Wanderlust: Online Continual Object Detection in the Real World

Jianren Wang[1] Xin Wang[2] Yue Shang-Guan[3] Abhinav Gupta[1]

[1]Carnegie Mellon University [2]Microsoft Research [3]University of Texas, Austin

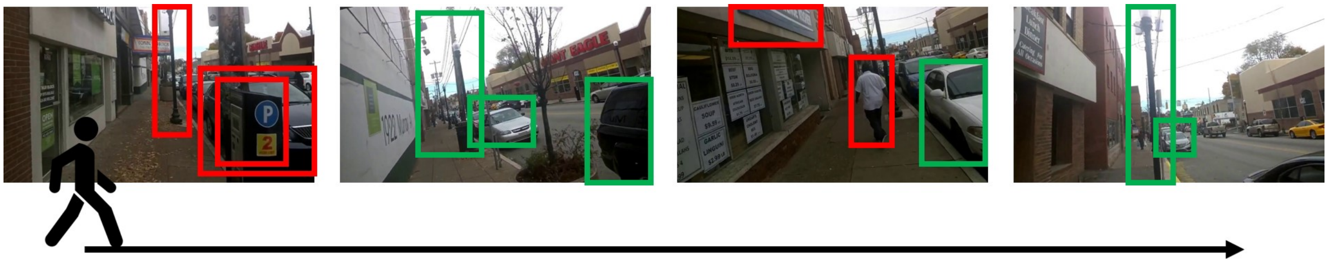


Figure 1: **Wanderlust**: Imagine an embodied agent is walking on the street. It may observe new classes and old classes simultaneously. The agent needs to learn fast given only a few samples (red) and recognize the subsequent instances of the class once a label has been provided (green). In this work, we introduce a new online continual object detection benchmark through the eyes of a graduate student to continuously learn emerging tasks in changing environments.

Abstract

Online continual learning from data streams in dynamic environments is a critical direction in the computer vision field. However, realistic benchmarks and fundamental studies in this line are still missing. To bridge the gap, we present a new online continual object detection benchmark with an egocentric video dataset, Objects Around Krishna (OAK). OAK adopts the KrishnaCAM videos, an ego-centric video stream collected over nine months by a graduate student. OAK provides exhaustive bounding box annotations of 80 video snippets (∼17.5 hours) for 105 object categories in outdoor scenes. The emergence of new object categories in our benchmark follows a pattern similar to what a single person might see in their day-to-day life. The dataset also captures the natural distribution shifts as the person travels to different places. These egocentric long running videos provide a realistic playground for continual learning algorithms, especially in online embodied settings. We also introduce new evaluation metrics to evaluate the model performance and catastrophic forgetting and provide baseline studies for online continual object detection. We believe this benchmark will pose new exciting challenges for learning from non-stationary data in continual learning. The OAK dataset and the associated benchmark are released at https://oakdata.github.io/.

1. Introduction

Modern object detectors have made substantial progress on internet images [5, 16, 37]. Nevertheless, challenges remain when detecting small objects [7], scaling to a large number of categories [15] or learning from only a few labeled examples [19, 44]. The detector often degenerates significantly when deployed on robots or ego-centric videos in an embodied environment [8].

If we take a closer look at the typical learning setup, most of the advances in object detection have been realized using static images in an offline learning setup. In this setup, the data is labeled with a fixed set of categories and divided into two parts: training and testing. There is a training phase where the detectors are learned by randomly shuffling and feeding the training data for hundreds of epochs, followed by an evaluation on the test set. However, this offline training and evaluation setup often does not reflect how humans or embodied AI agents learn.

Unlike the current *static offline* settings, humans receive a continuous temporal stream of visual data and train and test the model on the same visual data, which is an *online continual* setting. The categories of interest are unknown beforehand. The model needs to learn new object categories when objects belonging to previously unseen categories appear. Most of the learning happens online; we cannot use the training data repeatedly across hundreds of epochs.

A side effect of this online continual learning setting is catastrophic forgetting [30]. Though previous works [1, 21, 29, 40, 48] attempt to address the issue, they are usually evaluated offline and do not work well on structural prediction tasks like object detection. What hinders the progress in online continual learning is the lack of realistic datasets and benchmarks. Most of the current research [32, 41, 49] re-purpose existing static datasets such as VOC and COCO to evaluate continual object detection. These approaches use object categories one by one in a sequential manner. These manual splits and artificial setups differ from the scenarios often encountered by embodied agents, where the emergence of new tasks often follows the trajectories of the agents and the frequencies of the object instances vary from task to task. For example, the agents might observe the instances of the same category after a few hours or even days. They may visit some objects more often than others and revisit previously observed objects.

In this paper, we present a new online continual object detection benchmark. Our benchmark consists of a new labeled dataset – **OAK** (Objects Around Krishna). OAK uses the videos from the KrishnaCam [43] dataset – an ego-centric video dataset collected over nine months of a graduate student's life. OAK contains 80 labeled video snippets totaling around 17.5 hours (roughly 1/4 of the raw videos in KrishnaCam) with bounding box annotations of 105 object categories in outdoor scenes. OAK provides a natural data distribution and the task emergence following the trajectories of a single person. A few objects frequently appear due to redundancy in daily routines, while new objects and categories constantly appear as they visit various places. This dataset is a realistic playground to study online continual learning, enables the embodied agent to learn from a human's experience, and provides a unique opportunity for researchers to pursue the essence of lifelong learning by observing the same person in a long time span.

We introduce several new evaluation metrics in the online continual learning setup. In contrast to the previous task incremental or class incremental settings in continual learning, there is no explicit task boundary in our setup and new tasks emerge following the temporal order in the videos. Therefore, we evaluate the overall performance (continual average precision, CAP), transfer (backward/forward transfer, BWT/FWT), and forgetting (forgetfulness, F) of the models with an additional temporal dimension. We evaluate the models periodically on the frames held out from the same training video frames. The overall performance is aggregated from these evaluations and the transfer/forgetting is defined by the time intervals between the appearances of instances from the same tasks. We adapt several typical continual learning algorithms (e.g., iCaRL [35], EWC [21], Incremental fine-tuning) to object detection and find the performance of these approaches mediocre in the new benchmark, which leaves substantial room for future studies.

2. Related Work

Continual learning benchmarks. A large body of continual learning algorithms [1, 21, 29, 40, 48, 25, 46, 39, 45, 18, 24] have been developed and evaluated on image classification benchmarks such as Permuted MNIST [14], CIFAR-100 [22], and ImageNet [9]. More recently, Lomonaco and Maltoni [28] introduced CORe50, a collection of 50 domestic objects belonging to 10 categories, which supports image classification at object level (50 classes) or at category level (10 classes) and object detection in a recent update. In contrast to our work, the task splits in CORe50 are created manually and the benchmark is used for offline learning.

In the object detection domain, several incremental object detection algorithms [23, 27, 32, 41] adopt existing object detection datasets such as PASCAL VOC [11] and MS COCO [26] for evaluation. They split the categories and train the object detectors on a pre-defined order of the categories sequentially. Chen *et al.* [6] study the problem of continual learning by building NEIL with internet images. Kuznetsova *et al.* [23] extend the evaluation from static images to video snippets like Activity Daily Living (ADL) [33] dataset and YouTube Objects (YTO) datasets [34] for incremental domain adaptation.

The task splits in these benchmarks are often manually determined and the classes in the datasets are carefully balanced. Moreover, existing benchmarks are mostly constructed using static images [9, 14, 22] or samples from short object-centric video snippets (∼15 seconds) [28]. These existing benchmarks may not sufficiently unveil the challenges of building human-like continual learning agents given their artificial settings and static data sources.

Online continual learning settings. There is an emerging line of works on online continual learning [13, 2, 4, 3, 36]. Aljundi *et al.* [3] develop a system that keeps on learning over time in an online fashion, with data distributions gradually changing and without the notion of separate tasks. Ren *et al.* [36] recently extend the standard framework for few-shot learning to an online, continual setting. Similar to Aljundi *et al.*, our online continual learning settings do not have an explicit task boundary. Since the tasks emerge as the person moves around, we introduce new metrics to evaluate the model transfer and forgetting through time.

Ego-centric video recognition. Another related research direction is ego-centric video recognition [38, 12, 20]. Most works focus on developing methods that are suitable for the unique perspective in an ego-centric video using an offline setting [47, 20]. To this sense, the most related work is [10], where they tackle the problem of continual image classification. However, the dataset they use is [28] extremely short and clean, which cannot be sufficient to unveil the challenges in ego-centric video recognition.

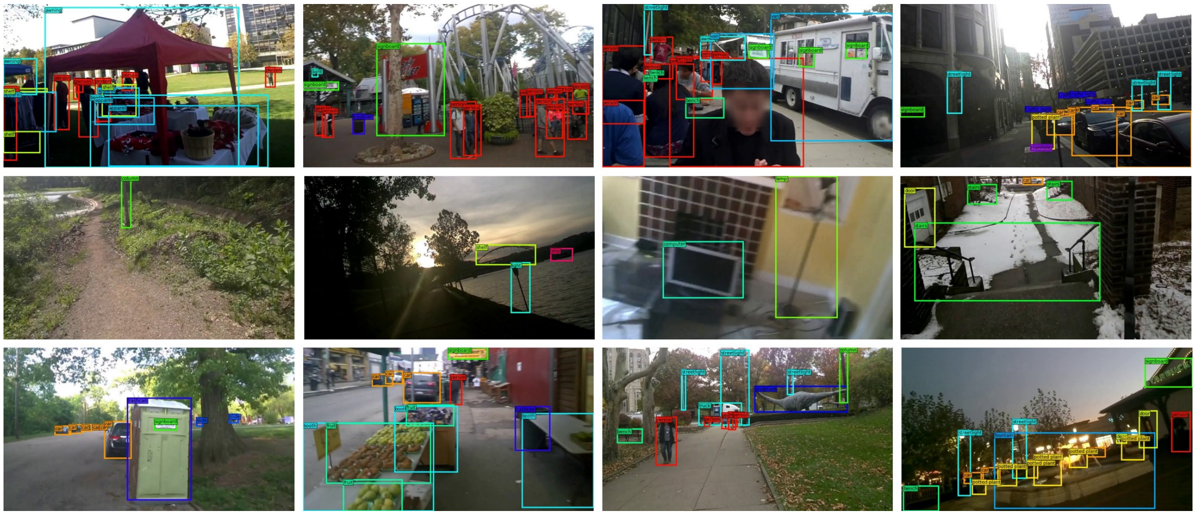


Figure 2: Examples of annotated frames in OAK. OAK captures daily outdoor activities of a single graduate student. The dataset spans a wide variety of environments and life experiences.

3. OAK Dataset

We introduce *Objects Around Krishna* (OAK), which labels the objects that appear as Krishna wanders and performs his daily routine. OAK is built upon the KrishnaCam [43] dataset, a large egocentric video stream spanning nine months of a graduate student's life. The original KrishnaCam dataset contains 7.6 million frames of 460 video snippets with a total length of 70.2 hours. The raw videos have a resolution of 720p and a frame rate of 30 fps. In OAK, we label around 1/4 of the original dataset and release the labeled data in the website.

Annotation setup. We consider 105 object categories in outdoor scenes where 16 classes are from the PASCAL VOC dataset [11] and the remaining categories are frequent classes determined by running the LVIS [15] pre-trained Mask R-CNN model [16] on the raw videos. The full list of the categories is provided in the website. We sample 80 videos from KrishnaCam, which uniformly span in time. Each video snippet is about 7∼15 minutes long. Objects of the 105 categories are exhaustively labeled at a frame rate of 0.5 fps. In total, OAK contains roughly 326K bounding boxes. Two human annotators are involved in labeling each frame to ensure the label quality. Objects in the frames are exhaustively annotated except for the tiny objects (less than $20\times$ 20 pixels). In Figure 2, we show some examples of the annotated frames in OAK.

Train and evaluation sets. In contrast to the offline settings, the model is expected to train while evaluating on the same data stream in an online continual learning setting to evaluate the catastrophic forgetting. Therefore, we hold out one frame every 16 labeled frames to construct an evaluation set and the remaining frames are used for training. The training and the evaluation sets cover a similar time range of 9 months with different sample rates. The models will be trained and evaluated in an online manner.

Dataset statistics. We show some data statistics in Figure 3.

Natural task distribution. In our online continual learning setting, a new task is defined as recognizing a new object category that was not encountered before. As shown in Figure 3a, the number of seen categories gradually increases and the appearance rate of novel classes decreases over time due to the repeated patterns in daily life.

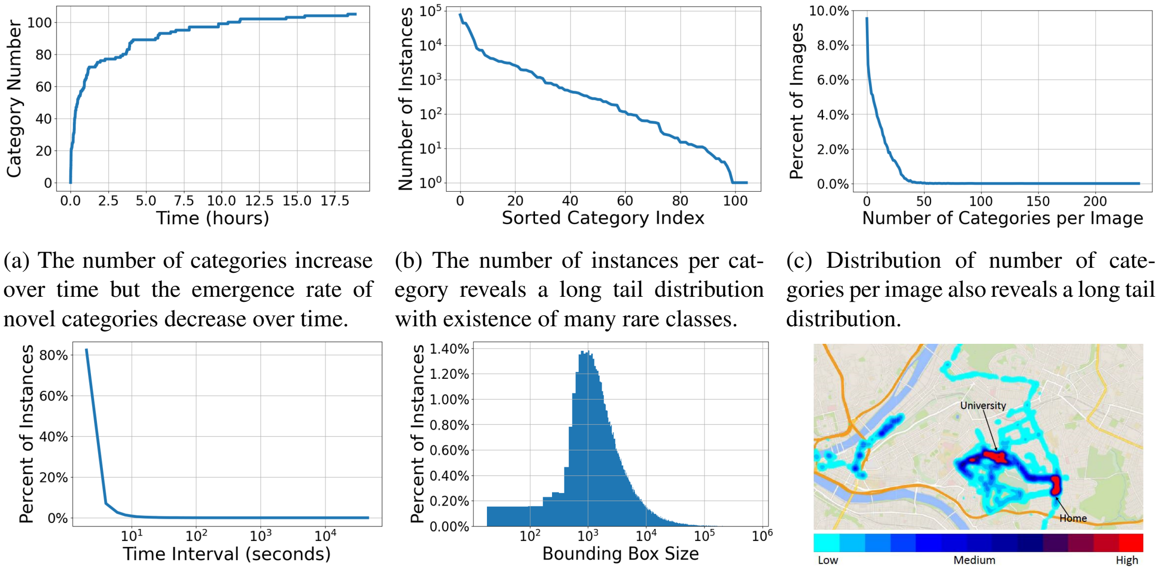
Long-tail distribution. In Figure 3b and 3c, we can see OAK has a long tail distribution both for the number of instances per category and the category counts per image. In Figure 3d, we show the distribution of time intervals between the reappearance of instances from the same category.

Skewed box sizes. As shown in Figure 3e, the box sizes in OAK are skewed towards small and medium sizes, which makes it hard for the detector to make correct predictions.

Diverse geo-locations. In Figure 3f, we plot the locations of the video recordings. We can see that places like campus and home are frequently visited while other places are only occasionally visited.

4. Online Continual Learning Benchmark

In this benchmark, we consider two online continual learning settings depending on whether the categories of interest are known beforehand. In both cases, the training data and labels emerge sequentially following the time stamps and the model is evaluated every N training steps. The main difference is the way that unseen categories are dealt with at the time of evaluation. If the model has a known vocab-



(a) The number of categories increase over time but the emergence rate of novel categories decrease over time.

(b) The number of instances per category reveals a long tail distribution with existence of many rare classes.

(c) Distribution of number of categories per image also reveals a long tail distribution.

(d) Distribution of time interval between the reappearances of data points from the same category.

(e) Distribution of the bounding box size (pixel space). The distribution is skewed to small / medium size objects.

(f) Location distribution. Places in red are often visited and places in blue are occasionally visited.

Figure 3: Dataset statistics. Best viewed digitally.

ulary of the classes referred as the `known` setting, we can simply report the average precision (AP) on each category at each evaluation although some categories may not have been trained at the time of evaluation (often leading to a lower evaluation result).

The case where the model has an open vocabulary of classes, referred as the `unknown` setting, is a bit more challenging but also more realistic. We introduce an IDK (I don't know) class for the unseen categories at the current time stamp. For all the objects from the unseen categories in the evaluation set, the model needs to predict IDK for a correct prediction. The average precision of predicting IDK is also part of the evaluation protocol, which indicates the model's ability to identify new classes. For simplicity, we don't require the model to distinguish the exact category among the unseen categories as long as the model predicts IDK for the unseen objects.

4.1. Evaluation Metrics

In the online continual learning setting, we focus on three aspects of the learned model: how well does the model perform overall? How well does the model transfer new knowledge? How resistant is the model to catastrophic forgetting? To this end, we introduce five evaluation metrics: continual average precision (CAP) and final average precision (FAP) for overall performance evaluation; forward transfer (FWT) for transfer performance evaluation; backward transfer (BWT) and forgetfulness (F) for forgetting performance evaluation. It's worth noting that transfer and forgetfulness are more comparable if two models have similar CAP. We adopt the commonly used AP50 (i.e., the average precision score with an IoU threshold of 50%) in object detection for measurement.

CAP shows the overall performance of the model in the time span of the entire video stream. Inspired by OSAKA [4], the accuracy of each timestep is evaluated using the current model instead of the final model. At each time t, the model is trained using a small batch ($t \times b$-th training frame to $(t+1) \times b$-th training frame, where b denotes batch size) of data from $\mathcal{D}_{\text{train}}$. After time t, this small batch of images is no longer allowed to be used. We continue this training process until the entire video stream is covered. Every N training steps, the model is evaluated on the test set $\mathcal{D}_{\text{test}}$. The reported $\texttt{CAP}_{t_i}$ (i^{th} evaluation step) is defined as

$$\texttt{CAP}_{t_i} = \frac{1}{C}\sum_{c=0}^{C}\texttt{CAP}_{t_i}^{c}, \tag{1}$$

where $\texttt{CAP}_{t_i}^{c}$ is the average precision (AP) of the class c on the test set. $\texttt{CAP}$ is then defined as the average values across different time stamps. That is,

$$\texttt{CAP} = \frac{1}{T}\sum_{i=0}^{T}\texttt{CAP}_{t_i} = \frac{1}{TC}\sum_{i=0}^{T}\sum_{c=0}^{C}\texttt{CAP}_{t_i}^{c}, \tag{2}$$

where T is the total evaluation times.

FAP is the final average precision of the last model when the model finishes training. This is a more fair evaluation metric when comparing with offline learning models as both models have observed the entire video stream.

FWT evaluates the forward transfer ability of new knowledge inspired by GEM [29]. FWT shows the influence that learning a scenario (a video clip, denoted as $\mathcal{S}_\mathrm{t}$) has on the performance for future scenarios ($\mathcal{S}_\mathrm{k}, k > t$). The scenarios are a short clip with a fixed interval of 0.94 hour in our benchmark. Positive forward transfer is possible when the model is able to perform "zero-shot" learning. Specifically, we divide both $\mathcal{D}_\mathrm{train}$ and $\mathcal{D}_\mathrm{test}$ evenly into T scenarios in temporal order, where each division of $\mathcal{D}^i_\mathrm{train}$ and $\mathcal{D}^i_\mathrm{test}$ is used as train set and test set of scenario $\mathcal{S}_\mathrm{i}$.

After the model finishes learning from scenario $\mathcal{S}_\mathrm{i}$, we evaluate its test performance on all T scenarios. By doing so, we construct the matrix $R \in \mathcal{R}^{T\times T}$, where $R_{i,j}$ is the test mean average precision (mAP) of the model on scenario $\mathcal{S}_\mathrm{j}$ after observing the last frame from $\mathcal{S}_\mathrm{i}$. Letting $\bar{b}$ be the vector of test mAP of an pre-trained object detector for each scenario, we define FWT as:

$$\mathtt{FWT} = \frac{1}{T-1}\sum_{i=2}^{T} R_{i-1,i} - \bar{b}_i. \tag{3}$$

BWT shows the influence that learning a scenario (a video clip, denoted as $\mathcal{S}_\mathrm{t}$) has on the performance on previous scenarios ($\mathcal{S}_\mathrm{k}, k < t$). Negative backward transfer is also known as forgetting. Specifically, we define BWT as:

$$\mathtt{BWT} = \frac{1}{T-1}\sum_{i=1}^{T-1} R_{T,i} - R_{i,i}. \tag{4}$$

Forgetfulness (F) estimates the model forgetting due to the sequential training. For a class c, we sort the $\mathtt{CAP}^c_{t_i}$ according to the time interval k between evaluation time t_i and the last time $t_i - k$ the model is trained on c. After the $\mathtt{CAP}^c_{t_i}$ is sorted, all $\mathtt{CAP}^c_{t_i}$ $(i = 0, \dots, T)$ are divided into K bins $B_{kmin}, ..., B_{kmax}$ according to the time interval k. The average $\mathtt{CAP}$ ($\mathtt{aCAP}_k$) of each bin B_k is defined as the model's performance for detecting class c after the model have not been trained on c for k time stamps. We define forgetfulness (F) of the class c as the weighted sum of the performance decrease at each time:

$$\mathtt{F}^c = \sum_{k=kmin}^{kmax} \frac{k - kmin}{\sum_{k=kmin}^{kmax} k - kmin} \times (\mathtt{aCAP}_{kmin} - \mathtt{aCAP}_k). \tag{5}$$

Thus, the overall forgetting is defined as

$$\mathtt{F} = \frac{1}{C}\sum_{c=0}^{C} \mathtt{F}^c. \tag{6}$$

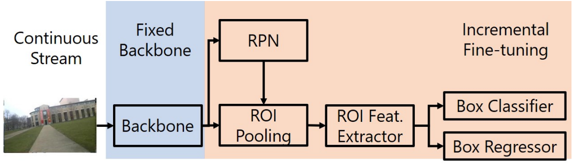


Figure 4: Incremental fine-tuning. The entire object detector, including both the feature extractor $\mathcal{F}$ and the box predictor ($\mathcal{R}$ & $\mathcal{C}$) are pretrained using PASCAL VOC. In online continual learning, the backbone is fixed, while RPN and box predictor are fine-tuned.

5. Experiments

We describe three widely adopted continual learning algorithms in Section 5.1 and present their performance on our benchmark in Section 5.2. Although the existing continual learning algorithms generally improve over the non-adaptive model, the `CAP` value for each of the algorithms is less than 20. This indicates our benchmark is challenging, which has a large room for future algorithm designs.

5.1. Continual Learning Algorithms

We select the three representative continual learning methods according to Parisi *et al.* [31]. Incremental fine-tuning is intuitive and widely adopted as a baseline for continual learning. iCaRL [35] is a widely used memory-based method. EWC [21] is a representative regularization-based method. We first show how each method is deployed to the `known` setting, and we then show that these methods can easily adapt to the `unknown` setting.

Incremental fine-tuning. The first baseline is incremental fine-tuning on the widely used two-stage object detector, Faster R-CNN [37], which is pretrained on the PASCAL VOC dataset. As shown in Figure 4, the feature learning components $\mathcal{F}$ include the backbone (e.g., ResNet [17], VGG16 [42]), the region proposal network (RPN), as well as a two-layer fully-connected (FC) subnetwork as a proposal-level feature extractor. There is also a box predictor composed of a box classifier $\mathcal{C}$ to classify the object categories and a box regressor $\mathcal{R}$ to predict the bounding box coordinates. The backbone features, as well as the RPN features, are class-agnostic. However, we find keeping the RPN updated during fine-tuning significantly improves the performance. Thus, we only keep the backbone fixed during the incremental fine-tuning process. Whenever new data comes in, we fine-tune the RPN, the box classifier, and the regressor on the new data.

iCaRL. The second baseline adapts from the iCaRL [35] algorithm proposed by Rebuffi *et al.* The original iCaRL is designed for image classification and we implement this approach with the Faster R-CNN based detector. As illustrated in Figure 5, iCaRL additionally includes a memory

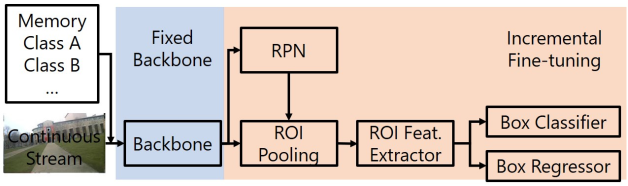


Figure 5: Illustraion of iCaRL. The network parameters are updated by minimizing the loss on both streaming data and memorized prioritized exemplars.

bank $\mathcal{M}$ to store the representative examples of each category, which are selected and updated randomly during each step. The memory bank has a fixed size (we set it to 5 images per category) and the old examples in the memory bank are replaced with newer data points. For each training step, we would randomly select a sample (image with one object label) from each class of the memory bank to train jointly. Same as incremental fine-tuning, we keep the backbone fixed and fine-tune the RPN, the box classifier, and the regressor when new data comes in.

EWC. The third baseline adapts from the EWC algorithm [21], proposed by Kirkpatrick *et al*. Similar to iCaRL, we apply the EWC algorithm to the box classifiers in Faster R-CNN. EWC does not require access to ground truth labels in the memory bank. The main idea of EWC is to impose constraints over the gradient updates so that the gradient updates on the new examples do not increase the classification loss on the old examples as illustrated by Figure 6. Interested readers can refer to the EWC paper [21] for detailed mathematical formulations.

Other baselines. We provide the performance of the Faster R-CNN pretrained on the PASCAL VOC data, denoted as *non-adaptation* in the result tables. We also provide the model performance using offline training. At each evaluation time t_i (i^{th} evaluation step), the entire video stream before t_i are used as one training set. We then conduct batch training offline and report the $\texttt{CAP}_{t_i}$ on $\mathcal{D}_{\text{test}}$. This baseline is denoted as *offline training* in the result tables.

Training details. For a fair comparison, we use ResNet-50 [17] as the backbone of all continual learning algorithms, and the base object detector, Faster R-CNN, is pretrained on PASCAL VOC [11]. OAK share the same 20 categories with the PASCAL VOC dataset and thus the pretrained model can be used as an initial point. In the `known` setting, the number of categories in the detector is fixed and set to 105. In the `unknown` setting, the number of categories increases through time. We add a column to the box classifier and a box regressor when new data contains previous unseen categories. At the prediction time, if the confidence of all classes is smaller than a threshold while the confidence of background also maintains a low status, the agent should predict IDK.

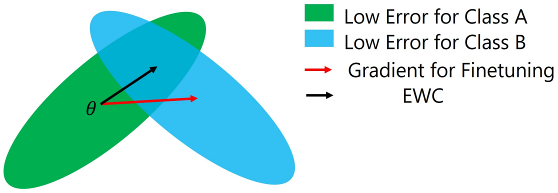


Figure 6: Illustration of EWC (figure adapted from [29]). When learning to predict a new class B, the gradient updates may hurt the performance of the old class A. EWC minimizes the classification loss without incurring a significant loss on the old classes.

5.2. Overall Model Performance

Known class vocabulary. In Table 1, we provide the overall model performance measured by continual average precision (`CAP`) and final average precision (`FAP`) under the `known` setting, where the vocabulary of the classes is known beforehand. The pretrained model on PASCAL VOC (Non-adaptation) only achieves 2.86 points on the new OAK dataset. Even for the classes that overlap with the original PASCAL VOC data (e.g., chair, dining table, etc.), the performance is extremely low. This indicates that OAK has a large domain gap with the existing detection dataset, which uses static Internet images.

For the three continual learning algorithms, the memory-based approach iCaRL outperforms the vanilla incremental fine-tuning. This demonstrates that the simple rehearsal strategy can still play an important role in the new online continual learning setting. However, the regularization-based approach EWC does not help with the overall performance in our setting, which might be due to the complication of the mixed task orders. It is also worth noticing that both EWC and iCaRL have a significant performance gap compared with the offline training. This indicates that the new online continual learning setting may require more innovation in the algorithm designs to deal with the new challenges posed by online learning. Additionally, since both of them are not designed for object detection, the model performance may be improved if one can design models specialized for object detection.

We notice that the model performance of the offline training is only about 49.48 points. It is a relatively low model performance compared with the model performance on other detection datasets such as PASCAL VOC or COCO where the AP50 scores are often above 50 points. This gap is also discovered by other egocentric video challenges [8, 38]. We conjecture the new challenges posed by OAK comes from the intrinsic features of ego-centric videos, such as motion blur, lots of occlusion and partially observed objects due to limited field of view and the long tail distribution.

We visualize the CAP for each class during online con-

Table 1: Overall performance of existing algorithms on OAK measured by continual average precision (CAP) and final average precision (FAP) with known class vocabulary (`known`). FAP is consistently better than CAP, which shows all algorithms benefit from learning from more data. Due to the distribution shift, the non-adaptation model has the lowest performance. The regularization-based EWC approach performs similar to vanilla incremental fine-tuning. While the memory-based iCaRL approach is significantly better than vanilla incremental fine-tuning. The offline training results show that OAK is a bit challenging to the current object detector even under the offline training setting.

Method	FAP	CAP	Top-20	Booth	Umbrella	Awning	Bag	Chair	Dining Table	Fireplug	Car
Non-adaptation	2.86	2.86	11.72	0.00	0.00	0.00	0.00	23.88	26.38	0.0	55.90
Incremental	12.47	10.47	33.84	0.20	2.46	**4.58**	**18.24**	**31.70**	36.51	39.53	69.68
EWC	12.55	10.52	33.80	0.21	3.02	4.53	17.79	31.38	36.67	38.73	**69.71**
iCaRL	**21.89**	**16.39**	**40.92**	**6.88**	**6.91**	3.59	10.67	28.72	**40.59**	**45.18**	67.18
Offline training	49.48	37.11	67.77	20.05	35.86	28.65	42.69	66.51	71.09	64.15	78.80

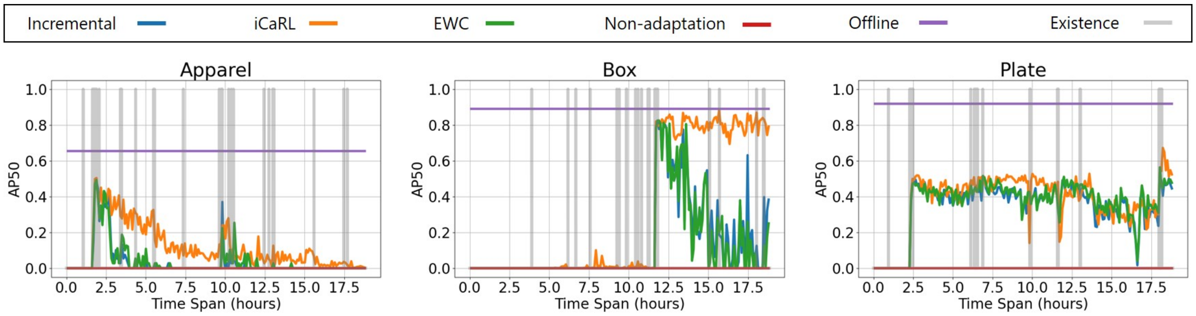


Figure 7: CAP_{t_i} changes on OAK test set, in chronological order. CAP_{t_i} increases as the model is trained on frames where each category exists, but as more time passes between examples, the forgetting effect is observed and the performance falls.

Table 2: Overall performance of the existing algorithms on OAK measured by continual average precision (CAP) and final average precision (FAP) with an unknown class vocabulary (`unknown`). In this evaluation, the model needs to predict IDK for the objects in the unseen categories. All the continual learning algorithms considered have a decreased performance compared with the `known` setting. Again Incremental finetuning and EWC achieve similar performance. iCaRL outperforms incremental fine-tuning and EWC but there is substantial room for improvement.

Method	FAP	Overall	Top-20	Booth	Umbrella	Awning	Bag	Chair	Dining Table	Fireplug	Car	IDK
Non-adaptation	2.86	2.86	11.72	0.00	0.00	0.00	0.00	23.88	26.38	0.0	55.90	-
Incremental	11.19	9.78	32.10	0.06	1.45	**3.87**	**16.68**	29.26	32.81	36.32	68.76	0.32
EWC	10.34	9.57	31.62	0.21	1.63	3.72	16.46	**29.76**	**33.39**	33.41	**68.76**	0.29
iCaRL	**17.56**	**14.70**	**41.42**	**3.58**	**4.83**	2.72	10.18	23.81	32.86	**39.84**	64.02	**0.36**
Offline training	46.40	36.88	71.35	18.90	31.90	29.36	41.36	64.59	70.57	63.77	78.36	1.21

tinual learning. As shown in Figure 7, the shading indicates when the agent observes new data points from a specific class, while the points on each curve are evaluation steps on OAK. Notice how in each class the performance increases initially, but as more time passes between examples, the forgetting effect is observed and the performance falls. When the next round of data is seen towards the end, the performance improves again.

Unknown class vocabulary. In Table 2, we provide the overall performance of the algorithms with an unknown class vocabulary measured by CAP and FAP. In this evaluation, the model needs to predict IDK for the objects in the unseen categories, which is more challenging than the `known` setting. As we can see from the table, all the continual learning algorithms considered have a decreased performance compared with the `known` setting. Again iCaRL outperforms incremental fine-tuning and EWC but there is substantial room for improvement.

5.3. Transfer and Forgetting

Transfer. In the first column of Table 4, we compare the forward transferability of three continual learning al-

Table 3: Forgetfulness of the continual learning algorithms. The forgetfulness (F) metric indicates the catastrophic forgetting and smaller F scores mean less forgetting on the learned knowledge. iCaRL is better at avoiding catastrophic forgetting than the incremental fine-tuning and EWC.

Method	Overall	Worst-20	Booth	Umbrella	Awning	Bag	Chair	Dining Table	Fireplug	Car
Incremental	5.77	22.28	0.60	**1.10**	1.28	-1.95	2.91	**-0.29**	11.89	5.75
EWC	5.96	23.48	0.43	1.85	1.13	**-2.00**	2.76	0.02	10.30	5.76
iCaRL	**1.78**	**14.72**	**-5.65**	2.03	**0.67**	-1.41	**1.50**	0.68	**6.79**	**3.32**

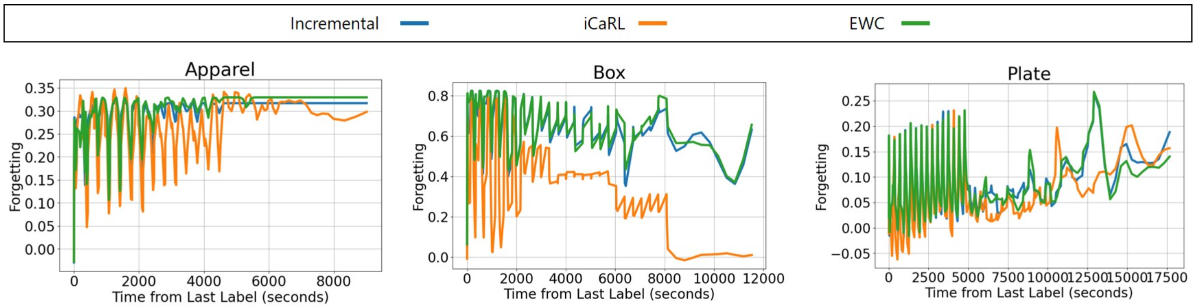


Figure 8: Forgetfulness of sample categories. We provide the evaluation curve on the test set under different time stamps.

Method	FWT	BWT
Incremental	15.01	-4.51
EWC	14.84	**-3.62**
iCaRL	**15.73**	-5.95

Table 4: Knowledge transferability under the known setting. Incremental fine-tuning has better forward transferability and worse backward transferability. This shows that gradient regularization can alleviate catastrophic forgetting at the cost of information gain.

gorithms under the known setting. Higher FWT indicates a faster learner. Incremental fine-tuning has higher FWT compared with EWC. EWC is more conservative about learning from new data points with regularization of the weight change. We don't compare transferability for unknown setting since the tasks keep changing.

Forgetting. In Table 3, we present the forgetfulness of the continual learning algorithms under the known setting. The forgetfulness (F) metric indicates the amount of catastrophic forgetting. Smaller F scores indicate less forgetting on the learned knowledge. iCaRL is significantly better at avoiding catastrophic forgetting than the incremental fine-tuning method and EWC. For more comparison with existing metrics, we also compare the backward transferability of three continual learning algorithms under the known setting (second column of Table 4). Higher BWT indicates less forgetting. EWC has higher BWT than Incremental fine-tuning since EWC finds solutions for new tasks without incurring significant losses on old tasks. We choose forgetfulness (F) as our main metric to estimate the model's forgetting due to the sequential training.

We also visualize the forgetfulness changes of each class during online continual learning. As shown in Figure 8, the forgetfulness increases as the time from seeing the last labeled data increases. Notice that in most cases, the incremental fine-tuning baseline performs best.

6. Conclusion

Online continual learning from continuous data streams in dynamic environments is of an increasing interest in the machine learning and computer vision community. However, realistic datasets and benchmarks, especially for object detection, are still missing. In this work, we presented a new online continual object detection benchmark dataset called OAK. OAK uses the videos from the KrishnaCAM dataset, which features an ego-centric video stream collected over a nine-month time span. OAK provides exhaustive annotations of 80 video snippets (∼17.5 hours) with 326K bounding boxes of 105 object categories in outdoor scenes. Our continual learning benchmark follows the life pattern of a single person, which is more realistic compared to the existing continual learning benchmarks. We introduced new evaluation metrics as well as baseline evaluations for two evaluation setups. We hope this work inspires research in embodied online continual learning of object detectors.

Acknowledgements. The authors would like to thank Yi Ru Wang, Samantha Powers, Kenneth Marino, Shubham Tulsiani for fruitful discussion and detailed feedback on the manuscript. Carnegie Mellon Effort has been supported by DARPA MCS, ONR Young Investigator, ONR muri.

References

[1] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 139–154, 2018. 2

[2] Rahaf Aljundi, Lucas Caccia, Eugene Belilovsky, Massimo Caccia, Min Lin, Laurent Charlin, and Tinne Tuytelaars. Online continual learning with maximally interfered retrieval. *arXiv preprint arXiv:1908.04742*, 2019. 2

[3] Rahaf Aljundi, Klaas Kelchtermans, and Tinne Tuytelaars. Task-free continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11254–11263, 2019. 2

[4] Massimo Caccia, Pau Rodriguez, Oleksiy Ostapenko, Fabrice Normandin, Min Lin, Lucas Caccia, Issam Laradji, Irina Rish, Alexande Lacoste, David Vazquez, et al. Online fast adaptation and knowledge accumulation: a new approach to continual learning. *arXiv preprint arXiv:2003.05856*, 2020. 2, 4

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020. 1

[6] Xinlei Chen, Abhinav Shrivastava, and Abhinav Gupta. Neil: Extracting visual knowledge from web data. In *International Conference on Computer Vision*, 2013. 2

[7] Gong Cheng and Junwei Han. A survey on object detection in optical remote sensing images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 117:11–28, 2016. 1

[8] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736, 2018. 1, 6

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2

[10] Luca Erculiani, Fausto Giunchiglia, and Andrea Passerini. Continual egocentric object recognition. *arXiv preprint arXiv:1912.05029*, 2019. 2

[11] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, Jan. 2015. 2, 3, 6

[12] Alireza Fathi, Xiaofeng Ren, and James M Rehg. Learning to recognize objects in egocentric activities. In *CVPR 2011*, pages 3281–3288. IEEE, 2011. 2

[13] Enrico Fini, Stéphane Lathuilière, Enver Sangineto, Moin Nabi, and Elisa Ricci. Online continual learning under extreme memory constraints. In *European Conference on Computer Vision*, pages 720–735. Springer, 2020. 2

[14] Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*, 2013. 2

[15] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5356–5364, 2019. 1, 3

[16] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 1, 3

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5, 6

[18] Khurram Javed and Martha White. Meta-learning representations for continual learning. *arXiv preprint arXiv:1905.12588*, 2019. 2

[19] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8420–8429, 2019. 1

[20] Georgios Kapidis, Ronald Poppe, Elsbeth Van Dam, Lucas Noldus, and Remco Veltkamp. Egocentric hand track and object-based human action recognition. In *2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, pages 922–929. IEEE, 2019. 2

[21] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 2, 5, 6

[22] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 2

[23] Alina Kuznetsova, Sung Ju Hwang, Bodo Rosenhahn, and Leonid Sigal. Expanding object detector's horizon: Incremental learning framework for object detection in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 28–36, 2015. 2

[24] Soochan Lee, Junsoo Ha, Dongsu Zhang, and Gunhee Kim. A neural dirichlet process mixture model for task-free continual learning. *arXiv preprint arXiv:2001.00689*, 2020. 2

[25] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017. 2

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2

[27] Xialei Liu, Hao Yang, Avinash Ravichandran, Rahul Bhotika, and Stefano Soatto. Multi-task incremental learning for object detection. *arXiv preprint arXiv:2002.05347*, 2020. 2

[28] Vincenzo Lomonaco and Davide Maltoni. Core50: a new dataset and benchmark for continuous object recognition. In *Conference on Robot Learning*, pages 17–26. PMLR, 2017. 2

[29] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6470–6479, 2017. 2, 5, 6

[30] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989. 2

[31] German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019. 5

[32] Juan-Manuel Perez-Rua, Xiatian Zhu, Timothy M Hospedales, and Tao Xiang. Incremental few-shot object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13846–13855, 2020. 2

[33] Hamed Pirsiavash and Deva Ramanan. Detecting activities of daily living in first-person camera views. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2847–2854. IEEE, 2012. 2

[34] Alessandro Prest, Christian Leistner, Javier Civera, Cordelia Schmid, and Vittorio Ferrari. Learning object class detectors from weakly annotated video. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3282–3289. IEEE, 2012. 2

[35] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 2, 5

[36] Mengye Ren, Michael L Iuzzolino, Michael C Mozer, and Richard S Zemel. Wandering within a world: Online contextualized few-shot learning. *arXiv preprint arXiv:2007.04546*, 2020. 2

[37] Shaoqing Ren, Kaiming He, Ross B Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 1, 5

[38] Xiaofeng Ren and Matthai Philipose. Egocentric recognition of handled objects: Benchmark and analysis. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–8. IEEE, 2009. 2, 6

[39] Paul Ruvolo and Eric Eaton. Ella: An efficient lifelong learning algorithm. In *International Conference on Machine Learning*, pages 507–515. PMLR, 2013. 2

[40] Jonathan Schwarz, Wojciech Czarnecki, Jelena Luketina, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. In *International Conference on Machine Learning*, pages 4528–4537. PMLR, 2018. 2

[41] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. Incremental learning of object detectors without catastrophic forgetting. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3400–3409, 2017. 2

[42] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015. 5

[43] K. K. Singh, K. Fatahalian, and A. A. Efros. Krishnacam: Using a longitudinal, single-person, egocentric dataset for scene understanding tasks. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9, 2016. 2, 3

[44] Xin Wang, Thomas Huang, Joseph Gonzalez, Trevor Darrell, and Fisher Yu. Frustratingly simple few-shot object detection. In *International Conference on Machine Learning*, pages 9919–9928. PMLR, 2020. 1

[45] Ju Xu and Zhanxing Zhu. Reinforced continual learning. *arXiv preprint arXiv:1805.12369*, 2018. 2

[46] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. *arXiv preprint arXiv:1708.01547*, 2017. 2

[47] Ye Yuan and Kris Kitani. Ego-pose estimation and forecasting as real-time pd control. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10082–10092, 2019. 2

[48] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pages 3987–3995. PMLR, 2017. 2

[49] Wang Zhou, Shiyu Chang, Norma Sosa, Hendrik Hamann, and David Cox. Lifelong object detection. In *arXiv:2009.01129*, 2020. 2