LIQA: Lifelong Blind Image Quality Assessment

Jianzhao Liu, Wei Zhou, Xin Li, *Graduate Student Member, IEEE*
Jiahua Xu, and Zhibo Chen, *Senior Member, IEEE*

***Abstract*—The image distortions are complex and dynamically changing in the real-world scenario, due to the fast development of the image processing system. The blind image quality assessment (BIQA) models may encounter the challenge of processing images with distortion types never seen before deployment. However, existing BIQA models generally cannot evolve with unseen distortion types adaptively, which greatly limits the deployment and application of BIQA models in real-world scenarios. To address this problem, we propose a novel Lifelong blind Image Quality Assessment (LIQA) approach, targeting to achieve the lifelong learning of BIQA. Without accessing to previous training data, our proposed LIQA can not only learn new knowledge, but also mitigate the catastrophic forgetting of learned knowledge. Specifically, we adopt the Split-and-Merge distillation strategy to train a single-head network that makes task-agnostic predictions. In the split stage, we first employ a distortion-specific generator to generate pseudo features of each previously seen distortion. Then, we utilize an auxiliary multi-head regression network to keep the response of each distortion. In the merge stage, we replay the pseudo features and use the pseudo labels generated by the auxiliary multi-head network to distill the knowledge of the multiple heads, which can build the final regression single head. Extensive experiments demonstrate that LIQA can perform well in handling both inner-dataset distortion shift and cross-dataset distortion shift. More importantly, our model can achieve stable performance even if the task sequences are long.**

***Index Terms*—Blind image quality assessment, lifelong learning, Split-and-Merge distillation, pseudo memory replay.**

I. Introduction

BLIND image quality assessment (BIQA) is a challenging problem, which aims to automatically predict perceptual image quality without any information of reference images. It has received widespread attention due to the high demand in practical applications where reference images are difficult to obtain or even unavailable. Since various distortions would be generated at each stage of signal processing (e.g. acquisition, compression, and transmission), a reliable general-purpose BIQA algorithm is urgently needed. Existing general-purpose BIQA models assume that all samples are available during the training phase, which requires the retraining of the network parameters on the entire dataset in order to adapt to changes in the data distribution [1]. However, learning models incrementally in this paradigm results in catastrophic forgetting of previously learned tasks when trained on sequential tasks, a phenomenon where the performance on the original (old) set of tasks degrades dramatically [2]–[6]. A static model, obviously, is suboptimal in the face of a dynamic environment. The network which can continually accumulate knowledge over different tasks without retraining from scratch is of urgent need and significance in real-world applications.

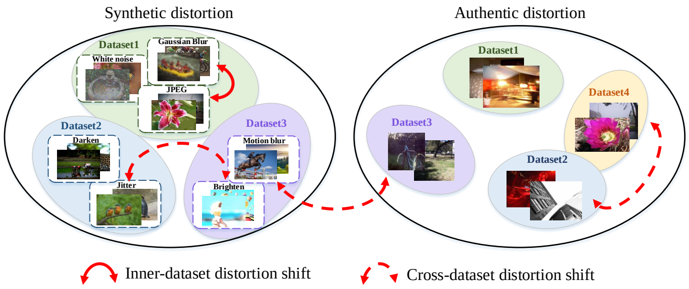


Fig. 1: Illustration of the distortion shift in IQA.

In IQA problems, distortion shift is the common and most crucial factor that leads to the catastrophic forgetting phenomenon during the sequential learning process. For example, for coding artifacts such as JPEG2000 and JPEG compression, lower sensitivities are assigned to image regions with higher activity while the artifacts in homogeneous regions are easier to observe. On the contrary, when the images are distorted by blur artifacts, strong edges are paid more attention rather than flat regions [7]. Therefore, there exist differences in human perceptual judgments of different distortion types. That is, combining different distortions could result in perceptual conflicts which further causes catastrophic forgetting.

As shown in Fig.1, the IQA data space can be categorized into various distortion scenarios (such as synthetic, authentic, screen content, VR and so on) [8]. Each distortion scenario contains multiple datasets and each dataset covers specific distortion types. In this paper, we divide the distortion shift into two levels: inner-dataset distortion shift and cross-dataset distortion shift. Here, inner-dataset distortion shift occurs when the BIQA model needs to sequentially learn novel distortions within the same dataset. For example, a BIQA model is first trained on partial distortion types in a dataset and employed in the system. Later, new distortion types are required to be added based on the same reference images. Cross-dataset distortion shift happens when the BIQA model needs to sequentially learn different datasets (under the same distortion scenario or different distortion scenarios). For example, the BIQA model is first trained on a synthetic dataset and then is provided for an authentic dataset. We define new tasks as learning new distortions covering inner-dataset distortions and cross-dataset distortions in this paper, and our LIQA targets at handling both inner-dataset distortion shift and cross-dataset distortion shift.

One straightforward way to mitigate catastrophic forgetting

This work was supported in part by NSFC under Grant U1908209, 61632001 and the National Key Research and Development Program of China 2018AAA0101400. (Jianzhao Liu and Wei Zhou contributed equally to this work.) (Corresponding author: Zhibo Chen.)

The authors are with the CAS Key Laboratory of Technology in Geo-Spatial Information Processing and Application System, University of Science and Technology of China, Hefei 230027 (e-mail: jianzhao@mail.ustc.edu.cn; weichou@mail.ustc.edu.cn; chenzhibo@ustc.edu.cn).

TABLE I: SRCC performance of each distortion of joint training when adding one novel distortion each task.

SRCC	CQ	JPEG	CSA1	WNCC	Q	JIT	IN	JP2K	CSA2	PIX	WN	CB
Initial	0.778	0.823	0.323	0.888	0.8551	0.889	0.861	-	-	-	-	-
+JP2K	0.838	0.865	0.179	0.910	0.882	0.880	0.860	0.821	-	-	-	-
+CSA2	0.754	0.820	0.381	0.886	0.772	0.907	0.881	0.859	0.835	-	-	-
+PIX	0.707	0.854	0.123	0.896	0.669	0.885	0.901	0.895	0.841	0.732	-	-
+WN	0.729	0.892	0.163	0.925	0.678	0.873	0.915	0.902	0.830	0.737	0.909	-
+CB	0.604	0.854	0.138	0.914	0.624	0.848	0.852	0.879	0.854	0.690	0.882	0.539

is joint training [9], [10], which combines all training data and retrains the model from scratch. However, this methodology is very inefficient. Sometimes, due to the storage limitation and privacy issues, previous data may not be accessible, making it hard to adopt the joint training strategy. In this paper, we focus on a more realistic and challenging setting of lifelong learning for BIQA which requires:

- **Striking a balance between stability and plasticity.** Stability refers to the ability to preserve learned knowledge and plasticity denotes the fast adaptation to new knowledge. An ideal continual learner can achieve a stability-plasticity trade-off while learning new tasks.
- **Unavailability of the previous data source.** One cannot utilize the training data of previous tasks for joint training or pre-built memory sets to replay when a new task arrives.
- **A single-head architecture that provides task-agnostic predictions.** Multi-head architecture refers to allocating specific parameters to each task and needs task identity when testing. In the real world, we cannot accurately narrate distortion types and expect a deployed model to well suit for all tasks learned before.
- **Handling two level distortion shift (i.e. inner-dataset distortion shift and cross-dataset distortion shift) simultaneously.** Task interference may occur when learning new distortion types within the current dataset or sequentially learning multiple datasets. We expect a universal framework to handle both situations.
- **Robustness to task permutations.** Task order is an important factor because in some scenarios when the performance is impaired seriously by a certain task, the learning of subsequent tasks will be affected. We expect the model to resist the negative influence of certain tasks during the whole incremental learning process.

We propose a new Lifelong blind Image Quality Assessment (LIQA) model aiming to tackle the catastrophic forgetting of BQIA. Unlike most classification methods that directly distillate knowledge from the former network, we adopt the Split-and-Merge strategy to train a single-head regression network. **In the split stage**, we first employ a generator with distortion-specific heads to memorize the data distributions of each seen distortion type. We split the regression network into a feature extractor and a prediction head. Instead of generating pseudo images, we choose to generate pseudo features before the prediction head [11]. The features are compact image representations and also the direct inputs of the regression head. Besides, features have lower dimensions compared with images (e.g. a 256×256 image can be embedded into a 512-dimensional vector) and are easier to be generated when the training data is limited. The generator is conditioned on the distortion type and the quality score. It can well control the category of pseudo features, thus avoiding the potential unbalanced problem of distortion types and quality ranges. Once the generator is trained, it can serve as a memory replayer to replay the generated pseudo features which resemble the real features of previous distortions. Then, we utilize an auxiliary multi-head regression network to generate pseudo labels with respect to the pseudo features of each distortion. **In the merge stage**, we distill the knowledge of the auxiliary multi-head regression network using the pseudo features and the corresponding pseudo labels to build the single-head regression network, avoiding the error propagation problem caused by the conflicts among different distortions of the single-head network. To illustrate the conflicts among different distortions, we utilize KADID-10K [12] database which contains 25 distortions, and randomly select 7 distortions for training the initial single-head network. Then we add one new distortion each task and utilize images of all seen distortions for joint training. We show the SRCC performance of each distortion at each task in Table I. We can see that that there exist conflicts between different distortions. For example, when adding CB, the performances of CQ, Q, IN and PIX degrade obviously. Supposing that we distill knowledge from the previous single-head network instead of the auxiliary multi-head network at the next task, some previous distortions' (e.g. CQ, Q, IN and PIX) pseudo labels are probably inaccurate due to forgetting at the completion of the previous task. When learning a new task, this will cause the continuous performance drop of these distortions due to error propagation, which is especially harmful when the task sequence is long. It should be noted that the Split-and-Merge strategy effectively resists the negative impact of a certain task (e.g. adding CB) during the sequential learning process and improves the robustness to task permutations.

In summary, our contributions are as follows:

- We drill down into the lifelong learning of BIQA and propose a LIQA framework which can effectively mitigate the catastrophic forgetting when learning new distortions.
- We adopt the Split-and-Merge strategy to train a single-head regression network, which can avoid the error propagation problem caused by the conflicts among different distortions of the single-head network.
- We design a generator that well controls the generation of pseudo features conditioned on the distortion type and the quality score. It serves as a memory replayer to consolidate the learned knowledge while learning new task, avoiding the error caused by the unbalanced distribution of distortion types and quality ranges.

- We conduct experiments to verify the effectiveness of LIQA when meeting with inner-dataset distortion shift on KADID-10K [12], and cross-dataset distortion shift on multiple datasets covering three synthetic datasets (LIVE [13], CSIQ [14] and KADID-10K [12]) and three authentic datasets (BID [15], CLIVE [16] and KonIQ-10K [17]).

The remaining parts of this paper are organized as: We review related work in Section II. In Section III, we introduce background knowledge. Our approach is presented in Section IV and experimental results are reported in Section V. We conclude our work and discuss several future directions in Section 6. Our codes will be available to the research community at http://staff.ustc.edu.cn/~chenzhibo/resources.html.

II. Related works

A. Blind image quality assessment

Unlike full-reference quality assessment methods that have access to full reference information [18]–[20], BIQA aims to automatically predict the subjective quality of a distorted image without accessing to the reference information. It can be roughly divided into two categories: distortion-specific and general-purpose approaches [21]. Distortion-specific methods are designed for a particular distortion type (e.g. blur [22], dehazing [23] and super-resolution [24], [25]). These methods deliver a poor generalization ability to other distortion types and can only be tested when the distortion type is known. In contrast, general-purpose methods that can perform across various distortion types are more practical. BRISQUE [26] utilizes statistics measured in the spatial domain and employs a generalized Gaussian distribution (GGD) model to capture various distorted image statistics. Yang et al. [27] proposed an unsupervised feature extraction approach for BIQA based on Karhunen-Loéve transform (KLT). Li et al. [28] utilized statistical structural and luminance features (NRSL) for BIQA. Freitas et al. [29] employed the statistics of the orthogonal color planes pattern (OCPP) descriptor to characterize image quality. Apart from the above-mentioned handcrafted features, there are many learning-based BIQA methods [30], [31] that usually follow a two-step network, i.e., feature extraction and quality prediction. In [32], Zhang et al. proposed a deep bilinear model that works for both synthetically and authentically distorted images. Recently, improving the training strategy of BIQA has become popular. Gao et al. [33] exploited preference image pairs to address the problem of insufficient training data. In [9], Zhang et al. learned data uncertainty and trained a deep neural network over massive image pairs by minimizing the fidelity loss. In [10], Zhang et al. further used the uncertainty training strategy and developed UNIQUE, which can obtain better generalization ability in the cross-database setting.

Although these BIQA methods have achieved great success, they obtain static models which lack the ability to evolve with unseen distortions. In contrast, our work tries to explore the sustainable learning ability of BIQA networks and thus helps BIQA networks to accumulate new knowledge and retain the learned knowledge at the same time.

B. Lifelong learning

Lifelong learning is also referred to as incremental learning or continual learning. The major challenge is to learn without catastrophic forgetting: performance on a previously learned task should not significantly degrade over time as new tasks are added. This is a direct result of a more general problem in neural networks, namely the stability-plasticity dilemma [34], where plasticity represents the ability to integrate new knowledge and stability requires that performance on previously learned tasks should not significantly degrade over time as new tasks are added. Recent works to overcome catastrophic forgetting can be roughly divided into three categories: regularization methods, parameter isolation methods [35] and replay methods. Prior-focused regularization-based approaches, such as EWC [36], online EWC [37] and SI [38], usually add a regularization term that discourages the alteration to weights important to previous tasks, which effectively prevents old knowledge from being erased or overwritten. Data-focused regularization-based methods, such as LWF [39], LFL [40] and DMC [41], employ a distillation loss to encourage the responses to previous tasks remain unchanged. Parameter isolation methods allocate task-specific parameters. One can dynamically accommodate new branches while freezing previous task parameters if there are no constraints on network size [42], [43]. When the architecture remains static, parameters of fixed parts are allocated to different tasks. HAT [44] learns hard attention masks to each task at the unit level. PackNet [45] iteratively assigns parameter subsets to consecutive tasks by constituting binary masks. Replay methods such as iCaRL [46] and ER [47] use representative samples selected from the small memory set while learning a new task. Due to the storage and privacy issues, the previous training data may be unavailable. Therefore, some replay methods such as DGR [48], PR [49], GFR [11] and BIR [50] utilize Generative Adversarial Network (GAN) to generate pseudo images or features to consolidate the learned knowledge.

Except for high-level classification tasks, lifelong learning has been applied to low-level tasks [51]–[53]. LIRA [51] adopts dynamic neural growing and pseudo image replay strategy to handle the lifelong learning of image restoration from unknown blended distortions. PIGWM [52] utilizes a parameter importance guided weights modification approach to address the lifelong learning of image de-raining. LWF-AW [53] is inspired by UNIQUE [10] and LWF [39]. It focuses on the continual learning of different IQA datasets. In this paper, we focus on a more general lifelong learning setting. We expect the model to handle both the inner-dataset distortion shift and the cross-dataset distortion shift. Actually, it still remains a challenge for the BQIA due to the particularity of quality assessment. First, the size of IQA datasets is limited. Second, the quality label of IQA is continuous instead of discrete like classification. It lacks the obvious boundaries of data due to the low aggregation of data especially for the authentically distorted images. In this case, we take steps toward the lifelong learning of BIQA and try to find a suitable way for BIQA networks to continually learn.

III. Preliminaries

A. Problem definition

Lifelong learning [35], [54] usually considers a sequence of tasks, receiving training data of just one task at a tome to perform training until convergence. Data $(\mathcal{X}^{\mathcal{T}}, \mathcal{Y}^{\mathcal{T}})$ is randomly drawn from distribution $D_{\mathcal{T}}$, with $(\mathcal{X}^{\mathcal{T}}$ a set of data samples for task $\mathcal{T}$, and $(\mathcal{Y}^{\mathcal{T}}$ the corresponding ground truth labels. The goal is to control the statistical risk of all seen tasks given limited or no access to data $(\mathcal{X}^{\mathcal{T}}, \mathcal{Y}^{\mathcal{T}})$ from previous tasks $t < \mathcal{T}$:

$$\sum_{t=0}^{\mathcal{T}} \mathbb{E}_{(\mathcal{X}^t, \mathcal{Y}^t)} \left[\ell \left(f_t \left(\mathcal{X}^t; \theta \right), \mathcal{Y}^t \right) \right], \tag{1}$$

with loss function ℓ, parameters θ, and f_t representing the network function for t-th task .

In this paper, we first define a task sequence $\mathbf{T} = \{T_t\}_{t=0}^{N}$, where T_0 is the base task and N denotes the total number of novel tasks. Each task T_t consists of a set of new distortion types. During learning current task $T_{\mathcal{T}}$, we can only get access to training data $D_{\mathcal{T}} = (\mathcal{X}^{\mathcal{T}}, \mathcal{S}^{\mathcal{T}}) = \{(\mathbf{x}_i^{\mathcal{T}}, s_i^{\mathcal{T}})\}_{i=1}^{n_{\mathcal{T}}}$, where $\mathbf{x}_i^{\mathcal{T}}$ represents the distorted image, $s_i^{\mathcal{T}}$ represents the ground truth perceptual quality score and $n_{\mathcal{T}}$ denotes the number of the $\mathcal{T}$-th task's training data. For any $t_1 \neq t_2$, $D_{t_1} \cap D_{t_2} = \emptyset$. T_0 is denoted as a base task, which consists of M_0 distortion types. Supposing the total number of distortion types as M_{all}, we can sequentially add $\Delta = \frac{M_{all} - M_0}{N}$ distortion types per task, resulting in N novel tasks (i.e. $T_1 - T_N$). We can say that the incremental step is equal to Δ. After the model has incrementally been trained up to $\mathcal{T}$-th task ($\mathcal{T} > 0$), we denote $M_{cur} = M_0 + \Delta * \mathcal{T}$ as the number of distortion types seen so far and denote $M_{pre} = M_0 + \Delta * (\mathcal{T} - 1)$ as the number of all previously seen distortions before learning the current task.

B. Evaluation metrics

For all experiments, we specially design two evaluation metrics for lifelong learning of BIQA: Correlation Index (C) and Forgetting Index (F), following the previous works [55], [56].

Correlation Index (C) After learning the $\mathcal{T}$-th task, we evaluate the average Spearman's Rank-order Correlation Coefficient (SRCC) between the predicted quality scores and the MOS/DMOS on the held-out test images of each seen distortion type. The correlation index $C_{\mathcal{T}}$ is defined as $C_{\mathcal{T}} = \frac{1}{M_{cur}} \sum_{j=0}^{M_{cur}-1} abs(\mathrm{SRCC}_{\mathcal{T},j})$, which is within the range of [0, 1]. The higher value means the better consistency with human opinions of perceptual quality.

Forgetting Index (F) We define the forgetting degree of a particular distortion as the difference between the maximum performance of this distortion through out the learning process in the past and the performance the current model has about it. The forgetting of the j-th distortion after the model has incrementally been trained up to $\mathcal{T}$-th task can be defined as:

$$f_j^{\mathcal{T}} = \max_{t \in \{0, \ldots, \mathcal{T}-1\}} abs(\mathrm{SRCC}_{t,j}) - abs(\mathrm{SRCC}_{\mathcal{T},j}) \tag{2}$$

where $\mathcal{T} > 0$, $j \in [0, M_{pre})$ and $f_j^{\mathcal{T}} \in [-1, 1]$. $abs(\cdot)$ denotes the absolute value function. We can average the forgetting of all previously seen distortions and obtain the forgetting index $F_{\mathcal{T}} = \frac{1}{M_{pre}} \sum_{j=0}^{M_{pre}-1} f_j^{\mathcal{T}}$. Lower $F_{\mathcal{T}}$ means less forgetting and better stability. Especially, when $F_{\mathcal{T}} < 0$, it means that the current task not only cannot impair the previous learned knowledge but also can contribute to the performance of previous tasks.

IV. Approach

A. Network architecture of LIQA

The framework of LIQA consists of four parts: a single-head regression network $\mathcal{R}$, an auxiliary multi-head regression network $\hat{\mathcal{R}}$, a generator $\mathcal{G}$ and a discriminator $\mathcal{D}$. For the single-head regression network, we employ a pre-trained ResNet-18 (without the final FC layers) as the feature extractor U and use two $FC - ReLU$ layers followed by a $Sigmoid$ function as the prediction head V. The auxiliary multi-head regression network has distortion-specific prediction heads $\hat{V}_{j=0}^{M_all-1}$ and the architecture of the feature extractor $\hat{U}$ is the same as that of the single-head regression network's.

The architectures of the generator and the discriminator are shown in Fig. 2. Instead of sampling noise vector form the standard normal prior $\mathcal{N}(\mathbf{0}, \mathbf{I})$, we sample noise vector $\tilde{\mathbf{z}}_j$ from $\mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{\sigma}_j^2)$, where $\boldsymbol{\mu}_j$ and $\boldsymbol{\sigma}_j$ are trainable mean and standard deviation for the distortion j. We adopt the reparameterization trick [57] to generate $\tilde{\mathbf{z}}_j$. $\tilde{\mathbf{z}}_j = \boldsymbol{\mu}_j + \boldsymbol{\sigma}_j \odot \mathbf{z}$, where $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\odot$ signifies the element-wise product. This makes it possible to generate distortion-specific features by restricting the sampling of the noise vector from the corresponding distribution. The generator consists of two parts: the shared embedding layers $E_{\mathcal{G}}$ which embed the noise vector to a latent vector and the distortion-specific generation head G_j which takes in the summation of the latent vector and the quality score to generate quality-related distortion-specific pseudo features $\tilde{\mathbf{h}}_j$.

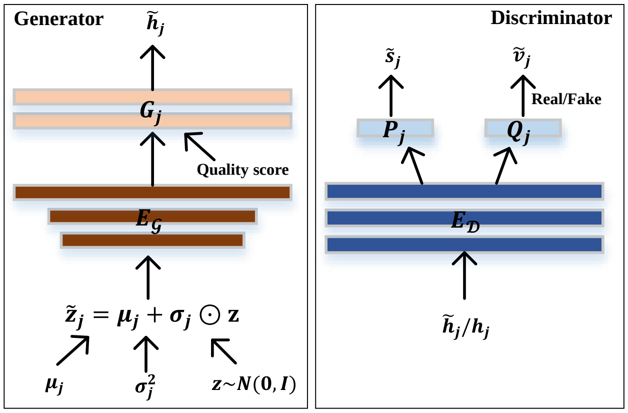


Fig. 2: Architectures of the generator and the discriminator.

The discriminator consists of three parts: the shared embedding layers $E_{\mathcal{D}}$ which embed the input feature vector to a latent vector, the distortion-specific quality prediction heads P_j which regresses the latent vector into a quality score $\tilde{s}_j$and the distortion-specific discrimination head Q_j which tells whether the input feature is real or fake.

Generally speaking, the generator is conditioned on the distortion index j and the quality score s, which can be denoted by $\tilde{\mathbf{h}}_j = \mathcal{G}(\mathbf{z}, s, j)$. The discriminator takes in the pseudo/real feature $\tilde{\mathbf{h}}_j/\mathbf{h}_j$ together with the distortion index j, which can be denoted by $(\tilde{s}_j, \tilde{v}_j) = \mathcal{D}(\tilde{\mathbf{h}}_j/\mathbf{h}_j, j)$.

B. Training strategy

The whole framework of LIQA is shown in Fig. 3. In the merge stage, we train the single-head regression network with the pseudo features replaying. In the split stage, we incrementally train the generator and the discriminator and then train the auxiliary multi-head regression network to learn each seen distortion type separately.

1) Training single-head regression network: Let us denote the current task by $T_{\mathcal{T}}$. The generator and the discriminator trained at the split stage at the former task is $\mathcal{G}_{\mathcal{T}-1}$ and $\mathcal{D}_{\mathcal{T}-1}$. The feature extractor and the prediction head of the current single-head network $\mathcal{R}_{\mathcal{T}}$ is $U_{\mathcal{T}}$ and $V_{\mathcal{T}}$ respectively. The prediction heads of the auxiliary multi-head regression network trained at previous tasks are $\hat{V}_{j_{t<\mathcal{T}}}$, where the subscript $j_{t<\mathcal{T}}$ denotes the index of the distortions at task $\mathcal{T}_{t<\mathcal{T}}$.

Training the feature extractor. Instead of freezing some layers of the feature extractor like [40], we fine-tune all the parameters of the feature extractor. Moreover, we employ feature distillation loss to prevent the forgetting of old knowledge and guarantee the stability of the feature extractor. The feature distillation loss during task $T_{\mathcal{T}}$ is defined as:

$$L^{FD}_{\mathcal{T}} = \mathbb{E}_{\mathbf{x}\sim\mathcal{X}^{\mathcal{T}}}\left[\|U_{\mathcal{T}}(\mathbf{x}) - U_{\mathcal{T}-1}(\mathbf{x})\|_2\right], \tag{3}$$

which is used to constrain that the features extracted by $U_{\mathcal{T}}$ do not drift far away from that by $U_{\mathcal{T}-1}$.

Training the prediction head. For training the prediction head, we employ pseudo replay loss to consolidate the knowledge of previous distortions and use L_2 loss to learn the new distortions arriving at $T_{\mathcal{T}}$. Given a random quality score $\bar{s}$, a random noise $\mathbf{z}$ and the distortion index $j_{t<\mathcal{T}}$, $\mathcal{G}_{\mathcal{T}-1}$ can generate the pseudo quality-related feature of distortion $j_{t<\mathcal{T}}$: $\tilde{\mathbf{h}}_{j_{t<\mathcal{T}}} = \mathcal{G}_{\mathcal{T}-1}(\mathbf{z}, \bar{s}, j_{t<\mathcal{T}})$. The pseudo replay loss can be defined as:

$$L^{PR} = \mathbb{E}_{z\sim\mathcal{N}(\mathbf{0},\mathbf{1}), s\sim\bar{\mathcal{S}}, j\sim p_{j_{t<\mathcal{T}}}}\left[\left\|\hat{V}_j(\tilde{h}_j) - V_{\mathcal{T}}(\tilde{h}_j)\right\|_2\right], \tag{4}$$

where $\tilde{h}_j = \mathcal{G}_{\mathcal{T}-1}(\mathbf{z}, s, j)$. $s \sim \bar{\mathcal{S}}$ means the quality score is randomly generated and $p_{j_{t<\mathcal{T}}}$ is the distortion index distribution of previous tasks $T_{t<\mathcal{T}}$. $\hat{V}_j$ is the prediction head for distortion j of the auxiliary multi-head regression network. For current task's training data, we adopt L_2 loss for training:

$$L^{MSE1}_{\mathcal{T}} = \mathbb{E}_{(\mathbf{x},s)\sim D^{\mathcal{T}}}\left[\|V_{\mathcal{T}}(U_{\mathcal{T}}(\mathbf{x})) - s\|_2\right] \tag{5}$$

Full objective. In summary, when $\mathcal{T} = 0$, we only have L_2 loss for training the single-head regression network:

$$L^{Total}_{\mathcal{T}} = L^{MSE1}_{\mathcal{T}}. \tag{6}$$

During task $T_{\mathcal{T}}$ ($\mathcal{T} > 0$), the total training loss is:

$$L^{Total}_{\mathcal{T}} = \lambda_{FD} L^{FD}_{\mathcal{T}} + \lambda_{PR} L^{PR}_{\mathcal{T}} + \lambda_{MSE} L^{MSE1}_{\mathcal{T}}, \tag{7}$$

where λ_{FD}, λ_{PR} and λ_{MSE} are hyper-parameters that control the relative importance of feature distillation loss, pseudo replay loss and L_2 loss respectively.

2) Training generator and discriminator: We freeze the single-head network trained at task $T_{\mathcal{T}}$ and train the generator $\mathcal{G}_{\mathcal{T}}$ to continually learn the current task's feature distribution. It should be noted that the trainable $\boldsymbol{\mu}_{j_{t<\mathcal{T}}}$, $\boldsymbol{\sigma}_{j_{t<\mathcal{T}}}$ and the previously learned generation heads $G_{j_{t<\mathcal{T}}}$ of $\mathcal{G}_{\mathcal{T}}$ are also frozen. Similarly, the previous quality prediction heads $P_{j_{t<\mathcal{T}}}$ and discrimination head $Q_{j_{t<\mathcal{T}}}$ of $\mathcal{D}_{\mathcal{T}}$ are frozen.

Adversarial loss. To make the generated pseudo features indistinguishable from real features, we adopt an adversarial loss:

$$\begin{aligned} L^{adv}_{\mathcal{T}} =& \mathbb{E}_{\mathbf{x}\sim\mathcal{X}^{\mathcal{T}}, j\sim p_{j_{t=\mathcal{T}}}}\left[\log \mathcal{D}^{r/f}_{\mathcal{T}}(U_{\mathcal{T}}(\mathbf{x}), j))\right] + \\ & \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(\mathbf{0},\mathbf{I}), s\sim\mathcal{S}^{\mathcal{T}}, j\sim p_{j_{t=\mathcal{T}}}}\left[\log(1 - \mathcal{D}^{r/f}_{\mathcal{T}}(\mathcal{G}_{\mathcal{T}}(\mathbf{z}, s, j), j))\right] \end{aligned} \tag{8}$$

where $\mathcal{D}^{r/f}_{\mathcal{T}}$ is the discrimination head for distortion j which gives the real/fake probability. $\mathcal{G}_{\mathcal{T}}$ tries to generate a feature conditioned on both the quality score s and the distortion index j, while $\mathcal{D}_{\mathcal{T}}$ tries to distinguish between the real and the fake features of distortion j.

Quality prediction loss. To generate pseudo features corresponding to the quality score s, we add auxiliary quality prediction heads on top of the discriminator and impose the quality prediction loss when optimizing both $\mathcal{D}_{\mathcal{T}}$ and $\mathcal{G}_{\mathcal{T}}$: a quality prediction loss of real features used to optimize $\mathcal{D}_{\mathcal{T}}$, and a quality prediction loss of fake features used to optimize $\mathcal{G}_{\mathcal{T}}$. In detail, the former is defined as

$$L^{qua\_r}_{\mathcal{T}} = \mathbb{E}_{(\mathbf{x},s)\sim D^{\mathcal{T}}, j\sim p_{j_{t=\mathcal{T}}}}\left[\|\mathcal{D}^{qua}_{\mathcal{T}}(U_{\mathcal{T}}(\mathbf{x}), j)), s\|_2\right], \tag{9}$$

where the term $\mathcal{D}^{qua}_{\mathcal{T}}$ represents the quality prediction head which predicts the quality value. On the other hand, the quality prediction loss of fake features is defined as

$$L^{qua\_f}_{\mathcal{T}} = \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(\mathbf{0},\mathbf{I}), s\sim\mathcal{S}^{\mathcal{T}}, j\sim p_{j_{t=\mathcal{T}}}}\left[\|\mathcal{D}^{qua}_{\mathcal{T}}(\mathcal{G}_{\mathcal{T}}(\mathbf{z}, s, j), j), s\|_2\right]. \tag{10}$$

By minimizing this objective, $\mathcal{G}_{\mathcal{T}}$ learns to generate quality-related features for specific distortion.

Alignment loss. During the training process, we synchronize $\mathcal{G}_{\mathcal{T}}$ with $\mathcal{G}_{\mathcal{T}-1}$, which means that the previous distortion features generated by $\mathcal{G}_{\mathcal{T}}$ should be the same as that generated by $\mathcal{G}_{\mathcal{T}-1}$. The generator alignment loss is defined as

$$L^{GA}_{\mathcal{T}} = \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(\mathbf{0},\mathbf{I}), s\sim\bar{\mathcal{S}}, j\sim p_{j_{t<\mathcal{T}}}}\left[\|\mathcal{G}_{\mathcal{T}}(\mathbf{z}, s, j) - \mathcal{G}_{\mathcal{T}\text{-}1}(\mathbf{z}, s, j)\|_2\right]. \tag{11}$$

Similarly, we also apply alignment loss to the current discriminator, which encourages the quality prediction values and the real/fake probability towards previous distortion features to be the same as that of the former discriminator.

$$\begin{aligned} L^{DA}_{\mathcal{T}} =& \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(\mathbf{0},\mathbf{I}), s\sim\bar{\mathcal{S}}, j\sim p_{j_{t<\mathcal{T}}}}\left[\left\|\mathcal{D}^{qua}_{\mathcal{T}}(\tilde{h}^{\mathcal{T}-1}_j, j) - \mathcal{D}^{qua}_{\mathcal{T}\text{-}1}(\tilde{h}^{\mathcal{T}-1}_j, j)\right\|_2\right] + \\ & \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(\mathbf{0},\mathbf{I}), s\sim\bar{\mathcal{S}}, j\sim p_{j_{t<\mathcal{T}}}}\left[\left\|\mathcal{D}^{r/f}_{\mathcal{T}}(\tilde{h}^{\mathcal{T}-1}_j, j) - \mathcal{D}^{r/f}_{\mathcal{T}\text{-}1}(\tilde{h}^{\mathcal{T}-1}_j, j)\right\|_2\right], \end{aligned} \tag{12}$$

where $\tilde{h}^{\mathcal{T}-1}_j = \mathcal{G}_{\mathcal{T}-1}(\mathbf{z}, s, j)$ denotes the pseudo features of distortion j generated by $\mathcal{G}_{\mathcal{T}-1}$.

Full objective. When $\mathcal{T} > 0$, the objective functions to optimize $G_{\mathcal{T}}$ and $D_{\mathcal{T}}$ are written respectively as:

$$L^{\mathcal{G}}_{\mathcal{T}} = -L^{adv}_{\mathcal{T}} + \lambda_{qua} L^{qua\_r}_{\mathcal{T}} + \lambda_{align} L^{GA}_{\mathcal{T}}, \tag{13}$$

$$L^{\mathcal{D}}_{\mathcal{T}} = L^{adv}_{\mathcal{T}} + \lambda_{qua} L^{qua\_f}_{\mathcal{T}} + \lambda_{align} L^{DA}_{\mathcal{T}}, \tag{14}$$

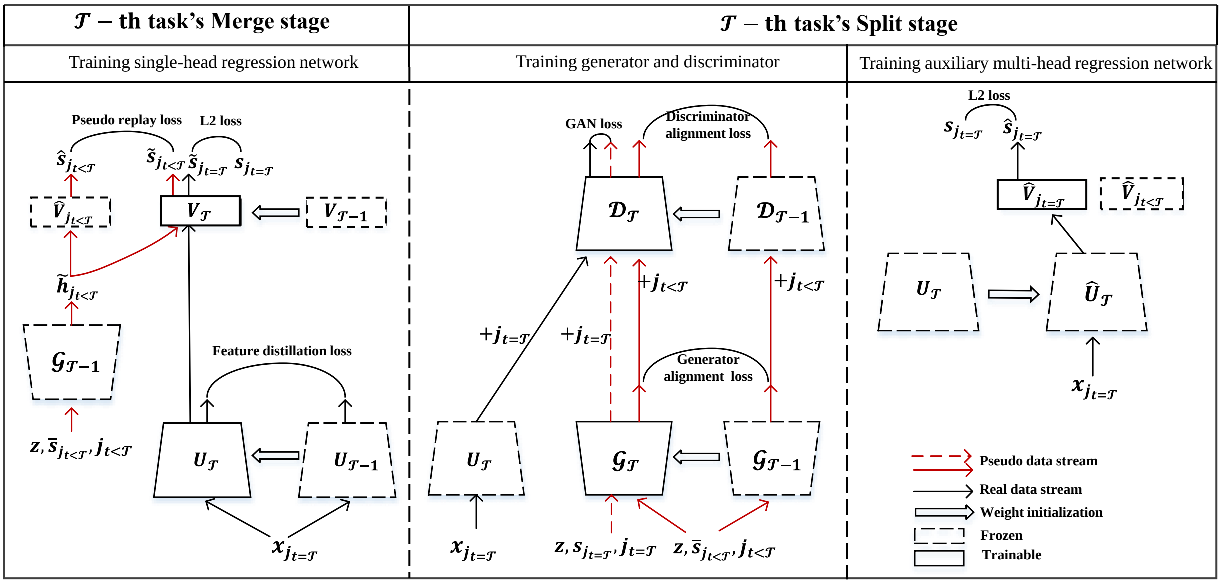

$\mathcal{T}$ – th task's Merge stage
$\mathcal{T}$ – th task's Split stage
Training single-head regression network
Training generator and discriminator
Training auxiliary multi-head regression network
Pseudo replay loss
L2 loss
$\hat{s}_{j_{t<\mathcal{T}}}$
$\bar{s}_{j_{t<\mathcal{T}}}$
$\hat{s}_{j_{t=\mathcal{T}}}$
$s_{j_{t=\mathcal{T}}}$
$\hat{V}_{j_{t<\mathcal{T}}}$
$V_{\mathcal{T}}$
$V_{\mathcal{T}-1}$
$\tilde{h}_{j_{t<\mathcal{T}}}$
$\mathcal{G}_{\mathcal{T}-1}$
$z, \bar{s}_{j_{t<\mathcal{T}}}, j_{t<\mathcal{T}}$
Feature distillation loss
$U_{\mathcal{T}}$
$U_{\mathcal{T}-1}$
$x_{j_{t=\mathcal{T}}}$
GAN loss
Discriminator alignment loss
$\mathcal{D}_{\mathcal{T}}$
$\mathcal{D}_{\mathcal{T}-1}$
$+j_{t=\mathcal{T}}$
$+j_{t=\mathcal{T}}$
$+j_{t<\mathcal{T}}$
$+j_{t<\mathcal{T}}$
Generator alignment loss
$U_{\mathcal{T}}$
$\mathcal{G}_{\mathcal{T}}$
$\mathcal{G}_{\mathcal{T}-1}$
$x_{j_{t=\mathcal{T}}}$
$z, s_{j_{t=\mathcal{T}}}, j_{t=\mathcal{T}}$
$z, \bar{s}_{j_{t<\mathcal{T}}}, j_{t<\mathcal{T}}$
L2 loss
$s_{j_{t=\mathcal{T}}}$
$\hat{s}_{j_{t=\mathcal{T}}}$
$\hat{V}_{j_{t=\mathcal{T}}}$
$\hat{V}_{j_{t<\mathcal{T}}}$
$U_{\mathcal{T}}$
$\hat{U}_{\mathcal{T}}$
$x_{j_{t=\mathcal{T}}}$
Pseudo data stream
Real data stream
Weight initialization
Frozen
Trainable

Fig. 3: Framework of LIQA. LIQA adopts Split-and-Merge distillation strategy to build the single-head regression network. U and $\hat{U}$ are the feature extractor of the single-head regression network and the multi-head regression network. V and $\hat{V}$ are the prediction head of the single-head regression network and the multi-head regression network. $\mathcal{G}$ is the generator and $\mathcal{D}$ is the discriminator. $\mathbf{z}$, s and j denote the random noise vector, the quality score and the distortion index respectively.

where λ_{qua} and λ_{align} are hyper-parameters controlling the relative importance of quality prediction loss and alignment loss, compared to the adversarial loss. When $\mathcal{T} = 0$, we do not have the alignment loss item.

3) Training auxiliary multi-head regression network: The feature extractor $\hat{U}_{\mathcal{T}}$ of the multi-head regression network $\hat{\mathcal{R}}_{\mathcal{T}}$ is initialized with the feature extractor $U_{\mathcal{T}}$ of the single-head network $\mathcal{R}_{\mathcal{T}}$. The training objective is defined as:

$$L^{MSE2} = \mathbb{E}_{(\mathbf{x},s)\sim D^{\mathcal{T}}, j\sim p_{j_{t=\mathcal{T}}}}\left[\left\|\hat{\mathcal{R}}_{\mathcal{T}}(\mathbf{x}, j), s\right\|_2\right], \qquad (15)$$

where the feature extractor $\hat{U}_{\mathcal{T}}$ and the previous distortions' prediction heads $\hat{V}_{j_{t<\mathcal{T}}}$ of $\hat{\mathcal{R}}_{\mathcal{T}}$ are frozen and only the current task's distortion prediction heads $\hat{V}_{j_{t=\mathcal{T}}}$ are trainable.

V. Experiments

A. Datasets

Following work [10], we conduct experiments on six IQA datasets, among which three are synthetically distorted (LIVE [13], CSIQ [14], KADID-10K [12]) and the others are authentically distorted (BID [15], CLIVE [16] and KonIQ-10K [12]). The summarization of the six datasets is shown in Table II.

The LIVE database includes 779 synthetically distorted images, which are generated from 29 reference images by corrupting them with five distortion types, i.e. JPEG-2000 compression, JPEG compression, white Gaussian noise, Gaussian blur, and fast fading rayleigh at five to eight intensity levels. DMOS of each distorted image ranges from 1 to 100 and is collected using the single stimulus continuous quality rating method. The CSIQ database consists of 866 synthetically distorted images which are derived from 30 original images distorted with six distortion types at four to five different intensity levels. The distortions are JPEG compression, JPEG-2000 compression, global contrast decrements, additive pink Gaussian noise, additive white Gaussian noise, and Gaussian blurring. The ratings are reported in the form of DMOS ranging from 0 to 1 using multi-stimulus absolute category rating method. KADID-10K consists of 10,125 distorted images derived from 81 pristine images considering 25 different distortion types at 5 intensity levels. The distortion types include Gaussian blur (GB), lens blur (LB), motion blur (MB), color diffusion (CD), color shift (CS), color quantization (CQ), two kinds of color saturation (CSA1 and CSA2), JPEG-2000 compression (JP2K), JPEG compression (JPEG), white noise (WN), white noise in color component (WNCC), impulse noise (IN), multiplicative noise (MN), denoise (DN), brighten (BR), darken (DA), mean shift (MS), jitter (JIT), non-eccentricity patch (NEP), pixelate (PIX), quantization (Q), color block (CB), high sharpen (HS), and contrast change (CC). The MOS of each image ranges from 1 to 5 and is collected using double stimulus absolute category rating with crowdsourcing.

The BID database contains 586 authentically distorted pictures taken by human users in a variety of scenes, camera apertures, and exposition times. The distorted images are mostly blurred, which not only include typical, easy-to-model blurring cases but also more complex, realistic ones. The MOS of each image ranges from 0 to 5 and is collected using the single stimulus continuous rating method. The CLIVE database contains 1,162 authentically distorted images captured from diverse mobile devices. Each image is collected without artificially introducing any distortions beyond those occurring during capture, processing, and storage by a user's device. The MOS of each image ranges from 0 to 100 and is collected by the single stimulus continuous quality rating with

TABLE II: Description of IQA databases. DisNum refers to the number of synthetic distortion types. MOS refers to Mean Opinion Score and a higher value denotes better perceptual quality. DMOS refers to Differential Mean Opinion Score and is inversely proportional to MOS. DisImageNum refers to the number of distorted images.

Database	Scenario	DisNum	Subjective Testing Methodology	Annotation	Range	DistImageNum
LIVE [13]	Synthetic	5	Single stimulus continuous quality rating	DMOS	[0,100]	779
CSIQ [14]	Synthetic	6	Multi stimulus absolute category rating	DMOS	[0,1]	866
KADID-10K [12]	Synthetic	25	Double stimulus absolute category rating with crowdsourcing	MOS	[1,5]	10,125
BID [15]	Authentic	-	Single stimulus continuous quality rating	MOS	[0,5]	586
CLIVE [16]	Authentic	-	Single stimulus continuous quality rating with crowdsourcing	MOS	[0,100]	1,162
KonIQ-10K [17]	Authentic	-	Single stimulus absolute category rating with crowdsourcing	MOS	[1,5]	10,073

crowdsourcing. The KonIQ-10K database consists of 10,073 authentically distorted images selected from a massive public multimedia database, YFCC100m [58]. The MOS of each image ranges from 1 to 5 and is collected by the single stimulus absolute category rating with crowdsourcing.

B. Implementation details

In order to test the effectiveness of LIQA facing with inner-dataset distortion shift, we adopted KADID-10K dataset and randomly split the 25 distortion types into two groups, i.e. a base group and a novel group. The base group includes 7 distortion types (CQ, JPEG, CSA1, WNCC, Q, JIT and IN) and is used for training the base task. The novel group includes 18 distortion types (JP2K, CSA2, PIX, WN, CB, GB, DA, CC, BR, NEP, MS, MB, MN, LB, DN, HS, CD and CS) and can be divided into 18, 9 and 3 novel tasks with the incremental step Δ set to 1, 2 and 6 respectively in our experiments. Then we randomly permuted the distortions in the novel group to test the robustness to the distortion order. Moreover, in order to test the effectiveness of LIQA when facing with cross-dataset distortion shift, we sequentially add one dataset per task.

Following the work of UNIQUE [10], we randomly sampled 80% images from each dataset for training, 10% for validation and the left 10% for testing. Specially, for the three synthetic datasets, we split the datasets according to the reference images in order to ensure content dependence. During training, we randomly cropped the images into 300×300 and during validation and testing, we cropped the images to 300×300 in the center. For each task, we trained the single-head regression network for 70 epochs, the generator and the discriminator for 500 epochs and the multi-head regression network for 70 epochs. We adopted an early-stopping strategy and chose the model that performed the best on the validation set for testing. Specially, we selected the best-performing network after 15 epochs to make sure the learning of the current task. For training the generator, we adopted a data augmentation strategy and expand the size of the original dataset tenfold offline, because the image for each distortion type are too few to train a good generator. But for training the regression network, we adopted the original dataset. Each experiment was run five times and the results were averaged.

We follow the protocol in [38] and tune hyper-parameters using coarse grid research strategy on the held-out validation set with the searching scope set to [0.0001, 0.001, 0.1, 1.0, 3.0, 5.0, 10.0, 20.0]. We select the hyper-parameters that achieve the best performance during the whole incremental learning process. Moreover, the optimal parameters are suitable for both inner-dataset and cross-dataset distortion shift lifelong learning scenarios. λ_{FD}, λ_{PR} and λ_{MSE} in Eq. 7 are set to 0.001, 10.0, 1.0 respectively. We set λ_{qua} and λ_{align} in Eq. 13 as well as Eq. 14 to 1.0 and 3.0, respectively. The learning rate of the feature extractor and the prediction heads for regression networks was set to $1e^{-4}$ when learning the base task. We lowered down the learning rate of the feature extractor to $1e^{-6}$ when learning the novel tasks. For simplicity, we linearly re-scaled the subjective scores of each of the six databases to [0, 1] [10], where higher value denotes better perceptual quality. The regression networks were trained using Adam with a batch size of 48 and the buffer size of pseudo features per batch was set to 1400. To generate pseudo features for each previous distortion, we split the re-scaled quality range into five interval segments, i.e. [0, 0.2), [0.2, 0.4), [0.4, 0.6), [0.6, 0.8), [0.8, 1.0]. For each quality interval, we allocated $1400/M_{pre}/5$ pseudo features to make sure that the generated pseudo features can well cover the distribution of the real features. The generator as well as the discriminator were trained using Adam with a batch size of 128.

C. Compared methods

Apart from fine-tuning and joint training, we also compare LIQA with three prior-focused regularization-based lifelong learning approaches, i.e. EWC [36], online EWC [37] and SI [38], one data-focused regularization-based method, i.e. LWF [39] as well as one replay method, i.e. GFR [11]. Besides, we also show the performance of the auxiliary multi-head regression network, which is denoted as “LIQA-multihead” in Fig. 4, Fig. 5, Fig. 6 and Fig. 7. Considering that we target at task-agnostic predictions, we do not compare LIQA with parameter isolation methods which require task-specific parameters and task identity during testing. The regression networks of all the compared approaches were trained with Adam with a batch size of 48 and the learning rate of $1e^{-4}$. We chose the best-performed model during validation for testing as the training process of LIQA.

Fine-tuning (FT): Modifying the parameters of an existing network to adapt to a new task. At current task $\mathcal{T}_{\mathcal{T}}$, we directly fine-tune the single-head regression network initialized by $\mathcal{R}_{\mathcal{T}-1}$ using the current task's training data $D_{\mathcal{T}}$. The loss of FT is defined as:

$$L_{\mathcal{T}}^{FT} = \mathbb{E}_{(\mathbf{x},s)\sim D_{\mathcal{T}}}\left[\|\mathcal{R}_{\mathcal{T}}(\mathbf{x}) - s\|_2\right]. \qquad (16)$$

EWC: EWC estimates the importance of parameter i by the i-th diagonal element of the current task $T_{\mathcal{T}}$'s Fisher

Information matrix, the regularization loss is defined as

$$L_{\mathcal{T}}^{reg-EWC} = \frac{1}{2}\sum_{t=0}^{\mathcal{T}-1}\Big(\sum_{i=1}^{N_{params}} \mathbf{F}_{ii}^{t}(\theta_i - \hat{\theta}_i^{t})^2\Big), \tag{17}$$

where $\hat{\theta}_i^t$ is the value of the parameter i after finishing training on task T_t. $\mathbf{F}_{ii}$ can be calculated by

$$\mathbf{F}_{ii}^{t} = \frac{1}{|D^t|}\sum_{(\mathbf{x},s)\sim D^t}\Big(\frac{\partial L_2(\mathcal{R}_t(\mathbf{x};\hat{\theta}^t),s)}{\partial \theta_i}\Big)^2, \tag{18}$$

where L_2 denotes L_2 loss function. The full objective of EWC is:

$$L_{\mathcal{T}}^{EWC} = L_{\mathcal{T}}^{FT} + \lambda_{EWC} L_{\mathcal{T}}^{reg-EWC} \tag{19}$$

In our experiment, we empirically set λ_{EWC} to 5000.0 via a coarse grid search on the held-out validation set following the work of [38].

Online EWC: The regularization term for online EWC is given by:

$$L_{\mathcal{T}}^{reg-onlineEWC} = \sum_{i=1}^{N_{params}} \tilde{\mathbf{F}}_{ii}^{\mathcal{T}-1}(\theta_i - \hat{\theta}_i^{(\mathcal{T}-1)})^2, \tag{20}$$

where $\tilde{\mathbf{F}}_{ii}^{\mathcal{T}-1}$ is a running sum of the i-th diagonal elements of the Fisher Information matrices of tasks $T_{t<=\mathcal{T}-1}$, i.e. $\tilde{\mathbf{F}}_{ii}^{\mathcal{T}-1} = \gamma\tilde{\mathbf{F}}_{ii}^{\mathcal{T}-2}$ + $\mathbf{F}_{ii}^{\mathcal{T}-1}$. $\gamma <= 1$ is a hyperparameter that governs the gradual decay of the contributions of previous tasks. The full objective of online EWC is given by:

$$L_{\mathcal{T}}^{onlineEWC} = L_{\mathcal{T}}^{FT} + \lambda_{onlineEWC} L_{\mathcal{T}}^{reg-onlineEWC}. \tag{21}$$

In our experiment, we empirically set γ to 1 and $\lambda_{onlineEWC}$ to 5000.0.

SI: SI estimates the importance for each parameter and protect the parameters important to previous tasks from changing. SI loss is defined as:

$$L_{\mathcal{T}}^{reg-SI} = \sum_{i=1}^{N_{params}} \Omega_i^{(\mathcal{T}-1)}(\theta_i - \hat{\theta}_i^{(\mathcal{T}-1)})^2, \tag{22}$$

where $\hat{\theta}_i^{(\mathcal{T}-1)}$ is the value of parameter i after finishing training on task $T_{\mathcal{T}-1}$. $\Omega_i^{(\mathcal{T}-1)}$ is the estimated importance of parameter i for all the previous tasks:

$$\Omega_i^{(\mathcal{T}-1)} = \sum_{t=0}^{\mathcal{T}-1}\frac{\omega_i^t}{(\Delta_i^t)^2 + \xi}, \tag{23}$$

where $\Delta_i^t = \hat{\theta}_i[{N_{iters}}^t] - \hat{\theta}_i[0^t]$, ${N_{iters}}^t$ is the number of iterations and $\hat{\theta}_i[0^t]$ indicates the value of parameter i right before starting training on task T_t. ξ is a small value (usually set to 0.1). ω_i^t counts the contribution of parameter i to the change in loss:

$$\omega_i^t = \sum_{n=1}^{N_{iters}}(\hat{\theta}_i[n^t] - \hat{\theta}_i[(n-1)^t])\frac{-\partial L_{total}[n^t]}{\partial \theta_i}, \tag{24}$$

where $\hat{\theta}_i[n^t]$ denotes the value of the parameter i after the n-th training iteration.

The full objective of SI is given by:

$$L_{\mathcal{T}}^{SI} = L_{\mathcal{T}}^{FT} + \lambda_{SI} L_{\mathcal{T}}^{reg-SI}. \tag{25}$$

We empirically set λ_{SI} to 100.0 in our experiments via a coarse grid search on the held-out validation set following the work of [38].

LWF: Using only examples for the new task, knowledge distillation is used for optimizing new task's performance while preserving responses on the previous tasks. The distillation loss can be calculated by:

$$L_{\mathcal{T}}^{Distill} = \mathbb{E}_{(\mathbf{x},s)\sim D_{\mathcal{T}}}\left[\|\mathcal{R}_{\mathcal{T}}(\mathbf{x}) - \mathcal{R}_{\mathcal{T}-1}(\mathbf{x})\|_2\right]. \tag{26}$$

The total loss should be of the form:

$$L_{\mathcal{T}}^{LWF} = L_{\mathcal{T}}^{FT} + \lambda_{LWF} L_{\mathcal{T}}^{Distill}. \tag{27}$$

GFR: Generate pseudo features conditioned on the category labels (discrete) and utilize the generated features paired with the given hard labels for replaying. To suit for the regression task, we implement the feature generation process by:

$$\tilde{h}_c = \mathcal{G}(\mathbf{z}, c), \tag{28}$$

where $c = mapping(s, j)$ and $\mathbf{z}$ is drawn from a normalized Gaussian distribution. $mapping$ denotes a mapping function which maps the combination of the distortion index j and the MOS value into a discrete category label. We re-linear the MOS to [0, 1] and split the quality range into five interval segments. Therefore, for the inner-dataset distortion shift experiments, we obtain 125 categories (25×5) in total. In addition, for the cross-dataset distortion shift experiments, we obtain 30 categories (6×5). The replay loss is calculated by:

$$L^{Replay_{GFR}} = \mathbb{E}_{z\sim\mathcal{N}(\mathbf{0},\mathbf{1}), c\sim p_{c_{t<\mathcal{T}}}}\left[\left\|c - V_{\mathcal{T}}(\tilde{h}_c)\right\|_2\right]. \tag{29}$$

The total loss for training the regression network of GFR can be defined as follows:

$$L_{\mathcal{T}}^{GFR} = L_{\mathcal{T}}^{FT} + \lambda_{GFR} L_{\mathcal{T}}^{Replay_{GFR}}. \tag{30}$$

Joint training (JT): All the previously learned tasks' training data is stored and combined with the current task's training data for training $\mathcal{R}_{\mathcal{T}}$. $\mathcal{R}_{\mathcal{T}}$ is initialized with $\mathcal{R}_{\mathcal{T}-1}$. The loss of JT is defined as:

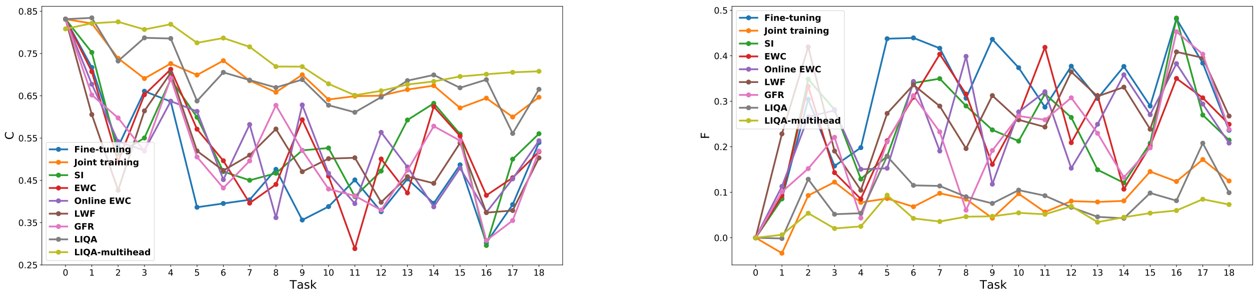
$$L_{\mathcal{T}}^{JT} = \mathbb{E}_{(\mathbf{x},s)\sim D_{t<=\mathcal{T}}}\left[\|\mathcal{R}_{\mathcal{T}}(\mathbf{x}) - s\|_2\right]. \tag{31}$$

It should be noted that JT can be regarded as an upper bound of lifelong learning methods which are not allowed to get access to previous training data $D_{t<\mathcal{T}}$.

D. Performance with respect to inner-dataset distortion shift

We first evaluate the performance of LIQA when faced with inner-dataset distortion shift on KADID-10K. We respectively set the incremental step Δ to 1, 2 and 6 and quantify the performance by computing the correlation index C and the forgetting index F of each task. The network is first trained on the base task which consists of 7 base distortion types and then sequentially trained on novel tasks.

When the incremental step is set to 1, we sequentially add 18 novel distortions following the permutation order of JP2K→CSA2→PIX→WN→CB→GB→DA→CC→BR→ NEP→ MS→ MB→ MN→ LB→DN→HS→ CD→ CS,



(a)

(b)

Fig. 4: Performance of inner-dataset distortion shift with incremental step set to 1. (a) Correlation index with respect to tasks. (b) Forgetting index with respect to tasks.

TABLE III: Performance comparison across various distortion types of KADID-10K at the last task. The best performance of each distortion among fine-tuning and the lifelong learning methods is highlighted in bold.

	CQ	JPEG	CSA1	WNCC	Q	JIT	IN	JP2K	CSA2	PIX	WN	CB	GB
FT	0.450	0.530	0.077	0.898	0.678	0.746	0.736	0.270	**0.802**	0.075	**0.838**	0.300	0.708
EWC	0.435	0.711	0.065	0.782	0.672	0.635	0.681	0.590	0.653	0.494	0.693	**0.380**	0.470
Online EWC	0.471	0.782	0.164	0.768	0.697	0.771	0.563	0.679	0.701	0.466	0.575	0.282	0.756
SI	0.235	0.794	0.060	0.710	0.658	0.789	0.454	0.741	0.654	0.433	0.629	0.318	**0.870**
LWF	0.231	0.415	0.046	0.800	0.590	0.379	0.807	0.360	0.697	0.044	0.697	0.120	0.850
GFR	0.416	0.591	0.012	0.834	0.594	0.680	0.791	0.335	0.745	0.108	0.751	0.231	0.789
LIQA	**0.622**	**0.825**	**0.403**	**0.903**	**0.827**	**0.864**	**0.879**	**0.855**	0.791	**0.627**	0.825	0.341	0.869
JT	0.515	0.814	0.378	0.825	0.398	0.886	0.864	0.851	0.731	0.680	0.796	0.221	0.900
	DA	CC	BR	NEP	MS	MB	MN	LB	DN	HS	CD	CS	Avg
FT	**0.546**	**0.136**	0.278	0.122	**0.402**	0.385	0.858	**0.814**	0.867	0.584	0.483	0.899	0.539
EWC	0.393	0.116	**0.597**	0.128	0.362	0.230	0.772	0.361	0.829	0.614	0.390	0.887	0.518
Online EWC	0.525	0.035	0.237	**0.236**	0.325	0.455	0.773	0.620	0.806	0.503	0.573	0.830	0.544
SI	0.516	0.115	0.458	0.162	0.366	**0.618**	0.770	0.764	0.797	0.561	**0.589**	**0.950**	0.560
LWF	0.327	0.102	0.270	0.161	0.300	0.512	0.776	0.854	0.836	0.187	0.431	0.850	0.466
GFR	0.492	0.051	0.340	0.264	0.318	0.363	0.869	0.790	0.900	0.275	0.560	0.896	0.519
LIQA	0.522	0.047	0.549	0.214	0.373	0.579	**0.917**	0.786	**0.949**	**0.783**	0.519	0.669	**0.665**
JT	0.357	0.123	0.711	0.283	0.047	0.900	0.807	0.854	0.858	0.844	0.728	0.797	0.647

leading to 18 novel tasks. The correlation index and the forgetting index with respect to tasks are shown in Fig. 4(a) and Fig. 4(b) respectively, from which we can see that both the correlation index and the forgetting index of fine-tuning are very unstable and the performances of previous tasks are easily influenced by the current task. Actually, setting the incremental step to 1 is hard for the network to continually learn new knowledge. It is because that the number of per distortion's training data is limited (only 325 images). Fine-tuning the network on such limited training data will cause the over-fitting problem and make the network easily be biased towards the current distortion. Intuitively, the network generalization ability will be impaired and the performances of the previously seen distortions will be influenced. When the current distortion distribution varies greatly from the previous distortions, the performance of previous distortions will drop drastically. When the current distortion distribution resembles some previous distortions', the performance of previous distortion may not undergo drastic changes. Taking the task#5 for example, when sequentially adding the training data of CB, the performance of the most distortions will be seriously impaired for fine-tuning. The correlation index of task#5 drops from 0.636 to 0.386 and the forgetting index increases from 0.198 to 0.437 compared with task#4. It is because that CB is a local distortion, whose distortion distribution varies greatly from that of the previously seen global distortions. Directly fine-tuning the network parameters on the training data of CB will make the network biased towards the currently learning distortion, leading to catastrophic forgetting of other distortion types. In contrast with fine-tuning, lifelong learning methods apparently mitigate the catastrophic forgetting phenomenon at task#5. For EWC, online EWC, SI, LWF, GFR and LIQA, the forgetting index is reduced to 0.213, 0.153, 0.179, 0.273, 0.211 and 0.179, respectively.

Taking a look at the whole incremental learning process, we can find that the performance of LIQA keeps stable while the performances of other lifelong learning methods will change drastically at certain task session. Taking task#11 (adding MS) for example, the correlation index of EWC, online EWC and SI even cannot catch up with that of fine-tuning. In contrast, LIQA which takes advantages of replaying pseudo features to consolidate the learned knowledge of previously learned distortions, can well resist the negative effect of certain task and obtain better robustness and stability when facing with inner-dataset distortion shift. Without access to the previous training data, LIQA can obtain comparable performance with joint training. Moreover, one thing that should be noted is that even if joint training utilizes training data of all seen task, adding new distortion type will also bring slight forgetting of previous distortion types, due to the conflict between different distortion types.

The SRCCs of each distortion type at the final task with incremental step set to 1 are listed in Table III, from which we can see that the performances of most distortions of LIQA outperform that of the other lifelong learning methods. By comparing LIQA with joint training, we can find that the performances of some distortions (i.e. CQ, JPEG, CSA1, WNCC, Q, IN, JP2K, CSA2, WN, CB, DA, MS, MN, DN) are better than that of joint training and the average SRCC of all seen distortions is also better than that of joint training. The performance of the final task illustrates that LIQA has better ability to preserve the previously learned knowledge (the performances of the 7 base distortions of LIQA are comparable with that of joint training). Besides, consolidating the learned knowledge does not interfere with the learning of new knowledge (the performances of the novel distortions are satisfactory).

When the incremental step is set to 2, we add two distortions per task following the order of (JP2K, CSA2)→(PIX, WN)→(CB, GB)→(DA, CC)→(BR, NEP)→(MS, MB)→ (MN,LB)→(DN, HS)→(CD, CS), leading to 9 novel tasks. The correlation index and the forgetting index with respect to tasks are shown in Fig. 5(a) and Fig. 5(b) respectively. By comparing Fig. 5(a) with Fig. 4(a), we can find that when each task contains 2 distortions, the catastrophic forgetting of fine-tuning becomes less serious. The worst correlation index of fine-tuning in Fig. 5(a) is 0.47 while the worst correlation index of fine-tuning in Fig. 4(a) is 0.30. The biggest forgetting index in Fig. 5(b) of fine-tuning decreases from 0.48 to 0.30 compared with Fig. 4(b). It is because that the negative effect of certain distortion will be weaken by another distortion. Taking the task#3 in Fig. 5(a) for example, combination of CB and GB will not seriously impair the performance of previous distortions. By comparing the forgetting index of all methods shown in Fig. 5(b), we can find that the forgetting of LIQA is low and stable, and can well preserve the learned knowledge while learning new tasks. In contrast, the forgetting indexes of EWC, online EWC, SI, LWF and GFR are unstable and sometimes very high (e.g. task#6 and task#8).

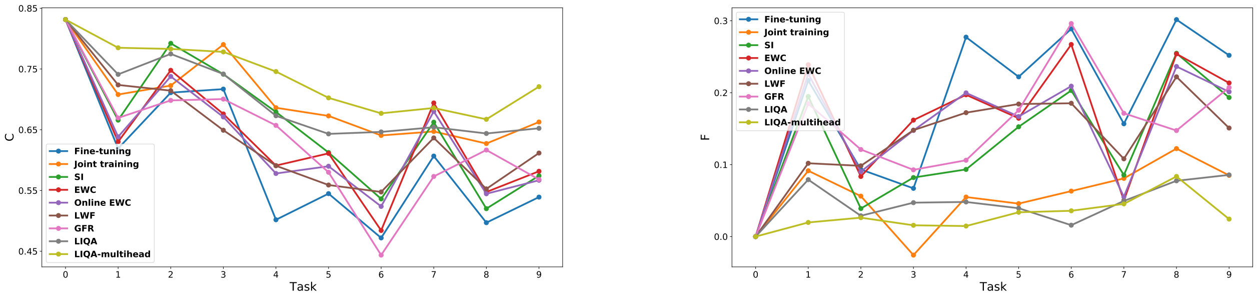
When the incremental step is set to 6, we add six distortions per task following the order of (JP2K, CSA2, PIX, WN, CB, GB)→(DA, CC, BR, NEP, MS, MB)→ (MN, LB, DN, HS, CD, CS), leading to 3 novel tasks. The correlation index and the forgetting index with respect to tasks are shown in Fig. 6(a) and Fig. 6(b) respectively. From Fig. 6(a) we can observe that the performance gap between fine-tuning and lifelong learning methods further shrinks. By comparing Fig. 5(b) and Fig. 6(a), we can find that the biggest forgetting index of fine-tuning at certain task session further decreases from 0.30 to 0.19. It is because that as the distortion types and the number of the training data increases, the network generalization ability is also improved. The catastrophic forgetting caused by certain distortion will be further suppressed.

To sum up, when the incremental step is set to 1, catastrophic problem will easily emerge. It is because the limited training data of certain distortion will make the network parameters over-fitted to the current task and destroy the generalization ability of the network. As the incremental step increases, the catastrophic forgetting of fine-tuning will be mitigated because the generalization ability is also improved. EWC, online EWC, SI, LWF and GFR can mitigate the catastrophic forgetting to some extent but the performance is unstable. By comparing Eq. 26 and Eq. 16, we can find the two optimization objectives are contradictory. Unlike classification tasks distilling the knowledge from multiple previous probability values separately, the quality regression network only has one scalar output value. Therefore, directly applying the idea of LWF to BIQA does not work. As for GFR, the way of generation conditioned on the discrete labels cannot produce accurate pseudo features. Besides, using the pseudo features paired with inaccurate hard pseudo labels further hinders its performance. In contrast, LIQA employs auxiliary multi-head regression network to generate soft pseudo labels. It can weaken the dependency to the accuracy of pseudo features and can strike a good balance between the stability and plasticity. Moreover, we should notice that the performance of the multi-head auxiliary network is already comparable even better than joint training, by only using the L2 loss. It is because that joint training adopts single-head network, which will cause the conflicts of different distortions. In contrast, the auxiliary multi-head network assigns specific head for each distortion type, avoiding the conflicts among distortions. LIQA distills knowledge from multi-head network and thus can achieve comparable performance compared with joint training.

E. Performance with respect to cross-dataset distortion shift

Inspired by the experimental results of UNIQUE [10], we can observe that directly linearly re-scale the subjective scores to a normalized range can generally obtain good performance when combining different datasets. Therefore, here we do not pay attention to designing a better algorithm to overcome the perceptual scale mismatch between different task. Instead, we adopt the linearly re-scaling strategy for simplicity and focus on the performance comparison with other lifelong learning methods under the same experimental setup.

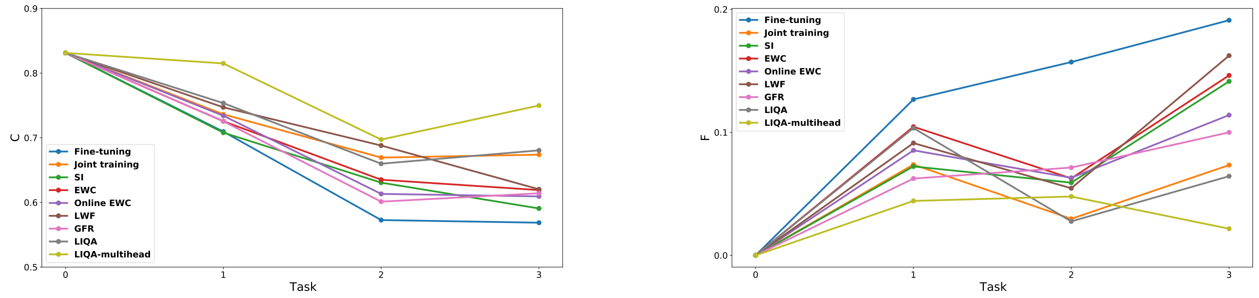
To evaluate the performance of LIQA when faced with cross-dataset distortion shift, we conduct experiments on six IQA datasets following the permutation order of LIVE→CSIQ→BID→CLIVE→KonIQ-10K→KADID-10K. We regard the LIVE dataset as the base dataset and the other five datasets as novel datasets. This setting covers datasets from both the same and different distortion scenarios. Considering that the authentic dataset cannot identify the distortion types, We regarded each dataset as a whole and trained LIQA with the incremental step to 1. The correlation index and the forgetting index are shown in Fig. 7(a) and Fig. 7(b) respectively. From Fig. 7(a) we can see that the shift from LIVE to CSIQ does not bring apparent forgetting for fine-tuning. It is because that LIVE and CSIQ both include synthetically distorted images and have 4 overlapped distortion types (JPEG-2000 compression, JPEG compression, white Gaussian noise and Gaussian blur). However, when adding BID dataset which includes authentically distorted images, the forgetting



(a)

(b)

Fig. 5: Performance of inner-dataset distortion shift with incremental step set to 2. (a) Correlation index with respect to task session. (b) Forgetting index with respect to task session.



(a)

(b)

Fig. 6: Performance of inner-dataset distortion shift with incremental step set to 6. (a) Correlation index with respect to tasks. (b) Forgetting index with respect to tasks.

indexes of fine-tuning and all the lifelong learning methods become apparent. It is known that the data distribution of BID dataset varies greatly from that of LIVE and CSIQ, and the apparent change makes the network biased towards the currently learning dataset, thus impairs the generalization ability towards other datasets. Similarly, we can find from Fig. 7(b) that the forgetting index increases during the shift from authentic dataset to synthetic dataset (i.e. task#5). Taking a look at the whole incremental learning process, we can find that LIQA obtains more stable performance compared with other lifelong learning methods and can effectively mitigate the forgetting in the face of the apparent cross-dataset distortion shift (e.g. LIQA reduces the forgetting at task#2 and task#5 compared with EWC, online EWC, SI, LWF and GFR). The performances of each dataset at the final task are listed in Table IV. From the table, we have several observations:

(1) LIQA aims to achieve the global optimal solution instead of the optimal performance of each dataset.

(2) EWC discourages the parameters important for the previous task from changing, which can achieve better performance when the two adjacent tasks are similar. It is because that when the next task is similar to the previous task (e.g. LIVE and CSIQ are both synthetic datasets and share 4 same distortions), it can utilizes the useful prior knowledge for training. However, when the two adjacent tasks are obviously different, EWC will fail (e.g. CSIQ is a synthetic dataset and BID is an authentic dataset).

(3) LWF uses the distillation loss Eq. 26 to distill the knowledge from the last single-head network. It uses the last network to generate the pseudo labels of the current task. Therefore, the last task's knowledge can be better preserved (KADID-10K is the finally learned dataset and KonIQ-10k is the last learned dataset). However, the performance of history tasks cannot be well preserved compared with the last task (the performance of the first learned dataset LIVE is the worst among all methods). In contrast, LIQA uses pesudo features of all previously learned datasets for distilling knowledge.

(4) For FT, it uses images from KADID-10K for training without considering the stability of previously learned datasets, thus overfitting to the finally learned dataset. For lifelong learning methods, guaranteeing the stability of previous knowledge will impair the plasticity slightly, so the performance of KADID-10K will be lowered to some extent.

Specially, we should note that the addition of KADID-10K impairs the performance of previous datasets even if we adopt joint training. It is because that the image number of KADID-10K is far larger than most of the datasets, the unbalanced data distribution of different datasets impairs the optimal performance of each dataset. To verify the conjecture, we equip the joint training with pesudo replay, which generates equal number of pseudo features for each dataset. By comparing the results of “JT” and “JT+PR”, we can find that the performance of each dataset is improved by utilizing pseudo features. It further verifies that pseudo features can address the unbalanced problem to some extent by controlling the quality range and the dataset.

TABLE IV: Performance comparison across various datasets at the last task session. The best performance of each dataset among fine-tuning and the lifelong learning methods is highlighted in bold. JT+PR denotes joint training equipped with pseudo features replaying.

	LIVE	CSIQ	BID	CLIVE	KonIQ-10K	KADID-10K	avg
FT	0.819	0.663	0.398	0.298	0.573	**0.942**	0.615
EWC	0.829	**0.760**	0.501	0.472	0.658	0.916	0.689
Online EWC	0.813	0.718	0.514	0.460	0.659	0.918	0.680
SI	0.820	0.725	0.580	0.453	0.699	0.900	0.696
LWF	0.784	0.721	0.635	0.497	**0.730**	0.905	0.712
GFR	0.799	0.702	0.484	0.301	0.652	0.941	0.647
LIQA	**0.844**	0.705	**0.642**	**0.572**	0.713	0.898	**0.729**
JT	0.886	0.937	0.663	0.581	0.778	0.921	0.794
JT+PR	0.929	0.962	0.690	0.735	0.808	0.937	0.844

F. Analysis and discussions

We first conduct ablation study to verify the effectiveness of each key component of LIQA. Then we further discuss the memory replay and the robustness to task permutations. In this part, we follow the experimental setting in Section V-D with the incremental step set to 1 regarding the inner-dataset distortion shift, and follow the experimental setting in Section V-E regarding the cross-dataset distortion shift.

1) Ablation study: We conduct experiments to verify the effectiveness of the Split-and-Merge distillation strategy as well as the feature distillation loss and the pseudo replay loss in Eq. 7. Specially, to verify the effectiveness of the Split-and-Merge distillation strategy, we replace the L_{PR} defined in Eq. 4 with:

$$L^{PR} = \mathbb{E}_{z\sim\mathcal{N}(\mathbf{0},\mathbf{1}),s\sim\bar{\mathcal{S}},j\sim p_{j_{t<\mathcal{T}}}}\left[\left\|V_{\mathcal{T}-1}(\tilde{h}_j) - V_{\mathcal{T}}(\tilde{h}_j)\right\|_2\right], \tag{32}$$

where $V_{\mathcal{T}-1}$ denotes the single-head regression network trained at the former task.

We implement three variants of LIQA and average the correlation indexes and the forgetting indexes over all tasks to represent the overall performance during the whole incremental learning process, which are denoted as $\bar{C}$ and $\bar{F}$ respectively. The comparison results of the three variants of LIQA as well as LIQA under inner-dataset distortion shift are shown in Table V. From the table we can see that pseudo replay plays a significant role in LIQA. Besides, without the Split-and-Merge strategy, $\bar{C}$ drops and $\bar{F}$ increases. It is due to that the pseudo labels given by the former single-head network are inaccurate when the former task has negative effect on most of the learned distortions (e.g. task#5). Directly distilling knowledge from the inaccurate single-head network will lead to the error-propagation problem thus hindering the consolidation of the previous distortions. In contrast, LIQA preserves the response of each previously learned distortion by the auxiliary multi-head regression network and resists the error-propagation problem. The feature distillation loss constrains that the features extracted by the current feature extractor $V_{\mathcal{T}}$ do not shift away from that by $V_{\mathcal{T}-1}$. The generator replays the pseudo features that resemble the data distribution of the features generated by $V_{\mathcal{T}-1}$ and the regression head is trained with the pseudo features. The drastic change of the features will lead to the mismatch between the real features of the previous distortions and the regression head trained with the pseudo features when testing. As shown in Table V, the $\bar{C}$ of LIQA w/o FD drops and $\bar{F}$ increases compared with LIQA.

TABLE V: Average correlation index and average forgetting index of different variants of LIQA. FD represents feature distillation and PR represents pseudo replay.

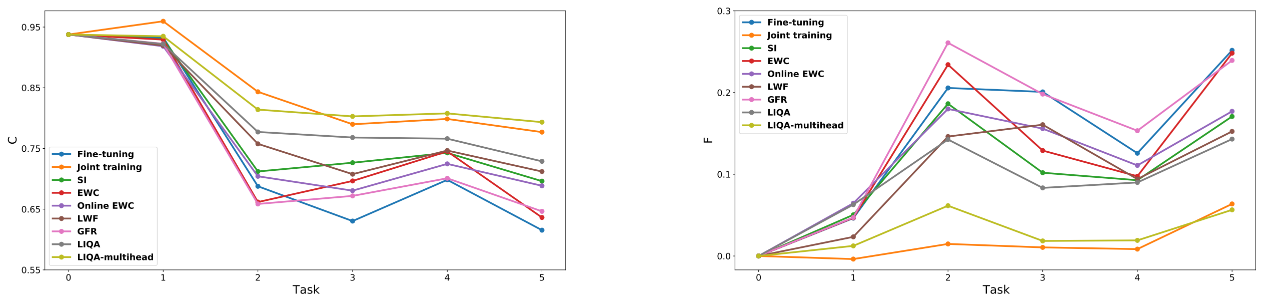
	w/o Split-and-Merge	w/o FD	w/o PR	LIQA
$\bar{\mathbf{C}}$	0.624	0.658	0.566	0.695
$\bar{\mathbf{F}}$	0.117	0.089	0.214	0.087

2) Memory replay: We further explore the memory replay strategy and the size of the replay buffer per batch under inner-dataset distortion shift. The average correlation index and the average forgetting index across tasks are shown in Table VI. Suppose that the allocated replay buffer size is denoted as N_{buf} per batch, and the number of the previously seen distortions is denoted as M_{pre}. The re-scaled quality score is within the range of [0, 1]. We split the re-scaled quality range into five interval segments, i.e. [0, 0.2), [0.2, 0.4), [0.4, 0.6), [0.6, 0.8), [0.8, 1.0].

TABLE VI: Average correlation index and average forgetting index for different replay strategies and different replay buffer sizes.

	Replay strategy				Replay buffer size			
	Random	Qua	Dist	Qua &Dist	350	700	1400	2800
$\bar{\mathbf{C}}$	0.635	0.689	0.646	0.695	0.651	0.687	0.695	0.697
$\bar{\mathbf{F}}$	0.112	0.072	0.075	0.087	0.092	0.076	0.087	0.081

We design four variants of replay strategy according to the two conditions of the generator: the quality score and the distortion type. The first variant is the random replay, which means that the pseudo features are generated given the random quality score chosen from the range of [0, 1] and the random distortion type chosen from the previously learned distortions (corresponding to $Random$ in Table VI). The second variant is to control the quality score and allocate $N_{buf}/5$ pseudo features to each of the five interval segments (corresponding to Qua in Table VI). The distortion type is randomly chosen from the previously seen distortions. The third variant is to control the distortion type and allocate N_{buf}/M_{pre} pseudo features to each of the seen distortions (corresponding to $Dist$ in Table VI). The quality score is randomly chosen from the range of [0, 1]. The fourth variant is the method that LIQA adopts (corresponding to $Qua\&Dist$ in Table VI). We allocate $N_{buf}/M_{pre}/5$ pseudo features to each seen distortion each quality interval segment. From Table VI we can see that the performance of $Qua\&Dist$ outperforms that of $Random$, Qua and $Dist$. It demonstrates that by well



(a)

(b)

Fig. 7: Performance of cross-dataset distortion shift with incremental step set to 1. (a) Correlation index with respect to tasks. (b) Forgetting index with respect to tasks.

controlling the distortion type and the quality range of the generated pseudo features, LIQA can avoid the unbalanced distribution and enhance the consolidation of the knowledge of learned distortions.

Moreover, we explore the effect of the replay buffer size per batch N_{buf}. As shown in Table VI, as the N_{buf} increases, the performance improves. Considering the balance between the performance and the complexity, we set N_{buf} to 1400 in LIQA.

3) Robustness to task permutations: To verify the robustness to different task permutations of LIQA, we conduct experiments of different task permutation orders under inner-dataset distortion shift and cross-dataset distortion shift respectively. For the inner-dataset distortion shift, we randomly permute the order of the 18 novel distortions and generate 5 different task orders.

- **Order1**: JP2K→CSA2→PIX→WN→CB→GB→DA→CC→BR→NEP→MS→MB→MN→LB→DN→HS→CD→CS
- **Order2**: PIX→MS→CD→CS→HS→MB→CSA2→WN→DN→BR→NEP→MN→CB→DA→CC→LB→GB→JP2K
- **Order3**: MB→DA→CD→MN→JP2K→GB→CC→BR→HS→DN→NEP→CS→CB→MS→LB→PIX→WN→CSA2
- **Order4**: CD→CC→HS→DN→BR→CSA2→LB→MS→JP2K→CB→GB→DA→MB→CS→NEP→WN→MN→PIX
- **Order5**: WN→CSA2→DA→NEP→MN→LB→PIX→GB→CS→CD→JP2K→CB→DN→HS→BR→MB→CC→MS

For the cross-dataset distortion shift, we randomly permute the order of the 5 novel datasets and generate 5 different task orders:

- **Order1**: CSIQ→BID→CLIVE→KonIQ-10k→KADID-10K
- **Order2**: CLIVE→CSIQ→KADID-10K→BID→KonIQ-10K
- **Order3**: KADID-10K→BID→CSIQ→KonIQ-10K→CLIVE
- **Order4**: KonIQ-10K→KADID-10K→CLIVE→CSIQ→BID
- **Order5**: CSIQ→BID→KADID-10K→CLIVE→KonIQ-10K

We compute the average correlation index and the average forgetting index across all tasks. The results are shown in Table VII and Table VIII. $Order1$ denotes the default order in Section V-D and Section V-E. From Table VII and Table VIII, we can see that the performances of different task permutations vary slightly, which illustrates that LIQA has good robustness to task permutations, benefiting from the Split-and-Merge distillation strategy which resists the negative effect of certain task during the whole incremental learning process.

TABLE VII: Average correlation index and average forgetting index for different task permutations under inner-dataset distortion shift

	Order1(default)	Order2	Order3	Order4	Order5
. $\bar{\mathbf{C}}$	0.695	0.674	0.698	0.694	0.710
$\bar{\mathbf{F}}$	0.087	0.070	0.082	0.113	0.086

TABLE VIII: Average correlation index and average forgetting index for different task permutations under cross-dataset distortion shift.

	Order1(default)	Order2	Order3	Order4	Order5
$\bar{\mathbf{C}}$	0.817	0.791	0.813	0.799	0.812
$\bar{\mathbf{F}}$	0.0867	0.123	0.104	0.118	0.102

4) Comparison with state-of-the-art BIQA methods: We compare LIQA with three SOTA BIQA methods: HyperIQA [59], DBCNN [60] and MEON [61]. HyperIQA adopts hyper network to estimate the image quality in a self-adaptive manner. DBCNN adopts bilinear pooling to fuse features extracted from networks trained on image classification task and distortion type classification task, which works for both synthetically and authentically distorted images. MEON consists of two subnetworks, *i.e.,* a distortion identification network and a quality prediction network, sharing the early layers. In contrast, LIQA simply adopts the pre-trained ResNet-18. For each task, we mix the images from previously learned tasks and the current task for training and report the average correlation index as well as the average forgetting index across all tasks. The results under inner-dataset distortion shift and cross-dataset distortion shift are shown in Table IX. From the results, we can find that LIQA outperforms MEON and HyperIQA under inner-dataset distortion shift and achieves the best performance under cross-dataset distortion shift when utilizing images from all seen tasks. It is because that the pseudo replay strategy serves as a strong data augmentation strategy while training, increasing the diversity of training samples. Specially, there exist severe data unbalance problem under the cross-dataset distortion shift. For example, the total number of images from KonIQ-10K is more than 10,000 while that from BID is less than 600. LIQA mitigates the data unbalance problem by generating pseudo features.

TABLE IX: Average correlation index and average forgetting index of different BIQA algorithms.

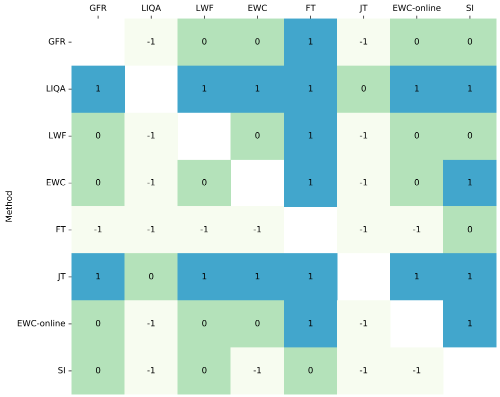
$\bar{C}/\bar{F}$ \ *Methods*	MEON [61]	DBCNN [60]	HyperIQA [59]	LIQA
inner-dataset	0.731/0.071	0.774/0.046	0.653/0.082	0.765/0.058
cross-dataset	0.868/-0.003	0.833/0.029	0.825/0.016	0.872/-0.001

TABLE X: Mean and std values of average correlation index and average forgetting index across five runs.

		FT	EWC	Online EWC	SI	LWF	GFR	JT	LIQA
$\bar{C}$	mean↑	0.489	0.541	0.538	0.486	0.512	0.539	0.673	0.695
	std↓	0.012	0.022	0.013	0.020	0.008	0.022	0.011	0.009
$\bar{F}$	mean↓	0.302	0.232	0.253	0.304	0.271	0.235	0.084	0.087
	std↓	0.009	0.018	0.017	0.009	0.005	0.019	0.008	0.009

5) Statistical analysis: We compute the mean and std values of average correlation index $\bar{C}$ as well as average forgetting index $\bar{F}$ of each method (under inner-dataset distortion shift with incremental step set to 1) across five runs. The results are shown in in Table X. For $\bar{C}$, the higher mean value represents better performance of the continual learner while lower std value represents more stable performance. For $\bar{F}$, the lower mean value means less forgetting of previous knowledge during the incremental learning process while lower std value means more stable performance. By comparing the mean values, we can see that LIQA can achieve the best performance among all the compared methods and can effectively mitigate the forgetting of previously learned knowledge. By comparing the std values, we can see that the performance of LIQA is stable.

We further use a hypothesis testing approach based on t-statistics [62] to demonstrate the superiority of LIQA. In our experiment, the two-sample t-test between the pair of $\bar{C}$ values at the 5% significance level is conducted. Fig. 8 shows the results of t-test, where the value 1/0/-1 indicates that row methods perform statistically better/comparably/worse than the column methods. From the results, we can see that LIQA outperforms FT as well as other lifelong learning methods, and performs comparably compared with JT without access to previous training data.



	GFR	LIQA	LWF	EWC	FT	JT	EWC-online	SI
GFR		-1	0	0	1	-1	0	0
LIQA	1		1	1	1	0	1	1
LWF	0	-1		0	1	-1	0	0
EWC	0	-1	0		1	-1	0	1
FT	-1	-1	-1	-1		-1	-1	0
JT	1	0	1	1	1		1	1
EWC-online	0	-1	0	0	1	-1		1
SI	0	-1	0	-1	0	-1	-1	

Fig. 8: Significance t-test results under inner-dataset distortion shift with incremental step set to 1.

VI. Conclusion

We propose a new LIQA framework to achieve the lifelong learning of BIQA. The proposed LIQA employs a generator conditioned on the distortion type and the quality score to generate pseudo features, which serves as a memory replayer when learning new tasks. In order to resist the negative effect of certain task during the whole incremental learning process, we employ an auxiliary multi-head regression network to generate predicted quality score of each seen distortion type. It avoids the conflicts between different distortion types and thus improve the robustness to task permutations. Extensive experiments verify that LIQA can effectively mitigate the catastrophic forgetting when facing with inner-dataset distortion shift and cross-dataset distortion shift during the sequential learning process.

In this paper, LIQA focuses on the close-set experimental setting where we can specify the novel distortions during sequential learning process. Also, LIQA can be extended to open-set experimental setting which we will explore in the future. Specifically, for the real-world collected images, we can first automatically discover the out-of-distribution images that the current model cannot handle. Then we cluster the images into several distortion types. Next we incrementally learn the found novel distortions.

References

[1] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, vol. 113, pp. 54–71, 2019.

[2] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of Learning and Motivation*. Elsevier, 1989, vol. 24, pp. 109–165.

[3] B. Ans, S. Rousset, R. M. French, and S. Musca, "Self-refreshing memory in artificial neural networks: Learning temporal sequences without catastrophic forgetting," *Connection Science*, vol. 16, no. 2, pp. 71–99, 2004.

[4] R. M. French, "Dynamically constraining connectionist networks to produce distributed, orthogonal representations to reduce catastrophic interference," in *Proceedings of the sixteenth annual conference of the cognitive science society*. Routledge, 2019, pp. 335–340.

[5] 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.

[6] R. Ratcliff, "Connectionist models of recognition memory: constraints imposed by learning and forgetting functions." *Psychological review*, vol. 97, no. 2, p. 285, 1990.

[7] J. Kim and S. Lee, "Deep learning of human visual sensitivity in image quality assessment framework," in *CVPR*, 2017, pp. 1676–1684.

[8] G. Zhai and X. Min, "Perceptual image quality assessment: a survey," *Science China Information Sciences*, vol. 63, no. 11, p. 211301, 2020.

[9] W. Zhang, K. Zhai, G. Zhai, and X. Yang, "Learning to blindly assess image quality in the laboratory and wild," in *ICIP*. IEEE, 2020, pp. 111–115.

[10] W. Zhang, K. Ma, G. Zhai, and X. Yang, "Uncertainty-aware blind image quality assessment in the laboratory and wild," *IEEE Transactions on Image Processing*, vol. 30, pp. 3474–3486, 2021.

[11] X. Liu, C. Wu, M. Menta, L. Herranz, B. Raducanu, A. D. Bagdanov, S. Jui, and J. v. de Weijer, "Generative feature replay for class-incremental learning," in *CVPR Workshops*, 2020, pp. 226–227.

[12] H. Lin, V. Hosu, and D. Saupe, "Kadid-10k: A large-scale artificially distorted iqa database," in *International Conference on Quality of Multimedia Experience*. IEEE, 2019, pp. 1–3.

[13] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3440–3451, 2006.

[14] E. C. Larson and D. M. Chandler, "Most apparent distortion: full-reference image quality assessment and the role of strategy," *Journal of Electronic Imaging*, vol. 19, no. 1, p. 011006, 2010.

[15] A. Ciancio, E. A. da Silva, A. Said, R. Samadani, P. Obrador *et al.*, "No-reference blur assessment of digital pictures based on multifeature classifiers," *IEEE Transactions on Image Processing*, vol. 20, no. 1, pp. 64–75, 2010.

[16] D. Ghadiyaram and A. C. Bovik, "Massive online crowdsourced study of subjective and objective picture quality," *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 372–387, 2015.

[17] V. Hosu, H. Lin, T. Sziranyi, and D. Saupe, "Koniq-10k: An ecologically valid database for deep learning of blind image quality assessment," *IEEE Transactions on Image Processing*, vol. 29, pp. 4041–4056, 2020.

[18] Z. Chen, J. Lin, N. Liao, and C. W. Chen, "Full reference quality assessment for image retargeting based on natural scene statistics modeling and bi-directional saliency similarity," *IEEE Transactions on Image Processing*, vol. 26, no. 11, pp. 5138–5148, 2017.

[19] Z. Chen, J. Xu, C. Lin, and W. Zhou, "Stereoscopic omnidirectional image quality assessment based on predictive coding theory," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 1, pp. 103–117, 2020.

[20] J. Liu, X. Li, Y. Peng, T. Yu, and Z. Chen, "Swiniqa: Learned swin distance for compressed image quality assessment," *arXiv preprint arXiv:2205.04264*, 2022.

[21] S. Xu, S. Jiang, and W. Min, "No-reference/blind image quality assessment: a survey," *IETE Technical Review*, vol. 34, no. 3, pp. 223–245, 2017.

[22] D. Li, T. Jiang, W. Lin, and M. Jiang, "Which has better visual quality: The clear blue sky or a blurry animal?" *IEEE Transactions on Multimedia*, vol. 21, no. 5, pp. 1221–1234, 2018.

[23] X. Min, G. Zhai, K. Gu, Y. Zhu, J. Zhou, G. Guo, X. Yang, X. Guan, and W. Zhang, "Quality evaluation of image dehazing methods using synthetic hazy images," *IEEE Transactions on Multimedia*, vol. 21, no. 9, pp. 2319–2333, 2019.

[24] W. Zhou, Q. Jiang, Y. Wang, Z. Chen, and W. Li, "Blind quality assessment for image superresolution using deep two-stream convolutional networks," *Information Sciences*, vol. 528, pp. 205–218, 2020.

[25] W. Zhou, Z. Wang, and Z. Chen, "Image super-resolution quality assessment: Structural fidelity versus statistical naturalness," *arXiv preprint arXiv:2105.07139*, 2021.

[26] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4695–4708, 2012.

[27] C. Yang, X. Zhang, P. An, L. Shen, and C.-C. J. Kuo, "Blind image quality assessment based on multi-scale klt," *IEEE Transactions on Multimedia*, 2020.

[28] Q. Li, W. Lin, J. Xu, and Y. Fang, "Blind image quality assessment using statistical structural and luminance features," *IEEE Transactions on Multimedia*, vol. 18, no. 12, pp. 2457–2469, 2016.

[29] P. G. Freitas, W. Y. Akamine, and M. C. Farias, "No-reference image quality assessment using orthogonal color planes patterns," *IEEE Transactions on Multimedia*, vol. 20, no. 12, pp. 3353–3360, 2018.

[30] W. Zhou, Z. Chen, and W. Li, "Dual-stream interactive networks for no-reference stereoscopic image quality assessment," *IEEE Transactions on Image Processing*, vol. 28, no. 8, pp. 3946–3958, 2019.

[31] J. Xu, W. Zhou, and Z. Chen, "Blind omnidirectional image quality assessment with viewport oriented graph convolutional networks," *IEEE Transactions on Circuits and Systems for Video Technology*, 2020.

[32] W. Zhang, K. Ma, J. Yan, D. Deng, and Z. Wang, "Blind image quality assessment using a deep bilinear convolutional neural network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 1, pp. 36–47, 2018.

[33] F. Gao, D. Tao, X. Gao, and X. Li, "Learning to rank for blind image quality assessment," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 10, pp. 2275–2290, 2015.

[34] S. T. Grossberg, *Studies of mind and brain: Neural principles of learning, perception, development, cognition, and motor control*. Springer Science & Business Media, 2012, vol. 70.

[35] M. Delange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars, "A continual learning survey: Defying forgetting in classification tasks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[36] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[37] J. Schwarz, W. Czarnecki, J. Luketina, A. Grabska-Barwinska, Y. W. Teh, R. Pascanu, and R. Hadsell, "Progress & compress: A scalable framework for continual learning," in *ICML*. PMLR, 2018, pp. 4528–4537.

[38] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," in *ICML*. PMLR, 2017, pp. 3987–3995.

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

[40] H. Jung, J. Ju, M. Jung, and J. Kim, "Less-forgetting learning in deep neural networks," *arXiv preprint arXiv:1607.00122*, 2016.

[41] J. Zhang, J. Zhang, S. Ghosh, D. Li, S. Tasci, L. Heck, H. Zhang, and C.-C. J. Kuo, "Class-incremental learning via deep model consolidation," in *WACV*, 2020, pp. 1131–1140.

[42] 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.

[43] A. Rosenfeld and J. K. Tsotsos, "Incremental learning through deep adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 3, pp. 651–663, 2018.

[44] J. Serra, D. Suris, M. Miron, and A. Karatzoglou, "Overcoming catastrophic forgetting with hard attention to the task," in *ICML*. PMLR, 2018, pp. 4548–4557.

[45] A. Mallya and S. Lazebnik, "Packnet: Adding multiple tasks to a single network by iterative pruning," in *CVPR*, 2018, pp. 7765–7773.

[46] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "icarl: Incremental classifier and representation learning," in *CVPR*, 2017, pp. 2001–2010.

[47] D. Rolnick, A. Ahuja, J. Schwarz, T. P. Lillicrap, and G. Wayne, "Experience replay for continual learning," *arXiv preprint arXiv:1811.11682*, 2018.

[48] H. Shin, J. K. Lee, J. Kim, and J. Kim, "Continual learning with deep generative replay," *arXiv preprint arXiv:1705.08690*, 2017.

[49] C. Atkinson, B. McCane, L. Szymanski, and A. Robins, "Pseudo-recursal: Solving the catastrophic forgetting problem in deep neural networks. arxiv 2018," *arXiv preprint arXiv:1802.03875*.

[50] G. M. van de Ven, H. T. Siegelmann, and A. S. Tolias, "Brain-inspired replay for continual learning with artificial neural networks," *Nature communications*, vol. 11, no. 1, pp. 1–14, 2020.

[51] J. Liu, J. Lin, X. Li, W. Zhou, S. Liu, and Z. Chen, "Lira: Lifelong image restoration from unknown blended distortions," in *ECCV*. Springer, 2020, pp. 616–632.

[52] M. Zhou, J. Xiao, Y. Chang, X. Fu, A. Liu, J. Pan, and Z.-J. Zha, "Image de-raining via continual learning," in *CVPR*, 2021, pp. 4907–4916.

[53] W. Zhang, D. Li, C. Ma, G. Zhai, X. Yang, and K. Ma, "Continual learning for blind image quality assessment," *arXiv preprint arXiv:2102.09717*, 2021.

[54] Z. Mai, R. Li, J. Jeong, D. Quispe, H. Kim, and S. Sanner, "Online continual learning in image classification: An empirical survey," *Neurocomputing*, vol. 469, pp. 28–51, 2022.

[55] A. Chaudhry, P. K. Dokania, T. Ajanthan, and P. H. Torr, "Riemannian walk for incremental learning: Understanding forgetting and intransigence," in *ECCV*, 2018, pp. 532–547.

[56] N. Díaz-Rodríguez, V. Lomonaco, D. Filliat, and D. Maltoni, "Don't forget, there is more than forgetting: new metrics for continual learning," *arXiv preprint arXiv:1810.13166*, 2018.

[57] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[58] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li, "Yfcc100m: The new data in multimedia research," *Communications of the ACM*, vol. 59, no. 2, pp. 64–73, 2016.

[59] S. Su, Q. Yan, Y. Zhu, C. Zhang, X. Ge, J. Sun, and Y. Zhang, "Blindly assess image quality in the wild guided by a self-adaptive hyper network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3667–3676.

[60] W. Zhang, K. Ma, J. Yan, D. Deng, and Z. Wang, "Blind image quality assessment using a deep bilinear convolutional neural network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 1, pp. 36–47, 2018.

[61] K. Ma, W. Liu, K. Zhang, Z. Duanmu, Z. Wang, and W. Zuo, "End-to-end blind image quality assessment using deep neural networks," *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1202–1213, 2017.

[62] D. C. Montgomery and G. C. Runger, *Applied statistics and probability for engineers*. John Wiley & Sons, 2010.