CONCLAD: COntinuous Novel CLAss Detector

Anonymous Author(s)
Affiliation
Address
email

Abstract

In the field of continual learning, relying on so-called oracles for novelty detection is commonplace albeit unrealistic. This paper introduces CONCLAD ("COntinuous Novel CLAss Detector"), a comprehensive solution to the under-explored problem of continual novel class detection in post-deployment data. At each new task, our approach employs an iterative uncertainty estimation algorithm to differentiate between known and novel class(es) samples, and to further discriminate between the different novel classes themselves. Samples predicted to be from a novel class with high-confidence are automatically pseudo-labeled and used to update our model. Simultaneously, a tiny supervision budget is used to iteratively query ambiguous novel class predictions, which are also used during update. Evaluation across multiple datasets, ablations and experimental settings demonstrate our method's effectiveness at separating novel and old class samples continuously. We will release our code upon acceptance.

1 Introduction and Related Work

Deployed AI models frequently encounter dynamic and evolving data distributions, where continuous model adaptation is paramount to safeguard performance. Reliable novelty detection is a key capability for adaptive AI. Novelty Detection will inform the model if there is new data and if so, which samples are novel and need to be learnt from. However, until now, novelty detection and continual adaptation have been tackled separately within different sub-fields of the AI scientific literature. Most research in continual learning (CL) [1, 2, 3, 4, 5, 6] relies on fully labeled data, despite the significant costs and impracticality of data labeling in real-world scenarios [7]. While there are some unsupervised CL solutions [8, 9, 10], they often rely on an unrealistic assumption: that for each new task and its incoming data, past classes do not appear alongside newly introduced classes, thereby eliminating the need for novelty detection. Removing this oracle assumption results in severe performance degradation due to overconfidence in erroneous predictions [11]: novel classes' samples may be incorrectly predicted to old classes, especially at task transition onset where the continual decision boundaries are still immature. Meanwhile, solutions for novelty or out-of-distribution (OOD) detection [12, 13, 14, 15, 16, 17] have primarily been designed and evaluated using a single, fixed split of old versus novel classes, rather than on continual splits. Additionally, conventional OOD models often lack the ability to continuously integrate and learn from newly detected data. When these models are forced to update, they can suffer from continual error propagation [11]: incorrect novelty predictions during the detection stage lead to incorrect parameter learning during the update stage, progressively degrading the overall system performance. The recently proposed incDFM [11] offers an innovative solution to continual novel class detection (CND). However, incDFM was designed for the simplistic scenario where only one novel class is introduced per task. This strong assumption allows incDFM to treat all samples flagged as novel as members of the single new class, enabling trivial pseudo-labeling for continual update. Due to this unrealistic one-class assumption, incDFM cannot be considered fully unsupervised. In more complex cases with multiple novel classes, incDFM fails to function effectively since it cannot distinguish between different novel classes. In effect, all samples from those multiple novel classes are erroneously assigned to a single new class. This multi-class "collapse" results in a poor estimate of the OOD/novelty distribution and consequently, poor performance. Generalizing to the scenario of continuous multi-class novelties is challenging, necessitating the creation of entirely new algorithmic components. **Our contribution is**

as follows: We propose CONCLAD (COntinuous Novel CLAss Detector), an iterative multi-class uncertainty estimation algorithm designed for generalized Continual Novelty Detection. We utilize the uncertainty scores to select (a) a very small fraction (0.3% - 1.25%) of samples from the unlabeled pool for supervision and (b) a suitable subset of the remaining unlabeled samples for automatic (unsupervised) pseudo-labeling. Through experimentation on various continual tasks and datasets, we demonstrate that CONCLAD excels in continually identifying the presence of (up to multiple) novel classes and accurately separating novel class samples from old ones.

2 Our Method

2.1. Problem Setting: Consider a continual agent $A(x,t)$ which needs to learn/adapt from a set of continual tasks. At each task t, $A(x,t)$ is presented with an initially unlabeled set of samples $U(t)$ 1 which consists in a mixture of unseen samples of its old/learnt classes $U_{old}(t)$ and unseen samples of new (novel) classes $U_{new}(t)$:

$$U(t) = U_{old}(t) \cup U_{new}(t), \text{where } U_{old}(t) = \{x|x \sim \bigcup_{k=1}^{t-1} D_k\}, U_{new}(t) = \{x|x \sim D_t\}, \quad (1)$$

Here D_t comprises samples from the set of new classes C^t_{new} introduced at task t, while $\bigcup_{k=1}^{t-1} D_k$ are samples belonging to all the old classes C^t_{old} that have been learned up to and including task $t-1$. Samples in $U_{old}(t)$ are "unseen", meaning they were never used, neither in the initial training nor during prior tasks' learning. Note that addressing data drifts in U_{old} is beyond the scope of this work.

2.2. Our solution: We introduce a continual novelty detector $N(x,t)$, operating alongside the continual agent, whose goal is to produce a reliable estimate of novel samples $\widehat{U}_{new}(t)$ while simultaneously estimating their respective novel-class labels. Simply performing a binary distinction between novel-class and old-class samples (as in incDFM[11]) leads to poor results in novel multi-class settings. Moreover, the dependence on task index t in $N(x,t)$ indicates that the novelty detector itself has to be continually updated so that novel classes at t are not considered novel at $t+1$. To obtain novel-class labels in $\widehat{U}_{new}(t)$, one can either used unsupervised clustering methods [18], or active supervision (i.e. labeling by an expert) [19, 20, 21]. Here, we share initial results using active supervision for a tiny fraction of $U(t)$ (0.3% - 2.5%), along with pseudo-labeling of confidently identified novel samples in $\widehat{U}_{new}(t)$. For all these tasks – novelty detection, sample selection for active labeling, and for pseudo-labeling – $N(x,t)$ relies foundationally on a novel, iterative multi-class uncertainty estimation method 2 defined and explained in the next sections.

2.2.1. Building block of CONCLAD's uncertainty formulation $S(i)$: CONCLAD's uncertainty estimation 2 uses the *feature reconstruction error* (FRE) [14], which is effective in novelty estimation for the closed-world and the single-class increment CL [11]. FRE involves learning a PCA transform $\mathcal{T}_m$ and its inverse $\mathcal{T}^\dagger_m$ for each class m. A test feature $u = g(x)$ is transformed by $\mathcal{T}_m$ and re-projected back using $\mathcal{T}^\dagger_m$, with FRE calculated as the ℓ_2 norm of the difference between the original and reconstructed vectors. High FRE scores indicate samples that don't belong to class m. In the simplified single-class increment CL [11], a single PCA transform is used for all ID data.

2.2.2. Step by Step Novelty Detection: Prior to deployment (task $t = 0$), we assume that an agent $A(x,t=0)$ has been trained to classify among a fixed set of pre-deployment classes C^0_{new}. Accordingly, CONCLAD's novelty detector $N(x,t=0)$ has been trained to recognize those classes as learnt/old by having computed FRE transforms for those classes, $\mathcal{T}_m, \forall m \in C^0_{new}$. For a given future task $t > 0$, as unlabeled data arrives, $N^{(i)}(x,t)$ follows an iterative procedure (indexed by an inner-loop index, i, which is distinct from outer-loop task-index t) to learn to detect if/what novelties are present. At the first inner iteration $i = 0$, initial supervision querying is performed by picking samples (subject to labeling budget) with high uncertainty scores w.r.t old classes defined as $S^0(u) \triangleq \min_{j \in C^t_{old}} FRE^0_j(u)$. b_0 is sampled uniformly among samples with $S^0(u) >$ mean$(S^0(u))$. At this point, novel classes can be identified (denoted by $|C^t_{new}|$ in section 2.1, assuming $|C^t_{new}| > 0$) and those few labeled samples are used to initialize parameters of $N^{(0)}(x,t)$: (1) Train a single layer perceptron, $N^{(i)}_{pl}(x,t)$ to learn an imperfect initial mapping to the $|C^t_{new}|$ novel classes. This layer, which performs pseudo-labeling (pl), contains output nodes only w.r.t novel classes. (2) compute rough estimates of per-novel-class PCA transforms $\{\mathcal{T}^{t,0}_m\}, m \in C^t_{new}$. Note that it's possible that not all true novel classes are found in this initial iteration and may be found in subsequent ones. For subsequent iterations $i > 0$, given an unlabeled sample $x \in U(t)$, $N^{(i)}_{pl}(x,t)$ predicts a pseudo-label $m, m \in C^t_{new}$ which then routes the selection of the corresponding PCA transform $\mathcal{T}^{t,i-1}_m$ resulting in the i^{textth} iteration's uncertainty score $S^i(x)$ 2:

$$S^i(u) = \min_{j \in C^t_{old}} \frac{FRE^0_j(x)}{FRE^{i-1}_m(x)}; i > 0, m = N^{(i)}_{pl}(x,t) \in C^{new}_t \tag{2}$$

$S^i(x)$ can be used to robustly categorize samples in $U(t)$ as: **(1) Novel with high-confidence:** These are samples with the highest score values (high numerator relative to the denominator). A high value of numerator implies large distance from previously seen classes C^t_{old}, while a low value of the denominator implies low distance from novel class m. Such a sample likely belongs to $U_{new}(t)$ and is a strong candidate to be pseudo-labeled. From these, we select the topmost most confident α percent to pseudo-label. **(2) Old-class with high-confidence:** lowest score values corresponding to low numerator (low distance w.r.t C^{t-1}_{old}) and high denominator value (high-distance from the predicted novel class m). Such a sample likely belongs to $U_{old}(t)$, i.e. to an old class that has already been learned; **(3) Ambiguous:** Samples for which the score is neither definitively high nor definitively low. These could be old-class samples having relatively high scores, or new-class samples having relatively low scores. Owing to this ambiguity, a clear determination cannot be made. Hence, these samples are excellent candidates for active querying to minimize novelty detection uncertainty. At each inner-loop iteration, accumulated active and pseudo-labeled samples are used to re-update $N^{(i+1)}(x,t)$'s parameters (pseudo-labeler $N^{(i)}_{ps}(x,t)$, and FRE transforms $\mathcal{T}^{t,i-1}_m$). At the end of the inner-loop, all accumulated active and pseudo labeled samples are used to compute final PCA transforms $\{\mathcal{T}^t_m\}$ for $m \in C^{new}_t$ to permanently update the novelty detector $N(x,t)$ so those classes are not flagged as novel subsequently. Note that the pseudo-labeler, since it maps only to a given tasks detected novel classes, will be re-initialized at another tasks' onset. Further methodology details, including inner-loop stopping criteria and ambiguity formulation, can be found in appendix sections.

3 Experiments

3.1. Setup: We evaluate on 4 datasets: Imagenet21K-OOD (Im21K-OOD) [22], Eurosat [23], iNaturalist-Plants-20 (Plants) [24] and Cifar100-superclasses [25], all of which were constructed to have no class overlap with Imagenet1K with the exception of Cifar100. Results for Cifar100 are included to enable direct comparison with baseline method incDFM [11]. We compare CONCLAD to: (1) incDFM [11], which first introduced an updatable continual novelty detector, albeit exclusively for single class novelties (see section 1); (2) DFM [26], originally proposed for static novelty detection. We also include semi-supervised CND baselines: (3) Experience-Replay "ER" [27, 6] uses entropy as a measure of novelty similar to [28] and also to select active labels; (3) PseudoER [29], same as ER, but iteratively pseudo-labels the most confident samples akin to CONCLAD. Other baselines are constructed (Fig 2 right table) from removing elements of CONCLAD such as the iterativeness (i.e. doing AL/Pseudo-labeling in one shot), etc. Implementations for CONCLAD and baselines: All use a large/foundation frozen feature extractor, e.g. ResNet50 [30] pre-trained on ImageNet1K via SwAV [31] or ViTs16 [32] pre-trained on Imagenet1K via DINO [33]. CONCLAD's A^{cl}_s (pseudo-labeling head) is a fully connected layer. Baselines ER, PseudoER's long-term classification head is a perceptron of size 4096. For ER and PseudoER we use a fixed replay buffer size containing pre-logit deep-embeddings and labels/pseudo-labels. We set the maximum buffer size to 5000 (2500 for Eurosat). At each incoming unlabeled pool, we fix a mixing ratio of 2:1 of old to new classes per task, with old classes drawn from a holdout set (0.35% of each dataset). For evaluation on the independent test set, we sample old and new classes with the same 2:1 proportion. Note that old classes act as distractors from the point of view of novelty detection. We set pseudo-labeling selection to $\alpha = 20\%$ of samples predicted as novel (appendix 4.1.1). For experiments not purposely varying the tiny supervision budget, we fix a labeling budget of 1.25% for Places, Plants and 0.625% for Eurosat, Im21K-OOD, as guided by Fig 1 *center* which varies the AL budget from 0.625% to 5%.

3.2. Results: We measure continual novelty detection performance with the common "Area Under the Receiver-Operating-Curve" (AUROC) metric. Note that, for fair evaluation, we measure CND on an independent test set with the same ratio of old to new class samples at each task. Fig. 1 (left) displays CND performance (AUROC) over all continual tasks (time) in the case of multi-class novelties per task (5 class increments for Im21K-OOD and 2 class increment for Eurosat). Additionally, figure X (center) shows the sensitivity of CONCLAD and other actively-supervised baselines (ER-entropy, PseudoER-entropy, section 3.1) when varying the tiny supervision budget (tested for a range of 0.32% to 5% of the unlabeled train data at each task). Fig. 1 (right) shows the effect of varying the novel class increment per task as measured by the AUROC score averaged over all continual tasks (with that given increment). Some interesting highlights: (1) we can see in Fig 1 (right) that the compared approach incDFM [11] performs reasonably well for the increment of only one novel class per task, for which it was originally proposed and tested by the authors. However, when the class increment increases, this method degrades in performance because it groups multiple novel classes with no distinction, which hurts detection. (2) PseudoER consistently under-performs ER because it is unable to produce high confidence pseudo-labels to be used in training and this in turn degrades its performance - this highlights the importance of our uncertainty metric 2 in measuring pseudo-label confidence. (3) It is evident in the above plots that even with tiny supervision budgets (e.g. 0.32%-1.25%), CONCLAD consistently outperforms the competing methods by a large margin over the several experimental variations.

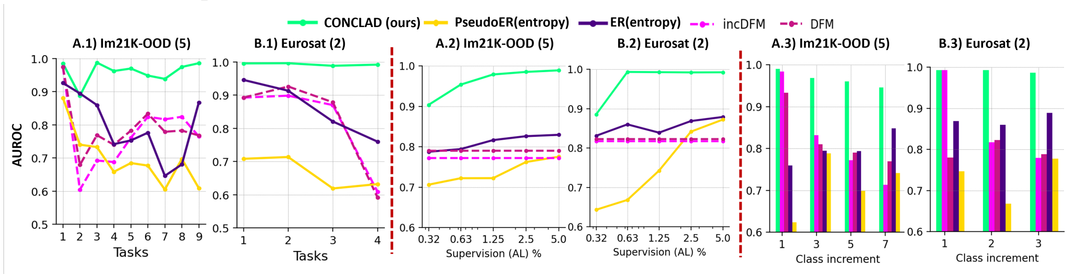

Figure 1: (*Left* A.1,B.1) Continual Novelty Detection performance measured by AUROC at each task. The number of novel classes introduced per task is in parenthesis. Overall, CONCLAD (green) significantly over-performs baselines; (*Center* A.2,B.2) Results varying the supervision budget; (*Right* A.3,B.3) Results varying Novel Class Increment per task. For (left,right) Supervision budget is 0.625% for CONCLAD, ER, PseudoER. Equivalent plots for Cifar100, Plants in appendix 4.3.

	Im21K		Plants		Eurosat		Cifar100	
	R50	ViT	R50	ViT	R50	ViT	R50	ViT
CONCLAD(ours)	**96.0**	**88.0**	**73.6**	**58.8**	**99.3**	**82.3**	**80.6**	**83.3**
incDFM	77.3	76.0	68.7	58.2	81.8	74.9	66.3	66.3
DFM	79.0	75.4	67.4	54.9	82.2	74.9	62.2	66.3
ER-Entropy	79.4	52.8	61.4	52.4	86.0	46.6	64.1	55.4
PseudoER-Entropy	69.9	54.2	60.6	52.5	66.8	46.5	59.5	52.6

Variations (R50)	Im21K	Plants	Eurosat	Cifar100
Default	**96.0**	**73.6**	**99.3**	80.6
Sup-Top	89.5	71.0	97.6	**81.9**
Sup-Rand	88.7	65.9	98.9	80.4
No-Iters	89.5	68.2	80.5	66.4
No-Pseudo	77.2	65.4	75.6	64.1

Figure 2: (Left) Continual Novelty Detection measured by AUROC; (Right) Ablations of CONCLAD. Supervision budget is 0.625% for Im21K, Eurosat and 1.25% for Plants, Cifar100

Fig 2 table (left) shows average AUROC results over all tasks for all 4 datasets and with two different feature extraction backbones (Vision Transformer "ViT" and Resnet50 "R50" described in section 3.1). In sum, similar conclusions can be reached here: CONCLAD significantly overperforms baselines over all the tested settings. Additionally, Fig 2 table (right) shows results for different ablations of CONCLAD: (*No-Pseudo*) Removing Pseudo-labeling from $S(i)$, i.e. computing per-novel class PCAs only with ground-truth label assignments obtained with the tiny labeling budget; (*Sup-Random*) Using random sampling to query ground truth labels with the same tiny budget; *Sup-Top* queries samples with highest uncertainty scores (i.e. most-confidently novel samples) for ground-truth labeling rather than ambiguous samples;(*No-Iters*) CONCLAD in oneshot. Use all supervision budget upfront and then pseudo-label in one-shot. The ablation results highlight the importance of minimizing error propagation via our method's iterativeness since No-Iters results in an average 11.2% decrease in performance. Similarly, we show that pseudo-labeling among the multiple novel classes detected is fundamental to performance given the AL budget's tiny size: No-Pseudo results in 16.8% average decrease. Finally, other active labeling strategies (i.e. Sup-Top) or lack-thereof (Sup-Rand) also decrease performance by 2.4% and 3.9% respectively, underscoring the informativeness of querying ambiguous samples for AL with the goal of continual novelty detection, refer to section 2.2.2.

Key Takeaways: In this work, we presented CONCLAD, a solution to the still under-explored problem of continual novelty detection (CND). Our method enables CND in the generalized setting of novelties containing up to multiple novel classes. To achieve this, CONCLAD includes a foundationally novel iterative multi-class uncertainty estimation procedure capable of effectively modelling the distribution of multiclass novelties, only with a tiny supervision budget. By minimizing the number of samples falsely flagged as novel or overlooked as old, we ensure minimal continual error propagation. Overall, CONCLAD outperforms baselines over multiple large-scale datasets and experimental variations. Yet, several challenges remain for CND, which we hope to address in future work. One nontrivial example is how to detect both novel classes and distribution shifts of old classes (e.g. noise, illumination, etc) together, with minimal to no supervision.

References

[1] James Kirkpatrick et al. "Overcoming catastrophic forgetting in neural networks". In: *Proceedings of the national academy of sciences* 114.13 (2017), pp. 3521–3526.

[2] Shixian Wen et al. "Beneficial Perturbation Network for designing general adaptive artificial intelligence systems". In: *IEEE Transactions on Neural Networks and Learning Systems* (2021).

[3] Brian Cheung et al. "Superposition of many models into one". In: *Advances in neural information processing systems* 32 (2019).

[4] Sylvestre-Alvise Rebuffi et al. "icarl: Incremental classifier and representation learning". In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2017, pp. 2001–2010.

[5] Amanda Rios and Laurent Itti. "Lifelong Learning Without a Task Oracle". In: *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE. 2020, pp. 255–263.

[6] Pietro Buzzega et al. "Rethinking experience replay: a bag of tricks for continual learning". In: *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE. 2021, pp. 2180–2187.

[7] German I Parisi et al. "Continual lifelong learning with neural networks: A review". In: *Neural Networks* 113 (2019), pp. 54–71.

[8] Zhiqi Kang et al. "A soft nearest-neighbor framework for continual semi-supervised learning". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 11868–11877.

[9] Matteo Boschini et al. "Continual semi-supervised learning through contrastive interpolation consistency". In: *Pattern Recognition Letters* 162 (Oct. 2022), pp. 9–14. ISSN: 0167-8655. DOI: 10.1016/j.patrec.2022.08.006. URL: http://dx.doi.org/10.1016/j.patrec.2022.08.006.

[10] Benedikt Bagus, Alexander Gepperth, and Timothée Lesort. *Beyond Supervised Continual Learning: a Review*. 2022. arXiv: 2208.14307 [cs.LG].

[11] Amanda Rios et al. "incdfm: Incremental deep feature modeling for continual novelty detection". In: *European Conference on Computer Vision*. Springer. 2022, pp. 588–604.

[12] Dan Hendrycks and Kevin Gimpel. "A baseline for detecting misclassified and out-of-distribution examples in neural networks". In: (2017).

[13] Shiyu Liang, Yixuan Li, and R Srikant. "Enhancing the reliability of out-of-distribution image detection in neural networks". In: (2018).

[14] Ibrahima Ndiour, Nilesh A Ahuja, and Omesh Tickoo. "Out-Of-Distribution Detection With Subspace Techniques And Probabilistic Modeling Of Features". In: *arXiv preprint arXiv:2012.04250* (2020).

[15] Kimin Lee et al. "A simple unified framework for detecting out-of-distribution samples and adversarial attacks". In: *Advances in Neural Information Processing Systems*. 2018, pp. 7167–7177.

[16] Haoqi Wang et al. "ViM: Out-Of-Distribution with Virtual-logit Matching". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022.

[17] Jie Ren et al. "Likelihood ratios for out-of-distribution detection". In: *Advances in Neural Information Processing Systems*. 2019, pp. 14707–14718.

[18] Yazhou Ren et al. "Deep clustering: A comprehensive survey". In: *IEEE Transactions on Neural Networks and Learning Systems* (2024).

[19] Vu-Linh Nguyen, Mohammad Hossein Shaker, and Eyke Hüllermeier. "How to measure uncertainty in uncertainty sampling for active learning". In: *Machine Learning* 111.1 (2022), pp. 89–122.

[20] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. "Deep bayesian active learning with image data". In: *International Conference on Machine Learning*. PMLR. 2017, pp. 1183–1192.

[21] Donggeun Yoo and In So Kweon. "Learning loss for active learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 93–102.

[22] Tal Ridnik et al. *ImageNet-21K Pretraining for the Masses*. 2021. arXiv: 2104.10972 [cs.CV].

[23] Patrick Helber et al. *EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification*. 2019. arXiv: 1709.00029 [cs.CV].

[24] Grant Van Horn et al. *The iNaturalist Species Classification and Detection Dataset*. 2018. arXiv: 1707.06642 [cs.CV].

[25] Alex Krizhevsky. "Learning Multiple Layers of Features from Tiny Images". In: *University of Toronto* (May 2012).

[26] Ibrahima J Ndiour, Nilesh A Ahuja, and Omesh Tickoo. "Subspace Modeling for Fast Out-Of-Distribution and Anomaly Detection". In: *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2022, pp. 3041–3045.

[27] David Rolnick et al. "Experience replay for continual learning". In: *Advances in Neural Information Processing Systems* 32 (2019).

[28] Rahaf Aljundi et al. "Continual novelty detection". In: *Conference on Lifelong Learning Agents*. PMLR. 2022, pp. 1004–1025.

[29] Dong-Hyun Lee. "Pseudo-Label : The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks". In: *ICML 2013 Workshop : Challenges in Representation Learning (WREPL)* (July 2013).

[30] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[31] Mathilde Caron et al. "Unsupervised learning of visual features by contrasting cluster assignments". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 9912–9924.

[32] Dosovitskiy Alexey. "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv preprint arXiv: 2010.11929* (2020).

[33] Mathilde Caron et al. "Emerging properties in self-supervised vision transformers". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 9650–9660.

[34] Yen-Chang Hsu et al. "Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 10951–10960.

[35] Utku Evci et al. "Head2toe: Utilizing intermediate representations for better transfer learning". In: *International Conference on Machine Learning*. PMLR. 2022, pp. 6009–6033.

[36] Alexander A Petrov, Barbara Anne Dosher, and Zhong-Lin Lu. "The dynamics of perceptual learning: an incremental reweighting model." In: *Psychological review* 112.4 (2005), p. 715.

[37] Guneet S Dhillon et al. "A baseline for few-shot image classification". In: *arXiv preprint arXiv:1909.02729* (2019).

[38] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[39] Amanda Rios and Laurent Itti. "Closed-loop memory GAN for continual learning". In: *arXiv preprint arXiv:1811.01146* (2018).

[40] Olga Russakovsky et al. "ImageNet Large Scale Visual Recognition Challenge". English (US). In: *International Journal of Computer Vision* 115.3 (Dec. 2015). Publisher Copyright: © 2015, Springer Science+Business Media New York., pp. 211–252. ISSN: 0920-5691. DOI: 10.1007/s11263-015-0816-y.

[41] Pengzhen Ren et al. "A survey of deep active learning". In: *ACM computing surveys (CSUR)* 54.9 (2021), pp. 1–40.

4 Appendix

4.1 Methodology Details

4.1.1 Thresholds for Stopping the inner-loop

The inner-loop is guided by two simple thresholds: (1) Threshold T_{inner} "roughly" estimates if there are any possible novel-class samples in the unlabeled task input data pool and is controlled by a single hyper-parameter, the number of standard deviations above the mean of an in-distribution validation set (2 STDs in our experiments). If no samples are found to be above T_{inner}, we reach the stopping criterion for our iterations. Our in-distribution validation set is conventionally defined to include a portion (0.1%) of the previous tasks' $k = 1 : t - 1$ novelty predictions that were held-out at previous tasks, i.e. not used to update $N(x, t)$ parameters. Importantly, the same in-distribution validation set is used for all compared baselines in our results section, as is common practice in the OOD/novelty-detection literature [11, 34]; (2) Finally, Threshold α tunes pseudo-labeling selection and is set to $\alpha = 20\%$ highest $S^i(u)$ scores (most confident) from the test samples found above T_{inner}. These two thresholds are not highly sensitive.

4.1.2 How to define Ambiguity

CONCLAD seeks to minimize novelty-detection uncertainty and model multiclass-novelties by selecting the most novel-vs-old ambiguous samples at each inner-loop iteration, i.e. scores $S^i(u)$ which are neither too high or too low. Our mathematical formulation uses the threshold T_{inner} defined in the previous section: we formulate ambiguousness as the inverse squared distance $\frac{1}{\|S^i(u)-T_{inner}\|^2}$ of scores to T_{inner}. Intuitively, this formula favors selecting samples that cannot be unambiguously predicted as either old or new since T_{inner} represents this rough decision boundary. Active selection is stopped when the tiny labelling budget is exhausted. The only exception to this Ambiguity formulation is at the first iteration $i = 0$ where we select homogeneously from samples above T_{inner}. This is the case because at $i = 0$ only old classes are used to compute the score function, $S^0(u) = \min_{j \in C^t_{old}} FRE^0_j(u)$ and so ambiguity cannot be defined in the same way as for the remainder of iterations.

4.1.3 Measuring per-class uncertainty in CONCLAD's $S^i(u)$ formulation

CONCLAD is agnostic to the elemental uncertainty metric used in its uncertainty scoring function ($S^i(u)$ Eq. 2 in section 2) as long as it can reliably estimate uncertainty w.r.t each novel class or old class. However, this is not an easy feat since many existing static uncertainty quantification approaches are not fully reliable [14, 11]. As discussed in the main text, CONCLAD currently leverages the *feature reconstruction error* (FRE) metric introduced in [14] to build Eq 2. For each in-distribution class, FRE learns a PCA (principal component analysis) transform $\{\mathcal{T}_m\}$ that maps high-dimensional features u from a pre-trained deep-neural-network backbone $g(x)$ onto lower-dimensional subspaces. During inference, a test-feature $u = g(x)$ is first transformed into a lower-dimensional subspace by applying $\mathcal{T}_m$ and then re-projected back into the original higher dimensional space via the inverse $\mathcal{T}_m^\dagger$. The FRE measure is calculated as the ℓ_2 norm of the difference between the original and reconstructed vectors:

$$FRE_m(u) = \|f(x) - (\mathcal{T}_m^\dagger \circ \mathcal{T}_m)u\|_2. \tag{3}$$

Intuitively, FRE_m measures the distance of a test-feature to the distribution of features from class m. If a sample does not belong to the same distribution as that mth class, it will usually result in a large reconstruction score FRE_m. FRE is particularly well suited for the continual setting since for each new class discovered at test-time, an additional principle component analysis (PCA) transform can be trained without disturbing the ones learnt for previous classes.

4.2 Experimental Methodology Details

4.2.1 Implementation Details for CONCLAD and Baselines

$N(x, t)$ operates on top of a large-scale/foundation models as feature extraction backbones, kept frozen throughout CONCLAD and baselines' training: (1) Most results use ResNet50 [30] unsupervisedly pre-trained on ImageNet1K via SwAV [31]. We extract features from the pre-logit AvgPool layer of size 2048 as deep-embeddings. We also experimented with other feature extraction points [14] but those under-performed w.r.t the pre-logit layer. (2) We also show results using ViTs16 [32] pretrained on Imagenet1K via DINO [33]. For ViTs16 we tried several extraction points, e.g. head, last norm later, different transformer block outputs with different pool factors (e.g. 2,4). Best results were obtained with the norm layer. Note that learning on frozen deep features is commonplace in vision CL and domain-adaptation fields [5, 11, 35]. It is theoretically based on the principle that low-level visual features from a large-scale/foundation frozen model are task nonspecific and do not need to be constantly re-learned. Rather, learning may happen upstream by utilizing the extracted deep features (at the last or inner-layers, or a combination thereof - an active research area) [36, 37, 35]. CONCLAD's $N(x, t)$ fully-connected pseudo-labeling layer is trained with ADAM [38], learning rate of 0.001, mini-batch of 10 and an average of 5 epochs at each inner-loop. We experimented with other possibilities of pseudo-labeler such as a 1-layer perceptron but obtained marginal performance gain. Baselines' ER and PseudoER long term classification head are implemented as a one layer perceptron of size 4096 (also tested variations with marginal variations in results). The ER/PseudoER replay buffer is set to a size 5000 deep-embeddings for Plants [24], Imagenet21K-OOD [22] and 2500 for eurosat and cifar100. We use a fixed-size memory buffer B_t with the same building strategy and training loss as in [39]: a buffer of fixed size and prioritizing homogeneous distribution among classes. That is, an equivalent number of samples of each class are removed if room is required for new classes and the buffer is full. Equal weight is given to old and new classes during ER. Lastly, baselines incDFM [11] and DFM [14] were trained using same hyper-parameters proposed by the authors and their open-source code.

4.2.2 Datasets:

Since the employed large/foundation feature extractor were pretrained on Imagenet1K, we evaluate CONCLAD on datasets that either do not contain class overlap with Imagenet1K (out-of-distribution w.r.t Imagenet1K [40]), or curated them by excluding any overlapping classes. The exception is cifar100, which was included due to it being a very popular and widespread dataset, also used in incDFM [11].

1. *Imagenet21K-OOD (Im21K-OOD) [22]*: We curated a subset of Imagenet21K containing the top-most populous 50 classes and that do not overlap with the classes present in Imagenet1K. We use a random set of 500 samples from each of the 50 classes. Because Imagenet21K is a superset of Imagenet1K, by excluding any overlapping class we guarantee orthogonality in our curated subset. We will release the full list of images chosen in this curation for reproducibility.
2. *iNaturalist-Plants-20 (Plants) [24]*: is a curated subset containing images from 20 OOD plant species, sourced from the iNaturalist project [24]. A super-set (larger) version of this subset was originally proposed by [**huang2021mos**] and has since been frequently used as test OOD dataset with respect to Imagenet1K [**xia2022usefulness**, **ming2022delving**]. Note that we use only 20 classes instead of the original 110 in the [**huang2021mos**] super-set since we remove classes with sample count below 140.
3. *Eurosat [23]*: An RGB dataset of 10 classes and 27K images of Sentinel-2 satellite images, which is also orthogonal to Imagenet1K.
4. *Cifar100-Superclasses (Cifar100) [25]*: We use the super-label granularity of Cifar100 dataset. This totals 20 labels (super) and 50K images. While Cifar100 is not orthogonal to Imagenet1K, we decided to showcase its results since it is a widespread dataset in CL.

4.2.3 Baselines

For continual novelty detection (CND), we include unsupervised baselines that also utilize FRE-based uncertainty measures: DFM [26] and incDFM [11]. The latter, incDFM [11], was the first to develop an updatable continual novelty detector for CND, albeit exclusively tested for the trivialized case of single class novelties only, see discussion in main paper section 1. Alternatively, DFM originally introduced the FRE measure 3 for static novelty detection. In the case of incDFM, their proposed scoring function after training/update could be directly used to compute novelty detection on a test set, in the continual setting. We use the author's official implementation of incDFM to generate results. For DFM, we adapted the method to the continual setting by storing one PCA transform $\mathcal{T}_j$ per task trained from all data predicted as novel at the previous task. The scoring function S^t_{DFM} for DFM is defined in equation 4, with T^t_{old} representing the count of how many past tasks with novelty(ies) have previously occured at time/task t.

$$S^t_{DFM}(u) = \min_{j \in T^t_{old}} FRE_j(u) \tag{4}$$

We also include semi-supervised baselines, with the same tiny supervision budget: (2) ER [27, 6], originally proposed for supervised CL is adapted to only use actively labeled samples (as embeddings) for replay; (3) We also adapt PseudoER [29] similar to ER but further incorporating pseudo-labeling of high confidence unlabeled samples for training. In both ER and PseudoER, we utilize the cumulative classification entropy as an uncertainty score to actively-label and Pseudo-Label (PseudoER). Similar to CONCLAD, we actively label ”ambiguous” samples according to the same formula as outlined in appendix 4.1.2 for superior results, then sampling according to the TOP heuristic (see section 3 discussion). We also tested with other common uncertainty metrics such as margin [41] but with inferior results.

4.3 Additional Results

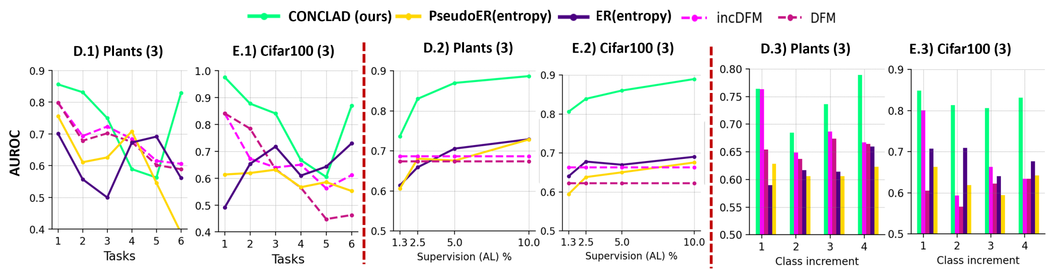

Figure 3: Results for Plants and Cifar100; (*Left* D.1,E.1) Continual Novelty Detection performance measured by AUROC at each task. The number of novel classes introduced per task is in parenthesis.(*Center* D.2,E.2) Results varying the supervision budget; (*Right* D.3,E.3) Results varying Novel Class Increment per task. For (left,right) Supervision budget is 1.25% for CONCLAD, ER, PseudoER. Overall, CONCLAD (green) significantly over-performs baselines