Multiband VAE: Latent Space Alignment for Knowledge Consolidation in Continual Learning

Kamil Deja[1] , **Paweł Wawrzyński**[1] , **Wojciech Masarczyk**[1] ,
Daniel Marczak[1] and **Tomasz Trzciński**[1,2,3]
[1]Warsaw University of Technology,
[2]Jagiellonian University,
[3]Tooploox
kamil.deja.dokt@pw.edu.pl

Abstract

We propose a new method for unsupervised generative continual learning through realignment of Variational Autoencoder's latent space. Deep generative models suffer from *catastrophic forgetting* in the same way as other neural structures. Recent *generative continual learning* works approach this problem and try to learn from new data without forgetting previous knowledge. However, those methods usually focus on artificial scenarios where examples share almost no similarity between subsequent portions of data – an assumption not realistic in the real-life applications of continual learning. In this work, we identify this limitation and posit the goal of generative continual learning as a knowledge accumulation task. We solve it by continuously aligning latent representations of new data that we call *bands* in additional latent space where examples are encoded independently of their source task. In addition, we introduce a method for controlled forgetting of past data that simplifies this process. On top of the standard continual learning benchmarks, we propose a novel challenging knowledge consolidation scenario and show that the proposed approach outperforms state-of-the-art by up to twofold across all experiments and the additional real-life evaluation. To our knowledge, Multiband VAE is the first method to show forward and backward knowledge transfer in generative continual learning.

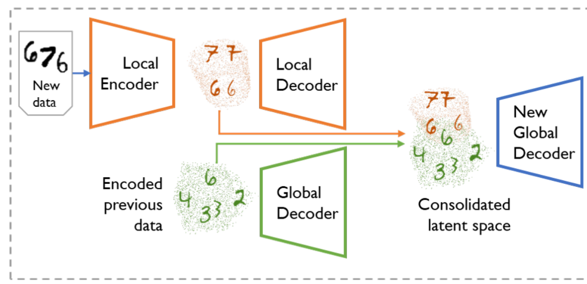


Figure 1: Overview of our Multiband VAE. With each new task, we first learn a *local* copy of our model to encode new data examples. Then we consolidate those with our current global decoder - main model which is able to generate examples from all tasks.

1 Introduction

Recent advances in generative models [Goodfellow *et al.*, 2014; Kingma and Welling, 2014] led to their unprecedented proliferation across many real-life applications. This includes high energy physics experiments at Large Hadron Collider (LHC) at CERN, where they are employed to speed up the process of particles collisions simulations [Paganini *et al.*, 2018; Deja *et al.*, 2020; Kansal *et al.*, 2021].

Those applications are possible, thanks to the main objective of generative methods, which is the modelling of complex data manifolds with simpler distributions. Unfortunately, this goal remains difficult to deliver in real-life situations where training data is presented to the model in separate portions, e.g., from consecutive periods of data gathering at CERN. The distributions of data within these portions often vary significantly, hence updating model with new examples leads to *catastrophic forgetting* of previous knowledge. In generative modelling this is observed through limited distribution of generated examples.

Generative continual learning methods aim to address these challenges usually in one of three ways: through regularization (e.g. [Nguyen *et al.*, 2018]), adjustment of the structure of a network to the next task (e.g. [Rao *et al.*, 2019]), or rehearsal of previously seen samples when training with new data (e.g. [Rebuffi *et al.*, 2017]). Nevertheless, practical applications of those methods are yet limited, so most of them focus on the artificial class-incremental (CI) training scenario. In this approach, consecutive portions of data (tasks) contain disjoint classes and share almost no similarity. While this is the most difficult scenario for discriminative models, we argue that the assumption of classes separation greatly simplifies the problem in generative modelling where task index might be used without reducing the method's generality (detailed discussion in the appendix).

Moreover, the assumption of task independence in CI scenario reduces the complexity of continual learning [Ke *et al.*, 2021]. Therefore, in this work, we postulate to investigate the adaptation of generative continual learning methods to the ever-changing data distribution. While, for the CI scenario, we expect no forgetting of previous knowledge, in other scenarios, where model is retrained with additional partially similar data, we should aim for performance improvement. This can be observed through *forward knowledge transfer* – higher performance on a new task, thanks to already incorporated knowledge, and *backward knowledge transfer* – better generations from previous tasks, when retrained on additional similar examples [Lopez-Paz and Ranzato, 2017].

Therefore, to simulate real-life conditions, we prepare a set of diversified continual learning scenarios with data splits following Dirichlet distribution, inspired by a similar approach in federated learning [Hsu *et al.*, 2019]. Our experiments indicate that this is indeed a more challenging setup for the majority of recent state-of-the-art continual generative models, which lack sufficient knowledge sharing between tasks.

To mitigate this problem, we propose a Multiband VAE. The core idea behind our method is to split the process of model retraining into two steps: (1) a local encoding of data from the new task into a new model's latent space and (2) a global rearrangement and consolidation of new and previous data. In particular, we propose to align local data representations from consecutive tasks through the additional neural network. In reference to the way how radio spectrum frequencies are allocated, we name data representations from different tasks *bands*. As in telecommunication, our goal is to limit interference between bands. However, we train our model to align parts that represent the same or similar data. To support knowledge consolidation between different bands, we additionally propose a controlled forgetting mechanism that enables the substitution of degraded reconstructions of past samples with new data from the current task.

The main contributions of this work are:

- A novel method for generative continual learning of Variational Autoencoder that counteracts catastrophic forgetting while being able to align even partially similar tasks at the same time.
- A simple method for controlled forgetting of past examples whenever a new similar data is presented.
- A novel knowledge consolidation training scenario that underlines limitations of recent state-of-the-art methods.

2 Related Works

Most of the works incorporating generative models in continual learning relate to generative rehearsal. In this technique, the base model is trained with a mixture of new data examples from the current task and recreation of previous samples generated by a generative model. This idea was first introduced by [Shin *et al.*, 2017], with Generative Adversarial Networks (GAN) trained with the self rehearsal method so-called Generative Replay (GR). [Lesort *et al.*, 2019] overview different generative models trained with the GR method. Our Multiband VAE is a direct extension to this technique.

Continual learning of generative models [Nguyen *et al.*, 2018] adapt regularization-based methods such as Elastic Weight Consolidation (EWC) [Kirkpatrick *et al.*, 2017], and Synaptic Intelligence (SI) [Zenke *et al.*, 2017] to the continual learning in generative models regularizing the adjustment of the most significant weights. The authors also introduce Variational Continual Learning (VCL), with adjustments in parts of the model architecture for each task.

In HyperCL, [von Oswald *et al.*, 2019] propose entirely different approach, where a hypernetwork generates the weights of the continually trained model. This yields state-of-the-art results in discriminative models task-incremental training but is also applicable to the generative models. In order to differentiate tasks, [Rao *et al.*, 2019] propose CURL that learns task-specific representation and deals with task ambiguity by performing task inference within the generative model. This approach directly addresses the problem of forgetting by maintaining a buffer for original instances of poorly-approximated samples and expanding the model with a new component whenever the buffer is filled. In BooVae, [Egorov *et al.*, 2021] propose an approach for continual learning of VAE with an additive aggregated posterior expansion. Several works train GANs in the continual learning scenarios either with memory replay [Wu *et al.*, 2018], with the extension to VAEGAN in Lifelong-VAEGAN by [Ye and Bors, 2020].

Continual learning with disentanglement In VASE by [Achille *et al.*, 2018], authors propose a method for continual learning of shared disentangled data representation. While encoding images with a standard VAE, VASE also seeks shared generative factors. A similar concept of mixed-type latent space was introduced in LifelongVAE [Ramapuram *et al.*, 2020], where it is composed of discrete and continuous values. In this work we also use a disentanglement method with binary latent space.

3 Method

In this section, we introduce Multiband VAE – a method for consolidating knowledge in a continually learned generative model. We propose to split generative replay training into two parts: (1) a local training that allows us to build a new data representations band in the latent space of VAE, and (2) global training where we attach a newly trained band to the already trained global model. As a part of the global training, we propose a controlled forgetting mechanism where we replace selected reconstructions from previous tasks with currently available data.

3.1 Knowledge Acquisition – Local Training

In the local training, we learn a new data representations band by training a VAE using only currently available data.

Let $\mathbf{x}_j^i$ denote the j-th sample of i-th task. Then, for given sample $\mathbf{x}_j^i$, and latent variable λ_j^i we use a decoder p_θ, which is trained to maximize posterior probability $p(\mathbf{x}_j^i|\lambda_j^i)$. To get the latent variable λ_j^i, we use encoder q_ϕ parametrized with weights vector ϕ that approximates probability $q(\lambda_j^i|\mathbf{x}_j^i)$.

To simplify the notation, let us focus on specific task i and drop the index. As in standard VAE, we follow optimization introduced by [Kingma and Welling, 2014] that maximizes the variational lower bound of log likelihood:

$$\max_{\theta,\phi} \mathbb{E}_{q(\lambda|\mathbf{x})}[\log p(\mathbf{x}|\lambda)] - D_{KL}(q(\lambda|\mathbf{x})\|\mathcal{N}(\vec{0}, I)). \quad (1)$$

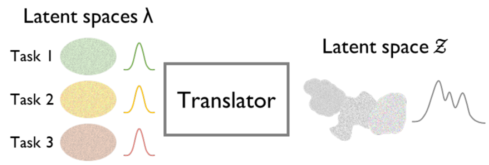


Figure 2: Our translator maps individual regularized latent spaces λ created by different local models to one global latent space $\mathcal{Z}$, where examples are stored independently of their source task.

where θ and ϕ are weights of encoder and decoder respectively. In the first task, this is the only part of the training, after which local decoder is remembered as a global one. In other cases we drop local decoder.

3.2 Shared Knowledge Consolidation

In the second – global part of the training, we align the newly trained band with already encoded knowledge. The simplest method to circumvent interference between bands is to partition the latent space of VAE and place new data representation in a separate area of latent space. However, such an approach limits information sharing across separate tasks and hinders forward and backward knowledge transfer. Therefore, in Multiband VAE we propose to align different latent spaces through an additional neural network that we call *translator*. Translator maps individual latent spaces which are conditioned with task id into the common global one where examples are stored independently of their source task, as presented in Fig 2.

To that end, we define a translator network $t_\rho(\lambda^i, i)$ that learns a common alignment of separate latent spaces λ^i conditioned with task id i to a single latent variable $\mathcal{Z}$, where all examples are represented independently of their source task. Finally, we propose a global decoder $p_\omega(\mathbf{x}|\mathcal{Z})$ that based on distribution approximated with latent variables $\mathcal{Z}$ learns to approximate original data distribution $\mathbf{x}$.

To counteract forgetting, when training translator and global decoder we use auto-rehearsal as in standard generative replay, with a copy of the translator and decoder frozen at the beginning of the task. As training pairs, we use combination of original images $\mathbf{x}$ with their encodings from local encoder λ, and for previous tasks, random values λ with generations $\mathbf{x}$ reconstructed with a frozen translator and global decoder. Fig. 3 presents the overview of this procedure.

We start translator training with a frozen global decoder, to find the best fitting part of latent space $\mathcal{Z}$ for a new band of data without disturbing previous generations. For that end we minimize the reconstruction loss:

$$\min_{\rho} \sum_{i=1}^{k} ||\mathbf{x}^i - p_\omega(t_\rho(\lambda^i, i))||_2^2, \tag{2}$$

where k is the number of all tasks.

Then, we optimize parameters of translator and global decoder jointly, minimizing the reconstruction error between outputs from the global decoder and training examples

$$\min_{\rho,\omega} \sum_{i=1}^{k} ||\mathbf{x}^i - p_\omega(t_\rho(\lambda^i, i))||_2^2, \tag{3}$$

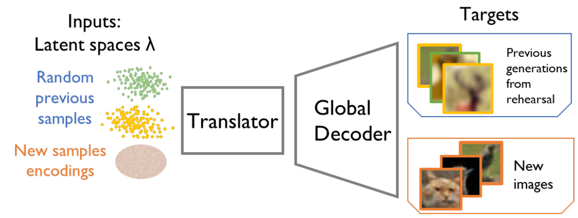


Figure 3: We train our translator and global decoder with new data encoded to latent space λ associated with original images, and samples of previous data generations generated in a standard rehearsal schema.

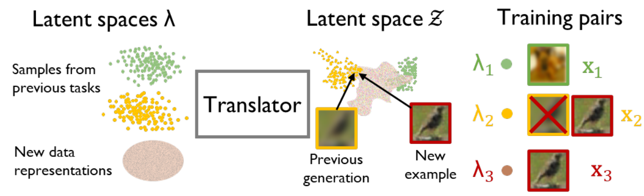


Figure 4: When creating rehearsal training pairs with generations from previous data examples, we calculate the similarity between sampled example and the closest currently available data sample in the common latent space $\mathcal{Z}$. If this similarity is above a given threshold, we allow forgetting of the previous reconstruction by substituting the target generation with a currently available similar image.

To generate new example t with Multiband VAE, we randomly sample task id $i \sim \mathcal{U}(\{1, \dots, k\})$, where k is the number of all tasks and latent representation $\lambda_t \sim \mathcal{N}(\vec{0}, I)$. These values are mapped with translator network to latent variable $\mathbf{z}_t$, which is the input to global decoder to generate $\mathbf{x}_t$. Therefore, translator and global decoder are the only models that are stored in-between tasks.

3.3 Controlled Forgetting

In a real-life scenario, it is common to encounter similar data examples in many tasks. In such a case, we would like our continuously trained model to refresh the memory of examples instead of combining vague, distorted memories with new instances. Therefore, we propose a mechanism for controlled forgetting of past reconstructions during the translator and global decoder joint training. To that end, when creating new training pairs, we compare representations of previous data reconstructions generated as new targets with representations of data samples from the current task in the common latent space $\mathcal{Z}$. If these representations are similar enough, we substitute previous data reconstruction with the current data sample as presented in Fig. 4.

More specifically, when training on task i, we first create a subset $\mathcal{Z}^i = t_\rho(q_\phi(\mathbf{x}^i), i)$ with representations of all currently available data in joint latent space $\mathcal{Z}$. Now, for each data sample $\mathbf{x}_j^l$ generated as a rehearsal target from previous task $l < i$ and random variable λ_j^l, we compare its latent representation $z_j = t_\rho(\lambda_j^l, j)$ with all elements of set $\mathcal{Z}^i$

$$sim(z_j) := \max_{z_q \in \mathcal{Z}^i} cos(z_j, z_q). \tag{4}$$

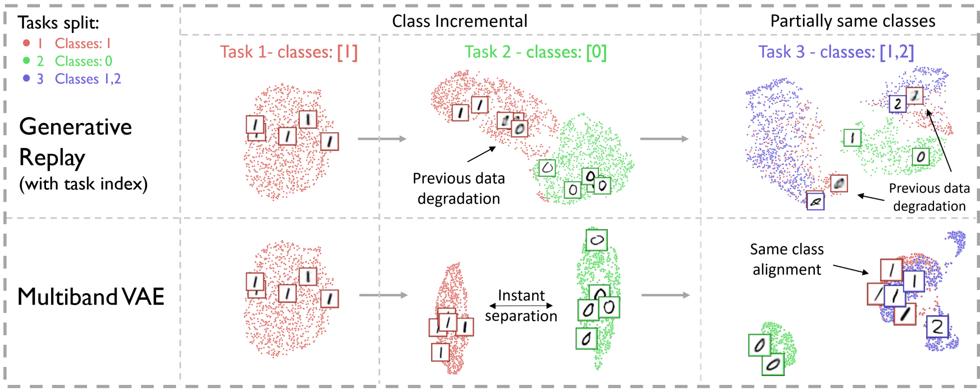


Figure 5: Visualization of latent space $\mathcal{Z}$ and generations from VAE in standard Generative Replay and our multiband training for the three tasks (different colors) in a case of entirely different new data distribution, and partially same classes. GR does not instantly separate data from different tasks, which results in the deformation of previously encoded examples. Contrary, our Multiband VAE can separate representations from different classes while properly aligning examples from the same new class if present.

If $sim(z_j) \geq \gamma$ we substitute target sampled reconstruction $\mathbf{x}_j^l$ with respective original image from $\mathbf{x}^i$. Intuitively, γ controls how much do we want to forget from task to task, with $\gamma = 0.9$ being a default value for which we observe a stable performance across all benchmarks.

4 Experiments

To visualize the difference between Generative Replay and Multiband VAE, in Fig. 5 we present a toy-example with the MNIST dataset limited to 3 tasks with data examples from 3 classes. When presented with data from a new distribution (different class in task 2), our method places a new band of data in a separate part of a common latent space $\mathcal{Z}$. On the other hand, the standard generative replay model learns to transform some of the previous data examples into currently available samples before it can distinguish them, even with additional conditioning on task identity. At the same time, when presented data with partially same classes as in task 3, our translator is able to properly align bands of data representations so that similar data examples (in this case ones) are located in the same area of latent space $\mathcal{Z}$ independently of the source task, without interfering with zeros and twos.

4.1 Evaluation Setup

For fair comparison, in all evaluated methods we use a Variational Autoencoder architecture similar to the one introduced by [Nguyen *et al.*, 2018], with nine dense layers. However, our Multiband VAE is not restricted to any particular architecture, so we also include experiments with a convolutional version. The exact architecture and training hyperparameters are enlisted in the appendix and code repository[1]. We do not condition our generative model with class identity since it greatly simplifies the problem of knowledge consolidation and applies to all evaluated methods. However, similarly to [Ramapuram *et al.*, 2020], we use additional binary latent space trained with Gumbel softmax [Jang *et al.*, 2016].

[1]https://github.com/KamilDeja/multiband_vae

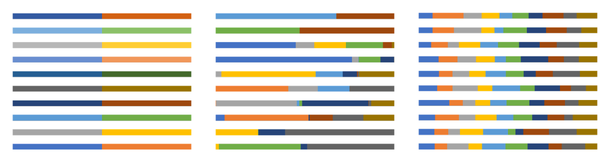


Figure 6: Class splits for different continual learning scenarios. In class incremental split each task consists of separate classes. For $\alpha = 1$ Dirichlet distribution, we have highly imbalanced splits with randomly occurring dominance of one or two classes. For higher values of parameter α, classes are split almost equally.

4.2 Evaluation

To assess the quality of our method, we conduct a series of experiments on benchmarks commonly used in continual learning (MNIST, Omniglot [Lake *et al.*, 2015]) and generative modeling – FashionMNIST [Xiao *et al.*, 2017]. Since the performance of VAE on diverse datasets like CIFAR is limited, in order to evaluate how our method scales to more complex data, we include tests on CelebA [Liu *et al.*, 2015]. For each dataset, we prepare a set of training scenarios designed to evaluate various aspects of continual learning. This is the only time we access data classes, since our solution is fully unsupervised.

To assess whether the model suffers from catastrophic forgetting, we run class incremental scenarios introduced by [Van de Ven and Tolias, 2019]. However, CI simplifies the problem of learning data distribution in the generative model's latent space since the identity of the task conditions final generations. Therefore, we also introduce more complex data splits with no assumption of independent task distributions. To that end, we split examples from the same classes into tasks, according to the probability $q \sim Dir(\alpha p)$ sampled from the Dirichlet distribution, where p is a prior class distribution over all classes, and α is a *concentration* parameter that controls similarity of the tasks, as presented in Fig. 6. In particular, we exploit the Dirichlet $\alpha = 1$ scenario, where the model has to learn the differences between tasks while

Num. tasks	Split-MNIST Class Incremental 5			MNIST Dirichlet $\alpha = 1$ 10			Split-Fashion MNIST Class Incremental 5			Fashion MNIST Dirichlet $\alpha = 1$ 10			CERN Class Inc. 5
Measure	FID ↓	Prec ↑	Rec ↑	FID ↓	Prec ↑	Rec ↑	FID ↓	Prec ↑	Rec ↑	FID ↓	Prec ↑	Rec ↑	Wass ↓
SI	129	77	80	153	75	76	134	28	24	140	21	19	21.1
EWC	136	73	82	120	79	83	126	25	25	137	24	22	29.7
Generative replay	120	79	87	254	70	65	96	43	58	133	35	43	11.1
VCL	68	85	94	127	78	80	104	30	32	138	21	20	24.3
HyperCL	62	91	87	148	78	75	108	46	33	155	35	21	7.8
CURL	107	95	77	181	84	74	86	47	64	83	46	56	16.8
Livelong-VAE	173	75	72	224	63	73	131	33	62	201	9	49	7.7
Livelong-VAEGAN	48	**98**	89	131	90	83	78	54	**79**	108	54	64	15.1
Multiband VAE	**24**	94	**97**	**41**	**92**	**96**	**61**	**66**	69	**82**	**62**	**65**	**6.6**
Multiband VAE (conv)	23	92	98	30	92	97	56	65	72	77	58	69	8.1

Table 1: Average FID and distribution Precision (Prec) and Recall (Rec) or Wassserstein distance between original and generated simulation channels, after the final task in different data incremental scenarios. Our method with vanilla architecture outperforms competing solution.

Num. tasks	Split-Omniglot Class Incremental 5			Split-Omniglot Class Incremental 20			Omniglot Dirichlet $\alpha = 1$ 20			FashionM→MNIST Class Incremental 10			MNIST→FashionM Class Incremental 10		
Measure	FID↓	Prec↑	Rec↑	FID↓	Prec↑	Rec↑	FID↓	Prec↑	Rec↑	FID↓	Prec↑	Rec↑	FID↓	Prec↑	Rec↑
SI	48	87	81	115	64	28	140	18	16	146	18	15	157	21	19
EWC	46	88	81	106	68	31	106	74	38	119	72	30	133	25	23
Generative replay	45	88	82	74	72	62	92	75	53	99	36	45	111	24	39
VCL	48	87	82	122	62	21	127	71	25	81	45	51	79	45	55
HyperCL	54	86	76	98	86	45	115	84	38	128	31	28	143	30	28
CURL	22	95	**95**	**31**	**96**	**92**	**26**	94	**92**	98	**69**	42	122	47	37
Lifelong-VAE	49	87	83	79	83	59	93	83	51	173	13	50	200	12	52
Lifelong-VAEGAN	31	96	90	71	83	70	63	85	78	127	34	61	91	52	**73**
Multiband VAE	**21**	**97**	93	33	95	86	41	**95**	83	**51**	65	**70**	**49**	**67**	**73**
Multiband VAE (conv)	12	98	96	24	95	91	24	96	91	49	68	70	49	70	70

Table 2: Average Fréchet Inception Distance (FID) and distribution Precision (Prec) and Recall (Rec) after the final task in different data incremental scenarios. In more challenging datasets Multiband VAE outperforms competing solutions.

consolidating representations for already known classes. In such a scenario we expect forward and backward knowledge transfer between tasks.

To measure the quality of generations from different methods, we use the Fréchet Inception Distance (FID) [Heusel *et al.*, 2017]. As proposed by [Bińkowski *et al.*, 2018], for simpler datasets, we calculate FID based on the LeNet classifier pre-trained on the whole target dataset. Additionally, we report the precision and recall of the distributions as proposed by [Sajjadi *et al.*, 2018]. As authors indicate, those metrics disentangle FID score into two aspects: the quality of generated results (Precision) and their diversity (Recall).

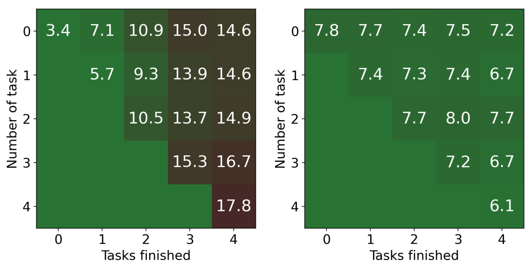
For each experiment, we report the FID, Precision, and Recall averaged over the final scores for each task separately. For methods that do not condition generations on the task index (CuRL and LifelongVAE), we calculate measures in comparison to the whole test set. The results of our experiments are presented in Tab. 1 and Tab. 2, where we show scores averaged over three runs with different random seeds.

To compare different continual-learning generative methods in a real-life scenario we also use real data from detector responses in the LHC experiment. Calorimeter response simulation is one of the most profound applications of generative models where those techniques are already employed in practice [Paganini *et al.*, 2018]. In our studies, we use a dataset of real simulations from Zero Degree Calorimeter in the ALICE experiment at CERN introduced by [Deja *et al.*, 2020], where a model is to learn outputs of 44×44 resolution energy depositions in calorimeter. Following [Deja *et al.*, 2020], instead of using FID, for evaluation, we benefit from the nature of the data and compare the distribution of real and generated channels – the sum of selected pixels that well describe the physical properties of simulated output. We report the Wasserstein distance between original and generated channels distribution to measure generations' quality. We prepare a continual learning scenario for this dataset by splitting examples according to their input energy, simulating changing conditions in the collider. In practice, such split lead to continuous change in output shapes with partial overlapping between tasks – similarly to what we can observe with Dirichlet based splits on standard benchmarks (see appendix for more details and visualisations).

As presented in Tab. 1, our model outperforms comparable methods in terms of quality of generated samples. Results of comparison on the Omniglot dataset with 20 splits (Tab. 2) indicate that for almost all of related methods, training with the data splits according to the Dirichlet $\alpha = 1$ distribution poses a greater challenge than the class incremental scenario. However, our Multiband VAE can precisely consolidate knowledge from such complex setups, while still preventing forget-

CelebA split Num. tasks	Class Incremental 5			Dirichlet $\alpha = 1$ 10			Dirichlet $\alpha = 100$ 10			Single split 1		
Measure	FID↓	Prec↑	Rec↑	FID↓	Prec↑	Rec↑	FID↓	Prec↑	Rec↑	FID↓	Prec↑	Rec↑
Separate models	103	31	21	105	24.5	7.6	109	28.4	10.6			
Generative Replay	105	23.4	14.9	109	14.6	7.4	102	17.2	11.6	88	35	30
Multiband VAE	95	28.5	23.2	93	33	22	89	36.2	28			

Table 3: Average FID, distribution Precision, and Recall after the final task on the CelebA dataset. Our Multiband VAE consolidates knowledge from separate tasks even in the class incremental scenario, clearly outperforming other solutions. With more even splits our method converges to the upper bound which is a model trained with full data availability.



Generative replay Multiband VAE

Figure 7: Comparison of Wasserstein distance ↓ between original simulation channels and generations from VAE trained with standard GR and our multiband training. Multiband VAE well consolidates knowledge with forward transfer (each row starts with better score) and backward knowledge transfer (improvement for some rows when retrained with more data). At the same time standard GR struggles to retain quality of generations on old tasks.

ting in CI scenario. This is only comparable to CURL that achieves this goal through additional model expansion. Experiments on more complex joint datasets, where examples are introduced from one dataset after another, indicate the superiority of Multiband VAE over similar approaches. In the real-life CERN scenario, our model also clearly outperforms other solutions. In Fig. 7 we present how generations quality for this dataset changes in standard generative replay and Multiband VAE, showing both forward and backward knowledge transfer in Multiband VAE.

Finally, we evaluate our model with a more complex dataset – CelebA with over 200 000 images of celebrity faces in 64x64 resolution. Based on annotated features, we split the dataset into 10 classes based on the hair color/cover (blonde, black, brown, hat, bald or gray). In Tab. 3 we show the results of experiments with this dataset split in class incremental and Dirichlet scenarios. For class incremental scenario, Multiband VAE learns to separate bands of examples from different tasks with disjoint distributions, while results improve if in training scenario model is presented with more similar examples. In the latter case, with Dirichlet $\alpha = 100$ splits, our model reaches the quality of the upper bound, which is a standard Variational Autoencoder trained with full access to all examples in the stationary training.

Ablation study The main contribution of this work is a multiband training procedure, yet we also introduce several mechanisms that improve knowledge consolidation. Tab. 4 shows how those components contribute to the final score.

Modification	FID↓
Generative replay	254
+ Two step training	64
+ Translator	53
+ Binary latent space	44
+ Controlled forgetting	41
+ Convolutional model	30

Table 4: Ablation study on the MNIST dataset with Dirichlet $\alpha = 1$ distribution. Average FID after the last task.

4.3 Memory Requirements and Complexity

The memory requirements of Multiband VAE are constant and equal to the size of the VAE with an additional translator, which is a small neural model with 2 fully connected layers. When training on the new task, our method requires additional temporary memory for the local model freed when finished. This is contrary to similar methods (HyperCL, VCL, CURL) which have additional constant or growing memory requirements. Computational complexity of our method is the same as for methods based on generative rehearsal (VCL, LifelongVAE, Lifelong-VAEGAN). In experiments, we use the same number of epochs for all methods, while for Multiband VAE we split them between local and global training.

5 Conclusion

In this work, we propose a new method for unsupervised continual learning of generative models. We observe that the currently employed class-incremental scenario simplifies the continual learning of generative models. Therefore, we propose a novel, more realistic scenario, with which we experimentally highlight the limitations of state-of-the-art methods. Finally, we introduce a new method for continual learning of generative models based on the constant consolidation of VAE's latent space. To our knowledge, this is the first work that experimentally shows that with continually growing data with even partially similar distribution, we can observe both forward and backward performance improvement. Our experiments on various benchmarks and with real-life data show the superiority of Multiband VAE over related methods, with upper-bound performance in some training scenarios.

Acknowledgments

This research was funded by Foundation for Polish Science (grant no POIR.04.04.00-00-14DE/18-00 carried out within the Team-Net program co-financed by the European Union under the European Regional Development Fund) and National Science Centre, Poland (grant no 2018/31/N/ST6/02374 and 2020/39/B/ST6/01511).

References

[Achille *et al.*, 2018] Alessandro Achille, Tom Eccles, et al. Life-long disentangled representation learning with cross-domain latent homologies. *Advances in Neural Information Processing Systems*, 31, 2018.

[Bińkowski *et al.*, 2018] Mikołaj Bińkowski, Danica J Sutherland, et al. Demystifying mmd gans. In *ICLR*, 2018.

[Deja *et al.*, 2020] Kamil Deja, Jan Dubiński, et al. End-to-end sinkhorn autoencoder with noise generator. *IEEE Access*, 9:7211–7219, 2020.

[Egorov *et al.*, 2021] Evgenii Egorov, Anna Kuzina, et al. Boovae: Boosting approach for continual learning of vae. *Advances in Neural Information Processing Systems*, 34, 2021.

[Goodfellow *et al.*, 2014] Ian Goodfellow, Jean Pouget-Abadie, et al. Generative Adversarial Networks. In *NeurIPS*, 2014.

[Heusel *et al.*, 2017] Martin Heusel, Hubert Ramsauer, et al. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in neural information processing systems*, pages 6626–6637, 2017.

[Hsu *et al.*, 2019] Tzu-Ming Harry Hsu, Hang Qi, et al. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019.

[Jang *et al.*, 2016] Eric Jang, Shixiang Gu, et al. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[Kansal *et al.*, 2021] Raghav Kansal, Javier Duarte, et al. Particle cloud generation with message passing generative adversarial networks. *Advances in Neural Information Processing Systems*, 34, 2021.

[Ke *et al.*, 2021] Zixuan Ke, Bing Liu, et al. Achieving forgetting prevention and knowledge transfer in continual learning. *Advances in Neural Information Processing Systems*, 34, 2021.

[Kingma and Welling, 2014] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *ICLR*, 2014.

[Kirkpatrick *et al.*, 2017] James Kirkpatrick, Razvan Pascanu, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

[Lake *et al.*, 2015] Brenden M Lake, Ruslan Salakhutdinov, et al. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[Lesort *et al.*, 2019] Timothée Lesort, Hugo Caselles-Dupré, et al. Generative models from the perspective of continual learning. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019.

[Liu *et al.*, 2015] Ziwei Liu, Ping Luo, et al. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[Lopez-Paz and Ranzato, 2017] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30:6467–6476, 2017.

[Nguyen *et al.*, 2018] Cuong V Nguyen, Yingzhen Li, et al. Variational continual learning. In *International Conference on Learning Representations*, 2018.

[Paganini *et al.*, 2018] Michela Paganini, Luke de Oliveira, et al. Calogan: Simulating 3d high energy particle showers in multilayer electromagnetic calorimeters. *Physical Review D*, 97(1):014021, 2018.

[Ramapuram *et al.*, 2020] Jason Ramapuram, Magda Gregorova, et al. Lifelong generative modeling. *Neurocomputing*, 404:381–400, 2020.

[Rao *et al.*, 2019] Dushyant Rao, Francesco Visin, et al. Continual unsupervised representation learning. *Advances in Neural Information Processing Systems*, 32, 2019.

[Rebuffi *et al.*, 2017] S. Rebuffi, A. Kolesnikov, et al. iCaRL: Incremental Classifier and Representation Learning. In *CVPR*, 2017.

[Sajjadi *et al.*, 2018] Mehdi SM Sajjadi, Olivier Bachem, et al. Assessing generative models via precision and recall. *arXiv preprint arXiv:1806.00035*, 2018.

[Shin *et al.*, 2017] Hanul Shin, Jung Kwon Lee, et al. Continual Learning with Deep Generative Replay. In *NeurIPS*, 2017.

[Van de Ven and Tolias, 2019] Gido M Van de Ven and Andreas S Tolias. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019.

[von Oswald *et al.*, 2019] Johannes von Oswald, Christian Henning, et al. Continual learning with hypernetworks. In *International Conference on Learning Representations*, 2019.

[Wu *et al.*, 2018] Chenshen Wu, Luis Herranz, et al. Memory replay gans: Learning to generate new categories without forgetting. In *NeurIPS*, 2018.

[Xiao *et al.*, 2017] Han Xiao, Kashif Rasul, et al. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms, 2017. arXiv:1708.07747.

[Ye and Bors, 2020] Fei Ye and Adrian G Bors. Learning latent representations across multiple data domains using lifelong vaegan. In *European Conference on Computer Vision*, pages 777–795. Springer, 2020.

[Zenke *et al.*, 2017] Friedemann Zenke, Ben Poole, et al. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pages 3987–3995. PMLR, 2017.