# Hyp-OW: Exploiting Hierarchical Structure Learning with Hyperbolic Distance Enhances Open World Object Detection

**Thang Doan**[*], **Xin Li, Sima Behpour, Wenbin He, Liang Gou, Liu Ren**

Bosch Research North America & Bosch Center for Artificial Intelligence (BCAI)
{thang.doan,xin.li9,sima.behpour,wenbin.he2,liang.gou,liu.ren}@us.bosch.com

## Abstract

Open World Object Detection (OWOD) is a challenging and realistic task that extends beyond the scope of standard Object Detection task. It involves detecting both known and unknown objects while integrating learned knowledge for future tasks. However, the level of ”unknownness” varies significantly depending on the context. For example, a tree is typically considered part of the background in a self-driving scene, but it may be significant in a household context. We argue that this contextual information should already be embedded within the known classes. In other words, there should be a semantic or latent structure relationship between the known and unknown items to be discovered. Motivated by this observation, we propose Hyp-OW, a method that learns and models hierarchical representation of known items through a SuperClass Regularizer. Leveraging this representation allows us to effectively detect unknown objects using a similarity distance-based relabeling module. Extensive experiments on benchmark datasets demonstrate the effectiveness of Hyp-OW, achieving improvement in both known and unknown detection (up to 6 percent). These findings are particularly pronounced in our newly designed benchmark, where a strong hierarchical structure exists between known and unknown objects.

## Introduction

Advances in Object Detection (OD) have unlocked a plethora of practical applications such as robotics (Zhou et al. 2022), self-driving cars (Balasubramaniam and Pasricha 2022), manufacturing (Malburg et al. 2021), and medical analysis (Yang and Yu 2021). Recent breakthroughs in attention-based neural network architecture, such as Deformable Transformers (Zhu et al. 2021), have yielded impressive performance in these settings. However, most of these approaches assume a fixed number of classes (closed-world assumption), which is rare in reality. Continual Object Detection (Menezes et al. 2023) takes a step further by incrementally adding new classes, resulting in a distribution shift in the input and the well-known phenomenon of *catastrophic forgetting* (Kirkpatrick et al. 2017; Doan et al. 2021) where the network forgets previously learned knowledge. Open World (OW) (Bendale and Boult 2015) takes these assumptions even further, introducing the detection and integration of newly discovered classes.

While the seminal work by (Bendale and Boult 2015) introduced OW framework, further advancements by (Joseph et al. 2021) extended it in two key aspects: the detection task and continual learning. However, a significant challenge within this framework lies in the absence of annotations for unknown objects, leading to biases toward known labels and potential confusion between unknown items and the background. This bias significantly impedes the accurate identification of unknown objects and presents a major hurdle in the detection process.

Previous approaches, often relying on shared features or objectness scores (Joseph et al. 2021; Gupta et al. 2022; Zohar, Wang, and Yeung 2023), as well as clustering methods (Wu et al. 2022b; Yu et al. 2022), have failed to address a critical challenge: defining what constitutes an ”unknown” object. Currently, there is no clear definition or prior knowledge available to effectively distinguish unknowns from the background. Its interpretation greatly varies depending on the context. For example, in a driving scene, a ”debris on the road” could be considered an unknown object (Balasubramaniam and Pasricha 2022), while in a camera surveillance context, it might be perceived as part of the background (Ingle and Kim 2022). Without considering the context, these works can only learn to differentiate knowns and unknowns at low level features such as texture or shape. As a consequence, they fail to model any hierarchical structures and similarities between known and unknown items, whether at the image level or dataset level.

Acknowledging this context information, we argue that a hierarchical structure must exist between the objects to be discovered and the known items (Hosoya, Suganuma, and Okatani 2022). This hierarchy is characterized by classes that share the same semantic context, belonging to the same category such as vehicles, animals, or electronics. Such hierarchical relationships enable the retrieval of common features and facilitate the discovery of unknown objects. For instance, a model trained on objects related to driving scenes can adequately detect stop signs or traffic lights but is not expected to recognize unrelated objects like a couch or any furniture.

Given this discrepancy, we propose modeling hierarchical relationships among items to enhance the discovery of unknowns. Ideally, items belonging to the same family (or

[*] corresponding author

category) should be closer to each other while being further away from different families (e.g., animals versus vehicles). To capture these structures, Hyperbolic Distance (Nickel and Kiela 2018; Park et al. 2021), which naturally maps hierarchical latent structures, such as graphs or trees, emerges as an ideal distance metric. This has the desirable property of capturing the affinity between unknown items and known items, thereby enhancing the detection of unknown objects.

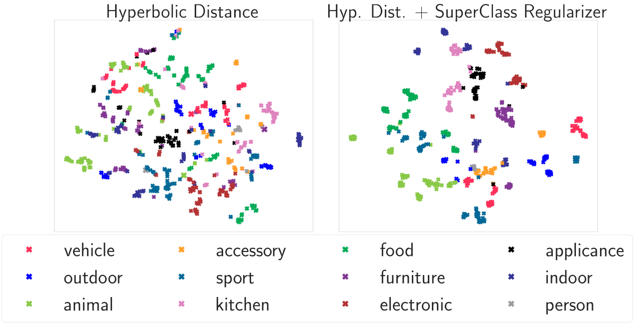

Figure 1: t-SNE plot of the learned class representations, with colors representing their respective categories. Our SuperClass Regularizer (right) learns the hierarchical structure by grouping together classes from the same category while pushing apart those from different categories.

**Contribution** Motivated by the aforementioned literature gap, we propose a **Hyp**erbolic Distance-based Adaptive Relabeling Scheme for **O**pen **W**orld Object Detection (Hyp-OW). Our contribution can be summarized in three parts:

- Hyp-OW is a simple yet effective method that learns innate hierarchical structure between objects grouping item from the same category closer while pushing classes from different categories further apart through a SuperClass Regularizer (illustrated in Figure 1, right).
- We propose an Adaptive Relabeling Scheme that enhances the detection of unknown objects by leveraging the semantic similarity between known and unknown objects in the hyperbolic space.
- Our experiments demonstrate significant improvements in both unknown recall (up to 6%) and known object detection performance (up to 5%) with Hyp-OW . These gains are particularly prominent when evaluating on our (designed) Hierarchical dataset that exhibits high inherent hierarchical structures.

# Related Work

## Open World Object Detection

The OWOD framework, introduced by (Joseph et al. 2021), has inspired many recent works due to its realistic and close-to-real-world setting that integrates newly discovered items into the base knowledge progressively. While the first stream of work was originally based on the Faster-RCNN model (Joseph et al. 2021; Yu et al. 2022; Wu et al. 2022b,a), more recent works have utilized Deformable Transformers due to their superior performance (Gupta et al. 2022; Zohar, Wang, and Yeung 2023). Joseph et al. (2021) introduced ORE, a Faster-RCNN-based model that learns class prototypes using contrastive learning with Euclidean distance. However, their approach relied on a held-out validation set where unknown items are explicitly labeled to learn an energy-based model to discriminate unknown items. (Yu et al. 2022) extended this setting by minimizing the overlap between the distributions of unknown and known classes. OW-DETR (Gupta et al. 2022) designed a novelty-branch head to relabel the top-k highest background scores as unknowns. These pseudo-labels relied on unmatched bounding box proposals with high backbone activation being selected as unknown objects. On the other hand, Wu et al. (2022a) decoupled the localization and classification tasks (introduced by Kim et al. (2022)) by learning a class-free head to localize objects. Recently, PROB (Zohar, Wang, and Yeung 2023) learned a probabilistic objectness score by learning common statistics for all objects using Mahalanobis distance (Lee et al. 2018) and considered all the remaining bounding box proposals as unknown items. During the evaluation phase, they filter out proposal bounding boxes using the latter probabilistic models.

## Class-Agnostic Object Detection

Another stream of work in the field of object detection is dubbed class-agnostic object detection, which focuses on localizing objects (Kim et al. 2022; Wu et al. 2022a; Jaiswal et al. 2021). The objective is to remove the class label information and learn a shared low-level feature representation that effectively captures the essence of an object. Kim et al. (2022) designed a pure localization head by introducing a second branch that is decoupled from the classification head. Jaiswal et al. (2021) introduced an adversarial objective loss function that penalizes label information in the encoded features. Pixel-wise class-free object detection (Gonçalves et al. 2022) used texture gray level quantization to retrieve objects. Saito et al. Saito et al. (2022) designed a new data augmentation method that pastes an annotated object onto an object-free background. Maaz et al. (2022) leveraged language models to improve unknown detection with their Multi-Modal Vision Transformers.

## Learning Hierarchical Representation with Hyperbolic Distance

Poincare embeddings have been widely used in the literature to learn hierarchical structures from complex symbolic or multi-relational data, which can be represented by graphs or trees, such as social networks or taxonomies (Nickel and Kiela 2018; Law et al. 2019). Due to its good performance, it has been applied to image classification as well (Khrulkov et al. 2020a; Yan et al. 2021; Yue et al. 2023; Ermolov et al. 2022). For example, Yan et al. (2021) used hierarchical clustering to approximate a multi-layered tree structure representation that guides the hyperbolic distance learning process. Similarly, Liu et al. (2020) used taxonomy embedding from GloVe (Pennington, Socher, and Manning 2014) to learn a finer-grained representation. Hyperbolic distance has also been used for object detection (Lang et al. 2022; Ge et al. 2022). Ge et al. (2022) was interested in learning context-object association rules by reasoning on different image scales. However, none of them leveraged the learned hyperbolic distance to retrieve unknowns items for OWOD.

# Background

## Problem Formulation

OWOD framework describes the setting where a user receives over the time a stream of $T$ tasks indexed by $t \in [1, T]$. Every task $t$ contains $C_t \in \mathbb{N}^*$ known classes (denoted by set $\mathcal{K}^{t}$[1]). The goal is to train an object detector module $f$ to accurately recognize the known classes but also discovering unknown classes (denoted by set $\mathcal{U}^t$). At the end of task $t$, $C_{t+1}$ unknown classes are labelled (with an oracle) and included in the next task $t+1$. The process repeat until task $T$ that does not contain anymore unknowns.

The dataset of task $t$ is defined as $\mathcal{D}^t = \{\mathcal{I}^t, \mathcal{Y}^t\}$ where $\mathcal{I}^t$ are image inputs and $\mathcal{Y}^t$ the corresponding labels. Each label consists of a list of bounding box locations along with their corresponding labels. We follow the setting of OWOD (Joseph et al. 2021) where a set of $K$ examplars of each class is stored in a replay buffer at the end of each task $t$ (to mitigate forgetting) to be replayed. Additionally, throughout the training, we will be storing item in a replay buffer $\mathcal{M}$ with a capacity of $m$ exemplar per class. We denote $\mathcal{B}$ the incoming batch.

## Deformable Transformers for OWOD

We adopt Deformable Transformers (Zhu et al. 2021) as our base detector, as it showed simplicity and high performance (Gupta et al. 2022). The image input is processed through a set of encoder-decoder modules to output $Q$ queries $\{\mathbf{q_i}\}_{i=1}^{Q}$, where $\mathbf{q_i} \in \mathbb{R}^d$ are the output embeddings. These queries served as input to different heads such as classification and localization heads. since $Q$ is higher than the number of ground-truth labels, the Hungarian algorithm (Kuhn 1955) is used to match the labeled ground-truth items with each query. We refer the reader to (Zhu et al. 2021) for more details.

## Hyperbolic Embeddings

A Hyperbolic space is a $n$-dimensional Riemann manifold defined as $(\mathbb{B}_c^n, g^{\mathbb{M}})$ with its Poincare ball $\mathbb{B}_c^n = \{x \in \mathbb{R}^n : c\|x\|^2 \leq 1, c \geq 0\}$ ($c$ being the constant curvature) and equipped with a Riemannian metric $g^{\mathbb{M}} = (\frac{2}{1-\|\mathbf{x}\|^2})^2 g^E$ where $g^E = \mathbf{I}_n$ is the Euclidian metric tensor. The transformation from the Euclidian to hyperbolic space is done via a bijection termed *exponential* mapping $\exp_{\mathbf{b}}^c : \mathbb{R}^n \rightarrow \mathbb{B}_c^n$.

$$\exp_{\mathbf{b}}^c(\mathbf{x}) = \mathbf{b} \oplus_c (\tanh(\sqrt{c}\frac{\lambda_{\mathbf{b}}^c\|\mathbf{x}\|}{2})\frac{\|\mathbf{x}\|}{\sqrt{c}\|\mathbf{x}\|}) \qquad (1)$$

with $\lambda_{\mathbf{b}}^c = \frac{2}{1-c\|\mathbf{b}\|^2}$ and the base point $\mathbf{b}$. The latter is often empirically taken as $\mathbf{b} = \mathbf{0}$ to simplify the formulas without impacting much the results (Ermolov et al. 2022). We will also adopt this value in our study.

Inside this hyperbolic space, the distance between two points $\mathbf{x}, \mathbf{y} \in \mathbb{B}_c^n$ is computed as:

$$d_{hyp}(\mathbf{x}, \mathbf{y}) = \frac{2}{\sqrt{c}} \arctan(\sqrt{c}\|{-\mathbf{x}} \oplus_c \mathbf{y}\|) \qquad (2)$$

where the addition operation $\oplus_c$ is defined as : $\mathbf{x} \oplus_c \mathbf{y} = \frac{(1+2c\langle\mathbf{x},\mathbf{y}\rangle+c\|\mathbf{y}\|^2)\mathbf{x}+(1-c\|\mathbf{x}\|^2)\mathbf{y}}{1+2c\langle\mathbf{x},\mathbf{y}\rangle+c^2\|\mathbf{x}\|^2\|\mathbf{y}\|^2}$.

From now on, we will denote $\mathbf{z_i}$ the projection of the queries $\mathbf{q_i}$ into the hyperbolic embedding space, i.e, $\mathbf{z_i} = \exp_{\mathbf{b}}^c(\mathbf{q_i})$. When $c \rightarrow 0$, we recover the Euclidian distance: $\lim_{c\rightarrow 0} d_{hyp}(\mathbf{x}, \mathbf{y}) = 2\|\mathbf{x} - \mathbf{y}\|$. This quantity is also related to the cosine similarity $d_{cos}(\mathbf{x}, \mathbf{y}) = 2 - 2\frac{<\mathbf{x},\mathbf{y}>}{\|\mathbf{x}\|\cdot\|\mathbf{y}\|}$ in the case of normalized vectors (See supplementary).

# Hyp-OW

In this section, we provide a detailed explanation of each module of our proposed method. Hyp-OW can be summarized by three main components (Figure 2): a Hyperbolic Metric Distance learning, a SuperClass Regularizer, and an Adaptive Relabeling Scheme to detect unknowns.

## Metric Learning with Hyperbolic Distance

We learn feature representation in the hyperbolic embedding space using a contrastive loss. The idea is to move closer features belonging to the same class $c$[2] while repelling them from features of different classes. Let's denote $\mathbf{z_i^c}$ any query $i$ matched with class $c \in \mathcal{K}$. To facilitate readability, we will omit the index $c$ whenever there is no confusion about the class context.

Throughout training, we maintain a replay buffer $\mathcal{M}$ where we store $m$ embedding features per class. For every query element $\mathbf{z_i}$ of the incoming batch $\mathcal{B}$, we sample $k = 1$ element of the same class from the replay buffer $\mathcal{M}$ denoted $\mathbf{z_{i^+}}$ (this element serves as the positive comparison) and consider the $2|\mathcal{B}| - 2$ remaining samples as the negative examples $\mathbf{z_{i^-}}$.

If we denote $\mathcal{A} = \mathcal{B} \cup \mathcal{M}$ and define a temperature $\tau_1$, the contrastive loss is then expressed as:

$$\mathscr{L}_{hyp} = -\sum_{i\in\mathcal{A}} \log \frac{\exp(\frac{-d_{hyp}(\mathbf{z_i},\mathbf{z_{i^+}})}{\tau_1})}{\sum\limits_{i^-\in\mathcal{A}\backslash\{i,i^+\}} \exp(\frac{-d_{hyp}(\mathbf{z_i},\mathbf{z_{i^-}})}{\tau_1})} \qquad (3)$$

This loss aims at attracting representation of $\mathbf{z_i}$ closer to its positive counterpart $\mathbf{z_{i^+}}$ while repelling from the negative examples $\mathbf{z_{i^-}}$, $i \in \mathcal{A}$.

## SuperClass Regularization

Numerous real-world datasets inherently possess hierarchical structures, allowing classes to be categorized. For instance, dogs and cats fall under the broader category of "animals," while cars and trucks belong to the category of "vehicles." To

[1] Whenever there is no ambiguity we will remove the task index $t$ to de-clutter the notation.

[2] By abuse of notation, we will also use $c$ for the class label since the meaning can be inferred from the context

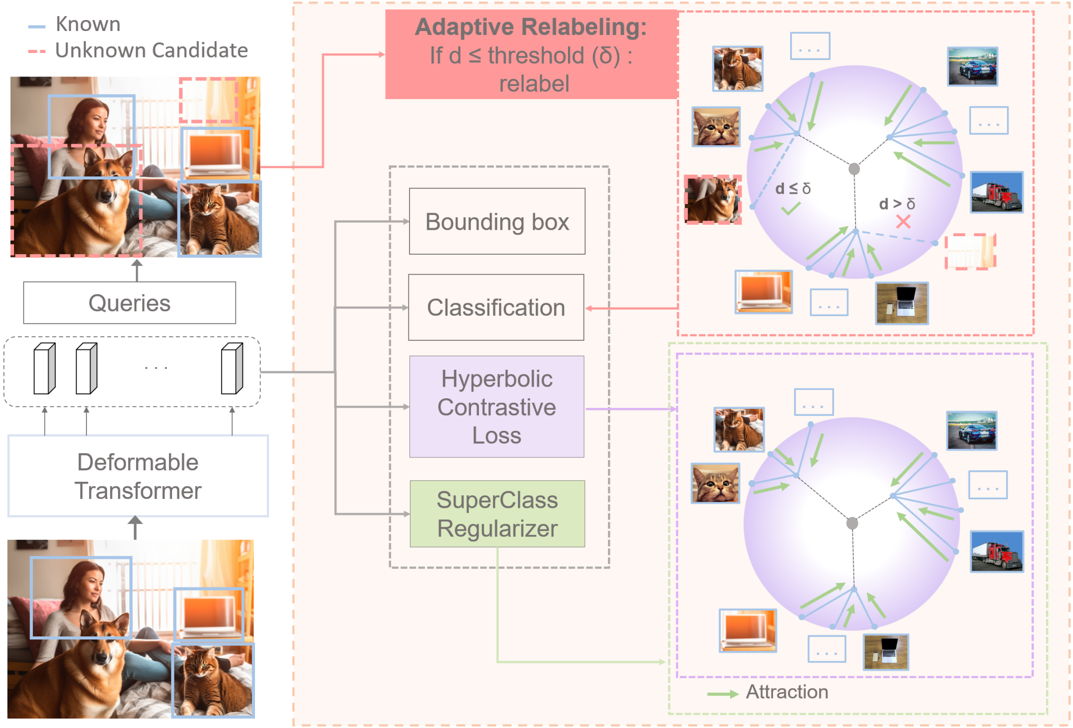

Figure 2: Overview of Hyp-OW. Comprising three core components: the *Hyperbolic Contrastive Loss* for representation learning at the class level; the *SuperClass Regularizer*, for semantic relationships at the category level; and the *Adaptive Relabeling* module, for unknown retrieval with the previously learned representation. If a distance $d$ between a candidate proposal and known items is lower than a certain threshold ($\delta$), the proposal is relabelled as unknown.

harness this inherent hierarchy, we introduce a SuperClass Regularizer (we will use ”SuperClass” and ”category” interchangeably in this context). In contrast to Eq 3, our proposed regularization encourages grouping at the SuperClass level rather than the class level.

Let’s denote $\mathcal{S}_p$ as the set of class indexes within categories $p = 1...P$ (we denote this set $\mathcal{P}$). For instance, the category ”vehicles” might encompass classes such as car, truck, bus, and so on... We approximate the category $p$ embedding by computing the Hyperbolic Average (Khrulkov et al. 2020b) (dubbed *HypAve*) of every embedding $\{\mathbf{z_i^c}\}_{i\in\mathcal{M}}$ of classes $c$ from the buffer $\mathcal{M}$ within this category ($c \in \mathcal{S}_p$) that is:

$$\overline{\mathbf{z}}_\mathbf{p} = HypAve(\{\mathbf{z_i^c}\}_{i\in\mathcal{M},c\in\mathcal{S}_p}) = \frac{\sum\limits_{i\in\mathcal{M},c\in\mathcal{S}_p} \gamma_i \mathbf{z_i^c}}{\sum\limits_{i\in\mathcal{M}} \gamma_i} \qquad (4)$$

where $\gamma_i = \frac{1}{\sqrt{1-c\|x_i\|^2}}$ is the Lorentz factor. For each element $\mathbf{z_i^c}$ of a batch $\mathcal{B}$, we extract its category embedding $\overline{\mathbf{z}}_\mathbf{p}$ ($c \in \mathcal{S}_p$) from the buffer $\mathcal{M}$. Using a temperature parameter $\tau_2$, we formulate our SuperClass regularizer as follows:

$$\mathscr{L}_{reg} = \sum_{i\in\mathcal{A},c\in\mathcal{S}_p} -\log \frac{\exp(\frac{-d_{hyp}(\mathbf{z_i^c},\overline{\mathbf{z}}_\mathbf{p})}{\tau_2})}{\sum\limits_{k\neq p}\exp(\frac{-d_{hyp}(\mathbf{z_i^c},\overline{\mathbf{z}}_\mathbf{k})}{\tau_2})} \qquad (5)$$

This loss encourages the features $\mathbf{z}_i^c$ of each class $c$ to be closer to its corresponding category embedding $\overline{\mathbf{z}}_p$, while simultaneously pushing it away from embeddings of other categories $\overline{\mathbf{z}}_k$, $k \neq p$. In essence, it fosters the grouping of similar items at the SuperClass/category level rather than the individual class level.

## Adaptive Relabeling of Unknowns with Hyperbolic Distance

We introduce our Adaptive Relabeling module, which *dynamically adapts* to the batch statistics to effectively detect unknowns. We can summarize this procedure in three steps: a) the hyperbolic mean, or *centroid*, is calculated for each class in $\mathcal{M}$,b) for all known items in an image, we compute the distance to each centroid, the greatest distance is labeled as $\delta_B$,c) for every unmatched bounding box, we calculate its distance to each centroid: if less than $\delta_B$, it is relabeled as ’unknown’ otherwise it is considered as a background.

a) We define $\underline{\mathbf{z_c}}$ [3] the hyperbolic average of class $c$ computed from the buffer $\mathcal{M}$ as:

[3] We differentiate from $\overline{\mathbf{z}}_\mathbf{p}$ with an underline to distinguish Hy-

$HypAve(\{\mathbf{z_i^c}\}_{i\in\mathcal{M}}) = \frac{\sum\limits_{i\in\mathcal{M}} \gamma_i \mathbf{z_i^c}}{\sum\limits_{i\in\mathcal{M}} \gamma_i}$ which can be seen as the centroid of each class $c$ in the hyperbolic embedding space.

b) We now use the matched queries to define: $\delta_{\mathcal{B}} = \max\limits_{i\in\mathcal{B},c\in\mathcal{K}} d_{hyp}(\mathbf{z_i}, \underline{\mathbf{z}}_\mathbf{c})$. In essence, $\delta_{\mathcal{B}}$ signifies the greatest distance from any known items in the batch $\mathcal{B}$ to all centroid $\underline{\mathbf{z}}_\mathbf{c}, c \in \mathcal{K}$ in the replay buffer $\mathcal{M}$.

c) This threshold is then utilized to relabel any unmatched query $\mathbf{z_u}$ as unknown if:

$$\min_{c\in\mathcal{K}} d_{hyp}(\mathbf{z_u}, \underline{\mathbf{z}}_\mathbf{c}) \leq \delta_{\mathcal{B}} \tag{6}$$

**Overall loss** All the aforementioned losses are finally optimized together as:

$$\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{bbox} + \alpha\mathcal{L}_{hyp} + \beta\mathcal{L}_{reg} \tag{7}$$

Where $\alpha, \beta \geq 0$ are coefficient controlling respectively the Hyperbolic and regularizer importance.

# Experiments

In this section, we start with describing our experimental setup. We then present comparative results against benchmark baselines, followed by in-depth ablation analysis of each component of Hyp-OW . Due to space limitations, we will defer detailed information to the Supplementary Material.

## Experimental Setup

**Implementation Details** We use Deformable DETR (Zhu et al. 2021) pretrained in a self-supervised manner (DINO (Caron et al. 2021)) on Resnet-50 (He et al. 2016) as our backbone. The number of deformable transformer encoder and decoder layers are set to 6. The number of queries is set to $Q = 100$ with a dimension $d = 256$. During inference time, the top-100 high scoring queries per image are used for evaluation. For our method, We used $c = 0.1$, $\tau_1 = 0.2$, $\tau_2 = 0.4$. For the set $S_p$ that defines the composition of each category (SuperClass), we adhere to the grouping used in MS-COCO dataset (Lin et al. 2014). All used hyperparameters can be found in the Supplementary.

**Metrics and Baselines** Following the current metrics used for OWOD, we utilize the mean average precision (mAP) for known items, while U-Recall serves as the primary metric to quantify the quality of unknown detection for each method (Gupta et al. 2022; Wu et al. 2022a; Zohar, Wang, and Yeung 2023; Maaz et al. 2022; Yu et al. 2022). Additional metric is discussed in Table 4. We consider the following baselines from literature: OW-DETR (Gupta et al. 2022) and PROB (Zohar, Wang, and Yeung 2023). While we included Faster R-CNN methods as informative references (ORE-EBUI (Joseph et al. 2021), UC-OWOD (Wu et al. 2022b), OCPL (Yu et al. 2022), 2B-OCD (Wu et al. 2022a)), our primary emphasis is on comparing against deformable Transformer-based methods to ensure a fair assessment following the evaluation procedure of PROB.

**Datasets** We consider two benchmarks from the literature: the OWOD Split (Joseph et al. 2021) and the OWDETR Split (Gupta et al. 2022). While the latter (OWDETR Split) strictly separates SuperClasses across tasks the first (OWOD) has mild semantic overlap between knowns and unknowns across tasks (See Supplementary Material). To closely mimic real-world scenarios, we consider a *Hierarchical Split* which ensures that each task includes at least one class from each category[4]. Each dataset is defined by four tasks $t = 1, 2, 3, 4$, containing 20 labelled classes each, for a total of 80 classes. When task $t$ starts, only the label of classes belonging to that task are revealed. For instance, task 1 only contains labels of classes from 0 to 19, while task 2 only contains labels of classes from 21 to 39, and so on. Composition of each dataset can be found in the Supplementary.

**Dataset Structure** To better understand the structure of each dataset, we define a semantic similarity measure using GloVe's embedding (Pennington, Socher, and Manning 2014). Denoting $\omega_\mathbf{c}, c \in \mathcal{K}$ ( respectively $\omega_\mathbf{k}, c \in \mathcal{U}$ ) the embedding of known (unknown) classes, the semantic overlap between knowns and unknowns for task $t \in [1, T-1]$ is:

$$S_t = \frac{1}{|\mathcal{U}^t|} \sum_{k\in\mathcal{U}^t} \max_{c\in\mathcal{K}^t} \frac{<\omega^\mathbf{c}, \omega^\mathbf{k}>}{\|\omega^\mathbf{c}\| \cdot \|\omega^\mathbf{k}\|} \tag{8}$$

Higher values indicate larger overlap between the knowns and unknowns. Figure 3 effectively quantifies the level of hierarchical structure inherent in each dataset, providing a clear insight into the composition of each split. This framework now offers us a consistent basis to evaluate each method across different hierarchical scenarios: Low regime (OW-DETR Split),Medium regime (OWOD Split) and High regime (Hierarchical Split). This metric is increasing as the number of knowns grows throughout the training

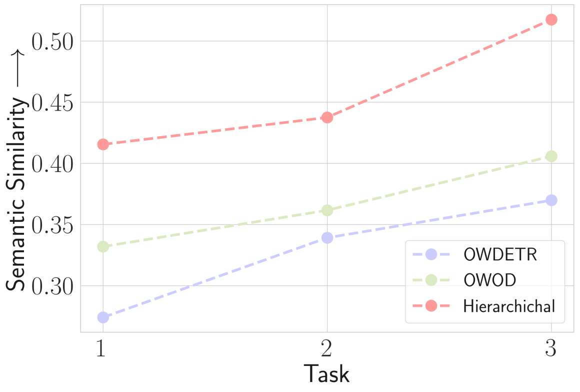

Figure 3: Semantic Similarity between knowns and unknowns across tasks for each Split.

perbolic Average of class and category

[4] While there are various ways to distribute the classes across tasks, our primary intention here is to introduce a third scenario for assessing all methods.

| | | **Task 1** | | **Task 2** | | **Task 3** | | **Task 4** |
|---|---|---|---|---|---|---|---|---|
| | Methods | U-Recall (↑) | mAP (↑) | U-Recall (↑) | mAP (↑) | U-Recall (↑) | mAP (↑) | mAP (↑) |
| Low | ORE - EBUI | 1.5 | 61.4 | 3.9 | 40.6 | 3.6 | 33.7 | 31.8 |
| | OW-DETR | 5.7 | 71.5 | 6.2 | 43.8 | 6.9 | 38.5 | 33.1 |
| | PROB | 17.6 | **73.4** | 22.3 | 50.4 | 24.8 | 42.0 | 39.9 |
| | Hyp-OW (Ours) | **23.9** | 72.7 | **23.3** | **50.6** | **25.4** | **46.2** | **44.8** |
| | Δ(Rel. Diff.) | +6.3 | ≤ 1.0 | +1.0 | ≤ 1.0 | ≤ 1.0 | +4.2 | +4.9 |
| Medium | ORE - EBUI | 4.9 | 56.0 | 2.9 | 39.4 | 3.9 | 29.7 | 25.3 |
| | UC-OWOD | 2.4 | 50.7 | 3.4 | 8.7 | 16.3 | 24.6 | 23.2 |
| | OCPL | 8.26 | 56.6 | 7.65 | 39.1 | 11.9 | 30.7 | 26.7 |
| | 2B-OCD | 12.1 | 56.4 | 9.4 | 38.5 | 11.6 | 29.2 | 25.8 |
| | OW-DETR | 7.5 | 59.2 | 6.2 | 42.9 | 5.7 | 30.8 | 27.8 |
| | PROB | 19.4 | **59.5** | 17.4 | 44.0 | 19.6 | 36.0 | 31.5 |
| | Hyp-OW (Ours) | **23.5** | 59.4 | **20.6** | **44.4** | **26.3** | **36.8** | **33.6** |
| | Δ(Rel. Diff.) | +4.1 | ≤ 1.0 | +3.2 | ≤ 1.0 | +6.7 | ≤ 1.0 | +2.1 |
| High | OW-DETR | 7.0 | 47.3 | 11.0 | 38.6 | 8.8 | 38.3 | 38.2 |
| | PROB | 29.4 | 49.6 | 43.9 | 42.9 | 52.7 | 41.3 | 41.0 |
| | Hyp-OW (Ours) | **34.9** | **49.9** | **47.5** | **45.5** | **55.2** | **44.3** | **43.9** |
| | Δ(Rel. Diff.) | +5.5 | ≤ 1.0 | +3.6 | +2.6 | +2.5 | +3.0 | +2.9 |

Table 1: Comparison on the three splits for unknown detection (U-Recall) and known accuracy (mAP). Hyp-OW improves significantly the U-Recall for the medium and high regime and known detection (mAP) for the low regime. Task 4 does not have U-Recall since all 80 classes are known at this stage.

## Benchmark Results

**Unknown Detection (U-Recall)** Table 1 shows the high performance gain of Hyp-OW over PROB on Medium regime and High regime of 3% on average (The row Δ indicates relative performance with respect to the second best algorithm). This highlights the utility of learning hierarchical structured representations and retrieving unknowns based on their similarity with known objects, as opposed to PROB, which learns a single mean representation for all objects. For the Low regime our method is performing on-par with PROB except for task 1 which shows a surprising improvement of 6 points. Overall, Hyp-OW demonstrates consistent and strong results on the three benchamrks.

**Known Accuracy (mAP)** Hyp-OW outperforms baseline benchmarks across all tasks in the Hierarchical Split and notably enhances performance for the last two tasks in the OW-DETR Split. This success can be attributed to the learned structural hierarchy, which groups classes of the same category (illustrated in t-SNE Figure 1). Moreover, our method exhibits robust performance even in the low regime. This can be attributed to the inherent presence of bounding box overlaps in object detection tasks, enabling the model to learn about co-occurring objects (refer to Supplementary Material).

| | U-Recall (↑) | mAP(↑) |
|---|---|---|
| $c = 0.0$ (Cosine Dist.) | 32.8 | 49.0 |
| $c = 0.1$ (Hyp-OW ) | **34.9** | **49.9** |
| $c = 0.2$ | 33.3 | 49.5 |
| $c = 0.5$ | 32.3 | 49.8 |

Table 2: Impact of curvature coefficient $c$ for Hierarchical Split Task 1 (↑ indicates larger values are better and ↓ indicates smaller values are better).

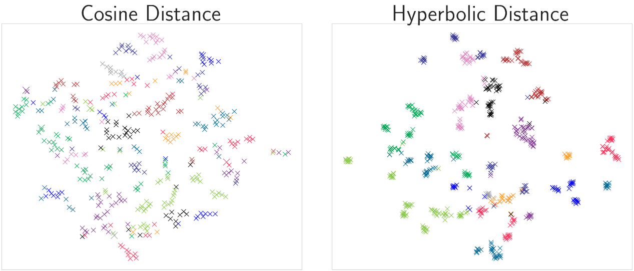

Figure 4: t-SNE plot of the learned class representations. Hyperbolic Distance tends to learns a better hierarchical structure than Cosine Distance.

## Ablation Analysis

We now aim to gain an in-depth understanding of Hyp-OW by systematically removing each component one by one to assess its direct impact (illustrated quantitatively in Table 3 for Hierarchical Split and in the Supplementary for OWOD-Split). Additionally, we perform a quantitative analysis of the unknown confusion. Qualitative visualizations can be found in the supplementary.

**Curvature Coefficient $c$:** We assess the impact of different hyperbolic distances alongside the cosine distance $c = 0$ (which is also linked with the Euclidian distance for normalized vectors, See Supplementary) in Table 2. While substituting hyperbolic distance with cosine distance ($c = 0$) negatively affects both U-Recall and mAP, higher $c$ degrades mainly the U-Recall. The hyperbolic embedding space is more suitable to learn data with latent hierarchical structure (see t-SNE plot Figure 4).

**Adaptive Relabeling:** The relabeling of unmatched bounding boxes as unknowns is governed by Eq 6. To evaluate its

| | Task 1 | | Task 2 | | Task 3 | | Task 4 |
|---|---|---|---|---|---|---|---|
| | U-Recall (↑) | mAP (↑) | U-Recall (↑) | mAP (↑) | U-Recall (↑) | mAP (↑) | mAP (↑) |
| Hyp-OW (Ours) | **34.9** | 49.9 | 47.5 | **45.5** | 55.2 | **44.3** | **43.9** |
| w/ Cosine Distance (c=0) | 32.8 | 49.0 | 46.4 | 45.4 | **55.4** | 43.2 | 43.1 |
| w/o SuperClass Regularizer | 32.0 | **50.0** | 47.1 | 45.1 | 52.9 | 43.7 | 43.5 |
| w/o Adaptive Relabeling | 34.7 | 41.2 | **47.6** | 38.9 | 54.1 | 36.5 1 | 36.1 |

Table 3: Impact of each component of Hyp-OW on Hierarchical Split. We observe that the Relabeling module (third line) significantly reduces mAP while maintaining U-Recall. On the other hand, the SuperClass Regularizer and Cosine Distance have primarily an impact on unknown detection. Task 4 does not have U-Recall since all 80 classes are known at this stage.

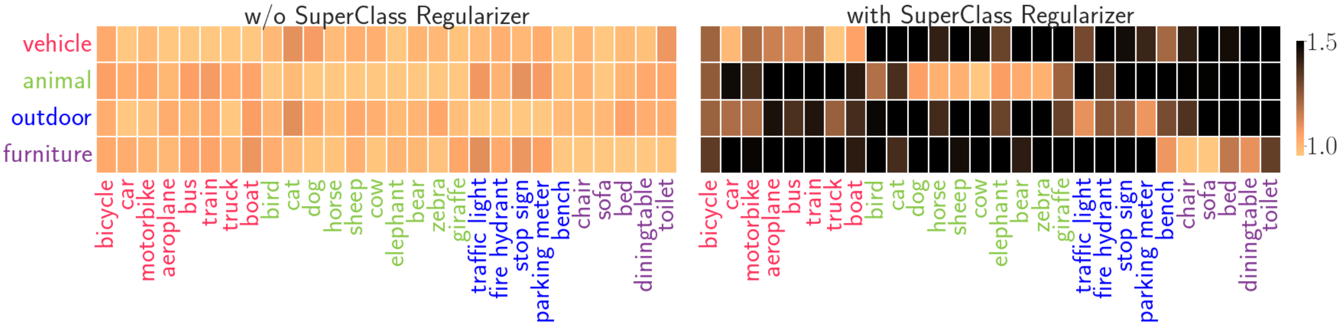

Figure 5: Hyperbolic Category - Class Distance Heatmap. The SuperClass Regularizer (right) narrows down the gap between classes (bottom) and their corresponding category embeddings (y-axis), simultaneously accentuating the separation with different categories. Without this regularizer (left), category inter-distance are much smaller (lighter color intensity).

impact, we constrast it with an alternative technique used by PROB (Zohar, Wang, and Yeung 2023), where all unmatched queries are classified as unknowns. Results are shown in Table 3 fourth row. Although the decrease in U-Recall is marginal, a noteworthy reduction in known accuracy (mAP) is observed. This decline can be attributed to the over-prediction of patches as unknowns, which results in misclassification of known objects. More detailed figures in the Supplementary show its efficacy where we notice that unknowns belonging to the same category as knowns exhibit lower Hyperbolic Distance, manifested as lighter colors.

**SuperClass Regularizer:** By setting $\beta = 0$ (Table 3: third row), we no longer enforce the grouping of items at the category level (compare t-SNE plots in Figure 1). We then observe a reduction in U-Recall of 2.9, 0.4, and 2.3 points respectively. Heatmap Figure 5 illustrates the hyperbolic distance from each class to every category's embedding (computed using Eq 4) with lighter colors indicating smaller distances. With our regularizer (right plot), we observe a wider range of values spanning from 0.7 to 2.30, compared to a smaller range of 0.78 to 1.2 without the regularizer. This emphasizes the impact of our regularizer, which effectively separates classes from distinct categories (depicted by the darker color in the right plot) while simultaneously bringing classes from similar category closer together.

**Unknowns Confusion:** We measure the A-OSE metric introduced by Joseph et al. (2021) (defined in the Supplementary Material) which quantifies the number of unknowns misclassified as knowns (lower is better). Table 4 showcases the results for Hierarchichal Split (see in Supplementary Material for OWOD Split). Comparing with PROB, Hyp-OW exhibits significantly fewer misclassifications across different tasks, with at least a twofold reduction for the first two tasks and approximately 20% less for the third task.

| | Task 1 | Task 2 | Task 3 |
|---|---|---|---|
| | A-OSE(↓) | A-OSE(↓) | A-OSE(↓) |
| OW-DETR | 42,540 | 26,527 | 20,034 |
| PROB | 14,962 | 8,929 | 5,387 |
| Hyp-OW (Ours) | **7,420** | **3,849** | **4,611** |

Table 4: A-OSE metric on Hierarchical Split. Hyp-OW exhibits a lower rate of unknowns misclassifition as knowns.

## Conclusion

The Open World Object Detection framework presents a challenging and promising setting, encompassing crucial aspects such as lifelong learning and unknown detection. In our work, we have emphasized the lack of a clear definition of unknowns and the need for a hierarchical or semantic relationship between known and unknown classes. This led us to propose Hyp-OW that focuses on learning and modeling the structural hierarchy within the dataset, which is then utilized for unknowns retrieval. Extensive experiments demonstrate significant improvement of Hyp-OW for both known and unknown detection (up to 6 percent) particularly in the presence of inherent hierarchy between classes. Future directions include leveraging knowledge from pretrained vision language models to detect desired unknowns (Zohar et al. 2023). We also hope that our hierarchical structural learning paradigm benefits adjacent fields such as OOD detection (Behpour et al. 2023), data pre-selection (Li et al. 2023) or instance segmentation (Wang et al. 2022).

## References

Balasubramaniam, A.; and Pasricha, S. 2022. Object Detection in Autonomous Vehicles: Status and Open Challenges. arXiv:2201.07706.

Behpour, S.; Doan, T.; Li, X.; He, W.; Gou, L.; and Ren, L. 2023. GradOrth: A Simple yet Efficient Out-of-Distribution Detection with Orthogonal Projection of Gradients. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Bendale, A.; and Boult, T. 2015. Towards Open World Recognition. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9650–9660.

Doan, T.; Abbana Bennani, M.; Mazoure, B.; Rabusseau, G.; and Alquier, P. 2021. A Theoretical Analysis of Catastrophic Forgetting through the NTK Overlap Matrix. In Banerjee, A.; and Fukumizu, K., eds., *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, 1072–1080. PMLR.

Ermolov, A.; Mirvakhabova, L.; Khrulkov, V.; Sebe, N.; and Oseledets, I. 2022. Hyperbolic vision transformers: Combining improvements in metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7409–7419.

Ge, S.; Mishra, S.; Kornblith, S.; Li, C.-L.; and Jacobs, D. 2022. Hyperbolic Contrastive Learning for Visual Representations beyond Objects. *arXiv preprint arXiv:2212.00653*.

Gonçalves, G. R.; Sena, J.; Schwartz, W. R.; and Caetano, C. A. 2022. Pixel-level Class-Agnostic Object Detection using Texture Quantization. In *2022 35th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, volume 1, 31–36.

Gupta, A.; Narayan, S.; Joseph, K.; Khan, S.; Khan, F. S.; and Shah, M. 2022. OW-DETR: Open-world Detection Transformer. In *CVPR*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hosoya, Y.; Suganuma, M.; and Okatani, T. 2022. More Practical Scenario of Open-set Object Detection: Open at Category Level and Closed at Super-category Level. *ArXiv*, abs/2207.09775.

Ingle, P. Y.; and Kim, Y.-G. 2022. Real-Time Abnormal Object Detection for Video Surveillance in Smart Cities. *Sensors*, 22(10).

Jaiswal, A.; Wu, Y.; Natarajan, P.; and Natarajan, P. 2021. Class-agnostic object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 919–928.

Joseph, K. J.; Khan, S.; Khan, F. S.; and Balasubramanian, V. N. 2021. Towards Open World Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021)*.

Khrulkov, V.; Mirvakhabova, L.; Ustinova, E.; Oseledets, I.; and Lempitsky, V. 2020a. Hyperbolic Image Embeddings. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Khrulkov, V.; Mirvakhabova, L.; Ustinova, E.; Oseledets, I.; and Lempitsky, V. 2020b. Hyperbolic Image Embeddings. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kim, D.; Lin, T.-Y.; Angelova, A.; Kweon, I. S.; and Kuo, W. 2022. Learning Open-World Object Proposals without Learning to Classify. *IEEE Robotics and Automation Letters (RA-L)*.

Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; Veness, J.; Desjardins, G.; Rusu, A. A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; Hassabis, D.; Clopath, C.; Kumaran, D.; and Hadsell, R. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13): 3521–3526.

Kuhn, H. W. 1955. The Hungarian Method for the Assignment Problem. *Naval Research Logistics Quarterly*, 2(1–2): 83–97.

Lang, C.; Braun, A.; Schillingmann, L.; and Valada, A. 2022. On Hyperbolic Embeddings in Object Detection. In *Pattern Recognition: 44th DAGM German Conference, DAGM GCPR 2022, Konstanz, Germany, September 27–30, 2022, Proceedings*, 462–476. Springer.

Law, M.; Liao, R.; Snell, J.; and Zemel, R. 2019. Lorentzian Distance Learning for Hyperbolic Representations. In Chaudhuri, K.; and Salakhutdinov, R., eds., *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 3672–3681. PMLR.

Lee, K.; Lee, K.; Lee, H.; and Shin, J. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31.

Li, X.; Behpour, S.; Doan, T.; He, W.; Gou, L.; and Ren, L. 2023. UP-DP: Unsupervised Prompt Learning for Data Pre-Selection with Vision-Language Models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft COCO: Common Objects in Context. *Lecture Notes in Computer Science*, 740–755.

Liu, S.; Chen, J.; Pan, L.; Ngo, C.-W.; Chua, T.-S.; and Jiang, Y.-G. 2020. Hyperbolic visual embedding learning for zero-shot recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9273–9281.

Maaz, M.; Rasheed, H.; Khan, S.; Khan, F. S.; Anwer, R. M.; and Yang, M.-H. 2022. Class-agnostic object detection with multi-modal transformer. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part X*, 512–531. Springer.

Malburg, L.; Rieder, M.; Seiger, R.; and Klein, P. 2021. Object Detection for Smart Factory Processes by Machine Learning. In *ANT/EDI40*.

Menezes, A. G.; de Moura, G.; Alves, C.; and de Carvalho, A. C. 2023. Continual Object Detection: A review of definitions, strategies, and challenges. *Neural Networks*, 161: 476–493.

Nickel, M.; and Kiela, D. 2018. Learning Continuous Hierarchies in the Lorentz Model of Hyperbolic Geometry. In Dy, J.; and Krause, A., eds., *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 3779–3788. PMLR.

Park, J.; Cho, J.; Chang, H. J.; and Choi, J. Y. 2021. Unsupervised hyperbolic representation learning via message passing auto-encoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5516–5526.

Pennington, J.; Socher, R.; and Manning, C. 2014. GloVe: Global Vectors for Word Representation. In Moschitti, A.; Pang, B.; and Daelemans, W., eds., *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543.

Saito, K.; Hu, P.; Darrell, T.; and Saenko, K. 2022. Learning to detect every thing in an open world. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings*, 268–284.

Wang, W.; Feiszli, M.; Wang, H.; Malik, J.; and Tran, D. 2022. Open-World Instance Segmentation: Exploiting Pseudo Ground Truth From Learned Pairwise Affinity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4412–4422.

Wu, Y.; Zhao, X.; Ma, Y.; Wang, D.; and Liu, X. 2022a. Two-Branch Objectness-Centric Open World Detection. In *Proceedings of the 3rd International Workshop on Human-Centric Multimedia Analysis*, HCMA ’22, 35–40. New York, NY, USA: Association for Computing Machinery.

Wu, Z.; Lu, Y.; Chen, X.; Wu, Z.; Kang, L.; and Yu, J. 2022b. UC-OWOD: Unknown-Classified Open World Object Detection. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings*, 193–210.

Yan, J.; Luo, L.; Deng, C.; and Huang, H. 2021. Unsupervised hyperbolic metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12465–12474.

Yang, R.; and Yu, Y. 2021. Artificial Convolutional Neural Network in Object Detection and Semantic Segmentation for Medical Imaging Analysis. *Frontiers in Oncology*.

Yu, J.; Ma, L.; Li, Z.; Peng, Y.; and Xie, S. 2022. Open-World Object Detection via Discriminative Class Prototype Learning. In *Proceddings of the 2022 IEEE International Conference on Image Processing (ICIP)*.

Yue, Y.; Lin, F.; Yamada, K. D.; and Zhang, Z. 2023. Hyperbolic Contrastive Learning. *arXiv preprint arXiv:2302.01409*.

Zhou, Z.; Li, L.; Fürsterling, A.; Durocher, H. J.; Mouridsen, J.; and Zhang, X. 2022. Learning-based object detection and localization for a mobile robot manipulator in SME production. *Robotics and Computer-Integrated Manufacturing*.

Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; and Dai, J. 2021. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In *International Conference on Learning Representations*.

Zohar, O.; Lozano, A.; Goel, S.; Yeung, S.; and Wang, K.-C. 2023. Open World Object Detection in the Era of Foundation Models. arXiv:2312.05745.

Zohar, O.; Wang, K.-C.; and Yeung, S. 2023. PROB: Probabilistic Objectness for Open World Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11444–11453.