CONTINUAL ANCHORED MANIFOLD EMBEDDINGS FOR LEARNING STABILITY (CAMELS)

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce **CAMELS**, a continual learning framework that leverages metric space constraints in latent space to preserve stable representations over time. Rather than constraining parameters or matching global prototypes, CAMELS anchors the internal structure of past tasks by preserving pairwise cosine similarities among replay samples—maintaining relative geometry without freezing coordinates and embeds different tasks in orthogonal subspaces. This formulation treats continual learning as the problem of preserving local isometries across evolving latent manifolds in high-dimensional embedding spaces. We provide theoretical guarantees that our approach bounds forgetting and classification risk by maintaining manifold consistency and prototype stability. Empirically, CAMELS outperforms or matches prior methods on standard benchmarks including Split, Rotated, and Permuted MNIST, as well as Split CIFAR-10. The resulting latent space is highly interpretable, revealing clear task and class structure that evolves throughout training. These results highlight the value of geometric structure preservation as a principled approach to learning stable, adaptable representations in sequential settings.

1 INTRODUCTION

Deep neural networks trained sequentially on multiple tasks suffer from catastrophic forgetting: new gradients overwrite weights that were important for past tasks (Kirkpatrick et al., 2017). Replay buffers alleviate forgetting by interleaving samples from previous tasks with current data, but how to use those samples most effectively remains open. Many recent works regularize the classifier logits (Li & Hoiem, 2016; Lopez-Paz & Ranzato, 2017; Rebuffi et al., 2017) or the parameter space (Aljundi et al., 2018; Chaudhry et al., 2019; 2018). By contrast, representation-centric approaches match either class prototypes (Asadi et al., 2023), topological summaries (Fan et al., 2024), or pair-wise similarities (Cha et al., 2021; Yu et al., 2023). Inspired by the observation that relative relationships carry more transferable information than absolute coordinates (Moschella et al., 2023), we propose to preserve only the intra-task pair-wise distances and to decouple different tasks through an online, batch-specific orthogonality constraint.

1.1 CONTRIBUTIONS

We develop a new representation learning-based approach to continual learning, **CAMELS**, whose novelties are threefold:

1. *Masked Manifold Anchoring*: An ℓ_2 loss on cosine distances that is applied exclusively to pairs drawn from the same task, avoiding spurious constraints across unrelated tasks
2. *Orthogonality Penalty*: We replace global sub-space regularisers with a per-batch orthogonality penalty: for each replay mini-batch we derive an orthonormal basis via QR factorisation and minimise the squared projection of current embeddings onto this basis
3. *Elegent Interpretability*: Our model lends itself to elegant interpretations and visualizations of the multi-task joint latent space following the continual learning stream.

We demonstrate that this simple combination—no teacher network, no task IDs at inference—achieves competitive and improved performance on both Permuted- and Split-MNIST while using the same fixed memory footprint as earlier methods such as GEM (Lopez-Paz & Ranzato, 2017) and A-GEM (Chaudhry et al., 2019).

1.2 Problem Formulation

Let $\{\mathcal{T}_1, \ldots, \mathcal{T}_T\}$ be a sequence of tasks, where each task $\mathcal{T}_t$ provides labeled samples $\{(x_i^{(t)}, y_i^{(t)})\}_{i=1}^{N_t}$. In *domain-incremental CL*, all tasks share the same label set but differ in input distribution, whereas in *class-incremental CL*, new classes are introduced sequentially throughout training (Lopez-Paz & Ranzato, 2017; Rebuffi et al., 2017). Define the *forgetting* on task i as $F_i = \max_{1 \le k \le i} A_{k,i} - A_{T,i}$, where $A_{k,i}$ is the accuracy on task i immediately after learning task k. The average forgetting is $\mathcal{F} = \frac{1}{T-1} \sum_{i=1}^{T-1} F_i$. We aim to find a sequence of parameters $\{\theta_t\}_{t=1}^{T}$ and a bounded memory buffer $\{\mathcal{M}_t\}_{t=1}^{T}$ that minimizes $\mathcal{F}$ while maintaining high accuracy throughout continual learning (Xu et al., 2023). Crucially, our model does not have access to task IDs (task incremental continual learning) and instead must learn them from shifts in the data distribution.

2 Related Work

Continual learning methods can be broadly grouped into parameter- and replay-based approaches. Parameter constraints mitigate forgetting by penalizing updates along important directions, as in Elastic Weight Consolidation (EWC)(Kirkpatrick et al., 2017) and its Riemannian Walk (RWalk) extension using KL-based sensitivity scores(Chaudhry et al., 2018), or through online importance estimation in Memory-Aware Synapses (MAS)(Aljundi et al., 2018). Replay methods enforce consistency with past data: Gradient Episodic Memory (GEM) projects gradients to avoid increasing loss on stored samples(Lopez-Paz & Ranzato, 2017), A-GEM reduces this overhead via approximate projections (Chaudhry et al., 2019), Dynamic Experience Replay (DER) selects only the most salient samples (Luo & Li, 2020), and Strong Experience Replay (SER) augments stored examples with a consistency loss to stabilize updates (Zhuo et al., 2023).

Representation-driven techniques focus on preserving or adapting the feature space. iCaRL maintains class exemplars and applies logit distillation to balance old and new classes (Rebuffi et al., 2017), while prototype–sample relation distillation pushes embeddings toward stored prototypes (Asadi et al., 2023). Contrastive objectives—employed by SCALE (Yu et al., 2023), Co²L (Cha et al., 2021), and supervised contrastive loss (SupCon)(Khosla et al., 2020)—group semantically similar examples, and persistent-homology distillation preserves the embedding manifold's topological structure(Fan et al., 2024). In contrast, our method anchors only masked pairwise distances and enforces per-batch orthogonality, eliminating the need for a global prototype bank or an external teacher network.

2.1 Biological Motivation

We base the underlying motivation of our latent space geometry constraints off of the concept of neural manifolds, wherein the brain represents distinct tasks and cognitive variables on low-dimensional neural manifolds that are rotated and pushed into nearly orthogonal subspaces, thereby minimizing interference and preserving representations over time. We review literature on neural manifolds and neural task representations as a biological motivation for Camels.

Population recordings in motor cortex reveal that **neural activity during reaching and grasping lies on a smooth, low-dimensional manifold capturing the bulk of response variance** (Gallego et al., 2017; Gallego, 2018). Across different movement types, these manifolds share a common geometry but can be rotated in state space to encode task-specific dynamics (Sabatini & Kaufman, 2024). Similarly, prefrontal circuits form distinct manifolds for different cognitive rules, with task contexts encoded in nearly orthogonal subspaces to prevent cross-talk (Timo Flesch, 2022; Hajnal et al., 2023).

Studies in parietal and prefrontal cortex show that **multiplexed representations of sensory inputs, motor plans, and contextual variables occupy separable, orthogonal dimensions within the same population** (Hajnal et al., 2023; McFayden et al., 2022). Hippocampal place-cell ensembles likewise remap into distinct smemi-orthogonal subspaces for different environments, maintaining stable spatial codes with minimal overlap (L Bashford, 2023) or, more formally, avoiding interference. These findings directly parallel our *per-batch orthogonality* penalty, which projects new-task embeddings away from previously occupied subspaces.

Even as task demands shift, the **relative geometry—pairwise angles and distances—among neural trajectories remains remarkably stable**, supporting consistent decoding and generalization (Pereira-Obilinovic et al., 2024). Population-level isometries have been documented in auditory cortex, where phonetic categories preserve their angular relationships across speaker changes, and in sensory areas that maintain metric structure under context switches (Pereira-Obilinovic et al., 2024; W. Jeffrey Johnston, 2024).

The hippocampus replays compressed trajectories along neural manifolds during rest and sleep, effectively anchoring memory representations and preventing forgetting (Avitan & Stringer, 2022). This biological replay

echoes our *manifold anchoring* via replay-buffer distances, suggesting that aligned latent geometries are a universal strategy for continual adaptation without interference.

Hence, our combined metric space objective of *intra*-task manifold anchoring and *inter*-task orthogonality is motivated from the geometry of "task neural manifolds" existent in the prefontal cortex, providing a neurological analogy for our latent space design.

3 Setup and Theoretical Result

Our model is designed to enforce the two above metric space constraints of isometric task learning and orthogonal task subspaces. We design a latent encoder architecture and training scheme that promotes generality of downstream use for the latent embeddings while following our geometric constraints.

3.1 Training Scheme

We retain a classic encoder backbone that down-projects our input to the latent space. For our experiments, we need a downstream model that maps our latent vectors to logits spanning N classes, noting that our similarity-conditioned latent representation can be "plugged in" to a potentially multimodal architecture, such as a classifier, generative decoder, or other. Our training algorithm can be summarized as follows:

We train our model by iteratively introducing new tasks as new datasets, each task being trained for M epochs. Each batch, we train in parallel on a replay minibatch that incorporates standard CE loss, manifold anchoring loss, and orthogonality loss.

Training objective (new task t). Consider a current-task batch $(\mathbf{x}_c, y_c)$ and a replay batch $(\mathbf{x}_r, y_r, \mathbf{z}^{\text{ref}})$, from which we get our latent embedding batches through passing these batches through our encoder $E_\theta : z_c = E_\theta(\mathbf{x}_c), z_r = E_\theta(\mathbf{x}_r)$ (note z_r and z_c are both matrices with first dimension as batch). Then, we define our pairwise similarity metric $D(Z)_{ij} = d(Z_i, Z_j)$ for a matrix of embedding vectors Z. Anchoring to z^{ref}, the matrix of "expert model"'s embeddings on x_r, we minimize the following loss (a complete explanation of loss terms can be found in the Appendix):

$$\mathcal{L} = \underbrace{\mathrm{CE}\big(h(\mathbf{x}_c), y_c\big)}_{\text{current}} + \underbrace{\mathrm{CE}\big(h(\mathbf{x}_r), y_r\big)}_{\text{replay}} + \lambda_{\text{man}} \underbrace{\frac{1}{B^2}\big\|D(\mathbf{z}_r) - D(\mathbf{z}^{\text{ref}})\big\|_F^2}_{\text{manifold}} + \lambda_{\text{orth}} \underbrace{\frac{1}{B}\sum_{i=1}^{B}\big\|Q_{\text{prev}}^\top \mathbf{z}_{r,i}\big\|_2^2}_{\text{orthogonal}}$$

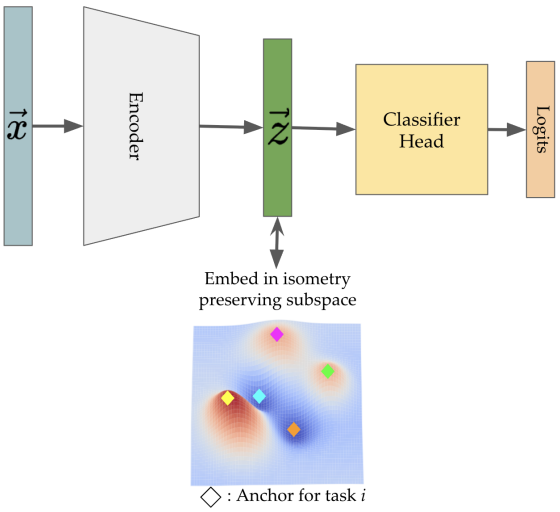


Figure 1: Diagram of Latent Model Architecture: we utilize a standard latent-classifier architecture with a loss trained on the metric structure of the encoder's output. In general, we choose classification as our representative task: the head can be replaced with any model accepting latent vectors as input.

3.2 CENTRAL THEOREM

The effectiveness of CAMELS is grounded in a geometric argument that isometric transformations of finite points in high dimensional space lead to minimal loss in distance-based classifier risk. Our main theoretical result (proved in Appendix B) bounds the total forgetting and empirical risk of our model in the strict continual learning setting given a fixed point of our multi-loss training objective.

Theorem 1 (Bounded forgetting and average risk)**.** *Assume tasks arrive sequentially, each satisfies a Johnson–Lindenstrauss (JL) embedding with distortion $\varepsilon < 1$ (Johnson & Lindenstrauss, 1984), and training reaches a stationary point where $\mathcal{L}_{\text{man}} = \mathcal{L}_{\text{orth}} = 0$. Then for every past task $t < T$,* ***(Isometry)*** *$\mathcal{M}_t$ is preserved up to ε-distortion,* ***(Classifier invariance)*** *Prototype distances $d_{\cos(w_y, \Phi(x))}$ are unchanged (Lemma 5),* ***(Risk bound)*** $\left|\mathcal{R}_t^{after} - \mathcal{R}_t^{before}\right| \leq C \cdot \varepsilon$ *for some constant C (Lopez-Paz & Ranzato, 2017; Fan et al., 2024),* ***(Average-risk guarantee)*** $\frac{1}{T}\sum_{t<T}\left(\mathcal{R}_t^{after} - \mathcal{R}_t^{before}\right) \leq D\varepsilon$. *for some constant D.*

In the following section, we gain more intuition for the connection between these latent space geometric concepts of task subspaces and our continual learning metrics of average empirical risk and catastrophic forgetting.

INTUITIVE THEORETICAL SKETCH OF CAMELS

We first provide a toy visualization in $\mathbb{R}^3$ of how an isometric transformation of previous tasks due to the addition of a new task manifold in our training scheme does not change a distance-based classification method.

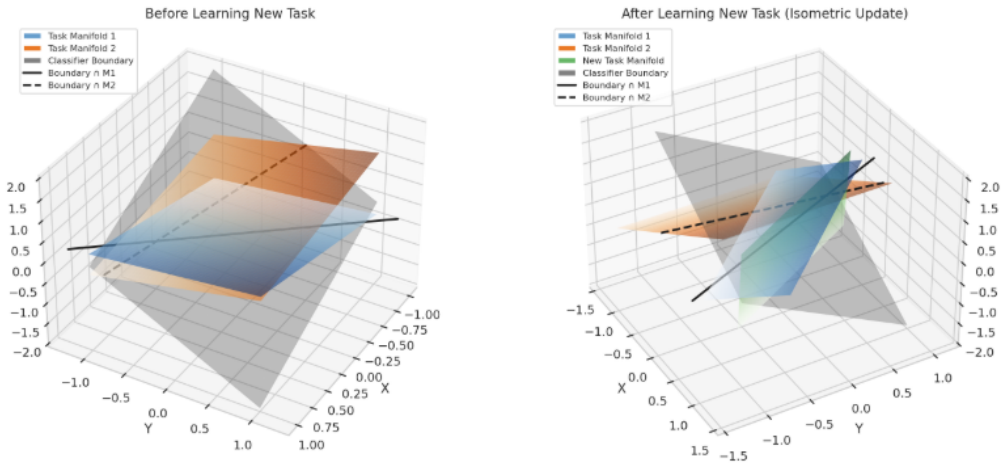


Figure 2: As a toy example, we take three sheets in $\mathbb{R}^3$ to represent task manifolds of a 3-task continual learning scheme. Specifically, task manifold 1 is represented by the blue plane, task manifold 2 is represented by the orange plane. Our classifier boundary is a linear decision boundary defined by the plane $x + y + z = 0$, whose intersection on each task manifold governs the classification behavior on that task. We see these intersections as the solid line (intersection with task manifold 1), and the dashed line (intersection with task manifold 2). The latent space after "training on task 3" is represented in the right plot, where now we have the inclusion of the green plane of task manifold 3, and task manifold 1 and 2 have both been isometrically rotated. However, what matters is the intersection of the classifier decision boundary on their planes, which after the transformation remains the exact same (once again shown through the solid and dashed lines). In this toy example, we clearly see that isometric transformations on old orthogonal tasks retain our classification performance on them, a geometric argument against catastrophic forgetting.

The full proof of our theorem is deferred to the appendix, but we offer an intuitive guide to how these losses constructively create our desired joint task latent space.

1. Using JL, we show that our isometry on finite points extends to an isometry on continuous space. At a stationary point with $\mathcal{L}_{\text{man}} = 0$, every pair of replayed embeddings $\{z_i, z_j\}$ satisfies

$$(1-\varepsilon)\,\|z_i - z_j\|_2^2 \;\leq\; \|\Phi_{\theta^{\text{new}}}(z_i) - \Phi_{\theta^{\text{new}}}(z_j)\|_2^2 \;\leq\; (1+\varepsilon)\,\|z_i - z_j\|_2^2$$

by the Johnson–Lindenstrauss lemma (Johnson & Lindenstrauss, 1984). Thus intra-task distances—and hence angles—are "locked in" up to a factor $(1 \pm \varepsilon)$.

2. Our isometry guarantee fixes the distance between class prototypes (centroids of embedding subspaces corresponding to our classifier decision regions). Since each class prototype w_y is just the mean of unit-norm embeddings, preserving all pairwise distances forces $\|w_y^{\text{new}} - w_y^{\text{old}}\|_2 = 0$. Equivalently, the centroid of any finite point set is unchanged under an exact isometry on that set (do Carmo, 1992).

3. The orthogonality loss blocks task interference; near-orthogonality in high dimensions limits random overlap. The per-batch orthogonality penalty drives new-task gradients into the orthogonal complement of $\text{span}(Z_r)$. Furthermore, in high dimensions, two random k-dimensional subspaces of $\mathbb{R}^d$ are almost orthogonal, with principal angles concentrating near $\frac{\pi}{2}$ and overlaps of order $O(1/\sqrt{d})$ (**?**). Hence even without explicit orthogonality, random task features would minimally interfere.

4. Quasi-isometric preservation of decision boundaries implies ε-preservation of empirical risk. The 0–1 classification loss is 1-Lipschitz in the ℓ_∞ distance between the old and new probability vectors. Since isometry and orthogonality together imply $\|g^{\text{new}}(x) - g^{\text{old}}(x)\|_\infty \leq \varepsilon$, each example's risk changes by at most ε, and averaging over the data yields $\left|\mathcal{R}_t^{\text{new}} - \mathcal{R}_t^{\text{old}}\right| \leq \varepsilon$ (Lopez-Paz & Ranzato, 2017; Fan et al., 2024).

5. Concentration in d provides an exponentially small tail. By Lévy's lemma, any 1-Lipschitz distortion of a point on the d-sphere concentrates with probability $1 - 2e^{-c\,d\,\varepsilon^2}$. Thus the chance of a large distance or risk shift decays exponentially in the latent dimension d (Ledoux, 2001).

All in all, isometries lock past-task geometry; orthogonality blocks new-task leakage; Lipschitz-continuity of the loss converts small geometric distortions into small risk changes; and concentration in high d makes large deviations vanishingly unlikely. Hence catastrophic forgetting is bounded by $O(\varepsilon)$ both per task and on average.

Through embedding space isometries, we thus show that CAMELS retains earlier performance while expanding into new orthogonal sub-spaces (Beshkov et al., 2022; Fan et al., 2024). Unlike parameter-space methods (EWC Kirkpatrick et al., 2017, OGD Farajtabar et al., 2020), CAMELS's guarantees are metric and task-agnostic, requiring no Fisher-matrix or gradient storage, only latent embedding vectors that are stored alongside replay buffer samples and logits.

4 Results

We evaluate CAMELS on three canonical *continual* benchmarks that all derive multiple tasks from the original MNIST digit corpus (LeCun et al., 1998) the test *input permutation*, *label partition*, and *domain rotation*, namely **Permuted MNIST, Split MNIST,** and **Rotated MNIST**, covering both domain- and class-incremental learning. Details about these benchmarks are in the appendix.

We first test ablations; in Table 1 that CAMEL, using relative (cosine similarity distance) anchoring to create an isometry with samples from the balanced replay buffer, performs better than both absolute anchoring (comparing current embeddings to previous embeddings as vectors) and the replay buffer baseline when taking a sweep over 10 trials for optimal hyperparameters for Permuted MNIST.

Method	latent dim	λ_{man}	λ_{orth}	Task 0	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6	Task 7	Task 8	Task 9	**Average**
CAMELS	64	2.0	1.0	92.13	92.26	91.26	92.47	91.51	94.11	93.20	95.25	96.00	96.70	**93.49**
Baseline	64	0.0	0.0	88.39	89.41	88.61	88.64	89.67	90.53	91.08	93.83	94.02	96.27	91.05
ANA	64	2.0	1.0	89.11	90.06	86.99	89.66	91.25	93.20	95.05	96.06	97.37	98.06	92.68

Table 1: Task Accuracy Percent: CAMEL, Baseline, and Absolute Norm Anchoring variants.

We next display results for the various MNIST benchmarks – as seen in Figure 3, that CAMELS is an improvement on each benchmark vis-a-vis other continual learning models.

Dataset	Method	Accuracy ± Std. (%) Task 1	Task 2	Task 3	Task 4	Task 5
PermutedMNIST	OGD	79.5 ± 2.3	88.9 ± 0.7	89.6 ± 0.3	91.8 ± 0.9	92.4 ± 1.1
	A-GEM	85.5 ± 1.7	87.0 ± 1.5	89.6 ± 1.1	91.2 ± 0.8	93.9 ± 1.0
	EWC	64.5 ± 2.9	77.1 ± 2.3	80.4 ± 2.1	87.9 ± 1.3	93.0 ± 0.5
	SGD	60.6 ± 4.3	77.6 ± 1.4	79.9 ± 2.1	87.7 ± 2.9	92.4 ± 1.1
	CAMEL (Ours)	95.87 ± 0.35	95.95 ± 0.44	95.74 ± 0.43	96.77 ± 0.59	98.02 ± 0.21
SplitMNIST	OGD	98.6 ± 0.8	99.5 ± 0.1	98.0 ± 0.5	98.8 ± 0.5	99.2 ± 0.3
	A-GEM	92.9 ± 0.8	96.3 ± 0.1	86.5 ± 0.5	92.3 ± 0.5	99.3 ± 0.3
	EWC	90.2 ± 5.7	98.9 ± 0.2	91.1 ± 3.5	94.4 ± 2.0	99.3 ± 0.2
	SGD	88.2 ± 5.9	98.4 ± 0.9	90.3 ± 4.5	95.2 ± 1.0	99.4 ± 0.2
	CAMEL (Ours)	98.94 ± 0.10	98.89 ± 0.09	99.1 ± 0.17	99.04 ± 0.17	99.79 ± 0.03
RotatedMNIST	OGD	75.6 ± 0.8	86.6 ± 0.1	91.7 ± 0.5	94.3 ± 0.5	93.4 ± 0.3
	A-GEM	72.6 ± 1.8	84.4 ± 1.6	91.0 ± 1.1	93.9 ± 0.6	94.6 ± 1.1
	EWC	61.9 ± 2.0	78.1 ± 1.8	89.0 ± 1.6	94.4 ± 0.7	93.9 ± 0.6
	SGD	62.9 ± 1.0	76.5 ± 1.5	88.6 ± 1.4	95.1 ± 0.5	94.1 ± 1.1
	CAMEL (Ours)	91.7 ± 0.47	92.12 ± 0.21	91.14 ± 0.41	92.94 ± 0.47	96.56 ± 0.2

Figure 3: Per-task test accuracy (%) ‡ SD across 5 tasks for CL MNIST Benchmarks

We are able to outperform existing methods in continual learning, especially with respect to the PermutedMNIST benchmark. We are able to do this without any task ID supervision and drastically lower variance between runs, demonstrating the consistency of our latent representations.

To compare across a different baseline, we looked to test our model on SplitCIFAR10. To tackle this, we constructed a DLA (Yu et al., 2019) backbone for our model, outputting a latent representation of our RGB images like with MNIST before. From there, the framework follows the same pipeline. The following table shows results for our model for 10 epochs for 5 tasks.

Baselines	**Split-CIFAR-10**
Conv. Neural Network	66.62 (±1.06)
Offline re-training + task oracle	80.42 (±0.95)
Task-Agnostic	
CAMELS (Ours)	85.84 (±0.93)
HVCL	78.41 (±1.18)
HVCL w/ GR	81.00 (±1.15)
CL-DR	86.72 (±0.30)
TLR	74.89 (±0.61)
MAS	73.50 (±1.54)

Table 2: Split-CIFAR-10 Accuracy across Task-Agnostic Models

We are within the margin of error for the model performing the best in this benchmark. This shows the versatility of our model with different model backbones and ability to learn more complex representations.

4.1 Manifold Loss Effectiveness

One visualization we performed was to see whether the manifold loss was truly preserving the metric of prior expert task subspaces – that is, given an old task's dataset after fully continual learning on all tasks, is our fully trained model's pairwise distance matrix of embeddings of $\{(x_i, y_i)\}_{i \in \text{len}(D_t)}$ consistent with the expert model embeddings (the embedding snapshots of the model freshly trained on task t). To verify this, we collect all embeddings of the oldest task and compute their pairwise distance matrix and compare it to the pairwise distance matrix of our final model evaluated on task 0 after trained on all T tasks. We average max pool for visualization purposes (to remove high frequency noise) but find strong metric similarities and very minor deviations, giving empirical justification for the following theoretical groundings of our technique.

In our training, we note that intermediate tasks tend to suffer the most catastrophic forgetting in certain cases, as they combine the lack of recency of old tasks with the "garbled" learning of combining replay buffer gradient updates

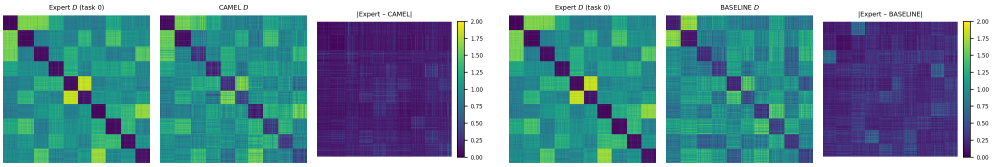


Figure 4: **(Left)** CAMELS' distance-matrix reconstruction on task 0: the expert model's D, the CAMELS model's D, and their absolute difference (all embedded in one row).
(Right) The same for the balanced-replay baseline. CAMELS clearly preserves the expert metric far more faithfully.

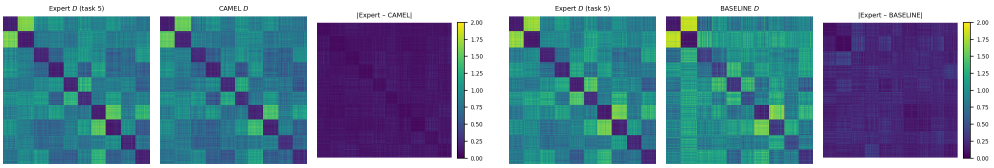


Figure 5: We perform the same comparison for an intermediate task $t = 5$ and see though less significant, CAMELS outperforms the baseline in task subspace metric reconstruction.

from old tasks with the gradient updates trying to converge on the current task. Hence, we also plot the distance matrix reconstruction between CAMELS' and the baseline for task 5 and indeed notice CAMELS' reconstruction of the expert task metric is far better.

4.2 Latent Space Interpretability

Our resulting latent space is highly interpretable as points in a metric space, clustered by both task similarity (input distribution) and output class value. We plot task embedding vectors from a 10 task permuted MNIST task sequence a using t-SNE dimensionality reduction to two dimensions . Note OL refers to Orthogonality Loss, AMAL refers to Absolute Metric Anchoring Loss (anchoring to the difference in Euclidean norm of embedding vector pairs), and RMAL refers to the relative metric anchoring loss, the cosine similarity between embedding vector pairs. We also provide in the appendix a comparison between expert and final model distance matrix plots, demonstrating the continuity of our isometries over training across 10 tasks.

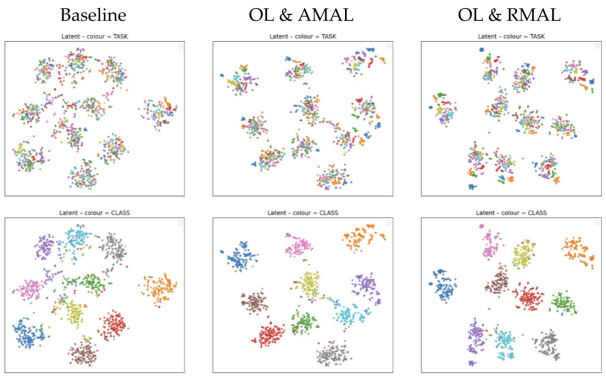


Figure 6: 2-D t-SNE plots of latent vectors uniformly sampled from $T = 10$ tasks for Permuted MNIST during final evaluation. On the top, each latent vector is coloured from the task (ie. permutation) it is taken from, and on the bottom, each embedding vector is coloured by its MNIST class (digit $0, \cdots, 9$) irrespective of permutation. As the above figure demonstrates, our metric space clustering and orthogonality guarantees enforce much clearer, well-defined task/class cluster separation compared to baselines that do not incorporate this loss.

5 Conclusion and Future Work

In this work, we present a novel continual-learning framework, **CAMELS**, a latent space-driven training algorithm combining manifold anchoring and subspace orthogonality to deliver robust, geometry-aware latent representations. Our empirical studies on Permuted and Split MNIST demonstrate that CAMELS not only matches but often exceeds the performance of existing replay-based and regularization methods, while maintaining a strong level of interpretability on its joint task latent space due to our metric space constraints. The intuitive appeal of CAMELS lies in its grounding in high-dimensional geometry, locking the relative structure of replayed samples and pushing new tasks into fresh latent directions, mirroring neural manifold strategies observed in biological learning. Some of the limitations of our framework include the sensitivity to new tasks, requiring changes to our hyperparameter set to accommodate. We see several avenues in continuing to develop CAMELS, including scaling to larger continual learning benchmarks (Core50, CIFAR-100), experimenting with different downstream models (latent space generative decoders, denoising models, etc.). We hope to also continue to refine our study of manifold loss dynamics, in particular experience-drift mechanisms that dynamically select replay samples for maximal isometric coverage, and continue to expand on our theoretical understanding of task learning as isometries.

In summary, CAMELS offers a scalable, theoretically grounded, and interpretable approach to continual learning. We believe its geometric insights and strong empirical performance will inspire a new line of research at the intersection of representation geometry and continual adaptation, and further help us understand the training dynamics of high dimensional latent spaces in continual settings.

REFERENCES

Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the 15th European Conference on Computer Vision (ECCV)*, pp. 139–154. Springer, 2018. doi: 10.1007/978-3-030-01219-9_9. URL https://doi.org/10.1007/978-3-030-01219-9_9.

Nader Asadi, Thomas Perrett, and Jian Zhang. Prototype–sample relation distillation: Towards replay-free continual learning. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, 2023. URL https://arxiv.org/abs/2303.14771.

Lilach Avitan and Carsen Stringer. Not so spontaneous: Multi-dimensional representations of behaviors and context in sensory areas. *Neuron*, 110(3):3064–3075, 2022. doi: 10.1016/j.neuron.2022.06.019. URL https://pubmed.ncbi.nlm.nih.gov/35863344/.

Georgi Beshkov, Silviu-Leonard Pintea, and Jan van Gemert. Isometric regularisation improves robustness and generalisation. *arXiv preprint arXiv:2211.01236*, 2022. URL https://arxiv.org/abs/2211.01236.

Leonard M. Blumenthal. *Theory and applications of distance geometry*. Oxford, 1953.

Lorenzo Bonicelli, Matteo Boschni, Angelo Porrello, Concetto Spampinato, and Simone Calderara. On the effectiveness of lipschitz-driven rehearsal in continual learning, 2022. URL https://openreview.net/pdf?id=TThSwRTt4IB.

Daniel Brignac, Niels Lobo, and Abhijit Mahalanobis. Improving replay sample selection and storage for less forgetting in continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 3532–3541, 2023. URL https://openaccess.thecvf.com/content/ICCV2023W/VCL/papers/Brignac_Improving_Replay_Sample_Selection_and_Storage_for_Less_Forgetting_in_ICCVW_2023_paper.pdf.

Hyuntak Cha, Jaeho Lee, and Jinwoo Shin. Co^{2}l: Contrastive continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9516–9525, 2021. URL https://openaccess.thecvf.com/content/ICCV2021/html/Cha_Co2L_Contrastive_Continual_Learning_ICCV_2021_paper.html.

Arslan Chaudhry, Puneet K. Dokania, Thalaiyasingam Ajanthan, and Philip H.S. Torr. Riemannian walk for incremental learning. In *ECCV*, 2018. URL https://arxiv.org/abs/1801.10112.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with A-GEM. In *ICLR*, 2019. URL https://arxiv.org/abs/1812.00420.

Aristotelis Chrysakis and Marie-Francine Moens. Online continual learning from imbalanced data. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1952–1961. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/chrysakis20a.html.

Manfredo P. do Carmo. *Riemannian Geometry*. Birkhauser, 1992. ISBN 9780817634902. URL https://link.springer.com/book/978081763490.

Yan Fan, Yu Wang, Pengfei Zhu, Dongyue Chen, and Qinghua Hu. Persistence homology distillation for semi-supervised continual learning. In *NeurIPS*, 2024. URL https://openreview.net/forum?id=qInb7EUmxz&referrer=%5Bthe%20profile%20of%20Pengfei%20Zhu%5D(%2Fprofile%3Fid%3D~Pengfei_Zhu1).

Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. Orthogonal gradient descent for continual learning. In *PMLR*, 2020. URL https://arxiv.org/pdf/1910.07104.

Yasin Findik and Farhad Pourkamali-Anaraki. D-cbrs: Accounting for intra-class diversity in continual learning. *IEEE ICIP*, 2022. URL https://arxiv.org/abs/2207.05897.

Juan A. Gallego, Matthew G. Perich, Lee E. Miller, and Sara A. Solla. Neural manifolds for the control of movement. *Neuron*, 94(5):978–984, 2017. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6122849/.

Juan A. et al. Gallego. Cortical population activity within a preserved neural manifold underlies multiple motor behaviors. *Nature Communications*, 9:4231, 2018. doi: 10.1038/s41467-018-06560-z. URL https://www.nature.com/articles/s41467-018-06560-z.

Marton Albert Hajnal, Duy Tran, Michael Einstein, Mauricio Vallejo Martelo, Karen Safaryan, Pierre-Olivier Polack, and Peyman Golshani. Continuous multiplexed population representations of task context in the parietal cortex. *Nature Communications*, 14:1234, 2023. doi: 10.1038/s41467-023-42441-w. URL https://www.nature.com/articles/s41467-023-42441-w.

William B. Johnson and Joram Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. In *Conf. in modern analysis and probability*, 1984. URL https://homes.cs.washington.edu/~jrl/teaching/cse422wi24/notes/docs/JL.pdf.

Prannay Khosla, Piotr Teterwak, Chen Wang, and Aaron et al. Sarna. Supervised contrastive learning. In *NIPS*, 2020. URL https://arxiv.org/abs/2004.11362.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, and et al. Overcoming catastrophic forgetting in neural networks. volume 114, pp. 3521–3526, 2017. URL https://doi.org/10.1073/pnas.1611835114.

Alexander Krawczyk and Alexander Gepperth. An analysis of best-practice strategies for replay and rehearsal in continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 4196–4204, June 2024. URL https://openaccess.thecvf.com/content/CVPR2024W/CLVISION/html/Krawczyk_An_Analysis_of_Best-practice_Strategies_for_Replay_and_Rehearsal_in_CVPRW_2024_paper.html.

S Kellis D Bjanes K Pejsa B W Brunton R A Andersen L Bashford, I Rosenthal. Neural subspaces of imagined movements in parietal cortex remain stable over years in humans. *eLife*, 12:e57772, 2023. URL https://pubmed.ncbi.nlm.nih.gov/37461446/.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791. URL https://ieeexplore.ieee.org/document/726791.

Michel Ledoux. *The Concentration of Measure Phenomenon*, volume 89 of *Mathematical Surveys and Monographs*. American Mathematical Society, 2001. ISBN 978-1-4704-1316-3. URL https://doi.org/10.1090/surv/089.

Xingyu Li, Bo Tang, and Haifeng Li. Adaer: An adaptive experience replay approach for continual lifelong learning. *arXiv preprint arXiv:2308.03810*, 2023. URL https://arxiv.org/abs/2308.03810.

Zhizhong Li and Derek Hoiem. Learning without forgetting. In *ECCV*, 2016. URL https://arxiv.org/abs/1606.09282.

Lei Liu, Li Liu, and Yawen Cui. Prior-free balanced replay: Uncertainty-guided reservoir sampling for long-tailed continual learning. In *Proceedings of the 32nd ACM International Conference on Multimedia (MM '24)*, pp. To appear. ACM, 2024. doi: 10.1145/3664647.3681106. URL https://arxiv.org/abs/2408.14976.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *NeurIPS*, 2017. URL https://arxiv.org/abs/1706.08840.

Jieliang Luo and Hui Li. Dynamic experience replay. In *ICLR*, 2020. URL https://arxiv.org/abs/2003.02372.

Jamie R. McFayden, Barbara Heider, Anushree N. Karkhanis, Shaun L. Cloherty, Fabian Munoz, Ralph M. Siegel, and Adam P. Morris. Robust coding of eye position in posterior parietal cortex despite gaze-related input variability. *Journal of Neuroscience*, 42(20):4116–4128, 2022. doi: 10.1523/JNEUROSCI.0674-21.2022. URL https://www.jneurosci.org/content/42/20/4116.

Luca Moschella, Valentino Maiorca, Marco Fumero, Antonio Norelli, Francesco Locatello, and Emanuele Rodola. Relative representations enable zero-shot latent space communication. *ICLR*, 2023. URL https://arxiv.org/abs/2209.15430.

Ulises Pereira-Obilinovic, Sean Froudist-Walsh, and Xiao-Jing Wang. Cognitive network interactions through communication subspaces in the primate cortex. *Cell Reports*, 25(12), 2024. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC11566003/.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph Lampert. iCaRL: Incremental classifier and representation learning. In *CVPR*, 2017. URL https://arxiv.org/abs/1611.07725.

David A. Sabatini and Matthew T. Kaufman. Reach-dependent reorientation of rotational dynamics in motor cortex. *Nature Communications*, 15:51308, 2024. doi: 10.1038/s41467-024-51308-7. URL https://www.nature.com/articles/s41467-024-51308-7.

Tsvetomira Dumbalska Andrew Saxe Christopher Summerfield Timo Flesch, Keno Juechems. Orthogonal representations for robust context-dependent task performance in brains and neural networks. *Neuron*, 109(5):4212–4219, 2022. URL https://pubmed.ncbi.nlm.nih.gov/35085492/.

Stefano Fusi W. Jeffrey Johnston. Modular representations emerge in neural networks trained to multiplex tasks. *bioRxiv*, 2024. URL https://www.biorxiv.org/content/10.1101/2024.09.30.615925v1. Preprint.

Zihao Xu, Xuan Tang, Yufei Shi, Jiafeng Zhang, Jian Yang, Mingsong Chen, and Xian Wei. Continual learning via manifold expansion replay. *arXiv preprint*, arXiv:2310.08038, 2023. URL https://arxiv.org/abs/2310.08038.

Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation, 2019. URL https://arxiv.org/abs/1707.06484.

Xiaofan Yu, Yunhui Guo, Sicun Gao, and Tajana Rosing. SCALE: Online self-supervised lifelong learning without prior knowledge. In *CVPR*, 2023. URL https://arxiv.org/abs/2208.11266.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *ICML*, 2017. URL https://arxiv.org/abs/1703.04200.

Jianshu Zhang, Yankai Fu, Ziheng Peng, Dongyu Yao, and Kun He. Core: Mitigating catastrophic forgetting in continual learning through cognitive replay. *arXiv preprint arXiv:2402.01348*, 2024. URL https://arxiv.org/abs/2402.01348.

Tan Zhuo, Zhiyong Cheng, Zan Gao, Hehe Fan, and Mohan Kankanhalli. Continual learning with strong experience replay. *arXiv preprint*, 2023. URL https://arxiv.org/abs/2305.13622.

A Experiment Details

All our experiments were done using Tesla P100 GPUs. Code is available on our GitHub, not linked to preserve anonymity.

A.1 Experiment Overview

In **Permuted MNIST** (PMNIST) (Kirkpatrick et al., 2017), each task applies a fixed but task-specific random permutation to the 28×28 pixel vector, leaving labels unchanged. All ten digits appear in every task, but the input distribution differs radically, making representation transfer challenging yet label–consistent. With **Split MNIST** (Zenke et al., 2017), the ten digits are split into five two–way classification tasks: $(0 \vee 1), (2 \vee 3), \ldots, (8 \vee 9)$. The input distribution is identical across tasks (the normal MNIST dataset), but class overlap is absent, emphasizing catastrophic forgetting of the classifier head rather than the feature extractor.In **Rotated MNIST** (RMNIST) (Lopez-Paz & Ranzato, 2017), task t rotates every image by a fixed angle $\theta_t \in [0°, 180°)$. Following Farajtabar et al. (2020), we use ten equidistant angles $(0°, 20°, \ldots, 180°)$. This produces a smooth domain shift, testing a method's ability to learn isometry-equivariant features.

A.2 Model Architecture

Encoder Input examples $\mathbf{x} \in \mathbb{R}^{d_{\text{in}}}$ is first mapped by a two–layer ReLU MLP

$$\mathbf{z} \;=\; \operatorname{norm}\Big(W_2\,\sigma(W_1\mathbf{x} + b_1) + b_2\Big) \;\in\; \mathbb{R}^D, \qquad \operatorname{norm}(\mathbf{v}) = \tfrac{\mathbf{v}}{\|\mathbf{v}\|_2},$$

where $D \ll d_{\text{in}}$ and $\operatorname{norm}(\cdot)$ keeps the embedding on the unit sphere so all downstream losses are scale-free.

Classifier head. *(replace with any downstream model)* A single linear layer $h(\mathbf{x}) = W_c\mathbf{z}$ with $W_c \in \mathbb{R}^{C \times D}$ produces logits for the C semantic classes shared by every task.

Task-balanced replay bank. After finishing task t we (i) freeze the current encoder, (ii) compute reference embeddings $\mathbf{z}^{\text{ref}} = E_{\theta_t}(\mathbf{x})$ for a balanced subset of training items, (iii) store triples $(\mathbf{x}, y, \mathbf{z}^{\text{ref}})$ in a task-specific bank of equal size. At run time each replay mini-batch is drawn from one bank, so its reference embeddings come from a single "expert" encoder. Intuitively, this expert represents our model freshly trained on the current tahttps://arxiv.org/abs/1606.09282sk t, which empirically has a maximal accuracy ranging around 98-99%. Another alternative is to continually add to the replay buffer throughout training epochs, but we find this process empirically outperforms continual buffer additions, as we are adding expert samples for future replay as opposed to intermediate latent activations.

Orthogonal task sub-spaces. From all the sampled embeddings of a given replay batch, we take the orthogonal matrix Q from the QR decomposition of the concatenated embedding matrix $\begin{bmatrix} z_1^{ref} & z_{ref2} & \cdots & z_R^{ref} \end{bmatrix}$ where R is the size of our replay batch. Q represents an orthogonal basis of the space spanned by these replay embedding vectors, taken as an approximation of prior task results. Then, we minimize the loss given in our Setup section.

Loss Function Recall, our loss function is:

$$\mathcal{L} = \underbrace{\mathrm{CE}\big(h(\mathbf{x}_c), y_c\big)}_{\text{current}} + \underbrace{\mathrm{CE}\big(h(\mathbf{x}_r), y_r\big)}_{\text{replay}} + \lambda_{\text{man}} \underbrace{\frac{1}{B^2}\big\|D(\mathbf{z}_r) - D(\mathbf{z}^{\text{ref}})\big\|_F^2}_{\text{manifold}} + \lambda_{\text{orth}} \underbrace{\frac{1}{B}\sum_{i=1}^{B}\big\|Q_{\text{prev}}^{\top}\mathbf{z}_{r,i}\big\|_2^2}_{\text{orthogonal}}$$

- **Current task classification loss:** This is the standard cross-entropy loss on the current task batch $(\mathbf{x}_c, y_c)$, which minimizes the empirical risk on the new task.
- **Replay task classification loss:** This cross-entropy loss is applied to a replay batch $(\mathbf{x}_r, y_r)$ from previously seen tasks, helping prevent catastrophic forgetting by reinforcing prior knowledge.
- **Manifold anchoring loss:** Here, $D(Z)_{ij} = d(Z_i, Z_j)$ is a pairwise cosine distance matrix. This term encourages the pairwise geometry of latent representations $\mathbf{z}_r = E_\theta(\mathbf{x}_r)$ to match the structure from a reference model $\mathbf{z}^{\text{ref}}$. This preserves the fine-grained relational structure within old-task embeddings without anchoring their absolute positions.

- **Orthogonal subspace penalty:** Q_{prev} is an orthonormal basis (e.g., via QR decomposition) spanning the replay embeddings. This term penalizes the projection of new embeddings onto the subspace spanned by prior tasks, encouraging new task representations to explore orthogonal directions and thus reduce interference.

A.3 Training Overview

Our full training algorithm is as follows:

Algorithm 1 One training step of CAMEL

Require: Current batch (x, y) from task T, replay buffer $\mathcal{B}$, parameters θ, hyper-params $\lambda_{\text{man}}, \lambda_{\text{orth}}$
Ensure : Updated parameters θ, buffer $\mathcal{B}$

Embed current data
- $Z_{\text{cur}} \leftarrow \text{Embed}(x; \theta)$; `map into unit-sphere`
- $\ell_{\text{CE}} \leftarrow \text{CE}\big(\text{Cls}(Z_{\text{cur}}), y\big)$; `classification loss`

Sample replay
- $(\tilde{x}, \tilde{y}, \tilde{z}, \tilde{t}) \leftarrow \mathcal{B}.\text{sample}(m)$ **if** $\tilde{x} \neq \emptyset$ **then**
  - $Z_{\text{rep}} \leftarrow \text{Embed}(\tilde{x}; \theta)$; `current embeddings of replay`
  - $\ell_{\text{rep}} \leftarrow \text{CE}\big(\text{Cls}(Z_{\text{rep}}), \tilde{y}\big)$; `replay CE`
  - $\ell_{\text{man}} \leftarrow \text{MaskedPairwise}(Z_{\text{rep}}, \tilde{z}, \tilde{t})$; `preserves intra-task distances`
  - $Q \leftarrow \text{QR}(\tilde{z}^{\top})$; `basis of replay subspace`
  - $\ell_{\text{orth}} \leftarrow \|Z_{\text{cur}} Q\|_F^2$; `carves new subspace`
- **else**
  - $\ell_{\text{rep}} \leftarrow 0 \;\; \ell_{\text{man}} \leftarrow 0 \;\; \ell_{\text{orth}} \leftarrow 0$

Total loss & update
- $\ell \leftarrow \ell_{\text{CE}} + \ell_{\text{rep}} + \lambda_{\text{man}}\, \ell_{\text{man}} + \lambda_{\text{orth}}\, \ell_{\text{orth}} \;\; \theta \leftarrow \theta - \eta\, \nabla_\theta\, \ell$; `gradient step`

Buffer update
- $\mathcal{B}.\text{add}(x, y, Z_{\text{cur}}, T)$; `store fresh embeddings`

return θ, $\mathcal{B}$

A.4 Replay Buffer Details

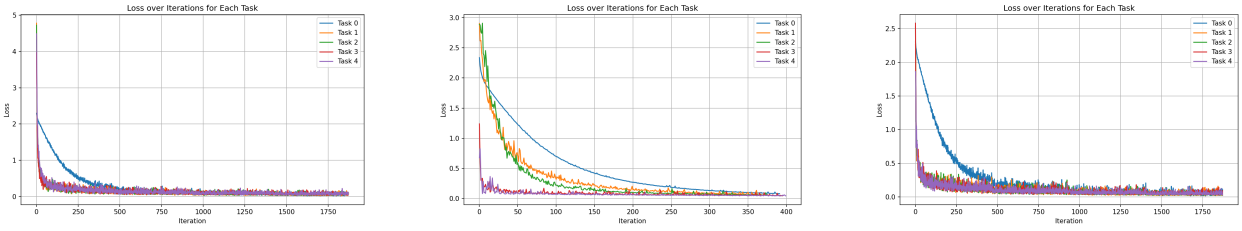
The design of our replay buffer is crucial for defining the latent activations used in masked-manifold anchoring and orthogonality penalties. Early on we experimented with an EMA "teacher" to generate soft targets in-flight, but this approach monotonically favored recent tasks and accelerated forgetting over long sequences. Especially important for manifold anchoring is a balanced replay buffer with respect to task and classes, where the joint balance is primarily important in class-incremental learning where each task may not evenly distribute amongst its output classes. Below details the evolution of our replay buffer across experiment development.

- A simple reservoir buffer treats every incoming sample equally, but this can cause under-representation of older or rare classes, leading to imbalanced coverage that exacerbates catastrophic forgetting (Brignac et al., 2023). Moreover, uniform reservoir does not account for task boundaries: in class-incremental settings it tends to flush out early-task samples too quickly, hurting stability (Aljundi et al., 2018).
- To mitigate class imbalance, we adopt Class-Balanced Reservoir Sampling (CBRS), which enforces equal quota per class within the buffer (Chrysakis & Moens, 2020). CBRS ensures that every class remains represented, preserving the manifold geometry needed for paired-distance losses (Krawczyk & Gepperth, 2024).
- Beyond class balance, tasks themselves may differ in difficulty or data volume. We therefore extend CBRS to a task×class–balanced reservoir, allocating a fixed quota to each (task, class) cell, and trimming excess uniformly across cells (Findik & Pourkamali-Anaraki, 2022). This design guarantees that replay mini-batches approximate the original task distributions, a best-practice highlighted for large-scale continual learning.

- At sampling time, we draw first in a balanced fashion across tasks and classes, then top up with any leftover capacity to reach the target batch size; this preserves diversity without biasing toward recent tasks (Liu et al., 2024).
- We also allow the buffer to adaptively reallocate space among tasks based on their observed forgetting rates, following strategies like Adaptive Experience Replay (AdaER) (Li et al., 2023) and Uncertainty-Guided Reservoir Sampling (Zhang et al., 2024). In practice, we find a static task–class balanced buffer yields the best mix of simplicity and performance in our CAMELS framework.

A.5 Experimental Results

In this section, we discuss the training dynamics of our framework, specifically looking at the individual losses for each of our experiment benchmarks.



(a) PermutedMNIST (b) SplitMNIST (c) RotatedMNIST

Figure 7: For each of our baseline experiment benchmarks, we see the above loss plots during training for (a) PermutedMNIST, (b) SplitMNIST, and (c) RotatedMNIST from left to right. For an individual plot, each task is separated into a separate loss curve independent of the others, so we could see the forward learning and convergence of our framework.

Comparing each of the loss plots, we can see similarities in convergence for (a) and (c) due to the nature of our dataset for training. With both PermutedMNIST and RotatedMNIST, we are applying a transformation to the entire dataset and per task training on a set as large as the initial dataset. As such, the model is able to learn in a similar fashion for both; however, RotatedMNIST seems to converge slower for the first task, indicative of the task difficulty. Unlike (a) and (c), SplitMNIST follows a shorter number of iterations due to the nature of the dataset, and sees a more coarse loss convergence. Especially with (b), we see the forward learning of each task at a finer resolution; however, this is present in all three, demonstrating the effectiveness of the framework for the problem formulation.

B Theoretical Results

In this section we provide a proof of our central theorem that links task-subspace isometries to bounds on catastrophic forgetting.

Throughout let $\Phi_\theta : \mathbb{R}^{d_0} \to \mathbb{S}^{d-1} \subset \mathbb{R}^d$ be the unit-normalized encoder (mapping to points on the unit sphere in $\mathbb{R}^d$, let $\mathcal{T}_t = \{(x_i, y_i)\}_{i=1}^{n_t}$ be the t^{th} task represented as an empirical sample from an underlying data distribution, and and denote the latent manifold $\mathcal{M}_t := \Phi_\theta(\operatorname{supp} \mathcal{T}_t)$. Then, $\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda_{\text{man}}\mathcal{L}_{\text{man}} + \lambda_{\text{orth}}\mathcal{L}_{\text{orth}}$ is our CAMELS objective (§3).

B.1 Supporting Lemmas

The following lemma proved in (Johnson & Lindenstrauss, 1984) will be useful:

Lemma 2 (Johnson-Lindenstrauss extension theorem)**.** *Given an n-point metric space X, $X \subset Y$, $\mathcal{H}$ a Hilbert space, $f : X \to \mathcal{H}$ a mapping. Then, there exists an extension $F : Y \to \mathcal{H}$ of f (that is, $F|_X = f$) where $Lip(F) \leq \sqrt{\log n} Lip(f)$ approximately.*

In fact, we use the above to prove the next lemma, showing that a zero manifold loss on a finite set of points can extent to an isometry on a broader space:

Lemma 3 (Masked–manifold isometry)**.** *If $\mathcal{L}_{\text{man}} = 0$ then for every past task $t < T$ Φ_θ restricted to $\mathcal{T}_t$ is an exact cosine–isometry; pair-wise distances are preserved. (Distance-preserving maps yield quasi-isometries by Johnson & Lindenstrauss, 1984; Blumenthal, 1953.)*

Proof. We want to show that if $\mathcal{L}_{\text{man}} = 0$ then $\Phi_\theta|_{\mathcal{T}_t}$ is an isometry (in $d_{\cos}$) on every past task $t < T$. Let $M_t = \{(i,j) \,|\, \tau_i = \tau_j,\ i \neq j\}$ be the set of intra–task pairs stored for task t. Zero masked–manifold loss implies

$$\forall (i,j) \in M_t : \ \left| d_{\cos\left(\Phi_\theta(x_i), \Phi_\theta(x_j)\right)} - d_{\cos\left(\Phi_{\theta^\star}(x_i), \Phi_{\theta^\star}(x_j)\right)} \right| = 0.$$

Hence pairwise distances on the finite set $S_t = \{x_i : (x_i, y_i) \in \mathcal{T}_t\}$ are preserved.

Consider the map T defined by taking a point $\Phi_\theta(x_i) \mapsto \Phi_{\theta^\star}(x_i)$; that is, the map that takes the representation of a point under our old encoder to the representation under our new encoder. This map is 1-Lipschitz on S_t by definition (1-Lipschitz is just an isometry). We invoke 2 here: we take our extension space as $conv(S_t)$, the convex hull of our points in S_t. Our gradient updates made small enough along with l_2 normalization (along with the assumption that our classifier is Lipschitz) ensures that our latent points can rest in a convex set (example: the ball of vectors at most unit norm). So, for all practical purposes, nothing will ever leave this convex set. Thus, for a mini-batch-size B, we can bound the Lipschitz constant of our map extension T' by $1 \pm \epsilon$ given that our dimension size $k \geq \epsilon^{-2} \log n$. As per the JL lemma, we choose our latent dim to be large enough such that for our desired ϵ this holds, and we have a quasi ϵ-isometry over the whole convex hull (that, is, distance between points differ by at most ϵ). This can also be extended to the entire support of $\mathcal{T}_t$ (Blumenthal, 1953) □

Lemma 4 (Per-batch orthogonality)**.** *Gradient descent on* $\mathcal{L}_{\text{orth}} = \|Q_{mb}^\top Z_c\|_F^2$ *drives the current embeddings* Z_c *into the orthogonal complement of* $\text{span}(Z_r)$*. Hence* $T_z\mathcal{M}_T \perp \text{span}(\mathcal{M}_{<T})$.

Proof. For unit z the loss w.r.t. a single sample is $\ell(z) = \|Q_{\text{mb}}^\top z\|_2^2$. Its gradient is $\nabla_z \ell = 2Q_{\text{mb}}Q_{\text{mb}}^\top z = 2\,\text{Proj}_{(Z_r)}(z)$. Recall that our Q_{mb} is the *orthonormal* matrix whose column space is equivalent to $\text{span}(Z_r)$ and hence our projection operator is just QQ^T. So, gradient descent therefore subtracts *exactly* the projection of z onto $\text{span}(Z_r)$; the unique stationary point is when that gradient is zero, hence when the projection vanishes, This occurs by definition when $z \perp \text{span}(Z_r)$. Extending to a basis of $T_z\mathcal{M}_T$ (the tangent space of the manifold $\mathcal{M}_T$ rooted at z) follows by linearity. Suppose we start at z and take a smooth curve $c : [0,1] \to \mathcal{M}_T$ such that $c(0) = z, \dot{c}(0) = v$ for some tangent direction $v \in \mathcal{T}_z\mathcal{M}_T$. Then,

$$\frac{d}{dt} l(\gamma(t))|_{t=0} = \langle \nabla_z l, v \rangle = 2\langle QQ^T z, v \rangle$$

But we showed that at the fixed point of our loss, $QQ^T z = 0$, so $\forall v$, linearity of QQ^T implies $QQ^T v = 0$ ($2\langle QQ^T z, v\rangle = 0 \implies 2\langle z, QQ^T v\rangle$). So the projection of any tangent vector v onto $span(Z_r)$ is zero, meaning our tangent space at the fixed point z is orthogonal to our previous tasks' span.

Intuitively, this means that gradient steps along our fixed point will not affect old task representations – we can improve on our current task in the orthogonal subspace of all previous tasks.

□

Lemma 5 (Prototype Stability)**.** *Under Lemma 3, the class prototypes* $w_y = \mathbb{E}[\,\Phi_\theta(x) \mid y]$ *are unchanged for* $y \in \mathcal{Y}_{\leq T-1}$*. Cluster-based decision rules therefore remain fixed (Asadi et al., 2023; Cha et al., 2021; Yu et al., 2023).*

Proof. Our prototype w_y for every output class y represents the average over all latent vectors $\Phi_\theta(x)$ that map to y in the classifier. We take $(y \in \mathcal{Y}_{\leq T-1})$ (in domain incremental learning, this is just the entire output class, but in class incremental learning this is instead restricted to the output classes that we have seen so far). Because learning every subsequent task is an isometry, we have that $\{w_y : y \in \mathcal{Y}_t\}$ also remain at the same pairwise distances as we learn more tasks. Take any w_y^{new} and w_y^{old}, S_y as the set of inputs that map to the class y. We can write the prototype difference as

$$w_y^{new} - w_y^{old} = \frac{1}{|S_y|} \sum_{x \in S_y} (\Phi_{\theta_{new}}(x) - \Phi_{\theta_{old}}(x))$$

Note that as all these embeddings are approximately unit norm, so $\Phi_{\theta_{new}}(x) - \Phi_{\theta_{old}}(x)_2^2$ which can be expanded as terms in the d cosine metric between new and old embeddings, that cancel each other out. Also note for some weighting coefficients $\alpha_x, \alpha_{x'}$

$$\begin{aligned}\langle w_y^{new}, w_{y'}^{new} \rangle &= \sum_{x \in S_y} \sum_{x' \in S_{y'}} \alpha_x \alpha_{x'} \langle \Phi_{\theta_{new}}(x), \Phi_{\theta_{new}}(x') \rangle \\ &= \sum_{x \in S_y} \sum_{x' \in S_{y'}} \alpha_x \alpha_{x'} \langle \Phi_{\theta_{old}}(x), \Phi_{\theta_{old}}(x') \rangle = \langle w_y^{old}, w_{y'}^{old} \rangle\end{aligned}$$

so we have that prototypes evolve as isometries as well.

Prototype–sample approaches use those distances directly (Asadi et al., 2023) or via a contrastive loss (Cha et al., 2021). For example, classification algorithms may use something along the lines of $\hat{y} = \text{argmax}_y \langle z_q, w_y \rangle$ for some query embedding z_q, and as this only depends on relative distances then w_y need not be re-optimised and is unaffected by updates that are orthogonal to (Z_r) (Lemma 4). □

Finally, we prove our central theorem.

B.2 Proof of Central Theorem

Our central theorem states that under the JL embedding assumption that our distortion $\epsilon < 1$, and if we have a stationary point where the two losses $\mathcal{L}_{man}, \mathcal{L}_{orth}$ are zero, then the risk increase on any past task is upper-bounded by $C\epsilon$.

Isometry preserves distances: Our masked manifold isometry lemma shows us that at the fixed point of manifold loss, we have that for any $x_i, x_j \in \mathcal{T}_t$,. By Lemma 3 and the JL lemma (Johnson & Lindenstrauss, 1984) , for any $x_i, x_j \in \mathcal{T}_t$

$$\left| d_{\cos(\Phi_{\theta^{\text{new}}}(x_i), \Phi_{\theta^{\text{new}}}(x_j))} - d_{\cos(\Phi_{\theta^{\text{old}}}(x_i), \Phi_{\theta^{\text{old}}}(x_j))} \right| \leq \varepsilon.$$

Orthogonality implies no cross-task drift: Our next lemma 4 implies $\Phi_{\theta^{\text{new}}}(\mathcal{M}_t) \subset \text{span}(\mathcal{M}_t)^{\perp} \oplus \mathcal{M}_t$, so movements along the tangent space of our optimized fixed points lie in directions that do not distort internal geometry of previous tasks

Classifier Behavior Invariance: Given a cosine-based softmax

$$g_y(x) = \frac{\exp(-d_{\cos(w_y, \Phi_\theta(x))})}{\sum_{y'} \exp(-d_{\cos(w_{y'}, \Phi_\theta(x))})}$$

Lemma 5 keeps w_y fixed and Step 1 keeps distances within ε. Note that $g_y(x)$ does not at all depend on absolute positions of $\Phi_\theta(x)$ and w_y, but only on the pairwise distances. Moreover, because g_y is 1-Lipschitz in $d_{\cos}$, our quasi ϵ-isometry over all points in the convex hull that we reached in the isometry lemma gives us that the change in class probability is $O(\varepsilon)$.

Risk bound: We can use our bound in the change of class probability to then bound empirical risk. Specifically, take the standard classification indicator loss $\ell(\hat{y}, y) = \mathbf{1}\{\hat{y} \neq y\}$ and assume its Lipschitz constant w.r.t. g_y is $L \leq 1$. Then, we will show that

$$\left| \mathcal{R}_t^{\text{new}} - \mathcal{R}_t^{\text{old}} \right| \leq L\,\varepsilon \;=\; C\varepsilon,$$

where $C = L$ collects constant factors (se Fan et al., 2024and the Lipschitz-driven analysis of rehearsal Lopez-Paz & Ranzato, 2017; Bonicelli et al., 2022).

For every x, our model outputs a probability vector $g(x) = (g_1(x), \cdots, g_C(x))$ where $g_i(x)$ gives the ith class probablity for the latent vector x. Because of our probability bound, we have $g^{new}(x) - g^{old}(x)_\infty \leq \epsilon \forall x$, ignoring constant terms. Define the empirical risk on task t as

$$\mathcal{R}_t(\theta) = \frac{1}{|\mathcal{D}_t|} \sum_{(x,y)\in\mathcal{D}_t} 1_{\hat{y}(x)\neq y} = \frac{1}{|\mathcal{D}_t|} \sum_{(x,y)\in\mathcal{D}_t} 1_{\arg\max_j g_j(x)\neq y}$$

Now suppose our g_j's shift by at most ϵ. Either we have that they yield the same argmax, in which case our loss term for that x will be unchanged, or we swap the argmax between two classes whose probabilities were bounded by ϵ. That is, for a given output class y, we have that $|l_y(g) - l_y(g')| \leq 1_{g-g'_\infty \geq \epsilon} = g - g'_\infty / \epsilon$ which is a constant in ϵ asymptotically due to our infinity norm bound shown above. So, element-wise,

$$|1_{\hat{y}^{new}\neq y} - 1_{\hat{y}^{old}\neq y}| = |l_y(g) - l_y(g')| \leq g - g'_\infty \leq \epsilon$$

Which immediately gives the result for average empirical risk $|\mathcal{R}_t^{new} - \mathcal{R}_t^{old}| \leq O(\epsilon)$ up to a constant factor.

The above bound immediately results in a **uniform bound** for both **average joint multi-task empirical risk** and **catastrophic forgetting**. □