Incremental Image De-raining via Associative Memory

Yi Gu[1], Chao Wang[2], Jie Li[2]

[1] Alibaba Cloud Computing Ltd.
[2] Department of Computer Science and Engineering, Shanghai Jiao Tong University
luoyi.gy@alibaba-inc.com, lijiecs@sjtu.edu.cn

Abstract

While deep learning models have achieved the state-of-the-art performance on single-image rain removal, most methods only consider learning fixed mapping rules on the single synthetic dataset for lifetime. This limits the real-life application as iterative optimization may change mapping rules and training samples. However, when models learn a sequence of datasets in multiple incremental steps, they are susceptible to catastrophic forgetting that adapts to new incremental episodes while failing to preserve previously acquired mapping rules. In this paper, we argue the importance of sample diversity in the episodes on the iterative optimization, and propose a novel memory management method, Associative Memory, to achieve incremental image de-raining. It bridges connections between current and past episodes for feature reconstruction by sampling domain mappings of past learning steps, and guides the learning to trace the current pathway back to the historical environment without storing extra data. Experiments demonstrate that our method can achieve better performance than existing approaches on both inhomogeneous and incremental datasets within the spectrum of highly compact systems.

Introduction

Single image de-raining is a fundamental computer vision problem which aims at removing undesirable rain-polluted artifacts for better image quality. It serves as the basis for other downstream applications such as tracking, detection and segmentation (Kang, Lin, and Fu 2011; Kim et al. 2013; Luo, Xu, and Ji 2015). Current state-of-the-art models (Li et al. 2018a; Ren et al. 2019; Wang et al. 2019a; Yang and Lu 2019; Zhang, Sindagi, and Patel 2019) are typically based on the deep neural network due to its superior performance. However, this success is conditioned on the auspicious setup, where all types of mapping rules are learned at once and known for lifetime. This setting is quite limited for real world applications.

In a more common scenario, the de-raining model is trained on a single dataset with multiple mapping rules. Whenever a new batch of datasets are collected and fed into the model for iteration, the model has to be retrained on the new dataset together with the old ones, which is undoubtedly time-consuming and computationally expensive. Therefore, the model should effectively and efficiently learn a sequence of datasets for iteration. Unfortunately, when datasets are sequentially and separately fed into the model for training, the model inevitably encounters the catastrophic forgetting (McCloskey and Cohen 1989). The network constantly forgets the knowledge obtained from previous tasks whilst learning new training samples. It results in an arbitrary degradation in the model performance on historical learned mapping rules.

Catastrophic forgetting has been extensively studied on the image classification task (Li and Hoiem 2017; Xiang et al. 2019; Lee et al. 2020; Wu et al. 2021), while has been tackled very recently in the image de-raining field (Zhou et al. 2021). Due to the limitation of edge devices such as mobile phones, current approach for incremental de-raining focuses on exerting penalties on the weight modification motivated by previous efforts on classification tasks. However, only constraining parameters is not strong enough to maintain acquired knowledge, leading to too much plasticity (Douillard et al. 2021). Different from existing strategies in image classification and de-raining, we do not solely or mainly rely on the parameter importance. While it remains unexplored in image de-raining, providing additional data to augment the episodic memory may be considered. However, because of constrained space overhead and training time, this data-based approach is not practical and suitable for the deployment of de-raining algorithms on compact systems.

In this paper, we investigate the incremental rain removal for multiple datasets that is suitable for compact devices. We argue the role of sample diversity in an episodic memory which implicitly affects feature representation in the learning process, and propose a new memory management scheme named Associative Memory (AM) to achieve incremental rain removal. We note that human associative memory does not rely on mechanically memorizing data, but tends to incorporate the acquired experience and summarize the correlation between two successively happening events. This inspires us to build connections between distinct data distributions. We propose to strengthen samples connecting tasks that fire synchronously. AM maintains a mapping memory capturing mappings between domains. When adapting to new tasks, model learns to perform the task by reusing inverse mappings which traces the current pathway back to the historical data distribution to augment the sample diversity of the memory. Considering that only observing the single side of reconstruction to past domains may exist connected neurons with unrelated firing flow, we use a parameter isolation strategy to impose past memory consolidation on new domains. Our associative memory management imitates human cognition process, which associates the representation to updated feature space without memorizing specific data.

Contributions in this work include: 1) We investigate the sample diversity in episodes for incremental de-raining, and introduce a common scheme for different experimental protocols. 2) We propose a memory management strategy that heuristically associates the current pathway with the historical representation. 3) We explore the latent synaptic transmission and provide a parameter isolation mechanism for a complementary feature representation. 4) Extensive experiments on standard benchmarks demonstrate the superior performance of our proposed method under incremental de-raining setting.

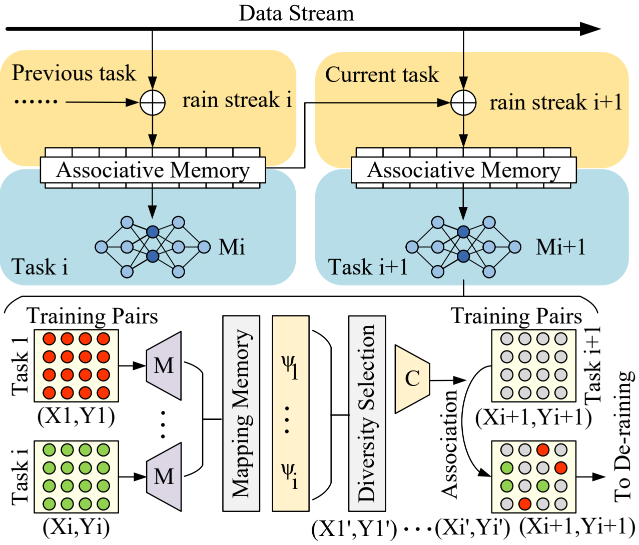

Figure 1: Associative Memory captures domain mappings from training pairs of each task and formulate them into a memory structure. Then it selects mappings from the memory to reconstruct features and remaps them to past domains. Associated features will participate in the basic model training along with original ones.

Related Work

Single Image De-raining

Image de-raining aims to recover a rain-free background layer from an image degraded by rain streaks and rain accumulation. It is a challenging work because of its ill-posed nature. Besides, the unavailability of temporal information, which could be seen as additional constraints, also brings challenges to solve image de-raining tasks. Therefore, different kinds of prior knowledge are applied into the optimization framework to generate optimal solutions to this problem. Typical methods of image de-raining are model-based approach, which are driven by image decomposition (Kang, Lin, and Fu 2012), sparse coding (Luo, Xu, and Ji 2015; Zhu et al. 2017), and priors based Gaussian mixture models (Li et al. 2016). These methods can only remove small and medium scales rain streaks effectively. Recently, image deraining methods have entered an era of deep learning. (Fu et al. 2017b,a) first proposes to remove rain streaks with a deep detail network (DetailNet). The network is able to take high frequency details as input and predict the residue of rain and clean images. Following this theory, many CNN based methods (Li et al. 2018b,c; Zhang and Patel 2018; Pan et al. 2018) are proposed. These methods apply more advanced network architectures and associate new related priors. They achieve better results both quantitatively and qualitatively. However, due to the limitation of the fully supervised learning paradigm, they tend to fail when dealing with some conditions of increment rain streaks that are sequentially added to the training process.

Incremental Learning for De-raining

Recently, increasing attention has been paid to the incremental image de-raining. (Zhou et al. 2021) first introduces this problem and tackle it with parameter importance that guides weights modification (PIGWM). It forces the weight of model to be similar to the optimal one of past tasks during gradient descent training and sets up a state-of-the-art for this task. Though this technique shows promising results on some conditions and can be successfully applied to the edge device, it still suffers from plasticity. Different from current work which relies on the regularization on optimization space to improve incremental de-raining, we argue the importance of sample diversity in episodes from another direction, and introduce a memory management method that utilizes sample diversity to achieve incremental de-raining without much memory overhead.

Associative Memory

Heuristics Mechanism

We suppose a sequence of N tasks to be learned in order, $T = \{T_1, ..., T_N\}$. Each task T_i is given a dataset of N_i paired instances, $T_i = \{\mathcal{X}_{i,j}, \mathcal{Y}_{i,j}\}_{j=1}^{N_i}$, where $\mathcal{X}_i$ and $\mathcal{Y}_i$ denote the original domain and ground truth domain respectively. In training session, the *i th* model H_i learns the current task T_i and aims to optimize the neural network to replicate T_i's real data distribution $\mathbb{P}_i$. Besides, H_i is required to generate rain removal results for historical rain streak tasks $T_1, ..., T_{i-1}$. However, the model has no access to the previous training data and can only use current data $\{\mathcal{X}_i, \mathcal{Y}_i\}$.

Fig 1 illustrates the heuristics mechanism. The target model can be formulated as the interaction of guiding the association to past environments and allowing the learning of new episodes. We propose to preserve the initial mapping between target and source domains which captures concepts when encountering new domains. When the basic model is trained for the *i-1 th* task T_{i-1}, a mapping agent M learns to memorize the cumulative input space and record the reconstruction mapping from source domain to target domain $\psi_{i-1}(\cdot)$. The agent M is usually a simple module, such as two convolution and deconvolution layers. We assume the memory buffer $\mathcal{M}$ contains mappings, targets and predictions. For the *i th* task, the model obtains new training paired samples $\{\mathcal{X}_i, \mathcal{Y}_i\}$.

A signal selector C chooses which set of inverse mappings from the mapping memory will proceed. This mapping ψ_i^{-1} reconstructs the current input space into the historical feature representation. We formalize this selection problem for sample diversity from the perspective of information theory. Let $\mathcal{B}$ be the observed batch of one iteration, and the memory update from the selector can be expressed as:

$$(\theta_i, \mathcal{M}) \longleftarrow (\mathcal{M}, \mathcal{B}, \theta_{i-1}, \psi^{-1}). \tag{1}$$

The memory carries useful information about both new tasks and past ones. We consider the information theory by its principled quantification for the data informativeness which can be employed in the memory selection. We propose to measure the informativeness of one batch of new training samples given memory $\mathcal{M}$, it can be defined as negative log conditional probability, formulated as:

$$\mathcal{L}_1((\mathcal{X}_i^{\mathcal{B}}, \mathcal{Y}_i^{\mathcal{B}}); \mathcal{M}) = -\log p(\mathcal{Y}_i^{\mathcal{B}} \mid \mathcal{Y}_i^{\mathcal{M}}; \mathcal{X}_i^{\mathcal{M}}, \mathcal{X}_i^{\mathcal{B}}), \tag{2}$$

Intuitively, the data distribution $\mathbb{P}_i$ itself will experience gradual generalization of concepts to new domains without forgetting the past distribution $\mathbb{P}_{past}$ at any moment. In other words, the model is continuously capturing and updating knowledge about past domains. Therefore, the informativeness of associated samples can be formulated as follows:

$$\mathcal{L}_2((\mathcal{X}_i^{\mathcal{B}}, \mathcal{Y}_i^{\mathcal{B}}); \mathcal{M}) = \log p(H^i(\psi^{-1}(\mathcal{Y}_i^{\mathcal{B}})) \mid \mathcal{Y}_i^{\mathcal{B}}, \mathcal{Y}_i^{\mathcal{M}}; \mathcal{X}_i^{\mathcal{M}}, \mathcal{X}_i^{\mathcal{B}}), \tag{3}$$

We denote η as the percentage of new task data backtracked to past episodes, which weighs the distribution between samples of new tasks and past ones. The informativeness of memory selection is described as the combination of new task samples and historical task ones to the $\mathcal{B}$:

$$HM((\mathcal{X}_i, \mathcal{Y}_i); \mathcal{M}) = \eta\mathcal{L}_1((\mathcal{X}_i, \mathcal{Y}_i); \mathcal{M}) + \mathcal{L}_2((\mathcal{X}_i, \mathcal{Y}_i); \mathcal{M}), \tag{4}$$

This term is relative to the information gain (Q., Cover, and Thomas 2006) and based on Jensen's inequality. Therefore, equation 4 can be rewritten as:

$$\begin{aligned} HM((\mathcal{X}_i, \mathcal{Y}_i); \mathcal{M}) =& \mathbb{E}_{p(\theta_i|\mathcal{Y}_i^{\mathcal{B}}, \mathcal{Y}_i^{\mathcal{M}}; \mathcal{X}_i^{\mathcal{B}}, \mathcal{X}_i^{\mathcal{M}})}[\log p(\mathcal{Y}_i \mid \theta_i; \mathcal{X}_i)] \\ & - \log p(H^i(\psi^{-1}(\mathcal{Y}_i^{\mathcal{B}})) \mid \mathcal{Y}_i^{\mathcal{M}}; \mathcal{X}_i^{\mathcal{M}}). \end{aligned} \tag{5}$$

Here θ_i denotes parameters of model H_i. It associates part of current ground truth with samples of previous tasks, which avoid saving original images or extracted features. We calculate the mean and standard deviation of HM for all observations and select samples whose value passes the sum of mean and deviation. Note that this proposed mechanism can be employed to any deep network architecture.

Parameter Isolation

Since the heuristics mechanism is only updated along the single side trajectory from $\mathbb{P}_i$ to $\mathbb{P}_{past}$ during the *i th* task training session, we introduce a parameter isolation mechanism which is similar to reducing the plasticity of synapses to alter the efficacy of synaptic transmission. The object function for incremental de-raining aims to reduce the empirical risk between current samples and past ones, while the historical knowledge is more effectively retained by heuristics memory system as discussed in Section . To determine the implementation of association and to find the vital synapse, we consider the neural network optimization from the probabilistic perspective.

For regular training process, parameter optimization is tantamount to finding the most probable data distribution for the model given all task data $\mathcal{D}$. We can use Bayes'rule to calculate the conditional probability $p(\theta|\mathcal{D})$:

$$\log p(\theta|\mathcal{X}, \mathcal{Y}) = \log p(\mathcal{X}, \mathcal{Y}|\theta) + \log p(\theta) - \log p(\mathcal{X}, \mathcal{Y}), \tag{6}$$

where $p(\mathcal{X}, \mathcal{Y})$ is the probability of data and $\log p(\theta)$ is the prior probability of parameters that can match all tasks.

For incremental de-raining setting, all task data is broken up into n batches $\mathcal{D} = \{\mathcal{D}_1, ..., \mathcal{D}_n\}$. We can derive the full objective for incremental de-raining from equation (6):

$$\begin{aligned} \log p(\theta_i|\mathcal{X}, \mathcal{Y}) =& \log p(\mathcal{X}_i, \mathcal{Y}_i|\theta_i) - \log p(\mathcal{X}_i, \mathcal{Y}_i) \\ & + \log p(\theta_i|\mathcal{X}_{past}, \mathcal{Y}_{past}), \end{aligned} \tag{7}$$

where $\{\mathcal{X}_i, \mathcal{Y}_i\}$ represents the data distribution of the ith task, and $\{\mathcal{X}_{past}, \mathcal{Y}_{past}\}$ represents the data distribution of previous $i-1$ tasks. Note that the probability of the ith task data $p(\mathcal{X}_i, \mathcal{Y}_i)$ is a constant value, and the log probability of the ith task data given the ith task parameters $\log p(\mathcal{X}_i, \mathcal{Y}_i|\theta_i)$ is simply tantamount to the preliminary loss function $\mathcal{L}$ at hand.

$\log p(\theta_i|\mathcal{X}_{past}, \mathcal{Y}_{past})$ is a posterior probability distribution term and it contains information about all previous task data, which is difficult to calculate. This term has to be approximated by diagonalized Laplace approximation (MacKay 1992) as a Gaussian distribution. The mean value is the parameters of θ^*_{past} and the variance is the diagonal reciprocal of the Fisher information matrix $\mathcal{F}$ corresponding to the parameters. Given this approximation, the optimization problem of reducing the plasticity can be defined as altering the efficacy of synaptic transmission from past distribution $\mathbb{P}_{past}$ to refined distribution $\mathbb{P}_i$ for approximation, formulated as:

$$\mathcal{F}_{D_{past}}(\theta) = \mathbb{E}_{\{\mathcal{X}_i, \mathcal{Y}_i\} \sim \mathbb{P}_i} \left(\left. \frac{\partial^2 f(\mathcal{Y}_i|\mathcal{X}_i; \theta)}{\partial^2 \theta} \right|_{\theta=\theta^*_{\text{past}}} \right), \tag{8}$$

where $f(\mathcal{Y}|\mathcal{X}; \theta)$ is the probability density function of $p(\theta \mid \mathcal{X}_{past}, \mathcal{Y}_{past})$ as known as $p\,(\theta \mid D_{past})$.

After task T_{i-1} is trained, the parameters θ_{i-1} of model H^{i-1} will be saved as the existing acquired knowledge for the next task T_i. This approximation term serves as the synaptic transmission between feature space of different tasks, alleviating over drifting in feature domains. The loss guides parameters of current circumstance θ_i to be updated to the associated feature representation. It only depends on the previous task parameters θ_{i-1} without additional data throughout the whole training process.

Model	Methods	Rain100H		Rain100L		Promotion on Rain100H	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
NLEDN	Baseline	15.84	0.532	34.53	0.958	**10.21**	**0.277**
	PIGWM	20.96	0.736	34.93	0.961		
	AM	26.05	0.809	35.05	0.966		
	Reference	27.11	0.835	35.26	0.963		
PreNet	Baseline	18.97	0.639	38.29	0.981	**9.87**	**0.261**
	PIGWM	28.08	0.89	36.95	0.975		
	AM	28.84	0.900	37.13	0.978		
	Reference	29.46	0.899	37.48	0.979		
PRN	Baseline	18.29	0.619	37.34	0.978	**9.68**	**0.274**
	PIGWM	27.88	0.88	35.64	0.967		
	AM	27.97	0.893	36.42	0.973		
	Reference	28.07	0.884	36.99	0.977		
SASI	Baseline	19.42	0.673	37.4	0.98	**10.63**	**0.238**
	PIGWM	29.76	0.879	36.73	0.968		
	AM	30.05	0.911	37.51	0.979		
	Reference	30.33	0.909	38.8	0.984		
REHEN	Baseline	14.31	0.423	37.34	0.974	**12.94**	**0.437**
	PIGWM	26.76	0.856	35.68	0.961		
	AM	27.25	0.86	37.81	0.969		
	Reference	27.97	0.864	37.41	0.98		

Table 1: Comparison of quantitative results in terms of PSNR and SSIM. Models are trained sequentially on the task sequence Rain100H-Rain100L using schemes of baseline, PIGWM an AM, respectively.

The parameter isolation enforces the model to update parameters to reconstruct the old experience as well as the new one. It preserves relevant past experience and generalize the concept to new domains. Therefore, the model can still continually retain the distribution of past domains $\mathbb{P}_{past}$ while integrating new tasks.

Note that datasets of different tasks are not completely independent in our setup of association system. To handle the conflict of mixed data distribution, we approximate the the distribution $p(\mathcal{X})$ by Monte-Carlo (MC) method as $p(\tilde{\mathcal{X}_i})$, when given the prior of the mixed sample $\tilde{\mathcal{X}_i}$. Therefore, we can have the following equation:

$$\begin{aligned} p(\mathcal{X}_i) &= \int_{\tilde{D}} p\left(\tilde{\mathcal{X}_i}\right) d\tilde{\mathcal{X}_i} \\ &\approx \frac{1}{\tilde{\mathcal{Y}_i}} \sum_{i=1}^{\tilde{\mathcal{Y}_i}} p\left(\tilde{\mathcal{X}_i}\right). \end{aligned} \tag{9}$$

where $\mathcal{X}_i, \tilde{\mathcal{X}_i}$ and $\tilde{\mathcal{Y}_i}$ denote samples in the current task, current samples mixed with selected samples from past tasks, and target samples, respectively. The distribution $\tilde{D}$ denotes the data distribution defined by mixed samples $\tilde{\mathcal{X}_i}$. We have used heuristics mechanism to select mappings from past tasks as random variables, and according to the Monte-Carlo method, the model will get real probability distributions of datasets → tasks. Finally, we optimize the model by combining the parameter isolation $\mathcal{F}$ with $\mathcal{L}$. We use the new trade-off weight $\lambda^{'}$ for balance.

Experiments

Experiment Setup

In order to make a fair comparison with state-of-the-art approaches, we follow the standard experiment setup of previous work (Zhou et al. 2021) for benchmark datasets, evaluation metrics and baseline implementations.

Benchmark Datasets. We evaluate all incremental de-raining methods on four benchmark datasets: Rain100L (Yang et al. 2017), Rain100H (Yang et al. 2017), Rain800 (Zhang, Sindagi, and Patel 2019) and Rain1400 (Fu et al. 2017c). Both Rain100L and Rain100H contain 1900 pairs of rainy and clean images, Rain800 has 800 pairs and Rain1400 has 1400 pairs. Following the previous work, we partition training and testing samples of each dataset according to the existing split. Besides, in order to explore the model performance on real world images after learning the task sequence, we also evaluate all methods on recent public available dataset SPA-Data (Wang et al. 2019b). In this work, we use the test set of SPA-Data for evaluation, which contains 1000 pairs of rainy images and their corresponding labeled clean images. Note that we only train de-raining models on the task sequence of synthetic datasets with no access to the SPA-Data. We also collect some real-life rainy images without ground truth on the Internet for qualitative comparison among all models.

Baseline. We benchmark our scheme against the latest method PIGWM (Zhou et al. 2021) designed for incremental rain removal. Note that our technology is non-exemplar

Model	Methods	Rain800		Rain100L		Promotion on Rain800	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
ID-cgan	Baseline	20.57	0.645	25.56	0.876	**3.45**	**0.190**
	PIGWM	23.36	0.822	24.13	0.856		
	AM	24.02	0.835	25.72	0.885		
	Reference	24.34	0.843	25.88	0.891		

Table 2: Comparison of quantitative results in terms of PSNR and SSIM. The model is trained sequentially on the task sequence Rain800-Rain100L using schemes of baseline, PIGWM an AM, respectively.

Model	Methods	Rain800		Rain100H		Promotion on Rain800	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
ID-cgan	Baseline	19.89	0.641	13.25	0.598	**4.04**	**0.189**
	PIGWM	23.08	0.815	11.16	0.532		
	AM	23.93	0.830	12.87	0.551		
	Reference	24.34	0.843	14.16	0.607		

Table 3: Comparison of quantitative results in terms of PSNR and SSIM. The model is trained sequentially on the task sequence Rain800-Rain100H using schemes of baseline, PIGWM an AM, respectively.

based and all approaches, ours included, do not use the practice of storing extra data considering the space overhead of revisiting historical samples. We also compare incremental approaches with two reference schemes: joint learning and transfer learning. Joint learning sets up the reference for incremental learning, where all task data is learned together at a time. Transfer learning sets up the baseline for incremental learning, where the data of each task is fed into the model sequentially for training. Here we use parameters obtained from the previous task to initialize the current model, so that preserved information can be fully leveraged. These two methods provide a comparison for the performance of incremental de-raining methods.

The purpose of our experiment setting is to verify the effectiveness of all incremental methods tackling the continual image rain removal problem. Since all methods are independent of any specific de-raining models, we use following representative models as the baseline de-raining architecture to integrate the above methods: NLEDN (Li et al. 2018a), PreNet (Ren et al. 2019), PRN (Ren et al. 2019), SASI (Wang et al. 2019a), REHEN (Yang and Lu 2019), ID-cgan (Zhang, Sindagi, and Patel 2019). All these models achieve state-of-the-art performances on single-image rain removal. We also abandon the non-local operation in NLEDN to ensure that the model architecture is consistent with that of previous work.

Implement Details. In the experiment setup of incremental rain removal, the model is exposed to a sequence of datasets. Each time step when the model learns a new task, parameters well-trained on the recent dataset will be updated by the new dataset without additional provisions of previous datasets. Following the closely related work, Rain100H and Rain100L (Rain100H-Rain100L) are sequentially fed into PreNet, PRN, NLEDN, REHEN and SASI. Besides, incremental task sequences Rain800-Rain100L and Rain800-Rain100H are executed on ID-cgan which first proposes Rain800. Furthermore, we experiment on a task sequence Rain100H-Rain100L-Rain1400 to validate the performance of multiple incremental datasets. Note that both PIGWM and AM are independent of specific model structure, so we keep all training techniques and parameters setting consistent with implementations in original papers for a fair comparison. All experiments are conducted on NVIDIA Tesla V100 GPUs. After training the sequence of all task datasets, we will assess the ultimate model on all task datasets and real world images. Our experiment aims to observe the improvement on the performance of historical task with limited decrease on that of new task.

Evaluation Metrics. We evaluate the quality of prediction through qualitative and quantitative analysis. The qualitative evaluation mainly relies on visual perception, and the observed pictures include synthetic images and real-world ones. The quantitative evaluation utilizes peak signal to noise ratio (PSNR) (Huynh-Thu and Ghanbari 2008) and structural similarity (SSIM) (Wang et al. 2004) to quantify the performance. PSNR measures the difference between corresponding pixel values, and SSIM measures the holistic similarity from three aspects close to the visual characteristic of human eye: brightness, contrast, and structure.

Results on Benchmark Datasets

Quantitative Results. Table 1, 2 and 3 present the SSIM and PSNR results of task sequence Rain100H-Rain100L, Rain800-Rain100L and Rain800-Rain100H for quantitative evaluations, respectively. Baseline rows refer to the transfer learning results while reference rows refer to the joint learning results. Note that we train each dataset individually as the joint learning result of each task. We use results that reported by original authors except AM. We follow the experiment setting of previous works to train modi-

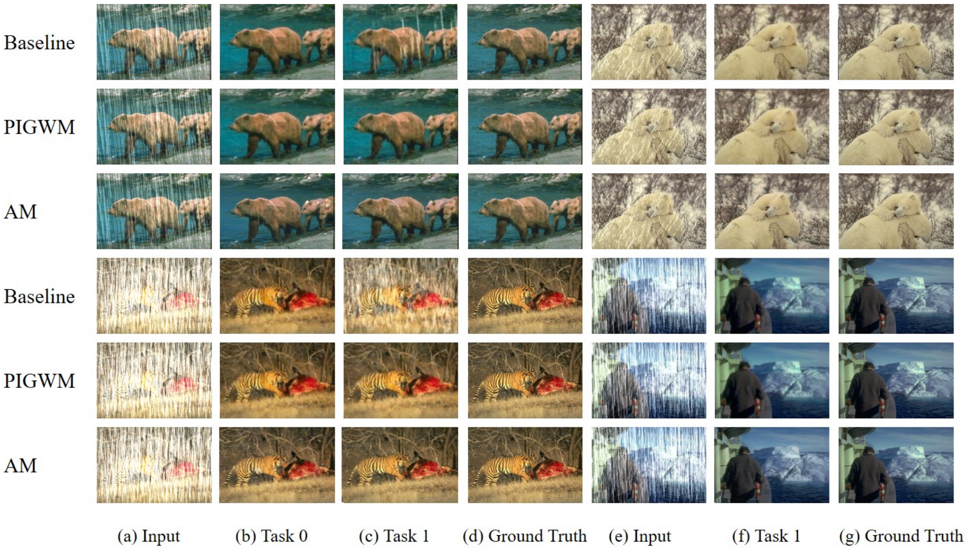

Figure 2: Visual comparison of rain-streaks removal results generated from the incremental de-raining process using model PreNet. (a) Input: rainy image from Rain100H; (b) Task 0: train and test on Rain100H; (c) Task 1: train model (b) on Rain100L and test on Rain100H; (d) Ground Truth: clean image of (a); (e) Input: rainy image from Rain100L; (f) Task 1: train model (b) on Rain100L and test on Rain100L; (g) Ground Truth: clean image of (e).

Test set	Rain100H	Rain100L	Rain1400
Baseline	15.31 / 0.424	28.88 / 0.892	31.90 / 0.927
PIGWM	28.18 / 0.891	36.85 / 0.975	28.06 / 0.864
AM	29.80 / 0.893	37.23 / 0.975	29.23 / 0.879
Reference	29.46 / 0.899	37.48 / 0.979	32.60 / 0.946

Table 4: PSNR and SSIM results of PreNet trained on the task sequence Rain100H-Rain100L-Rain1400.

fied de-raining models integrated with the AM structure and evaluate the well-trained model on test sets of benchmark datasets. As shown in these tables, Reference is not subject to incremental de-raining conditions and it takes the achievement of the highest numerical performance for granted in most cases, which sets up the baseline for incremental de-raining setting. Baseline obviously completely forgets the acquired knowledge. Both AM and PIGWM retain the content coherence and does not to be influenced by changing circumstances. Between them, AM obviously works better than PIGWM and achieves the greatest improvement over the baseline method across all task sequences on all models. We can conclude that AM illustrates good generalization and effectiveness, showing the superior performance compared to existing methods on image rain removal.

Qualitative Results. In order to visually illustrate the catastrophic forgetting of incremental de-raining and the performance of different approaches, we show the learning process on task sequences. Specifically, we take the promising PreNet as one example, and rain removal results are summarized in Figure 2. These results illustrate that our inference model obtains predictions indistinguishable from ground truth and works well on all task datasets. We also compare AM with other schemes. Reference performs better than other methods and achieves the most visually satisfactory results. The results of PIGWM are visually close to those of AM, indicating that both can effectively deal with the catastrophic forgetting. But historical outputs of PIGWM seem increasingly blurry and some reinforced generation artifacts (Wang et al. 2018) exist. It indicates that PIGWM is more sensitive to artifacts and will be reinforced during the intermediate task training (Zhai et al. 2019), while AM shows more robustness and less sensitiveness to them. Baseline is unable to capture previous concepts and suffers from catastrophic forgetting. Overall, hallucinated results produced by AM are perceptually convincing and AM performs particularly well on all task sequences without forgetting historical learned mapping rules.

Extension to Multiple Datasets

We utilize PreNet as the baseline model and extend all methods to multiple datasets. For a set of n tasks, we follow the step similar to the dynamic programming for incremental learning. When learning the ith task, we can regard the past $i{-}1$ tasks as a whole task, and its corresponding well-trained model can already solve these $i - 1$ tasks. Then, the ith task is sequentially fed into this model for training, and the model continues to learn new task while retaining parameters of last tasks throughout sequential training on all cases. Table 4 presents results on the task sequence Rain100H-Rain100L-Rain1400, which further demonstrate the best effectiveness of AM among all existing approaches. Furthermore, the overhead of mapping memory per task is much smaller than saving subsets of training data of each task, and the time overhead of AM is higher than that of Baseline and PIGWM but lower than that of Reference.

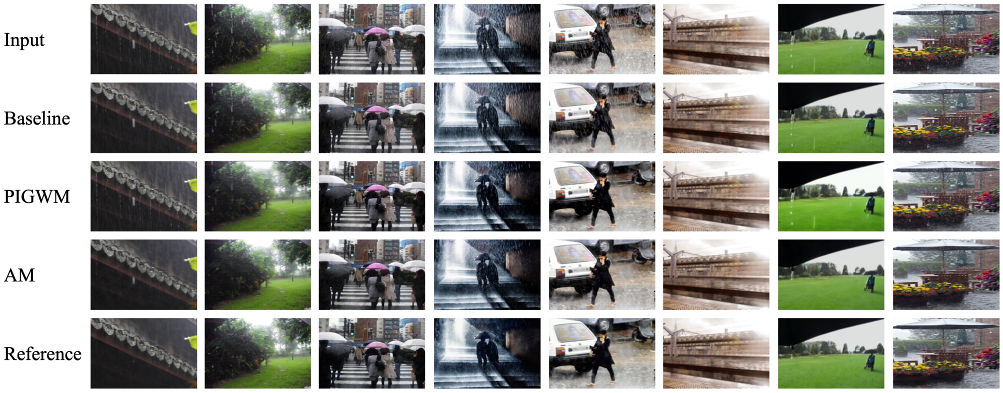

Figure 3: Some removal results on real rain images obtained from the Internet using model PRN.

Real World	Baseline	PIGWM	AM	Reference
PSNR	33.69	34.83	34.93	35.02
SSIM	0.949	0.952	0.955	0.957

Table 5: PSNR and SSIM results of REHEN on SPA-Data. The model is only trained on the task sequence Rain1400-Rain100H-Rain100L without real-world SPA-Data.

Results on Real World Data

Quantitative Results. We select REHEN as the baseline model and assess the performance of all methods on the real world dataset SPA-Data. These models are all trained on the task sequence Rain1400-Rain100H-Rain100L, and we use the ultimate model after training all datasets for evaluation. Table 5 presents results on the SPA-Data comparing our method with other approaches. Reference model provides a ideal reference for the quantitative analysis. When countered with real world images, baseline model produces the worst results which only learns mapping rules of recent tasks and suffers from the catastrophic forgetting. AM achieves more closely homogeneous performance across multiple synthetic datasets compared with another incremental deraining method PIGWM. It reaches the best generalization ability in real world scenarios.

Qualitative Results. We also collect some real-world images downloaded from the Internet and evaluate our final model on these images. We take PRN as the baseline structure and some removal results of all methods are shown in Figure 3. Reference results show the ideal rain removal effect if diverse samples with different types of rain streaks can be obtained in the episodic memory. Due to the catastrophic forgetting, the baseline model has limited capability on the image rain removal under incremental conditions, and has not completely eliminated rain streaks. Both AM and PIGWM can remove the real-world rain streaks and preserve some details effectively, but slight rain streaks still exist in output predictions of PIGWM. It demonstrates that our model still memorizes acquired rain streaks and maintains great superiority.

Module	PSNR	SSIM
SASI	19.42	0.673
SASI+Heuristics	29.14	0.883
SASI+Heuristics+Parameter Isolation	30.05	0.911

Table 6: PSNR and SSIM results of each module on SASI trained on the task sequence Rain100H-Rain100L.

Ablation Study

We analyze each component of AM to illustrate the impact on the performance of final models. We take SASI as the baseline model for example. As shown in Table 6, SASI achieves the worst performance in all cases since tasks are reached sequentially and cannot be recurred in a long-time interval. it is necessary to utilize the heuristics mechanism to effectively reconstruct the current domain with large incremental steps. It illustrates that using the heuristics mechanism improves PSNR from 19.42 dB to 29.14 dB and SSIM from 0.673 to 0.883, which contributes the most to the final performance. By dampening the feature reconstruction, the two metrics are slightly improved. It depresses the adaptation of new episodes and guides the prediction coherent with previous circumstances. We can conclude that exploiting past memory consolidation for gradient descent reduces the plasticity and proves beneficial. Ultimately, after all components are added, our full model obtains the best results.

Conclusion

In this paper, we explore the sample diversity for incremental image de-raining and propose an associative memory management scheme mediated by heuristics mechanism and parameter isolation. Experiments demonstrate that AM significantly generates satisfactory results without forgetting historical tasks and it performs better than existing approaches within the span of multiple rain streaks. It can also be extended to any model in a plug-and-play mode. Since we still need to provide the known task data for new task learning, in subsequent research, we will focus on enhancing the ability to identify the unknown input sample and decide whether to incrementally learn from it.

Acknowledgments

This work has been partially supported by the National Key R&D Program of China under Grant 2020AAA0103902, the National Key R&D Program of China Nos. 2020YFB1806700, NSFC Grant 61932014 and Project BE2020026 supported by the Key R&D Program of Jiangsu, China. Jie Li is the corresponding author.

References

Douillard, A.; Chen, Y.; Dapogny, A.; and Cord, M. 2021. Plop: Learning without forgetting for continual semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4040–4050.

Fu, X.; Huang, J.; Ding, X.; Liao, Y.; and Paisley, J. 2017a. Clearing the Skies: A Deep Network Architecture for Single-Image Rain Removal. *IEEE Transactions on Image Processing*, 26(6): 2944–2956.

Fu, X.; Huang, J.; Zeng, D.; Huang, Y.; Ding, X.; and Paisley, J. 2017b. Removing Rain From Single Images via a Deep Detail Network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Fu, X.; Huang, J.; Zeng, D.; Huang, Y.; Ding, X.; and Paisley, J. 2017c. Removing rain from single images via a deep detail network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3855–3863.

Huynh-Thu, Q.; and Ghanbari, M. 2008. Scope of validity of PSNR in image/video quality assessment. *Electronics letters*, 44(13): 800–801.

Kang, L.-W.; Lin, C.-W.; and Fu, Y.-H. 2011. Automatic single-image-based rain streaks removal via composition. *IEEE transactions on image processing*, 21(4): 1742–1755.

Kang, L.-W.; Lin, C.-W.; and Fu, Y.-H. 2012. Automatic Single-Image-Based Rain Streaks Removal via Image Decomposition. *IEEE Transactions on Image Processing*, 21(4): 1742–1755.

Kim, J.-H.; Lee, C.; Sim, J.-Y.; and Kim, C.-S. 2013. Single-image deraining using an adaptive nonlocal means filter. In *2013 IEEE international conference on image processing*, 914–917. IEEE.

Lee, J.; Hong, H. G.; Joo, D.; and Kim, J. 2020. Continual Learning with Extended Kronecker-factored Approximate Curvature. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9001–9010.

Li, G.; He, X.; Zhang, W.; Chang, H.; Dong, L.; and Lin, L. 2018a. Non-locally enhanced encoder-decoder network for single image de-raining. In *Proceedings of the 26th ACM international conference on Multimedia*, 1056–1064.

Li, G.; He, X.; Zhang, W.; Chang, H.; Dong, L.; and Lin, L. 2018b. Non-Locally Enhanced Encoder-Decoder Network for Single Image De-Raining. 1056–1064. Association for Computing Machinery. ISBN 9781450356657.

Li, X.; Wu, J.; Lin, Z.; Liu, H.; and Zha, H. 2018c. Recurrent Squeeze-and-Excitation Context Aggregation Net for Single Image Deraining. In *Proceedings of the European Conference on Computer Vision (ECCV).*

Li, Y.; Tan, R. T.; Guo, X.; Lu, J.; and Brown, M. S. 2016. Rain streak removal using layer priors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2736–2744.

Li, Z.; and Hoiem, D. 2017. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12): 2935–2947.

Luo, Y.; Xu, Y.; and Ji, H. 2015. Removing rain from a single image via discriminative sparse coding. In *Proceedings of the IEEE International Conference on Computer Vision*, 3397–3405.

MacKay, D. J. 1992. A practical Bayesian framework for backpropagation networks. *Neural computation*, 4(3): 448–472.

McCloskey, M.; and Cohen, N. J. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, 109–165. Elsevier.

Pan, J.; Liu, S.; Sun, D.; Zhang, J.; Liu, Y.; Ren, J.; Li, Z.; Tang, J.; Lu, H.; Tai, Y.-W.; and Yang, M.-H. 2018. Learning Dual Convolutional Neural Networks for Low-Level Vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Q., C.; Cover, T. M.; and Thomas, J. A. 2006. *Elements of information theory*. Elements of information theory.

Ren, D.; Zuo, W.; Hu, Q.; Zhu, P.; and Meng, D. 2019. Progressive image deraining networks: A better and simpler baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3937–3946.

Wang, T.; Yang, X.; Xu, K.; Chen, S.; Zhang, Q.; and Lau, R. W. 2019a. Spatial attentive single-image deraining with a high quality real rain dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12270–12279.

Wang, T.; Yang, X.; Xu, K.; Chen, S.; Zhang, Q.; and Lau, R. W. 2019b. Spatial Attentive Single-Image Deraining With a High Quality Real Rain Dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).*

Wang, Y.; Wu, C.; Herranz, L.; van de Weijer, J.; Gonzalez-Garcia, A.; and Raducanu, B. 2018. Transferring GANs: generating images from limited data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 218–234.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4): 600–612.

Wu, C.; Herranz, L.; Liu, X.; van de Weijer, J.; Raducanu, B.; et al. 2018. Memory replay gans: Learning to generate new categories without forgetting. In *Advances in Neural Information Processing Systems*, 5962–5972.

Wu, Z.; Baek, C.; You, C.; and Ma, Y. 2021. Incremental learning via rate reduction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1125–1133.

Xiang, Y.; Fu, Y.; Ji, P.; and Huang, H. 2019. Incremental learning using conditional adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, 6619–6628.

Yang, W.; Tan, R. T.; Feng, J.; Liu, J.; Guo, Z.; and Yan, S. 2017. Deep joint rain detection and removal from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1357–1366.

Yang, Y.; and Lu, H. 2019. Single image deraining via recurrent hierarchy enhancement network. In *Proceedings of the 27th ACM International Conference on Multimedia*, 1814–1822.

Zhai, M.; Chen, L.; Tung, F.; He, J.; Nawhal, M.; and Mori, G. 2019. Lifelong gan: Continual learning for conditional image generation. In *Proceedings of the IEEE International Conference on Computer Vision*, 2759–2768.

Zhang, H.; and Patel, V. M. 2018. Density-Aware Single Image De-Raining Using a Multi-Stream Dense Network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, H.; Sindagi, V.; and Patel, V. M. 2019. Image deraining using a conditional generative adversarial network. *IEEE transactions on circuits and systems for video technology*, 30(11): 3943–3956.

Zhou, M.; Xiao, J.; Chang, Y.; Fu, X.; Liu, A.; Pan, J.; and Zha, Z.-J. 2021. Image De-Raining via Continual Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4907–4916.

Zhu, L.; Fu, C.-W.; Lischinski, D.; and Heng, P.-A. 2017. Joint bi-layer optimization for single-image rain streak removal. In *Proceedings of the IEEE international conference on computer vision*, 2526–2534.