Vector Quantization with Self-Attention for Quality-Independent Representation Learning

Zhou Yang[1] Weisheng Dong [1*] Xin Li[2] Mengluan Huang[1] Yulin Sun[1] Guangming Shi[1]
[1]Xidian University [2]West Virginia University
{yang_zhou,mlhuang}@stu.xidian.edu.cn {wsdong,gmshi}@mail.xidian.edu.cn
xin.li@mail.wvu.edu daitusun@gmail.com

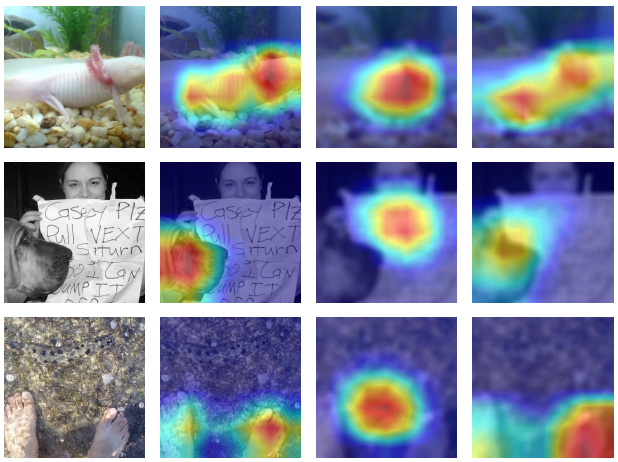
Abstract

Recently, the robustness of deep neural networks has drawn extensive attention due to the potential distribution shift between training and testing data (e.g., deep models trained on high-quality images are sensitive to corruption during testing). Many researchers attempt to make the model learn invariant representations from multiple corrupted data through data augmentation or image-pair-based feature distillation to improve the robustness. Inspired by sparse representation in image restoration, we opt to address this issue by learning image-quality-independent feature representation in a simple plug-and-play manner, that is, to introduce discrete vector quantization (VQ) to remove redundancy in recognition models. Specifically, we first add a codebook module to the network to quantize deep features. Then we concatenate them and design a self-attention module to enhance the representation. During training, we enforce the quantization of features from clean and corrupted images in the same discrete embedding space so that an invariant quality-independent *feature representation can be learned to improve the recognition robustness of low-quality images. Qualitative and quantitative experimental results show that our method achieved this goal effectively, leading to a new state-of-the-art result of 43.1 % mCE on ImageNet-C with ResNet50 as the backbone. On other robustness benchmark datasets, such as ImageNet-R, our method also has an accuracy improvement of almost 2%. The source code is available at https://see.xidian.edu.cn/faculty/wsdong/Projects/VQSA.htm*

1. Introduction

The past few years have witnessed the remarkable development of deep convolutional neural networks (DCNNs) in many recognition tasks, such as classification [9,20,31,46], detection [43, 48] and segmentation [5, 19]. Although its performance has exceeded that of humans in some datasets, its robustness still lags behind [10, 15]. Many researchers [11, 21, 22, 53] have shown that the performance of deep models trained in high-quality data decreases dramatically with low-quality data encountered during deployment, which usually contain common corruptions, including blur, noise, and weather influence. For example, the vanilla ResNet50 model has an accuracy of 76 % in the clean ImageNet validation set, but its average accuracy is less than 30 % in the same dataset contaminated by Gaussian noise.

As shown in Fig. 1, the model's attention map is seriously affected by image quality. That is, the model pays



(a) Image (b) Clean (c) QualNet (d) Ours

Figure 1. The Grad-CAM [45] maps of different models on defocus blur images. (a) The clean images. (b) The maps of vanilla ResNet50 [20] model on clean images. (c) and (d) show the maps of QualNet50 [29] and our proposed method on defocus blur images with severity level 3. The results show that our method still can focus on the salient object area without being seriously affected by corruption. **Best viewed in color.**

*Corresponding author.

close attention to an incomplete or incorrect area of the defocus blur image, which results in a wrong prediction. Deep degradation prior (DDP) [55] also reveals that deep representation space degradation is the reason for the decrease in model performance, and finding a mapping between degraded and clean features will certainly benefit downstream classification or detection tasks.

Generally, a simple and effective method is to consider these common corruptions and to use low-quality simulated images for augmented (fine-tuning) training. However, as described in [29], this method does not explicitly map low-quality features to high-quality features; instead, the model tends to learn the average distribution among corruptions, resulting in limited performance improvement. In addition, when the type of corruption is unknown during training, this approach is heavily based on well-designed augmentation strategies to improve generalization and robustness.

To enhance the feature representation of low-quality images, another feasible solution is to use pairs of clean and degraded images for training and align the degraded features with the corresponding clean ones. Both DDP [55] and QualNet [29] use high-quality features extracted from clean images as supervision and improve low-quality features through a distillation-like approach [1, 25, 26, 58]. Although they learn the mapping relationship between low- and high-quality feature representation, paired images are time-consuming for training, and some irrelevant features that are not useful for recognition are also forcibly aligned, such as background, color, lighting, and other features which are affected by the image quality.

The vector quantization (VQ) process is essentially a special case in sparse representation, that is, the representation coefficient is a one-hot vector [50]. Similarly to the application of sparse representation in image restoration [13, 59], the compression-reconstruction learning framework of VQ is also conducive to the removal of noisy redundant information and learning essential features for recognition. Inspired by this, we propose to use VQ to bridge the gap between low- and high-quality features and learn a quality-independent representation. Specifically, we first add a vector quantizer (codebook) to the recognition model. During the training process, due to the codebook update mechanism, both low- and high-quality features will be assigned to the same discrete embedding space. Subsequently, since direct hard quantization (selecting and replacing) may lose some useful information, we choose to concatenate the quantized features with the original ones and use a self-attention module to further enhance the quality-independent representation and weaken irrelevant features. In summary, the main contributions of this paper are listed below.

- To the best of our knowledge, it is the first time that we propose to introduce vector quantization into the recognition model for learning quality-independent feature representation and improving the models' robustness on common corruptions.

- We concatenate the quantized feature vector with the original one and use the self-attention module to enhance the quality-independent feature representation instead of direct replacement in the standard vector quantization method.

- Extensive experimental results show that our method has achieved higher accuracy on benchmark low-quality datasets than several current state-of-the-art methods.

2. Related Works

2.1. Low-quality Image Recognition

There are many solutions to the recognition of low-quality images. Data augmentation is a simple way to build a robust model by generating diverse data with some well-designed augmentation strategies and learning an invariant representation. AutoAugment [8] is the first method to use reinforcement learning to find the optimal data augmentation strategy. DeepAugment [21] used a model similar to the generative adversarial network (GAN) to generate augmented images. Augmix [23] randomly performed different data augmentations on images and then mixed them to form the final augmented output.

Another common practice is to restore low-quality images first, which means that the parameters of the recognition model are standard and fixed. There are many image restoration algorithms [4, 6, 60]. However, as described in [39], the use of dehazing methods to restore the haze image does not help improve classification performance. The restored image and the high-quality image may still have differences in feature space. Recently, a recognition-friendly restoration method has been proposed for low-quality images. [32] first proposed the solution to image denoising and classification simultaneously. URIE [47] proposed a universal image enhancement module and introduced cross-entropy loss to train classification and image restoration jointly.

Recently, many studies have been done to solve this problem from the perspective of feature representation. They proposed using paired data to learn the feature mapping relationship so that models can extract high-quality-like features even from low-quality images. DDP [55] designed a feature de-drifting module to align low-quality features with the corresponding high-quality ones. QualNet [29] used an invertible neural network as a decoder to transform the paired features into the image domain, and closing the two decoded images aligns the potential features. In addition, many studies [22, 28, 37] also proposed some bench-

marks on various high-level vision tasks for low-quality images.

2.2. Vector Quantization

Sparse representation learning aims to represent input signals well with a sparse coefficient vector from the learned dictionary. The components in the dictionary can be considered as different atoms of signals. Sparse representation learning has many advantages, such as eliminating noise from signals, enhancing the robustness of representation, and being used for compressed sensing. There are many works that apply sparse representation to image denoising [13], super-resolution [59], etc.

Vector quantization is a classic method of compressed coding [16]. It needs a codebook and a quantization strategy. In general, the mean square error (MSE) is used to find the most similar pattern in the codebook to replace the original input data vector. As it is a kind of distortion coding, it requires the codebook to represent the various inputs well, like an overcomplete dictionary. Vector quantization can be considered as discrete representation learning; that is, the representation coefficient is a one-hot vector. Many researchers [7, 16, 34, 38] have shown that learning discrete representation not only contributes to visual understanding but also improves the robustness of models. Recently. VQ-VAE [51] used a neural network named codebook to learn a discrete visual representation of images. This method can learn the discrete feature distribution of an image effectively and is widely used in many generative models. Along this line of research, discrete representation learning has been widely used in many vision tasks [14, 17, 35, 40–42].

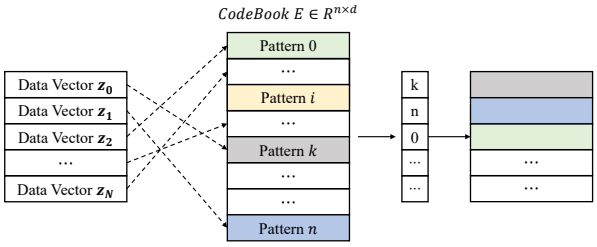


Figure 2. The process of vector quantization in compressed coding. Noted that $\boldsymbol{z_i} \in R^d$.

2.3. Self-attention

The attention mechanism is widely used in deep learning. If all the information in the image is learned without any priority, the model tends to overfit, resulting in poor generalization ability in new test data. Therefore, especially for recognition tasks, researchers hope that the model can pay more attention to the salient object itself in the image while ignoring the background and other irrelevant information. SE-Net [27] proposed a channel-wise attention module, and CBAM [56] combined channel attention with spatial attention in a convolutional block attention module. These methods are widely used in the design of model structures.

The self-attention module in the transformer was first proposed in [54] in natural language processing (NLP). It does not rely on the traditional recurrent neural networks (RNNs) or DCNNs architecture but can perform many NLP tasks such as machine translation excellently. Inspired by its success, many works [12, 18, 33, 49] introduced this architecture in computer vision tasks, called Vision Transformers (ViTs). This makes the performance of machine learning in many vision tasks continue to reach new levels [3, 18, 33]. And researches also revealed that the self-attention module can strengthen relevant features, ignore local differences, and improve the model robustness [36, 62].

3. Method

In this section, we first introduce some preliminaries and background of low-quality image recognition and vector quantization in Sec. 3.1. Then our proposed vector quantization with self-attention for the quality-independent representation learning method is explained in detail in Sec. 3.2 and 3.3. Finally, Sec. 3.4 introduces the overall architecture and training objective of our method.

3.1. Preliminaries

Low-quality Image Recognition: For the low-quality image recognition problem, given a clean dataset $D = \{(\boldsymbol{x}, y)\}$ with image $\boldsymbol{x}$ and the corresponding label y, we can generate various corrupted images using multiple types of degradation functions $\Phi = \{\varphi_1, \varphi_2, ..., \varphi_n\}$. Therefore, the corrupted image can be formulated as follows.

$$\tilde{\boldsymbol{x}} = \varphi_k(\boldsymbol{x}). \tag{1}$$

Note that the identity function is also contained in the set $\boldsymbol{\Phi}$ to preserve the clean original images. Generally, these images will be sent to the model for augmented training, expecting good generalization performance on various common corruptions.

Suppose that the feature extracted from the backbone network is marked as $\boldsymbol{z}$, we have the cross-entropy loss function as the training objective:

$$L_{ce} = -\log \frac{exp(H(\boldsymbol{z})_c)}{\sum_c exp(H(\boldsymbol{z})_c)}, \tag{2}$$

where N is the number of examples in the mini-batch, $H(\cdot)$ represents the head network, its output is the logit vector, and c is the label of the feature $\boldsymbol{z}$.

Vector Quantization: As shown in Fig. 2, for an input data vector $\boldsymbol{z}$, suppose that we have a learned codebook $E \in$

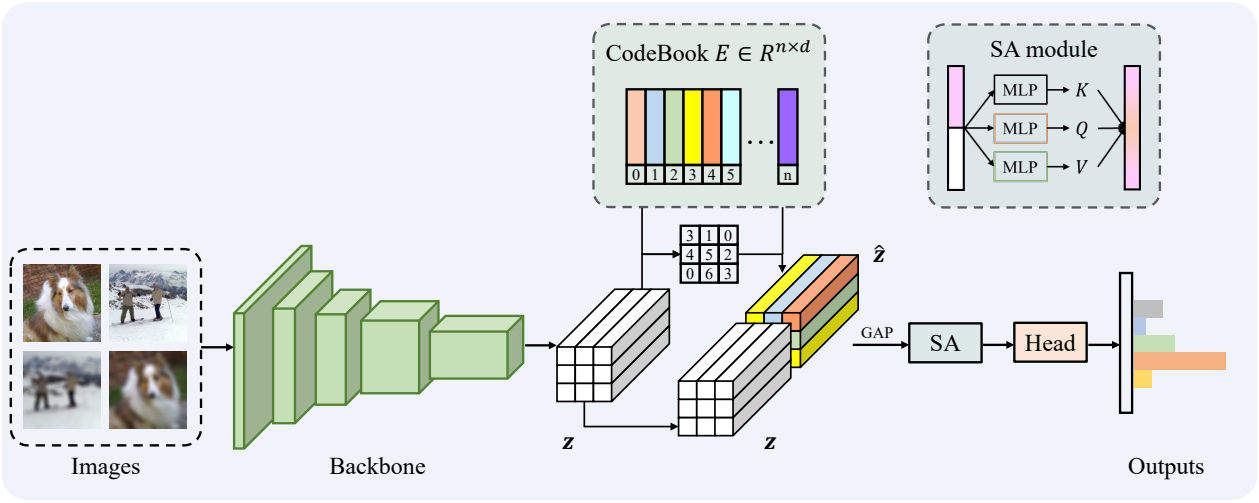


Figure 3. The overall architecture of our proposed method. The mini-batch of input contains both clean and corrupted images. Features extracted from the backbone network are quantized by the codebook module. Then $\boldsymbol{z}$ and $\hat{\boldsymbol{z}}$ are concatenated and pooled. Subsequently, after being enhanced by the SA module, the features are input into the head network to get the final output results.

$R^{n \times d}$ containing multiple patterns, we can obtain its code k using the following formula:

$$\begin{aligned} \hat{\boldsymbol{z}} &= E(\boldsymbol{z}) = \boldsymbol{e_k}, \\ &where \;\; k = arg\min_i ||\boldsymbol{z} - \boldsymbol{e_i}||_2^2, \end{aligned} \tag{3}$$

where n represents the number of vectors in the codebook and d denotes the dimension of each one. $\boldsymbol{e_i}$ is the i-th item in the codebook. When decoding, the vector $\boldsymbol{e_k}$ is extracted from the codebook with the index k to complete VQ, and $\boldsymbol{e_k}$ is usually denoted by $\hat{\boldsymbol{z}}$, that is, the quantized vector.

It is obvious that a good codebook plays a key role in vector quantization. VQ-VAE [51] proposed a variational autoencoder with a learnable codebook neural network module to represent an image. They used a straight-through estimator [2] with VQ loss, commitment loss, and reconstruction loss to train the entire model end-to-end. In this paper, we use the codebook module to represent features, hoping to learn a quality-independent feature for low-quality image recognition.

3.2. Quality-independent Representation Learning

Motivation: As mentioned in Sec. 1 and 2, many researchers have shown that the degradation of features is the cause of the decrease in the recognition performance of low-quality images. Therefore, the key idea of this paper is to make the model learn a quality-independent feature representation. Assume that the quality-independent feature vector $\hat{\boldsymbol{z}}$ of an image is a linear combination of a series of features (atoms):

$$\hat{\boldsymbol{z}} = \sum_{i=0}^{n} \alpha_i \cdot \boldsymbol{e_i} = \alpha_0 \cdot \boldsymbol{e_0} + \alpha_1 \cdot \boldsymbol{e_1} + \cdots + \alpha_n \cdot \boldsymbol{e_n}. \tag{4}$$

If the atoms form an overcomplete space $E \in R^{n \times d}$, it can be sparsely represented by E as:

$$\hat{\boldsymbol{\alpha}} = arg\min_{\boldsymbol{\alpha}} ||\boldsymbol{\alpha}||_0, \;\; s.t. \;\; E \cdot \boldsymbol{\alpha} \approx \hat{\boldsymbol{z}}. \tag{5}$$

For a degraded image, the obtained feature can be considered as the noisy version of $\hat{\boldsymbol{z}}$, that is, $\boldsymbol{z} = \hat{\boldsymbol{z}} + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon}$ observes a Gaussian distribution with mean zero and standard deviation σ. Recovering $\hat{\boldsymbol{z}}$ from $\boldsymbol{z}$ is a maximum *a posteriori* probability (MAP) problem. Through sparse representation, we can solve this problem by the following optimization objective:

$$\hat{\boldsymbol{\alpha}} = arg\min_{\boldsymbol{\alpha}} ||\boldsymbol{z} - E \cdot \boldsymbol{\alpha}||_2^2 + \lambda \cdot ||\boldsymbol{\alpha}||_0. \tag{6}$$

Then the quality-independent feature is given by $\hat{\boldsymbol{z}} = E \cdot \hat{\boldsymbol{\alpha}}$. Similar to sparse representation learning, we can make the model learn the essential feature and eliminate redundant information by constructing an overcomplete feature space and sparse representation coefficients.

Method: But there are two variables to learn in Eq. (6): the overcomplete space E and the coefficient $\boldsymbol{\alpha}$. The conventional sparse dictionary learning update algorithm is complex and end-to-end training in deep neural networks is difficult. Inspired by the discrete vector quantization (VQ), we can directly set the coefficient $\boldsymbol{\alpha}$ as a one-hot vector, so that the latter term in Eq. (6) becomes a constant. Although this operation may cause some distortion of the feature vector, we will improve it later in Sec. 3.3. Replacing $E \cdot \boldsymbol{\alpha}$ with $\hat{\boldsymbol{z}}$ in Eq. (6), we have $min||\boldsymbol{z} - \hat{\boldsymbol{z}}||_2^2$ to learn such a discrete representation space. This optimization objective is the same as the VQ-loss function in VQ-VAE. They use a parameterized module called codebook to learn a discrete

representation space. The VQ-loss function for updating the codebook parameters is as follows:

$$L_{vq} = ||\hat{\boldsymbol{z}} - sg(\boldsymbol{z})||_2^2, \tag{7}$$

where $sg(\cdot)$ represents the stop-gradient operation, $\hat{\boldsymbol{z}}$ is the quantized vector or the quality-independent feature in our method.

Combining Eqs. (7) and (3) as our training objectives, we can use deep neural networks to learn the codebook end-to-end. Therefore, from the above discussion, we choose to introduce the codebook module into our recognition model to learn a quality-independent, essential feature representation through a large number of augmented data containing clean and degraded images, simultaneously.

3.3. Enhance the Representation by Self-attention

In standard vector quantization, it usually replaces the feature $\boldsymbol{z}$ with the quantized vector $\hat{\boldsymbol{z}}$ directly and inputs it into the subsequent network. But, as shown in Eq. (3) and Sec. 2, vector quantization uses the one-hot coding vector, that is, always select the closest vector from the codebook space, which will inevitably lead to the loss of some useful information. The experimental results also show that although $\hat{\boldsymbol{z}}$ contains quality-independent features, performance improvement is limited. And some information may be lost due to the one-hot coding vector, the direct replacement operation, or the incomplete codebook in the early training stage. Based on this consideration, we choose to fuse $\boldsymbol{z}$ and $\hat{\boldsymbol{z}}$ to supplement lost information and further improve representation. Through the results of the carefully designed ablation experiment described in Sec. 4.1, we finally choose to concatenate the quantized vector $\hat{\boldsymbol{z}}$ with the feature $\boldsymbol{z}$.

As described in Sec. 2, for low-quality image recognition task, we hope that the model can pay more attention to the features of the object itself in the image while ignoring the background and other irrelevant information. To enhance the extracted quality-independent feature, we input the fused feature representation $concat(\hat{\boldsymbol{z}}, \boldsymbol{z})$ into a self-attention module. Specifically, we use three MLP networks to generate K, Q, V, respectively. Then the output $\boldsymbol{z_{sa}}$ can be formulated as:

$$\boldsymbol{z_{sa}} = softmax(K \cdot Q^T/\sqrt{2}) \cdot V. \tag{8}$$

Through the self-attention module, the model can adaptively further retain key information and improve the quality-independent representation described in Sec. 3.2. The feature will then be sent to the head network to produce the final classification results.

3.4. Overall Architecture and Training Strategy

The general structure of our approach is shown in Fig. 3. The batch input of the model includes clear images and simulated low-quality degraded images for augmented training. The data are generated by Eq. (1) as described in Sec. 3.1. As described in Sec. 3.1, the similarity between the feature $\boldsymbol{z}$ extracted from the backbone network and the item in the codebook was calculated, and the most similar item is selected as the quantized feature $\hat{\boldsymbol{z}}$. The vector quantization process can be expressed by the formulation Eq. (3). And we use Eq. (7) to optimize the parameters in the codebook module.

In addition to Eq. (7), there is a commitment loss in standard vector quantization as an additional constraint on the encoder (backbone). The commitment loss can be formulated as follows.

$$L_{cmt} = ||\boldsymbol{z} - sg(\hat{\boldsymbol{z}})||_2^2. \tag{9}$$

This loss is used to optimize the parameters in the backbone network and aims to prevent collapse and make the output of the backbone network consistent with the codebook embedding space.

Subsequently, we concatenate $\boldsymbol{z}$ with $\hat{\boldsymbol{z}}$, transform it as a vector through global average pooling (GAP), and then input them into the self-attention module. The output feature can be obtained from Eq. (8). Then the enhanced feature representation passes through two fully connected layers to get the final classification result. The overall training objective of our method consists of three losses, as stated in Eqs. (2), (7) and (9), that is, cross-entropy loss, VQ loss, and commitment loss. Total loss can be formulated as:

$$L_{total} = L_{vq} + \beta \cdot L_{cmt} + \lambda \cdot L_{ce}, \tag{10}$$

where β is the weight of commitment loss, we set $\beta = 0.25$ according to the original setting in [51], λ is the weight of loss balance, we empirically set it at 1. Finally, the model uses the total loss function for end-to-end training.

4. Experiments

Dataset: The ImageNet-C benchmark dataset [22] contains 19 types of common corruptions in 5 severity levels. Among them, 15 corruption types (Gaussian / shot / impulse noise, glass / motion / defocus / zoom blur, contrast, elastic, JPEG, pixelate, frost, fog, snow, and brightness) are in 4 categories for augmented training and 4 corruption types (speckle noise, Gaussian blur, spatter, and saturate) to test the robustness of the model.

In Sec. 4.1 - 4.2, we use the 15 types of degradation function and the identity function as described in Sec. 3.1 to generate the training data. Then we tested the performance on the ImageNet-C validation set. The mean corruption error (mCE) is the metric for low-quality image classification, which can be calculated using the following.

$$mCE = \frac{1}{15}\sum_{c=1}^{15}\frac{\sum_{s=1}^{5} Err_{c,s}}{\sum_{s=1}^{5} Err_{c,s}^{AlexNet}}, \tag{11}$$

where c represents the index of corruption type, s is the severity level. Note that the error is normalized by the vanilla AlexNet error in the same validation set.

In Sec. 4.3, we first compare the robustness of our method with other methods on 4 types of unseen corruption in ImageNet-C through direct testing. Then we adopt the same augmentation strategy and training settings in [21] to train and test the robustness of the distribution shift validation set ImageNet-R [21] and the adversarial dataset ImageNet-A [24]. ImageNet-R contains sketches, art, painting, toys, and other style images, while ImageNet-A provides adversarial examples.

Training Settings: In our method, we used ResNet-50 [20] and ResNeXt-101 [57] as our backbone. For the codebook $E \in R^{n \times d}$, we set $n = 10000$ and $d = 2048$. We employed Adam [30] as the optimizer with an initial learning rate of 0.001. The cosine annealing function was adopted as the learning rate adjustment strategy. The batch size was set at 512 and 4× Nvidia GeForce RTX 3090 were used for training. Noted that some results of the comparison method in this section are reproduced according to their paper.

4.1. Ablation Study

First of all, we conducted several experiments to investigate the impact of each module in our proposed method on the performance of low-quality image recognition. We roughly divide our module into three parts, namely the codebook, the fusion mode, and the self-attention module. Without codebook means, we use the corrupted generated data to fine-tune the vanilla model. For feature fusion, we adopted three different modes, "replace" means directly replacing the original feature vector with the quantized one as standard vector quantization, "add" means that we add the codebook item to the original feature, and the concatenate operation we finally used is denoted by "concat".

The experimental results are shown in Table 1. "clean" and "mCE" represent the top-1 accuracy and mCE results on clean and corrupted images, respectively. From the table, we can obtain that concat is the best mode for feature fusion. The reason may be that the original feature and the quantized item can complement each other with useful information. However, direct replacement or addition will cause the lost or mixture of key information, which may affect subsequent recognition. Moreover, the experimental results also show that each module in our proposed method contributes to the improvement of recognition performance.

We also calculated the parameters and FLOPS of our method for training. Since our method is plug-and-play, the increased parameters and FLOPS are mainly from the codebook and self-attention module. These results are shown in Table 1. We can conclude that, compared to the baseline model and QualNet, the additional parameters and FLOPS required by our method are acceptable. We have also studied the setting of the codebook size n, as detailed in the supplementary material Sec. B.

CodeBook	Fusion mode	SA	clean ↑	mCE ↓
-	-	-	73.1	53.7
✓	replace	-	74.3	50.1
✓	add	-	74.7	48.9
✓	concat	-	76.2	45.7
✓	concat	✓	**76.6**	**43.1**

Table 1. The ablation study of our method on each module. Without codebook means the fine-tuning method. "replace" means directly replace $\boldsymbol{z}$ with $\hat{\boldsymbol{z}}$, "add" represents we add $\hat{\boldsymbol{z}}$ to $\boldsymbol{z}$, and the concatenate operation we finally used is denoted by "concat". **The best result are indicated in bold.**

Model	mCE ↓	# Params.	# FLOPS
Baseline [20]	53.7	$\mathbf{2.5 \times 10^7}$	**4.11G**
QualNet [29]	50.3	1.2×10^8	14.46G
Ours	**43.1**	5.8×10^7	4.13G

Table 2. The mCE value, number of parameters, and the FLOPS of different models for training. Noted that the backbone network of all models is ResNet50.

4.2. Comparison with State-Of-The-Art Methods

In this section, we have compared our proposed method with other state-of-the-art methods to demonstrate the effectiveness of our model in low-quality image recognition. We selected several recent methods, such as DDP [55], URIE [47], and QualNet [29]. The experimental training settings are consistent with these methods. We tested the accuracy (higher is better) and mCE (lower is better) of the ImageNet-C validation set on two backbone network architectures, ResNet-50 and ResNeXt-101.

Method	Backbone	Clean	Known	UnKnown	mCE ↓
Vanilla [20]	ResNet50	76.1	39.1	46.7	76.7
DDP [55]		72.1	48.2	50.7	62.78
URIE [47]		73.8	55.1	56.5	55.7
QualNet [29]		75.4	61.1	58.1	50.3
Ours		**76.6**	**65.6**	**60.2**	**43.1**
Vanilla [20]	ResNeXt101	79.6	47.1	55.5	69.7
QualNet [29]		77.8	65.5	63.3	42.6
Ours		**80.3**	**68.6**	**64.5**	**37.9**

Table 3. The average top-1 accuracy of clean images, 15 types of corrupted images that are known in training, 4 types of unknown corrupted images, and the mCE value over ImageNet-C. We compared our method with others in ResNet50 and ResNeXt101 backbone network. **The best results are indicated in bold.**

As shown in Table 3, "clean", "known", and "unknown" represent the average top-1 accuracy on clean images, images with 15 types of corruption that are known during

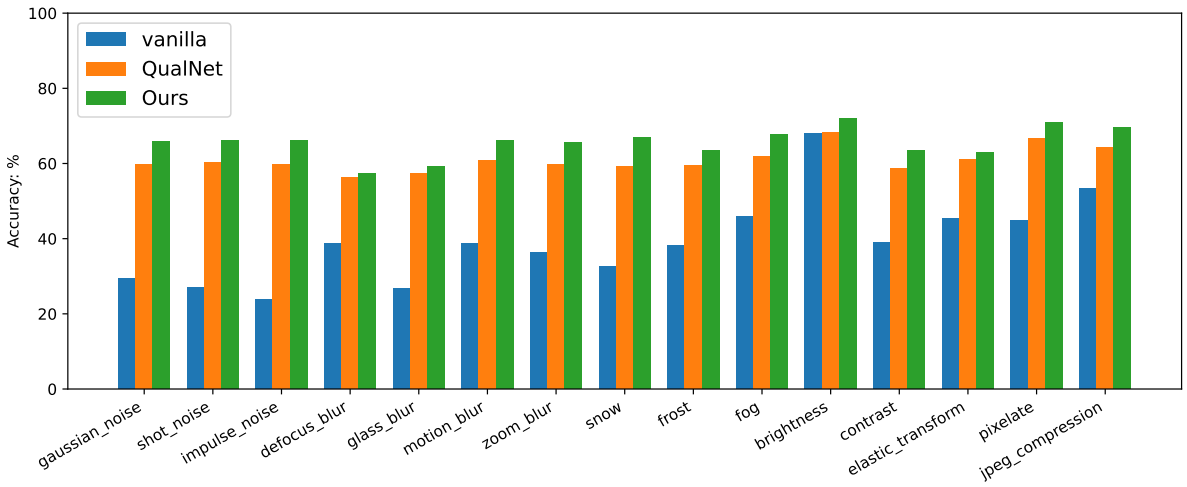


Figure 4. The detailed top-1 accuracy results of the different methods for each corruption type in benchmark dataset ImageNet-C. **Best viewed in color.**

training, and 4 types of unknown corruption, respectively. "mCE" is the mean Corruption Error which is already normalized as described at the beginning of Sec. 4. The detailed top-1 accuracy of the 15 types of corruption can be seen in Fig. 4. On the basis of the above results, we find that not only does our method perform better on low-quality images but also improves the accuracy of clean images, from which we can verify the superiority of our method.

4.3. Model Robustness

In this section, we tested the robustness of our proposed method. First, we applied our model directly to four types of corruption (speckle noise / Gaussian blur / spatter / saturation) that were unknown during ImageNet-C training [22]. All comparison methods were conducted in the same training settings as us. From the results shown in Table 4, we can see that our method has better recognition accuracy on low-quality images than other methods.

Method	speckle noise	gaussian blur	spatter	saturate
Vanilla [20]	35.5	49.2	41.9	60.2
AugMix [23]	50.6	47.2	53.3	61.5
ANT [44]	58.1	43.1	52.4	61.3
DeepAugment [21]	61.1	52.1	53.8	61.5
QualNet [29]	63.1	50.5	54.0	62.3
Ours	**68.4**	**54.1**	**55.7**	**62.8**

Table 4. The detailed average top-1 accuracy results on 4 types of unknown corruptions during training. This group of experiments is designed to test the robustness of the model. **The best results are indicated in bold.**

To further explore the robustness of our method, we also performed experiments on another common benchmark dataset ImageNet-R [21], ImageNet-A [24]. The ImageNet-R dataset contains multiple styles of images, such as sketch, toy, and painting, and is used for testing the model's generalization ability on distribution shift data. ImageNet-A generates the corresponding adversarial samples for the images in the ImageNet validation set. Due to the plug-and-play property of our method, in these experiments, we combined our method with others to investigate whether it can continue to bring gains. It is worth noting that in these experiments we just use the same augmented training strategy as the baseline model without using any corrupted images. The experimental results are shown in Table 5. It is obvious that the performance of these distribution shift data is further improved after combining our methods. The result of our method on ImageNet-A is slightly less than that of DAT [35] because it used adversarial training.

Method	Clean	ImageNet-C ↓	ImageNet-A	ImageNet-R
Vanilla [20]	76.1	76.7	0.0	36.2
+ Ours	**76.6**	**71.1**	**3.7**	**38.6**
DeepAugment [21]	76.6	60.4	3.5	42.2
+ AugMix [23]	75.8	53.5	3.9	46.8
+ DAT [35]	77.1	50.8	**6.8**	47.8
DAu+AM+Ours	**77.4**	**48.7**	5.9	**49.3**

Table 5. The top-1 accuracy results on ImageNet-A and ImageNet-R (higher is better). For ImageNet-C, we still use mCE value to evaluate the performance of the model (lower is better). It's worth noting that in these experiments, we just use the same augmented training strategy as the baseline model without any corrupted images. **The best results are indicated in bold.**

4.4. Analysis of Quality-Independent Features

In this section, we performed visualization analysis to show that our method can learn a quality-independent fea-

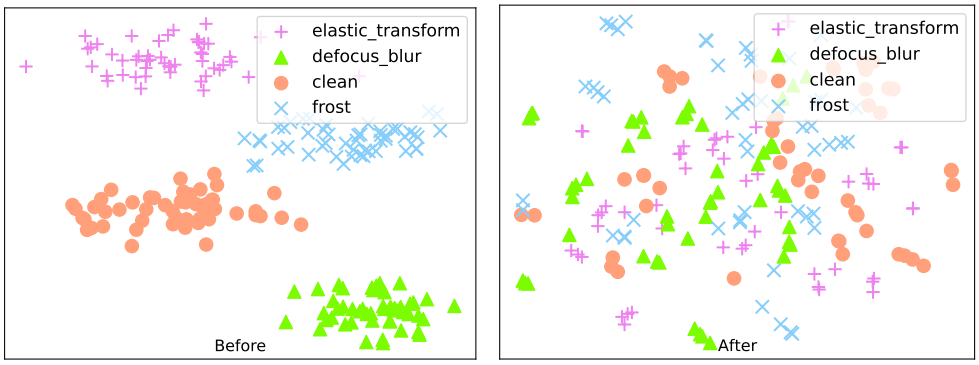


Figure 5. Corruption-wise feature distribution. Symbols with different colors are from different corruptions.

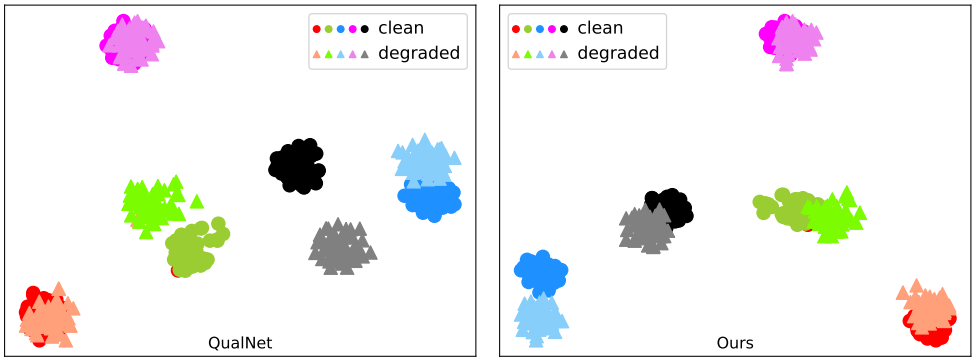


Figure 6. Class-wise feature distribution. Symbols with similar colors have the same labels.

ture representation.

First, we created t-SNE [52] visualization to show the learned feature distribution of our proposed VQ-based method. The results are shown in Fig. 5 and Fig. 6, respectively. In Fig. 5, "Before": Although the features are extracted from images with the same label, their distribution is divided, and this may result in poor recognition performance on low-quality images. "After": The feature distribution becomes relatively centralized through our method. In Fig. 6, marks with large color differences represent different labels of image features. Dot marks denote clean features, while triangle marks indicate degraded ones. It shows that whether the images are clean or degraded, our method can better aggregate features in the same class. These results show that the features extracted by our method are independent of image quality.

We also performed Grad-CAM [45, 61] on the model to show the attention map. From Fig. 1 we find that, compared to other methods, our model can still focus on class-relevant regions without interference from low-quality images. This qualitative result also proves that our method can extract quality-independent features, and the model has better robustness against common corruption. More Grad-CAM results can be found in our supplementary material in Sec. A.

5. Limitations & Conclusion

This paper has presented a plug-and-play method for low-quality image recognition through vector quantization and self-attention. Among them, VQ can map clean and multiple degraded features to the same discrete space to extract quality-independent features, which is beneficial for robust recognition. Experimental results in various settings verified the superiority of our method. Despite the promising results of our VQ-based approach, the optimality of a strategy that simply selects the most similar item in the codebook to quantify the input is questionable. We have experimentally attempted to select multiple items (e.g., top-k similar items), but the results did not improve. Therefore, the optimization of VQ-based quality-independent representation learning deserves further study.

6. Acknowledgment

This work was supported in part by the Natural Science Foundation of China under Grant 61991451 and Grant 61836008. Xin Li's work is partially supported by the NSF under grant IIS-2114664, CMMI-2146076, and the WV Higher Education Policy Commission Grant (HEPC.dsr.23.7).

References

[1] Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D Lawrence, and Zhenwen Dai. Variational information distillation for knowledge transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9163–9171, 2019. 2

[2] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013. 4

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 3

[4] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. *arXiv preprint arXiv:2204.04676*, 2022. 2

[5] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 1

[6] Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. Rethinking coarse-to-fine approach in single image deblurring. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4641–4650, 2021. 2

[7] Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, pages 1–2. Prague, 2004. 3

[8] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 113–123, 2019. 2

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1

[10] Samuel Dodge and Lina Karam. Understanding how image quality affects deep neural networks. In *2016 eighth international conference on quality of multimedia experience (QoMEX)*, pages 1–6. IEEE, 2016. 1

[11] Samuel Dodge and Lina Karam. A study and comparison of human and deep learning recognition performance under visual distortions. In *2017 26th international conference on computer communication and networks (ICCCN)*, pages 1–7. IEEE, 2017. 1

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[13] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image processing*, 15(12):3736–3745, 2006. 2, 3

[14] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 3

[15] Robert Geirhos, Carlos R Medina Temme, Jonas Rauber, Heiko H Schütt, Matthias Bethge, and Felix A Wichmann. Generalisation in humans and deep neural networks. *arXiv preprint arXiv:1808.08750*, 2018. 1

[16] Robert Gray. Vector quantization. *IEEE Assp Magazine*, 1(2):4–29, 1984. 3

[17] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10696–10706, 2022. 3

[18] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 3

[19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 1

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 6, 7

[21] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8349, 2021. 1, 2, 6, 7

[22] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 1, 2, 5, 7

[23] Dan Hendrycks, Norman Mu, Ekin D Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*, 2019. 2, 7

[24] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15262–15271, 2021. 6, 7

[25] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1921–1930, 2019. 2

[26] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 2

[27] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 3

[28] Christoph Kamann and Carsten Rother. Benchmarking the robustness of semantic segmentation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8828–8838, 2020. 2

[29] Insoo Kim, Seungju Han, Ji-won Baek, Seong-Jin Park, Jae-Joon Han, and Jinwoo Shin. Quality-agnostic image recognition via invertible decoder. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12257–12266, 2021. 1, 2, 6, 7

[30] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[31] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012. 1

[32] Ding Liu, Bihan Wen, Xianming Liu, Zhangyang Wang, and Thomas S Huang. When image denoising meets high-level vision tasks: A deep learning approach. *arXiv preprint arXiv:1706.04284*, 2017. 2

[33] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 3

[34] Chengzhi Mao, Lu Jiang, Mostafa Dehghani, Carl Vondrick, Rahul Sukthankar, and Irfan Essa. Discrete representations strengthen vision transformer robustness. *arXiv preprint arXiv:2111.10493*, 2021. 3

[35] Xiaofeng Mao, Yuefeng Chen, Ranjie Duan, Yao Zhu, Gege Qi, Shaokai Ye, Xiaodan Li, Rong Zhang, and Hui Xue. Enhance the visual representation via discrete adversarial training. *arXiv preprint arXiv:2209.07735*, 2022. 3, 7

[36] Xiaofeng Mao, Gege Qi, Yuefeng Chen, Xiaodan Li, Ranjie Duan, Shaokai Ye, Yuan He, and Hui Xue. Towards robust vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12042–12051, 2022. 3

[37] Claudio Michaelis, Benjamin Mitzkus, Robert Geirhos, Evgenia Rusak, Oliver Bringmann, Alexander S Ecker, Matthias Bethge, and Wieland Brendel. Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484*, 2019. 2

[38] Nasser M Nasrabadi and Robert A King. Image coding using vector quantization: A review. *IEEE Transactions on communications*, 36(8):957–971, 1988. 3

[39] Yanting Pei, Yaping Huang, Qi Zou, Yuhang Lu, and Song Wang. Does haze removal help cnn-based image classification? In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 682–697, 2018. 2

[40] Jialun Peng, Dong Liu, Songcen Xu, and Houqiang Li. Generating diverse structure for image inpainting with hierarchical vq-vae. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10775–10784, 2021. 3

[41] Zhiliang Peng, Li Dong, Hangbo Bao, Qixiang Ye, and Furu Wei. Beit v2: Masked image modeling with vector-quantized visual tokenizers. *arXiv preprint arXiv:2208.06366*, 2022. 3

[42] Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating diverse high-resolution images with vq-vae. 2019. 3

[43] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99, 2015. 1

[44] Evgenia Rusak, Lukas Schott, Roland S Zimmermann, Julian Bitterwolf, Oliver Bringmann, Matthias Bethge, and Wieland Brendel. A simple way to make neural networks robust against diverse image corruptions. In *European Conference on Computer Vision*, pages 53–69. Springer, 2020. 7

[45] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 1, 8

[46] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[47] Taeyoung Son, Juwon Kang, Namyup Kim, Sunghyun Cho, and Suha Kwak. Urie: Universal image enhancement for visual recognition in the wild. In *European Conference on Computer Vision*, pages 749–765. Springer, 2020. 2, 6

[48] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790, 2020. 1

[49] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021. 3

[50] Joseph Turian, Lev Ratinov, and Yoshua Bengio. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394, 2010. 2

[51] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 3, 4, 5

[52] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 8

[53] Igor Vasiljevic, Ayan Chakrabarti, and Gregory Shakhnarovich. Examining the impact of blur on recognition by convolutional networks. *arXiv preprint arXiv:1611.05760*, 2016. 1

[54] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3

[55] Yang Wang, Yang Cao, Zheng-Jun Zha, Jing Zhang, and Zhiwei Xiong. Deep degradation prior for low-quality image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11049–11058, 2020. 2, 6

[56] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 3

[57] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017. 6

[58] Ting-Bing Xu and Cheng-Lin Liu. Data-distortion guided self-distillation for deep neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5565–5572, 2019. 2

[59] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma. Image super-resolution via sparse representation. *IEEE transactions on image processing*, 19(11):2861–2873, 2010. 2, 3

[60] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14821–14831, 2021. 2

[61] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. 8

[62] Daquan Zhou, Zhiding Yu, Enze Xie, Chaowei Xiao, Animashree Anandkumar, Jiashi Feng, and Jose M Alvarez. Understanding the robustness in vision transformers. In *International Conference on Machine Learning*, pages 27378–27394. PMLR, 2022. 3