O1O: Grouping of Known Classes to Identify Odd-One-Out

by

Mısra Yavuz

A Dissertation Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in

Computer Science and Engineering

KOÇ ÜNİVERSİTESİ

April 30, 2024

O1O: Grouping of Known Classes to Identify Odd-One-Out

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's thesis by

Mısra Yavuz

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

_____

Assist. Prof. Fatma Güney (Advisor)

_____

Assoc. Prof. Emre Akbaş

_____

Assist. Prof. Ayşegül Dündar

Date: _____

ABSTRACT

O1O: Grouping of Known Classes to Identify Odd-One-Out
Mısra Yavuz
Master of Science in Computer Science and Engineering
April 30, 2024

Object detection methods trained on a fixed set of known classes struggle to detect objects belonging to unknown classes in real-world scenarios. Open-world methodologies have emerged in recent years as a solution for the limitations of closed-set approaches. The main goal of open-world object detection is to detect and identify novelties while maintaining closed-set abilities. One common approach involves incorporating approximate supervision with pseudo-labels corresponding to candidate locations of objects, typically obtained in a class-agnostic manner. While previous attempts mainly rely on the appearance of objects, we propose that geometric cues provide a better solution as the source of pseudo-labels. By considering not just how objects look but also their shapes and relative locations, we aim to improve the system's ability to detect unfamiliar objects. Although additional supervision from pseudo-labels improves unknown object detection, it also introduces confusion for known classes. We observed a notable decline in the model's performance for detecting known objects in the presence of noisy pseudo-labels. To address this problem, we drew inspiration from human cognitive science. Studies about how humans mentally represent objects found that humans group objects based on their common attributes, which then helps to compare and identify the different ones given a group of objects. We applied a similar concept by organizing known object classes into a smaller set of superclasses by learning discriminative superclass representations. By doing so, our model can identify similarities between classes within a superclass, thereby facilitating the detection of unknown classes through an odd-one-out scoring mechanism. Our experiments on open-world detection benchmarks demonstrate significant improvements in unknown recall consistently across all tasks. Crucially, we achieve this without compromising known performance, thanks to better partitioning of the feature space with superclasses.

ÖZETÇE

O1O: Bilinen Sınıfların Gruplandırılması ile Aykırı Olanın Bulunması

Mısra Yavuz

Bilgisayar Bilimleri ve Mühendisliği, Yüksek Lisans

30 Nisan 2024

Sabit bir sınıf kümesi üzerinden, sadece belirli nesneler ile eğitilen nesne algılama yöntemleri, gerçek dünya senaryolarında bilinmeyen sınıflara ait nesneler ile karşılaştıklarında onları tespit etmekte zorlanır. Kapalı dünya varsamıyla eğitilen modellerin zayıf yönlerini çözmek adına, son yıllarda açık dünya yöntemleri ortaya çıkmıştır. Açık dünya nesne tespitindeki temel amaç, kapalı dünya varsımıyla eğitilen modellerin yeteneklerini korurken, yeni ve değişik olanları da tespit edebilmek ve tanımlayabilmektir. Yaygın bir yaklaşım olarak, nesnelerin olası konumlarına karşılık gelen ve genellikle nesne kategorilerinden bağımsız bir şekilde elde edilen sözde etiketler, modelin eğitiminde ek denetim sinyali olarak kullanılabilir. Önceki girişimler esas olarak nesnelerin görünümüne dayanırken, biz geometrik ipuçlarının sözde etiket kaynağı olarak daha iyi bir çözüm sağlayabileceğini öneriyoruz. Nesnelerin yalnızca görünüşünü değil aynı zamanda şekillerini ve göreceli konumlarını da göz önünde bulundurarak sistemin tanıdık olmayan nesneleri tespit etme yeteneğini geliştirmeyi amaçlıyoruz. Sahte etiketlerden gelen ek denetim, bilinmeyen nesne tespitini geliştirse de, bilinen sınıflar için de kafa karışıklığına neden olmaktadır. Gürültülü sözde etiketleri kullandığımızda, modelin bilinen nesnelerin tespiti performansında dikkate değer bir düşüş gözlemledik. Bu sorunu çözmek için bilişsel bilimden ilham aldık. Nesnelerin insan zihninde nasıl temsil edildiğine ilişkin yapılan araştırmalarda, insanların nesneleri ortak özelliklerine göre gruplandırdığı, ve bu özellik gruplarının daha sonra bir grup nesneyi karşılaştırmada kullanılarak aralarındaki farklı nesneyi tanımlamaya yardımcı olduğu bulundu. Biz de benzer bir konsepti uygulayarak, bilinen nesne sınıflarını daha küçük bir süper sınıf kümesi hiyerarşinde düzenledik, ve bu üst sınıflar için ayırt edici temsiller öğrendik. Böylelikle, modelimiz bir süper sınıf grubu içindeki sınıflar arasındaki benzerlikleri belirleyebilir ve bilinmeyen nesnelerle karsılaştığında benzer olmayanı bulma yaklaşımıyla öğrenilen kategorilerden kolayca ayrıştırabilir. Açık dünya algılama kıyaslamaları üzerindeki deneylerimiz,

tüm görevlerde tutarlı olarak bilinmeyen nesnelerin tespitinde önemli geliştirmeler yakaladığımızı göstermektedir. En önemlisi, özellik alanının süper sınıflarla daha iyi bölümlenmesi sayesinde, bilinen kategorilerin performansından ödün vermeden bunu başarabiliyoruz.

ACKNOWLEDGMENTS

This thesis is the output of two years of hard work, dedication, and collaboration. While it is a testament to my efforts, it is also a reflection of the invaluable support and guidance I have received from the people in my life.

First and foremost, I am deeply grateful to my dear family. My beloved parents, Dilek and Oktay, their unwavering support has been a guiding light throughout my journey. Thank you for always trusting in me. Öykü, having an older sister is like having a compass that has experienced things just before me to guide me through life; you have made everything easier for me. Azra, my little sister, you know how special you are to me. Thank you for the endless supply of dog photos, memes, and late-night talks over the phone.

I extend my heartfelt gratitude to my advisors. Fatma Güney, thank you for not only being an amazing advisor but also for making us an important part of your life and supporting us in any way you can. João F. Henriques, your expertise and incredible feedback have been invaluable in shaping my research journey. I would also like to thank Assoc. Prof. Emre Akbaş and Assist. Prof. Ayşegül Dündar for participating in my thesis defense committee and reviewing my research.

The group who made this experience significantly better, AVG. All the past and current members, I am so happy to have met you. I will not forget our coffee breaks, daily chats, and inside jokes. ENG123 has been a second home throughout this journey, and it wouldn't be the same without you.

Finally, my closest, Batuhan. I can't thank you enough for always being there through the highs and lows. Thank you for listening to my needs and reminding me of what is important. I hope to be a similar source of support for you. Thank you for bringing Cesur into my life; countless pets and smiles have brought light and joy through the difficult times. To many more routines together.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

OWOD	Open-World Object Detection
DETR	Detection Transformer
RPN	Region Proposal Network
O1O	Odd-One-Out

Chapter 1

INTRODUCTION

1.1 Overview

The development of perception models is crucial for accurate modeling of the world and understanding of the environment, spanning from autonomous systems to safety applications. Among perception tasks, object detection plays a central role in identifying and locating objects within images or video frames. Over time, significant advancements have driven the field forward, enhanced by the continuous evolution of deep learning architectures, alongside the increased availability of datasets featuring extensive numbers of samples, diverse labels and domains. Yet, traditional approaches often struggle in dynamic environments, particularly in open-world scenarios. Here, the system encounters objects from both familiar and unfamiliar classes, posing a challenge for conventional methods to adapt effectively. Recognizing novelty is critical for various applications, ranging from autonomous driving to anomaly detection in medical applications or monitoring systems. Therefore, there's a need to shift towards more adaptable and resilient methodologies capable of navigating the uncertainties and complexities of real-world settings.

Open-world object detection (OWOD) methods offer one approach for developing more robust perception systems. In the open-world scenario, methods are required to recognize or detect objects from both known and unknown classes. Different from the open-set approach [Scheirer et al., 2012, Dhamija et al., 2020], open-world methods can be considered as episodic learners since they are required to incrementally learn about previously unknown classes in consecutive task settings [Bendale and Boult, 2015, Joseph et al., 2021]. The key obstacle in the open-world scenario is the limited access to a static dataset containing only a fixed set of classes to learn from. Without

any effort to recognize unknown classes, the standard object detection approach partitions the representation space between the known classes, also known as closed-set training. Since unseen classes are not represented in the learned space, this approach generalizes poorly to the open-world setting. Models trained in a closed-

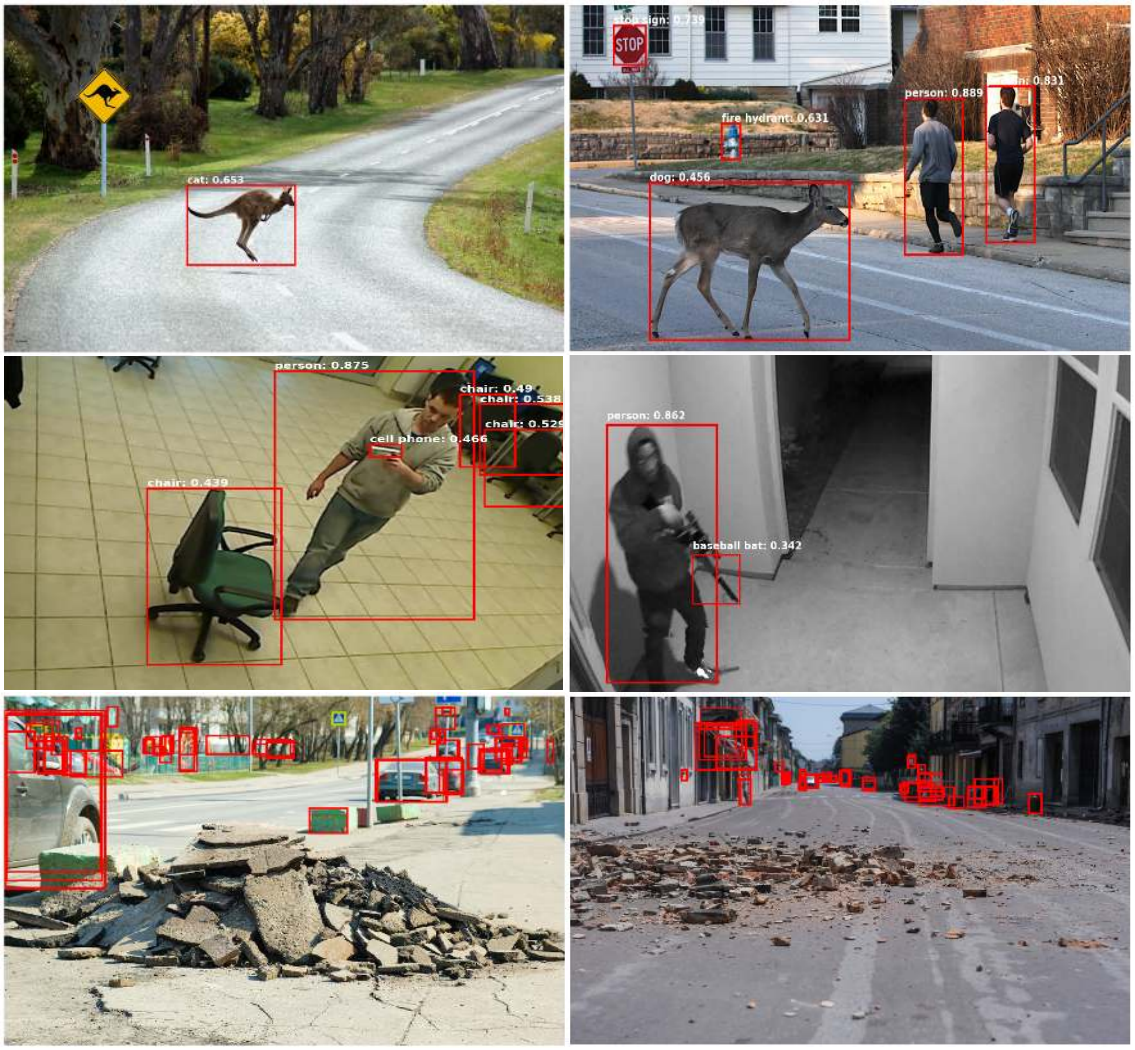

Figure 1.1: **Failing to Identify Unknown Objects.** Predictions of a Deformable DETR model [Zhu et al., 2021] trained on the full COCO dataset [Lin et al., 2014]. Novel classes of animals outside the set of COCO labels, such as kangaroo and deer, are located but misclassified as cat and dog. In the second row, the guns are misclassified as a cell phone and baseball bat, and partially mislocated. For the first two rows, irrelevant boxes with lower confidence scores are filtered out for visualization purposes. For the final row, we visualize all predicted boxes, without the labels for clarity, to show that accidental debris on the road is completely ignored by the model.

set manner either fail to locate the unknown concepts in the image altogether or incorrectly label them as one of the known objects.

We illustrate a few failure cases in Fig. 1.1 by running inference on a Deformable DETR [Zhu et al., 2021] model trained on the full COCO dataset [Lin et al., 2014]. The first row showcases rare animals encountered in traffic scenarios. These animal classes are not included in the annotation set of COCO. The separation of the feature space leads the model to make incorrect predictions, with considerably high scores. Still, the model can detect that there is an animal on the road, which can induce a break and prevent an accident if employed on a self-driving vehicle. On the other hand, the second row highlights the importance of unknown detection in security systems. In both images, the people are holding guns in their hands. However, the model confuses them with a cell phone and a baseball bat, which are typically not considered to be dangerous objects. In an automated system, these detections might not trigger an alarm and could be overlooked until it's too late. Lastly, the final row demonstrates the failure to locate examples. There is accidental debris on the road, which can cause damage to the car if passed over. Yet, even when visualizing the entire set of predictions, including those with low confidence, we observe that none of them focus on the hazardous area. Overall, these examples support the need for unknown-aware perception methodologies that can adapt to real-world scenarios.

1.2 Motivation

A common strategy to transition into an open-world setting is to learn the known classes very well and call everything else unknown. This strategy works well for dense estimation tasks like semantic segmentation, where every pixel is labeled. However, in object detection, the model needs to locate the objects precisely and classify the regions inside predicted bounding boxes. For example, in transformer-based object detection architectures like DETR [Carion et al., 2020], queries attend to interesting regions in a scene in the encoder and decoder modules. Then, the precise localization of objects and classification of the regions are done in separate branches independently. While the strategy of labeling everything else as unknown can still

Figure 1.2: **Noisy Pseudo-labels.** We illustrate top-10 proposals obtained by our RPN using different sources of input, all with confidence scores greater than 0.5. RGB might focus on irrelevant small regions (**top**) and is biased towards person instances (**middle**) due to its high frequency in the dataset. Predicting depth is more challenging for indoor scenes, hence, can result in inaccurate boxes (**bottom**). Surface normals are directional based, therefore, might focus on subparts with different structures (**top, middle**). Overall, geometric cues have less semantic bias.

be applied, it is limited by the model's capability to distinguish unknown objects from the background and place accurate bounding boxes around them. Therefore, unknown object detection methods typically use additional information to learn to localize all objects.

In object detection, both the localization and classification are supervised using ground truth annotations for known classes. In the case of unknown classes, however, there is no ground truth data available for training, mimicking the novelty scenarios of a real-world setting. The lack of data has led to the mining of unknown objects from highly activated areas in the feature maps of intermediate layers [Gupta et al.,

2022] or by using proposals of object localization networks [Fang et al., 2023]. These pseudo-labeling strategies generate approximate ground truth for unknown objects, enabling the model to search for objects beyond the known classes. However, pseudo-labels may not necessarily correspond to object regions (Fig. 1.2), and directly using noisy targets without any safeguarding degrades the performance of known classes. As we show in our experiments, encouraging the model to detect unknown objects often comes with the trade-off of compromising the performance of known object detection. Therefore, it is essential to propose a strategy to preserve the knowledge about known classes.

Alternatively, to avoid the negative consequences of pseudo-labels, certain existing methods rather approximate an objectness function that is learned solely from known concepts yet capable of generalizing across all objects. This can be achieved in the form of energy score-based prototypes [Joseph et al., 2021] foreground-background classification [Gupta et al., 2022], IoU-based localization scores [Wu et al., 2022a], or Gaussian distribution fitting [Zohar et al., 2023]. While these methods prove to be more effective than their closed-set counterparts, their performance in detecting unknown objects typically lags behind that of methods employing additional supervision via pseudo-labels. The reason behind this difference can be explained from an optimization point of view. Using pseudo-labels acts as an explicit source of supervision by penalizing the model when it fails to detect the candidate object region. On the other hand, approximating an objectness function is an implicit way of finding a generic representation capturing all objects. However, due to the nature of the optimization process, when there are no feedback signals from the potential unknowns, the model will inevitably be biased toward known objects.

1.3 Approach

In this work, we resort to pseudo-labels for their superior performance in unknown detection. We experiment with different sources of pseudo-labels, both appearance-based and geometric-based. For the appearance-based approach, we use RGB images. By geometric cues, we refer to visual information derived from the physical

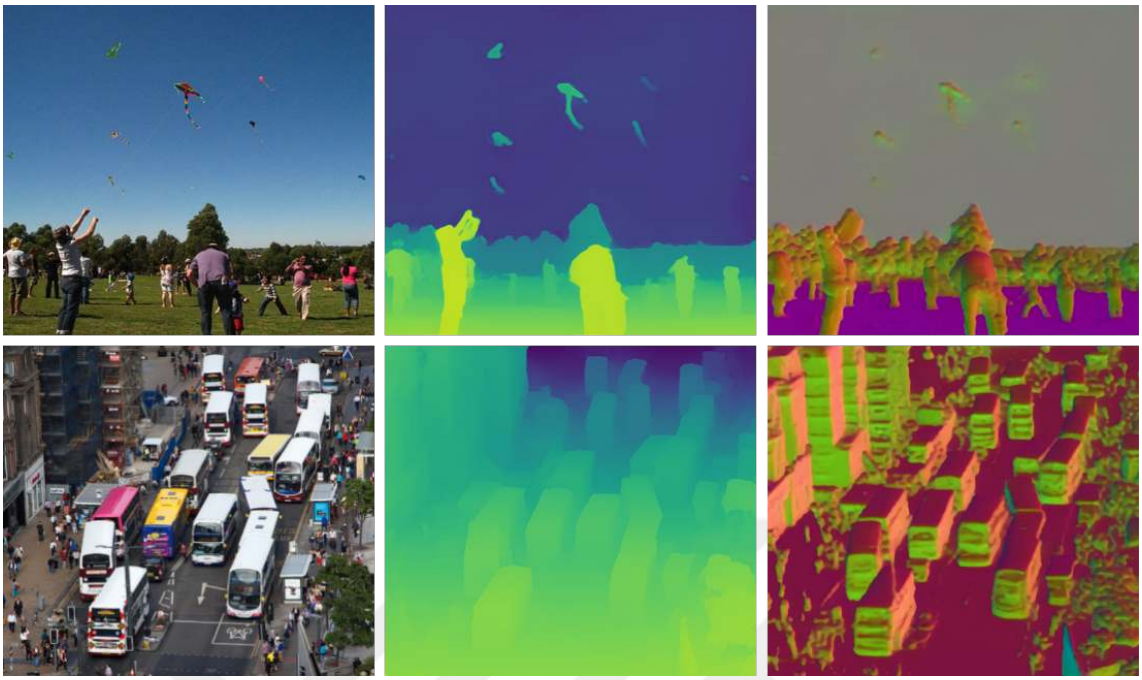

Figure 1.3: **Visualization of Predicted Depth and Surface Normal Maps.** This figure illustrates the depth (**middle**) and surface normal (**right**) maps generated by the Omnidata [Eftekhar et al., 2021] prediction models, utilized in [Huang et al., 2023]. Geometric representations provide a semantic-free point of view despite the complexity of the scenes.

characteristics of objects, such as their shape and relative spatial position. These cues offer generic information about objects that is more disentangled from semantics compared to textures or patterns. Compared to RGB, they can be more robust to variations in lighting and color, representing both knowns and unknowns more uniformly, thus promising to be more effective for localization supervision, as demonstrated in Fig. 1.3. Recent work in class-agnostic object detection [Huang et al., 2023] reports improved recall by incorporating geometric cues such as predicted depth and surface normal maps into region proposal networks (RPN) [Ren et al., 2015]. However, the potential of geometric cues remains unexplored in open-world object detection (OWOD) where the goal is not only to improve unknown performance but also to maintain known performance while learning novel categories incrementally. As a first step in this direction, we trained a previous state-of-the-art OWOD model, PROB [Zohar et al., 2023], with geometric pseudo-labels.

PROB [Zohar et al., 2023] learns a class-agnostic distribution to represent generic objectness. This essentially pushes all matched object queries towards an objectness center, forcing them to be similar to each other (Fig. 1.4). In the existence of additional pseudo-labels, this approach degrades known performance due to noisy pseudo-labels that may not correspond to an object (Fig. 1.2).

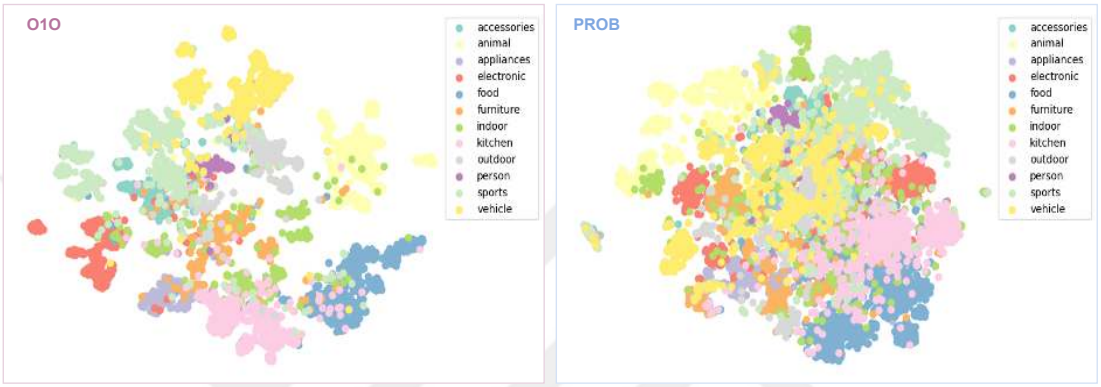

Figure 1.4: **Superclass Prior vs. Single Objectness Distribution.** This figure compares the projections of object queries trained with superclass prior in our approach O1O (**left**) vs. fitting a single objectness distribution to matched queries as in PROB [Zohar et al., 2023] (**right**) for Task 4 (all classes) on S-OWOD. O1O shapes the representation space by encouraging queries of similar classes to group together. This grouping in the representation space allows us to identify odd-one-out queries as unknown objects.

Learning a single distribution to represent any object is a strong approximation to capture the diverse characteristics of object groups and is expected to perform poorly as the space of known concepts gets larger (Fig. 1.4). The other extreme is learning class prototypes by fitting a distribution to each known class [Du et al., 2022], which is challenging because of the varying frequencies and specifics of each class and creates more confusion for the model. Studies in human perception suggest that humans use common attributes, such as material, shape, color, and functional properties, to recognize and describe objects [Hebart et al., 2020]. When a novel object is encountered, humans leverage the existing attribute groups in their knowledge, finding similarities to known concepts or recognizing unfamiliarity otherwise. Inspired by this idea, we propose a middle ground between the two extremes by

grouping similar object classes into superclasses. By partitioning the query embedding space into superclasses, we recalibrate the confidence scores for known classes using a conditional probability formulation. Our experiments demonstrate that recalibrated classification scores prevent degradation in the performance of known classes when using pseudo-labels for unknown objects and achieve the best balance between known and unknown performance.

Our contributions can be summarized as follows:

- We explore the potential of geometric cues for obtaining pseudo-labels in OWOD for the first time. We experiment with different sources of pseudo-labels and compare their effectiveness.

- We find that geometric pseudo-labels improve the unknown performance significantly, exceeding published state-of-the-art in OWOD across all tasks.

- We identify the problem of performance loss in known classes caused by noisy and inaccurate pseudo-labels and propose a solution as shaping query representations via superclass supervision.

- We show that our solution with superclasses helps recover known performance to the level of closed-set object detectors, achieving the best balance between known and unknown performances.

- We evaluate our overall pipeline, named O1O, on incremental learning tasks. As the model is trained on new tasks, the superclass supervision helps to achieve state-of-the-art performance for both previously known and newly introduced classes, showcasing its potential for shaping query representations.

- During the scope of this work, I have also worked in Out-of-Distribution (OoD) Segmentation in parallel. We proposed an unknown region segmentation method, RbA [Nayal et al., 2023], by discovering the underlying properties of mask classification models [Cheng et al., 2022].

Chapter 2

RELATED WORK

2.1 Closed-set Object Detection

The field of object detection has undergone significant evolution over approximately two decades, marked by transitions from traditional two-stage to more streamlined one-stage approaches and, most recently, the emergence of transformer-based models. Since the initial work on object detection in the early 2000s, traditional methods such as Histogram of Oriented Gradients (HOG) Features [Dalal and Triggs, 2005] and Deformable Part Models (DPM) [Felzenszwalb et al., 2009] laid the groundwork for subsequent advancements.

With the rise of deep neural networks, the early efforts were followed by two-stage detectors like the R-CNN [Girshick et al., 2013] family, including Fast R-CNN [Girshick, 2015] and Faster R-CNN [Ren et al., 2015], which dominated the field for several years. These methods typically involve a region proposal network (RPN) to generate candidate object regions, followed by a classification and bounding box regression stage. However, the need for separate proposal generation and classification stages led to inefficiencies in speed and memory usage. In response, one-stage detectors like YOLO [Redmon et al., 2016] and SSD [Liu et al., 2016] were introduced, which directly predict object bounding boxes and class probabilities from fixed grid cells or anchor boxes. Although these models achieved real-time inference speeds, they often struggled with accuracy, especially for small objects.

More recently, transformer-based architectures, inspired by their success in natural language processing, have been adapted to object detection tasks. Models like DETR [Zhu et al., 2021] and Deformable DETR [Zhu et al., 2021] employ transformer networks and an encoder-decoder pipeline design to directly predict object positions and classes without the need for anchor boxes or explicit region proposals.

Although the initial DETR architecture is quite inefficient and needs 500 epochs for convergence, which is up to 20 times slower than Faster RCNN, the follow-up work on DETR-based methodologies offers profound ways to improve both accuracy and efficiency. More information regarding DETR variants is shared in Section 3.1. This paradigm shift towards transformer-based object detection architectures is consistent in sub-fields as it is also adapted in the open-world object detection literature.

2.2 Open-world Object Detection

The problem of detecting unknown objects has been tackled early on as an open-set recognition problem where the goal is to identify concepts that are semantically disjoint from the labels in the training set [Miller et al., 2017, Hall et al., 2018, Miller et al., 2018, Dhamija et al., 2020, Miller et al., 2021, Han et al., 2022, Zheng et al., 2022]. Simultaneously, other methods, pioneered by [Joseph et al., 2021], extended this formulation to open-world object detection (OWOD). In OWOD, the aim is to detect unknown objects and incrementally learn novel categories without compromising the performance of previously known classes. This setting aligns more closely with the real-life requirements in ever-changing environments.

Existing OWOD methods follow different approaches to separate objects from the background, whether known or unknown. Most methods extract pseudo-labels, i.e. candidate bounding boxes for unknown objects as auxiliary supervision [Gupta et al., 2022, Ma et al., 2023], while other efforts include predicting an objectness score [Joseph et al., 2021, Wu et al., 2022a, Zohar et al., 2023]. Generally, OWOD methods can be categorized based on their object detection architecture.

2.2.1 RPN-based Approaches

Earlier approaches build on two-stage architectures in the R-CNN family [Girshick et al., 2013, Girshick, 2015, Ren et al., 2015]. A common strategy involves generating pseudo-labels, which potentially correspond to unknown objects, using methods like Selective Search [Uijlings et al., 2013] or utilizing the region proposal network

(RPN). Specifically, ORE [Joseph et al., 2021] utilize the region proposal network (RPN) for auto-labeling potential unknowns and uses an energy-based classification head for identifying knowns and unknowns. Building on ORE [Joseph et al., 2021], UC-OWOD [Wu et al., 2022b] also use pseudo-labels generated by the RPN but modify the classification head to account for multiple potential unknown classes via similarity-based clustering. On the other hand, OCPL [Yu et al., 2022] learns class-specific prototypes to reduce the overlap between seen and unseen class distributions. 2B-OCD [Wu et al., 2022a] propose a two-branch framework, where one branch classifies the RPN proposal to one of the known classes, and the other branch does localization and learns objectness by predicting the IoU score of a proposal as an indicator of class-agnostic localization quality. RandBox [Wang et al., 2023] randomly generate bounding box proposals at each iteration to alleviate the bias of known classes and propose a matching score to select the proposals that are most likely foreground unknown objects. Following an autoencoding paradigm, MEPU [Fang et al., 2023] learn separate distributions for foreground and background based on reconstruction errors and utilize the learned distributions for selecting pseudo-unknowns.

2.2.2 DETR-based Approaches

More recent work builds on the transformer-based architecture DETR [Zhu et al., 2021] and its variants. Initial work OW-DETR [Gupta et al., 2022] aim to capture unknowns with a scoring function based on the attention activation of intermediate layers. They deploy two classification heads: one for separating knowns from unknowns and another for distinguishing foreground from background. Follow-up work CAT [Ma et al., 2023] combine Selective Search [Uijlings et al., 2013] and attention activations as in [Gupta et al., 2022] for generating pseudo-labels. They decouple localization and classification in a cascade fashion to minimize the bias of known classes. PROB [Zohar et al., 2023] learn a distribution of objectness by jointly optimizing the parameters of a multivariate Gaussian on the matched object queries of DETR. Similarly, USD [He et al., 2023] also learn a distribution of ob-

jectness but based on the early decoder layers rather than the final decoder layer, which is generally used for class label and bounding box coordinate prediction. Additionally, they generate pseudo-labels using the Segment Anything Model (SAM) [Kirillov et al., 2023], which significantly contributes to their performance on unknown objects. While it is intriguing to utilize foundational models for OWOD, the definition of unknown becomes blurry for SAM, which has been trained on a large collection of internet images. Our method also builds on the variations of DETR. Differently, we explore the potential of geometric cues to detect unknowns, and superclass supervision for known representations.

2.3 Class-agnostic Object Detection

A common property of the proposed methods for OWOD is seeking an objectness scoring function that would allow mining auxiliary bounding boxes to be used as pseudo-unknowns for supervision. The demand for such a function stems from the need to separate boxes of true unknowns from those of irrelevant background. The problem of separating foreground and background has also been studied in the class-agnostic object detection literature as an approximation of objectness. [Maaz et al., 2021a, Zhao et al., 2022, Kim et al., 2021, Saito et al., 2022, Maaz et al., 2021b, Huang et al., 2023]. Some methods utilize the rich semantics learned from natural language to ease the localization of objects in images [Maaz et al., 2021a, Zhao et al., 2022]. LDET [Saito et al., 2022] augment images by synthetically pasting foreground objects to alleviate the bias caused by unlabelled foreground objects in existing datasets. OLN [Kim et al., 2021] estimates the objectness of proposals with localization quality supervised by ground truth labels. GOOD [Huang et al., 2023] argues that geometric cues carry more relevant information about objectness that can be generalized across categories and outperforms the methods that rely on RGB information only. In this work, we follow-up on the claim of GOOD [Huang et al., 2023], and utilize the potential of geometric cues as a source of structured localization priors.

Chapter 3

METHODOLOGY

We introduce O1O, an open-world object detection method to locate and classify objects by identifying unknowns. We first provide background on the transformer-based architectures for object detection, including the extensions that we adopt for efficiency gains in Section 3.1. We describe the pseudo-labeling methods employed by previous state-of-the-art methods in Section 3.2. Next, we introduce geometric cues to obtain pseudo-labels for unknown detection in Section 3.3. To protect the performance of known classes in the presence of additional noisy supervision from pseudo-labels, we propose to group known classes into superclasses and formulate classification as a conditional probability distribution in Section 3.4. Lastly, in Section 3.5, we explain the incremental learning strategy that we use for open-world tasks.

3.1 Background on DETR

DEtection TRansformer, or DETR in short, has been proposed to show the potential of the transformer architecture for object detection [Carion et al., 2020] and has taken over object detection with multiple extensions since then. Given a set of learnable object queries, DETR reasons about the relations between objects as well as their relation with the image features using a Transformer. Formally, DETR initializes N_{query} learned positional encoding vectors. In the transformer encoder, image features interact with each other and produce weighted attention maps. In the transformer decoder, queries first interact with each other and then with the feature maps from the encoder. This results in an embedding $\mathbf{q}_i \in \mathbb{R}^d$ for each object query $i \in N_{query}$, incorporating information about other queries and the image context. Each object query is then independently decoded into box coordinates and class

labels using a feedforward network (FFN) followed by a linear layer for output generation. During training, ground truth objects are matched to object queries via bipartite matching before computing the classification and bounding box losses. Queries unmatched with ground truth objects are responsible for predicting a special class label, typically denoted as K+1, which in closed-set methods represents the 'no-object' case, while in open-set methods, it is used for the unknown class. Note that unmatched queries do not receive any supervision for localization.

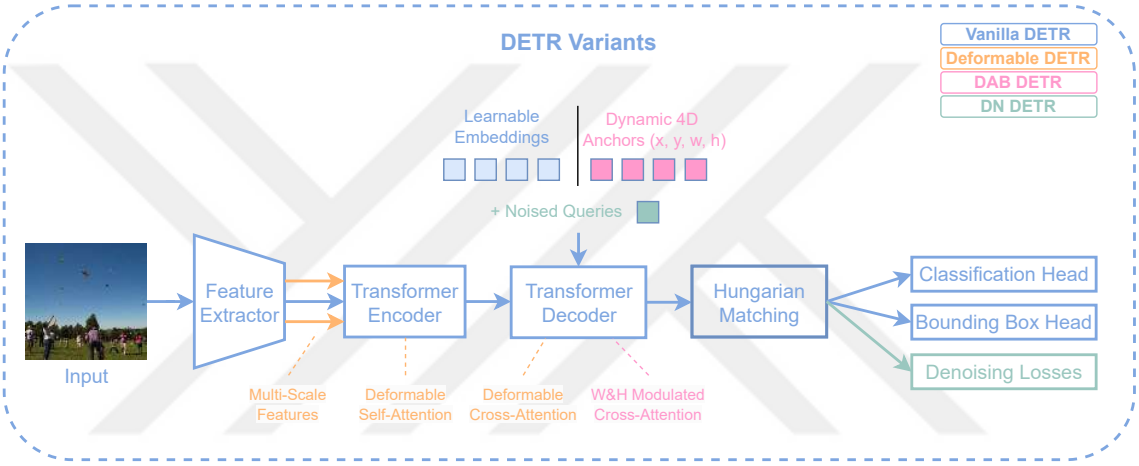

Figure 3.1: **Overview of DETR Variants.** We demonstrate the initial pipeline proposed by DETR by highlighting the improvements that are introduced by the following work. We represent the proposed extensions of each new method with their corresponding colors specified in the top-right. In the end, we use DN-DAB-Deformable DETR as our detection model, which combines the performance and convergence advantages of all variants.

Extensions: Deformable DETR [Zhu et al., 2021] extends DETR by incorporating a more efficient attention module known as deformable attention. This attention mechanism focuses on a small set of key sampling points around a 2D reference point, enhancing the model's ability to attend to relevant image features. Leveraging this efficiency, Deformable DETR integrates multi-scale image features, which are particularly beneficial for detecting small objects. Deformable DETR significantly improves the convergence capabilities of DETR. Another variant, DAB-DETR [Liu et al., 2021], improves the cross-attention operation by employing dynamically updated 4D anchor boxes consisting of a reference point (x, y) and a reference anchor

size (w, h) and modulates the cross-attention. The more recent DN-DETR [Li et al., 2022] further improves the convergence with denoising techniques during training. DN-DETR identifies bipartite matching as the bottleneck for slow convergence and mitigates it by injecting noisy ground truth boxes into the transformer decoder and then training the model to reconstruct the original boxes. This approach effectively halves the number of training epochs required to match the performance of Deformable DETR.

While most recent work in unknown object detection [Gupta et al., 2022, Zohar et al., 2023] builds on Deformable DETR [Zhu et al., 2021], we incorporate improvements from both DN-DETR [Li et al., 2022] and DAB-DETR [Liu et al., 2021], as shown in Fig. 3.1. Our primary objective is to benefit from improved convergence properties for fast experimentation while still achieving top performance in terms of known classes. Note that these modifications to the architecture improve efficiency but do not directly contribute to known or unknown performance. For fairness, we report the performance of the SOTA unknown object detection method, PROB [Zohar et al., 2023] using the same architecture (See Section 5.1).

3.2 Background on Pseudo-labeling Methods

In DETR-based methods for unknown object detection, the class index $K + 1$ represents the unknown class in addition to the K known classes. This additional class index is used to supervise unmatched queries to predict the unknown label. Both background regions and potential unknowns undergo similar treatment. The primary challenge for separating unknown objects from the background regions is the lack of information regarding the spatial locations of unknowns within a scene. As a result, unmatched queries only receive semantic supervision, which limits the model's ability to locate objects beyond the known set. Previous methods often employ pseudo-labeling strategies (Fig. 3.2) to obtain an auxiliary signal for training to localize all objects instead of only known ones.

RPN Proposals: Initial methods based on the Faster-RCNN backbone, including ORE [Joseph et al., 2021], UC-OWOD [Wu et al., 2022b], and 2B-OCD

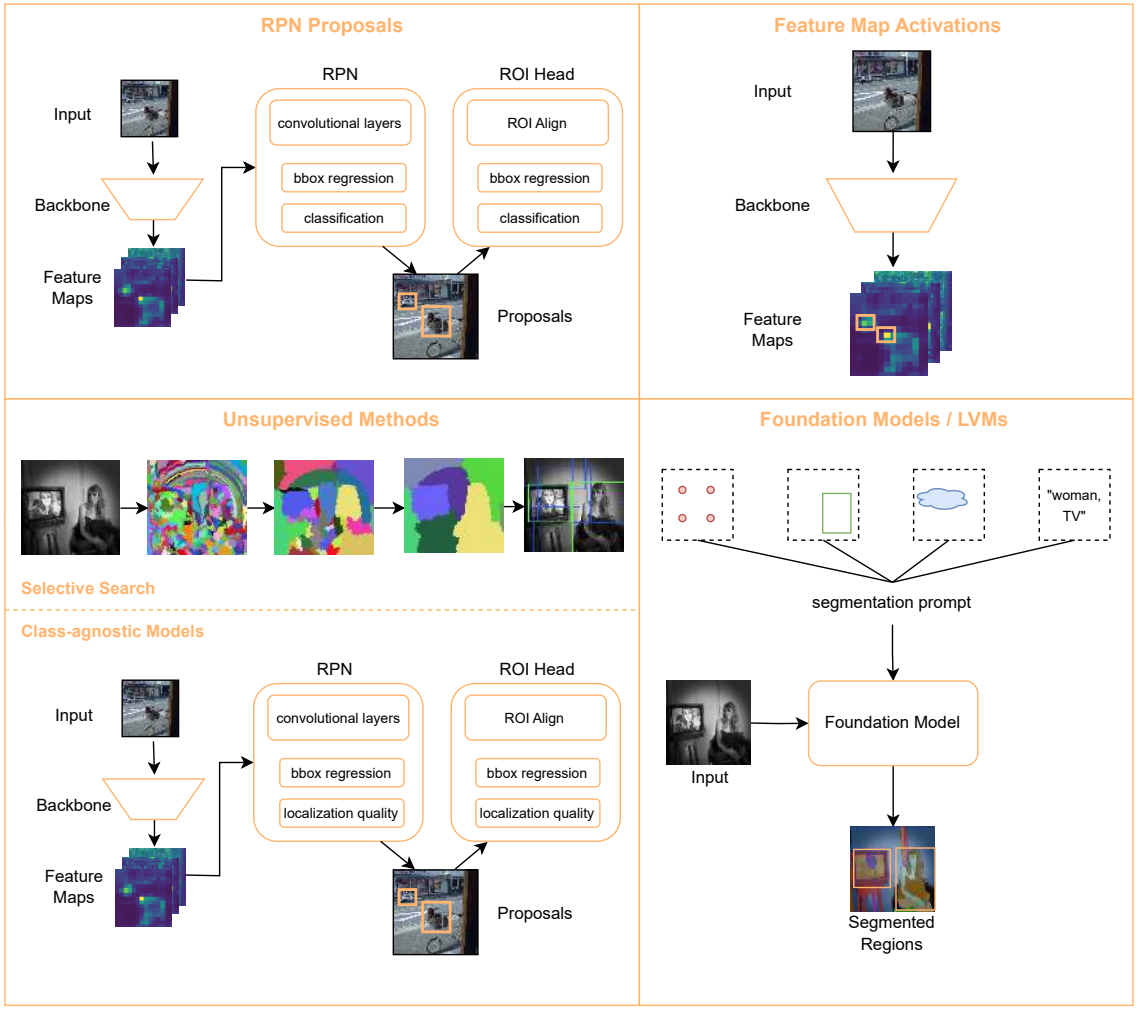

Figure 3.2: **Overview of Pseudo-labeling Methods.** High-level explanations of different pseudo-labeling approaches. The first row represents easy plug-ins. RPN proposals can either be obtained from the existing RPN of Faster-RCNN architecture, or with a separately trained one in the case of DETR-based methods. Feature map activations can easily be extracted from the feature extraction backbone. Approaches in the second row require the use of an additional network or a similarity-based algorithm. Class-agnostic models might be pre-trained or trained from scratch to obey the rules of specific benchmarks. Moreover, multiple strategies can be combined to acquire a more extensive set of pseudo-labels.

[Wu et al., 2022a], utilize the proposals generated by the already existing RPN. The Region Proposal Network (RPN) is a component of the Faster-RCNN network architecture responsible for generating region proposals, which are candidate bounding boxes likely to contain objects. After filtering out proposals overlapping with known ground truth, the approach involves either selecting the top-k proposals or

applying a threshold based on the objectness score predicted by the RPN. While this strategy helps to supervise the model with potential unknowns, its effectiveness is limited by the bias of the RPN towards known classes, caused by the optimization process using only known labels.

Activations: Another strategy is based on the hypothesis that high activation areas in the feature maps should correspond to real objects in the scene, encompassing both known and unknown entities. OW-DETR [Gupta et al., 2022] employs this idea as an attention-driven pseudo-labeling mechanism, identifying top-k scoring regions based on the magnitude of the activation maps generated by the feature extraction backbone. This approach mirrors how humans recognize objects in a scene by focusing on areas of high visual importance. However, it's important to note that feature maps are often not smooth, and the feature backbone is fine-tuned during the object detection training phase, thereby inheriting bias towards known categories.

Unsupervised Methods: Traditional approaches like Selective Search [Uijlings et al., 2013] can also be used to extract proposals. Selective Search is a hierarchical segmentation approach from fine to coarse details and combines similar neighboring regions to generate potential object regions. More recently, class-agnostic object detectors such as OLN [Kim et al., 2021] and instance segmentation models like FreeSOLO [Wang et al., 2022] have also proven effective in identifying interesting regions in an unsupervised manner. Since they are trained independently of the object detector, their candidate boxes can be free from known-label bias and support unknown detection. However, they rely solely on RGB signals, limiting their ability to detect unknown concepts in complex regions or challenging camera viewpoints. On the other hand, RandBox [Wang et al., 2023] proposes employing a random sampler to mitigate bias in a trained proposal generator. This encourages exploration but comes with the drawback of increased noise.

Large Vision Models (LVMs): With the recent popularity of large vision models, they have been increasingly utilized in various downstream tasks in a zero-shot manner. In the case of OWOD, USD [He et al., 2023] utilize proposals from the Segment Anything model [Kirillov et al., 2023] as additional labels. USD applies

several post-processing steps, including IoU and objectness score filtering, to remove noise resulting from subpart segmentations of SAM. They significantly improve unknown recall; however, the success is primarily a result of SAM's capabilities, as demonstrated in their ablations. Moreover, the training set of a foundational model is incomparable to the previously mentioned strategies. This results in an unfair advantage and raises the question of what is truly 'unknown' for a model that has encountered extensive training data. Furthermore, they utilize the pseudo-labels directly for learning the unknown class boundaries. In contrast, we design an odd-one-out selection strategy by better shaping the query representation space through superclass supervision.

Hybrid Approach: To benefit from different advantages, some methods propose the use of a combination of pseudo-labeling strategies. Recent work CAT [Ma et al., 2023] combines the input-driven pseudo-labeling strategy Selective Search with the model-driven pseudo-labeling strategy of OW-DETR. Similarly, MEPU [Fang et al., 2023] combines unsupervised region proposal generation algorithms, such as Selective Search or FreeSolo, with a modified version of the class-agnostic Object Localization Network (OLN) to expand the set of candidate unknown objects. Hybrid approaches benefit from the advantages of different strategies; however, combining multiple networks makes it harder to control the noise, and the overall pipeline becomes over-complicated and inefficient.

3.3 Geometric Pseudo-labels for Unknown Objects

Recent work in class-agnostic object localization [Huang et al., 2023] proposes to use geometric cues as input to the RPN module in addition to the RGB input of the detection stage. While RGB-based methods suffer from over-fitting to semantic features of known classes, geometric cues can help to remove the label bias. Specifically, GOOD [Huang et al., 2023] trains a region proposal network (RPN) using geometric cues such as depth and surface normal maps as input. By focusing on object shapes, relative spatial locations (depth), and directional changes (normal vectors), GOOD can retrieve more accurate pseudo-labels compared to other class-agnostic detectors

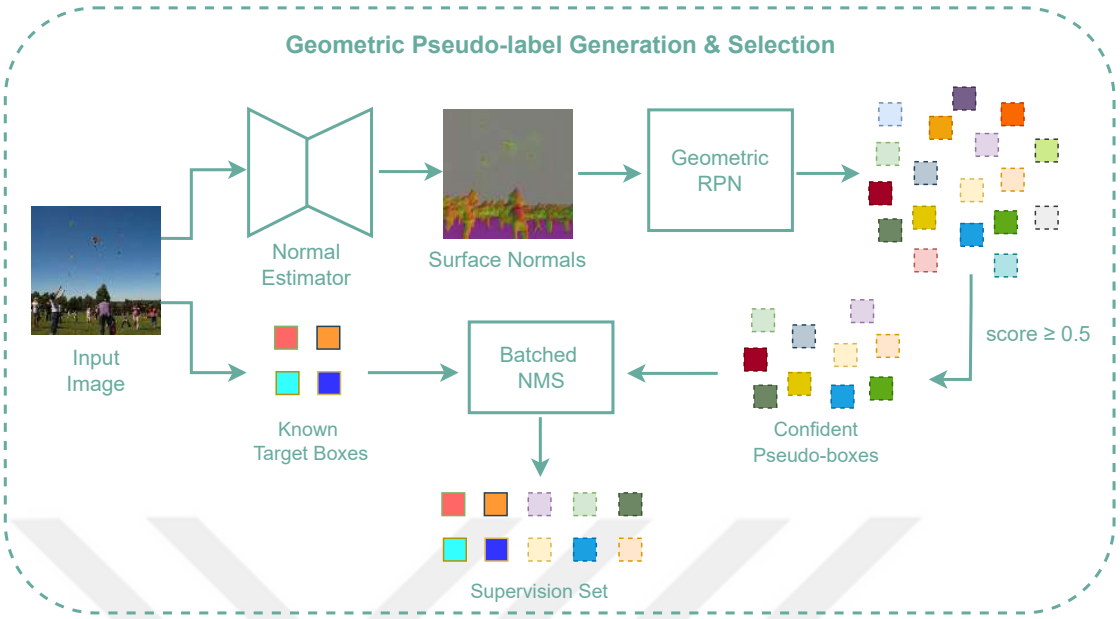

Figure 3.3: **Detailed Overview of Geometric Pseudo-label Generation and Selection.** Utilizing an off-the-shelf surface normal estimator, we train the Geometric RPN module with predicted geometric cues. From the large set of proposals, we select the high-confidence ones by thresholding. To remove duplicates between the pseudo-labels and the real known targets, we apply Non-Maximum Suppression(NMS). In the final set of supervision labels, boxes with solid borders represent the known targets, while dashed border ones represent unknown pseudo-labels.

that rely solely on RGB input.

In this work, we follow the same approach as GOOD and train an RPN module with geometric inputs to obtain a set of pseudo-labels without any categories. Fig. 3.3 highlights the detailed pipeline of our pseudo-label generation and selection, including the post-processing steps. We use surface normals as geometric cues in our model and report an in-depth ablation of this choice in Section 5.2. Specifically, the input image is first passed through a monocular surface normal estimation model, DPT-Hybrid [Ranftl et al., 2021], pre-trained on the Omnidata Starter Dataset [Eftekhar et al., 2021]. The RPN module takes the predicted surface normal map as input and generates a set of proposals consisting of bounding box coordinates and objectness confidence scores. We filter the pseudo-boxes with a fixed threshold of 0.5 to remove the regions less likely to contain objects. After obtaining the high-confidence pseudo-boxes, we pass them, along with the real known targets, to a

batched Non-Maximum Suppression (NMS) module. This module removes duplicate boxes that overlap with the known ones. The filtered set, consisting of both known target boxes and unknown pseudo-boxes, is used to supervise the detection part of our model. For fairness, we adhere closely to OWOD benchmark settings regarding data partitioning into known and unknown classes. While training the RPN, we ensure that there are no objects of unknown classes according to the benchmark setting considered. This cannot be ensured while using sources like SAM [Kirillov et al., 2023], trained on a large collection of internet images.

Previous approaches employing pseudo-labels commonly limit the number of unknowns between 1 to 5 per image to mitigate adverse effects on known performance arising from increased model confusion. In contrast, our method leverages a larger number of pseudo-annotations per image (refer to Section 5.3), a crucial factor contributing to our notable unknown recall performance. Importantly, we achieve this without any negative impact on known performance, attributed to our effective superclass supervision.

3.4 Shaping Queries Through Superclass Supervision

Following the standard DETR-based approach, our detection model takes an image as input and outputs bounding box locations and class scores corresponding to objects on the image. Let $\mathbf{c} \in \{0,1\}^{K+1}$ be a random variable representing the one-hot encoded class of a bounding box over a predefined set of K known classes and the unknown label $K+1$. Based on the embedding of query vectors $\{\mathbf{q}_i\}_{i=1}^{N_{query}}$, we re-design the classification head.

We classify regions by decomposing the probability of a class with respect to superclasses. A superclass is a group of classes that share some semantic properties. For example; cats, dogs, and birds can be grouped as *animals*, and cars, buses, and motorcycles can be grouped as *vehicles*. A study in human cognitive science [Hebart et al., 2020] reveals that our mental representation of objects is heavily based on the properties they share. According to this study, there are some categorical dimensions humans resort to while recognizing an object, such as the color, material, shape,

where it's found, or for what it can be used. More interestingly, in the odd-one-out task, a small number of these dimensions suffice to find the odd object in a triplet, suggesting a grouping of these properties to recognize similarities between objects. Inspired by this, in our approach to unknown object detection, we propose to learn common properties between known classes by grouping them into superclasses. That way, we can better judge similarities between known classes, i.e. recalibrating the confidence of what is known, and then use it to accurately identify what is unknown.

3.4.1 Superclass Prior

In DETR, the per-class classification head outputs the probability of belonging to a class $\mathbf{c}$ for each query $\mathbf{q}$ as $p(\mathbf{c}|\mathbf{q})$. In addition to that, we train a superclass head to group known classes into superclasses. Formally, the superclass head outputs a probability of belonging to a superclass $\mathbf{s}$ as $p(\mathbf{s}|\mathbf{q})$ over the set of superclasses S. For each known class, we recalibrate its confidence score by multiplying it with the probability of the superclass that it belongs to:

$$p'(\mathbf{c}|\mathbf{q}) = \sum_{\mathbf{s}\in S} p(\mathbf{c}|[\![\mathbf{c} \in \mathbf{s}]\!], \mathbf{q}) \cdot p(\mathbf{s}|\mathbf{q}) \tag{3.1}$$

where $[\![\cdot]\!]$ denotes Iverson bracket. Since known classes are fixed, we can pre-define the set of superclasses based on the known classes (see Section 4.2). In practice, this reduces conditioning on the superclass $\mathbf{s}$ in the first term to a lookup operation to find the corresponding superclass for a class. Intuitively, we shape the representation space of known classes by encouraging similar properties for queries matched to classes within a superclass.

3.4.2 Unknown Probability

For unknown probability $p(u|\mathbf{q})$, we apply the odd-one-out approach and define its probability as the complement event of being known. Following a similar approach as our other work [Nayal et al., 2023], we make a simplifying assumption and consider each class probability to be independent. Then, we can sum the recalibrated

probabilities of belonging to known classes and subtract it from 1 to obtain the probability of being unknown:

$$p(u|\mathbf{q}) = 1 - \sum_{\mathbf{c} \in K} p'(\mathbf{c}|\mathbf{q}) \tag{3.2}$$

Essentially, the decoupled conditional probability (3.1) helps the model to update its belief for a query about belonging to a class by considering the similarities with its superclass. If the model is not confident enough to label a query as known, then (3.2) becomes higher, and the query is predicted as unknown. In our experiments, we show that learning a structured prior with superclasses recovers the known performance by reducing the confusion caused by noisy supervision from pseudo-labels for unknown classes.

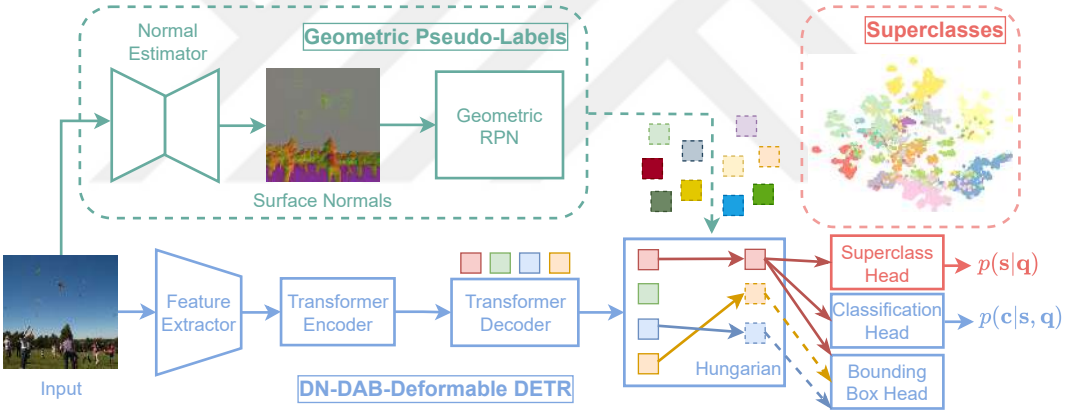

Figure 3.4: **Complete Pipeline of Proposed O1O.** Building on Deformable DETR [Zhu et al., 2021] (in blue), we first add supervision for unknowns with geometric pseudo-labels (in green). We first extract pseudo-labels (dashed boxes) from a Region Proposal Network (RPN) trained on surface normal maps (in green). This allows us to localize unknown objects based on geometric cues in a class-agnostic manner. However, noisy pseudo-labels can adversely affect the known performance of the model. To address this challenge, we propose grouping queries into superclasses using a superclass head (in red). By integrating the learned prior from superclasses into the scoring function, we achieve an optimal balance between known and unknown performance.

3.4.3 Training

The complete pipeline of O1O is illustrated in Fig. 3.4. We initially train the RPN solely on the known classes of the current task and store the generated proposals. This approach eliminates the need for redundant forward passes through both the surface normal estimator and the geometric RPN module during detection. Then, using the combined set of known and pseudo-unknown targets, we train our model end-to-end on a weighted combination of localization, per-class classification, and superclass losses:

$$\mathcal{L} = \underbrace{\mathcal{L}_{\text{bbox}} + \mathcal{L}_{\text{giou}}}_{\text{localization}} + \underbrace{\mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{sup}}}_{\text{classification}} \tag{3.3}$$

We use the same loss function used in DETR-variants and only add the classification loss $\mathcal{L}_{\text{sup}}$ (shown in blue) to train the superclass head. We use focal loss [Lin et al., 2017] for superclass classification to address the imbalance issue in the distribution of superclasses. We apply the classification losses only on queries matched to known targets. In other words, we only apply the localization losses for the queries matched to pseudo-labels. For the classification of unknowns, we use the knowledge of the model on known classes to reason about what is unknown.

Since we build on extended Deformable DETR with denoising [Li et al., 2022], following their formulation, we use a denoising loss for each loss component, including the superclass loss. Specifically, for the noised bounding box queries, denoising bounding box loss learns to reconstruct the original coordinates while denoising classification losses predict the original target labels.

3.4.4 Inference

In inference, we use the decomposition in (3.1) to obtain classification confidence scores of known classes. Since one class only belongs to a specific superclass, and this matching is given in the label set, this is not privileged information but rather a prior we set. We only use the currently known class label set to access superclass information, and pseudo-unknowns are not considered to belong to any superclasses. For the unknown score, we use (3.2) and decide a threshold on a subset of the training

set, such that the 95% of known instances would be classified correctly. For any value below this confidence threshold, we set the unknown probability to 0. This helps us control the trade-off on the known performance that comes with being greedy towards unknowns.

3.5 Incremental Learning

Apart from unknown detection, OWOD formulation requires incrementally learning novel classes while still maintaining the performance on previously learned concepts. However, training a model continually on multiple tasks typically results in catastrophic forgetting [McCloskey and Cohen, 1989]. To prevent such a phenomenon, the common practice is to store a small set of exemplars which is then used to fine-tune the model on the previous tasks. Most approaches in OWOD select the exemplar set randomly. On the other hand, an active selection strategy can be applied for a more effective replay. For example, PROB [Zohar et al., 2023] uses their objectness approximation to actively select images containing high and low-scoring instances, representing easy and hard instances.

We implement a similar active exemplar selection strategy using our class conditional probability (3.1). Specifically, we calculate the class-wise conditional probabilities for all queries on the training set of the previous task. Then, for each class, we select the images containing the highest 25 and the lowest 25 scoring instances and save them as exemplars to be used in the fine-tuning stages. Instances with the highest scores represent easily classifiable cases and serve as highly representative examples of their classes. Conversely, those with the lowest scores capture challenging scenarios where repeated exposure can potentially help the model improve its predictions. Together, they provide enough balanced chances for the model to remember what to do under different circumstances without causing any overfitting problems to specific types of examples. If the number of selected images turns out to be higher than the previous work, we randomly drop out the extra number of images to be comparable to the previous work.

In practice, the incremental learning procedure can be summarized as follows: To

adapt to a new task, we utilize our model pre-trained on the previous task's dataset, execute the exemplar selection strategy, and store the selected set of images. We then proceed to train our model on the current task's dataset and find an updated set of exemplars. Finally, we employ the exemplar fine-tuning strategy using the combined set of previous and current exemplars to regain prior knowledge while maintaining the newly attained category information.

Chapter 4

EXPERIMENTS

In this section, we present the experimental details and main results. We begin by explaining the OWOD benchmarks in Section 4.1 and describing our superclass separation approach in Section 4.2. Section 4.3 outlines the evaluation metrics employed, while Section 4.4 provides an overview of our architectural and training specifics. Finally, in Section 4.5, we conduct a comprehensive quantitative and qualitative comparison of our method O1O with previous state-of-the-art approaches.

4.1 Benchmarks

We report results on two existing OWOD benchmarks. M-OWOD, proposed by [Joseph et al., 2021], is composed of both MS-COCO [Lin et al., 2014] and PASCAL VOC [Everingham et al., 2010] images. 80 categories of the MS-COCO dataset are split into four non-overlapping tasks $\{T_1, T_2, T_3, T_4\}$ with 20 classes being introduced at each. In each task, T_t, only the annotations of classes that have been introduced until T_t are used for training, and the rest are removed. The test set contains instances from all 80 classes, and the known vs. unknown performance is calculated according to the known-unknown split. S-OWOD, proposed by [Gupta et al., 2022], only consists of MS-COCO images. Classes are split according to the supercategories they belong to. For example, Task 1 includes all animal classes, while Task 3 introduces all sports classes. Denoting the differences, these benchmarks are also known as Mixed-superclass and Separated-superclass, respectively. For more details, please refer to [Joseph et al., 2021] and [Gupta et al., 2022].

Benchmark	Task	Superclasses Used During Training
S-OWOD	1	animal, person, vehicle
	2	accessory, appliance, furniture, outdoor
	3	food, sports
	4	electronic, indoor, kitchen
M-OWOD	1	animal, electronic, furniture, kitchen, person, vehicle
	2	accessory, animal, appliance, outdoor, vehicle
	3	food, sports
	4	electronic, furniture, indoor, kitchen

Table 4.1: **Superclass Separation Across Tasks.** This table displays the superclasses learned at each task of both OWOD benchmarks. In S-OWOD tasks, where superclasses are separated, all superclasses are learned from scratch and to completion upon introduction. However, in M-OWOD tasks with mixed superclasses, some superclasses (underlined) are initially introduced partially and then resumed when the remaining classes are learned in subsequent tasks.

4.2 Superclass Information

Both OWOD benchmarks are based on the set of MS-COCO classes, which follow a superclass hierarchy specified in the official COCO annotation files. Following the same grouping and split criteria as the OWOD benchmarks, the superclasses introduced at each task for both S-OWOD and M-OWOD benchmarks can be found in Table 4.1. Since S-OWOD tasks are designed to have perfect superclass separation, each superclass introduced in a task is learned from scratch and to completion. On the other hand, on M-OWOD, a task might have some of the classes belonging to a certain superclass in the current training set. For example, in M-OWOD Task 1, only *bird*, *cat*, *cow*, *dog*, *horse*, and *sheep* classes are learned from the animal superclass; while *elephant*, *bear*, *zebra*, and *giraffe* are learned in Task 2. Still, if any class of a superclass is present, we introduce it at the current task. Unseen classes are still considered unknown, following the benchmark rules. As the unseen classes become available in the continual learning procedure, we continue training the specific weights of partially introduced superclasses. We underline partially learned super-

classes in Table 4.1. Although the partially learned superclass representations hurt the odd-one-out scoring in theory, our experimental results show that our method can still generalize to such a setting.

4.3 Evaluation Metrics

We follow the common practice in OWOD literature by reporting the mean average precision (mAP) for known performance. mAP, calculated from the area under the precision-recall curve, serves as the overall accuracy of the detection model. Since not all objects are annotated in any given dataset, unknown precision is ill-defined. A model might detect an object in a location where a real object exists, but if it does not belong to the predefined categories in the dataset, it might have been excluded from the ground truth annotations. For example, although many animal classes are present in COCO annotations, giraffe is not one of them. Given an image of a giraffe, we want our model to detect it as unknown. Formally, this would count as a false positive. However, we do not want to penalize the model for detecting such objects. Therefore, we use recall (U-Recall) as the main metric for evaluating the ability to retrieve unknown objects. U-Recall focuses on the model's capability to detect unknown objects without penalizing it for false positives.

4.4 Implementation Details

Our pipeline comprises two main steps: first, we train the Region Proposal Network (RPN) to extract pseudo-labels, and then we train our open-world detection model.

4.4.1 Pseudo-labels Extraction Step

We train the RPN using surface normal maps estimated by the DPT-Hybrid model [Ranftl et al., 2021] from the Omnidata repository [Eftekhar et al., 2021], following a similar approach as GOOD [Huang et al., 2023]. The RPN is trained from scratch for each task, adhering to the benchmark rules to exclude any unknown instances from training data. We select the pseudo-labels by thresholding the RPN's confidence score with 0.5 and merging them with real targets by applying Non-Maximum

Suppression with an IoU threshold of 0.5. To ensure that no known box is excluded due to an overlap with a pseudo-box, we assign a confidence score of 1.0 for known target boxes and use the RPN's objectness score for pseudo-labels.

4.4.2 OWOD Step

We build our method on the extended version of Deformable DETR [Zhu et al., 2021] with denoising and 4D anchor boxes [Liu et al., 2021, Li et al., 2022]. We use the same backbone as the prior work, i.e. ResNet-50 FPN [He et al., 2015] pre-trained on ImageNet [Deng et al., 2009] in a self-supervised manner [Caron et al., 2021]. We set the number of queries to 100 and the query embedding dimension to 256.

4.4.3 Training

We decompose our loss function into localization and classification components. Localization losses are applied to queries matched to both knowns and pseudo-unknowns, while classification losses are only applied to queries matched to known targets. The localization loss is a weighted combination of L1 loss and generalized Intersection over Union (gIoU) loss, with coefficients of 5 and 2, respectively. For both per-class and superclass classification, we utilize focal loss with coefficients set to 2. We follow the default hyperparameters of DN-DAB-Deformable DETR for the localization and classification denoising tasks. We add a denoising task for superclass classification, using the same hyperparameters as the per-class denoising loss, by randomly flipping the ground truth superclass labels for noised queries to facilitate model learning and stability.

Our model is trained on 4 GPUs with a batch size of 6 and a starting learning rate of 2e-4. To accommodate the rapid convergence of DETR extensions, we decrease the learning rate by a factor of 10 at epoch 18. We train for 30 epochs on M-OWOD Task 1 and 20 epochs on S-OWOD Task 1. The larger dataset size of S-OWOD Task 1 enables our model to converge earlier compared to M-OWOD Task 1. For incremental tasks, we train for 10 epochs with a learning rate of 2e-5. Fine-tuning from exemplar replay is performed for 20 epochs, with a learning rate schedule

of 2e-4 for the first half and 2e-5 for the second half. Overall, our architectural improvements cut down training time by approximately half compared to previous DETR-based approaches.

4.5 Open-world Object Detection Performance

In Table 4.2, we present O1O's performance in OWOD, comparing it to previous approaches across different tasks on two benchmarks. O1O consistently outperforms all methods across all tasks on both benchmarks except for USD [He et al., 2023] on S-OWOD in terms of unknown performance (U-Recall) while maintaining a competitive or the highest Known mAP. Note that there's a trade-off between known and unknown performance, as each method can make a fixed number of predictions in total. While increasing unknown recall by predicting more unknowns is possible, it would inevitably degrade known performance. Therefore, striking a balance between these two metrics is crucial.

4.5.1 Quantitative Comparison

The closest competitors in Table 4.2 are from DETR-based approaches, including OW-DETR [Gupta et al., 2022], PROB [Zohar et al., 2023], CAT [Ma et al., 2023], and USD [He et al., 2023]. We encountered a bug in the evaluation of M-OWOD due to duplicated filenames in the test set. After resolving this issue[1], we report the improved results of PROB as PROB†. PROB does not rely on any pseudo-labels but instead learns an objectness distribution. This allows them to achieve the best performance in terms of known mAP on several tasks. However, in terms of unknown recall, PROB falls behind other approaches that use pseudo-labels such as MEPU [Fang et al., 2023] and USD [He et al., 2023]. Our method also utilizes pseudo-labels and outperforms all these methods except for USD [He et al., 2023] on S-OWOD in terms of unknown performance, while being among the top three in

[1]This issue might apply to other DETR-based methods. However, since the code or checkpoints are not available for them, except for OW-DETR [Gupta et al., 2022], which is outperformed by PROB, we could not check or re-evaluate them.

Task IDs	Task 1		Task 2				Task 3				Task 4		
Methods	U-Recall (↑)	mAP (↑) Current known	U-Recall (↑)	mAP (↑) Previously known	mAP (↑) Current known	mAP (↑) Both	U-Recall (↑)	mAP (↑) Previously known	mAP (↑) Current known	mAP (↑) Both	mAP (↑) Previously known	mAP (↑) Current known	mAP (↑) Both
ORE - ebui [Joseph et al., 2021]	4.9	56.0	2.9	52.7	26.0	39.4	3.9	38.2	12.7	29.7	29.6	12.4	25.3
UC-OWOD [Wu et al., 2022b]	2.4	50.7	3.4	33.1	30.5	31.8	8.7	28.8	16.3	24.6	25.6	15.9	23.2
OCPL [Yu et al., 2022]	8.26	56.6	7.65	50.6	27.5	39.1	11.9	38.7	14.7	30.7	30.7	14.4	26.7
2B-OCD [Wu et al., 2022a]	12.1	56.4	9.4	51.6	25.3	38.5	11.6	37.2	13.2	29.2	30.0	13.3	25.8
OW-DETR [Gupta et al., 2022]	7.5	59.2	6.2	53.6	33.5	42.9	5.7	38.3	15.8	30.8	31.4	17.1	27.8
RandBox [Wang et al., 2023]	10.6	61.8	6.3	-	-	45.3	7.8	-	-	39.4	-	-	35.4
PROB [Zohar et al., 2023]	19.4	59.5	17.4	55.7	32.2	44.0	19.6	43.0	22.2	36.0	35.7	18.9	31.5
CAT [Ma et al., 2023]	23.7	60.0	19.1	55.5	32.7	44.1	24.4	42.8	18.7	34.8	34.4	16.6	29.9
MEPU-FS [Fang et al., 2023]	31.6	60.2	30.9	57.3	33.3	44.8	30.1	42.6	21.0	35.4	34.8	19.1	30.9
MEPU-SS [Fang et al., 2023]	30.3	60.0	30.6	57.0	33.1	44.5	30.0	42.2	20.5	35.0	34.3	18.9	30.4
USD - asf [He et al., 2023]	21.6	59.9	19.7	56.6	32.5	44.6	23.5	43.5	21.9	36.3	35.4	18.9	31.3
USD [He et al., 2023]	36.1	58.4	34.7	54.3	31.4	42.7	33.3	41.5	20.5	34.5	33.4	16.6	29.2
PROB† [Zohar et al., 2023]	28.3	66.4	26.4	62.6	39.2	50.9	29.3	49.6	33.5	44.2	44.0	26.5	39.7
O1O (Ours)	49.3	65.1	50.3	61.0	45.0	53.0	49.5	50.0	36.5	45.5	46.2	31.0	42.4
ORE - ebui [Joseph et al., 2021]	1.5	61.4	3.9	56.5	26.1	40.6	3.6	38.7	23.7	33.7	33.6	26.3	31.8
OW-DETR [Gupta et al., 2022]	5.7	71.5	6.2	62.8	27.5	43.8	6.9	45.2	24.9	38.5	38.2	28.1	33.1
PROB [Zohar et al., 2023]	17.6	73.4	22.3	66.3	36.0	50.4	24.8	47.8	30.4	42.0	42.6	31.7	39.9
CAT [Ma et al., 2023]	24.0	74.2	23.0	67.6	35.5	50.7	24.6	51.2	32.6	45.0	45.4	35.1	42.8
MEPU-FS [Fang et al., 2023]	37.9	74.3	35.8	68.0	41.9	54.3	35.7	50.2	38.3	46.2	43.7	33.7	41.2
MEPU-SS [Fang et al., 2023]	33.3	74.2	34.2	67.5	41.0	53.6	33.6	50.0	37.5	45.8	43.2	33.5	40.8
USD - asf [He et al., 2023]	19.2	72.9	22.4	64.9	38.9	51.2	25.4	50.1	34.7	45.0	43.4	33.6	40.9
USD [He et al., 2023]	51.1	69.8	52.1	60.9	33.8	46.7	49.9	45.8	30.2	40.6	39.0	28.7	36.4
O1O (Ours)	49.8	72.6	51.1	65.3	44.9	54.6	48.1	49.5	41.5	46.8	47.3	42.0	45.9

Table 4.2: **Comparison on the OWOD Benchmarks.** We compare our method O1O to the state-of-the-art on M-OWOD (**top**) and S-OWOD (**bottom**) across 4 different tasks on each. We report recall for unknown classes (U-Recall) and mAP@0.5 for known classes, separately for previously, currently, and all known objects. In each column, we highlight the **first**, **second** and **third** best results. O1O outperforms all existing OWOD methods across all tasks in terms of U-Recall except for USD [He et al., 2023] on S-OWOD. Note that USD relies on SAM proposals which significantly contributes to its performance as can be seen from inferior results without SAM (USD - asf). O1O is also consistently among the top-performing methods in terms of known mAP. Since all 80 classes are known in Task 4, U-Recall is not computed. PROB† indicates our evaluation of PROB [Zohar et al., 2023] after fixing the duplicate issue on the test set of M-OWOD.

terms of known performance.

S-OWOD vs. M-OWOD: While the two benchmarks differ in their partitioning of superclasses, our method demonstrates the ability to generalize to both, as indicated by the results. The superclass separation between tasks in S-OWOD simplifies the learning process, leading to better-known performance across all methods compared to M-OWOD. Additionally, the five times larger size of the Task 1 dataset

in S-OWOD contributes to higher known detection performances on this benchmark. The abundance of training data enables the model to learn more effectively, resulting in improved detection accuracy.

4.5.2 Comparison on Pseudo-label Approach

To assess the effectiveness of geometric pseudo-labels, we compare our method to previous approaches that also utilize pseudo-labels from alternative sources. Specifically, we compare against MEPU [Fang et al., 2023], which uses pseudo-labels from Object Localization Network (OLN [Kim et al., 2021]). OLN, like GOOD [Huang et al., 2023], trains the Region Proposal Network in a class-agnostic manner but relies solely on appearance cues. By leveraging geometric cues as an additional source for pseudo-labels, as proposed in GOOD [Huang et al., 2023], our method outperforms MEPU across all benchmarks. Additionally, we compare our method to USD [He et al., 2023], which utilizes pseudo-labels from SAM [Kirillov et al., 2023] and achieves an impressive unknown recall. Despite the advantage of USD in terms of pseudo-label source due to the internet-scale training data of SAM [Kirillov et al., 2023], our method, leveraging geometric pseudo-labels, outperforms USD on M-OWOD and approaches its performance on S-OWOD.

4.5.3 Qualitative Comparison

We visualize the top-10 proposals of OW-DETR [Gupta et al., 2022], PROB [Zohar et al., 2023] and our model O1O in Fig. 4.1 for S-OWOD and in Fig. 4.2 for M-OWOD. In most cases, OW-DETR and our method have a better ranking than PROB, prioritizing known objects with higher confidence and representing unknown predictions with lower scores. As shown in Fig. 4.1, our model can detect the umbrellas in the background in the first row, can locate the monitor and tape precisely in the second row, and pans and the towel in the third row. Some detected unknowns are not even annotated in the official COCO dataset, therefore not contributing to the Unknown Recall metrics, such as tape, pan, and towel. O1O can detect the traffic lights and label them as unknowns, while previous work failed to do so. Lastly,

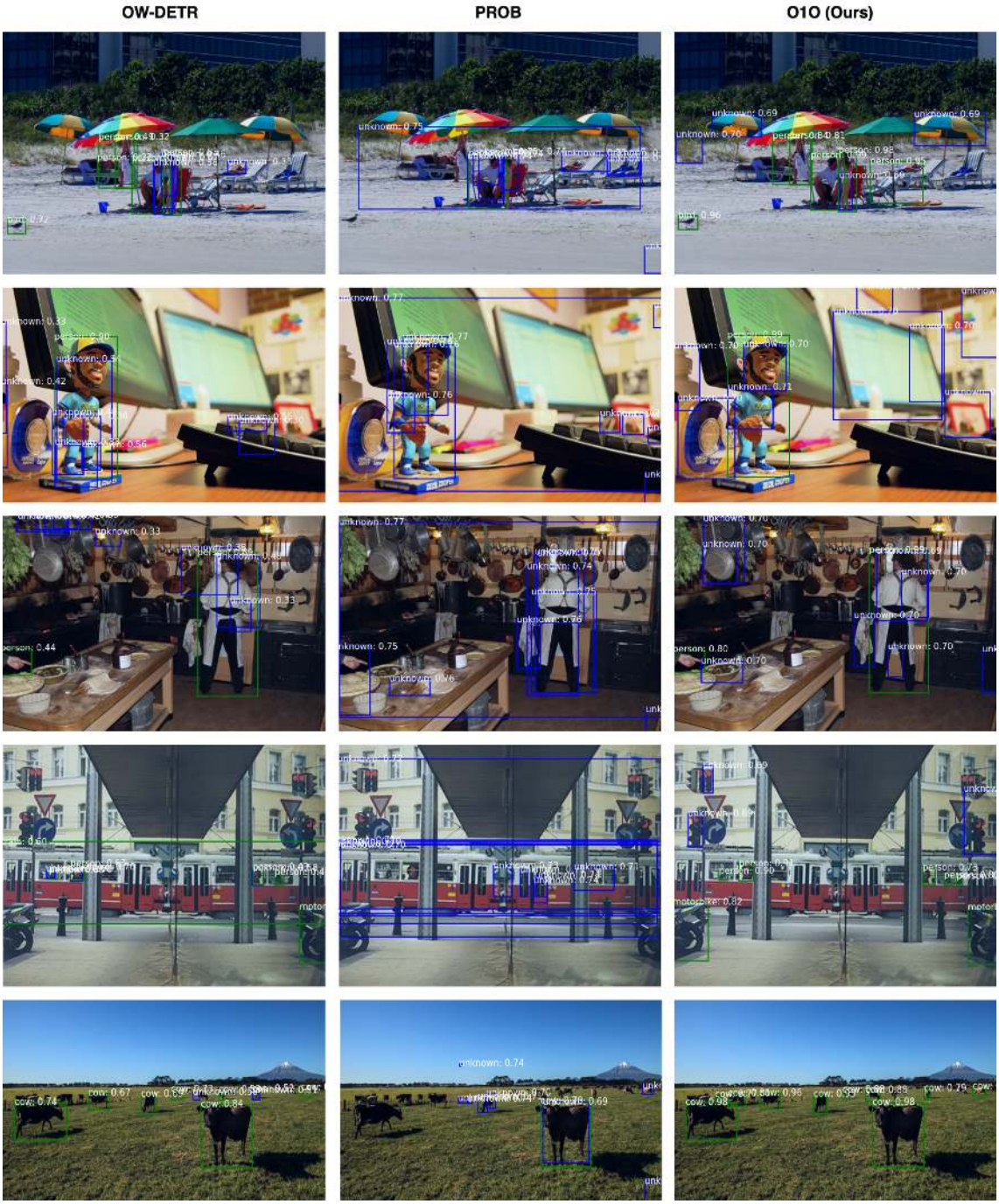

Figure 4.1: **Qualitative Comparison on S-OWOD.** Visualizations of top-10 proposals of OW-DETR, PROB, and O1O(Ours).

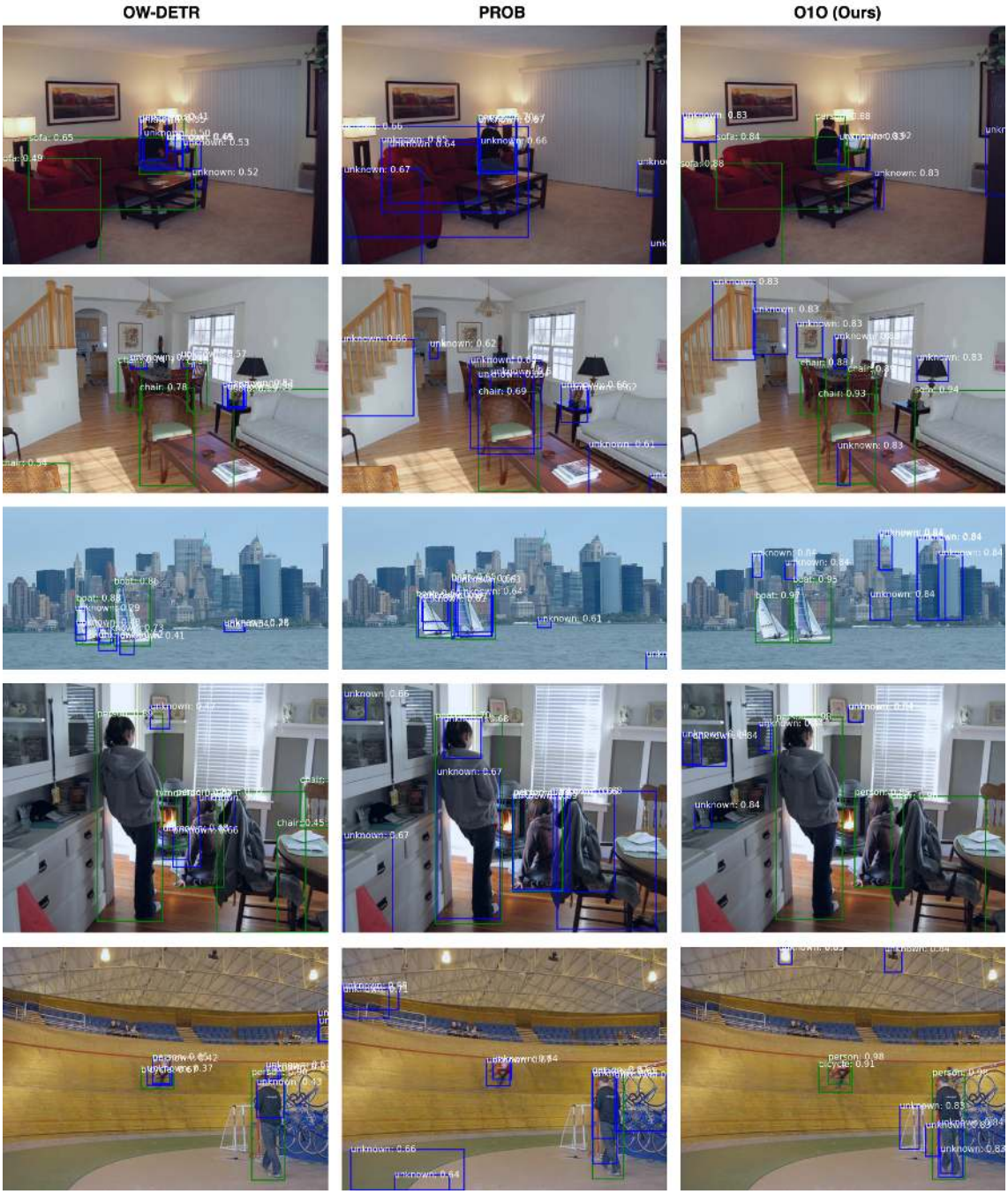

Figure 4.2: **Qualitative Comparison on M-OWOD.** Visualizations of top-10 proposals of OW-DETR, PROB, and O1O(Ours).

without unknowns, such as the last row, our method can label knowns without any confusion. Our method performs similarly on M-OWOD as illustrated in Fig. 4.2. It can precisely detect the lamps, frames, buildings, or small objects in the background while maintaining a meaningful ranking between known and unknown predictions.

4.5.4 Incremental Learning

The active selection strategy, proposed by PROB [Zohar et al., 2023], results in an impressive performance for both previously and the current known in Task 1 on the M-OWOD benchmark. We adopt a similar strategy using our proposed class conditional probability (3.1) to select the exemplars. As more classes become known in Task 2, our method starts to catch up, eventually outperforming previous methods in Task 3 and 4 when most/all classes become known. This is because our superclass representation benefits more from the variety that comes with newly introduced classes in each task. By improving its class conditional probabilities, it consequently enhances our selection of exemplars for the next tasks. On S-OWOD, other methods such as MEPU [Fang et al., 2023] and CAT [Ma et al., 2023] also perform competitively, benefiting from the availability of more data to learn from in Task 1. We observe a similar trend in our model's performance to M-OWOD. As more classes become known through each task, our model improves its representation of superclasses, leading to increases in its ranking in terms of known performance.

In Fig. 4.3 and 4.4, we visualize the predictions of our checkpoints across different tasks of S-OWOD and M-OWOD, respectively. We show the predictions matched to ground truth objects to showcase the development clearly. Although some classes are unknown in the first tasks, our model can still locate them. As the space of known categories expands, O1O learns to label them correctly without forgetting the previously learned classes, showing the effectiveness of our exemplar replay finetuning.

Figure 4.3: **Incremental Learning Performance Across Tasks.** Visualizations of predictions matched to ground truth objects across the tasks of S-OWOD.

Figure 4.4: **Incremental Learning Performance Across Tasks.** Visualizations of predictions matched to ground truth objects across the tasks of M-OWOD.

Chapter 5

ANALYSIS

In this section, we first report the results of our component ablation in Section 5.1. We ablate each component incrementally to show the effect of each on known and unknown performance in Table 5.1. We use Task 1 of the M-OWOD benchmark for the component ablation study. We also perform an additional ablation study on the source of pseudo-labels in Section 5.2, to prove the strength of geometric cues across both benchmarks. Finally, we highlight some failure cases of our method in Section 5.4.

5.1 Component Ablation

5.1.1 Architecture

Our baseline is Deformable DETR, a common choice in previous DETR-based approaches. Initially, we augment the model with a K+1 unknown index and supervise each unmatched query to predict this label. This greedy strategy of predicting everything else as unknown already achieves a good U-Recall but leads to some loss in known mAP. Next, we evaluate the effect of incorporating architectural improvements to Deformable DETR. Although these changes greatly enhance convergence and slightly improve known mAP performance, they do not yield substantial gains in unknown performance.

5.1.2 Geometric Pseudo-labels

Unknown performance improves significantly when we add the geometric pseudo-labels from GOOD [Huang et al., 2023]. For context, the previous state-of-the-art method USD reports 36.1 for unknown recall [He et al., 2023]. This improvement can be largely attributed to SAM proposals, as evidenced by the significantly lower recall

Components	U-Recall (↑)	mAP (↑)
Deformable DETR	32.6	59.2
+ Arch. Improvements	31.6	61.4
+ Geo. Pseudo-Labels	50.7	55.0
+ Superclasses	49.3	65.1
PROB	28.3	66.4
+ Arch. Improvements	27.8	66.1
+ Geo. Pseudo-Labels	50.1	59.0

Table 5.1: **Component Analysis.** We perform an ablation study on Task 1 of M-OWOD by integrating our contributions one by one to the baseline, Deformable DETR, starting with architectural improvements, then geometric pseudo-labels, and finally superclasses. We also apply the same changes to PROB at the bottom part by using their objectness head instead of superclasses.

without SAM (USD - asf vs. USD). Despite the detector having the same training set limitations as SAM, the introduction of geometric pseudo-labels facilitates a substantial performance increase, reaching 50.7%.

5.1.3 Superclasses

The impressive improvement in unknown performance comes at the cost of significant degradation in the known mAP (61.4 → 55.0), which contradicts the intended goal of OWOD to preserve known performance. Our approach with superclasses addresses this issue by recalibrating the classification scores, resulting in a good balance between known and unknown performance. The visualization of the top-10 predictions by our model compared to PROB in Fig. 5.1 clearly shows the benefit of this recalibration.

5.1.4 PROB with Benefits:

To further clarify our contribution with superclasses, we apply the same changes to PROB at the bottom part of Table 5.1. Compared to the greedy Deformable DETR

Figure 5.1: **Qualitative Comparison of Top-10 Predictions.** In this figure, we visualize top-10 predictions for PROB [Zohar et al., 2023] (**top**) and our model O1O (**bottom**). We color known predictions in green and unknown in blue. As can be seen from the top row, the top-10 predictions of PROB are almost all unknown, whereas our model can predict known classes like dog, bike on the first, person on the second, and person and car on the third sample with high confidence. Our model ranks unknowns with relatively lower confidence scores, indicating a better calibration of class confidence scores. Overall, our unknown predictions tightly fit objects thanks to supervision from geometric proposals and correctly locate objects in the background.

baseline, PROB's objectness probability helps maintain a high known performance, by ranking background regions lower in the confidence scores. Applying the same architectural improvements to PROB does not affect its performance. Incorporating geometric pseudo-labels improves the unknown performance of PROB by almost doubling the unknown recall (27.8 $\rightarrow$ 50.1) but also negatively impacts the known performance (-7.1 in mAP), similar to what we observed with our method. PROB's approach, which involves learning a single distribution to represent objectness, fails to mitigate the degradation in known performance. This underscores the importance of shaping the representation space with superclasses (Fig. 1.4).

5.1.5 Unknown Scoring Function

We define our unknown probability based on the known class confidence scores (3.2). In Table 5.2, we report the performance of alternative scoring functions on S-OWOD,

Metrics	U-Recall (↑)	mAP (↑)
$1 - \max\big(p\left(\mathbf{s}\middle|\mathbf{q}\right)\big)$	50.7	72.2
$1 - \max\big(p'\left(\mathbf{c}\middle|\mathbf{q}\right)\big)$	52.8	68.2
$1 - \text{sum}\big(p'\left(\mathbf{c}\middle|\mathbf{q}\right)\big)$	52.5	71.5
$1 - \text{sum}\big(p'\left(\mathbf{c}\middle|\mathbf{q}\right)\big)$+ thr.	49.8	72.6

Table 5.2: **Inference Score Ablation.** We perform an ablation study on the inference score used for unknowns on S-OWOD Task 1. The first row corresponds to MSP [Hendrycks and Gimpel, 2017] applied to superclass probabilities p, and the second row is MSP applied to recalibrated class probabilities p' (3.1). The third row is an extended version of MSP, which uses the union of recalibrated class probabilities p', as explained in (3.2). Because of its better balance, we use the third with a threshold that is experimentally set on the training set.

Task 1. For a confident known prediction, the maximum softmax probability (MSP) [Hendrycks and Gimpel, 2017] becomes high, and the inverse becomes low, so we can obtain the unknown probability by simply subtracting it from 1. We apply MSP on superclass probabilities alone in the first row and on recalibrated class probabilities (3.1) in the second row. In the third row, we extend the max in MSP to the union of recalibrated class probabilities. This indicates the combined belief of the model about a prediction to belong to any known class. Depending on the type of scoring function, there is a trade-off between known and unknown performances. We found the third approach to have the best balance between U-Recall and known mAP. Furthermore, we calculate a threshold on a subset of the training set such that 95% of the known instances would be classified correctly. This thresholding helps us control the trade-off between known and unknown slightly better without violating any principles.

5.2 Source of Pseudo-labels

To analyze the strength of geometric pseudo-labels, we performed an additional study to compare them to RGB-based proposals. As shown in Table 5.3, while

Source	U-Recall (↑)	mAP (↑)
RGB	51.5	62.8
Depth	48.0	63.0
Surface Normals	51.6	63.6
RGB + Depth	51.5	62.1
Depth + Surface Normals	51.0	62.8
RGB + Surface Normals	53.3	63.4
RGB + Depth + Surface Normals	52.8	62.5
RGB	48.7	70.3
Depth	45.9	72.6
Surface Normals	52.9	71.0
RGB + Depth	48.4	71.3
Depth + Surface Normals	49.0	71.5
RGB + Surface Normals	53.9	70.7
RGB + Depth + Surface Normals	50.3	71.4

Table 5.3: **Pseudo-label Source Ablation.** Assessment of geometric vs RGB-based pseudo-labels on Task 1 of both M-OWOD (**top**) and S-OWOD (**bottom**) benchmarks. In terms of unknown recall, surface normals outperform both RGB and depth, especially on S-OWOD. While RGB labels alone perform well for unknowns, they confuse known detection, resulting in reduced mAP. Depth alone generally performs poorly for unknown detection and introduces noise on M-OWOD. Although combining RGB and surface normals leads to higher unknown recall, it further worsens known performance.

the Unknown Recall of RGB and surface normals versions are very close to each other on the M-OWOD benchmark, the difference becomes visible in S-OWOD. The reason behind such discrepancy lies in the effect of dataset size. The M-OWOD Task 1 dataset is approximately one-sixth the size of the S-OWOD Task 1 dataset. Therefore, any use of additional annotations helps the model to improve its recall significantly. However, in the abundance of training instances, such as in S-OWOD, the model already leverages the full spectrum of information available from RGB input. When the proposals obtained from surface normals are used for S-OWOD, it outperforms the RGB variant by +4.2 in Unknown Recall with a +0.7 better Known mAP. In both benchmarks, RGB pseudo-labels cause the most confusion, hurting the known mAP considerably.

Depth alone yields the poorest performance among the three methods in terms of unknown performance and fails to enhance performance when combined with others. Interestingly, it introduces additional noise that hinders both known and unknown object detection on M-OWOD, while on S-OWOD, it has the highest known performance. Due to the tradeoff between known and unknown performance, if one is notably low, it leaves room for the other to be high.

Overall, the combination of RGB and surface normals pseudo-labels yields the highest unknown recall on both benchmarks. However, the increased volume of pseudo-labels negatively impacts known performance further. Striving for a more balanced and simple approach, we opt to rely solely on surface normals as our primary source of pseudo-labels.

5.3 Number of Pseudo-labels

After deciding which source of pseudo-labels to use, the next question is how many to use. To address this, we conducted experiments with different numbers of pseudo-labels on M-OWOD Task 1, using surface normals as the source. We implement the number of pseudo-labels as a constraint after the NMS module and confidence thresholding. This means that after removing overlapping or low-confidence boxes, we select the top-k boxes, where k is the number of known ground truth targets plus

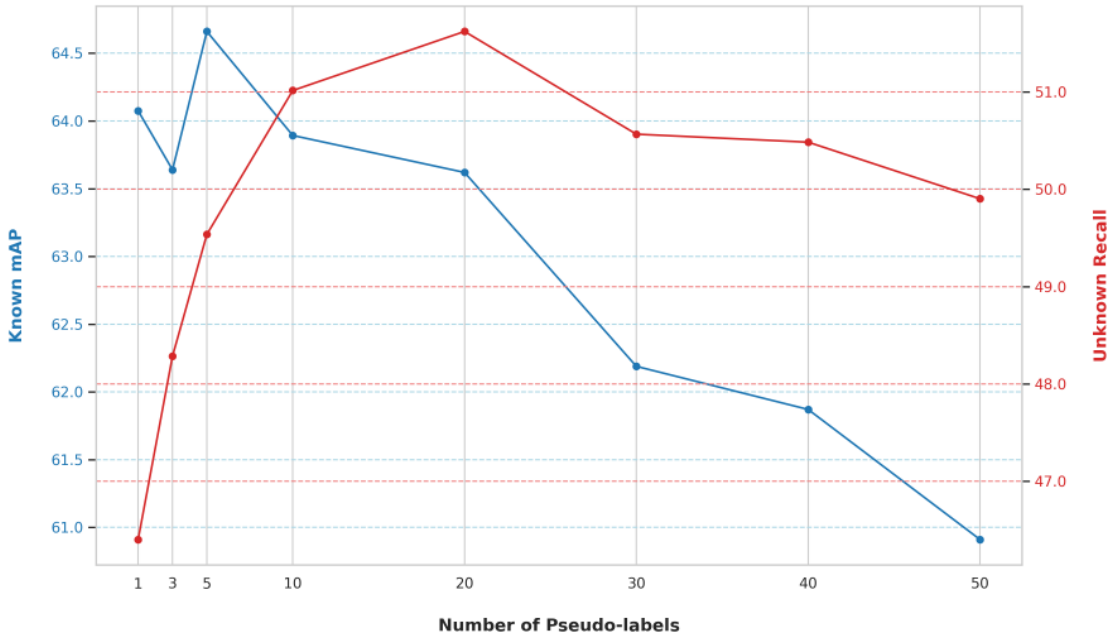

Figure 5.2: **Number of Pseudo-labels Ablation.** We study the effect of utilizing different numbers of pseudo-labels on Task 1 of M-OWOD, using surface normals as the source. Unknown Recall shows a clear improvement until 20 labels, after which it starts to decrease. Known mAP remains within a range until 5 labels, then shows a clear decrease trend, particularly after surpassing 20 labels. Optimal performance appears to be attained with 20 pseudo-labels, balancing accuracy, and noise.

the number of pseudo-labels.

We report how Known mAP values evolve against Unknown Recall in Fig. 5.2. We start with small numbers such as 1, 3, and 5, which are typically used in previous work [Joseph et al., 2021, Gupta et al., 2022, Ma et al., 2023], and increase the number up to half of the total number of queries we used, which is 100. It is expected to see a trade-off between known and unknown performance, which is confirmed by our experiments. As the number of pseudo-labels increases, Unknown Recall shows a clear improvement up until the number 20. Known mAP has an unstable response to the number of pseudo-labels up until the number 5, remaining in a range, after which it exhibits a clear decreasing trend, as expected.

After reaching an equilibrium point at 20, both Known mAP and Unknown Recall start to decline because the noisy and inaccurate pseudo-supervision dominates the signals coming from the actual ground truth supervision. As the equilibrium

point, we choose 20 as our number of pseudo-labels. This is not implemented as a hard constraint, meaning if there are fewer than 20 pseudo-boxes with a confidence threshold greater than 0.5, we use exactly how many there are.

5.4 Failure Cases

Since our pseudo-labels depend on predicted surface normals, our model is vulnerable to cases when the surface normals cannot be estimated accurately. One failure mode occurs when textual differences in expansive backgrounds, such as the sky or floor, create variations in local normal values. This encourages the model to predict objects in such regions, as in the first row of Fig. 5.3. Another failure mode arises in complex indoor scenes containing numerous objects. This can be attributed to factors such as objects with flat shapes, like keyboards, or densely populated regions with entangled objects, as illustrated in the second row of Fig. 5.3. Our model occasionally ranks blank regions that do not correspond to real objects higher than known classes, such as the person and dog in the upper left, the table in the upper right, the sofa/chair in the lower left, or the person in the lower right of Fig. 5.4.

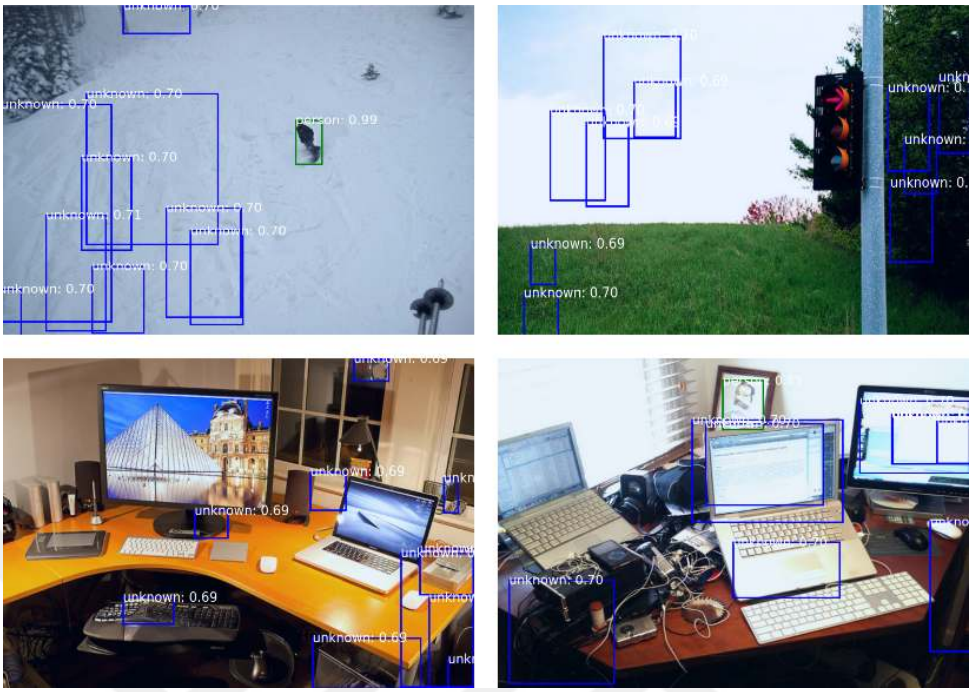

Figure 5.3: **Failure cases on S-OWOD.** Due to the reliance on estimated surface normals, our model may generate redundant pseudo-labels in regions with local textural differences (**top**) or struggle to locate objects in crowded scenes (**bottom**).

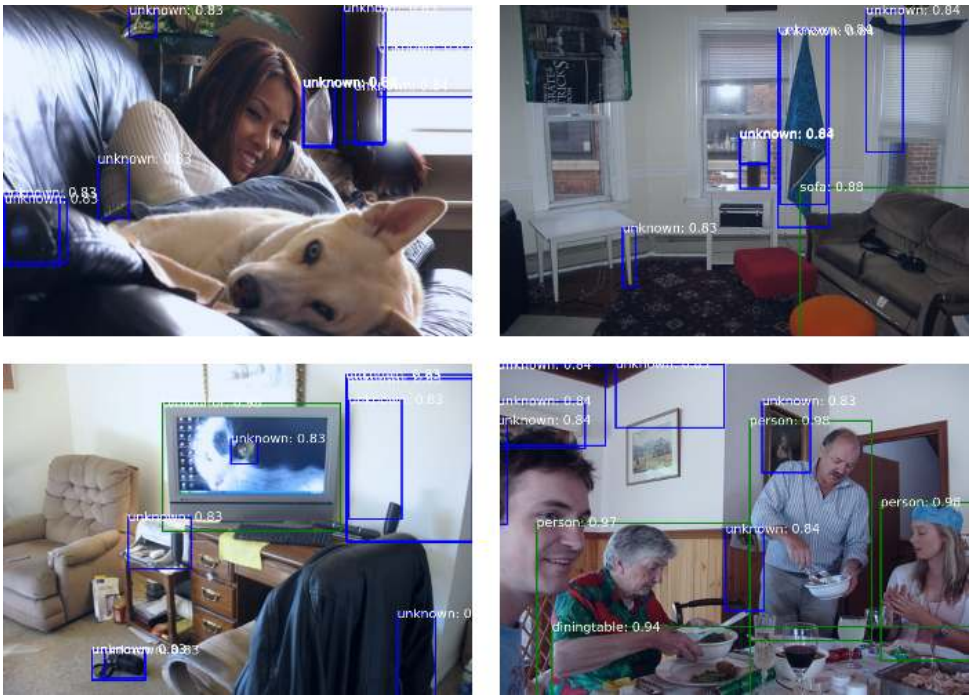

Figure 5.4: **Failure cases on M-OWOD.** Examples of ranking redundant unknown predictions higher than some known instances.

Chapter 6

CONCLUSION

6.1 Summary

As deep learning models evolve, our capacity to tackle real-world problems improves. In this thesis, we propose a solution for detecting novel objects within a given scene. We build on DETR-based object detection models by better utilizing unmatched queries for unknown object detection. Specifically, we first integrate pseudo-labels from a class-agnostic region proposal network based on geometric cues, i.e. surface normal maps. Integrating geometric pseudo-labels increases unknown recall significantly since the model is encouraged to look for all objects, including the known category set and beyond. We also demonstrate that any pseudo-label source is effective compared to none. Yet, depending on the characteristics of the training dataset or inference input, geometric cues can provide additional signals to retrieve more objects in complex scenes, thus increasing unknown recall. However, noisy pseudo-labels can lead to overfitting to irrelevant patterns rather than capturing the underlying true relationships. Therefore, it requires better modeling of the known classes to handle the increased confusion.

To better handle the effects of noisy supervision, we introduce a prior on the queries via superclass classification in addition to per-class classification scores. Inspired by human cognitive science, we use superclass categories to learn discriminative prototypes that capture inter-class similarities. Superclass classification probabilities recalibrate the scores of known classes based on semantic proximity, effectively reducing confusion caused by pseudo-labels. Additionally, we identify the unknown, or odd-one-out, by calculating the probability of not belonging to any known group sharing common attributes. Our method, O1O, achieves improved performance on known classes with the assistance of superclass representation shap-

ing while also ranking among the top-performing methods for unknown recall. We also empirically validate our contributions and investigate the choice of pseudo-label source, the number of pseudo-labels, and the inference scoring function through further analysis.

6.2 Limitations and Future Work

While our pseudo-labeling strategy demonstrates promising results, it is essential to acknowledge its limitations. One significant constraint arises from its reliance on predicted geometric cues for proposal generation. When these cues cannot be reliably computed, the effectiveness of our approach diminishes. If accurate measurements of depth and surface normals were available, the negative effects of noisy labels could be reduced. Moreover, since proposals are predicted by a network and not guaranteed to be perfect, striking a balance between noise reduction and accuracy remains a critical challenge.

Our approach of incorporating geometric signals represents only one method within a broader landscape of possibilities. In this thesis, we explored alternative approaches, including concatenating geometric cues with the RGB input, embedding them through cross-attention operations to guide the query vectors to focus on candidate regions, decoupling RGB and geometric signals with two sets of queries to prevent bias leakage, and more. However, we encountered challenges in managing the optimization process and balancing supervision between known and unknown objects. While these results were obtained as part of our exploration process, we omit them here to maintain focus and clarity. Nevertheless, the investigation into better design and engineering choices for incorporating geometric signals presents an avenue for future research.

In contrast to our current approach, which employs a one-level hierarchy of superclasses, there's potential in exploring more fine-grained hierarchies, especially for expansive categories like vehicles. For example, vehicles with long bodies, such as buses and trains, exhibit significant visual differences from two-tiered vehicles, such as bicycles and motorcycles. Forcing them into a single representation may

lead to suboptimal convergence. Better modeling of the hierarchy levels can further enhance the effectiveness of our approach. Given that label sets in most datasets typically consist of tens or, at most, hundreds of magnitudes, this approach seems manageable through human effort. However, Large Language Models (LLMs) can also be employed in the process of proposing different hierarchy levels. While this investigation falls beyond the scope of our work, we hope our initial steps serve as inspiration for other researchers.

On another note, it is crucial to assess the practicality and real-life applicability of O1O critically. Even though our model shows impressive performance both quantitatively and qualitatively, there remains uncertainty regarding its performance in real-world scenarios. Mainly due to the evaluation protocol of OWOD benchmarks, all previous methods and our method O1O focus on unknown recall instead of unknown precision. Although this emphasis on unknown recall is partially attributed to the missing-label problem, it also creates an exploitable vulnerability that can be strategically manipulated. By using the maximum number of predictions and being greedy toward regions outside known objects, unknown recall can be pushed at the cost of false positives, hence, decreased unknown precision. With this in mind, we made an effort to show that not only our averaged statistics but also our top-k predictions are meaningful and informative. Even so, we believe better evaluation protocols are necessary for the future of OWOD research. As an example, anomaly segmentation benchmarks offer a more accurate framework for evaluating unknown recognition models. Specifically, for self-driving cases, Segment Me If You Can (SMIYC) [Chan et al., 2021] and Fishyscapes [Blum et al., 2021] benchmarks present challenging real-life driving scenarios for models trained on common driving datasets like CityScapes [Cordts et al., 2016] and Mapillary [Neuhold et al., 2017]. The scenarios included in these benchmarks are strategic and representative of the real world, including instances like animals crossing the road, boxes left in the middle of the road, and diverse weather conditions. On the contrary, the COCO split benchmarks of OWOD are generated in an unnatural way and do not depict real-life challenges. We hope to see more comprehensive evaluation protocols for the OWOD community in the near future.

BIBLIOGRAPHY

[Bendale and Boult, 2015] Bendale, A. and Boult, T. E. (2015). Towards open world recognition. In *CVPR*.

[Blum et al., 2021] Blum, H., Sarlin, P.-E., Nieto, J. I., Siegwart, R. Y., and Cadena, C. (2021). The fishyscapes benchmark: Measuring blind spots in semantic segmentation. *IJCV*, 129:3119–3135.

[Carion et al., 2020] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). End-to-end object detection with transformers. In *ECCV*.

[Caron et al., 2021] Caron, M., Touvron, H., Misra, I., J'egou, H., Mairal, J., Bojanowski, P., and Joulin, A. (2021). Emerging properties in self-supervised vision transformers. In *ICCV*.

[Chan et al., 2021] Chan, R., Lis, K., Uhlemeyer, S., Blum, H., Honari, S., Siegwart, R. Y., Salzmann, M., Fua, P., and Rottmann, M. (2021). SegmentMeIfYouCan: A benchmark for anomaly segmentation. *arXiv.org*, 2104.14812.

[Cheng et al., 2022] Cheng, B., Misra, I., Schwing, A. G., Kirillov, A., and Girdhar, R. (2022). Masked-attention mask transformer for universal image segmentation. In *CVPR*.

[Cordts et al., 2016] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *CVPR*.

[Dalal and Triggs, 2005] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR*.

[Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet: a large-scale hierarchical image database. In *CVPR*.

[Dhamija et al., 2020] Dhamija, A. R., Günther, M., Ventura, J., and Boult, T. E. (2020). The overlooked elephant of object detection: Open set. In *WACV*.

[Du et al., 2022] Du, X., Gozum, G., Ming, Y., and Li, Y. (2022). SIREN: Shaping representations for detecting out-of-distribution objects. In *NeurIPS*.

[Eftekhar et al., 2021] Eftekhar, A., Sax, A., Malik, J., and Zamir, A. (2021). Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In *ICCV*.

[Everingham et al., 2010] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *IJCV*, 88:303–338.

[Fang et al., 2023] Fang, R., Pang, G., Zhou, L., Bai, X., and Zheng, J. (2023). Unsupervised recognition of unknown objects for open-world object detection. *arXiv.org*.

[Felzenszwalb et al., 2009] Felzenszwalb, P. F., Girshick, R. B., McAllester, D., and Ramanan, D. (2009). Object detection with discriminatively trained part-based models. *PAMI*, 32:1627–1645.

[Girshick, 2015] Girshick, R. (2015). Fast r-cnn. In *ICCV*.

[Girshick et al., 2013] Girshick, R. B., Donahue, J., Darrell, T., and Malik, J. (2013). Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*.

[Gupta et al., 2022] Gupta, A., Narayan, S., Joseph, K., Khan, S., Khan, F. S., and Shah, M. (2022). OW-DETR: open-world detection transformer. In *CVPR*.

[Hall et al., 2018] Hall, D., Dayoub, F., Skinner, J., Zhang, H., Miller, D., Corke, P., Carneiro, G., Angelova, A., and Sünderhauf, N. (2018). Probabilistic object detection: Definition and evaluation. In *WACV*.

[Han et al., 2022] Han, J., Ren, Y., Ding, J., Pan, X., Yan, K., and Xia, G. (2022). Expanding low-density latent regions for open-set object detection. In *CVPR*.

[He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. In *CVPR*.

[He et al., 2023] He, Y., Chen, W., Tan, Y., and Wang, S. (2023). USD: unknown sensitive detector empowered by decoupled objectness and segment anything model. *arXiv.org*, 2306.02275.

[Hebart et al., 2020] Hebart, M. N., Zheng, C. Y., Pereira, F., and Baker, C. I. (2020). Revealing the multidimensional mental representations of natural objects underlying human similarity judgements. *Nature human behaviour*, 4(11):1173–1185.

[Hendrycks and Gimpel, 2017] Hendrycks, D. and Gimpel, K. (2017). A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *ICLR*.

[Huang et al., 2023] Huang, H., Geiger, A., and Zhang, D. (2023). GOOD: Exploring geometric cues for detecting objects in an open world. In *ICLR*.

[Joseph et al., 2021] Joseph, K. J., Khan, S., Khan, F. S., and Balasubramanian, V. N. (2021). Towards open world object detection. In *CVPR*.

[Kim et al., 2021] Kim, D., Lin, T.-Y., Angelova, A., Kweon, I.-S., and Kuo, W. (2021). Learning open-world object proposals without learning to classify. *RA-L*, PP:1–1.

[Kirillov et al., 2023] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., and Girshick, R. (2023). Segment anything. In *ICCV*.

[Li et al., 2022] Li, F., Zhang, H., Liu, S., Guo, J., Ni, L. M., and Zhang, L. (2022). DN-DETR: Accelerate detr training by introducing query denoising. In *CVPR*.

[Lin et al., 2017] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. In *ICCV*.

[Lin et al., 2014] 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. In *ECCV*.

[Liu et al., 2021] Liu, S., Li, F., Zhang, H., Yang, X., Qi, X., Su, H., Zhu, J., and Zhang, L. (2021). Dab-detr: Dynamic anchor boxes are better queries for detr. In *ICLR*.

[Liu et al., 2016] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). Ssd: Single shot multibox detector. In *ECCV*.

[Ma et al., 2023] Ma, S., Wang, Y., Fan, J., yu Wei, Y., Li, T. H., Liu, H., and Lv, F. (2023). CAT: localization and identification cascade detection transformer for open-world object detection. In *CVPR*.

[Maaz et al., 2021a] Maaz, M., Rasheed, H. A., Khan, S., Khan, F. S., Anwer, R. M., and Yang, M. (2021a). Class-agnostic object detection with multi-modal transformer. In *ECCV*.

[Maaz et al., 2021b] Maaz, M., Rasheed, H. A., Khan, S. H., Khan, F. S., Anwer, R. M., and Yang, M.-H. (2021b). Multi-modal transformers excel at class-agnostic object detection. *arXiv.org*, 2111.11430.

[McCloskey and Cohen, 1989] McCloskey, M. and Cohen, N. J. (1989). Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier.

[Miller et al., 2018] Miller, D., Dayoub, F., Milford, M., and Sünderhauf, N. (2018). Evaluating merging strategies for sampling-based uncertainty techniques in object detection. In *ICRA*.

[Miller et al., 2017] Miller, D., Nicholson, L., Dayoub, F., and Sünderhauf, N. (2017). Dropout sampling for robust object detection in open-set conditions. In *ICRA*.

[Miller et al., 2021] Miller, D., Sunderhauf, N., Milford, M., and Dayoub, F. (2021). Uncertainty for identifying open-set errors in visual object detection. *RA-L*, 7:215–222.

[Nayal et al., 2023] Nayal, N., Yavuz, M., Henriques, J. F., and Güney, F. (2023). Rba: Segmenting unknown regions rejected by all. In *ICCV*.

[Neuhold et al., 2017] Neuhold, G., Ollmann, T., Rota Bulo, S., and Kontschieder, P. (2017). The mapillary vistas dataset for semantic understanding of street scenes. In *ICCV*.

[Ranftl et al., 2021] Ranftl, R., Bochkovskiy, A., and Koltun, V. (2021). Vision transformers for dense prediction. In *ICCV*.

[Redmon et al., 2016] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *CVPR*.

[Ren et al., 2015] Ren, S., He, K., Girshick, R. B., and Sun, J. (2015). Faster R-CNN: towards real-time object detection with region proposal networks. *PAMI*, 39:1137–1149.

[Saito et al., 2022] Saito, K., Hu, P., Darrell, T., and Saenko, K. (2022). Learning to detect every thing in an open world. In *ECCV*.

[Scheirer et al., 2012] Scheirer, W. J., de Rezende Rocha, A., Sapkota, A., and Boult, T. E. (2012). Toward open set recognition. *PAMI*, 35:1757–1772.

[Uijlings et al., 2013] Uijlings, J. R. R., van de Sande, K. E. A., Gevers, T., and Smeulders, A. W. M. (2013). Selective search for object recognition. *IJCV*, 104:154 – 171.

[Wang et al., 2022] Wang, X., Yu, Z., De Mello, S., Kautz, J., Anandkumar, A., Shen, C., and Alvarez, J. M. (2022). Freesolo: Learning to segment objects without annotations. In *CVPR*.

[Wang et al., 2023] Wang, Y., Yue, Z., Hua, X.-S., and Zhang, H. (2023). Random boxes are open-world object detectors. In *ICCV*.

[Wu et al., 2022a] Wu, Y., Zhao, X., Ma, Y., Wang, D., and Liu, X. (2022a). Two-branch objectness-centric open world detection. *Proceedings of the 3rd International Workshop on Human-Centric Multimedia Analysis*.

[Wu et al., 2022b] Wu, Z., Lu, Y., Chen, X., Wu, Z., Kang, L., and Yu, J. (2022b). UC-OWOD: unknown-classified open world object detection. In *ECCV*.

[Yu et al., 2022] Yu, J., Ma, L., Li, Z., Peng, Y., and Xie, S. (2022). Open-world object detection via discriminative class prototype learning. In *ICIP*.

[Zhao et al., 2022] Zhao, S., Zhang, Z., Schulter, S., Zhao, L., B.G, V. K., Stathopoulos, A., Chandraker, M., and Metaxas, D. N. (2022). Exploiting unlabeled data with vision and language models for object detection. In *ECCV*.

[Zheng et al., 2022] Zheng, J., Li, W., Hong, J., Petersson, L., and Barnes, N. (2022). Towards open-set object detection and discovery. *CVPR Workshops*.

[Zhu et al., 2021] 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 *ICLR*.

[Zohar et al., 2023] Zohar, O., Wang, K.-C., and Yeung, S. (2023). PROB: probabilistic objectness for open world object detection. In *CVPR*.

Appendix A

A.1 OWOD Benchmark Details

Table A.1: Details of the M-OWOD (**top**) and S-OWOD (**bottom**) Benchmarks.

	Task 1	**Task 2**	**Task 3**	**Task 4**
Semantic Split	VOC Classes	Accessory, Animal, Appliance, Outdoor	Food, Sports	Electronic, Indoor, Kitchen, Furniture
# Train Images	16551	45520	39402	40260
# Train Instances	47223	113741	114452	138996
# Test Images	4952	1914	1642	1738
# Test Instances	14976	4966	4826	6039

Semantic Split	Animal, Person Vehicle	Accessory, Appliance, Furniture, Outdoor	Food, Sports	Electronic, Indoor, Kitchen
# Train Images	89490	55870	39402	38903
# Train Instances	421243	163512	114452	160794
# Test Images	3793	2351	1642	1691
# Test Instances	17786	7159	4826	7010

In Table A.1, we report some details of the OWOD benchmarks. M-OWOD, proposed in [Joseph et al., 2021], is the first OWOD benchmark. It is based on the MS-COCO [Lin et al., 2014] and Pascal VOC datasets [Everingham et al., 2010], and each task in this benchmark introduces exactly 20 classes. S-OWOD, proposed in [Gupta et al., 2022], is solely based on the MS-COCO dataset. Each task in S-OWOD introduces 19, 21, 20, and 20 classes, respectively. The test set of each benchmark consists of examples from each class and is the same for all its tasks. However, we specify the numbers per superclass to illustrate the semantic coverage

across the label set.

Semantic Split: As explained in Section 4.2, the M-OWOD benchmark is also named the Mixed-superclass benchmark because of its entangled semantic split. The first task is designed to include all Pascal VOC classes, which include classes belonging to animal, electronic, furniture, kitchen, person, and vehicle superclasses. The remaining classes belonging to animal and vehicle superclasses are introduced in Task 2, and the remaining classes of electronic, furniture, and kitchen superclasses are introduced in Task 4. To address the data leakage resulting from the mixed superclass split of M-OWOD, S-OWOD designs the semantic splits of the tasks by considering each superclass to be fully introduced in its corresponding task. In Table Table A.1, we specify the most frequent superclasses in M-OWOD's semantic split for clarity.

Dataset Size: The biggest difference between the first tasks of the two benchmarks is the dataset size. As evident from Table A.1, the number of training images in the S-OWOD Task 1 set is approximately 5.4 times the number of training images in the M-OWOD Task 1. Similarly, the number of training instances is almost 9 times the number of training instances in M-OWOD Task 1. This difference in dataset size changes the training dynamics significantly. Methods trained on S-OWOD have higher Known mAP values than their M-OWOD counterparts. While the difference in dataset size diminishes in the subsequent tasks, training on the first task plays a vital role in shaping the network's intermediate representation space.

True Unknown Correspondence: Another difference between the benchmarks arises from the dataset splits of the two. In S-OWOD Task 1 images, there may be unknown objects in the background. However, their annotations are removed during training to adhere to the protocol. Thus, the model essentially sees them but does not explicitly recognize them as objects. Subsequently, when we incorporate pseudo-labels, they might overlap with the real unknown objects that were removed according to the protocol. In contrast, initially in M-OWOD Task 1 images, there are no unknown objects in the background. Therefore, even when we use pseudo-labels, they do not overlap with the test time unknown categories. Nevertheless, utilizing them can encourage the model to identify regions beyond the

known classes, potentially including those unannotated in the full COCO set.

A.2 More Implementation Details

We utilize the repository of GOOD [Huang et al., 2023] for the RPN module and pseudo-label extraction. Our implementation for the detection model is based on the Deformable-DETR [Zhu et al., 2021] and DN-DETR repositories [Li et al., 2022]. We use the default hyperparameters unless stated otherwise. Here, we provide additional details about our introduced superclass supervision.

A.2.1 Superclass Classification

For superclass classification, we use a single linear layer to transform the hidden dimension $H = 256$ of query vectors to the number of superclasses. We adopt the auxiliary supervision as well by placing a clone of the superclass classification layer after each decoder layer with shared parameters. At each task, even if one class of a superclass is known (especially for M-OWOD), we start training the classification weights for it. Later, when new classes of the specific superclass are introduced, we resume training of the specific weights, and our model adapts to the new group of classes by shaping its representation.

For the loss function, we use focal loss, following the per-class classification formulation. Focal loss [Lin et al., 2017] is a version of the cross-entropy loss function, which is proposed to address the class imbalance by assigning higher weights to misclassified examples and effectively down-weights them. As DN-DETR employs denoising losses, we also add a label denoising loss for superclass classification. Similarly, it is implemented with focal loss. This denoising loss randomly flips the ground truth superclass labels for a set of noised queries and teaches the model to reconstruct the true labels, enhancing the stability and convergence of the overall training process.