Adaptive Robust Continual Learning based on Bayesian Uncertainty Propagation

Deepak Kandel
Chester F. Carlson Center For Imaging Science
Rochester Institute of Technology
dk6012@rit.edu

Dimah Dera
Chester F. Carlson Center For Imaging Science
Rochester Institute of Technology
dimah.dera@rit.edu

***Abstract*—Robust continual learning (CL) poses fundamental challenges and is essential for developing reliable and adaptable intelligent systems. Learning models must sustain a robust performance as they adapt to dynamically evolving environments through sequential learning, effectively overcoming the catastrophic forgetting problem. This paper proposes a novel, trustworthy CL framework based on the Bayesian variational uncertainty learned during training on each task. We integrate the Bayesian inference and propagate the first two moments of the variational posterior distribution over the probabilistic model's parameters. The variational moments (mean and covariance matrix) are learned simultaneously during training on each task and then used to estimate the predictive distribution. The covariance matrix of the variational posterior distribution captures the variational uncertainty in the learned parameters (particularly critical in the context of sequential learning within dynamic real-world settings). We develop an adaptive evidence lower bound (ELBO) loss function that supports managing the stability-elasticity dilemma. The variational continual optimization minimizes the expected log-likelihood of the data given the model's parameters and the Kullback–Leibler (KL) divergence between the variational distributions learned from the current and previous tasks weighted by an uncertainty-based metric. Moreover, we advance an architecture-based CL technique that masks important network parameters learned from each task based on their variational uncertainty. The proposed Bayesian regularization and architecture-based CL approaches prevent significant changes in the parameters of the learning models to preserve representations of previous tasks. The experiments on benchmark datasets demonstrate the robustness of the proposed framework when learning in continual scenarios compared to the stat-of-the-art CL homologs.**

***Index Terms*—Robust and trustworthy continual learning, variational uncertainty, and mean-covariance propagation.**

I. Introduction

Unlike humans, who possess a remarkable natural ability to continuously learn new skills and knowledge and develop the capacity for self-doubt, deep neural networks (DNNs) lack this ability when it comes to making decisions. Humans have a natural cognitive intuition for probabilities and can develop an inherent uncertainty surrounding their choices unless these decisions can be cross-verified against a known ground truth. In contrast, contemporary deep learning models are typically deterministic, yielding outputs that are singular point estimates [1]. The lack of inherent uncertainty in models' parameters and predictions presents a significant challenge when it comes to placing complete trust in DNNs' decisions. This is particularly important when learning from dynamically changing environments in domains such as medical diagnosis, autonomous driving, and national security [2]. DNNs should be designed to accumulate knowledge, adapt to changing data distributions, and incrementally acquire new skills without forgetting previously learned information. The concept of trustworthy continual learning (CL) revolves around building, maintaining, and ensuring confidence and reliability in the performance of DNNs when continuously learning in non-stationary environments. Thus, we can rely on decisions generated by these models. To ensure trustworthiness, one of the crucial considerations is the quantification of uncertainty associated with DNNs' parameters and predictions.

Recently, continual learning has been an active area of research among deep learning communities. Traditional learning methods have shown a drastic degradation of accuracy when a model learns a new task, which overwrites knowledge learned from previous tasks, also known as *Catastrophic Forgetting* [3]. For instance, self-driving cars face the formidable challenge of continuously acquiring knowledge about their dynamic surroundings as they navigate through various scenarios. It is computationally demanding, making it impractical to completely retrain the models every time a new object or situation is encountered. Autonomous systems must possess the ability to adapt and expand their understanding of the environment over time. The lack of adaptability can lead to serious accidents involving human lives, an undesirable outcome that must be prevented at all costs.

The Bayesian theory of learning provides a probabilistic principle for reasoning and making decisions, supplying an inherent ability to mitigate catastrophic forgetting [4]. The Bayesian inference facilitates incorporating beliefs and performing inference in the presence of uncertainty. The available data is employed to derive the posterior distribution over the parameters, which is used for determining the predictive distribution of new data points by marginalizing the model's parameters. The variance of the predictive distribution provides a quantitative measure of uncertainty in each prediction [5]. Due to the high dimensional parameter space of DNNs, the exact computation of the posterior distribution is infeasible mathematically [6]. Variational Inference approximates the posterior distribution efficiently, and it is compatible with the backpropagation optimization [6], [7].

In this paper, we propose novel Bayesian CL with Architecture Initiative and Regularization (CLAIR)-based approaches that quantify uncertainty and improve robustness when continuously learning new tasks without completely forgetting old tasks. The main contributions are summarized as follows:

- Develop an adaptive (AdaCLAIR) optimization by defining a new evidence lower bound (ELBO) loss function that minimizes the Kullback–Leibler (KL) divergence between the variational distributions over the parameters learned from the current and previous tasks. The proposed AdaCLAIR prevents considerable changes in the variational parameters to preserve knowledge learned from previous tasks and mitigate the problem of catastrophic forgetting.
- Preserve important parameters learned from previous tasks by weighing the KL divergence term in the AdaCLAIR ELBO loss by an uncertainty-defined metric based on the variational posterior variance learned from the previous task. This AdaCLAIR loss supports managing the stability-elasticity dilemma.
- Advance an uncertainty-based mask (MasCLAIR) to support efficient management of network resources and mitigate catastrophic forgetting. MasCLAIR freezes parameters with high confidence (low uncertainty) from previous tasks while updating parameters with low confidence for the upcoming tasks.
- Demonstrate superior and reliable performance (less forgetting) in CL scenarios compared to the state-of-the-art models using benchmark datasets.

The paper is organized as follows. Section II briefly recalls recent state-of-the-art CL methods. In Section III, we elaborately explain the proposed CLAIR frameworks. Section IV presents the experiments and evaluation results. We discuss and analyze the performance of the proposed methods in Section V. Section VI is the conclusion.

II. Related Works

The problem of catastrophic forgetting in the CL literature is addressed roughly by the three learning paradigms: dynamic network architecture, regularization-based methods, and dual memory system-based architectures, according to the comprehensive survey in [8].

The *dynamic architectural* strategies involve making structural modifications to the neural network architecture, which can include expanding, trimming, or locking specific components of the network to suit various tasks [9]–[14]. Applying these techniques helped mitigate forgetting to some extent; however, the computational complexity remained a bottleneck — especially in real-world scenarios where computational or memory constraints exist.

In *regularization-based* methods, the knowledge learned from previous tasks is restricted from being drastically updated during training on new tasks, which can be done by regularizing the objective function [15]–[20]. However, due to the extra loss term in the objective function, the model could suffer from extra computational overhead [21]. In addition, the regularization was applied to all parameters regardless of which parameters held important information from the previous tasks. In the *dual memory-based* approach, the representative samples of earlier tasks are partially stored to be used later during training for new tasks [22], [23]–[26].

The Bayesian framework of learning has been studied in the literature for a few decades, including Laplace approximation [27], Hamiltonian Monte Carlo [28], variational inference (VI) [29], and probabilistic backpropagation [30]. Nguyen *et al.* introduced Variational Continual Learning (VCL) [4] where new posterior distribution was obtained by multiplying the previous posterior with the likelihood of the data in the new task. They also posited that by introducing a small core set from the previous task, the model experiences less forgetting. Ebrahimi *et al.* used the uncertainty associated with the weights to update the learning rate during training. [31].

Although a few methods in the literature considered the Bayesian inference to address continual learning challenges, these state-of-the-art techniques relied on frequentist probability or sampling approaches to quantify uncertainty [31]–[33]. They follow Monte Carlo (MC) sampling by drawing one random sample from the variational distribution and passing it forward through the network layers. One random sample is not a sufficient representative of the variational distribution. The moments of the distribution are generally *not propagated* through the networks' architecture and various operations within the Bayesian CL framework. Estimating the uncertainty in the models' predictions required performing multiple passes (MC samples) through the model layers and computing the sample variance of different predictions. Moreover, such measures of uncertainty in CL models have not been analyzed under CL scenarios to show how uncertainty changes when learning a sequence of tasks.

In contrast, we propose the CLAIR framework that propagates the mean and covariance of the variational posterior distribution to simultaneously learn predictions along with the predictive uncertainty. In the proposed framework, we develop a new adaptive ELBO loss function and design a masking technique that controls how parameters are updated for the new tasks based on the variational parameters' uncertainty. The parameters' uncertainty learned from the previous tasks is used to filter out the less important parameters during the training on the current task to balance the *stability-plasticity* dilemma—an act of learning to adapt to new tasks while preserving the trusted knowledge from previous tasks.

III. Adaptive Continual Learning based on Variational Inference (AdaCLAIR) Framework

In this section, we formalize the mathematical notations for the AdaCLAIR framework considering a convolutional neural network (CNN) with a total of C convolutional layers and L fully connected layers. A non-linear activation function is introduced after every convolutional and fully connected layer. There is a max-pooling layer after every convolution operation. The network's learnable parameters—weights with biases augmented in the weight matrices—are represented by $\mathcal{W} = \left\{\{\{\mathbf{W}^{(k)}\}_{k=1}^{K_c}\}_{c=1}^{C}, \{\mathbf{U}^{(l)}\}_{l=1}^{L}\right\}$, where $\{\{\mathbf{W}^{(k)}\}_{k=1}^{K_c}\}_{c=1}^{C}$ is

the set of K_c kernels in the c^{th} convolutional layer, and $\{\mathbf{U}^{(l)}\}_{l=1}^{L}$ is the set of weight matrices in L fully-connected layers. We consider an input image $\mathbf{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, where I_1, I_2, and I_3 represent the image height, width, and number of channels respectively.

A. Notations and Review of Variational Inference

We define a prior probability distribution over the model parameters, $\mathcal{W} \sim p(\mathcal{W})$. We impose the independence assumption between the probabilistic convolutional kernels within and across layers. This independence assumption makes sense and can be beneficial, as it (1) extracts uncorrelated features within and across layers and (2) develops a feasible optimization problem, as estimating the joint distribution of all kernels and all layers is mathematically intractable in large DNN models [5]. After observing training samples $\mathcal{D} = \{\mathbf{X}_i, \mathbf{y}_i\}_{i=1}^{N}$ and using the prior distribution $p(\mathcal{W})$, we approximate the true unknown posterior distribution, $p(\mathcal{W}|\mathcal{D})$, with a simpler parametric variational distribution $q_\phi(\mathcal{W})$. The optimal parameters ϕ^* of this variational approximation are estimated by minimizing the KL divergence between the variational and the true posterior distributions, $\mathrm{KL}\left[q_\phi(\mathcal{W}) \| p(\mathcal{W}|\mathcal{D})\right]$. The loss is known as the evidence lower bound (ELBO), $\mathcal{L}(\boldsymbol{\phi}; \mathcal{D})$ [7].

$$\mathcal{L}(\boldsymbol{\phi}; \mathcal{D}) = -\mathbb{E}_{q_\phi(\mathcal{W})}\left\{\log p(\mathcal{D}|\mathcal{W})\right\} + \mathrm{KL}\left[q_\phi(\mathcal{W}) \| p(\mathcal{W})\right]. \tag{1}$$

The first term in the ELBO loss is the log-likelihood expectation of the training data given the weights. The second term serves as a regularizer on the variational parameters of every convolution and fully connected layer and is simplified as,

$$\begin{aligned}\mathrm{KL}\left[q_\phi(\mathcal{W}) \| p(\mathcal{W})\right] = &\sum_{c=1}^{C}\sum_{k=1}^{K_c} \mathrm{KL}[q_\phi(\mathbf{W}^{(k)}) \| p(\mathbf{W}^{(k)})] \\ &+ \sum_{l=1}^{L} \mathrm{KL}[q_\phi(\mathbf{U}^{(l)}) \| p(\mathbf{U}^{(l)})].\end{aligned} \tag{2}$$

We assume the initial variational distribution has a diagonal covariance matrix and use the first-order Taylor series to approximate the expected log-likelihood in (1) as follows,

$$E_{q_\phi}\{\log p(\mathcal{D}|\mathcal{W})\} \approx \frac{-1}{2N}\sum_{i=1}^{N} \log(\prod \boldsymbol{\sigma}^2_{\hat{\mathbf{y}}_i}) + \|\mathbf{y}_i - \boldsymbol{\mu}_{\hat{\mathbf{y}}_i}\|_2^2 \odot \frac{1}{\boldsymbol{\sigma}^2_{\hat{\mathbf{y}}_i}} \tag{3}$$

where N refers to independently and identically distributed (iid) data points for a specific task, $\mathbf{y}_i$ is the ground truth output of the i^{th} data sample, $\boldsymbol{\mu}_{\hat{\mathbf{y}}_i}$ and $\boldsymbol{\sigma}^2_{\hat{\mathbf{y}}_i}$ are the mean and variance of the predicted output $\hat{\mathbf{y}}_i$ and $\prod$ is a product for $\boldsymbol{\sigma}^2_{\hat{\mathbf{y}}_i}$ entries. The regularization term in (2) is the KL-divergence between two multivariate Gaussian distributions, i.e., the variational posterior distribution and the prior distribution, defined over the network parameters for all convolution and fully connected layers [34].

B. Uncertainty Propagation

We propagate the moments of the variational distributions, $q_\phi(\mathcal{W})$, i.e., the mean and covariance matrix, through all layers of the proposed CLAIR models considering the original ELBO loss in Equation 1, named PlaCLAIR, and the proposed AdaCLAIR and MasCLAIR following [5]. All the learnable parameters, including the convolutional filters, units in fully connected layers, and activation maps, are probabilistic. The propagation through linear operations, such as convolution, matrix multiplication, and inner product, is done using closed-form mathematical relations. We adopt the first-order Taylor approximation to propagate the mean and covariance matrix of the variational distributions through non-linear activation functions in the Bayesian CLAIR models. Let $\mathbf{z} = f(\mathbf{x})$, where $\mathbf{z}$ and $\mathbf{x}$ are vectorized feature maps before and after the non-linear activation function, f, for any layer in the CLAIR models. The mean and covariance matrix of $\mathbf{x}$ propagate through f to $\mathbf{z}$ using the first-order Taylor approximation. Thus, $\boldsymbol{\mu}_\mathbf{z}$ and $\boldsymbol{\Sigma}_\mathbf{z}$ are derived as follows,

$$\boldsymbol{\mu}_\mathbf{z} \approx f(\boldsymbol{\mu}_\mathbf{x}), \qquad \boldsymbol{\Sigma}_\mathbf{z} \approx \boldsymbol{\Sigma}_\mathbf{x} \odot (\nabla f(\boldsymbol{\mu}_\mathbf{x})\ \nabla f(\boldsymbol{\mu}_\mathbf{x})^T), \tag{4}$$

where ∇ represents the gradient of the function f with respect to $\mathbf{x}$ evaluated at $\boldsymbol{\mu}_\mathbf{x}$ and $\odot$ represents the Hadamard product. The results presented in Equation 4 hold true for any non-linear activation function, including hyperbolic tangent (Tanh), sigmoid, or rectified linear unit (ReLU). The variational uncertainty is measured by the variance of the variational distribution over the probabilistic parameters. By propagating the variational moments through all layers, we obtain the moments of the predictive distribution, $p(\hat{\mathbf{y}}|\hat{\mathbf{X}}, \mathcal{D})$. The mean of $p(\hat{\mathbf{y}}|\hat{\mathbf{X}}, \mathcal{D})$, i.e., $\boldsymbol{\mu}_{\hat{\mathbf{y}}}$, represents the prediction, while the variance, $\boldsymbol{\sigma}^2_{\hat{\mathbf{y}}}$, reflects the prediction uncertainty.

C. Adaptive Optimization with Bayesian Continual Learning

Suppose there are $\{1, \dots T\}$ CL tasks. Each task t consists of N_t data samples, $\mathcal{D}_t = \{(\mathbf{X}_i, \mathbf{y}_i)\}_{i=1}^{N_t}$. The datasets of each task are assumed to be iid from their corresponding distribution. In the Bayesian setting, the variational distribution is $q_{\phi_t}(\mathcal{W}_t)$, where ϕ_t represents the variational parameters for the t^{th} task. We develop a new adaptive evidence lower bound (ELBO) loss function, where the second term computes the KL divergence between the variational distributions over the parameters learned from the current and previous tasks, i.e., $\mathrm{KL}[q_{\phi_t}(\mathcal{W}_t)||q_{\phi_{t-1}}(\mathcal{W}_{t-1})]$. In the case of the first task, the prior distribution is set to a zero-centered Gaussian distribution. In addition, we multiply the updated KL term with an uncertainty-defined metric (referred to as κ_{t-1}) based on the variational variance learned from the previous task. The adaptive ELBO loss is then derived as the following.

$$\mathcal{L}(\boldsymbol{\phi}_t, \mathcal{D}_t) = -\mathbb{E}_{q_{\phi_t}}[\log p(\mathcal{D}_t|\mathcal{W}_t)] + \kappa_{t-1}\mathrm{KL}[q_{\phi_t}||q_{\phi_{t-1}}]. \tag{5}$$

The variational parameters at task t are given as $\boldsymbol{\phi}_t = \left\{\{\{\boldsymbol{\mu}_{k_t}, \boldsymbol{\Sigma}_{k_t}\}_{k=1}^{K_c}\}_{c=1}^{C}, \{\{\boldsymbol{\mu}_{h_t}, \boldsymbol{\Sigma}_{h_t}\}_{h=1}^{H_l}\}_{l=1}^{L}\right\}$, where $\boldsymbol{\mu}_{k_t}$ and $\boldsymbol{\Sigma}_{k_t}$ are the mean and covariance matrix of the k^{th} kernel in the c^{th} convolution layer, and $\boldsymbol{\mu}_{h_t}$ and $\boldsymbol{\Sigma}_{h_t}$ are the mean and covariance matrix of the h^{th} weight vector in the l^{th} fully connected layer. The covariance matrices are assumed $\boldsymbol{\Sigma}_{k_t} = \sigma^2_{k_t}\mathbf{I}$, and $\boldsymbol{\Sigma}_{l_t} = \sigma^2_{l_t}\mathbf{I}$, where $\mathbf{I}$ is an identity matrix.

$$\mathrm{KL}[q_{\phi_t}\|q_{\phi_{t-1}}] = \kappa_{k_{t-1}} \sum_{c=1}^{C} \sum_{k=1}^{K_c} \|\boldsymbol{\mu}_{k_{t-1}} - \boldsymbol{\mu}_{k_t}\|_2^2 + d_k\left(\frac{\sigma_{k_t}^2}{\sigma_{k_{t-1}}^2} + \log\left(\frac{\sigma_{k_t}^2}{\sigma_{k_{t-1}}^2}\right)\right) + \kappa_{h_{t-1}} \sum_{l=1}^{L} \sum_{h=1}^{H_l} \|\boldsymbol{\mu}_{h_{t-1}} - \boldsymbol{\mu}_{h_t}\|_2^2 + d_h\left(\frac{\sigma_{h_t}^2}{\sigma_{h_{t-1}}^2} + \log\left(\frac{\sigma_{h_t}^2}{\sigma_{h_{t-1}}^2}\right)\right). \tag{6}$$

Thus, the adaptive weighted KL regularization term is updated according to Equation 6, where d_k is the dimension of the k^{th} kernel in the c^{th} convolution layer and d_h is the dimension of the h^{th} weight vector in the l^{th} fully connected layer. The uncertainty-defined metrics for the convolution and fully connected layers are $\kappa_{k_{t-1}} = \frac{1}{\sigma_{k_{t-1}}^2}$ and $\kappa_{h_{t-1}} = \frac{1}{\sigma_{h_{t-1}}^2}$, respectively.

In the proposed adaptive ELBO loss (Equations 5 and 6), we penalize drastic changes in the variational parameters when training on a new task based on the uncertainty learned from the previous task. If the uncertainty learned from the previous task is high, $\sigma_{k_{t-1}}^2 \gg 0$ or $\sigma_{h_{t-1}}^2 \gg 0$, then $0 < \kappa_{k_{t-1}} \ll 1$ or $0 < \kappa_{h_{t-1}} \ll 1$, respectively. In this case, the adaptive KL regularization will play less of a role in the optimization, allowing the parameters to learn from the new task. Conversely, suppose the uncertainty learned from the previous task is low. In that case, then $\kappa_{k_{t-1}} \gg 1$ or $\kappa_{h_{t-1}} \gg 1$, and the adaptive KL regularization will preserve those parameters with low uncertainty by penalizing significant changes in their values. Thus, the uncertainty in the parameters, learned from the previous task and measured by the variational variance, decides the important parameters to preserve and irrelevant parameters to update when learning a new task. The proposed adaptive ELBO loss supports managing the stability-elasticity dilemma.

D. Static Uncertainty-based Mask—MasCLAIR

Learnable parameters (weights and biases) in DNNs are not all at the same level of importance—some parameters are more important for inference than others [35]. Especially in the CL scenario, it is very crucial to track the parameters' level of importance while the model is trained on sequential tasks, which can cause interference during inference time. Identifying the level of importance of the network parameters enables managing network resources while continuously learning new tasks. Thus, we can preserve the previously learned information and make use of less important parameters for learning new tasks.

We design an uncertainty-based mask ($\mathcal{M}$) that depends on the variational variance of the learned parameters. Parameters with lower variance are more important and need to be prevented from being updated during training for the new tasks. The uncertainty mask is a binary matrix, where the zeros correspond to the parameters with low variance (highly important parameters) and the ones correspond to the parameters with high variance (irrelevant parameters). The degree of importance of the DNN's parameters is defined based on a threshold, i.e., the 90^{th} percentile of the variational variance histogram per layer. The parameters with the variance values less than 90^{th} percentile are considered highly important parameters. The mask then multiplies the gradient during gradient descent optimization for each task, and the zeros in the mask preserve important parameters. In contrast, the ones in the mask release irrelevant parameters to efficiently learn from new tasks.

$$\boldsymbol{\mu}_{k,t}^{(i)} = \boldsymbol{\mu}_{k,t}^{(i-1)} - \mathcal{M}_{t-1} \odot \gamma \nabla \mathcal{L}(\boldsymbol{\phi}_t, \mathcal{D}_t), \tag{7}$$

where $\boldsymbol{\mu}_{k,t}^{(i)}$ is the variational mean for the k^{th} kernel at task t and i^{th} iteration of gradient descent and γ is the learning rate.

E. Multi Head vs Single Head Setting

In CL literature, there are two main strategies for performing inference for each task: the single-head and the multi-head settings. In the multi-head setting, a separate, fully connected layer or "head" is used for each task. Each head is responsible for learning and retaining knowledge related to its specific task [21]. The models with the multi-head setting need to know the task ID for each input sample during both training and inference. Thus, during inference time, the output layer corresponding to a specific task is used to make predictions about input samples from that task. On the other hand, the single-head CL models have a single output layer that is shared across all tasks [21]. The models in the single-head setting do not need to know which task the input sample belongs to make a prediction. The models can learn to make predictions for all tasks without a task ID. The single-head CL models are simple and efficient but more prone to catastrophic forgetting [4]. The multi-head models are less prone to forgetting but more complex and computationally expensive [21]. In our proposed CLAIR models, we adopt the single-head setting to train the models on each task.

F. Computational Complexity

The trade-off between accuracy and robustness versus computational complexity is well-known in Bayesian models. However, in the proposed CLAIR models, the number of learnable parameters is comparable to its deterministic variants. The diagonal covariance assumption for the prior distribution allows for adding a single parameter (the variance) for every kernel in the convolution layers and every weight vector in the fully connected layers. This assumption helps moderate the computational complexity of our proposed model. For example, suppose a CNN with $C = 10$ convolutional layers where each convolution layer has ($K_c = 32$) kernels of size 5×5. The network also has one fully connected layer ($L = 1$) with ($H = 10$ the size of the output vector). The number of additional parameters in this case is equal to $C * K_c + H = 330$. The total number of parameters for the deterministic model is $(10 * 5 * 5 * 32 + 4,608 * 10 = 54,080)$, assuming the size of the feature map of the last convolution layer is $12 * 12 * 32$. On the other hand, the total number of parameters for the corresponding Bayesian model is $(10 * 5 * 5 * 32 + 10 * 32 + 4,608 * 10 + 10 = 54,410)$. Therefore, the increase in

the number of parameters for the CLAIR models is $\approx 0.6\%$ compared to the deterministic counterpart.

IV. Experiments and Results

A. Experimental Setting

In this section, we evaluate the performance of the proposed CLAIR frameworks compared to the Elastic Weight Consolidation (EWC) [15], Bayes-By-Backprop (BBB) [6] and Variational Continual Learning (VCL) [4]. We adopt the adaptive momentum (ADAM) optimization algorithm with the polynomial decay as a learning rate scheduler for training the models on each task [36]. We evaluate the models' performance on each task using the task-wise average classification accuracy. The output predictive variance (i.e., the diagonal element of the predictive covariance matrix that corresponds to the predicted class) is also reported to show the model uncertainty when learning new tasks sequentially. Moreover, to assess the learning stability, we evaluate the forgetting measure (FM) and backward transfer (BWT) defined in Section IV-B.

In our experiments, we consider the proposed CLAIR model with the original ELBO loss in Equation 1, named PlaCLAIR, and compare it to BBB. We also compare the proposed AdaCLAIR and MasCLAIR with the state-of-the-art Bayesian models, including EWC and VCL. The network architecture for the models contains one convolutional layer (32 kernels with size 5×5), a rectified linear unit (ReLU) activation function, a max-pooling layer, and a fully connected layer followed by a softmax layer. We adopt two CL scenarios using the benchmark MNIST dataset, i.e., split MNIST and permuted MNIST [37]. For the split MNIST experiment, we train the models for 10 epochs with an initial learning rate of 0.001. For the permuted MNIST experiment, we train the models for 5 epochs with an initial learning rate of 0.001. The difference in these sets of hyperparameters is due to the nature of the classification task and varying data distribution in each experiment. The training and validation of the proposed CLAIR models and the comparison against other state-of-the-art models are performed using the A100 GPU computing cluster provided by RIT Research Computing [38] and RTX A6000 Ada Generation workstation.

B. Evaluation Metrics

1) Average Accuracy (A_t): To balance the stability and plasticity of the proposed models compared to state-of-the-art models and make a fair comparison, we average the test accuracy of each model on the current and previous tasks. Let $a_{t,j} \in [0, 1]$ denote the classification accuracy evaluated on the test set of the j^{th} task after sequential learning of the t^{th} task ($j \leq t$). Then, the average test accuracy of the model trained on task t and tested on task t and $t-1$ is computed as follows:

$$A_t = \frac{1}{2}(a_{t,t} + a_{t,t-1}). \tag{8}$$

2) Forgetting Measure (FM): The forgetting measure (FM) denotes how much of learned knowledge is lost while learning new tasks. The forgetting rate for task j, when learning task t, i.e., $f_{j,t}$, given that $j < t$, is calculated by the difference between the model's maximum performance obtained in the past and its current performance [8].

$$f_{j,t} = \max_{i \in \{1,\ldots,t-1\}} (a_{i,j} - a_{t,j}), \quad \forall j < t. \tag{9}$$

FM for the t^{th} task is the average forgetting of all old tasks.

$$FM_t = \frac{1}{t-1} \sum_{j=1}^{t-1} f_{j,t}. \tag{10}$$

3) Backward Transfer (BWT): The backward transfer (BWT) refers to the average influence of learning a new task (t) on previously learned tasks ($1, \cdots, t-1$), where $j < t$ [8].

$$BWT_t = \frac{1}{t-1} \sum_{j=1}^{t-1} |a_{t,j} - a_{j,j}|. \tag{11}$$

4) Total Average Accuracy: Assume all test data of all T tasks are available for evaluation, and the average test accuracy of task t after training on $t-1$ sequential tasks is A_t. The average task accuracy is then given as,

$$A_{avg} = \frac{\sum_{t=1}^{T} A_t}{T}. \tag{12}$$

C. Image Classification: Split MNIST

In this experiment, we divide MNIST dataset samples into 5 non-overlapping sequential tasks corresponding to binary image classification problems, i.e., digits (0 - 1), (2 - 3), (4 - 5), (6 - 7), and (8 - 9). Table I presents the average test accuracy per task, A_t, the forgetting measure (FM), and the backward transfer (BWT) of the split MNIST experiment for the proposed CLAIR models compared to the state-of-the-art EWC, BBB and VCL models. We show that the learned knowledge is significantly retained when using the adaptive ELBO loss in the AdaCLAIR model and the uncertainty mask in the MasCLAIR model. The proposed AdaCLAIR and MasCLAIR models demonstrate higher accuracy and less catastrophic forgetting compared to other probabilistic homologs. Figure 1 shows the average variance (uncertainty) of the proposed CLAIR models compared to the BBB and VCL models' MC sample variance. The average variance is collected for every task in the split MNIST experiment.

D. Image Classification: Permuted MNIST

In the permuted MNIST experiment, the original MNIST dataset is modified by randomly rearranging pixels within each image. Thus, the classification tasks of MNIST images become more challenging for machine learning models [39]. All models are trained and validated for five different permutations with random seeds of 1, 2, 3, 4, and 5 as sequential CL tasks. Table II shows the average test accuracy per task, A_t, the forgetting measure (FM), and the backward transfer (BWT) for the proposed CLAIR models compared to the state-of-the-art EWC and VCL models. Figure 2 shows the average uncertainty measured by the predictive variance of the proposed CLAIR models compared to the MC sample variance of the BBB and VCL models. The average variance is collected for every task in the permuted MNIST experiment.

TABLE I
RESULTS OF THE SPLIT MNIST EXPERIMENT. THE ACCURACY, THE FORGETTING MEASURE (FM), AND THE BACKWARD TRANSFER (BWT) ARE COMPUTED USING EQUATIONS 8, 10 AND 11, RESPECTIVELY.

	Evaluation Metrics	Task 1 0-1	Task 2 2-3	Task 3 4-5	Task 4 6-7	Task 5 8-9
EWC	Accuracy	99%	89.5%	90.5%	92%	90.5%
	FM	-	9.5	8.5	7	8.5
	BWT	-	9.5	4.5	3.33	2.25
BBB	Accuracy	99%	86.5%	81%	83.5%	93%
	FM	-	12.5	12	7.67	9.8
	BWT	-	12.5	6.75	8	1.88
VCL	Accuracy	99%	89%	81.5%	88.5%	93%
	FM	-	10	17.5	10.5	4
	BWT	-	10	8.75	5.5	3.5
Proposed PlaCLAIR	Accuracy	96%	90.5%	93.8%	90.5%	88%
	FM	-	5.5	3.8	6.3	5.8
	BWT	-	**5.5**	1.13	2.92	**1.88**
Proposed AdaCLAIR	Accuracy	99%	91%	86.5%	94%	93%
	FM	-	8	12.5	5	6
	BWT	-	8	6.25	3.68	3.75
Proposed MasCLAIR	Accuracy	**99%**	**91.8%**	**95.8%**	**97.4%**	**96.2%**
	FM	-	7.2	**2.4**	**1.27**	**1.25**
	BWT	-	7.2	**0.1**	**0.8**	2.1

TABLE II
RESULTS OF THE PERMUTED MNIST EXPERIMENT WITH FIVE DIFFERENT PERMUTATIONS. THE ACCURACY, THE FORGETTING MEASURE (FM), AND THE BACKWARD TRANSFER (BWT) ARE COMPUTED USING EQUATIONS 8, 10, AND 11, RESPECTIVELY.

	Evaluation Metrics	Task 1 Perm 1	Task 2 Perm 2	Task 3 Perm 3	Task 4 Perm 4	Task 5 Perm 5
EWC	Accuracy	95.4%	86.2%	78.5%	84%	87%
	FM	-	9.18	18.86	11.36	8.36
	BWT	-	9.18	8.93	6.12	5.34
VCL	Accuracy	**98%**	91.5%	80.5%	74%	80%
	FM	-	6.5	17.5	24	18
	BWT	-	6.5	8.75	9.6	9.5
Proposed PlaCLAIR	Accuracy	97%	86.5%	89.3%	90.3%	89.9%
	FM	-	10.5	5.25	3.82	2.86
	BWT	-	10.5	2.45	1.65	1.48
Proposed AdaCLAIR	Accuracy	97.1%	92.5%	92.8%	92.3%	93.2%
	FM	-	4.6	4.35	4.85	3.95
	BWT	-	4.5	2.18	1.96	1.73
Proposed MasCLAIR	Accuracy	96%	**93%**	**93.5%**	**93.8%**	**93.8%**
	FM	-	**3**	**1.5**	**0.9**	**0.6**
	BWT	-	**3**	**1.5**	**1.07**	**0.7**

V. DISCUSSIONS AND ANALYSIS

We observe from Table I that the proposed CLAIR models, i.e., PlaCLAIR, AdaCLAIR, and MasCLAIR, maintain higher accuracy on the sequence of tasks. We highlight the highest accuracy across all models for each task. For example, the accuracy values of the MasCLAIR on tasks 1 to 5, respectively, are 99%, 91.8%, 95.8%, 97.4%, and 96.2%. In comparison, the accuracy values of the VCL model on tasks 1 to 5 are 99%, 89%, 81.5%, 88.5%, and 93%, respectively. The proposed MasCLAIR model provides accuracy values that are at least 2.8% higher as compared to other models. Similarly, we observe higher accuracy of the PlaCLAIR and AdaCLAIR models compared to the accuracy of the EWC, BBB, and VCL models. We also notice that the FM and BWT values of the proposed models are smaller than those of the EWC, BBB, and VCL models. For example, the BWT values of the PlaCLAIR for tasks 2 to 5, respectively, are 5.5%, 1.13%, 2.92%, and 1.88%. In comparison, the BWT values of the BBB model on tasks 2 to 5, respectively, are 12.5%, 6.75%, 8%, and 1.88%.

By observing Table II using the permuted MNIST experiment, we notice that the proposed AdaCLAIR and MasCLAIR models outperform all other models on the sequential CL tasks. The accuracy values of the MasCLAIR on tasks 1 to 5 are 96%, 93%, 93.5%, 93.8%, and 93.8%, respectively. On the other hand, the VCL accuracy values on tasks 1 to 5 are 98%, 91.5%, 80.5%, 74%, and 80%, respectively. The VCL model's accuracy degrades $\approx 24\%$ compared to $\approx 3\%$ for the MasCLAIR model and $\approx 4.8\%$ for the AdaCLAIR model. Similarly, the FM and BWT values are much smaller for the proposed MasCLAIR and AdaCLAIR than for all other models. The highest accuracy values and the lowest FM and BWT values are highlighted in the table for each task. It is noteworthy that, initially, all the models (in the split MNIST and permuted MNIST experiments) exhibited good performance on the first task. However, once the models are trained on new tasks, the proposed CLAIR models maintain robust behavior and mitigate the catastrophic forgetting phenomenon as compared to other state-of-the-art models.

Table III shows a comparison between the proposed CLAIR models and the current state-of-the-art models, including Incremental Classifier and Representation Learning (iCaRL) [40], Learning to Prompt (LP) [41], Gradient Episodic Memory (GEM) [42], Riemannian Walk (RWalk) [18], Robust Continual Learning through a Comprehensively Progressive Bayesian Neural Network (RCL-CPB) [21]), EWC and VCL using the total average accuracy (Equation 12) for both split and permuted MNIST. The proposed CLAIR models (and particularly MasCLAIR) demonstrate dominant accuracy over the state-of-the-art CL models. In the split MNIST experiment, MasCLAIR achieves 96.1%, and in the case of permuted MNIST, it achieves 94%. In addition to this superior accuracy, the proposed models capture the uncertainty associated with the predictions, measured by the predictive variance, bolstering the trustworthiness of the proposed models.

A. Trustworthiness of the CLAIR Models

By conducting exhaustive experiments, we demonstrate the ability of the proposed Bayesian CL with Architecture Initiative and Regularization (CLAIR)-based models to sequentially acquire and preserve knowledge from a sequence of tasks. The CLAIR models effectively mitigate the adverse impacts of forgetting, as compared to the state-of-the-art model in the literature. The predictive variance associated with each model prediction quantifies the inherent uncertainty, thus cautioning

TABLE III
THE TOTAL AVERAGE ACCURACY OF THE CLAIR MODELS COMPARED TO THE STATE-OF-THE-ART MODELS ON THE SPLIT AND PERMUTED MNIST.

Models	*Total Average Accuracy (A¯avg)*	
	Split MNIST	Permuted MNIST
iCarL	55.8% [21]	–
LP	61.2% [21]	82.0% [21]
GEM	94.3% [21]	93.1% [21]
RWalk	82.5% [21]	91.7% [21]
RCL-CPB	83.8% [21]	92.7% [21]
EWC	86.2%	92.3%
VCL	90.2%	84.8%
PlaCLAIR (ours)	91.8%	90.6%
AdaCLAIR (ours)	92.7%	93.6%
MasCLAIR (ours)	**96.1**%	**94**%

against unwarranted trust in the model's output, which is particularly important in mission-critical scenarios.

We notice from Figures 1 and 2 that the prediction variance (uncertainty) of the proposed models is higher than the MC variance of the BBB and VCL models. In the split MNIST experiment, the average predictive variance of PlaCLAIR, AdaCLAIR, and MasCLAIR starts with low values as we train and validate the models on the same task. Later, when we introduce new tasks, the average variance values increase gradually, highlighting the fact that the model establishes some uncertainty with the previous tasks after learning new tasks. Even though the proposed CLAIR models adapt well to new tasks and provide high accuracy, the predictive variance increases when introducing new tasks. We interpret the variance behavior as the model builds higher confidence for the current tasks (low uncertainty) and lower confidence for the previous tasks (higher uncertainty) due to the change of the data distribution of different digits in the split MNIST experiment. We expect the variance to decrease if we introduce samples from previous tasks while learning new tasks.

In the permuted MNIST experiment, the average variance starts at high levels when training on the first permutation of MNIST as the first task. Then, the variance values decrease with incoming tasks as the models adapt to the data distribution resulting from different permutations in the new tasks. We will explore the variance (uncertainty) behavior of the proposed CLAIR models with more continual learning experiments in the future to understand the robustness and adaptability to sequentially streaming data.

VI. CONCLUSIONS

In this paper, we introduce an innovative Bayesian CL with Architecture Initiative and Regularization (CLAIR)-based models, i.e., AdaCLAIR and MasCLAIR, for facilitating continual learning by leveraging the principles of the variational uncertainty propagation. We adopt the Bayesian theory of learning and propagate the moments of the variational posterior distribution through the layers of the convolutional neural network (CNN) to learn uncertainty over the parameters measured by the variance of the variational distribution. We

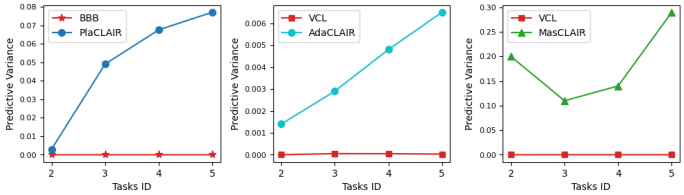

Fig. 1. Average predictive variance (uncertainty) of the proposed PlaCLAIR, AdaCLAIR, and MasCLAIR models compared to the Monte Carlo sampling variance of the BBB and VCL models for the split MNIST experiment.

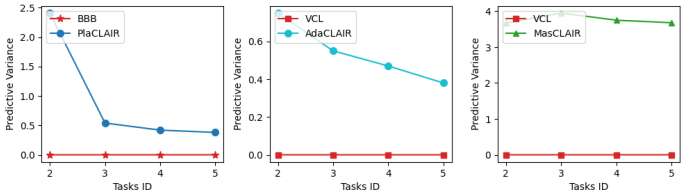

Fig. 2. Average predictive variance (uncertainty) of the proposed PlaCLAIR, AdaCLAIR, and MasCLAIR models compared to the Monte Carlo sampling variance of the BBB and VCL models for the permuted MNIST experiment.

develop a new adaptive evidence lower bound (ELBO) loss function that penalizes considerable changes in the variational parameters by minimizing the KL divergence between the variational posterior distributions learned from the current and previous tasks weighted by an uncertainty-defined metric. The newly formulated ELBO objective function effectively serves as a mechanism to impede drastic changes in important parameters and preserve knowledge learned from previous tasks based on the uncertainty associated with those parameters. We also advance an uncertainty-based mark to freeze parameters with high confidence (or low uncertainty) from previous tasks while learning new tasks. The proposed CLAIR models inherently addressed the common challenge of catastrophic forgetting in sequential learning scenarios. The experimental results, conducted on both split MNIST and permuted MNIST datasets, demonstrate a notable improvement in the trustworthiness and reliability of the proposed models compared to state-of-the-art models.

ACKNOWLEDGMENT

This work was supported by the National Science Foundation Award CRII # 2401828. We would also like to thank the Rochester Institute of Technology Artificial Intelligence Seed Funding Award.

REFERENCES

[1] J. Wu, "Introduction to convolutional neural networks," *National Key Lab for Novel Software Technology. Nanjing University. China*, vol. 5, no. 23, p. 495, 2017.

[2] P. P. Shinde and S. Shah, "A review of machine learning and deep learning applications," in *IEEE Fourth International Conference on Computing Communication Control and Automation*, 2018, pp. 1–6.

[3] I. J. Goodfellow, M. Mirza, D. Xiao, A. Courville, and Y. Bengio, "An empirical investigation of catastrophic forgetting in gradient-based neural networks," *arXiv preprint arXiv:1312.6211*, 2013.

[4] C. V. Nguyen, Y. Li, T. D. Bui, and R. E. Turner, "Variational continual learning," in *International Conference on Learning Representations*, 2018.

[5] D. Dera, N. C. Bouaynaya, G. Rasool, R. Shterenberg, and H. M. Fathallah-Shaykh, "PremiUm-CNN: Propagating uncertainty towards robust convolutional neural networks," *IEEE Transactions on Signal Processing*, vol. 69, pp. 4669–4684, 2021.

[6] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight uncertainty in neural network," in *International conference on machine learning*. PMLR, 2015, pp. 1613–1622.

[7] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, "Variational inference: A review for statisticians," *Journal of the American Statistical Association*, vol. 112, no. 518, pp. 859–877, 2017.

[8] L. Wang, X. Zhang, H. Su, and J. Zhu, "A comprehensive survey of continual learning: Theory, method and application," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[9] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016.

[10] J. Yoon, E. Yang, J. Lee, and S. J. Hwang, "Lifelong learning with dynamically expandable networks," in *6th International Conference on Learning Representations, ICLR*, 2018.

[11] C. Hung, C. Tu, C. Wu, C. Chen, Y. Chan, and C. Chen, "Compacting, picking and growing for unforgetting continual learning," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[12] X. Li, Y. Zhou, T. Wu, R. Socher, and C. Xiong, "Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting," in *International Conference on Machine Learning (ICML)*, 2019, pp. 3925–3934.

[13] X. Nie, S. Xu, X. Liu, G. Meng, C. Huo, and S. Xiang, "Bilateral memory consolidation for continual learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 16 026–16 035.

[14] R. Ramesh and P. Chaudhari, "Model zoo: A growing brain that learns continually," in *International Conference on Learning Representations (ICLR)*, 2022. [Online]. Available: https://openreview.net/forum?id=WfvgGBcgbE7

[15] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, vol. 114, no. 13, pp. 3521–3526, mar 2017. [Online]. Available: https://doi.org/10.1073%2Fpnas.1611835114

[16] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017.

[17] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *International conference on machine learning (ICML)*, 2017, pp. 3987–3995.

[18] A. Chaudhry, P. K. Dokania, T. Ajanthan, and P. H. Torr, "Riemannian walk for incremental learning: Understanding forgetting and intransigence," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 532–547.

[19] S. I. Mirzadeh, M. Farajtabar, R. Pascanu, and H. Ghasemzadeh, "Understanding the role of training regimes in continual learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 7308–7320, 2020.

[20] R. Aljundi, F. Babiloni, M. Elhoseiny, M. Rohrbach, and T. Tuytelaars, "Memory aware synapses: Learning what (not) to forget," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 139–154.

[21] G. Yang, C. S. Y. Wong, and R. Savitha, "Robust continual learning through a comprehensively progressive bayesian neural network," *arXiv preprint arXiv:2202.13369*, 2022.

[22] A. ROBINS, "Catastrophic forgetting, rehearsal and pseudorehearsal," *Connection Science*, vol. 7, no. 2, pp. 123–146, 1995.

[23] D. Shim, Z. Mai, J. Jeong, S. Sanner, H. Kim, and J. Jang, "Online class-incremental continual learning with adversarial shapley value," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 11, 2021, pp. 9630–9638.

[24] P. Buzzega, M. Boschini, A. Porrello, and S. Calderara, "Rethinking experience replay: a bag of tricks for continual learning," in *IEEE 25th International Conference on Pattern Recognition (ICPR)*, 2021, pp. 2180–2187.

[25] X. Jin, A. Sadhu, J. Du, and X. Ren, "Gradient-based editing of memory examples for online task-free continual learning," *Advances in Neural Information Processing Systems*, vol. 34, pp. 29 193–29 205, 2021.

[26] Z. Sun, Y. Mu, and G. Hua, "Regularizing second-order influences for continual learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 20 166–20 175.

[27] D. J. C. Mackay, *Bayesian methods for adaptive models*. California Institute of Technology, 1992.

[28] R. M. Neal, *Bayesian learning for neural networks*. Springer Science & Business Media, 2012, vol. 118.

[29] G. E. Hinton and D. van Camp, "Keeping the neural networks simple by minimizing the description length of the weights," in *Proceedings of the Sixth Annual Conference on Computational Learning Theory*. New York, NY, USA: Association for Computing Machinery, 1993, p. 5–13. [Online]. Available: https://doi.org/10.1145/168304.168306

[30] J. M. Hernández-Lobato and R. Adams, "Probabilistic backpropagation for scalable learning of bayesian neural networks," in *International conference on machine learning (ICML)*, 2015, pp. 1861–1869.

[31] S. Ebrahimi, M. Elhoseiny, T. Darrell, and M. Rohrbach, "Uncertainty-guided continual learning with bayesian neural networks," in *International Conference on Learning Representations (ICLR)*, 2020.

[32] N. Loo, S. Swaroop, and R. Turner, "Generalized variational continual learning," in *International Conference on Learning Representations (ICLR)*, 2021.

[33] N. Skatchkovsky, H. Jang, and O. Simeone, "Bayesian continual learning via spiking neural networks," *Frontiers in Computational Neuroscience*, vol. 16, p. 1037976, 2022.

[34] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[35] C. Bishop, "Pattern recognition and machine learning," *Springer google schola*, vol. 2, pp. 35–42, 2006.

[36] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, San Diega, CA, USA, 2015.

[37] L. Deng, "The mnist database of handwritten digit images for machine learning research," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141–142, 2012.

[38] R. I. of Technology, "Research computing services," 2019. [Online]. Available: https://www.rit.edu/researchcomputing/

[39] G. M. van de Ven and A. S. Tolias, "Three continual learning scenarios," in *NeurIPS Continual Learning Workshop*, vol. 1, no. 9, 2018.

[40] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "icarl: Incremental classifier and representation learning," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 2001–2010.

[41] Z. Wang, Z. Zhang, C.-Y. Lee, H. Zhang, R. Sun, X. Ren, G. Su, V. Perot, J. Dy, and T. Pfister, "Learning to prompt for continual learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 139–149.

[42] D. Lopez-Paz and M. Ranzato, "Gradient episodic memory for continual learning," *Advances in neural information processing systems*, vol. 30, 2017.