CNN-Based Classification of Degraded Images with Awareness of Degradation Levels

Kazuki Endo, Masayuki Tanaka, *Member, IEEE,* and Masatoshi Okutomi, *Member, IEEE*

***Abstract*—Image classification needs to consider the existence of image degradations in practice. Although degraded images have various levels of degradation, the degradation levels are usually unknown. This paper proposes a convolutional neural network to classify degraded images by using a restoration network and an ensemble learning. The proposed network can automatically infer ensemble weights by using estimated degradation levels of degraded images and features of restored images, where the degradation levels are estimated internally. The proposed network is mainly discussed with JPEG distortion, while degradations of both Gaussian noise and blurring are also examined. We demonstrate that the proposed network can classify degraded images over various levels of degradation. This paper also reveals how the image-quality of training data for a classification network affects the classification performance of degraded images.**

***Index Terms*—Degraded Image, Classification, Convolutional Neural Network, Ensemble, Restoration**

I. INTRODUCTION

IMAGE classification has been investigated in many reports due to the progress of the deep convolutional neural network (CNN) [1], [2], [3], [4], [5], [6]. These reports focus on the classification of clean images without any degradation. However, digital images in practical applications are usually degraded by noise, blur, compression, and other degradations. The performance of image classification is significantly dropped in the existence of degradations. Therefore, it is very important to construct a robust CNN against image degradations. Such a robust CNN is useful for autonomous driving, surveillance camera, etc. CNN-based classification of degraded images also needs to deal with low-quality images and high-quality images in practice. For example, JPEG compression, which is the de-facto standard of image compression, can have several levels of image quality against a clean image. There are various kinds of degradations and degradation levels. Therefore, it is required to classify degraded images with various levels of degradation when degraded images are input into a classification network.

Classification of degraded images has been proposed in several papers [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17]. A trivial approach is to input degraded images into a classification network trained with clean images. But this approach fails to classify degraded images due to the lack of knowledge of image degradations. A straightforward approach to overcome this deficit is to train a classification network with degraded images, where the network architecture is identical to that for clean images [9]. These two approaches are shown in Fig. 1-(a). Figure 1-(b) shows a sequential network that consists of an image restoration network and an image classification network [8], [13]. In the sequential network, degraded images are firstly restored, and then restored images are input into the image classification network. The sequential network has two options, which are to train the classification network with clean images or with restored images.

The classification network in Fig. 1-(a) trained with degraded images and the sequential network in Fig. 1-(b) can improve the classification performance of degraded images.

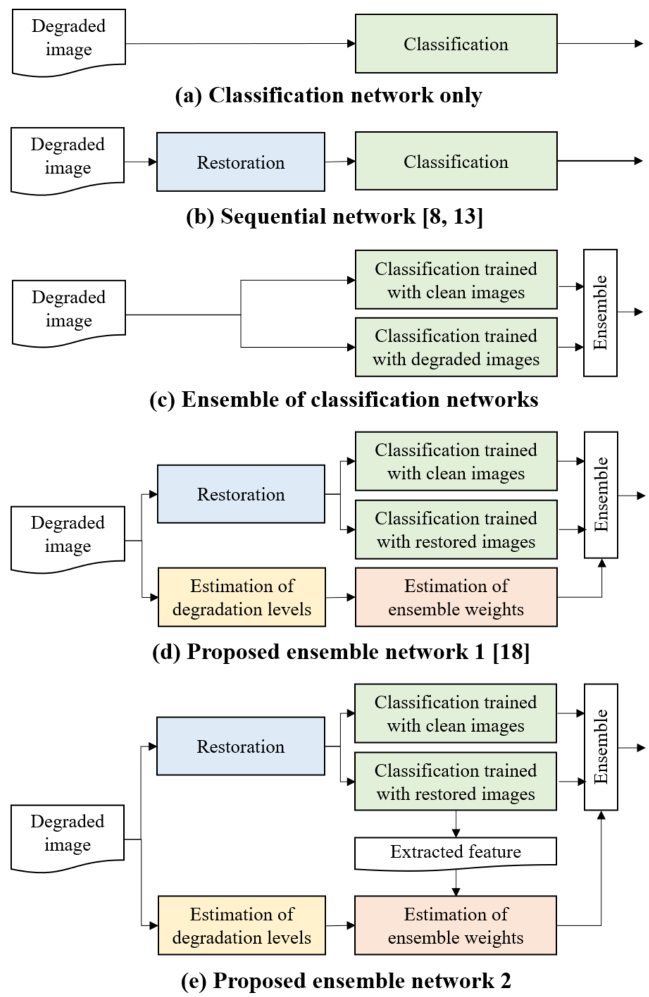


Fig. 1. Classification networks of degraded images

K. Endo, M. Tanaka and M. Okutomi are with the Department of Systems and Control Engineering, School of Engineering, Tokyo Institute of Technology, Meguro-ku, Tokyo, Japan e-mail: kendo@ok.sc.e.titech.ac.jp, mtanaka@sc.e.titech.ac.jp, mxo@sc.e.titech.ac.jp.

Manuscript received August XX, 2020; revised August XX, 20XX.

However, it is difficult for them to improve the classification performance of degraded images while keeping the classification accuracy of high-quality images. Figure 1-(c) is one of the solutions for this difficulty. It is an ensemble network of two classifiers trained with different image quality datasets; clean images and degraded images, which is described in [12]. In our previous study [18], we have proposed a CNN-based classification network of degraded images is an ensemble network of two sequential networks, as seen in Fig. 1-(d). The ensemble weights of the ensemble network depend on the degradation levels inferred by the estimation network of degradation levels. The ensemble network in Fig. 1-(d) shows better performance than that in Fig. 1-(c).

This paper proposes an ensemble network whose ensemble weights depend on not only degradation levels but also the features of a classification network trained with restored images, as shown in Fig. 1-(e). The proposed ensemble network includes an ensemble network in our previous study as a particular case. This paper focuses on JPEG distortion as an image degradation and analyzes the details of our proposed method. However, the proposed method can be applied to other image degradations, as demonstrated in section V. For the restoration of degraded images and the estimation of degradation levels, we adopt an existing CNN-based network [19], [20], [21], [22], [23], [24], [25], [26].

Our main contributions are three points as follows. The first point is to reveal how the classification performance of degraded images is affected by training data for a classification network, i.e. clean images or degraded images, and a restoration before a classification network. The second point is that we propose an ensemble network of sequential networks, which can estimate suitable ensemble weights depending on both an estimated degradation level and a feature extracted from the classification network trained with restored images. Our proposed network outperforms the sequential networks for most of all degradation levels. The last point is that we confirm the effectiveness of our proposed method for several degradations; JPEG distortion, additive Gaussian noise, and Gaussian blur.

This paper is an extended version of our conference paper [18] as follows. 1) An estimation network of ensemble weights is extended to consider the features of a classification network trained with restored images. 2) We add the fine-tuning of both the classification network trained with restored images and the estimation network of ensemble weights. 3) We also add the experimental validation for the classification of Gaussian blurring images. 4) The STL-10 [27] dataset is used in addition to the CIFAR [28] dataset for the classification of JPEG images.

II. Related works

A. Restoration of degraded images

CNN-based image restoration has been investigated and shown high-performance [19], [20], [21], [22], [24], [25]. Zhang *et al.* have proposed a residual CNN which has dilated convolutional layers with batch normalization for the image restoration such as super-resolution, removing noise and blur [19]. Recently, the performance of a classification network is used to evaluate single image super-resolution (SISR) [23] where the classification network is fixed. Our proposed method also uses both a restoration and a classification network. However, the classification network is trainable, and the restoration network is fixed. Although this paper uses a restoration which is the same network as proposed in [19], the other restoration algorithms are applicable to the proposed method.

B. Estimation of degradation levels

Estimation of degradation levels has been reported in [26], [29], [30]. Uchida *et al.* have proposed a CNN-based pixel-wise estimation of JPEG quality factors [26]. Liu *et al.* have proposed a noise level estimation method by selecting weak textured patches from a single noisy image [29], [30]. This paper uses almost the same network for the estimation of degradation levels, as proposed in [26]. However, other estimation algorithms are also applicable to the proposed method.

C. Classification of degraded images

Classification of degraded images, such as low-resolution, noise, blurring, compression, etc., has been investigated [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17]. Pei *et al.* have shown the impact of image degradation on the classification performance under several kinds of degradation [13]. Especially for haze and motion-blur, they have empirically shown that there are not much differences between the classification network only trained with degraded images and the sequential network incorporated a restoration [10], [13]. Endo *et al.* have proposed a classification network whose inputs are a degraded image and a degradation level [11]. They have shown that the degradation level can help the classification of degraded images and improve the classification performance. Gosh *et al.* have proposed an ensemble network of classification networks only trained with JPEG images [12]. They have also proposed a method based on maximum a posteriori (MAP) by using estimated JPEG quality factors and a simple method based on maximum probability. Chen *et al.* have proposed a network to cope with both a restoration and a recognition of degraded images simultaneously [17]. The network is not a sequential network but has common layers between the restoration and the recognition. Recently, consistency regularization approaches have been reported to classify degraded images [7], [16]. Wan *et al.* have focused on JPEG image classification and proposed a training method to keep the consistency of features extracted from clean images and JPEG images [7]. All weights of the JPEG image classifier are the same as the clean image classifier except for a residual mapping block, which is a restoration network. They focused on JPEG quality factors under 50. Pei *et al.* have proposed a consistency guided network to classify degraded images, which uses a classification network trained with clean images as a fixed consistency regularizer [16]. They did not investigate JPEG distortion but other degradations; haze, motion-blur, salt-and-pepper, low resolution, Gaussian blur, and Gaussian noise. Wan [7] and Pei [16] have not paid much attention to high-quality images but focused on middle-quality and low-quality images. However, it is hard to keep the classification accuracy of both high-quality and low-quality images [31]. In this paper, we cover a wide range of degradation levels. Our proposed method can classify high-quality and low-quality images well by the suitable ensemble of two classification networks; a classification network trained with clean images and a classification network trained with restored images. This paper mainly focuses on JPEG distortion but investigates Gaussian noise and Gaussian blur.

D. Ensembles of classification networks

Ensembles of classification networks are sometimes used to improve the performance. Xu *et al.* have proposed a CNN which has a shared network-in-network and branched fully convolutional sub-networks with multiple loss functions [32]. Though the CNN is trained with multiple loss functions, the final prediction in testing is an ensemble mean of predictions by each sub-network. The ensemble weights for sub-networks are not learned but constant. Ong *et al.* have proposed ensembles of several deep neural networks for semantic video classification [33]. In their approach, ensemble weights are learned in freezing all the weights of classification networks. Although our proposed method also learns ensemble weights for sequential networks, our proposed method considers both an estimated degradation level and a feature of a restored image to estimate ensemble weights. Thanks to this dependency of ensemble weights, the proposed method can classify degraded images well for several degradation levels.

III. PROPOSED METHOD

A. Proposed network

Our proposed network is shown in Fig. 1-(e). The proposed network consists of five sub-networks; a restoration network, two classification networks, an estimation network of degradation levels, and an estimation network of ensemble weights. Firstly, degraded images are restored by the restoration network. The restored images are fed into two classification networks; the classification network trained with clean images and the classification network trained with restored images. The classification networks infer each own probability vector. The features of the classification network trained with restored images are fed into the estimation network of ensemble weights. On the other hand, degraded images are also fed into the estimation network of degradation levels. Estimated degradation levels are fed into the estimation network of ensemble weights. The estimation network of ensemble weights infers ensemble weights for two probability vectors predicted by two classification networks. The weights take values in $[0, 1]$, and the summation of the weights is one. Finally, the predicted probability is calculated by weighted averaging.

Figure 2 shows the details of each network. The restoration network is almost the same as proposed in [19], where a batch normalization [34] is omitted for simplicity. The estimation network of degradation levels is almost the same as the network proposed in [11], [26]. The classification network is a VGG-like network [1], where we use a spatial dropout [3] and a convolution pooling [2] instead of a max pooling. The estimation network of ensemble weights has two inputs; the feature of the fully connected (FC) layer, which has 1024 elements, extracted from the classification network trained with restored images and an estimated degradation level. The feature is fed into FC layers and rectified linear units (ReLU), where we use a dropout. The output from FC1, which is denoted by f, is concatenated with the estimated degradation level represented by q. Finally, ensemble weights are estimated by a sigmoid function whose input is the linear combination of f and q.

The sigmoid function $w(f, q)$ in the estimation network of ensemble weights is defined by the following equation.

$$w(f, q) \overset{def}{=} \frac{1}{1 + \exp(-(af + bq + c))}, \tag{1}$$

where $a, b, c \in \mathbb{R}^1$, c denotes a bias. If a is identical to zero, $w(f, q)$ is simplified as

$$w(q) \overset{def}{=} \frac{1}{1 + \exp(-(bq + c))}. \tag{2}$$

Eq. (2) is the same sigmoid function as seen in our previous study [18]. We call the sigmoid functions Eqs. (1) and (2) as weight functions. For simplicity, $w_{f,q}$ and w_q denote $w(f, q)$ and $w(q)$, respectively.

B. Training procedure

Here, we describe the training procedure of the proposed network. Firstly, the restoration network is trained with pairs of degraded images and clean images, where its loss function is the mean square error (MSE) between clean images and restored images. Degraded images are generated from clean images by applying some degradation operations. Clean images are easily obtained from websites due to no need for any annotations.

The estimation network of degradation levels is trained with pairs of degraded images and true degradation levels. Its loss function is the MSE between true and predicted degradation levels. Degraded images can be generated in the same way as in training the restoration network, where true degradation levels are known.

Two classification networks are trained with different data. One is trained by using annotated clean images without any degradation. Another one is trained by using restored images with annotation. Degraded images with annotation are generated by applying some degradation operations to annotated clean images. Restored images with annotation are acquired from the degraded images by using the restoration network, where all the weights of the restoration network are fixed during the training of the classification network. Each loss function of the two networks is the cross-entropy between the correct labels and the predicted labels.

Finally, the estimation network of ensemble weights is trained by using degraded images with annotation, where its loss function is the cross-entropy between the correct labels and the predicted labels. When the estimation network of ensemble weights is trained, all the weights of the following three

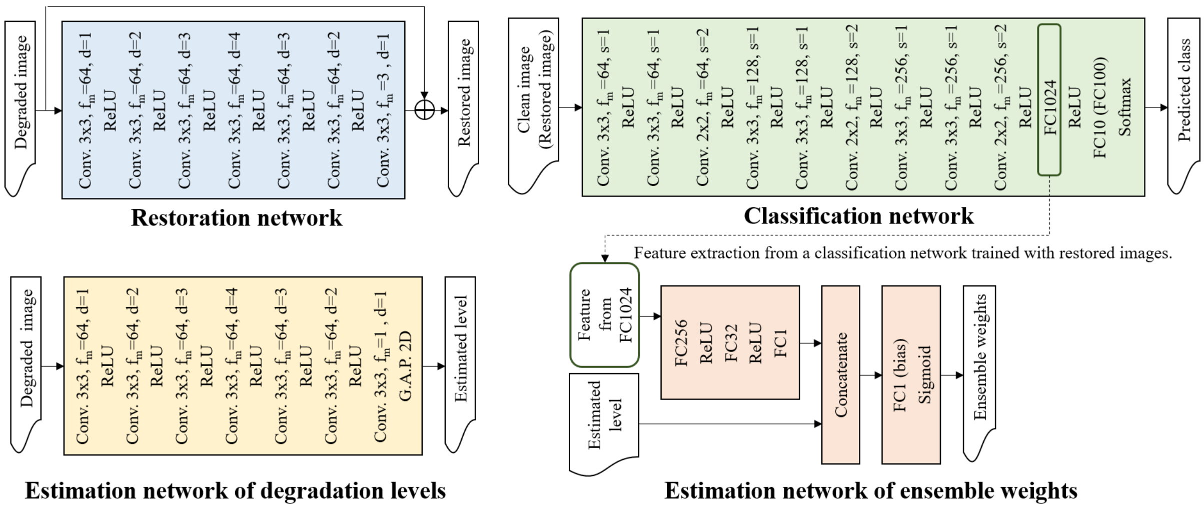


Fig. 2. Details of each network in the proposed network, where 3x3 or 2x2 represents the filter size, f_m is the dimension of feature map, d is the dilation rate, and s is the stride. "G.A.P." denotes global average pooling. The classification network has two choices for training dataset; clean images and restored images. The feature of FC1024 is extracted from the classification network trained with restored images.

networks are fixed; the restoration network, the estimation network of degradation level, and the classification network trained with clean images. On the other hand, it is possible to be trained further for the classification network trained with restored images. Specifically, there are two steps as follows. The first step is to fix the classification network trained with restored images when the estimation network of ensemble weights is trained. The next step is to fine-tune both the estimation network of ensemble weights and the classification network trained with restored images.

Note that the proposed network with the weight function Eq. (2) is equivalent to an ensemble network proposed in our previous study [18] if the classification network, which is trained with restored images, is also fixed. Thus, the proposed method includes the previous study as a special case. Adamax [35] is used as an optimizer for all the training.

C. Discussion

The essential idea of the proposed network is an ensemble learing of two classification networks trained with different data; clean images and restored images. The classification network trained with clean images can classify high-quality images but can not classify low-quality images well. On the other hand, the classification network trained with restored images shows good performance for low-quality images. Therefore, if the proposed ensemble network can decide the ensemble weights by depending on the degradation level of degraded images, degraded images can be classified over various levels of degradation. The key point is a dependency of the weight function on estimated levels of degradation. The weight functions of the proposed network depend on not only the estimated levels of degradation but also the features extracted from each restored image. The proposed network is expected to improve further by taking into account the individual features of each restored image.

Five sub-networks of the proposed network are trained in order. Another way of the training procedure is to train the proposed network in an end-to-end manner. For ease, we focus on the training of a sequential network, which is composed of a restoration network and a classification network, in the proposed network. A sequential network trained in an end-to-end manner did not show good performance, as described in IV-C. It indicates that the sequential network becomes just a deeper classification network in the end-to-end training and can not give a specific function to the restoration network. Therefore, five sub-networks are trained in order. Finally, further improvement is expected by fine-tuning of both the classification network trained with restored images and the estimation network of ensemble weights.

The proposed method can adopt different configurations or state-of-the-art networks for five sub-networks. The classification performance is expected to improve by using better sub-networks.

IV. Experiments

Experiments are mainly focused on JPEG distortion as an image degradation. The JPEG quality factor is used as a degradation level of JPEG distortion in this paper. Firstly, general settings, i.e. datasets, data augmentation, and a performance metric for classification, are described in IV-A and IV-B. Moreover, preliminary experiments are explained in IV-C. Then, the classification performance of degraded images is analyzed for classification networks in terms of incorporating a restoration network and training data, as described in IV-D. Our proposed method is evaluated in terms of weight functions and retraining the classification network trained with restored images or not, as explained in IV-E and IV-F. Finally, we confirmed the effectiveness of the proposed method under different datasets in IV-G and IV-H. The reproduction code is available on the Internet[1].

[1] http://www.ok.sc.e.titech.ac.jp/res/CNNIR/IRDI/

TABLE I
CLASSIFICATION ACCURACY FOR ORIGINAL CIFAR-10.

Ours	Xu [32] without DA	Xu [32] with DA
0.914	0.905	0.919

* DA denotes data augmentation.

TABLE II
INTERVAL MEAN ACCURACY OF THE SEQUENTIAL NETWORK IN TERMS OF DIFFERENT TRAINING PROCEDURES FOR JPEG CIFAR-10.

	End-to-end	Training in order*
$\overline{Acc}(1, 100)$	0.780	0.840

* It is the same as "Seq (res)" in Table VI.

TABLE III
COMPARISON OF NETWORKS IN TERMS OF INCORPORATING A RESTORATION AND TRAINING DATA FOR A CLASSIFICATION.

Name	Cla (org)	Cla (deg)	Seq (org)	Seq (res)
Restoration	-	-	✓	✓
Training data	Original	Degraded*	Original	Restored

* "Degraded" means JPEG in the case of JPEG distortion.

TABLE IV
CPSNR [dB] OF RESTORATION AND ESTIMATION OF DEGRADATION LEVELS FOR JPEG CIFAR-10.

Degradation level	10	30	50	70	90
Degradation*	23.28	26.73	28.25	29.82	33.65
Restoration	24.29	27.91	29.45	31.00	34.51
Estimated level	10.67	31.15	49.89	68.73	90.59
Standard deviation	1.17	2.70	3.53	3.03	1.65

* Degradation and degradation level denote JPEG distortion and the JPEG quality factor, respectively. JPEG compression is applied to CIFAR-10 test images with each JPEG quality factor.

A. Datasets and data augmentation

Three datasets were used to train both the restoration network and the estimation network of degradation levels; Yang91 [36], Urban100 [37], and General100 [38]. We generated 64×64 sized patches from each image and applied data augmentation to them by using transpose, horizontal, and vertical flips. Then, JPEG compression was applied to the patches, where JPEG quality factors were randomly sampled from 1 to 100[2].

The CIFAR datasets [28] were used to train the classification networks and the estimation network of ensemble weights. Data augmentation was applied to the CIFAR images; zoom, shearing, horizontal flip, rotation, vertical, and horizontal shifts. After that, JPEG compression was also applied to each image in the same way mentioned above. We denote these compressed CIFAR images as "JPEG CIFAR".

B. Interval mean accuracy

We use an interval mean accuracy as a metric to evaluate the classification performance of images degraded with different degradation levels. The following definition of the interval mean accuracy has been introduced in [11].

$$\overline{Acc}\left(\boldsymbol{\theta}; Q_l, Q_u\right) \overset{def}{=} \frac{\sum_{i=l}^{u} Acc\left(\boldsymbol{g}(\mathrm{D}\left(\mathbf{X}, Q_i\right); \boldsymbol{\theta}), \mathbf{Y}\right)}{|u - l + 1|},$$

where $\{Q_i \in \mathbb{R}^1 | i \in \mathbb{Z}\}$ denotes degradation levels, $\mathrm{D}\left(\mathbf{X}, Q\right)$ is a degradation operator with a degradation level Q for clean images $\mathbf{X}$, $\boldsymbol{g}(\cdot; \boldsymbol{\theta})$ represents a classification network with parameters $\boldsymbol{\theta}$, $\mathbf{Y}$ represents true labels for $\mathbf{X}$, and Acc is an accuracy. The accuracy is a ratio dividing the number of predicted class labels, which coincide with correct class labels, by the number of all test samples.

Accuracies for each degradation level have some fluctuations, even if the degradation levels are adjacent. The interval mean accuracy can remove those fluctuations by averaging accuracies over some degradation levels. For example, it is also easier to understand the tendency of a classification network over different degradation levels by dividing the whole range of degradation levels into low-quality, middle-quality, and high-quality. Therefore, the interval mean accuracy helps to compare the performance of different classification networks.

[2]The details of the JPEG compression algorithm depend on the library. Python Image Library(PIL) was used for JPEG compression. Note that the images compressed with the JPEG quality factor 100 also have JPEG distortion.

TABLE V
COMPARISON OF THE PROPOSED ENSEMBLE NETWORKS IN TERMS OF WEIGHT FUNCTIONS AND FINE-TUNING.

Name	w_q (fix)*	$w_{f,q}$ (fix)	w_q (tune)	$w_{f,q}$ (tune)
Weight function	w_q	$w_{f,q}$	w_q	$w_{f,q}$
Fine-tuning**	-	-	✓	✓

* w_q (fix) has been proposed in the previous study[18].
** w_q (fix) and $w_{f,q}$ (fix) are fine-tuned, respectively.

C. Preliminary experiments

Two preliminary experiments are presented here. One is to confirm the performance of the classification network that we used. The other is to explain why the proposed ensemble network is not trained in an end-to-end manner.

We trained our classification network with original CIFAR-10. Table I shows the classification accuracy of original CIFAR-10 test data. The accuracy of our classification network is 0.914. It is almost the same level as reported in [32]. Our classification network is not state-of-the-art but enough level for the experiments of following subsections.

For the second experiment, we focus on the sequential network composed in the proposed ensemble network as discussed in III-C. We trained the sequential network, whose all the weights were randomly initialized, in an end-to-end manner. Although the sequential network is composed of a restoration and a classification network, the loss function only considered the cross-entropy loss without any restoration losses. Table II shows the interval mean accuracy of JPEG CIFAR-10 for JPEG quality factors 1 to 100, where training in order denotes a training procedure proposed in III-B. Training in order outperforms the end-to-end, as seen in Table II. Therefore, the proposed ensemble network is not trained in the end-to-end manner but is trained in order.

TABLE VI
INTERVAL MEAN ACCURACY OF JPEG CIFAR-10.

$\overline{Acc}$ of	Existing					Proposed			
(Q_l, Q_u)	Cla (org)	Cla (deg)	Seq (org)	Seq (res)	Ens (Cla)	w_q (fix) [18]	$w_{f,q}$ (fix)	w_q (tune)	$w_{f,q}$ (tune)
(1,20)	0.431	0.724	0.569	0.736	0.677	0.737	0.737	0.744	**0.745**
(21,40)	0.700	0.844	0.802	0.852	0.825	0.855	0.859	0.855	**0.860**
(41,60)	0.763	0.857	0.849	0.864	0.844	0.870	0.875	0.869	**0.876**
(61,80)	0.799	0.866	0.874	0.870	0.858	0.883	0.885	0.877	**0.886**
(81,100)	0.861	0.874	0.902	0.878	0.885	**0.903**	0.902	0.895	0.902
(1,100)	0.711	0.833	0.799	0.840	0.818	0.850	0.852	0.848	**0.854**

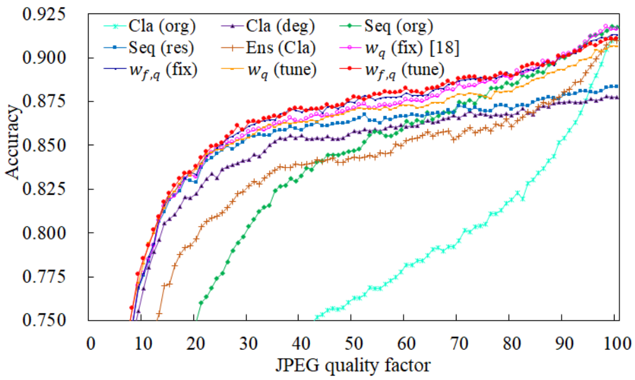


Fig. 3. Accuracy of JPEG CIFAR-10.

D. Performance analysis of classification networks for degraded images

We compare the performance of four networks summarized in Table III. "Cla (org)" and "Cla (deg)" are classification networks only trained by using original CIFAR-10 and JPEG CIFAR-10, respectively. "Seq (org)" and "Seq (res)" are sequential networks, where their classification networks are trained by using original CIFAR-10 and restored images of JPEG CIFAR-10, respectively. Table IV shows the color peak signal-to-noise ratio (CPSNR) of both JPEG CIFAR-10 images and images restored by the restoration network we used. Some samples of both JPEG CIFAR-10 and restored images can be seen in Fig. 11.

Figure 3 and Table VI show the accuracy and the interval mean accuracy of JPEG CIFAR-10, respectively. "Cla (org)" shows low performance under the existence of JPEG distortion. "Seq (org)", which incorporates the restoration network before "Cla (org)", outperforms "Cla (org)" for all JPEG quality factors. It shows that the restoration network helps "Cla (org)" to classify degraded images. However, "Seq (org)" does not show enough performance for low quality factors when comparing to "Cla (deg)" which is directly trained with JPEG CIFAR-10. "Cla (deg)" shows roughly better performance than "Cla (org)", but worse for the quality factors over around 95. On the other hand, "Seq (res)" slightly outperforms "Cla (deg)" for all JPEG quality factors, but still underperforms "Cla (org)" for high quality factors. When comparing "Seq (res)" and "Seq (org)", "Seq (res)" is better than "Seq (org)" for the quality factors under around 70, but worse over it. Therefore, as for high-quality images, the classification network trained with clean images outperforms the classification networks trained with degraded images or restored images whether the restoration network is incorporated or not.

The best network of existing approaches is "Seq (res)" for the quality factors under around 70, "Seq (org)" for over it. That is, the sequential networks outperform classification networks only.

E. Performance analysis of ensemble networks

The proposed ensemble network uses two sequential networks that have a classification network trained with clean images and one trained with restored images. These sequential networks could outperform classification networks only, as shown in IV-D. Therefore, using these sequential networks is reasonable for ensemble learning.

Table V shows four ensemble networks compared in terms of weight functions and fine-tuned ensemble networks. There are two weight functions defined by Eqs. (1) and (2). "w_q (fix)" and "$w_{f,q}$ (fix)" denote ensemble networks which can train an estimation network of ensemble weights only, where weight functions are w_q and $w_{f,q}$, respectively. "w_q (tune)" and "$w_{f,q}$ (tune)" are fine-tuned networks of "w_q (fix)" and "$w_{f,q}$ (fix)", respectively. Fine-tuned networks mean to retrain both an estimation network of ensemble weights and a classification network trained with restored images. "Ens (Cla)" is also analyzed in addition to four ensemble networks. "Ens (Cla)" denotes an ensemble network, as shown in Fig. 1-(c), whose decision is taken by the maximum probability of "Cla (org)" or "Cla (deg)". Table IV also shows the performance of an estimation network for JPEG quality factors which we used.

At first, the proposed method is compared with existing methods; "Seq (org)", "Seq (res)", and "Ens (Cla)". The proposed method includes four ensemble networks, as shown in Table V. Here, "w_q (fix)" is used for the comparison with existing methods because "w_q (fix)" is the simplest of four proposed ensemble networks. Figure 3 and Table VI show that "w_q (fix)" outperforms both "Seq (org)" and "Seq (res)" for almost all JPEG quality factors. Moreover, "w_q (fix)" outperforms "Ens (Cla)".

Now, four ensemble networks of the proposed method are compared. To confirm the effect of weight functions, "w_q (fix)" is compared with "$w_{f,q}$ (fix)". "$w_{f,q}$ (fix)" slightly outperforms "w_q (fix)" for quality factors under around 90, as shown in Fig. 3. Therefore, considering features of restored images in the weight function contributes to improve the classification performance.

Finally, the effect of fine-tuning is confirmed for "w_q (fix)" and "$w_{f,q}$ (fix)". "$w_{f,q}$ (tune)" outperforms "$w_{f,q}$ (fix)" for almost all degradation levels. However, "w_q (tune)" underperforms "w_q (fix)" for three intervals over the quality factor 41,

TABLE VII
ACCURACY OF JPEG CIFAR-10 FOR EACH JPEG QUALITY FACTOR.

JPEG quality factor	Wan [7]*	Proposed w_q (fix) [18]	Proposed $w_{f,q}$ (tune)
10	0.703	0.777	**0.785**
20	0.748	0.832	**0.838**
30	0.763	0.858	**0.863**
40	0.770	0.863	**0.871**
50	-	0.871	**0.878**
60	-	0.876	**0.882**
70	-	0.885	**0.888**
80	-	0.888	**0.890**
90	-	**0.902**	0.900
100	-	**0.916**	0.908

* Wan *et al.* presented numerical results for JPEG quality factors under 40 in their paper [7].

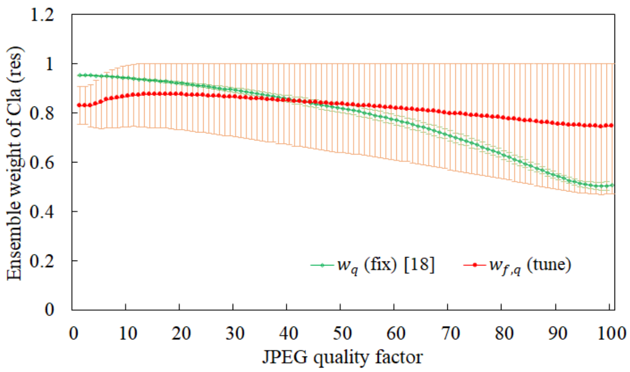


Fig. 4. The mean and standard deviation (1σ) of inferred ensemble weights for the classification network trained with restored images in the case of JPEG CIFAR-10, where 1σ interval was truncated by the maximum weight 1.

as shown in Table VI. The results show that fine-tuning is effective for "$w_{f,q}$ (fix)".

That is, "$w_{f,q}$ (tune)" outperforms other networks for almost all quality factors under around 90, as shown in Fig. 3. However, "$w_{f,q}$ (tune)" underperforms "w_q (fix)" for quality factors over around 90. This is also confirmed from the accuracies of JPEG CIFAR-10 for each JPEG quality factor, as shown in Table VII. Table VII also shows that "w_q (fix)" and "$w_{f,q}$ (tune)" outperform Wan [7] for low quality factors. The results show that the proposed ensemble networks "$w_{f,q}$ (tune)" and "w_q (fix)" can classify both high-quality and low-quality images well.

F. Analysis of ensemble weights

Here, we analyse how ensemble weights are inferred when changing JPEG quality factors. Two ensemble networks, which are "$w_{f,q}$ (tune)" and "w_q (fix)", are focused on this analysis because the networks show good performance as explained in IV-E. Figure 4 shows mean and standard deviation of inferred ensemble weights for the classification network trained with restored images in the case of JPEG CIFAR-10. The mean and standard deviation of inferred ensemble weights are calculated over all test data of JPEG CIFAR-10 for each JPEG quality factor. We used 1σ interval as the standard deviation which was truncated by the maximum ensemble weight 1.

TABLE VIII
INTERVAL MEAN ACURRACY OF JPEG CIFAR-100.

$\overline{Acc}$ of (Q_l, Q_u)	Existing Cla (org)	Cla (deg)	Seq (org)	Seq (res)	Ens (Cla)	Proposed w_q (fix)[18]	$w_{f,q}$ (tune)
(1,20)	0.202	0.448	0.324	0.453	0.391	0.454	**0.461**
(21,40)	0.407	0.561	0.493	0.564	0.540	0.572	**0.581**
(41,60)	0.465	0.577	0.545	0.579	0.567	0.594	**0.603**
(61,80)	0.504	0.583	0.582	0.586	0.580	0.611	**0.617**
(81,100)	0.582	0.591	0.628	0.592	0.616	0.637	**0.638**
(1,100)	0.432	0.552	0.514	0.555	0.539	0.573	**0.580**

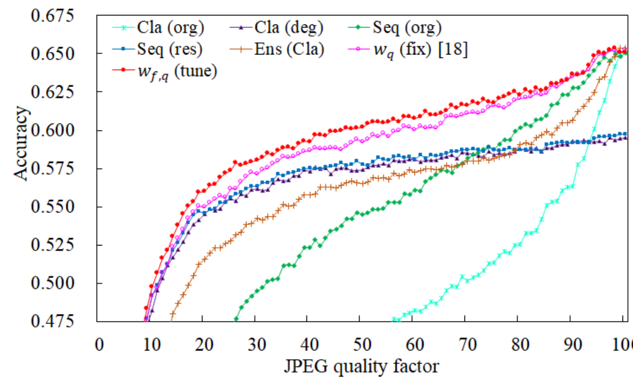


Fig. 5. Accuracy of JPEG CIFAR-100.

The mean of inferred ensemble weights roughly decreases as JPEG quality factors increase for "$w_{f,q}$ (tune)" and "w_q (fix)". In other words, ensemble weights of the classification network trained with restored images are decreasing, and those of the classification network trained with clean images are increasing for high-quality images. It is quite reasonable because "Seq (org)" outperforms "Seq (res)" for high-quality images, as shown in IV-D. The slope of "w_q (fix)" is sharper than that of "$w_{f,q}$ (tune)". It shows that "w_q (fix)" is more sensitive to JPEG quality factors than "$w_{f,q}$ (tune)".

"$w_{f,q}$ (tune)" has bigger standard deviations of inferred ensemble weights than "w_q (fix)". That is because there are two sources of fluctuation in $w_{f,q}$; a feature extracted from the classification network trained with restored images and an estimated quality factor. On the other hand, w_q has the only source of fluctuation which is the estimated quality factor of a JPEG image. The result indicates that the weight function "$w_{f,q}$" can reflect the feature of each image in inferring ensemble weights.

G. Experiments for CIFAR-100 dataset

We confirm the effectiveness of the proposed method for JPEG CIFAR-100 as another dataset in the case of JPEG distortion. Figure 5 and Table VIII show the accuracy and the interval mean accuracy of JPEG CIFAR-100, respectively. "$w_{f,q}$ (tune)" outperforms the other networks for almost all JPEG quality factors including high quality factors. Thus, the proposed method is also effective for JPEG CIFAR-100.

TABLE IX
INTERVAL MEAN ACURRACY OF JPEG STL-10.

$\overline{Acc}$ of	Existing					Proposed	
(Q_l, Q_u)	Cla (org)	Cla (deg)	Seq (org)	Seq (res)	Ens (Cla)	w_q (fix)[18]	$w_{f,q}$ (tune)
(1,20)	0.522	0.595	0.524	0.602	0.579	**0.605**	**0.605**
(21,40)	0.636	0.667	0.652	0.671	0.671	**0.682**	**0.682**
(41,60)	0.647	0.676	0.670	0.680	0.680	0.694	**0.696**
(61,80)	0.656	0.683	0.680	0.687	0.687	0.702	**0.706**
(81,100)	0.687	0.688	0.698	0.692	0.706	0.712	**0.716**
(1,100)	0.630	0.662	0.645	0.666	0.664	0.679	**0.681**

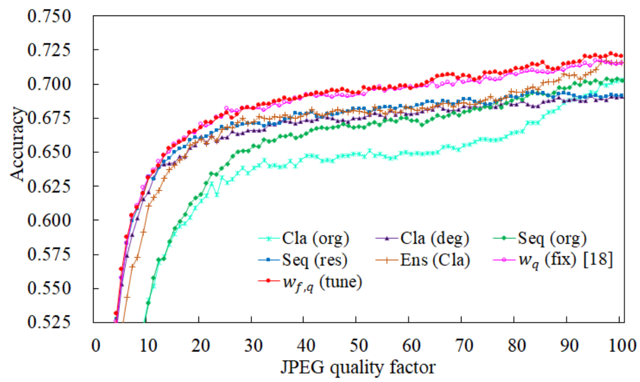


Fig. 6. Accuracy of JPEG STL-10.

H. Experiments for STL-10 dataset

Here, the effectiveness of the proposed method is confirmed for the STL-10 [27] dataset. The STL-10 dataset includes pairs of 96×96 images and their labels. We used 5,000 annotated images for training data and 8,000 annotated images for test data. The STL-10 images without annotations were not used. The STL-10 images are resized into 32×32 images from 96×96 images. Then, JPEG compression is applied to the resized STL-10 images by changing the JPEG quality factors from 1 to 100. We call the images "JPEG STL-10" as with the CIFAR images. Figure 12 shows sample images of both JPEG STL-10 and restored images. We used the same pre-trained networks for both an image restoration and an estimation of degradation levels as the case of JPEG CIFAR.

Figure 6 and Table IX show the accuracy and the interval mean accuracy of JPEG STL-10, respectively. The proposed method "$w_{f,q}$ (tune)" outperforms other networks for almost all quality factors as with JPEG CIFAR. Therefore, the proposed method is effective for not only the CIFAR dataset but also the STL-10 dataset.

V. APPLICATIONS FOR OTHER DEGRADATION

Here, we demonstrate some applications of the proposed method. The proposed method is applicable to not only JPEG distortion but also other degradations. We confirmed the additive Gaussian noise and Gaussian blur as examples of other degradations, where CIFAR-10 and CIFAR-100 datasets were used.

TABLE X
CPSNR [dB] OF RESTORATION AND ESTIMATION OF DEGRADATION LEVELS FOR GAUSSIAN NOISY CIFAR-10.

Degradation level	10	20	30	40	50
Degradation*	28.13	22.11	18.59	16.09	14.15
Restoration	33.47	29.74	27.58	26.05	24.87
Estimated level	10.03	20.11	30.16	40.11	49.26
Standard deviation	0.40	0.45	0.57	0.72	0.54

* Degradation and degradation level denote the additive Gaussian noise and its noise level, respectively. The Gaussian noise is added to CIFAR-10 test images with each noise level.

TABLE XI
INTERVAL MEAN ACCURACY OF GAUSSIAN NOISY CIFAR-10.

$\overline{Acc}$ of	Existing					Proposed	
(Q_l, Q_u)	Cla (org)	Cla (deg)	Seq (org)	Seq (res)	Ens (Cla)	w_q (fix)[18]	$w_{f,q}$ (tune)
(0,10)	0.796	0.851	0.903	0.862	0.838	0.904	**0.905**
(11,20)	0.328	0.844	0.883	0.852	0.509	0.889	**0.893**
(21,30)	0.138	0.829	0.848	0.838	0.271	0.865	**0.874**
(31,40)	0.106	0.809	0.797	0.821	0.198	0.835	**0.848**
(41,50)	0.101	0.785	0.740	0.798	0.173	0.804	**0.823**
(0,50)	0.304	0.824	0.835	0.835	0.406	0.860	**0.869**

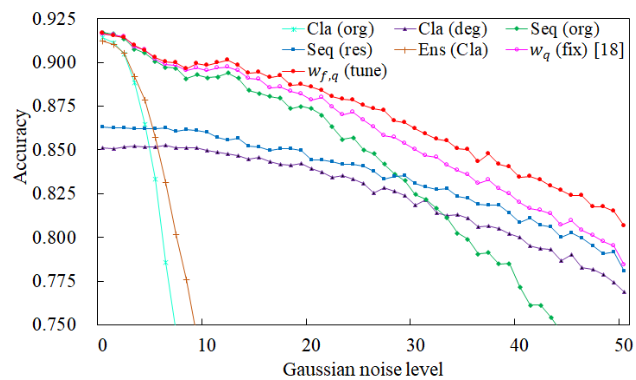


Fig. 7. Accuracy of Gaussian noisy CIFAR-10.

A. Additive Gaussian noise

We applied the proposed method for the additive Gaussian noise. A degradation operator was just replaced from JPEG compression to Gaussian noise. We added CIFAR images and Gaussian noise whose noise level changed from 0 to 50 in 1.0 steps, where the noise level is a standard deviation of Gaussian distribution for the 8-bit image. We call the images "Gaussian noisy CIFAR". Table X shows the CPSNR of restored images and the estimation of the Gaussian noise level for Gaussian noisy CIFAR-10. Some examples of both Gaussian noisy CIFAR-10 and restored images can be seen in Fig. 13.

Figure 7 and Table XI show the accuracy and the interval mean accuracy of Gaussian noisy CIFAR-10, respectively. Figure 8 and Table XII show the accuracy and the interval mean accuracy of Gaussian noisy CIFAR-100, respectively. "$w_{f,q}$ (tune)" outperforms the other networks for the Gaussian noisy CIFAR.

TABLE XII
INTERVAL MEAN ACCURACY OF GAUSSIAN NOISY CIFAR-100.

	Existing					Proposed	
$\overline{Acc}$ of (Q_l, Q_u)	Cla (org)	Cla (deg)	Seq (org)	Seq (res)	Ens (Cla)	w_q (fix)[18]	$w_{f,q}$ (tune)
(0,10)	0.505	0.566	0.640	0.574	0.585	**0.644**	**0.644**
(11,20)	0.139	0.560	0.612	0.569	0.458	0.620	**0.623**
(21,30)	0.052	0.548	0.564	0.557	0.411	0.587	**0.597**
(31,40)	0.029	0.528	0.509	0.536	0.290	0.553	**0.567**
(41,50)	0.014	0.506	0.453	0.515	0.166	0.524	**0.539**
(0,50)	0.155	0.542	0.557	0.551	0.386	0.587	**0.595**

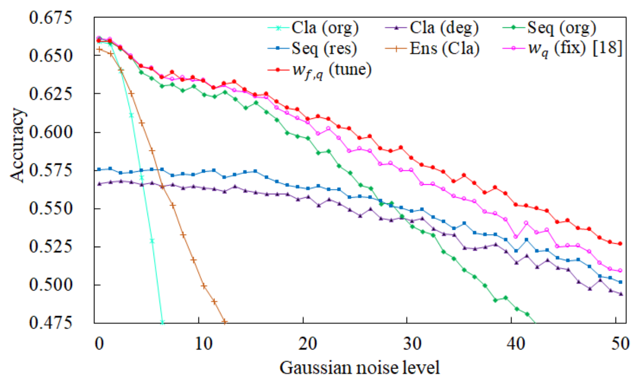


Fig. 8. Accuracy of Gaussian noisy CIFAR-100.

TABLE XIII
CPSNR [dB] OF RESTORATION AND ESTIMATION OF DEGRADATION LEVELS FOR GAUSSIAN BLURRING CIFAR-10.

Degradation level	1.0	2.0	3.0	4.0	5.0
Degradation*	25.73	21.27	19.46	18.39	17.66
Restoration	31.58	27.56	24.88	23.06	21.52
Estimated level	1.04	1.51	2.98	3.99	4.84
Standard deviation	0.08	0.08	0.09	0.11	0.07

* Degradation and degradation level denote the Gaussian blur and the standard deviation of its kernel, respectively. The Gaussian blur is applied to CIFAR-10 test images with each degradation level.

B. Gaussian blur

The next application is the case of Gaussian blur. We applied the Gaussian blurring filter to CIFAR images, where the standard deviation of its kernel changed from 0 to 5 in 0.1 steps. We call the filtered images "Gaussian blurring CIFAR". Table XIII shows the CPSNR of restored images and the estimated standard deviations of the Gaussian blur kernel for Gaussian blurring CIFAR-10. There are some examples of both Gaussian blurring CIFAR-10 and restored images in Fig. 14.

Figure 9 and Table XIV show the accuracy and the interval mean accuracy of Gaussian blurring CIFAR-10, respectively. Figure 10 and Table XV show the accuracy and the interval mean accuracy of Gaussian blurring CIFAR-100, respectively. Unlike the case of JPEG and Gaussian noise, "Cla (org)" shows good performance for high-quality images as shown in Fig. 9 and Fig. 10 because the blurring effect of high-quality images is very small. However, "$w_{f,q}$ (tune)" almost outperforms the other networks for the Gaussian blurring CIFAR. Thus, the proposed method is also effective for the case of Gaussian blur.

TABLE XIV
INTERVAL MEAN ACCURACY OF GAUSSIAN BLURRING CIFAR-10.

	Existing					Proposed	
$\overline{Acc}$ of (Q_l, Q_u)	Cla (org)	Cla (deg)	Seq (org)	Seq (res)	Ens (Cla)	w_q (fix)[18]	$w_{f,q}$ (tune)
(0,1.0)	0.883	0.856	0.905	0.883	0.900	**0.912**	0.909
(1.1,2.0)	0.373	0.841	0.861	0.872	0.823	0.886	**0.889**
(2.1,3.0)	0.208	0.807	0.679	0.846	0.792	0.849	**0.852**
(3.1,4.0)	0.183	0.766	0.445	0.807	0.748	0.808	**0.812**
(4.1,5.0)	0.171	0.722	0.293	0.758	0.698	0.758	**0.768**
(0,5.0)	0.374	0.799	0.642	0.834	0.794	0.844	**0.847**

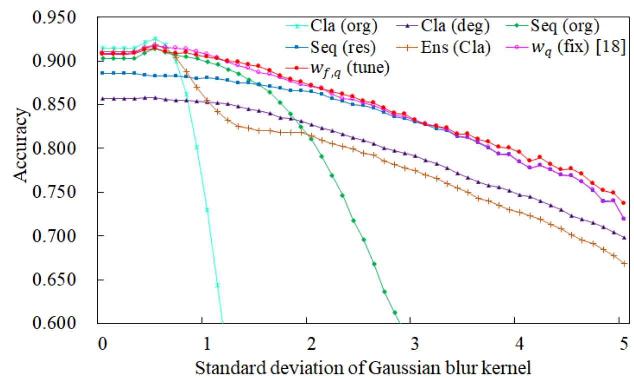


Fig. 9. Accuracy of Gaussian blurring CIFAR-10.

TABLE XV
INTERVAL MEAN ACCURACY OF GAUSSIAN BLURRING CIFAR-100.

	Existing					Proposed	
$\overline{Acc}$ of (Q_l, Q_u)	Cla (org)	Cla (deg)	Seq (org)	Seq (res)	Ens (Cla)	w_q (fix)[18]	$w_{f,q}$ (tune)
(0,1.0)	0.609	0.528	0.638	0.588	0.624	**0.652**	**0.652**
(1.1,2.0)	0.163	0.526	0.563	0.579	0.401	0.605	**0.609**
(2.1,3.0)	0.055	0.500	0.408	0.558	0.353	0.564	**0.567**
(3.1,4.0)	0.035	0.467	0.256	0.521	0.338	0.523	**0.531**
(4.1,5.0)	0.030	0.424	0.149	0.471	0.301	0.471	**0.484**
(0,5.0)	0.187	0.490	0.407	0.544	0.408	0.565	**0.570**

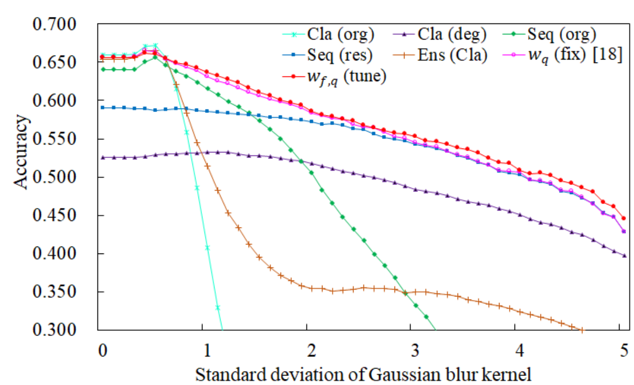


Fig. 10. Accuracy of Gaussian blurring CIFAR-100.

VI. CONCLUSIONS

This paper has proposed the ensemble network which shows higher performance for various degradation levels. Firstly,

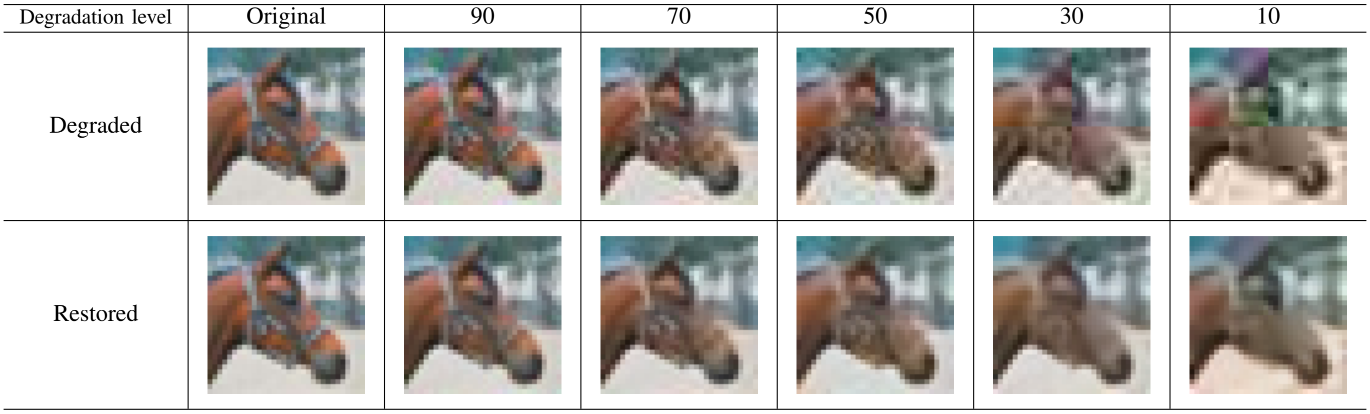


Fig. 11. JPEG CIFAR-10 images and their restored images. The degradation level means the JPEG quality factor.

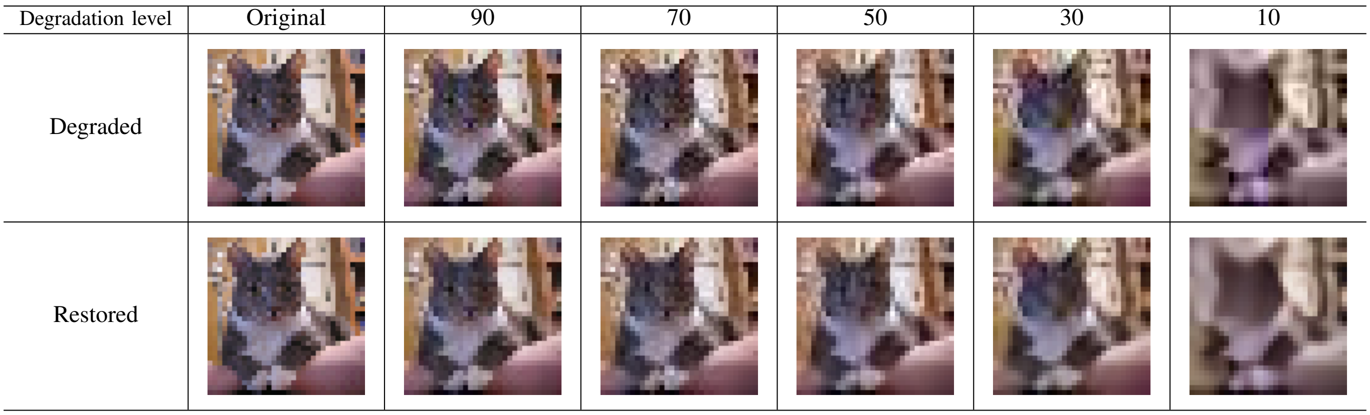


Fig. 12. JPEG STL-10 images and their restored images. The degradation level means the JPEG quality factor.

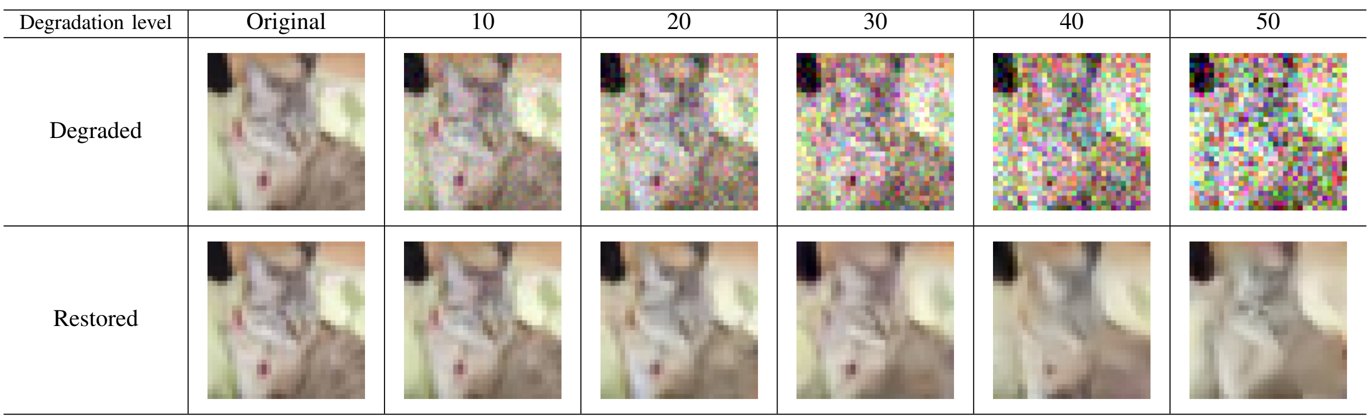


Fig. 13. Gaussian noisy CIFAR-10 images and their restored images. The degradation level means the Gaussian noise level.

we confirmed that two sequential networks, which are incorporating a restoration network into a classification network, outperform the classification networks only trained with clean or degraded images. Then, we also found the sequential network shows the different performance depending on an image-quality of training data for classification networks. Based on the results, the proposed network was constructed by using ensemble learning of the sequential networks. The ensemble weights of the proposed network were automatically inferred depending on both the estimated degradation levels and the features of each image extracted from the classification network trained with restored images. The result showed that the proposed network infers the ensemble weights suitably when changing degradation levels. Finally, we have shown that the proposed network is effective for not only the JPEG distortion but also the additive Gaussian noise and the Gaussian blur.

A further practical extension of the proposed method is to cope with the mixture of some degradations because the type of degradation is not necessarily unique. Moreover, the proposed method can be applied to other degradations. These

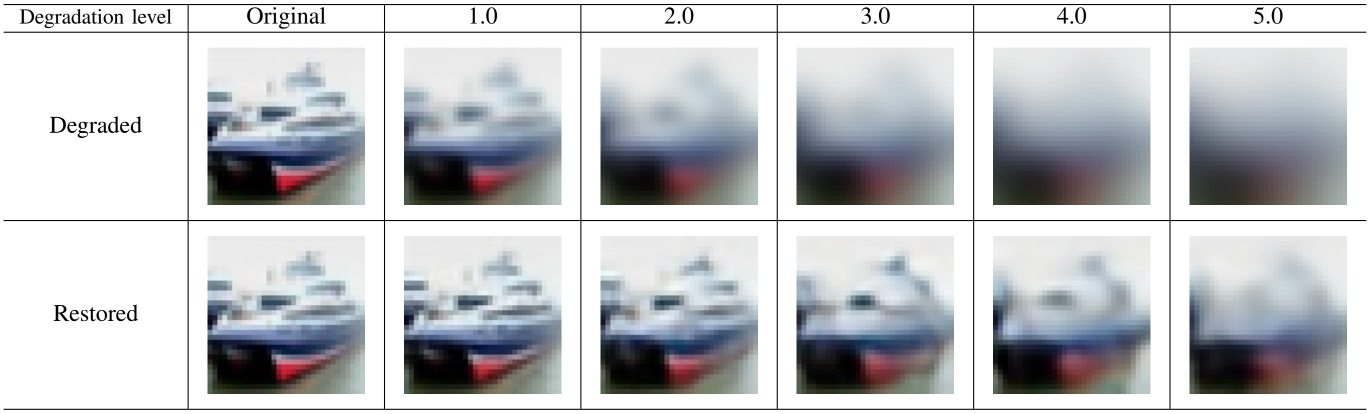


Fig. 14. Gaussian blurring CIFAR-10 images and their restored images. The degradation level means the standard deviation of the Gaussian blur kernel.

remaining tasks would be our future work.

References

[1] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[2] J. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving simplicity: the all convolutional net," in *International Conference on Learning Representations Workshop Track*, 2015.

[3] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 648–656.

[4] M. Tanaka, "WiG: Weighted sigmoid gate unit for an activation function of deep neural network," *Pattern Recognition Letters*, vol. 135, pp. 354–359, July 2020.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[6] ——, "Identity mappings in deep residual networks," in *European Conference on Computer Vision*, 2016, pp. 630–645.

[7] S. Wan, T. Wu, H. Hsu, W. Wong, and C. Lee, "Feature consistency training with jpeg compressed images," *IEEE Transactions on Circuits and Systems for Video Technology*, Early Access 2019.

[8] D. Cai, K. Chen, Y. Qian, and J. Kämäräinen, "Convolutional low-resolution fine-grained classification," *Parttern Recognition Letters*, vol. 119, pp. 166–171, March 2019.

[9] X. Peng, J. Hoffman, S. Yu, and K. Saenko, "Fine-to-coarse knowledge transfer for low-res image classification," in *IEEE International Conference on Image Processing*, 2016.

[10] Y. Pei, Y. Huang, Q. Zou, H. Zang, X. Zhang, and S. Wang, "Effects of image degradations to cnn-based image classification," *arXiv:1810.05552*, 2018.

[11] K. Endo, M. Tanaka, and M. Okutomi, "Cnn-based classification of degraded images," in *Proceedings of IS&T International Symposium on Electronic Imaging*, 2020.

[12] S. Ghosh, R. Shet, P. Amon, A. Hutter, and A. Kaup, "Robustness of deep convolutional neural networks for image degradations," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018, pp. 2916–2920.

[13] Y. Pei, Y. Huang, Q. Zou, X. Zhang, and S. Wang, "Effects of image degradation and degradation removal to cnn-based image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, November 2019.

[14] J. Seo and H. Park, "Object recognition in very low resolution images using deep collaborative learning," *IEEE Access*, vol. 7, September 2019.

[15] Z. Wang, S. Chang, Y. Yang, D. Liu, and T. S. Huang, "Studying very low resolution recognition using deep networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4792–4800.

[16] Y. Pei, Y. Huang, and X. Zhang, "Consistency guided network for degraded image classification," *IEEE Transactions on Circuits and Systems for Video Technology*, Early Access 2020.

[17] R. Chen, L. Mihaylova, H. Zhu, and N. C. Bouaynaya, "A deep learning framework for joint image restoration and recognition," *Circuits, Systems, and Signal Processing*, vol. 39, pp. 1561–1580, March 2020.

[18] K. Endo, M. Tanaka, and M. Okutomi, "Classifying degraded images over various levels of degradation," in *IEEE International Conference on Image Processing*, October 2020.

[19] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep cnn denoiser prior for image restoration," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[20] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: residual learning of deep cnn for image denosing," *IEEE Transaction on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[21] J. Kim, J. Lee, and K. Lee, "Accurate image super-resolution using very deep convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[22] C. Dong, C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European Conference on Computer Vision*, 2014, pp. 184–199.

[23] S. Park, H. Son, S. Cho, K. Hong, and S. Lee, "Srfeat: single image super-resolution with feature discrimination," in *European Conference on Computer Vision*, 2018, pp. 455–471.

[24] K. Uchida, M. Tanaka, and M. Okutomi, "Non-blind image restoration based on convolutional neural network," in *IEEE Global Conference on Consumer Electronics*, October 2018.

[25] ——, "Estimation and restoration of compositional degradation using convolutional neural networks," *arXiv:1812.09629v1*, 2018.

[26] ——, "Pixelwise jpeg compression detection and quality factor estimation based on convolutional neural network," in *Proceedings of IS&T International Symposium on Electronic Imaging*, 2019.

[27] A. Coates, H. Lee, and A. Y. Ng, "An analysis of single-layer networks in unsupervised feature learning," in *International Conference on Artificial Intelligence and Statistics*, April 2011.

[28] A. Krizhevsky, "Learning multiple layers of features from tiny images," Master's thesis, Department of Computer Science, University of Toronto, 2009.

[29] X. Liu, M. Tanaka, and M. Okutomi, "Noise level estimation using weak textured patches of a single noisy image," in *IEEE International Conference on Image Processing*, September 2012.

[30] ——, "Single-image noise level estimation for blind denoising," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 5226–5237, December 2013.

[31] A. Gnansambandam and S. H. Chan, "One size fits all: Can we train one denoiser for all noise levels?" in *International Conference on Machine Learning*, 2020.

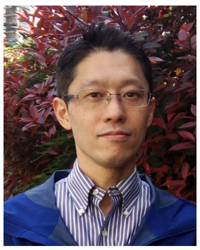
[32] C. Xu, C. Lu, X. Liang, J. Cao, W. Zheng, T. Wang, and S. Yan, "Multi-loss regularized deep neural network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 12, December 2016.

[33] E. Ong, S. S. Husain, M. Bober-Irizar, and M. Bober, "Deep architectures and ensembles for semantic video classification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 12, December 2019.

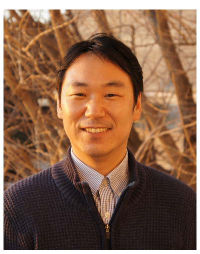
[34] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*, 2015.

[35] D. Kingma and J. Ba, "Adam: a method for stochastic optimization," in *International Conference on Learning Representations*, 2015.

[36] J. Yang, J. Wright, T. Huang, and Y. Ma, "Image super-resolution as sparse representation of raw image patches," in *IEEE Conference on Computer Vision*, 2008.
[37] J. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5197–5206.
[38] C. Dong, C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *European Conference on Computer Vision*, 2016.



Kazuki Endo received a bachelor's degree in mathematics and a master's degree in industrial engineering and management from Tokyo Institute of Technology, Tokyo, Japan, in 1997 and 1999, respectively. He joined Industrial Bank of Japan, Ltd., which is current Mizuho Bank, Ltd., Tokyo, Japan, in 1999. He was in charge of developing valuation algorithms and systems of derivatives. Since 2014, he has worked for a startup Topology Inc., Tokyo, Japan. Since 2019, he has belonged to the Ph.D. course at Tokyo Institute of Technology.



Masayuki Tanaka received his bachelor's and master's degrees in control engineering and Ph.D. degree from Tokyo Institute of Technology in 1998, 2000, and 2003. He joined Agilent Technology in 2003. He was a Research Scientist at Tokyo Institute of Technology from 2004 to 2008. Since 2008, he has been an Associate Professor at the Graduate School of Science and Engineering, Tokyo Institute of Technology. He was a Visiting Scholar with Department of Psychology, Stanford University, CA, USA.

Masatoshi Okutomi received a B.Eng. degree from the Department of Mathematical Engineering and Information Physics, the University of Tokyo, Japan, in 1981 and an M.Eng. degree from the Department of Control Engineering, Tokyo Institute of Technology, Japan, in 1983. He joined Canon Research Center, Canon Inc., Tokyo, Japan, in 1983. From 1987 to 1990, he was a visiting research scientist in the School of Computer Science at Carnegie Mellon University, USA. In 1993, he received a D.Eng. degree for his research on stereo vision from Tokyo Institute of Technology. Since 1994, he has been with Tokyo Institute of Technology, where he is currently a professor in the Department of Systems and Control Engineering, the School of Engineering.