Visual Intelligence

RESEARCH Open Access

Continual learning with Bayesian model based on a fixed pre-trained feature extractor

Yang Yang[1,2], Zhiying Cui[1,3], Junjie Xu[1,3], Changhong Zhong[1,3], Wei-Shi Zheng[1,3] and Ruixuan Wang[1,2,3*]

Abstract

Deep learning has shown its human-level performance in various applications. However, current deep learning models are characterized by catastrophic forgetting of old knowledge when learning new classes. This poses a challenge such as in intelligent diagnosis systems where initially only training data of a limited number of diseases are available. In this case, updating the intelligent system with data of new diseases would inevitably downgrade its performance on previously learned diseases. Inspired by the process of learning new knowledge in human brains, we propose a Bayesian generative model for continual learning built on a fixed pre-trained feature extractor. In this model, knowledge of each old class can be compactly represented by a collection of statistical distributions, e.g., with Gaussian mixture models, and naturally kept from forgetting in continual learning over time. Unlike existing class-incremental learning methods, the proposed approach is not sensitive to the continual learning process and can be additionally well applied to the data-incremental learning scenario. Experiments on multiple medical and natural image classification tasks reveal that the proposed approach outperforms state-of-the-art approaches that even keep some images of old classes during continual learning of new classes.

Keywords: Continual learning, Bayesian model, Generative approach, Fixed feature extractor

1 Introduction

Deep learning models, particularly convolutional neural networks (CNNs), have demonstrated human-level performance in various applications, such as in healthcare [1–4], surveillance [5–8], and machine translation [9, 10]. However, particularly in the healthcare domain, most intelligent diagnosis systems are limited to the diagnosis of only one or a few diseases and cannot be easily extended once deployed, and therefore cannot diagnose all diseases of certain tissues or organs (e.g., skin or lung) as medical specialists do. Since collecting data of all (e.g., skin or lung) diseases is challenging due to various reasons (e.g., privacy and limited data sharing), it is impractical to train an intelligent system diagnosing all diseases at once. One possible solution is to make the intelligent system have the continual or lifelong learning ability, such that it can continually learn to diagnose more and more diseases without resourcing (or resourcing few) original data of previously learned diseases [11]. Such continual learning of new classes may also appear in other applications such as in automated retail stores [12]. However, current intelligent models are characterized by catastrophic forgetting of old knowledge when learning new classes [13–15].

Researchers have recently proposed multiple types of continual learning approaches to reduce catastrophic forgetting of old knowledge particularly in deep learning models [16–20]. The overall objective is to help the updated classifier accurately recognize both new and old classes, when only data of new classes and few (or even no) data of old classes are available during classifier updating. However, almost all existing approaches modify the feature extraction part of the classifiers either in parameter values or in structures during continual learning of new classes. In contrast, humans seem to learn new knowledge

*Correspondence: wangruix5@mail.sysu.edu.cn
[1] School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China
[2] Department of Network Intelligence, Peng Cheng Laboratory, Shenzhen, China
Full list of author information is available at the end of the article

by adding memory of the learned new information without modifying the (e.g., visual) perceptual pathway. Therefore, one possible cause of catastrophic forgetting in existing models is the change in the feature extraction part (corresponding to the perceptual pathway in human brains) when learning new knowledge. With this consideration, we propose a generative model for continual learning built on a fixed pre-trained feature extractor, which is different from all existing (discriminative) models. The generative model can naturally prevent knowledge of each old class from being forgotten without storing original images of old classes or regenerating synthetic images during continual learning. Experiments on two skin disease classification tasks and two natural image classification tasks demonstrate that the proposed approach outperforms state-of-the-art approaches which even keep some images of old classes during continual learning. The proposed approach provides a new direction for the investigation of continual learning, i.e., exploring effective ways to represent and store knowledge of each class based on a fixed but powerful feature extractor. Note that this work is an extension of the previous conference publication [21] in the following aspects.

1) In the methodology, the statistical distribution of each feature output for each class is extended from the parametric Gaussian mixture model (GMM) to the non-parametric Kernel density estimation (KDE). Empirical evaluation indicates that both methods are effective in the representation of statistical distributions for the proposed method.
2) The application and effectiveness of the proposed method are extended from general class-incremental learning to two new scenarios, i.e., few-shot continual learning and data-incremental learning.
3) Empirical comparisons with very recent state-of-the-art continual learning methods were performed.
4) The effect of feature output size on continual learning performance is extensively evaluated.
5) Qualitative evaluation was performed on the separability of distributions between classes.
6) Two natural image datasets, CIFAR100 and CUB200, were employed to further support the effectiveness and superiority of the proposed method. In addition, extensive sensitivity study of hyper-parameters was performed on these new datasets.
7) A comprehensive literature review is included.

2 Related work

There are typically two types of continual learning problems, task-incremental and class-incremental. Task-incremental learning presumes that one model is incrementally updated to solve an increasing number of tasks, often with multiple tasks sharing a common feature extractor but having task-specific classification heads. Task identification is presumed to be available during inference, i.e., users know which classification head should be applied when predicting the class label of new test data. This setting is impractical for intelligent diagnosis systems where old and new diseases need to be diagnosed together. In contrast, class incremental learning presumes that one model is incrementally updated to predict more and more classes sharing a single classification head. This approach is more relevant to the continual learning of new diseases. Thus, our study focuses on the class-incremental learning problem. Existing approaches to the two types of continual learning can be roughly divided into four groups: regularization-based, expansion-based, distillation-based, and regeneration-based.

Regularization-based approaches often estimate model components (e.g., kernels in CNNs) crucial for old knowledge, and try to change them as little as possible with the help of regularization loss terms when learning new knowledge [16, 22–28]. The importance of each model parameter can be measured by the sensitivity of the loss function to changes in the model parameter, as in the elastic weight consolidation (EWC) method [16], or by the sensitivity of the model output to small changes in the model parameter, as in the memory aware synapses (MAS) method [29]. The importance of each kernel in a CNN model can be measured based on the magnitude of the kernel (e.g., L2 norm of the kernel matrix), as in PackNet [27]. Regularization may also be designed to ensure that certain gradient-based measurement is not increased during learning for the stored data of old tasks in the memory, as in GEM [30] and its extensions A-GEM [31], or to update the model only in certain feature subspace that is irrelevant to the old knowledge, as in LOGD [32] and Adam-NSCL [33]. A new model architecture can be designed such that part of the model is allowed to be more easily updated than the others as in AANet [34].

Regularization-based approaches could help models keep old knowledge in the first few rounds of continual learning where little new knowledge needs to be learned. However, it would become increasingly difficult to continually learn new knowledge, particularly at later rounds of continual learning, because more and more kernels in CNNs become crucial and therefore should be kept unchanged to increase old knowledge.

To make models more flexible in learning new knowledge, expansion-based approaches are developed to modify model structures by adding new kernels, layers, or even sub-networks when learning new knowledge [17, 35–43]. For example, Aljundi et al. [35] proposed employing an additional network for a new task and training an expert model to make decisions about which network to use during inference. It turns a class-incremental learning problem into a task-incremental problem at the cost of additional parameters. As another example, Yoon et al. proposed a dynamically expandable network (DEN) [17] by selectively retraining the network and expanding kernels at each layer if necessary. Most expansion-based and regularization-based approaches were initially proposed for task-incremental learning, although some of them (e.g., EWC) can be extended for class-incremental learning. One exception is the recently proposed state-of-the-art method DER [40] and the simple DER [41], where the feature extractor trained at each round of continual learning is aggregated into the updated classifier over class-incremental learning. In addition to CNN backbones, the Transformer backbone was also recently used in class-incremental learning [42], where learnable task-specific input tokens at the last self-attention block of the Transformer are learned at each round.

In comparison, distillation-based approaches can be directly applied to continual learning of new classes by distilling knowledge from the old classifier (for old classes) to the new classifier (for both new and old classes) while learning new knowledge [18, 19, 44–48], where the old knowledge is often implicitly represented by soft outputs of the old classifier with a stored small amount of old images and/or new classes of images as the inputs. A distillation loss is added to the original cross-entropy loss during training the new classifier, where the distillation loss helps the new classifier have similar relevant output compared to the output of the old classifier for any input image. The well-known methods include the learning without forgetting (LwF) [18], incremental classifier and representation learning (iCaRL) [19], and the end-to-end incremental learning (End2End) [46]. More recently, the distillation has been extended to intermediate CNN layers, either by keeping feature map activation unchanged as in the learning without memorizing (LwM) [49], or by keeping the spatial pooling unchanged along the horizontal and vertical directions as in PODNet [50], or by keeping the normalized global pooling unchanged at last convolutional layers as in learning a unified classifier incrementally by rebalancing (UCIR) [51].

These distillation-based methods achieve state-of-the-art performance for the class-incremental learning problem. However, such methods would become insufficient with continual learning of more classes, either because stored old data become too small to be representative for each old class, or because the outputs of the old classifier with new classes of data as inputs cannot represent the knowledge of old classes due to underlying differences between new classes and each old class.

In addition, regeneration-based approaches have also been proposed particularly when no old data are available while learning new classes. The basic idea is to train an auto-encoder [52–54] or generative adversarial network (GAN) [20, 55–57] to produce enough realistic data for each old class when learning new classes. The potential issue is that fine-grained lesion features may not be well learned by the generative model, which would result in unsatisfying synthetic data when updating the intelligent diagnosis system. Different from all the existing approaches, we propose a simple but effective generative model that is based on a fixed pre-trained feature extractor and does not store any old data.

3 A generative model for continual learning

The proposed method is inspired by two interesting findings in neuroscience. One finding is that most infants cannot form episodic memory before 3 years old [58–60], and the other finding is that humans continually form memory from infants to elderly people [61]. One hypothetical explanation is that the visual pathway in younger infants' brains might be rapidly changing with daily visual stimuli from their surroundings and then become firm with little change after approximately 3 years of age. Humans can continually learn new visual knowledge through their whole lives, probably because they form new memories about the new knowledge, but without changing the visual pathway which works as a visual feature extractor. This could help explain why current deep learning models are characterized by catastrophic forgetting of old knowledge, i.e., model parameters or model structures from the feature extractor part are always changed to some extent in almost all continual learning approaches. With this consideration, we propose a human-like continual learning framework, i.e., first pre-training a feature extractor, then fixing the feature extractor and forming new memory for new knowledge. In the following part, we will introduce one general way to pre-train the feature extractor, one statistical method to represent the memory, and one Bayesian model to predict the class of any new (test) data after continual learning each time.

3.1 Fixed pre-trained feature extractor

An ideal feature extractor should output two different feature vectors if two input images were visually different, meanwhile visually more similar inputs should result in more similar feature vectors from the feature extractor. The visual feature extractor (i.e., visual pathway) in younger infants is probably taught in a certain self-supervised way, although the mechanism of self-supervision in the infant brain has not been explicitly understood [59]. While it is worth exploring various self-supervised learning approaches (e.g., auto-encoder) to train a feature extractor, here we leave the self-supervision exploration for future work, and adopt a simpler but widely used approach, i.e., pre-training a CNN classifier with a relatively large number of images whose classes or domains are relevant but different from those in the task of interest and then using the pre-trained CNN feature extractor (often consisting of all the convolutional layers; Fig. 1, top row) for the continual learning classification task of interest. It is expected that the pre-trained feature extractor would probably be powerful enough to discriminate different input images in the task of interest. Experiments in this study verify that even such a simple approach using a fixed pre-trained feature extractor can already help significantly reduce catastrophic forgetting of old knowledge with the proposed generative approach. It is worth noting that, during continual learning of new classes in the classification task of interest, the pre-trained feature extractor is fixed and not updated. The knowledge of each learned new class is represented and stored as described in the following subsection.

3.2 Memory formation

Different from state-of-the-art continual learning approaches, which often store a small number of original images for each old class, the proposed approach stores not original images but the statistical information of each class based on the feature extractor outputs of all training images belonging to the class. Here, each element of the output feature vector is assumed to represent a certain type of visual feature. Then, based on the class of training images, the distribution of each feature is estimated and collected together to form the memory of the knowledge of the specific class (Fig. 1, second and third rows, each row for one class). Formally, denote by $D_c = \{\mathbf{x}_i, i = 1, \dots, N_c\}$ the set of training images for class c, $\mathbf{z}_i = [z_{i1}, z_{i2}, \dots, z_{ik}, \dots, z_{iK}]^\mathsf{T}$ the L_2-normalized output feature vector from the feature extractor for the input image $\mathbf{x}_i$, and $\mathbf{f} = [f_1, f_2, \dots, f_k, \dots, f_K]^\mathsf{T}$ the vector of random variables representing the output of the feature extractor, and then the statistical distribution of the k-th feature f_k for class c can be represented by a probability density distribution $p(f_k|c, D_c)$,

$$p(f_k|c, D_c) = g\big(\{z_{ik}, i = 1, \dots, N_c\}\big), \quad \forall k \in \{1, \dots, K\}, \quad (1)$$

where $g(\cdot)$ could be any appropriate distribution estimator. Here a Gaussian mixture model (GMM) with a small number of S components is adopted to represent $g(\cdot)$ for its simplicity. Since each Gaussian component can be compactly represented by its mean and standard deviation, in total, only $2 \cdot S \cdot K$ numbers are stored in the memory to represent the knowledge of each class. D_c is omitted from $p(f_k|c, D_c)$ in the following for simplicity. Figure 2 demonstrates representative distributions of eight randomly selected features for three classes, which clearly indicates that there exist differences in the statistical distributions of individual features between classes. Note that here, a K-dimensional Gaussian is not used to represent the distribution of the K-dimensional feature vector because the multi-dimensional Gaussian not only requires storing more parameters per class in the memory, but also more importantly, requires many more training data to obtain good estimate of the mean and covariance matrix for each class. However, in general, only hundreds of training images are available for each class, and such a small number of images are often far from sufficient to estimate the mean and covariance matrix of the high-dimensional feature vector.

3.3 Bayesian model for prediction

Based on the statistical distributions of visual features for each class, we propose a generative classification model with the Bayesian rule for prediction. Given a test image

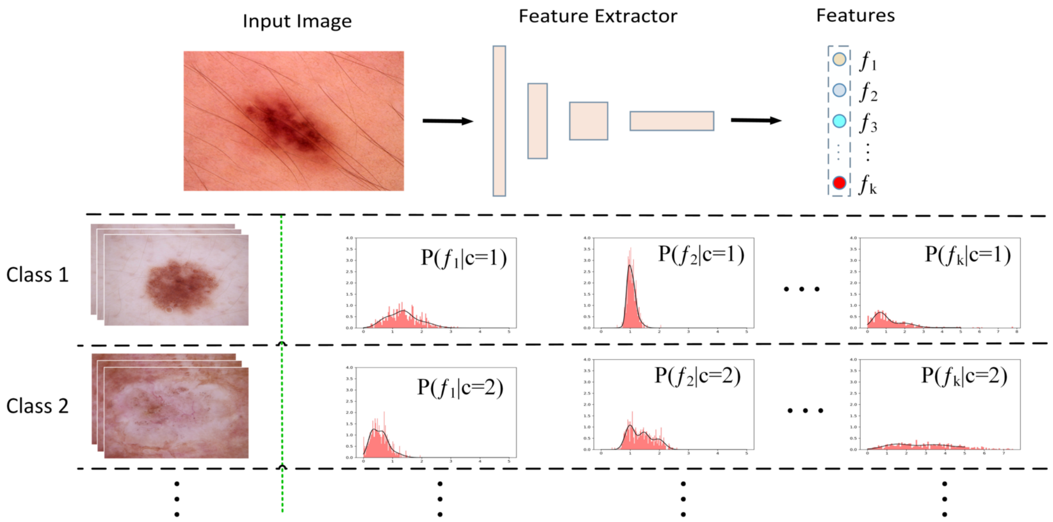

Figure 1 Fixed pre-trained feature extractor (top) and memory formation (middle to bottom). The feature extractor is pre-trained and fixed during continual learning. The memory of each class is represented by a set of statistical distributions over features

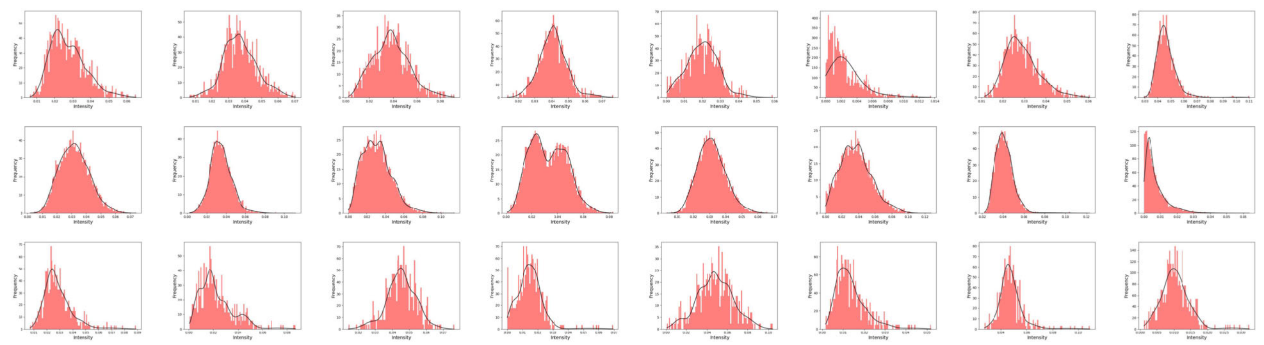

Figure 2 Representative statistical distributions of eight randomly selected features for three classes from the CIFAR100 dataset. Each class is from a different continual learning round. Each row is for one class, and each column represents the statistical distributions of the same feature for three different classes. In each subfigure, the curve represents the GMM approximating the histogram of the feature based on all training images of the class

$\mathbf{x}_j$, denote by $\mathbf{z}_j = [z_{j1}, z_{j2}, \ldots, z_{jk}, \ldots, z_{jK}]^\top$ the corresponding output from the feature extractor, and $p(c|\mathbf{z}_j)$ the probability of the test image belonging to class c. Then, based on the Bayes rule, we can calculate

$$p(c|\mathbf{z}_j) = \frac{p(\mathbf{z}_j|c) \cdot p(c)}{\sum_{m=1}^{M} p(\mathbf{z}_j|m) \cdot p(m)}, \tag{2}$$

where M is the number of classes learned thus far. Considering that potential correlations between certain feature components are probably caused by co-occurring visual parts of a specific class of objects, it can be assumed that different feature components f_k's are conditionally independent given specific class c. Then, the logarithm of Equation (2) gives

$$\log p(c|\mathbf{z}_j) = \sum_k \log p(f_k = z_{jk}|c) + \log p(c) - \alpha, \tag{3}$$

where $\alpha = \log \sum_m p(\mathbf{z}_j|m)p(m)$ can be considered a constant for different classes. In Equation (3), the likelihood function value $p(f_k = z_{jk}|c)$ for each feature element k can be directly obtained based on the previously stored knowledge $p(f_k|c)$ (Equation (1)) in the memory. The prior $p(c)$ for class c can be simply estimated based on the ratio of the number of training images for this class over the total number of training images of all learned classes thus far, i.e., $p(c) = N_c / \sum_m N_m$. Note that in this case, the number of training images for each class needs to be stored in the memory such that $p(c)$ can be easily updated when new classes' knowledge is learned as above (Equation (1)). Based on Equation (3), the class of the test image $\mathbf{x}_j$ would be directly predicted as the one with the highest value of $\log p(c|\mathbf{z}_j)$ over all classes learned thus far. The proposed method is summarized in Algorithms 1 and 2.

Algorithm 1 Incremental learning of class c

```
Input: $D_c = \{\mathbf{x}_i, i = 1, \ldots, N_c\}$ \\ the set of training images for class $c$
Output: $p(f_k|c, D_c)$ \\ the statistical distribution of the $k$-th feature
$F \leftarrow$ the pre-trained feature extractor;
for $i = 1, 2, \ldots, N_c$ do
    $\mathbf{z}_i = L_2(F(\mathbf{x}_i))$; \\ extract features of each image and perform $L_2$ normalization
end for
for $k = 1, 2, \ldots, K$ do
    $p(f_k|c) = g(\{z_{ik}, i = 1, \ldots, N_c\})$; \\ perform distribution estimator (e.g., GMM) for each feature
end for
```

Algorithm 2 Bayesian model for prediction

```
Input: $\mathbf{x}_j$ \\ a test image
Output: predicted class label $y^*$
$F \leftarrow$ the pretrained feature extractor;
$\mathbf{z}_j = L_2(F(\mathbf{x}_j))$; \\ extract features of the test image and perform $L_2$ normalization
for $c = 1, 2, \ldots, C$ do \\ $C$: number of learned classes
    $\log p(\mathbf{z}_j|c) = 0$;
    for $k = 1, 2, \ldots, K$ do
        $\log p(\mathbf{z}_j|c) = \log p(\mathbf{z}_j|c) + \log p(f_k = z_{jk}|c)$; \\ estimate log likelihood
    end for
    $\log p(c|\mathbf{z}_j) = \log p(\mathbf{z}_j|c) + \log p(c)$; \\ from Equation (3), omitting constant $\alpha$
end for
$y^* \leftarrow \arg\max_{c=1,\ldots,C} \log p(c|\mathbf{z}_j)$
```

The advantages of the proposed approach over existing continual learning approaches are clear. First, the knowledge of each old class is statistically represented by the set of likelihood functions (Equation (1)) and compactly stored in the memory. Our approach does not need to store any original images for each old class in the memory and instead only stores the Gaussian mixture model (GMM) parameters (together with the dataset size for each class). When learning a set of new classes at each continual learning stage, the stored GMM parameters are collected from the memory and then used by the Bayesian rule for class probability prediction. In contrast, almost all the strong baselines (including those state-of-the-art methods) need to store a small subset of old images per class (e.g., a total of 2000 images) in the memory, and each stored old image will be used together with the new classes of images as input to the model for model training. Therefore, old knowledge will not be forgotten over continual learning of new classes. In comparison, old knowledge will be inevitably and gradually forgotten over multiple rounds of continual learning in existing approaches, either due to the changes in the feature extractor or due to the reduced number of original images to be stored in the limited memory. Second, the final performance of the proposed approach over multiple rounds of continual learning is not affected by the number of learning rounds and the number of new classes added in each round. In contrast, in existing approaches, more rounds of continual learning with smaller number of new classes added each time would often lead to worse classification performance at later round of continual learning. Therefore, the proposed approach is more robust to various continual learning conditions with little forgetting of old knowledge.

4 Experimental evaluation

4.1 Experimental setup

The proposed approach was extensively evaluated on a diverse group of image datasets, including two natural image datasets and two medical skin image datasets. Medical image datasets are normally quite different from natural image datasets in terms of appearance and textures. Each dataset is briefly summarized as follows (also see Table 1).

CIFAR100 [62] is a dataset of natural images of daily objects, including various animals, plants, outdoor and indoor scenes, and vehicles. It consists of 100 classes, 500 training images and 100 test images for each class. The size of each image is quite small, only 32 × 32 pixels.

Table 1 Datasets for diverse image classification tasks, from natural to medical images, small scale to relatively large scale, and general to fine-grained classifications. Image size varies greatly in Skin40. [120, 500] means that image width and height vary in the range between 120 and 500 pixels

Dataset	#Classes	Training set	Test set	Image size
CIFAR100	100	50,000	10,000	32 × 32
CUB200	190	5694	5496	[120, 500]
Skin7	7	8010	2005	600 × 450
Skin40	40	2000	400	[260, 1640]

CUB200 [63] is a dataset of bird species typically used for fine-grained recognition [64]. It contains 11,788 images for 200 categories, approximately 60 images per class. Note that although part locations, binary attributes and bounding boxes are provided, only image-level species labels are used in our classification experiments. For experiments on the CUB200 and the CIFAR100 datasets, the adopted fixed feature extractor is directly from the pre-trained CNN model (e.g., VGG-Net [65] or ResNet101 [66]) based on the ImageNet dataset [67], where the last fully connected layer is removed and the remaining part is used for the feature extractor. Considering that the pre-trained feature extractor is based on the ImageNet dataset, the TinyImageNet dataset, which is a subset of ImageNet and often adopted for continual learning, was not used in the experiments here. Additionally, for a similar reason, the ten common classes (Black footed Albatross, Laysan Albatross, Sooty Albatross, Indigo Bunting, American Goldfinch, Ruby throated Hummingbird, Green Violetear, Blue Jay, Dark eyed Junco and Red breasted Merganser) between CUB200 and ImageNet were removed from CUB200, resulting in 190 classes for continual learning on the CUB200 dataset.

Skin7 [68] is a skin lesion dataset from the challenge of dermoscopic image classification held by the International Skin Imaging Collaboration (ISIC) in 2018. It consists of 7 disease categories; each image is of size 600 × 450 pixels. This dataset presents severe class imbalance, with the largest class 60 times larger than the smallest class.

Skin40 is a subset of 193 classes [69] of skin disease images collected from the Internet. The 40 classes with relatively more number of images (60 images per class) were chosen from the 193 classes to form the Skin40 dataset, while the remaining 153 classes (10 to 40 images per class) were used to train a CNN classifier whose final classification layer was then removed to form the fixed feature extractor in most experiments relevant to the Skin7 and the Skin40 datasets. Notably, there is no overlap between the 153 classes (for training the feature extractor in advance) and the classes in Skin7 and Skin40 (for continual learning evaluation).

During training the feature extractor based on the 153 skin image classes, each image was randomly cropped within the scale range [0.8, 1.0] and then re-sized to 224 × 224 pixels, followed by random horizontal and vertical flipping. The mini-batch stochastic gradient descent (batch size 32) was used to train the feature extractor, with an initial learning rate of 0.01 and then divided by 10 at the 35th, 70th, and 105th epoch, respectively. Weight decay (0.0005) and momentum (0.9) were also applied. The feature extractor was trained for 120 epochs with observed convergence.

In each experiment, multiple rounds of continual learning were performed, with a few (e.g., 2, 5, 10) new classes to be learned at each round. After each round of continual learning, the mean class recall (MCR) over all classes learned thus far was calculated. For each experiment, the average and standard deviation of MCR over five runs were reported, where the five orders of classes to be continually learned were fixed and used in the proposed approach and baseline methods. Unless otherwise mentioned, ResNet-101 was used as the backbone for the feature extractor, and the dimension of feature vector K was 2048 and the number (S) of Gaussian components in each GMM model was empirically set to 2 based on a small validation set for each dataset.

4.2 Effectiveness of the generative model

This section evaluates the effectiveness of the proposed approach by comparing it with recent state-of-the-art strong baselines, including iCaRL [19], end-to-end incremental learning (End2End) [46], learning a unified classifier incrementally via rebalancing (UCIR) [51], distillation and retrospection (DR) [47], learning without forgetting (LwF) [18], adaptive aggregation networks (AANets) [34], and separated softmax for incremental learning (SSIL) [44]. The suggested hyper-parameter settings in the original work were adopted unless otherwise mentioned. In each round of continual learning, for the iCaRL, End2End, DR, UCIR, AANets, and SSIL, which need a certain amount of old data, the number of images stored (i.e., memory size) for all old classes is respectively 2000 on CIFAR100, 400 on CUB200, 50 on Skin7, and 100 on Skin40. The memory size was chosen such that the stored number of images for each class was only a small portion of the original training images at the last round of continual learning. For each experiment, an upper-bound result was also reported (e.g., Fig. 3 and Fig. 4, green star) by training a non-continual classifier with all classes of training data.

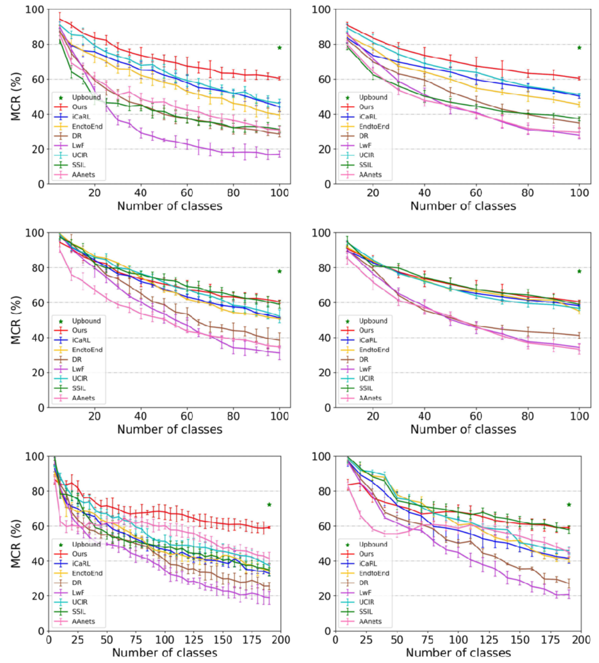

Figure 3 Performance comparison on natural image datasets. First row: learning 5 (left) and 10 (right) classes at each round on CIFAR100, with CNN classifiers initially trained from scratch for each baseline method. Second row: learning 5 (left) and 10 (right) classes at each round on CIFAR100, with CNN classifiers initially fine-tuned from the same pre-trained feature extractor as that used in the proposed approach. Third row: learning 5 (left) and 10 (right) classes at each round on CUB200, with CNN classifiers initially fine-tuned from the same pre-trained feature extractor. *X*-axis in each sub-figure represents the accumulated number of learned classes in the corresponding continual learning task

All the baselines were previously evaluated by initially training a CNN classifier from scratch before starting continual learning. Therefore, the proposed approach was first compared to the baselines where each initial CNN classifier for each baseline was trained from scratch. In this case, as clearly displayed in Fig. 3 (first row), the proposed approach outperforms all the strong baselines. Compared to the strongest baseline, the absolute improvement by the proposed approach is respectively 14.3% (first row, left) and 9.3% (first row, right) at the last round of continual learning when learning 5 and 10 new classes in each round, respectively. However, the comparison could be considered unfair because the proposed approach used a pre-trained feature extractor while the baselines did not. In consideration of this point, the proposed approach was also compared with the baselines where each initial CNN classifier at the first learning round was fine-tuned from the same pre-trained feature extractor for each baseline method. In this case, Fig. 3 (second row; also see Table 2) demonstrates that although some strong baselines have slightly better performance in the first several rounds of continual learning, the performance of most baselines decreases faster than the proposed approach particularly when more continual learning rounds are involved (left), and the proposed approach performs either best (left) or equivalently well (right) compared to the strong baselines at the last several rounds of continual learning. The more rounds of continual learning there are, the larger final gap between the proposed approach and the strong baselines at the last round of continual learning. This is further confirmed on the CUB200 dataset where more rounds of continual learning occurred compared to on CIFAR100 with the same settings. As illustrated in Fig. 3 (third row, left; also see Table 3), the classification performance of the proposed approach decreases much more slowly than that of all the baselines, and the proposed approach quickly outperforms all the strong baselines after a few rounds of continual learning, although the same pre-trained feature extractor was initially used to fine-tune each CNN classifier in each baseline. It can be consistently observed that the gap between the strongest baseline and the proposed ap-

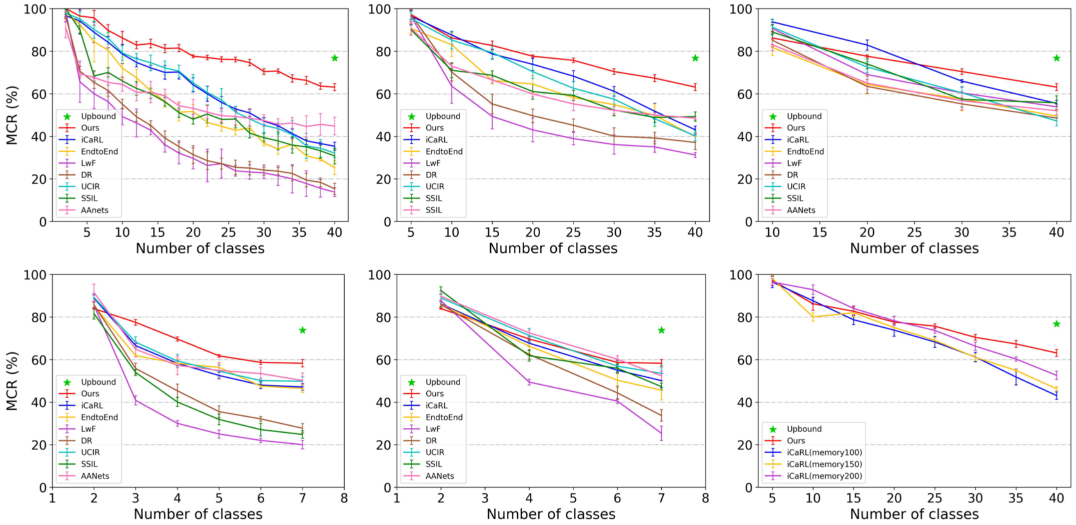

Figure 4 Performance comparison on medical image datasets. First row (from left): learning 2, 5, 10 classes at each round on Skin40. Second row (from left): learning 1 and 2 classes at each round on Skin7, and comparison with iCaRL with varying memory size on Skin40. *X*-axis in each sub-figure represents the accumulated number of learned classes in the corresponding continual learning task

Table 2 Performance comparison on the CIFAR100 dataset. Five classes are learned at each round with CNN classifiers initially fine-tuned from the same pre-trained feature extractor

Class #	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100
LwF	98.83	90.67	84.89	79.87	74.62	69.01	63.67	60.25	56.37	53.6	49.92	47.06	42.78	40.94	37.21	34.22	33.74	32.9	31.79	31.14
DR	98.35	93.91	87.54	82.74	75.88	73.52	69.44	65.1	60.59	58.69	54.57	53.46	48.7	46.03	45.33	43.88	43.46	41.09	39.49	38.43
ICaRL	98.49	91.71	88.1	84.25	80.64	76.4	75.32	71.9	70.31	67.62	65.5	63.19	61.75	59.79	58.32	57.5	56.87	54.03	52.8	51.2
EndtoEnd	99.44	93.85	90.74	86.32	85.25	82.63	79.28	74.3	71.03	66.81	65.63	61.87	60.43	59.2	56.17	54.33	53.67	53.07	51.53	50.81
UCIR	99.11	91.93	88.56	85.76	84.41	79.84	78.95	76.57	74.59	71.5	68.86	67.42	65.17	63.75	61.54	58.37	57.6	56.02	54.36	52.16
SSIL	97.6	93.6	90.27	82.4	79.76	79.57	76.91	76.17	74.13	72.9	72.25	69.15	68.46	66.07	65.59	63.89	62.51	61.69	60.39	58.9
AANets	90.27	76.03	72.16	67.22	62.68	58.78	57.03	53.64	51.87	50.47	46.89	43.54	42.657	40.91	40.08	39.01	38.49	36.94	35.32	34.72
Ours	94.4	90.9	86.26	83.55	82.24	77.46	75.37	73.55	72.08	70.48	69.34	67.43	66.55	65.62	63.42	63.26	62.31	62.43	61.62	60.47

Table 3 Performance comparison on the CUB200 dataset. Ten classes are learned at each round with CNN classifiers initially fine-tuned from the same pre-trained feature extractor

Class #	10	20	30	40	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190
LwF	97.15	84.95	77.62	64.79	61.71	58.49	57.56	51.23	46.39	44.85	40.61	37.89	35.54	30.34	28.88	25.33	23.92	20.65	20.96
DR	97.16	87.34	79.68	67.84	64.79	62.5	60.86	58.2	52.14	50.14	51.34	44.44	42.57	38.31	35.5	35.63	29.71	29.46	27.11
ICaRL	97.34	89.38	85.21	79.15	71.9	67.72	65.69	59.7	58.86	57.55	55.46	52.69	51.13	50.25	47.79	45.69	44.22	42.57	41.59
EndtoEnd	98.25	92.13	88.55	87.8	77.97	74.7	73.27	67.26	65.24	60.53	61.22	57.08	53.39	51.44	51.75	46.21	43.28	41.7	40.74
UCIR	98.62	91.85	91	88.98	75.94	74.77	70.73	67.25	65.19	63.5	62.01	59.12	58.3	57.93	50.39	49.4	47.46	46.2	45.94
SSIL	99.64	93.13	88.1	85.69	74.49	72.99	71.45	70.35	69.39	67.81	66.4	67.67	65.11	64.16	62.5	61.04	61.12	59.28	57.93
AANets	83.32	66.33	58.31	55.58	55.51	57.09	60.51	60.79	60.15	58.69	60.62	58.98	57.25	55.77	54.17	51.85	50.64	47.93	45.03
Ours	83.55	84.71	76.04	73.52	71.65	69.92	66.8	67.46	68.14	67.92	66.91	65.28	63.18	62.18	61.22	60.8	60.49	59.03	59.19

proach becomes increasingly larger with more rounds of continual learning.

As expected, Fig. 3 also shows that the final-round performance of the proposed approach is not affected by the number of new classes to be learned in each round. The performance of the proposed approach is approximately 60% in MCR in the last round of continual learning on both the CIFAR100 and the CUB200 datasets, regardless of how many rounds of continual learning are performed and how many new classes are learned in each round. In comparison, the final performance of each baseline becomes worse with more rounds of continual learning (correspondingly with a smaller number of new classes to be learned at each round; also see Fig. 5, right). In addition, the performance

Table 4 Performance comparison on the SKin40 dataset. Two classes are learned at each round with CNN classifiers initially fine-tuned from the same pre-trained feature extractor

Class #	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40
LwF	98.75	65.62	60	56.25	49.25	46.25	42.85	36.08	32.07	29.62	26.28	27.15	23.75	23.22	22.8	21.48	20	17.77	15.52	13.78
DR	96.67	70.59	65.33	61.47	55.13	49.18	45.17	39.18	35.16	31.4	28.51	27.14	25.41	25.04	24.13	23.6	22.55	19.43	18.33	15.32
ICaRL	96.67	94.17	88.87	84.17	78.66	74.72	72.14	70.02	70.18	64.17	59.97	55.91	52.81	50.95	47.11	45	41.27	37.87	36.57	35.33
EndtoEnd	98.33	92.5	84.45	80.42	72	67.5	61.43	56.45	51.3	51.67	46.52	44.72	42.82	43.93	36.67	34.06	36.28	30.83	29.19	25.25
UCIR	97.77	95.14	90.13	86.13	79.12	76.43	74.55	72.13	70.61	65.13	60.53	57.11	51.33	48.17	45.12	43.63	40.78	35.51	34.33	32.12
SSIL	100	90.12	68.33	70.51	66.42	62.51	60.01	56.25	51.11	48.33	50.54	47.92	48.08	41.43	39.33	37.81	35.88	35.34	33.16	30.75
AAnet	91	69.35	67.6	65.29	64.39	60.67	60.76	58.94	54.36	53.15	51.34	49.69	49.25	48.49	47.42	45.59	46.28	44.57	45.6	44.78
Ours	100	96.5	95.67	89.75	86.2	82.83	83.57	81.25	81.44	77.6	77	76.17	76.08	74.64	70.4	70.62	67.18	66.28	63.58	63.1

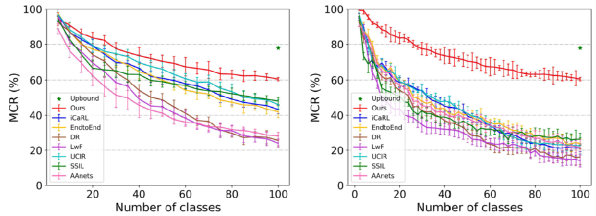

Figure 5 Continual learning performance on CIAF100 with smaller memory (left) or more learning rounds (right). Left: learning 5 classes at each round with only 1000 images stored in the memory. Right: learning 2 classes at each round. The CNN classifiers were initially fine-tuned from the same pre-trained feature extractor

Table 5 Performance comparison on the SKin7 dataset. One class is continually learned at each round with CNN classifiers initially fine-tuned from the same pre-trained feature extractor

Class #	2	3	4	5	6	7
LwF	85.54	40.82	30.15	25.56	22.27	20.09
DR	85.71	55.83	45.36	35.53	32.16	27.73
ICaRL	88.86	66.53	58.22	52.5	47.98	47.09
EndtoEnd	85.26	61.83	58.25	56.29	47.53	46.44
UCIR	89.2	68.12	59.33	54.52	50.17	49.77
SSIL	81.57	53.92	40.05	31.84	27.06	24.79
AANets	91.51	64.81	57.25	54.88	53.4	50.14
Ours	83.91	77.47	69.63	61.72	58.64	58.44

of existing state-of-the-art methods is seriously affected by the number of old data stored in the memory. As Fig. 5 (left) demonstrates, performance of the strong baselines significantly decreases when the memory size is reduced from 2000 to 1000, while the proposed approach is not affected by the memory size at all. These results clearly support that the proposed Bayesian generative model is effective in reducing the catastrophic forgetting of old knowledge, probably because the knowledge of old classes is kept unchanged in the form of statistical distribution over continual learning.

The proposed approach works effectively not only on the natural image datasets, but also on medical datasets. As depicted in Fig. 4 (also see Table 4 and Table 5), with a certain number of new classes to be continually learned at each round on both the Skin40 and Skin7 datasets, the proposed approach always performs better than all the strong baselines particularly at later rounds of continual learning, although the same pre-trained feature extractor was used to initially fine-tune the CNN classifier for each baseline method (which is the default setting in the following sections). Even with more images of old classes stored for the representative strong baseline iCaRL, the proposed approach still performs better (Fig. 4, second row, last), again supporting that the proposed approach is more effective in preventing old knowledge from being forgotten.

4.3 Generalizability and robustness of the generative model

The proposed approach is a general framework that can employ different feature extractor backbones or use different ways to represent and store old knowledge in specific applications. As Table 6 shows, the proposed approach performs consistently better than strong baselines on all the four datasets with different feature extractor backbones (Vgg19, ResNet18, ResNet34 and ResNet101), supporting that the proposed approach is not limited to specific feature extractor structures.

Additionally, the proposed approach is not limited to the specific Gaussian mixture model (GMM) representation for old knowledge. For example, in addition to the parametric GMM model, the well-known nonparametric Kernel density estimation (KDE) was also used to approximate the statistical distribution of each feature output, where the kernel width is empirically determined based on a small validation set from each dataset. Figure 6 depicts that the proposed approach based on KDE works equivalently well compared to that based on GMM for the representation of old knowledge. Because the proposed approach is not limited to specific ways to represent old knowledge, a potentially more effective representation of old knowledge would increase the performance of continual learning by the proposed approach. This remains to be explored in future work.

Table 6 Performance on various feature extractor backbones. The results after the last round of continual learning were reported, with 1 (Skin7), 5 (Skin40), 5 (CIFAR100), and 5 (CUB200) new classes per round

Dataset	VGG19				ResNet18				ResNet34				ResNet101			
	LwF	iCaRL	IR	Ours	LwF	iCaRL	IR	Ours	LwF	iCaRL	IR	Ours	LwF	iCaRL	IR	Ours
CIFAR100	20.7	35.3	39.4	**48.9**	26.4	48.3	43.7	**57.2**	27.9	49.6	44.5	**61.8**	31.1	51.2	52.2	**60.5**
CUB200	11.3	28.1	26.6	**46.3**	14.2	30.1	32.2	**54.9**	14.4	31.2	33.7	**56.5**	15.2	33.6	35.2	**59.2**
Skin7	18.9	39.7	38.3	**46.5**	19.8	44.3	46.2	**55.6**	20.1	46.9	48.3	**56.8**	20.1	47.1	49.8	**58.4**
Skin40	27.4	33.6	32.5	**52.8**	30.4	41.8	37.1	**61.9**	31.1	42.3	39.5	**62.8**	31.2	43.1	40.2	**63.1**

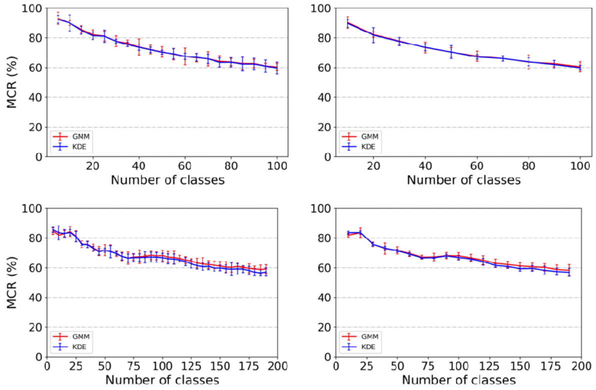

Figure 6 Performance comparison between GMM and KDE models for old knowledge representation respectively on CIFAR100 (first row) and CUB200 (second row), when learning 5 (first column) or 10 (second column) new classes at each round

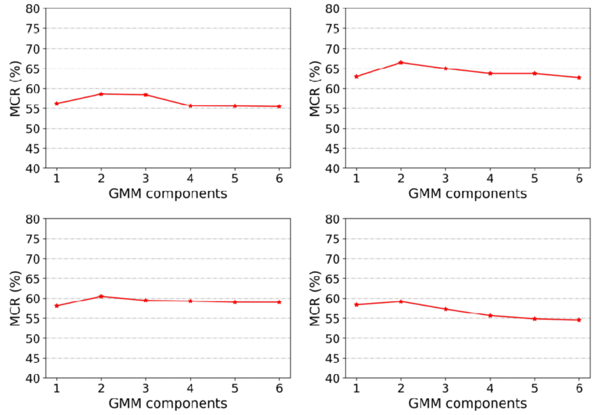

Figure 7 Stable performance with varying GMM components on Skin7 (first row, left), Skin40 (first row, right), CIFAR100 (second row, left) and CUB200 (second row, right). At each round of continual learning, 1, 5, 10, and 10 new classes were continually learned, respectively, on Skin7, Skin40, CIFAR100, and CUB200 datasets

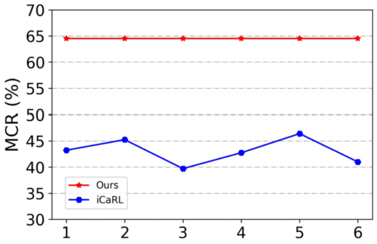

Figure 8 Final-round learning performance with different class orders during continual learning on Skin40. Five new classes were continually learned at each round. Different numbers in the *x*-axis represent different sequences of class orders

To evaluate the robustness of the generative model, the GMM with varying numbers of Gaussian components and different orders of classes to be continually learned were tried during continual learning. As clearly displayed in Fig. 7, the generative model works stably with different numbers of Gaussian components in the GMM on all four datasets, although GMM with two components works slightly better than GMM with fewer or more components. Note that the proposed approach still outperforms all the strong baselines when the number of GMM components is larger than two. In addition, with six different orders of classes to be continually learned, the performance of the proposed approach does not change at the last round of continual learning, while the performance of the representative iCaRL baseline method clearly varies with different class orders (Fig. 8). This is because knowledge of each previously learned old class is compactly stored and is not changed throughout the whole process of continual learning by the proposed approach. In comparison, almost all the strong baseline methods inevitably update the feature extractor during continual learning, which would then change the representation of each stored old data and further change the representation of old knowledge, differently with different orders of classes to be learned.

4.4 Wide application scenarios of the generative model

One reason to explore continual learning techniques is due to the difficulty in collecting data from all classes. Such difficulty may cause another two challenging problems, few-shot continual learning where only a few number of training images are available for each new class at each round of continual learning, and data-incremental continual learning where the classifier would be updated continuously with new data of existing classes (rather than with

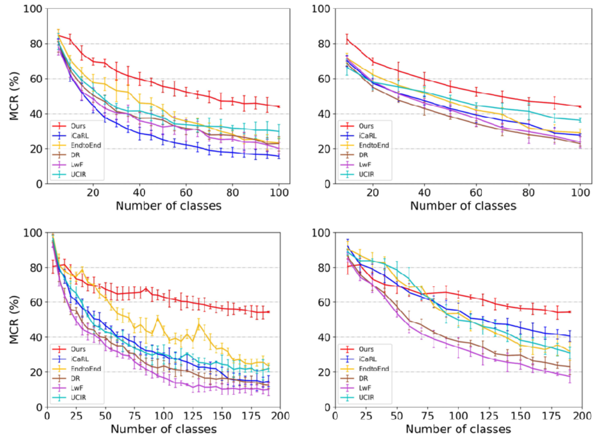

Figure 9 Few-shot continual learning performance on CIFAR100 and CUB200. Only 10 training images are available for each new class during continual learning on CIFAR100 (first row) and CUB200 (second row), with 5 (first column) or 10 (second column) new classes learned each time

data of new classes). To check whether the proposed approach works effectively in the scenario of few-shot continual learning, in the experiment, only 10 training images for each new class were provided to update the model for each method. The memory size was set to 200 for all the baseline methods, which need to store a small set of original images. As illustrated in Fig. 9, while the strongest baseline method changes with varying datasets and numbers of learned new classes per round, the best performance is always from the proposed approach. This clearly supports that the proposed approach still works effectively even if only a limited number of data are available during continual learning.

For data-incremental continual learning, with a few new images provided for each class at each round, it can be observed that the performance of the proposed approach increases over rounds of continual learning, while the representative iCaRL method cannot effectively improve its learning performance shortly after the memory used in iCaRL becomes full (Fig. 10). This is probably because the proposed approach can naturally update the representation of each class with more data, without discarding the information of data that previously appeared. In comparison, the performance of the updated classifier by existing methods often depends on the limited original data stored in memory and the data that appeared more recently. Note that the existing methods would perform even worse without storing old data by memory. This experiment demonstrates that the proposed approach can be used to handle two types (i.e., class-incremental and data-incremental) of continual learning, while existing methods can only handle the class-incremental learning task.

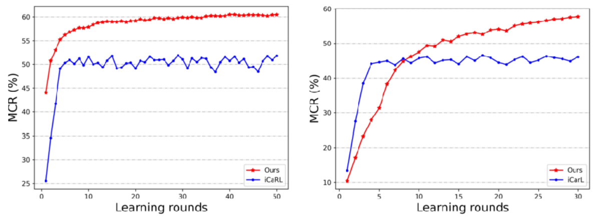

Figure 10 Data-incremental continual learning performance on CIFAR100 (left) and CUB200 (right). All classes are available from the beginning, but only 10 (CIFAR100) or 1 (CUB200) new images are available for each class at each round of continual learning. Memory size is respectively set 2000 (left) and 400 (right) for the representative method iCaRL

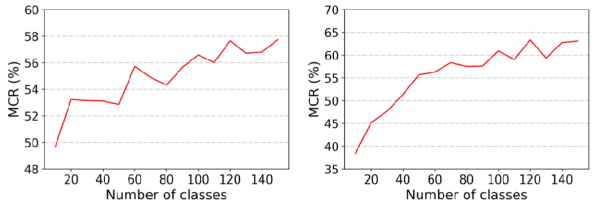

Figure 11 Effect of feature extractor on continual learning. More classes (*X*-axis) used to train feature extractors result in better performance on Skin7 (left) and Skin40 (right). Note that the classes used to train the feature extractor are not overlapped with the set of classes to be learned during continual learning

4.5 Effect of the feature extractor

The proposed approach is based on a fixed pre-trained feature extractor. To confirm that better feature extractors would help the generative model perform better in continual learning, the original 153 classes of skin image data used for training the feature extractor (before starting to continually learn new skin disease classes) were gradually reduced to only 10 classes, each time using such a reduced number of classes to train the feature extractor, and then the performance of the proposed approach at the last round of continual learning on both the Skin7 and Skin40 datasets was calculated. As illustrated in Fig. 11, more classes used for training the feature extractor generally result in better performance of the proposed approach. The feature extractor trained by more classes of data would probably have learned to extract more types of features and therefore could be more generalizable to a new but relevant domain. Consistent with the observation and explanation, when the feature extractor is fixed by random parameter weights (i.e., without any training), the classifier in continual learning showed the worst performance (MCR is 21% on Skin7, 6% on Skin40; not shown in Fig. 11). These results strongly suggest that exploring better ways to obtain a better feature extractor would further improve performance of the generative model in continual learning.

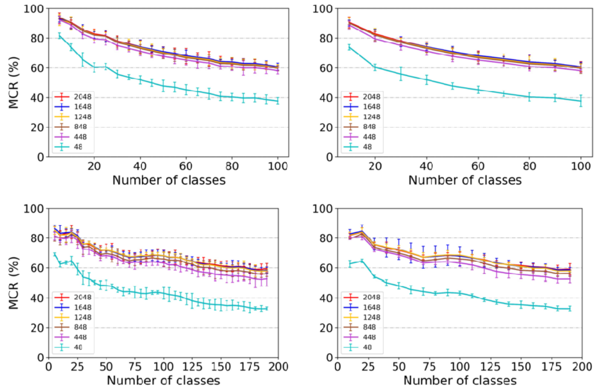

Figure 12 Continual learning performance with different numbers of feature outputs. First row: learning 5 (left) and 10 (right) new classes at each round on CIFAR100. Second row: learning 5 (left) and 10 (right) classes at each round on CUB200. Curves in different colors correspond to different numbers of feature outputs during continual learning

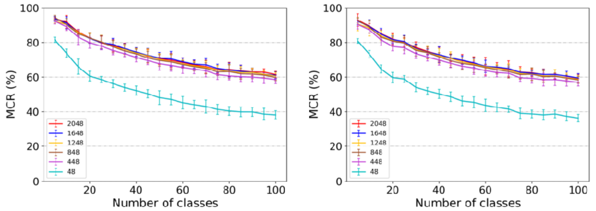

Figure 13 Continual learning performance with different numbers of feature outputs based on alternative strategies to obtain features. Left: the ResNet101 backbone was modified by adding one additional fully connected layer on top of the last convolutional layer and then pre-trained with the ImageNet dataset, and the feature vector output of the added FC layer is used for subsequent analysis. Right: PCA was applied to reduce the output vector of the originally pre-trained ResNet101 with the ImageNet dataset, where the principal components were obtained based on the ImageNet dataset as well

Another factor in the feature extractor that potentially affects continual learning performance is the size of the feature extractor outputs. In general, more outputs could represent more types of feature information and therefore help the proposed approach represent richer information in each class. To confirm this hypothesis, varying numbers of feature outputs were randomly sampled from the original 2048 outputs, and then continual learning was performed based on the randomly sampled feature outputs. As demonstrated in Fig. 12, the performance clearly decreases when the number of feature extractor outputs used is fewer than 1000, with more drops in performance corresponding to fewer outputs. Interestingly, the performance changes little when the number of feature extractor outputs decreases from the original 2048 to 1248. This may be because some outputs are highly correlated, such that removal of some of the outputs would not affect the representation power by the remaining outputs. This provides an opportunity to use a relatively smaller number of outputs for knowledge representation when memory resources are very limited. A similar finding was obtained when directly pre-training a feature extractor with different numbers of output features or applying PCA to the high-dimensional feature vector output of a pre-trained and fixed feature extractor (Fig. 13).

5 Conclusion

In this study, we propose a Bayesian generative model for continual learning of new classes. The model does not update the feature extractor but generates statistical information to represent the knowledge of each class. Without storing any original data, the generative model can prevent knowledge of each old class from being forgotten and outperforms existing state-of-the-art approaches, which often store a small amount of old data. The model is not limited to any specific feature extractor backbone and the ways to represent statistical information, and the final-round performance is not affected by the process of continual learning such as the number of new classes to be learned each time or the number of rounds of continual learning. In addition to continually learning new classes, the model can also consistently improve the classification performance by continuously learning from new data of existing classes. This study suggests a new direction to solve the catastrophic forgetting issue in continual learning, i.e., exploring effective ways to represent knowledge based on certain fixed but powerful pre-trained feature extractor. Better pre-trained feature extractor could also be explored to further improve the performance of the generative approach.

Acknowledgements
We appreciate Xuchen Xie, Defeng Zhao and Junze Ye from Sun Yat-sen University for discussions and comments. This article is a revised and expanded version of a paper entitled Continual Learning with Bayesian Model Based on a Fixed Pre-trained Feature Extractor, which was presented at the 24th International Conference of Medical Image Computing and Computer Assisted Intervention (MICCAI 2021), held at Strasbourg, France, from September 27 to October 1, 2021.

Funding
This work is supported in part by the National Natural Science Foundation of China (Grant Nos. 62071502, U1811461), and the Guangdong Key Research and Development Program (Grant No. 2020B1111190001).

Abbreviations

CNNs, convolutional neural networks; DEN, dynamically expandable network; DER, dynamically expandable representation; End2End, end to end; EWC, elastic weight consolidation; GAN, generative adversarial network; GEM, gradient episodic memory; GMM, gaussian mixture model; iCaRL, incremental classifier and representation learning; KDE, kernel density estimation; LOGD, layerwise optimization by gradient decomposition; LwF, learning without forgetting; MAS, memory aware synapses; PODNet, pooled outputs distillation network; UCIR, unified classifier incrementally via rebalancing.

Availability of data and materials

All datasets supporting this study are open source. The ImageNet dataset can be downloaded at https://www.image-net.org/download.php and the CUB200 dataset is at https://www.vision.caltech.edu/datasets/cub_200_2011/. The Skin7 dataset is available at https://doi.org/10.7910/DVN/DBW86T. The Skin198 dataset can be obtained by contacting liujiaxuan@mail.nankai.edu.cn.

Declarations

Competing interests

The authors declare no competing interests.

Author contributions

YY, ZC and JX collected data and performed the experiments. YY, ZC, and CZ contributed significantly to experiment analysis and manuscript preparation. YY and RW performed the data analyses and wrote the manuscript. WSZ helped perform the analysis with constructive discussions. RW revised the manuscript. All authors read and approved the final manuscript.

Author details

[1]School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China. [2]Department of Network Intelligence, Peng Cheng Laboratory, Shenzhen, China. [3]Key Laboratory of Machine Intelligence and Advanced Computing, MOE, Guangzhou, China.

Received: 17 September 2022 Revised: 3 December 2022
Accepted: 24 February 2023 Published online: 08 May 2023

References

1. Ardila, D., Kiraly, A. P., Bharadwaj, S., Choi, B., Reicher, J. J., Peng, L., Tse, D., Etemadi, M., Ye, W., Corrado, G., Naidich, D. P., & Shetty, S. (2019). End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography. *Nature Medicine 25*(6), 954–961.
2. De Fauw, J., Ledsam, J. R., Romera-Paredes, B., Nikolov, S., Tomasev, N., Blackwell, S., Askham, H., Glorot, X., O'Donoghue, B., Visentin, D., Van Den Driessche, G., Lakshminarayanan, B., Meyer, C., Mackinder, F., Bouton, S., Ayoub, K., Chopra, R., King, D., Karthikesalingam, A., Hughes, C. O., Raine, R., Hughes, J., Sim, D. A., Egan, C., Tufail, A., Montgomery, H., Hassabis, D., Rees, G., Back, T., Khaw, P. T., Suleyman, M., Cornebise, J., Keane, P. A., & Ronneberger, O. (2018). Clinically applicable deep learning for diagnosis and referral in retinal disease. *Nature Medicine 24*, 1342–1350.
3. Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature 542*, 115–118.
4. McKinney, S. M., Sieniek, M., Godbole, V., Godwin, J., Antropova, N., Ashrafian, H., Back, T., Chesus, M., Corrado, G. S., Darzi, A., Etemadi, M., Garcia-Vicente, F., Gilbert, F. J., Halling-Brown, M., Hassabis, D., Jansen, S., Karthikesalingam, A., Kelly, C. J., King, D., Ledsam, J. R., Melnick, D., Mostofi, H., Peng, L., Reicher, J. J., Romera-Paredes, B., Sidebottom, R., Suleyman, M., Tse, D., Young, K. C., De Fauw, J., & Shetty, S. (2020). International evaluation of an AI system for breast cancer screening. *Nature 577*, 89–94.
5. Ciaparrone, G., Sánchez, F. L., Tabik, S., Troiano, L., Tagliaferri, R., & Herrera, F. (2020). Deep learning in video multi-object tracking: a survey. *Neurocomputing 381*, 61–88.
6. Sam, D. B., Peri, S. V., Sundararaman, M. N., Kamath, A., & Radhakrishnan, V. B. (2021). Locate, size and count: accurately resolving people in dense crowds via detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence 43*, 2739–2751.
7. Xiong, H., Lu, H., Liu, C., Liu, L., Cao, Z., & Shen, C. (2019). From open set to closed set: counting objects by spatial divide-and-conquer. In *Proceedings of the IEEE international conference on computer vision* (pp. 8361–8370). Los Alamitos: IEEE.
8. Hu, X., Xu, X., Xiao, Y., Chen, H., He, S., Qin, J., & Heng, P.-A. (2019). SINet: a scale-insensitive convolutional neural network for fast vehicle detection. *IEEE Transactions on Intelligent Transportation Systems 20*(3), 1010–1019.
9. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI Blog.
10. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. *30*, pp. 5998–6008). Red Hook: Curran Associates.
11. Baweja, C., Glocker, B., & Kamnitsas, K. (2018). Towards continual learning in medical imaging. arXiv preprint. arXiv:1811.02496.
12. Diethe, T., Borchert, T., Thereska, E., Balle, B., & Lawrence, N. (2019). Continual learning in practice. arXiv preprint. arXiv:1903.05202.
13. French, R. M. (1999). Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences 3*(4), 128–135.
14. Goodfellow, I. J., Mirza, M., Xiao, D., Courville, A., & Bengio, Y. (2013). An empirical investigation of catastrophic forgetting in gradient-based neural networks. arXiv preprint. arXiv:1312.6211.
15. Kemker, R., McClure, M., Abitino, A., Hayes, T. L., & Kanan, C. (2018). Measuring catastrophic forgetting in neural networks. In *Proceedings of the AAAI conference on artificial intelligence 2018* (pp. 3390–3398). Menlo Park: AAAI Press.
16. Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., & Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., & Hadsell, R. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences of the United States of America 114*(13), 3521–3526.
17. Yoon, J., Yang, E., Lee, J., & Hwang, S. J. (2018). Lifelong learning with dynamically expandable networks. [Paper presentation]. The 6th International Conference on Learning Representations (ICLR 2018), Vancouver, Canada. https://openreview.net/forum?id=Sk7KsfW0-
18. Li, Z., & Hoiem, D. (2018). Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence 40*(12), 2935–2947.
19. Rebuffi, S.-A., Kolesnikov, A., Sperl, G., & Lampert, C. H. (2017). iCaRL: incremental classifier and representation learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2001–2010). Los Alamitos: IEEE.
20. Shin, H., Kwon Lee, J., Kim, J., & Kim, J. (2017). Continual learning with deep generative replay. In I. Guyon, U. von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. *30*, pp. 2990–2999). Red Hook: Curran Associates.
21. Yang, Y., Cui, Z., Xu, J., Zhong, C., Wang, R., & Zheng, W.-S. (2021). Continual learning with Bayesian model based on a fixed pre-trained feature extractor. In *Medical image computing and computer assisted intervention* (pp. 397–406). Berlin: Springer.
22. Abati, D., Tomczak, J., Blankevoort, T., Calderara, S., Cucchiara, R., & Bejnordi, B. E. (2020). Conditional channel gated networks for task-aware continual learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3931–3940). Los Alamitos: IEEE.
23. Ahn, H., Cha, S., Lee, D., & Moon, T. (2019). Uncertainty-based continual learning with adaptive regularization. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. *32*, pp. 4394–4404). Red Hook: Curran Associates.
24. Fernando, C., Banarse, D., Blundell, C., Zwols, Y., Ha, D., Rusu, A. A., Pritzel, A., & PathNet, D. W. (2017). Evolution channels gradient descent in super neural networks. arXiv preprint. arXiv:1701.08734.
25. Jung, S., Ahn, H., Cha, S., & Moon, T. (2020). Continual learning with node-importance based adaptive group sparse regularization. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (Vol. *33*, pp. 3647–3658). Red Hook: Curran Associates.
26. Kim, H.-E., Kim, S., & Lee, J. (2018). Keep and learn: continual learning by constraining the latent space for knowledge preservation in neural networks. In *Medical image computing and computer assisted intervention* (pp. 520–528). Berlin: Springer.
27. Mallya, A., & Lazebnik, S. (2018). PackNet: adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7765–7773). Los Alamitos: IEEE.
28. Zenke, F., Poole, B., & Ganguli, S. (2017). Continual learning through synaptic intelligence. *Proceedings of Machine Learning Research 70*, 3987–3995.
29. Aljundi, R., Babiloni, F., Elhoseiny, M., Rohrbach, M., & Tuytelaars, T. (2018). Memory aware synapses: learning what (not) to forget. In *Proceedings of the European conference on computer vision* (pp. 139–154). Berlin: Springer.
30. Lopez-Paz, D., & Ranzato, M. (2017). Gradient episodic memory for continual learning. In I. Guyon, U. von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. *30*, pp. 6467–6476). Red Hook: Curran Associates.

31. Chaudhry, A., Ranzato, M., Rohrbach, M., & Elhoseiny, M. (2018). Efficient lifelong learning with a-gem. arXiv preprint. arXiv:1812.00420.
32. Tang, S., Chen, D., Zhu, J., Yu, S., & Ouyang, W. (2021). Layerwise optimization by gradient decomposition for continual learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 9634–9643). Los Alamitos: IEEE.
33. Wang, S., Li, X., Sun, J., & Xu, Z. (2021). Training networks in null space of feature covariance for continual learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 184–193). Los Alamitos: IEEE.
34. Liu, Y., Schiele, B., & Sun, Q. (2021). Adaptive aggregation networks for class-incremental learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2544–2553). Los Alamitos: IEEE.
35. Aljundi, R., Chakravarty, P., & Tuytelaars, T. (2017). Expert gate: lifelong learning with a network of experts. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3366–3375). Los Alamitos: IEEE.
36. Hung, C.-Y., Tu, C.-H., Wu, C.-E., Chen, C.-H., Chan, Y.-M., & Chen, C.-S. (2019). Compacting, picking and growing for unforgetting continual learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. *32*, pp. 13647–13657). Red Hook: Curran Associates.
37. Karani, N., Chaitanya, K., Baumgartner, C., & Konukoglu, E. (2018). A lifelong learning approach to brain MR segmentation across scanners and protocols. In *Medical image computing and computer assisted intervention* (pp. 476–484). Berlin: Springer.
38. Li, X., Zhou, Y., Wu, T., Socher, R., & Xiong, C. (2019). Learn to grow: a continual structure learning framework for overcoming catastrophic forgetting. In *International conference on machine learning* (pp. 3925–3934). PMLR.
39. Rajasegaran, J., Hayat, M., Khan, S. H., Shahbaz Khan, F., & Shao, L. (2019). Random path selection for continual learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. *32*, pp. 12648–12658). Red Hook: Curran Associates.
40. Yan, S., Xie, J., & He, X. (2021). DER: dynamically expandable representation for class incremental learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3014–3023). Los Alamitos: IEEE.
41. Li, Z., Zhong, C., Liu, S., Wang, R., & Zheng, W.-S. (2021). Preserving earlier knowledge in continual learning with the help of all previous feature extractors. arXiv preprint. arXiv:2104.13614.
42. Douillard, A., Ramé, A., Couairon, G., & Cord, C. (2022). DyTox: transformers for continual learning with dynamic token expansion. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 9285–9295). Los Alamitos: IEEE.
43. Cheraghian, A., Rahman, S., Ramasinghe, S., Fang, P., Simon, C., Petersson, L., & Harandi, M. (2021). Synthesized feature based few-shot class-incremental learning on a mixture of subspaces. In *Proceedings of the IEEE international conference on computer vision* (pp. 8661–8670). Los Alamitos: IEEE.
44. Ahn, H., Kwak, J., Lim, S., Bang, H., Kim, H., & Moon, T. (2021). Ss-il: separated softmax for incremental learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 844–853). Los Alamitos: IEEE.
45. Iscen, A., Zhang, J., Lazebnik, S., & Schmid, C. (2020). Memory-efficient incremental learning through feature adaptation. In *Proceedings of the European conference on computer vision* (pp. 699–715). Berlin: Springer.
46. Castro, F. M., Marín-Jiménez, M. J., Guil, N., Schmid, C., & Alahari, K. (2018). End-to-end incremental learning. In *Proceedings of the European conference on computer vision* (pp. 233–248). Berlin: Springer.
47. Hou, S., Pan, X., Loy, C. C., Wang, Z., & Lin, D. (2018). Lifelong learning via progressive distillation and retrospection. In *Proceedings of the European conference on computer vision* (pp. 437–452). Berlin: Springer.
48. Meng, Q., & Shin'ichi, S. (2020). ADINet: attribute driven incremental network for retinal image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4033–4042). Los Alamitos: IEEE.
49. Dhar, P., Singh, R. V., Peng, K.-C., Wu, Z., & Chellappa, R. (2019). Learning without memorizing. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4033–4042). Los Alamitos: IEEE.
50. Douillard, A., Cord, M., Ollion, C., Robert, T., & Valle, E. (2020). PODNet: pooled outputs distillation for small-tasks incremental learning. In *Proceedings of the European conference on computer vision* (pp. 86–102). Berlin: Springer.
51. Hou, S., Pan, X., Loy, C. C., Wang, Z., & Lin, D. (2019). Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 831–839). Los Alamitos: IEEE.
52. Hayes, T. L., Kafle, K., Shrestha, R., Acharya, M., & Kanan, C. (2020). REMIND your neural network to prevent catastrophic forgetting. In *Proceedings of the European conference on computer vision* (pp. 466–483). Berlin: Springer.
53. Rao, D., Visin, F., Rusu, A., Pascanu, R., Whye Teh, Y., & Hadsell, R. (2019). Continual unsupervised representation learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. *32*, pp. 7645–7655). Red Hook: Curran Associates.
54. Riemer, M., Klinger, T., Bouneffouf, D., & Franceschini, M. (2019). Scalable recollections for continual lifelong learning. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 1352–1359). Menlo Park: AAAI Press.
55. Ostapenko, O., Puscas, M., Klein, T., Jahnichen, P., & Nabi, M. (2019). Learning to remember: a synaptic plasticity driven framework for continual learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 11321–11329). Los Alamitos: IEEE.
56. Rios, A., & Itti, L. (2019). Closed-loop memory gan for continual learning. In *Proceedings of the international joint conference on artificial intelligence (IJCAI 2019)* (pp. 3332–3338). IJCAI.
57. Xiang, Y., Fu, Y., Ji, P., & Huang, H. (2019). Incremental learning using conditional adversarial networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 6619–6628). Los Alamitos: IEEE.
58. Bauer, P. J. (2015). A complementary processes account of the development of childhood amnesia and a personal past. *Psychological Review 122*(2), 204–231.
59. Ribordy, F., Jabès, A., Lavenex, P. B., & Lavenex, P. (2013). Development of allocentric spatial memory abilities in children from 18 months to 5 years of age. *Cognitive Psychology 66*(1), 1–29.
60. Scarf, D., Gross, J., Colombo, M., & Hayne, H. (2013). To have and to hold: episodic memory in 3-and 4-year-old children. *Developmental Psychobiology 55*(2), 125–132.
61. Nadel, L., Hupbach, A., Gomez, R., & Newman-Smith, K. (2012). Memory formation, consolidation and transformation. *Neuroscience and Biobehavioral Reviews 36*(7), 1640–1645.
62. Krizhevsky, A., & Hinton, G. (2009). *Learning multiple layers of features from tiny images*. Technical report, University of Toronto.
63. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., & Perona, P. (2010) *Caltech-UCSD Birds 200*. Technical report, California Institute of Technology.
64. Lin, T.-Y., RoyChowdhury, A., & Maji, S. (2015). Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE international conference on computer vision* (pp. 1449–1457). Los Alamitos: IEEE.
65. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint. arXiv:1409.1556.
66. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778). Los Alamitos: IEEE.
67. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: a large-scale hierarchical image database. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 248–255). Los Alamitos: IEEE.
68. Codella, N. C. F., Rotemberg, V., Tschandl, P., Celebi, M. E., Dusza, S. W., Gutman, D., Helba, B., Kalloo, A., Liopyris, K., Marchetti, M. A., Kittler, H., & Halpern, A. (2019). Skin lesion analysis toward melanoma detection 2018: a challenge hosted by the international skin imaging collaboration (ISIC). arXiv preprint. arXiv:1902.03368.
69. Sun, X., Yang, J., Sun, M., & Wang, K. (2016). A benchmark for automatic visual classification of clinical skin disease images. In *Proceedings of the European conference on computer vision* (pp. 206–222). Berlin: Springer.

Publisher's Note