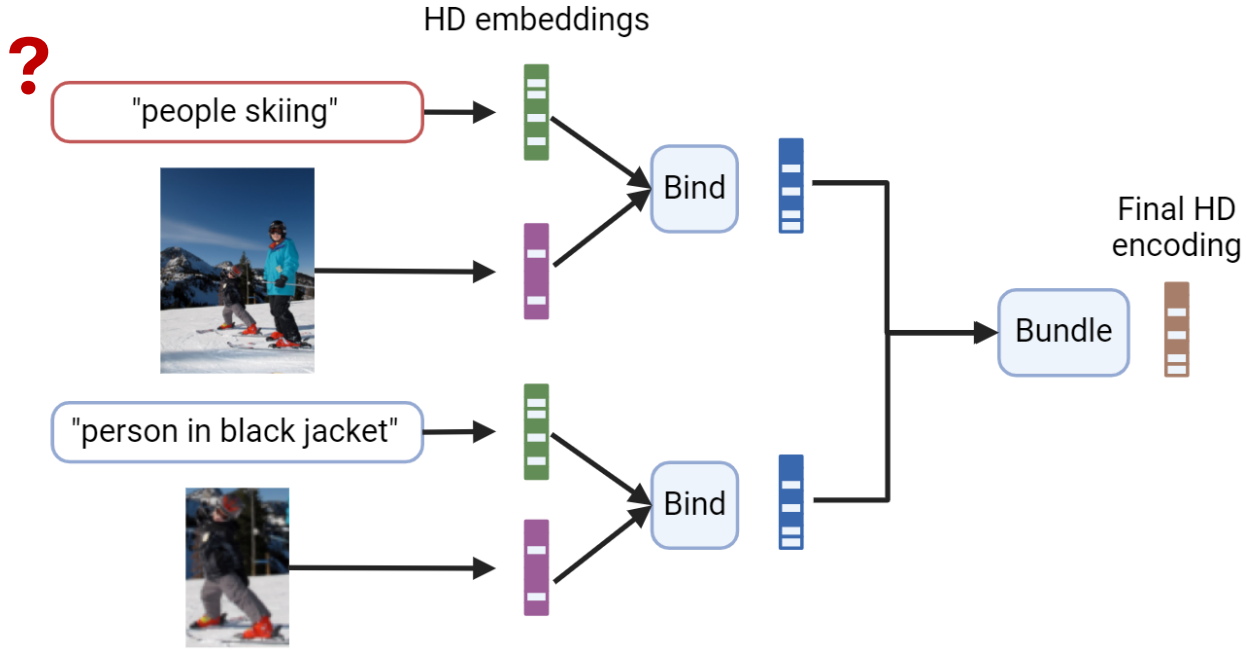
CLIP Baseline:
0.4105

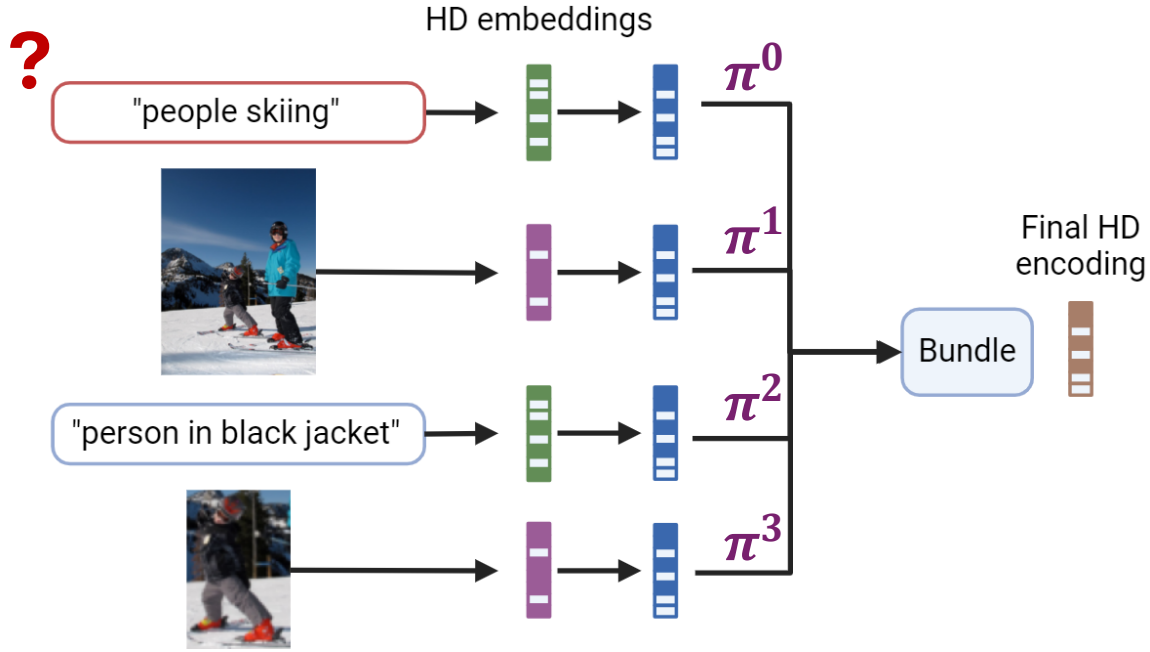
Takeaway

Bundling performs better when query does not contain parts of the DB encodings

Results: Retrieval when DB has Visual Grounding Examples

What is the best HD encoding for "<BOX> has <PHRASE> in <IMAGE>"?
Should we use **bundling or binding** to combine HD vectors of different components?





? **Do image captions help?**

π **Permutation operation to encode order**

Implementation Details

- DB has 4 components per item: image, caption, object crop, phrase
- Query is an image and phrase referring to an object
- Applied on the RefCOCO dataset
- Similarity search done using Facebook's FAISS library

Results: Retrieval when DB has Visual Grounding Examples

DB Encoding (Query with Image and Phrase)	ViT-B-32	ViT-L-14
$(\text{Image} \otimes \text{caption}) \oplus (\text{bbox crop} \otimes \text{phrase})$	0.091	0.103
$\text{Image} \oplus \text{caption} \oplus \text{bbox crop} \oplus \text{phrase}$	**0.657**	**0.679**
$\pi^0(\text{Image}) \oplus \pi^1(\text{caption}) \oplus \pi^2(\text{bbox crop}) \oplus \pi^3(\text{phrase})$	0.628 (0.577 w/o captions)	0.661 (0.582 w/o captions)
CLIP Baseline	0.629	0.663

Takeaway

Bundling performs best. Captions help performance due to overlap with query phrase.

	$\text{Image} \oplus \text{caption} \oplus \text{bbox crop} \oplus \text{phrase}$ With CLIP-ViT-B-32
HD encoding method #1	0.603
HD encoding method #2 (Fly LSH)	**0.657**

Takeaway

FlyLSH method amplifies the similarities/differences between CLIP features leading to better retrieval.

Results: example retrievals

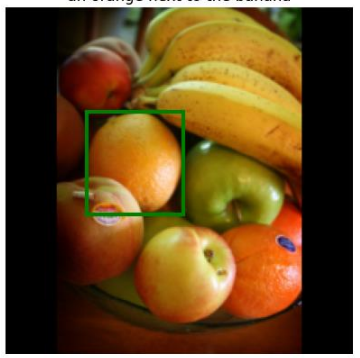
Query in the reference DB

Query

an orange next to the banana

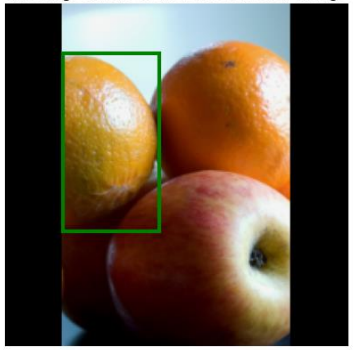
Top 6 retrievals

'an orange next to the banana'



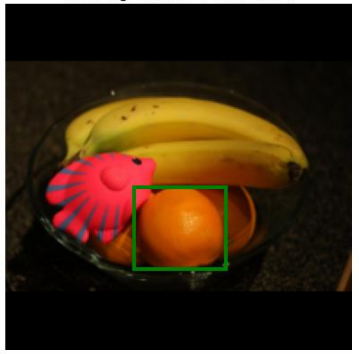
1

'an orange that is to the left of another orange'



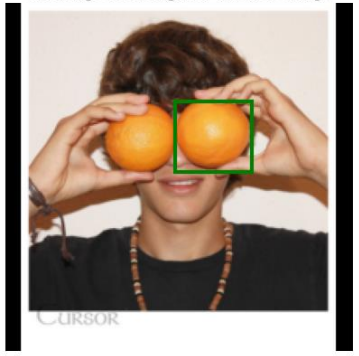
2

'an orange next to a rubber fish'



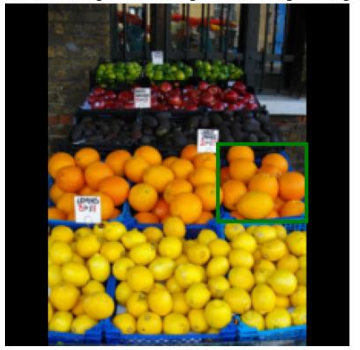
3

'an orange to the right of another orange'



4

'the basket of oranges on the right of the large orange sign'



5

'the right slice of orange in the right hand picture'

6

Results: example retrievals

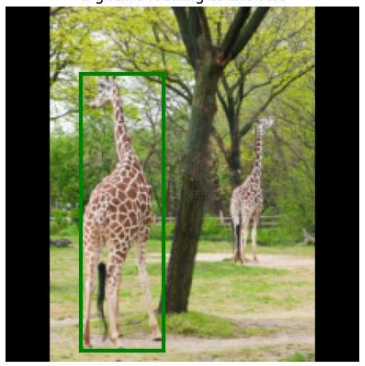
Query not in the reference dataset

Query

giraffe facing right

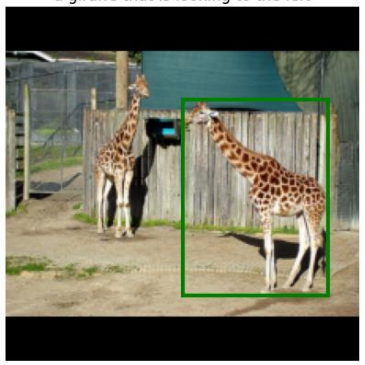
Top 6 retrievals

'a giraffe looking to the left'



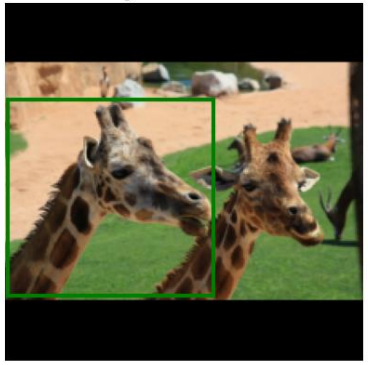
1

'a giraffe that is looking to the left'



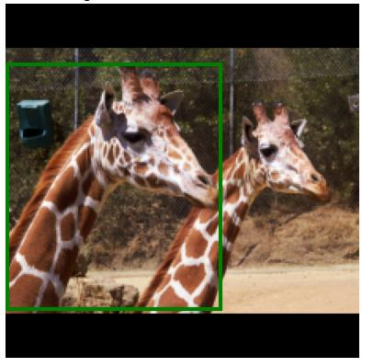
2

'a giraffe with close together horn nubs and a closed mouth'



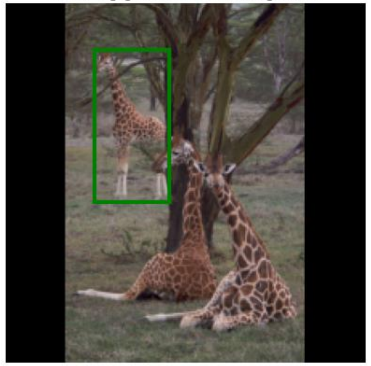
3

'a giraffe closest to the viewer'



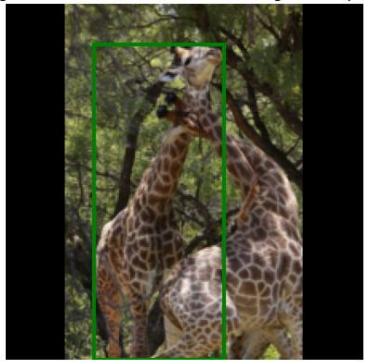
4

'standing giraffe in the background'



5

'the giraffe whos head is visible in the right hand picture'



6

Results: Specificity to Objects in the same image

Query in the reference DB

Query

a glass of water

Top 1 neighbor
'a glass of water'
distance = 0.5550

Top 2 neighbor
'a glass of water on the table'
distance = 0.7512

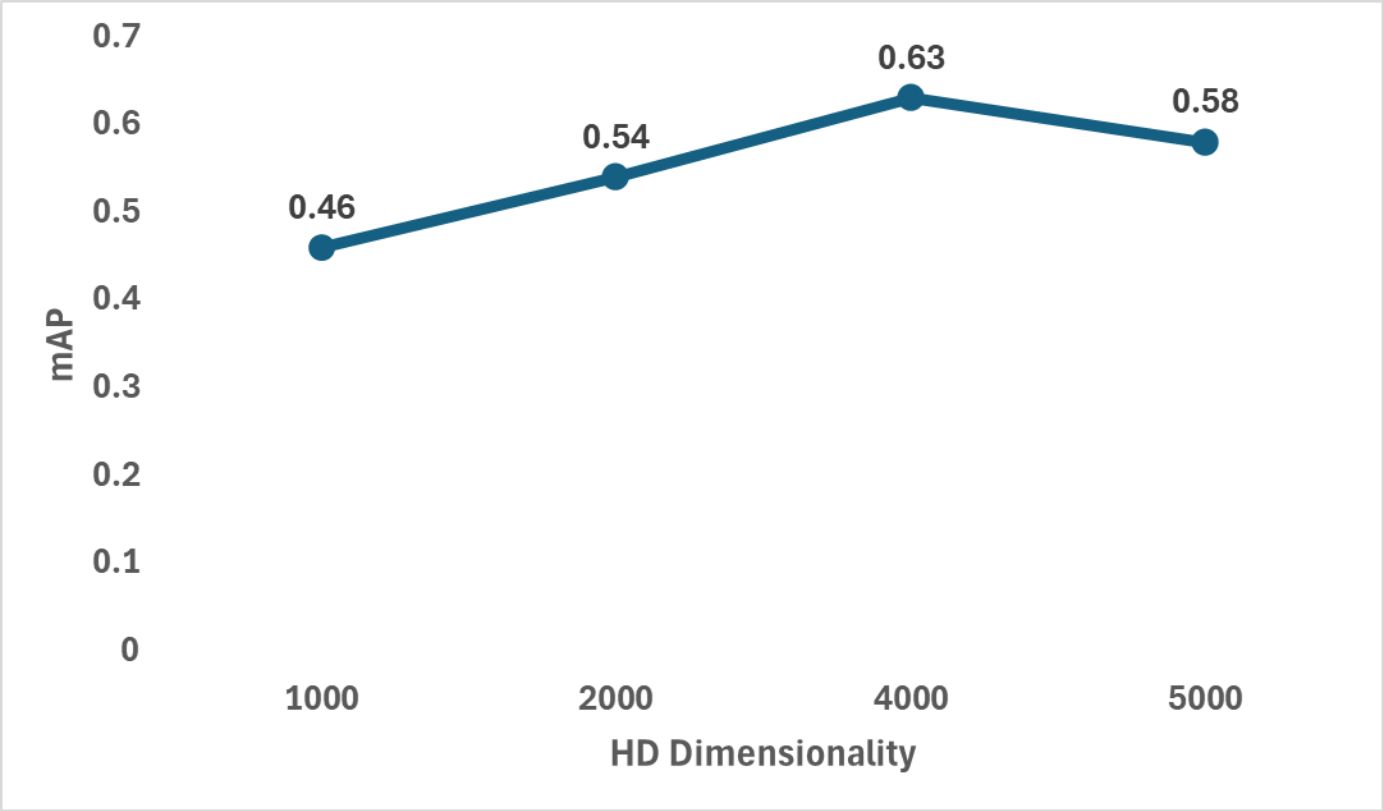
Top 3 neighbor
'a glass of water'
distance = 0.9900

Top 4 neighbor
'the white full coffee cup'
distance = 1.0286

Top 5 neighbor
'on a table a white cup is place with coffee and hearten shape is decorated'
distance = 1.0674

Results: effect of HD dimensionality

Retrieval Performance



Visual Grounding Performance

	Test set precision
1000	0.32
2000	0.32
4000	**0.36**
5000	0.33
CLIP Baseline	0.33

Performance is as good as the CLIP baseline for retrieval and is higher for visual grounding.

Section Summary

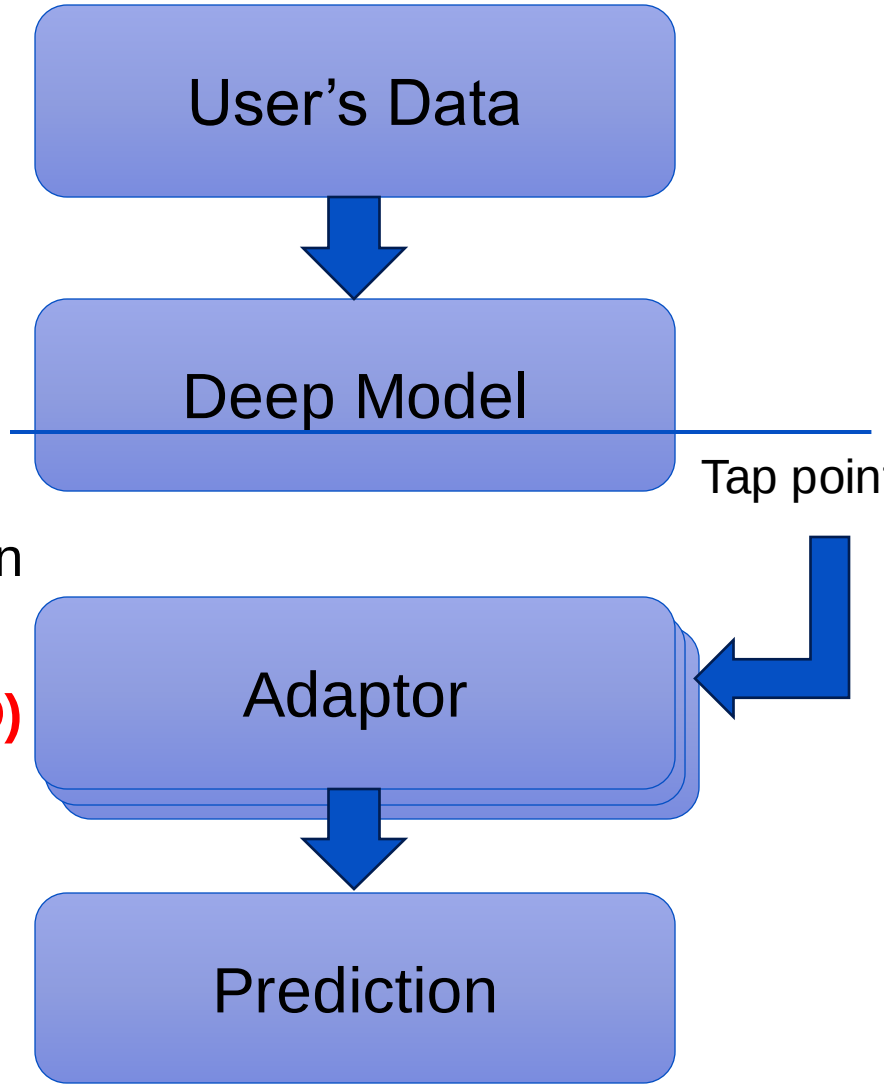
- Retrieval with HD encoding using the fly LSH algorithm performs as well as using CLIP features while offering more advantages in terms of storage efficiency and search speed
- Similarity matching training helps preserve alignment between images and text

Our HD-based retrieval pipeline is more edge friendly

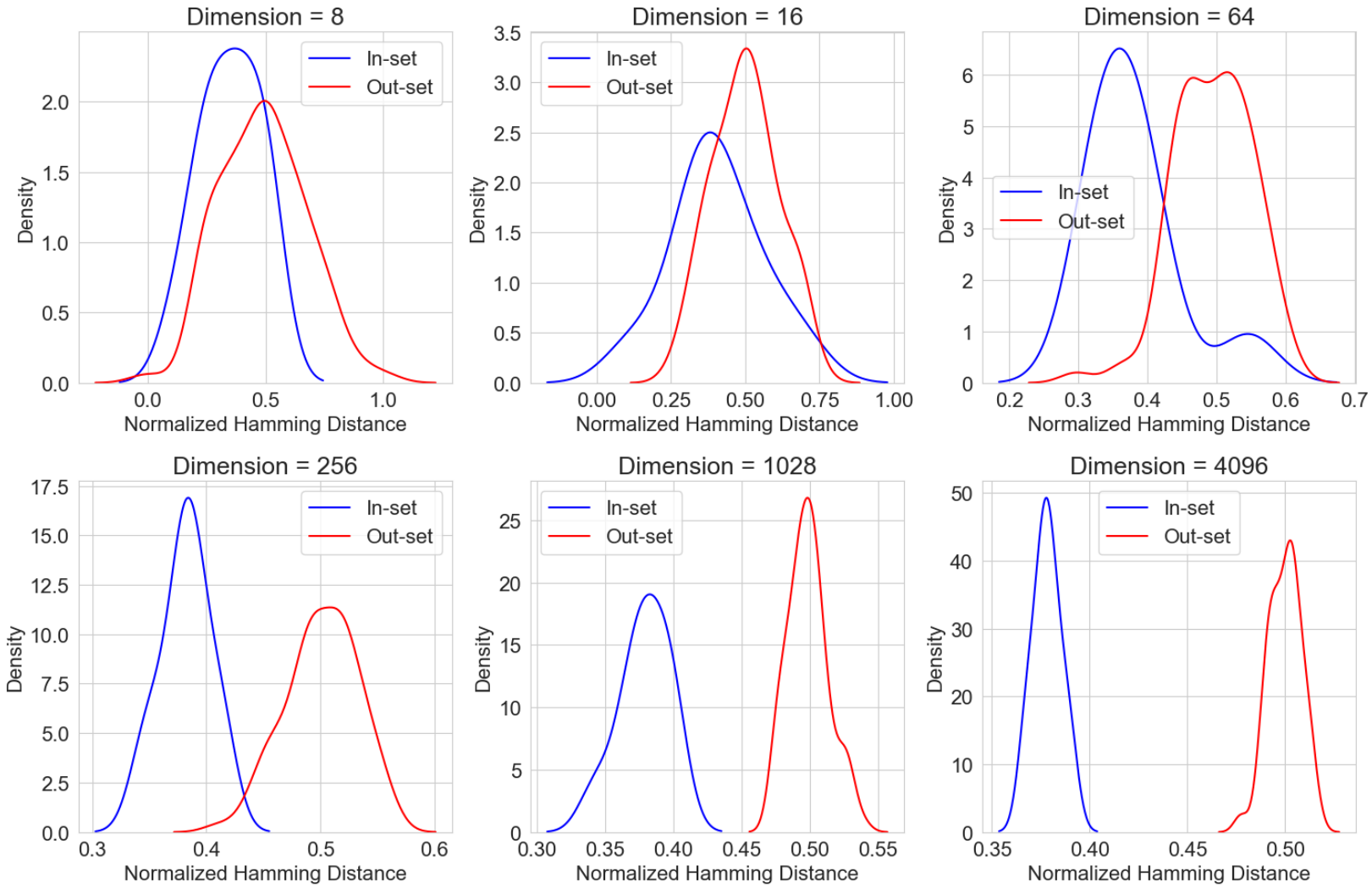
- Sparsity allows for more efficient storage
- Multiple objects (e.g., images, phrases, crops, captions) can be represented more compactly with HD vectors without losing information or increasing DB size, in contrast to CLIP features
- Our similarity-based zero-shot visual grounding approach works better than HD than with CLIP features
- Our approach can be used with generic multimodal foundation models that align vision and language.

Tutorial Outline

1. Notebooks on Hyperdimensional Computing (HD)
   - Introduction to HD computing
   - Domain Adaptation to depth images
2. Slides on Case studies
   - Domain Adaptation in object detection
   - Domain Adaptation in video activity recognition
   - Application to retrieval in Retrieval Augmented Generation
3. **Optimize the tap point for the adaptor**
   - **When do I need an adaptor? Out of Distribution (OOD) Detection using HD**
   - Where do I put the tap? Some theory

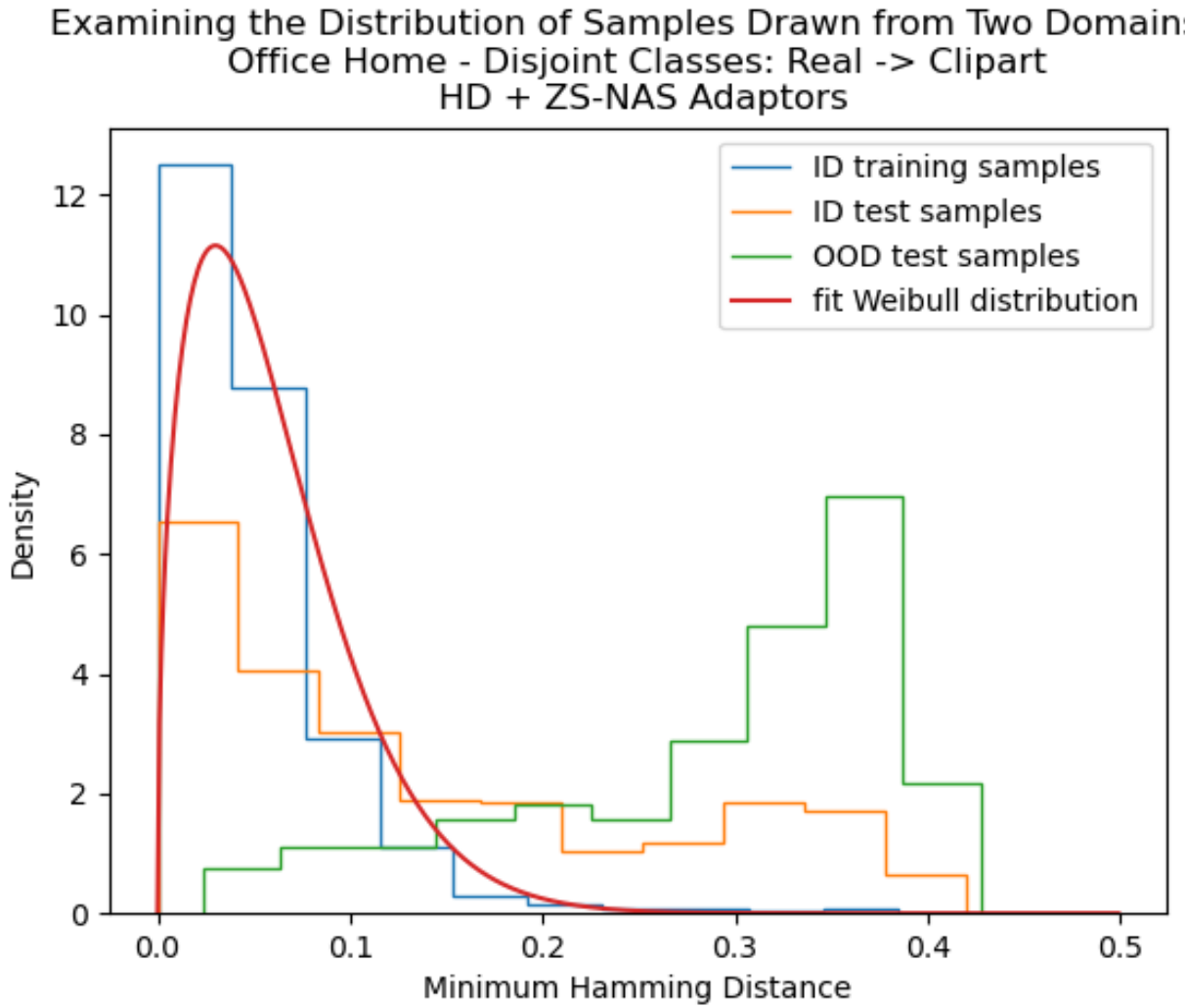


Recap from Notebook



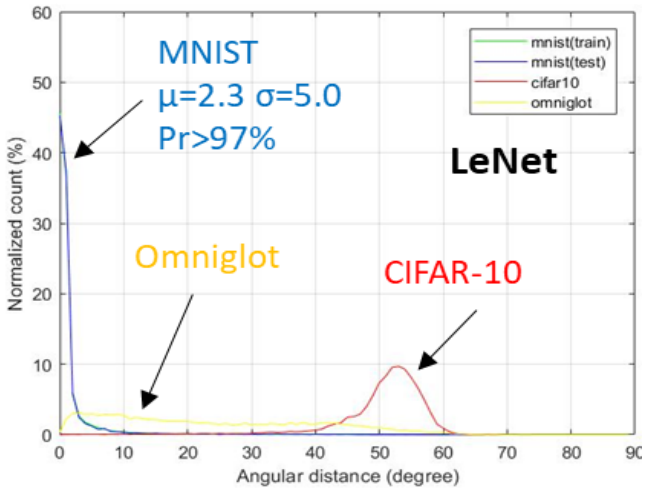
Bundling any subset of HD vectors has discriminative power at small sample sizes

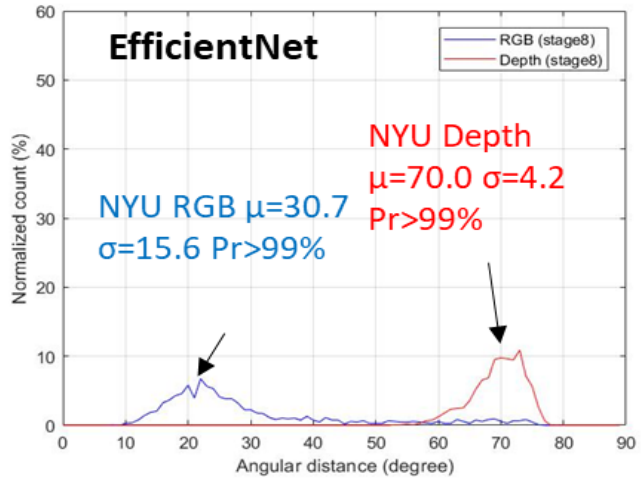
Out-of-Distribution Detection: When do I need an Adaptor?



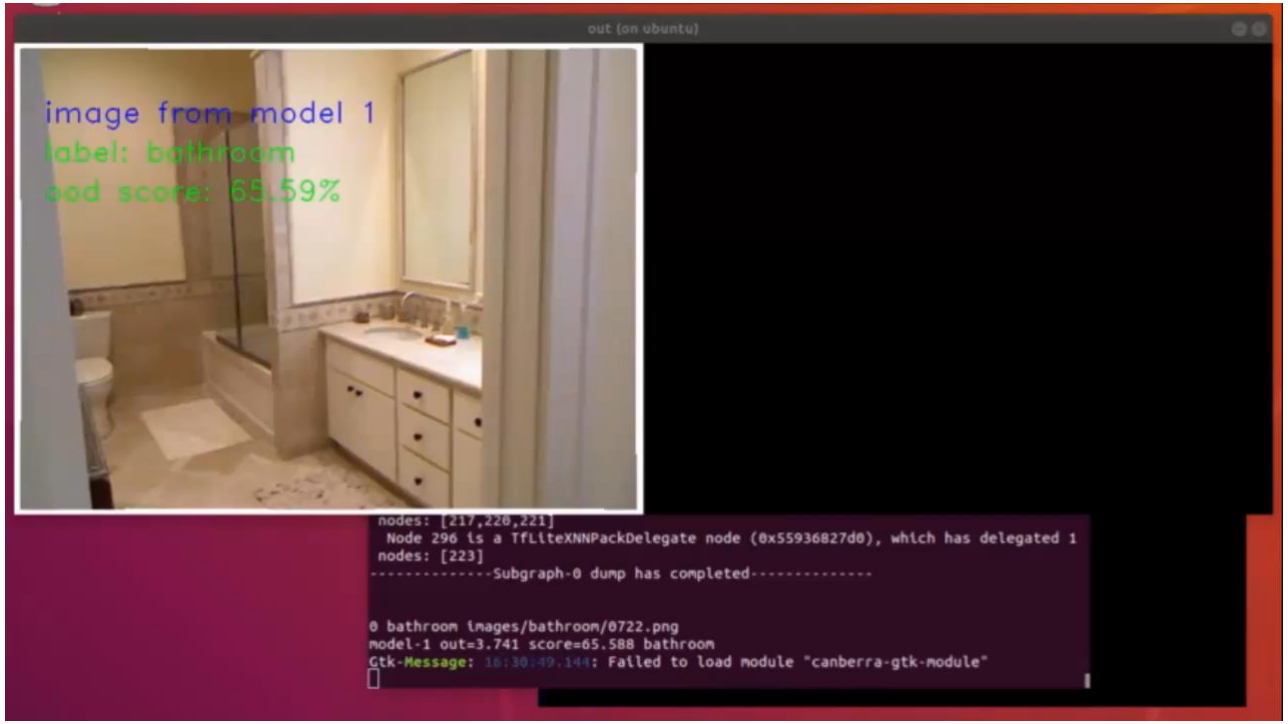
Domain Adaptation with OOD Detection

Out-of-distribution (OOD) sample detection to quantify domain shift from HD angle distances





Demo Video

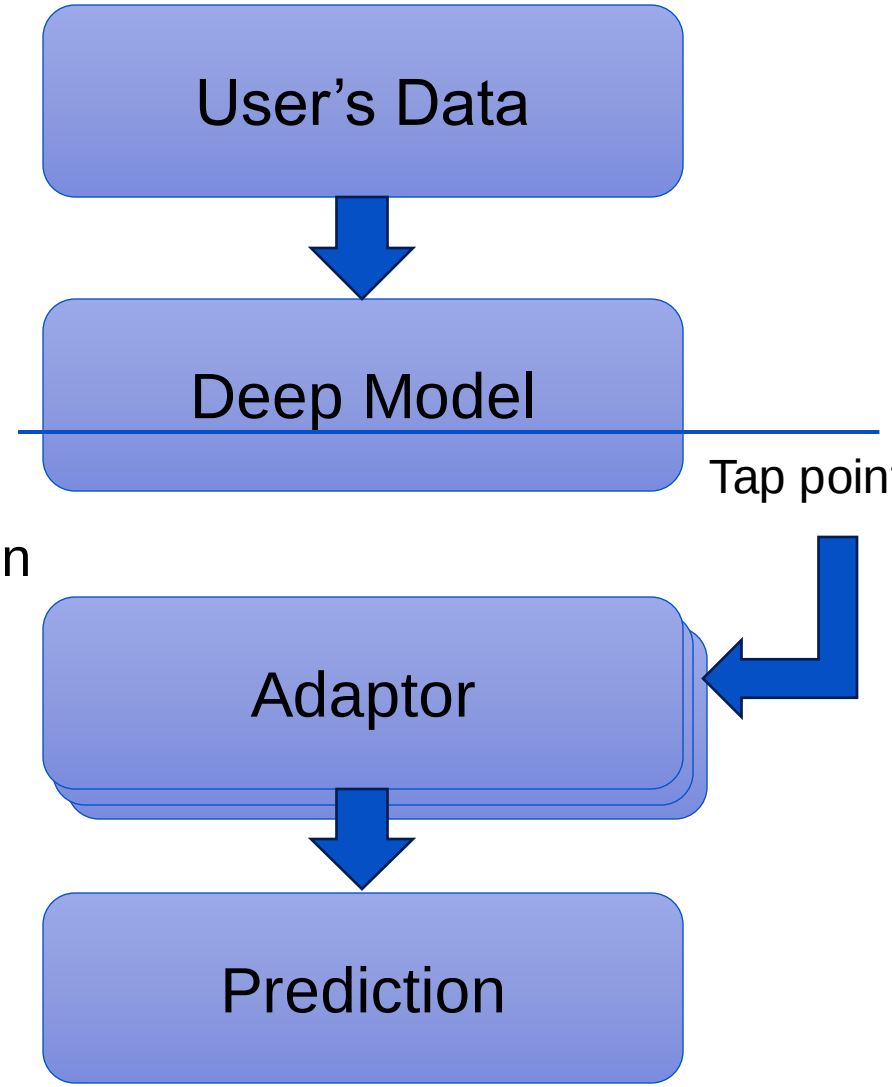


1. Encoder pretrained on NYU RGB images, adapt to NYU depth images
2. OOD adaptors are from Zero-shot NAS optimization
3. Detect ID samples** and OOD samples
4. If high confidence on OOD sample, detect it from the trained adaptors

* EfficientNet V2: The V1 used in the above diagram is for illustrative purpose
** ID samples: in-distribution samples

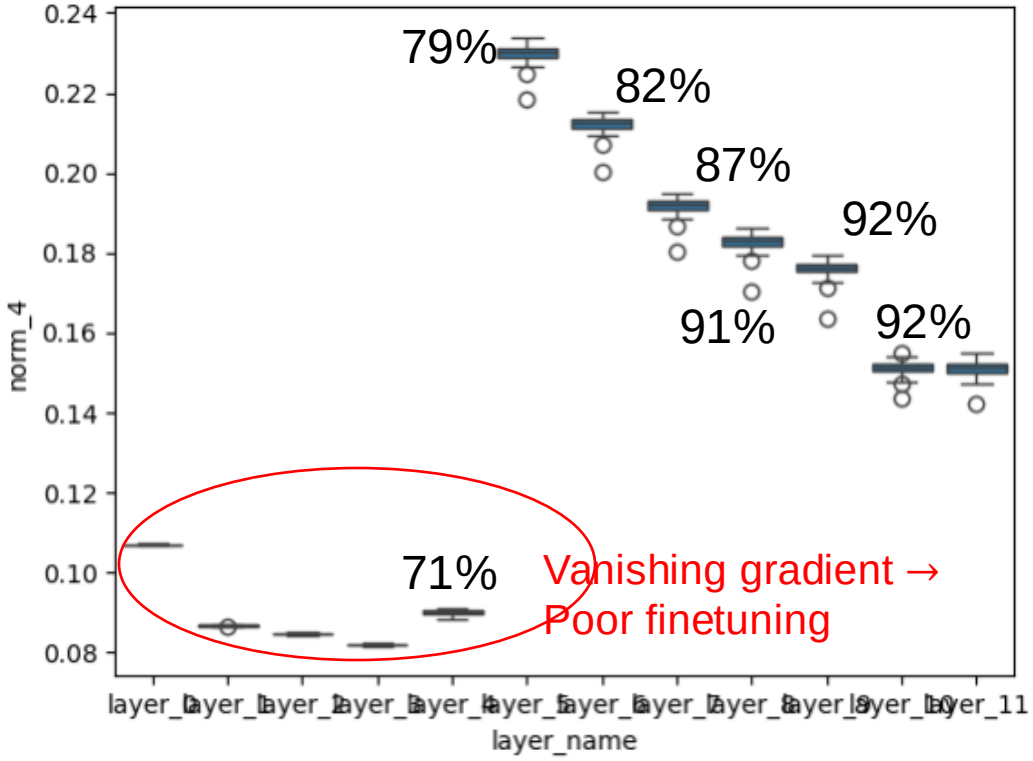
Tutorial Outline

1. Notebooks on Hyperdimensional Computing (HD)
   - Introduction to HD computing
   - Domain Adaptation to depth images
2. Slides on Case studies
   - Domain Adaptation in object detection
   - Domain Adaptation in video activity recognition
   - Application to retrieval in Retrieval Augmented Generation
3. **Optimize the tap point for the adaptor**
   - When do I need an adaptor? Out of Distribution (OOD) Detection using HD
   - **Where do I put the tap? Some theory**



Identify Adaptor Taps using the distribution of gradients

Procedure: (1) Using Off-the-shelf ViT, estimate theoretical quantities related to the Jacobian distribution at each layer. (2) Finetune an MLP at each layer to correlate.
X axis: ViT Layer # where adaptor is fit.
Y-axis: 4-norm of feature vector BEFORE training.
Annotation: Test accuracy AFTER training.



Theory predicts vanishing gradients at shallow layers for finetuning ViT. That correlates with low test accuracy.

The 4-norm might capture the diminishing returns of tap-point depth vs finetuned test accuracy.

Why do the shallow layers have vanishing gradients? See next slide.

Theorem (Hanin, Nica 2018): With He initialization (cite), the output of adaptor $M^{(d)}u$ will be distributed as

$$\frac{n_0}{n_d}\left|\left|M^{(d)}u\right|\right|_2^2 \approx^d \exp\left(N\left(-\frac{1}{2}\beta,\beta\right)\right)$$

and the Jacobian will be distributed

$$J^{(d)^T}J^{(d)} =^d M^{(d)^T}M^{(d)}$$

with $\beta = 5\sum_{i=1}^{d}\frac{1}{n_i} + \frac{\mu_4-3}{pn_1}\left|\left|\boldsymbol{u}\right|\right|_4^4$ and n_i is the width of the ith layer.

Estimate Performance of Finetuning (Forward only)

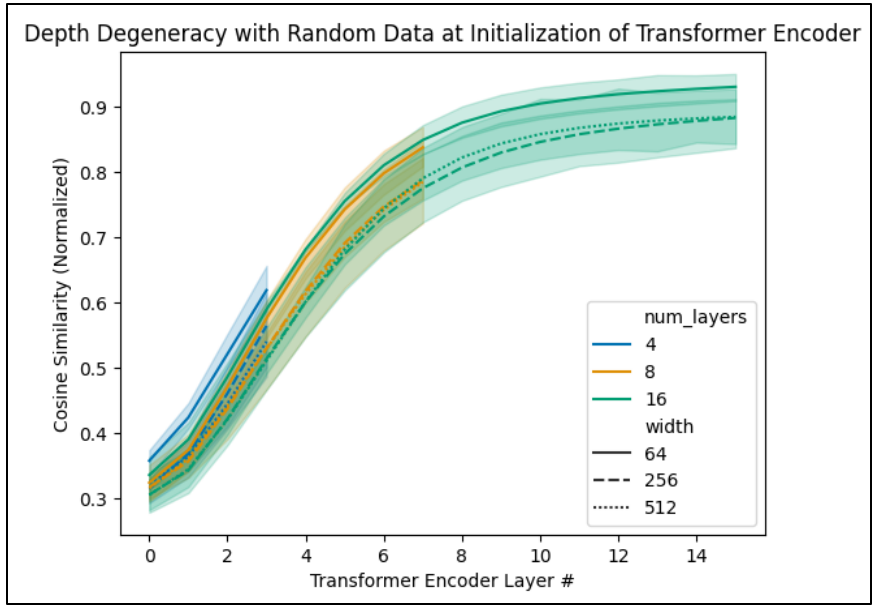
Theorem (Jakub, Nica 2023): With He initialization (cite), the angle between feature vectors at depth l evolves as

$$E[\ln \sin^2 \theta^{l+1}] = \mu(\theta^l, n_l) + O(n_l^{-2})$$

$$Var[\ln \sin^2 \theta^{l+1}] = \sigma^2(\theta^l, n_l) + O(n_l^{-2})$$

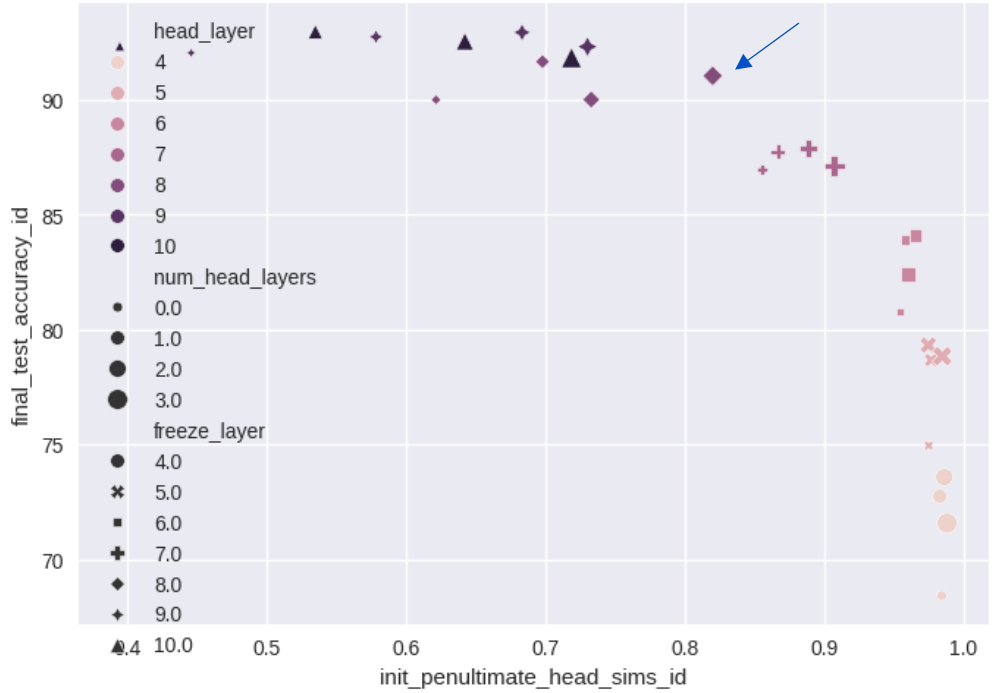
Where $\mu(\theta^l, n_l) < \ln \sin^2 \theta^l$ and $\sigma^2 \to 0$ as $l, n \to \infty$.

Our validation on BERT architecture



When this theorem predicts that the NN architecture forces inputs to become highly correlated on initialization, this serves a warning that the network may train poorly.

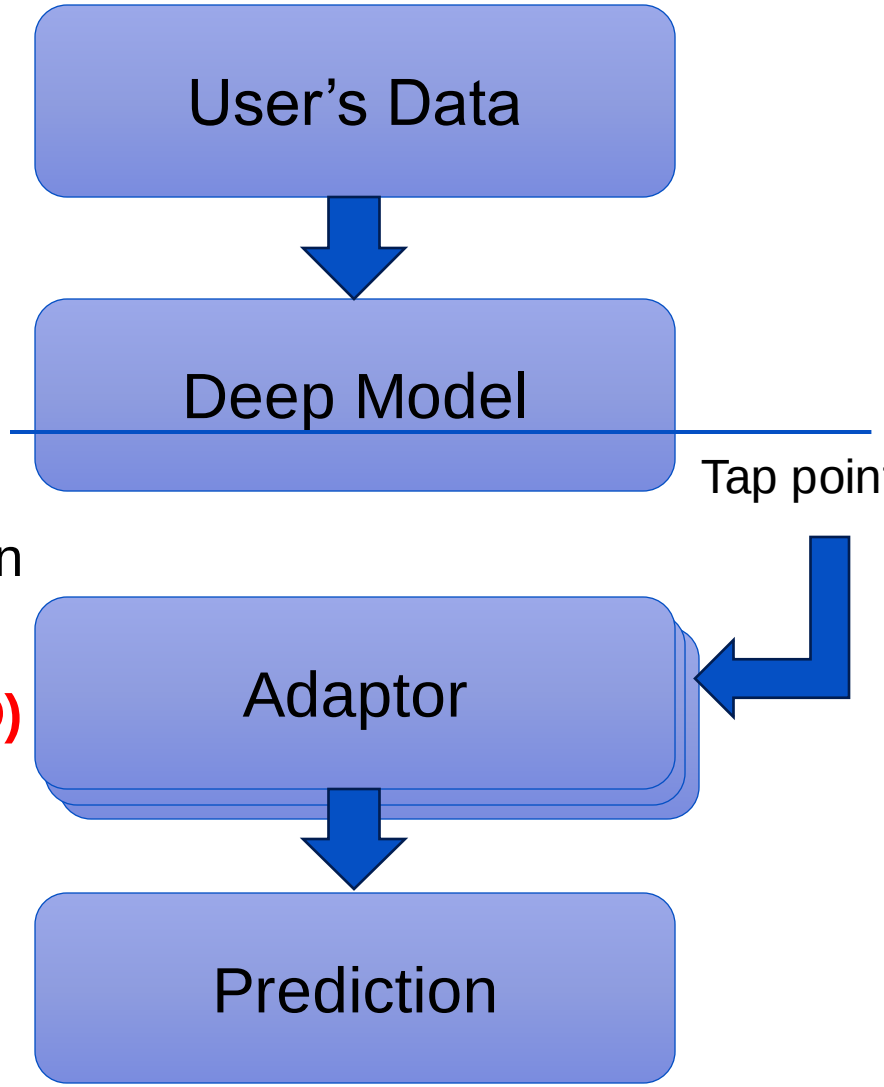
Procedure: (1) Using Off-the-shelf ViT, estimate all-pairs cosine similarity at each layer. (2) Finetune an MLP at each layer to correlate.
X axis: Cosine similarity BEFORE training.
Y-axis: Performance AFTER finetuning (CIFAR10).
Colors: Tap-point Layer # of ViT.



Avg. Cosine similarity of ViT layer outputs before finetuning is predictive of final performance. SGD is unable to adapt highly correlated representations in shallow ViT layers to high accuracy.

Tutorial Summary

1. Notebooks on Hyperdimensional Computing (HD)
   - Introduction to HD computing
   - Domain Adaptation to depth images
2. Slides on Case studies
   - Domain Adaptation in object detection
   - Domain Adaptation in video activity recognition
   - Application to retrieval in Retrieval Augmented Generation
3. **Optimize the tap point for the adaptor**
   - **When do I need an adaptor? Out of Distribution (OOD) Detection using HD**
   - Where do I put the tap? Some theory



Thank you for your attention! Thanks to my SRI collaborators!

Jun Hu, Michael Piacentino, Zachary Daniels, Abdo Sharafeldin, Michael Isnardi, Saurabh Farkya, Michael Lomnitz, Phil Miller, Indu Kandaswamy, David Zhang, Joe Zhang