Robust Feature Rectification of Pretrained Vision Models for Object Recognition

Shengchao Zhou^{1,2}**, Gaofeng Meng**^{1,2,3*}**, Zhaoxiang Zhang**^{1,2,3}**, Richard Yi Da Xu**^{4}
and Shiming Xiang^{1,2}

[1] NLPR, Institute of Automation, Chinese Academy of Sciences,
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences,
[3] CAIR, HK Institute of Science and Innovation, Chinese Academy of Sciences,
[4] FSC1209, Kowloon Tong Campus, Hong Kong Baptist University
{zhoushengchao2021, zhaoxiang.zhang}@ia.ac.cn, {gfmeng, smxiang}@nlpr.ia.ac.cn, xuyida@hkbu.edu.hk

Abstract

Pretrained vision models for object recognition often suffer a dramatic performance drop with degradations unseen during training. In this work, we propose a RObust FEature Rectification module (ROFER) to improve the performance of pretrained models against degradations. Specifically, ROFER first estimates the type and intensity of the degradation that corrupts the image features. Then, it leverages a Fully Convolutional Network (FCN) to rectify the features from the degradation by pulling them back to clear features. ROFER is a general-purpose module that can address various degradations simultaneously, including blur, noise, and low contrast. Besides, it can be plugged into pretrained models seamlessly to rectify the degraded features without retraining the whole model. Furthermore, ROFER can be easily extended to address composite degradations by adopting a beam search algorithm to find the composition order. Evaluations on CIFAR-10 and Tiny-ImageNet demonstrate that the accuracy of ROFER is 5% higher than that of SOTA methods on different degradations. With respect to composite degradations, ROFER improves the accuracy of a pretrained CNN by 10% and 6% on CIFAR-10 and Tiny-ImageNet respectively.

Introduction

Recent years have witnessed a remarkable advancement of object recognition with the help of convolutional neural network (CNN) models (Brock et al. 2021; He et al. 2016; Krizhevsky, Sutskever, and Hinton 2012). However, the models often suffer a performance drop with degradations unseen during training (Hendrycks and Dietterich 2019; Dodge and Karam 2016). As illustrated in Figure 1, a VGG19 (Simonyan and Zisserman 2015) trained on clear images predicts the correct label of a clear *cheetah* image, while the prediction on the blurred image fails. Worse still, as a ubiquitous situation, many images are affected by composite degradations, making the problem even more severe.

Retraining the models on degraded images is a direct solution, however, it is time-consuming and error-prone to annotate degraded images. Moreover, since degradations can compound in numerous orders, a large-scale dataset of compositely degraded images is required to deal with various composite degradations.

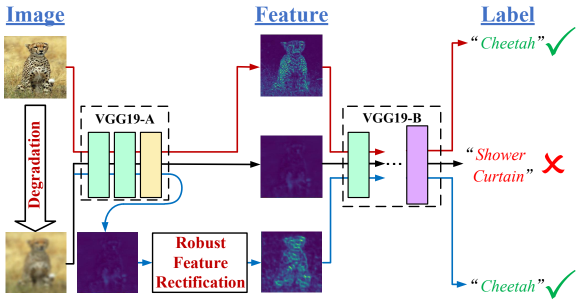


Figure 1: Illustration of feature rectification. Pretrained on clear images, a VGG19 predicts the correct label of a clear *cheetah* image. When the image is blurred, the feature map is degraded and the prediction fails. Our proposed "Robust Feature Rectification" rectifies the degraded feature to correct the wrong prediction.

Only requiring unlabeled degraded images during training, existing methods commonly adopt unsupervised strategies, most of which can be divided into three categories, namely enhancement-based, restoration-based, and domain adaptation-based (DA-based) methods. Enhancement-based methods train an image enhancement module and perform recognition after enhancement to mitigate the influence of degradations (Pei et al. 2021; Li et al. 2017; Dai et al. 2016). Restoration-based methods train a plugging-in module to restore the features of degraded images, making them resemble clear features (Wang et al. 2020). DA-based methods transfer the model to degraded images by training it on generated pseudo labels or pseudo images (Rusak et al. 2021; Schneider et al. 2020; Russo et al. 2018; Motiian et al. 2017).

Despite their effectiveness on recognition with degradations, existing methods have deficiencies as follows: (1) Aiming at improving the visual quality, the image enhancement module in enhancement-based methods can introduce artifacts interfering with the recognition. (2) Most methods are designed to address only one single degradation during testing, not versatile enough to deal with different degradations simultaneously. (3) Methods designed for composite

[*] Corresponding author.

degradations require a large-scale dataset for training (**?**Suganuma, Liu, and Okatani 2019), which consists of images affected by various composite degradations.

To address the above problems, we propose a RObust FEature Rectification module (ROFER) to improve the performance of pretrained CNNs against degradations. Specifically, to deal with various degradations, ROFER takes the features of degraded images as input and estimates the type and intensity of degradations affecting them. According to the estimation, it produces some kernels as the first layer of a Fully Convolutional Network (FCN) and leverages the FCN to rectify the features, rather than visual quality, from the degradation by pulling them back to clear features. ROFER is a general-purpose module that can address various degradations simultaneously, including blur, noise, and low contrast. Besides, it can be plugged into pretrained models seamlessly to rectify the features without retraining the whole model. Furthermore, by adopting a beam search algorithm to find the composition order, ROFER can be easily extended to address composite degradations when trained only on unlabeled images affected by a single degradation.

In summary, our contributions are three-fold:

- We propose a RObust FEature Rectification module (ROFER) to improve the performance of pretrained CNNs against degradations. ROFER can be plugged into pretrained models seamlessly to rectify the degraded image features by pulling them back to clear features.
- ROFER is a general-purpose module that can address various degradations simultaneously, including blur, noise, and low contrast. Adopting a beam search algorithm to find the composition order, ROFER can be extended to address composite degradations effectively when trained on images affected by a single degradation.
- Evaluations on CIFAR-10 and Tiny-ImageNet demonstrate that ROFER obtains 5% higher accuracy than SOTA methods on different degradations. With respect to composite degradations, ROFER improves the accuracy of pretrained CNNs by 10% and 6% on CIFAR-10 and Tiny-ImageNet respectively.

Related Work

Image Enhancement

Image enhancement can improve the visual quality of images (Tran et al. 2021; Tu et al. 2022; Kupyn et al. 2018; Guo et al. 2019). Zamir *et al.* (Zamir et al. 2021) and Cho *et al.* (Cho et al. 2021) proposed CNN-based deblurring methods. Zhang *et al.* (Zhang et al. 2017; Zhang, Zuo, and Zhang 2018) developed a series of denoising works. Lee *et al.* (Lee, Lee, and Kim 2013) proposed a contrast enhancement algorithm. Although image enhancement mitigates the effect of degradations on recognition, for better visual quality, it can introduce artifacts interfering with the recognition. By contrast, our proposed method rectifies the degraded features by pulling them back to clear features, avoiding artifacts.

Feature Restoration

Feature restoration methods restore the image features from distortions for high-level tasks (Zhang et al. 2020; Zhou et al. 2020; Wang et al. 2022; Sun et al. 2018). With respect to the recognition task, Tan *et al.* (Tan, Yan, and Bare 2018) proposed a feature super-resolution method to produce high-resolution features. Wang *et al.* (Wang et al. 2020) developed a feature de-drifting module that maps the degraded features to clear features. Although belonging to this category of methods, our proposed ROFER can predict the type and intensity of degradations to address a variety of them simultaneously, while the above methods only deal with one single degradation during testing. Moreover, ROFER can be extended to address composite degradations, which is not achieved by the above methods.

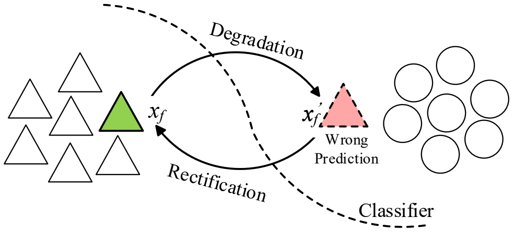


Figure 2: Illustration of feature drifting by degradations. The degradation pushes the original clear features across the decision boundary, yielding wrong predictions. Our proposed ROFER rectifies the degraded features by pulling them back to their original positions to correct the predictions.

Domain Adaptation

Domain adaptation adapts the model trained on a source domain to another target domain (Xu et al. 2022; Stojanov et al. 2021; Cui et al. 2020; Sankaranarayanan et al. 2018; Bousmalis et al. 2017). To help pretrained models recognize with degradations, Schneider *et al.* (Schneider et al. 2020) proposed to adapt the batch normalization statistics to degraded images. Rusak *et al.* (Rusak et al. 2021) proposed to train models on pseudo labels generated for degraded images. While being able to handle composite degradations, these methods require a large-scale dataset of compositely degraded images for training. However, with a beam search algorithm, training on singly degraded images is enough to extend ROFER to address composite degradations.

Methodology

Problem Statement

Given an object recognition model $\Phi(f(\cdot))$ consisting of a feature extraction backbone f and a classifier Φ, it is trained on a set of clear images D_{train}=$\{(x_{\text{i}}, l_{\text{i}})\}_{\text{i}=1}^{N}$, where x_{i} is an image and l_{i} is its label. However, images in the testing dataset D_{test}=$\{(x'_{\text{test,i}}, l_{\text{test,i}}, deg_{\text{i}})\}_{\text{i}=1}^{M}$ are degraded, where there are degradations Deg=$\{deg^{(1)}, ..., deg^{(n)}\}$ and every $deg_{\text{i}} \in Deg$. Each $x'_{\text{test,i}}$ is a degraded image produced by affecting its original clear image $x_{\text{test,i}}$ with deg_{i} and $l_{\text{test,i}}$ is its groud-truth label. Our goal is to improve the performance of the model $\Phi(f(\cdot))$ on degraded images in D_{test}.

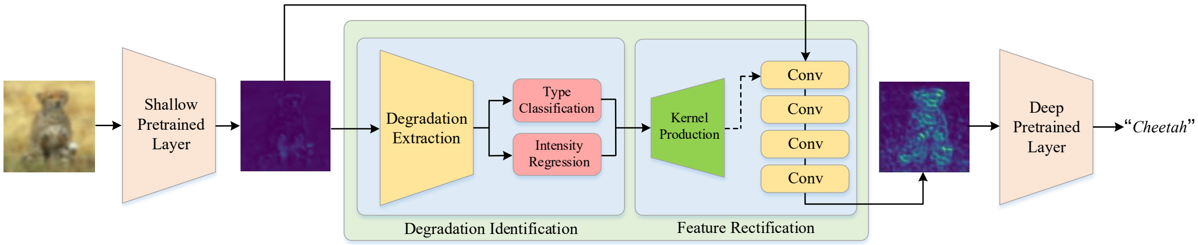


Figure 3: Architecture overview of ROFER. It works in two stages "Degradation Identification" and "Feature Rectification" to address one degradation. In the first stage, the type and intensity of degradations are predicted. In the second stage, one degradation is selected to be addressed and used to produce some kernels as the first layer of a Fully Convolutional Network (FCN). Then, the degraded features are rectified by the FCN. The whole procedure can iterate to address composite degradations.

Motivation

Our motivation comes from the observation that the performance drop mainly results from feature drifting rather than poor visual quality. As illustrated in Figure 2, given the feature x_f extracted by f from a clear image x and its label l, the classifier Φ predicts the correct label. However, when x is affected by a degradation to be x', x_f is pushed across the decision boundary of Φ to be x'_f, yielding a wrong prediction $l' \neq l$. To correct the prediction, a module $\hat{M}$ that pulls x'_f back to x_f is beneficial, formulated as:

$$\hat{M} = \arg\min_{M} |M(x'_f) - x_f|,$$
$$s.t.\ \Phi(M(x'_f)) = l. \tag{1}$$

Feature Rectification of Single Degradation

Overview With the above purpose, we propose a module ROFER, which learns to rectify the image features. As the overview illustrated in Figure 3, in one iteration, ROFER takes the feature as input and works in two stages to address a single degradation. In the first stage "Degradation Identification", the type and intensity of degradations affecting the feature are predicted. In the second stage "Feature Rectification", according to the predictions, one degradation is selected to be addressed and used to produce some kernels that are regarded as the first layer of a Fully Convolutional Network (FCN). Then, the FCN takes the feature as input and rectifies it from the selected degradation. The rectified feature is sent to deep layers for further process.

Degradation Identification To deal with various degradations, the "Degradation Identification" stage, in which the type and intensity of degradations are predicted, is proposed. The procedure of it is illustrated in Figure 3, the "Degradation Identification" part.

First, a backbone CNN named "Degradation Extraction" (DegExt) takes the degraded feature x'_f as input to produce a down-sampled result x'_{down}:

$$x'_{\text{down}} = \text{DegExt}(x'_f). \tag{2}$$

In our design, the backbone consists of 3 convolutional layers followed by batch normalization and ReLU operations. x'_{down} is then flattened to a 1D vector and sent to two heads "Type Classification" (Cls) and "Intensity Regression" (Reg) for prediction. Both heads are 3-layer fully connected networks and the predictions are two vectors $P' = (p'_0, p'_1, p'_2, \ldots, p'_n)$ and $Q' = (q'_1, q'_2, \ldots, q'_n)$ respectively:

$$P' = \text{Cls}(x'_{\text{down}}),\ Q' = \text{Reg}(x'_{\text{down}}), \tag{3}$$

where each $p'_{\text{k}}(\text{k} \geq 1)$ predicts the probability whether $deg^{(k)}$ affects the feature and p'_0 is the probability that there is no degradation. Each q'_{k} predicts the intensity of $deg^{(k)}$.

Feature Rectification With the predictions about the type and intensity of degradations, the degraded feature can be rectified accordingly and the "Feature Rectification" stage is proposed for implementation. The procedure of it is illustrated in Figure 3, the "Feature Rectification" part.

First, one degradation is selected according to the predictions. Assuming that the selected degradation is $deg^{(i)}$, then every value of the intensity vector Q' except q'_{i} is set to zero. After that, a module named "Kernel Production" (KerProd), which is a 2-layer fully connected network, takes the modified Q' as input and produces another vector V:

$$V = \text{KerProd}(Q'). \tag{4}$$

V is then reshaped into a kernel K of size $c \times c \times 3 \times 3$ and regarded as the first layer of a FCN, where c is the channel number of x'_f. The FCN, which consists of 4 convolutional layers followed by batch normalization and ReLU operations, takes x'_f as input and rectifies it from $deg^{(i)}$:

$$x_f^{\text{rec}} = \text{FCN}(K(x'_f)). \tag{5}$$

Extension to Composite Degradations

Although ROFER can address a single degradation in the above procedure, as a more difficult and ubiquitous situation in reality, many images are affected by composite degradations. One strategy to extend ROFER to composite degradations is iterating the procedure for a single degradation and rectifying the features from the degradation with the biggest p'_{i} of P' in Eq.(3) in every step. In this way, training on compositely degraded images can help ROFER predict degradations accurately in every step to rectify features according to the composition order. However, this training paradigm requires an extremely large-scale dataset since degradations can compound in numerous orders. Instead, ROFER only requires images affected by a single degradation for training. However, only learning from single degradation, ROFER is ignorant about the change of properties of each degradation when they compound, resulting in inaccurate predictions and rectification orders during testing. For instance, if an image x is affected by Gaussian blur first and low contrast then, ROFER can incorrectly rectify features from Gaussian blur first and low contrast then.

Therefore, a modified beam search algorithm, which explores multiple promising directions in every step during searching, is adopted to find the composition order of degradations. In each step, one degraded feature is rectified from several degradations respectively, producing multiple rectified results. Every rectified feature repeats the procedure until all features are judged by the module to be clear enough and the best-rectified feature is selected for further process. In this way, it is more effective to find the composition order and obtain the feature that is rectified accordingly.

Optimization Objective

During training, assuming that a clear image x is affected by $deg^{(i)}$ to become x', four optimization objectives are proposed to update ROFER.

Classification Loss With "Type Classification" predicting which degradation is affecting x'_f, the first loss $\mathcal{L}_{\text{cls}}$ optimizes it to make accurate predictions:

$$\mathcal{L}_{\text{cls}} = \sum_{\text{k}=0}^{n} [-p_{\text{k}} \cdot \log(p'_{\text{k}}) - (1 - p_{\text{k}}) \cdot \log(1 - p'_{\text{k}})], \quad (6)$$

where p_{k} indicates the truth whether x'_f is affected by $deg^{(k)}$ and equals 0, except p_{i} which equals 1. $\mathcal{L}_{\text{cls}}$ calculates binary cross entropy loss (BCELoss) between p_{k} and p'_{k}.

Regression Loss With "Intensity Regression" predicting the intensity of degradations, the second loss $\mathcal{L}_{\text{reg}}$ helps its predictions to be as close to groud-truth as possible:

$$\mathcal{L}_{\text{reg}} = \sum_{\text{k}=1}^{n} 1(\text{k} = \text{i}) \cdot |q_{\text{k}} - q'_{\text{k}}|, \quad (7)$$

where q_{k} indicates the real intensity of the degradation $deg^{(k)}$ and only q_{i} is not 0. The function $1(\cdot)$ returns 1 if the input condition is satisfied and 0 else, indicating that the prediction loss calculated for $deg^{(k)}(\text{k} \neq \text{i})$ is neglected. $\mathcal{L}_{\text{reg}}$ calculates absolute difference between q_{k} and q'_{k}.

Rectification Loss With the FCN rectifying features from degradations, the third loss $\mathcal{L}_{\text{rec}}$ makes the rectified features element-wise closer to clear features:

$$\mathcal{L}_{\text{rec}} = \sum_{\text{k}=0}^{T} (x^{\text{rec}}_{f,k} - x_{f,\text{k}})^2 - \alpha \cdot cos(x^{\text{rec}}_f, x_f), \quad (8)$$

where T is the total element number of x^{rec}_f (the same as x_f). cos calculates the cosine similarity and α is a scaling hyperparameter. $\mathcal{L}_{\text{rec}}$ calculates the element-wise square difference and cosine similarity between x^{rec}_f and x_f.

Total Loss Finally, we aggregate all aforementioned optimization objectives to form the total loss $\mathcal{L}$ to optimize ROFER, while keeping the recognition model fixed:

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{reg}}. \quad (9)$$

Experiments

Experimental Setup

Datasets and Backbones CIFAR-10 combined with CIFAR-100 and Tiny-ImageNet combined with CUB-200-2011 are selected datasets for evaluation. When testing on CIFAR-10 (or Tiny-ImageNet), only CIFAR-100 (or CUB-200-2011) is used to train ROFER. Following the generation method in (Hendrycks and Dietterich 2019), every dataset has clear and generated degraded images, where there are five levels of intensity for each degradation. The higher the intensity level, the more severe the degradation. AlexNet, VGG19, and ResNet-18 are selected backbones, which are pretrained only on clear images in the testing dataset.

Implementation Details Recognition accuracy is selected as the main evaluation metric. In all experimental settings, the baseline is the pretrained backbone CNN without modifications. The training batch size is 100 and the optimizer is SGD with 0.01 as the learning rate, 0.9 as the momentum. The training epoch is 30 and the scaling parameter α in Eq.(8) is set to be 10. The first max-pooling layer's outputs of backbones are the rectified features of ROFER. Training and testing are finished on a single TITAN RTX GPU. Codes are implemented in Python of 3.6.2 and Pytorch of 1.7.1.

Single Degradation

As a basic situation, methods are compared to see whether they can address a single degradation. Three types of degradations, including Gaussian noise, Gaussian blur, and low contrast are considered since they are among the most common degradations. We select eight image enhancement methods (SRN-Deblur (Tao et al. 2018), MIMO-UNet (Cho et al. 2021), MPRNet (Zamir et al. 2021), HE (Gonzalez and Woods 2008), LDR (Lee, Lee, and Kim 2013), WAHE (Arici, Dikbas, and Altunbasak 2009), SADNet (Chang et al. 2020), DnCNN (Zhang et al. 2017)), two domain adaptation methods (ENT (Rusak et al. 2021), BNA (Schneider et al. 2020)) and one feature restoration method (FD-Module (Wang et al. 2020)) as comparison.

Intensity levels of one degradation are 2 and 4 during training so that tested intensities (1,2,3,4,5) are lower than, in the middle of, the same as or higher than them to evaluate the generalization to different intensity conditions. Compared methods are trained on one degradation at a time, while ROFER is trained on three degradations at the same time to prove its ability to address various degradations simultaneously. Results on CIFAR-10 are illustrated in Figure 4. The results reveal that ROFER can address various degradations simultaneously and outperforms compared methods.

Composite Degradations

Compared with single degradation, composite degradations are more difficult and ubiquitous in reality. Therefore, meth-

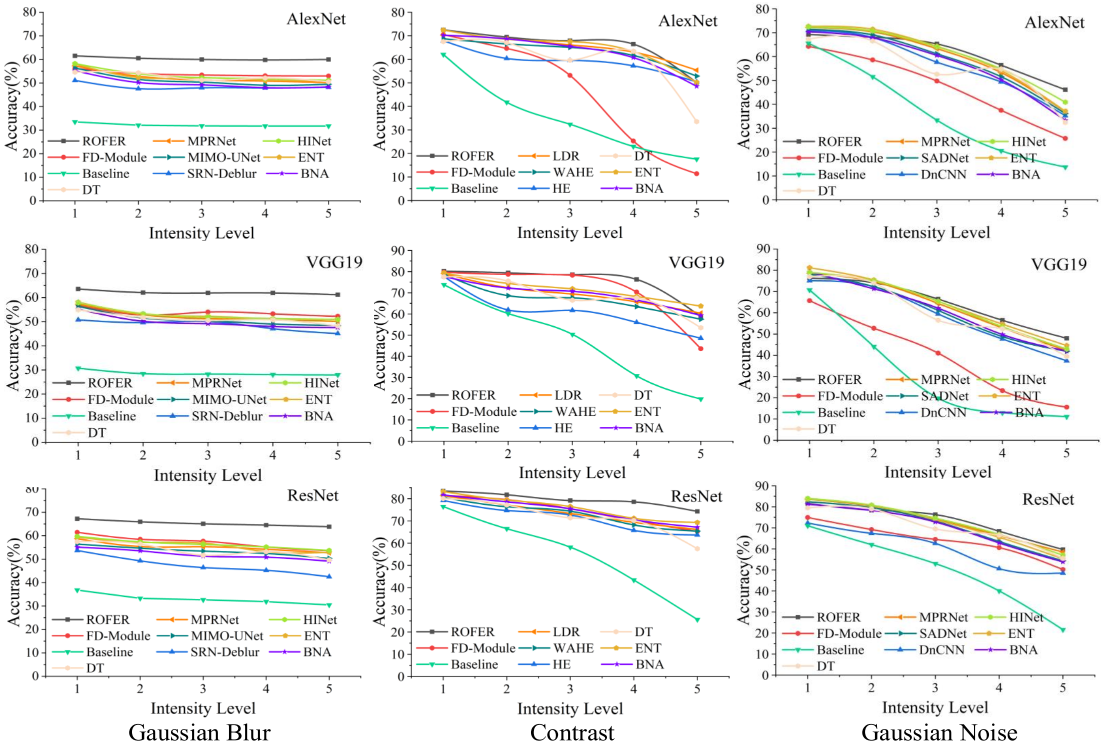


Figure 4: Accuracy comparison on CIFAR-10 for a single type of degradation. “DT” represents directly training the backbone on the degraded images.

ods are compared when there are composite degradations. To lower the influence of untrained intensities on compared methods when evaluating the generalization, the intensity of degradation during training are 2 and 5 while during testing, it is 3 or 4.

Comparison with Feature Restoration First, ROFER is compared with Feature Dedrifting Module (FD-Module) which maps degraded features to clear features (Wang et al. 2020). Considered degradation types are the same as those in Section “Single Degradation”, plus motion blur and impulse noise. One degraded image in the training dataset is affected by a single degradation. In the testing dataset, one image is affected by a random number of degradations ranging from 1 to 5, in random order. In every step of beam search, ROFER rectifies one feature from two degradations respectively. Results are listed in Table 1. The results give that ROFER improves the accuracy of backbones in all situations, even when 5 degradations compound. By contrast, FD-Module even decreases the accuracy of backbones.

Comparison with Image Enhancement and Domain Adaptation ROFER is furtherly compared with an image enhancement method proposed for composite degradations (OWANet (Suganuma, Liu, and Okatani 2019)) and domain adaptation method (ENT (Rusak et al. 2021)). Gaussian blur, Gaussian noise and JPEG compression are considered degradations as in (Suganuma, Liu, and Okatani 2019). ROFER is trained on singly degraded images while other methods are trained on compositely degraded images. Results are illustrated in Table 2. The results reveal that ROFER obtains better performance than image enhancement and domain adaptation methods.

Ablation Study

To help with feature rectification, techniques including the modified beam search algorithm and “Degradation Identification” are adopted. In this section, some experiments are conducted to validate their effect.

Effect of Beam Search To help ROFER find the composition order of degradations, the beam search algorithm is adopted. In Section “Composite Degradation”, we fix the setting that ROFER rectifies one feature from two degradations respectively in every step of beam search. Here, the performance is compared to see whether it drops when the beam search is deleted and thus one feature is rectified only from the most possible degradation in every step. The training and testing datasets are the same as those employed in Section “Composite Degradation”, “Comparison with Feature Restoration” part. Results on images affected by no more than three degradations are illustrated in Table 3. The results reveal that without beam search, ROFER suffers a performance drop. The reason is that when one feature is rectified from only one degradation in every step, an incorrect step of rectification directly fails the final result.

Another experiment is conducted to see how many percentages of finally selected features are rectified according

Deg Num	Methods	CIFAR-10			Tiny-ImageNet		
		AlexNet	VGG19	ResNet	AlexNet	VGG19	ResNet
$\leq 2/=2$	Baseline	27.3/21.4	24.6/18.7	30.2/24.6	5.5/3.0	4.6/1.8	6.8/4.4
	FD-Module	28.2/20.7	25.7/19.7	31.4/23.9	5.9/3.1	5.0/1.8	7.1/4.3
	ROFER	**40.3/33.6**	**38.7/29.4**	**43.5/37.2**	**11.2/7.1**	**11.1/4.6**	**12.2/8.1**
$\leq 3/=3$	Baseline	17.3/14.8	16.7/14.5	17.8/15.1	2.5/1.2	2.4/1.3	2.7/1.6
	FD-Module	17.5/13.6	16.4/12.2	18.0/14.9	2.6/1.3	2.6/1.2	2.7/1.6
	ROFER	**29.4/24.3**	**26.3/22.7**	**29.6/25.7**	**5.0/3.9**	**5.1/4.0**	**5.3/4.5**
$\leq 5/=5$	Baseline	11.6/11.3	11.1/11.0	11.3/11.0	0.7/0.7	0.6/0.5	0.7/0.6
	FD-Module	11.7/11.4	11.2/10.6	11.4/10.5	0.8/0.6	0.6/0.6	0.7/0.6
	ROFER	**18.1/16.8**	**17.7/16.0**	**18.7/15.5**	**1.3/1.1**	**1.1/1.0**	**1.2/1.1**

Table 1: Accuracy comparison between FD-Module and ROFER on images affected by a certain number of degradations.

Deg Num	Methods	CIFAR-10		Tiny-ImageNet	
		Alex	VGG	Alex	VGG
≤ 3	OWANet	41.5	36.8	5.3	4.3
	ENT	42.4	38.0	5.4	4.4
	ROFER	**43.6**	**39.3**	**5.8**	**4.9**
$=3$	OWANet	35.6	30.5	4.9	3.6
	ENT	36.4	**31.2**	4.9	3.7
	ROFER	**37.2**	30.7	**5.3**	**4.2**

Table 2: Comparison between OWANet, ENT and ROFER on images affected by no more than 3 degradations. “Alex” and “VGG” represent AlexNet and VGG19 respectively.

Deg Num	Methods	CIFAR-10		Tiny-ImageNet	
		Alex	VGG	Alex	VGG
≤ 3	Baseline	17.3	16.7	2.5	2.4
	ROFER	**29.4**	**26.3**	**5.0**	**5.1**
	ROFER(no BS)	26.1	23.7	25.7	4.4
$=3$	Baseline	14.8	14.5	1.2	1.3
	ROFER	**24.3**	**22.7**	**3.9**	**4.0**
	ROFER(no BS)	21.5	17.3	3.0	3.3

Table 3: Accuracy analysis to see whether the beam search (BS) improves the performance on images affected by no more than 3 degradations. “Alex” and “VGG” represent AlexNet and VGG19 respectively.

Deg Num	Methods	CIFAR-10		Tiny-ImageNet	
		Alex	VGG	Alex	VGG
≤ 3	ROFER	15.0	**25.8**	**20.6**	**25.2**
	ROFER(no BS)	**19.9**	18.5	16.7	21.7
$=3$	ROFER	**12.3**	**25.2**	**19.9**	**23.4**
	ROFER(no BS)	9.4	17.4	15.3	18.4

Table 4: Percentage analysis of how many selected features are rectified according to the composition order on images affected by no more than 3 degradations. “Alex” and “VGG” represent AlexNet and VGG19 respectively.

Deg Num	Methods	CIFAR-10		Tiny-ImageNet	
		Alex	VGG	Alex	VGG
≤ 3	Baseline	17.3	16.7	2.5	2.4
	ROFER	**29.4**	**26.3**	**5.0**	**5.1**
	ROFER(no ID)	21.0	20.3	3.2	3.5
$=3$	Baseline	14.8	14.5	1.2	1.3
	ROFER	**24.3**	**22.7**	**3.9**	**4.0**
	ROFER(no ID)	17.4	15.1	2.0	2.2

Table 5: Accuracy analysis to see whether the “Degradation Identification” (ID) stage affects on images affected by no more than 3 degradations. “Alex”,“VGG” represent AlexNet and VGG19 respectively.

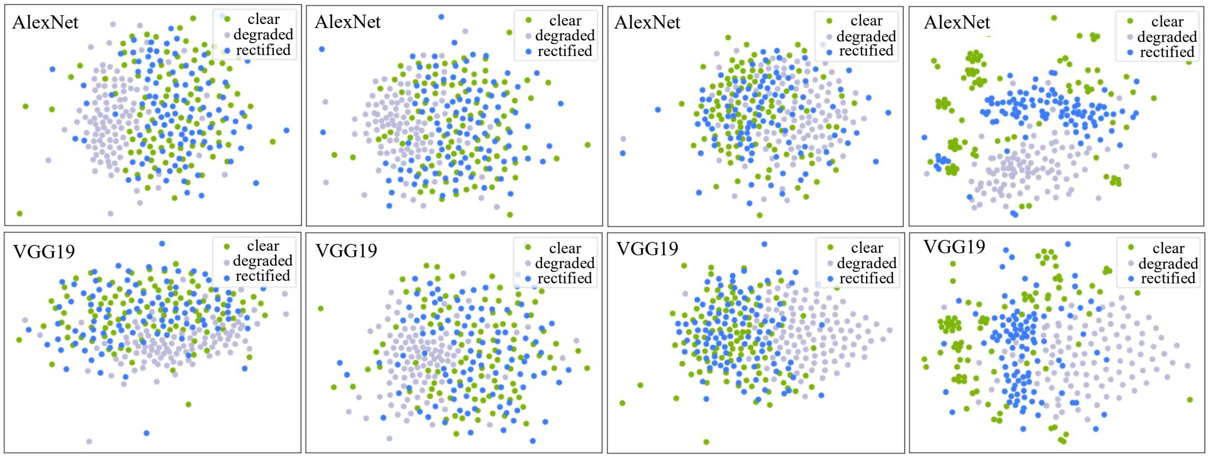
to the composition order when the beam search is adopted. The training and testing datasets are the same as those employed in Section “Composite Degradation”, “Comparison with Feature Restoration” part. Results on images affected by no more than three degradations are illustrated in Table 4. The results reveal that with beam search, it is more possible for ROFER to find the composition order of degradations and correctly rectified features.

Effect of Degradation Identification The “Degradation Identification” stage (ID) is designed to predict degradations, allowing ROFER to address them individually. Performance is compared to see whether it drops if ID is deleted. In this situation, ROFER cannot predict degradations and addresses different degradations in the same way. Without predictions, it cannot address composite degradations in an iterative manner but in one step. Experiments are conducted on composite degradations and the training and testing datasets are the same as those in Section “Composite Degradation”, “Comparison with Feature Restoration” part. Results on three degradations in Table 5 give that the performance drops if the ID stage is deleted.

Discussion

Performance Stability on Clear Images

In this part, experiments are conducted to see whether ROFER influences the performance of backbones on clear images. FD-Module is a comparison since it is another plug-and-play module based on feature restoration. The training dataset is the same as those employed in Section “Composite



Gaussian Blur   Contrast   Gaussian Noise   Composite

Figure 5: Illustration of feature distribution with T-SNE (van der Maaten and Hinton 2008). Some feature points are collected from CIFAR-10 with AlexNet or VGG19 as backbones and those in green, gray and blue are the clear, degraded and rectified features respectively. It is revealed that the rectified features are closer to clear features and their distributions are more similar.

Methods	CIFAR-10			Tiny-ImageNet		
	Alex	VGG	Res	Alex	VGG	Res
Baseline	76.2	83.3	88.4	30.6	46.6	54.8
FD-Module	68.0	78.0	82.3	17.4	29.0	38.7
ROFER	76.0	82.0	88.0	24.4	41.9	51.4

Table 6: Accuracy comparison between FD-Module and ROFER on clear images. “Alex”, “VGG” and “Res” represent AlexNet, VGG19 and ResNet backbones respectively.

Feature Level	CIFAR-10			Tiny-ImageNet		
	Blur	Cont	Noise	Blur	Cont	Noise
Mid	55.2	70.4	60.1	18.5	11.2	24.5
Deep	48.5	67.0	54.2	15.2	9.8	21.8
Low	**62.6**	**73.3**	**66.8**	**21.0**	**15.0**	**26.9**

Table 7: Accuracy comparison between selecting different levels of the feature of AlexNet to rectify.

Degradation”, “Comparison with Feature Restoration” part, while testing images are clear. Results are illustrated in Table 6. From the results, the performance influence is minor with ROFER while it decreases a lot with FD-Module.

Visualization of Rectified Features

In this part, some visual results are illustrated to prove the effectiveness of ROFER. Some clear, degraded and rectified features of images in CIFAR-10 are collected with AlexNet and VGG19 as backbones. T-SNE (van der Maaten and Hinton 2008) is implemented to visualize the feature distribution and results are illustrated in Figure 5. Points in green, gray and blue are the clear, degraded and rectified features respectively. From the results, the rectified features are closer to clear features and their distributions are more similar, regardless of the degradation and the backbone. Therefore, ROFER is effective to rectify the degraded features and pull them back to clear features.

Feature Selection for Rectification

For all backbones, the output feature of the first max-pooling layer is selected to be rectified by ROFER. Experiments on single degradation are conducted to see whether the performance is better if selecting other features to rectify. AlexNet is selected as the backbone, where there are 5 convolutional layers. Outputs of the first, third and last convolutional layer (all with max-pooling if there exists) are low-, middle- and deep-level features to be rectified. Degradations are the same as those in Section “Single Degradation” and the intensity level of one degradation during training and testing is 3. Results are illustrated in Table 7. The results reveal that ROFER obtains the best performance when rectifying the low-level features.

Conclusion

In this paper, we propose a RObust FEature Rectification module (ROFER) to improve the performance of pretrained CNNs against degradations for object recognition. Specifically, it rectifies the image features, rather than the visual quality, from degradations by pulling them back to clear features. ROFER is a general-purpose module that can address various degradations simultaneously. Besides, it can be plugged into pretrained CNNs seamlessly. By adopting a beam search to find the composition order, it is furtherly extended to address composite degradations, only requiring unlabeled images affected by a single degradation for training. Extensive experiments demonstrate that ROFER outperforms existing methods for recognition with single or composite degradations. Inspired by the merits of multi-scale reconstruction for image enhancement, in the future, we plan to select multiple features of a backbone and rectify them simultaneously, to explore and widen the potential of ROFER.

Acknowledgments

This research was supported by the National Key Research and Development Program of China under Grant No. 2018AAA0100400, the National Natural Science Foundation of China under Grants 61976208, 62076242, and the InnoHK project.

References

Arici, T.; Dikbas, S.; and Altunbasak, Y. 2009. A histogram modification framework and its application for image contrast enhancement. *IEEE Transactions on image processing*, 18(9): 1921–1935.

Bousmalis, K.; Silberman, N.; Dohan, D.; Erhan, D.; and Krishnan, D. 2017. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3722–3731.

Brock, A.; De, S.; Smith, S. L.; and Simonyan, K. 2021. High-performance large-scale image recognition without normalization. In *International Conference on Machine Learning*, 1059–1071. PMLR.

Chang, M.; Li, Q.; Feng, H.; and Xu, Z. 2020. Spatial-adaptive network for single image denoising. In *European Conference on Computer Vision*, 171–187. Springer.

Cho, S.-J.; Ji, S.-W.; Hong, J.-P.; Jung, S.-W.; and Ko, S.-J. 2021. Rethinking coarse-to-fine approach in single image deblurring. In *Proceedings of the IEEE/CVF international conference on computer vision*, 4641–4650.

Cui, S.; Wang, S.; Zhuo, J.; Su, C.; Huang, Q.; and Tian, Q. 2020. Gradually vanishing bridge for adversarial domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12455–12464.

Dai, D.; Wang, Y.; Chen, Y.; and Van Gool, L. 2016. Is image super-resolution helpful for other vision tasks? In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1–9. IEEE.

Dodge, S.; and Karam, L. 2016. Understanding how image quality affects deep neural networks. In *Eighth International Conference on Quality of Multimedia Experience (QoMEX)*, 1–6.

Gonzalez, R. C.; and Woods, R. E. 2008. *Digital image processing*. Upper Saddle River, N.J.: Prentice Hall.

Guo, S.; Yan, Z.; Zhang, K.; Zuo, W.; and Zhang, L. 2019. Toward Convolutional Blind Denoising of Real Photographs. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 1712–1722.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.

Hendrycks, D.; and Dietterich, T. 2019. Benchmarking Neural Network Robustness to Common Corruptions and Perturbations. *Proceedings of the International Conference on Learning Representations*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, 1097–1105.

Kupyn, O.; Budzan, V.; Mykhailych, M.; Mishkin, D.; and Matas, J. 2018. DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 8183–8192.

Lee, C.; Lee, C.; and Kim, C.-S. 2013. Contrast enhancement based on layered difference representation of 2D histograms. *IEEE transactions on image processing*, 22(12): 5372–5384.

Li, B.; Peng, X.; Wang, Z.; Xu, J.; and Feng, D. 2017. AOD-Net: All-in-One Dehazing Network. In *IEEE International Conference on Computer Vision (ICCV)*, 4780–4788.

Motiian, S.; Piccirilli, M.; Adjeroh, D. A.; and Doretto, G. 2017. Unified deep supervised domain adaptation and generalization. In *Proceedings of the IEEE international conference on computer vision*, 5715–5725.

Pei, Y.; Huang, Y.; Zou, Q.; Zhang, X.; and Wang, S. 2021. Effects of Image Degradation and Degradation Removal to CNN-Based Image Classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(4): 1239–1253.

Rusak, E.; Schneider, S.; Gehler, P.; Bringmann, O.; Brendel, W.; and Bethge, M. 2021. Adapting ImageNet-scale models to complex distribution shifts with self-learning. *arXiv preprint arXiv:2104.12928*.

Russo, P.; Carlucci, F. M.; Tommasi, T.; and Caputo, B. 2018. From source to target and back: symmetric bi-directional adaptive gan. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8099–8108.

Sankaranarayanan, S.; Balaji, Y.; Castillo, C. D.; and Chellappa, R. 2018. Generate to adapt: Aligning domains using generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8503–8512.

Schneider, S.; Rusak, E.; Eck, L.; Bringmann, O.; Brendel, W.; and Bethge, M. 2020. Improving robustness against common corruptions by covariate shift adaptation. *Advances in Neural Information Processing Systems*, 33: 11539–11551.

Simonyan, K.; and Zisserman, A. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations*.

Stojanov, P.; Li, Z.; Gong, M.; Cai, R.; Carbonell, J.; and Zhang, K. 2021. Domain adaptation with invariant representation learning: What transformations to learn? *Advances in Neural Information Processing Systems*, 34: 24791–24803.

Suganuma, M.; Liu, X.; and Okatani, T. 2019. Attention-Based Adaptive Selection of Operations for Image Restoration in the Presence of Unknown Combined Distortions. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 9031–9040.

Sun, Z.; Ozay, M.; Zhang, Y.; Liu, X.; and Okatani, T. 2018. Feature Quantization for Defending Against Distortion of Images. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 7957–7966.

Tan, W.; Yan, B.; and Bare, B. 2018. Feature Super-Resolution: Make Machine See More Clearly. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 3994–4002.

Tao, X.; Gao, H.; Shen, X.; Wang, J.; and Jia, J. 2018. Scale-recurrent Network for Deep Image Deblurring. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 8174–8182.

Tran, P.; Tran, A. T.; Phung, Q.; and Hoai, M. 2021. Explore image deblurring via encoded blur kernel space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11956–11965.

Tu, Z.; Talebi, H.; Zhang, H.; Yang, F.; Milanfar, P.; Bovik, A.; and Li, Y. 2022. Maxim: Multi-axis mlp for image processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5769–5780.

van der Maaten, L.; and Hinton, G. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86): 2579–2605.

Wang, W.; Zhang, J.; Zhai, W.; Cao, Y.; and Tao, D. 2022. Robust Object Detection via Adversarial Novel Style Exploration. *IEEE Transactions on Image Processing*, 31: 1949–1962.

Wang, Y.; Cao, Y.; Zha, Z. J.; Zhang, J.; and Xiong, Z. 2020. Deep Degradation Prior for Low-Quality Image Classification. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 11046–11055.

Xu, X.; Wei, P.; Chen, W.; Liu, Y.; Mao, M.; Lin, L.; and Li, G. 2022. Dual Adversarial Adaptation for Cross-Device Real-World Image Super-Resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5667–5676.

Zamir, S. W.; Arora, A.; Khan, S.; Hayat, M.; Khan, F. S.; Yang, M.-H.; and Shao, L. 2021. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14821–14831.

Zhang, J.; Cao, Y.; Zha, Z.-J.; and Tao, D. 2020. Nighttime dehazing with a synthetic benchmark. In *Proceedings of the 28th ACM international conference on multimedia*, 2355–2363.

Zhang, K.; Zuo, W.; Chen, Y.; Meng, D.; and Zhang, L. 2017. Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising. *IEEE Transactions on Image Processing*, 26(7): 3142–3155.

Zhang, K.; Zuo, W.; and Zhang, L. 2018. FFDNet: Toward a Fast and Flexible Solution for CNN-Based Image Denoising. *IEEE Transactions on Image Processing*, 27(9): 4608–4622.

Zhou, Y.; Xie, H.; Fang, S.; Li, Y.; and Zhang, Y. 2020. CR-Net: A center-aware representation for detecting text of arbitrary shapes. In *Proceedings of the 28th ACM international conference on multimedia*, 2571–2580.