PAPER

End-to-End Deep ROI Image Compression*

Hiroaki AKUTSU†a), *Member and* **Takahiro NARUKO**†, *Nonmember*

SUMMARY In this paper, we present the effectiveness of image compression based on a convolutional auto encoder (CAE) with region of interest (ROI) for quality control. We propose a method that adapts image quality for prioritized parts and non-prioritized parts for CAE-based compression. The proposed method uses annotation information for the distortion weights of the MS-SSIM-based loss function. We show experimental results using a road damage image dataset that is used to check damaged parts and an image dataset with segmentation data (ADE20K). The experimental results reveals that the proposed weighted loss function with CAE-based compression from F. Mentzer et al. learns some characteristics and preferred bit allocations of the prioritized parts by end-to-end training. In the case of using road damage image dataset, our method reduces bpp by 31% compared to the original method while meeting quality requirements that an average weighted MS-SSIM for the road damaged parts be larger than 0.97 and an average weighted MS-SSIM for the other parts be larger than 0.95.

key words: *deep image compression, ROI, quality control*

1. Introduction

The internet of things (IoT) is expected to play a key role in achieving sustainable development goals (SDGs) set by the United Nations in 2015 [2]. According to IDC, IoT devices are expected to generate over 90 ZB of data (IoT data) in 2025 [3]. Image data generated by IoT devices (such as surveillance cameras, on-vehicle cameras, and smartphones) is enormous and is generated at every moment. These vast amounts of data are expected to enable solutions that improve public services in the future. To transfer and store this increasing data, technology that provides a high compression ratio for the data is needed.

For example, in the field of public infrastructure maintenance, massive data are considered to be generated. Because the population is concentrated in urban areas, rural areas are depopulated and infrastructure is getting older. Moreover, due to the declining birthrate and the aging population, human resources for public services are not enough. Therefore, it is thought that maintenance will be automated by IoT manner in the future. Furthermore, autonomous vehicles are expected to generate huge amounts of data throughout the city, and these vast amounts of image data will be used for various digital solutions in the future.

If it is possible to create a specialized compressor for these applications, the cost of their systems can be reduced. It can be considered that there are two types of image compression: general-purpose compression such as JPEG and compression specialized for application. Deep learning based compression has some advantages as compression specialized for application, because by using loss functions and image datasets suitable for the application, a compressor specialized for the application can be automatically generated.

In this paper, we present the effectiveness of image compression based on a convolutional auto encoder (CAE) with region of interest (ROI) for quality control. The proposed method uses annotation information (e.g. bounding boxes or segmentation maps) for the distortion weights of the MS-SSIM-based loss function for training the compression network. We show experimental results using images taken by on-vehicle cameras used to check damaged regions on the roads for maintenance work. And we also show another experimental results using ADE20K dataset with segmentation data to show our proposed method's effectiveness.

2. Related Works

2.1 Convolutional Autoencoder Based Image Compression

Leading research [5]–[7] has covered the compression methods for images using neural networks. These methods train a CAE with a large amount of training data. An image compression technique using a neural network has the advantage that an arbitrary differentiable function can be set as a loss function and a compressor is trained in an end-to-end manner. In general, image quality measures such as PSNR (a mean squared error based metric) and MS-SSIM [8] (which qualifies structural similarities) are used as the loss function. CAE-based image compression methods such as [5], [6] automatically learn to adjust to the bit rate necessary for each part on an image by using a technique called an "importance map" with end-to-end learning.

Selective generative compression [9] generates portions of images by a generative adversarial network (GAN) to improve a compression rate further. This method can dramatically improve a compression rate up to 0.1 bpp or less

Manuscript received October 2, 2019.
Manuscript revised December 31, 2019.
Manuscript publicized January 24, 2020.

†The authors are with the R&D Group, Hitachi, Ltd., Yokohama-shi, 244–0817 Japan.

*A preliminary version of this paper was presented at CVPR 2019 Workshop as "End-to-End Learned ROI Image Compression" [1] by the same author. The explanation and experimental results were expanded.

a) E-mail: hiroaki.akutsu.cs@hitachi.com

DOI: 10.1587/transinf.2019EDP7264

instead of storing the details of images. However, because our purpose in this paper is to keep important details such as the damaged parts of roads, the problem with this approach is that it changes the shape and characteristics of the parts.

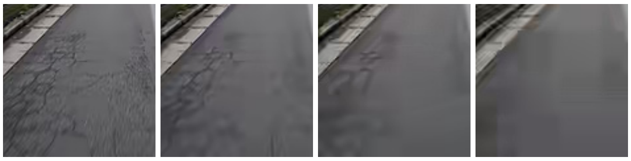
2.2 Other Codecs

JPEG is an image compression codec that has been widely used as a standard on the internet for decades. The JPEG2000 image coding standard provides a feature called region of interest (ROI) [10]. It changes the compression rate and the quality for each area so that areas specified as important have high quality. This approach is similar to our approach. However, our approach differs in that the encoder and decoder automatically learns the features of the important part in the end-to-end manner from the training data with annotation.

BPG [11] is one of the latest image compression codecs, based on a subset of the HEVC open video compression standard. These methods are designed to be general purpose and are often evaluated by the PSNR as a benchmark.

Figure 1 shows road damage images encoded by BPG at different qualities. We found that the details of the damaged parts disappear at a low bit rate of 1.0 bpp or less.

In this paper, we examined the effectiveness of applying the state-of-the-art CAE-based image compression method with ROI. We assumed that end-to-end learning gives the CAE-based methods better compression rates compared to conventional methods.



q = 32 (bpp: 1.164)

q = 36 (bpp: 0.754)

q = 40 (bpp: 0.457)

q = 44 (bpp: 0.255)

Fig. 1 Road damage images at different qualities encoded by BPG (using Adachi_20170906093840 [4]).

3. Proposed Method

Assuming that there are important and unimportant parts in the image, we aim to control the allocation of the amount of bits according to the specified image quality for prioritized and non-prioritized parts. Our method adds the following steps to the CAE-based image compression method.

1. Use a loss function with a distortion loss that changes parameters of an image quality metric for each area according to the auxiliary annotation information.
2. Append a new encoder input channel and feed it annotation information for manual quality control (optional).

In order to apply the proposed method 1, it is necessary that annotation imformation indicating important parts are available in addition to the image data as the network training data. If the annotation information is always provided even after training, the compression ratio sometimes can be improved by applying method 2 additionally. However, as describe in Chapter 4, since this difference is very small, it works well even if there is no annotation information after training the network. Therefore, from a practical point of view, method 2 is optional.

3.1 Network Architecture

We employ [6] as the network architecture of the compressor. Figure 2 shows the entire architecture overview. The annotation information A is a two-dimensional array ($W \times H$) of values representing the degree of importance of each pixel on the image. A is use for calculating distortion weights of our MS-SSIM-based loss function during network training. A is optionally used as encoder input for manual quality control during and after the network training.

If we use the annotation information A as encoder input, A is input into 1 out of 4 channels of the encoder. Image data in RGB format ($3 \times W \times H$) is input into 3 out of 4 channels of the encoder.

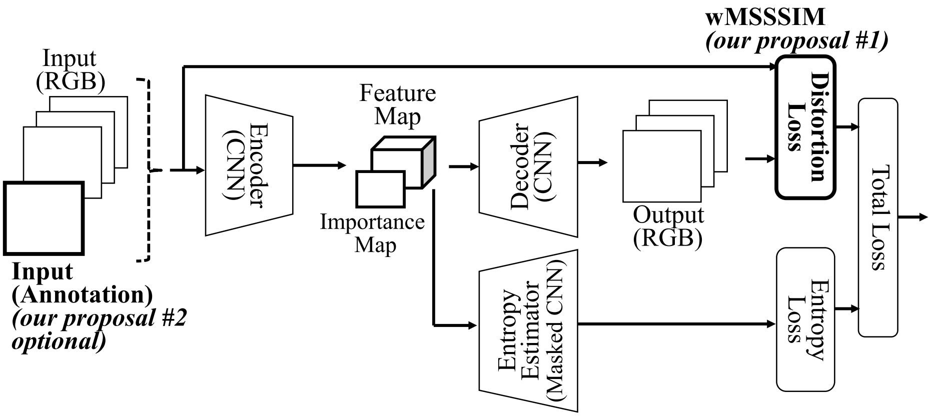


Fig. 2 Network architecture overview.

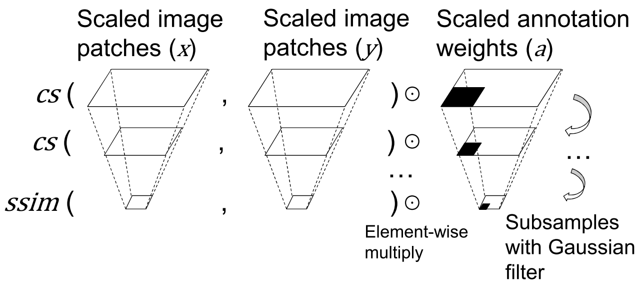

Scaled image patches (x)
Scaled image patches (y)
Scaled annotation weights (a)
cs (
cs (
$ssim$ (
,
,
,
) ⊙
) ⊙
...
) ⊙
...
Element-wise multiply
Subsamples with Gaussian filter

Fig. 3 $wMSSSIM$ calculation overview.

The importance map which is introduced in [6] is a CNN (the last layer of the encoder) channel output for masking the other CNN channels output in order to control the number of active channels for each area of the image. Mask bits are constructed from the importance map values and element-wise multipled with the other channel outputs, details are shown in [6]. By our proposal loss function, CNN learns to produce the importance map to control image quality of the important part and the other part.

3.2 Loss Function

We defined weighted MS-SSIM (wMSSSIM), image quality metrics that reflect annotation information. They are used for image quality assessment and loss function for quality control in this paper.

3.2.1 Weighted MS-SSIM

SSIM is an image quality metric that takes structural similarity for good approximation of perceived image quality, and multi-scale SSIM (MS-SSIM) is a multi-scale extension of SSIM [8].

Let $\mathbf{x}_{i,j}$ and $\mathbf{y}_{i,j}$ be the ith local image patches at the jth scale, let $a_{i,j}$ be the ith local annotation weight at the jth scale, let M be the number of scales, let β_j be the scale weight at the jth scale, let $ssim$ be the local SSIM metric function, and let cs be the local contrast and structure metric function, then the weighted MS-SSIM ($wMSSSIM$) is computed as

$$wMSSSIM = [\frac{\sum_i a_{i,M} ssim(\mathbf{x}_{i,M}, \mathbf{y}_{i,M})}{\sum_i a_{i,M}}]^{\beta_M} \prod_{j=1}^{M-1} [\frac{\sum_i a_{i,j} cs(\mathbf{x}_{i,j}, \mathbf{y}_{i,j})}{\sum_i a_{i,j}}]^{\beta_j}. \tag{1}$$

Figure 3 shows the entire $wMSSSIM$ calculation overview. In MS-SSIM calculation, the images are subsampled to each scale and Gaussian filtering is performed to the images for the local $ssim$ and cs calculation. Our method also performs the same process for the annotation information A to calculate $a_{i,j}$ to realize the natural image quality change at the boundaries between the priority parts and the other parts. In our experiment, A has a constant positive value c for the prioritized area and 0 for the non-prioritized area. Let A_j be the subsampled A for each scale j (note that $A = A_1$), and let $\mathbf{a}_{i,j}$ be the local annotation patches from A_j, then we get $a_{i,j}$ by performing Gaussian filter to $\mathbf{a}_{i,j}$. We used parameter $M = 5$ and $\beta_j = \{0.0448, 0.2856, 0.3001, 0.2363, 0.1333\}$ same as paper [8]. Gaussian filter parameter is $\sigma = 1.5$ and Gaussian kernel size is set to 11.

Table 1 Experimental conditions.

Items	Conditions
Base model [6]	Encoder and Decoder: 3 Layer 2DCNN + 15 Residual blocks Entropy Estimator: 2 Layer 3DCNN (masked) + 1 Residual block
Training iteration	100,000 iterations of batches
Train data (RoadDamageDataset)	6,925 cliped images from [4] (Width:160, Height:160)
Test data (RoadDamageDataset)	1,811 files from [4] (exclude images with no damaged parts) (Width:256, Height:256)
Train data (ADE20K)	5,268 cliped car images from [13] (Width:160, Height:160)
Test data (ADE20K)	132 files from [13] (exclude images with no car parts) (Width:256, Height:256)
Quality settings (RoadDamageDataset)	$T_p = 0.03$ ($wMSSSIM_p = 0.97$), $T_{np} = 0.05$ ($wMSSSIM_{np} = 0.95$) $\lambda = 5$
Quality settings (ADE20K)	$T_p = 0.02$ ($wMSSSIM_p = 0.98$), $T_{np} = 0.05$ ($wMSSSIM_{np} = 0.95$) $\lambda = 25$

By taking a weighted average using scaled annotation weights $a_{i,j}$ for each scale to the MS-SSIM, the image quality metrics reflect the importance of each part.

We referred to paper [12] that uses information content weight with MS-SSIM. In [12], the information content weight are calculated from only the local image information to improve performance of perceptual quallity. Our approach is different in that we use specified external annotation information for weight.

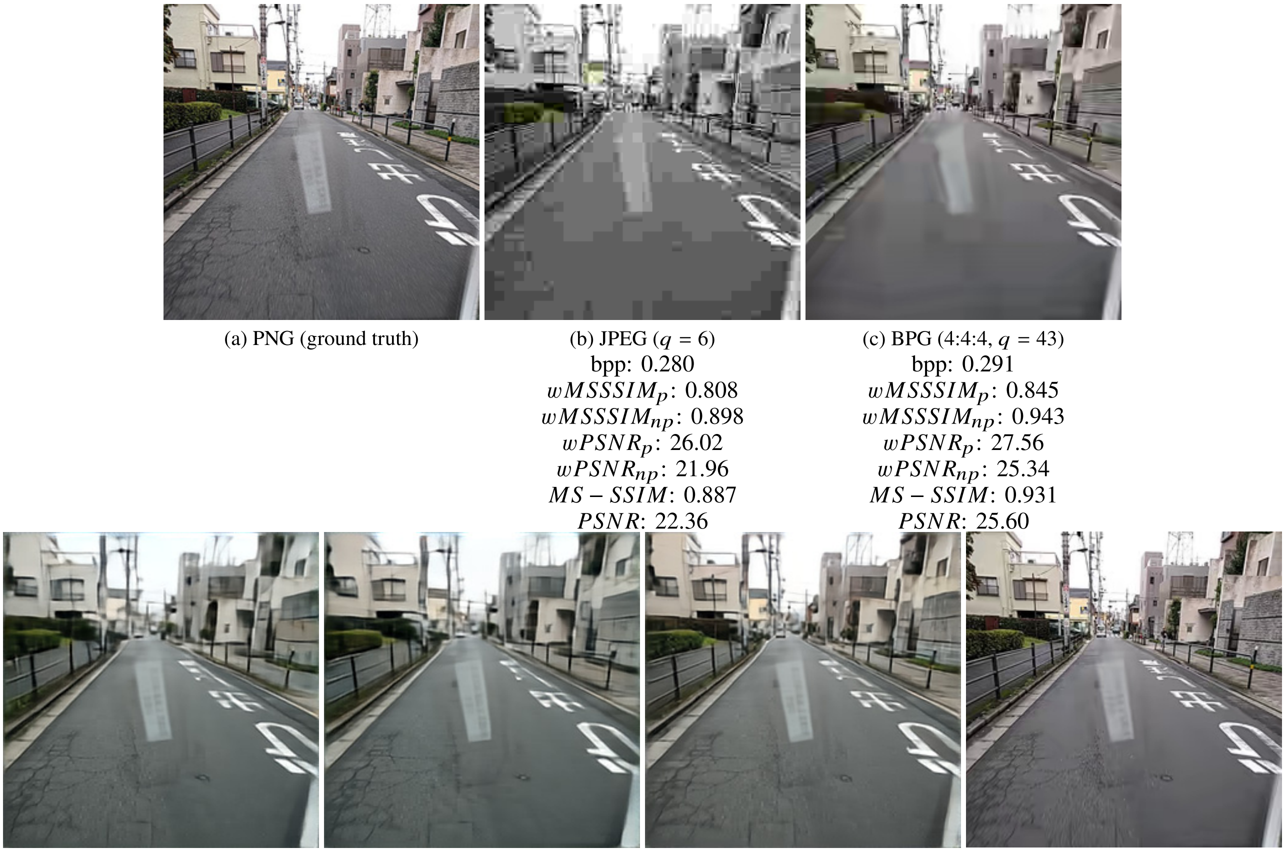
3.2.2 Quality Control Loss Function

In order to optimize the rate-distortion trade-off in image compression by end-to-end learning, the following loss function is generally used as in CAE-based compression [5]–[7].

$$\mathcal{L} = \mathcal{L}_e + \lambda \mathcal{L}_d \tag{2}$$

$\mathcal{L}_e$ represents the information entropy that corresponds to bpp. $\mathcal{L}_e$ is calculated by an entropy estimator based on CNN (see [6] for details). λ is a parameter that determines the desired rate-distortion trade-off. $\mathcal{L}_d$ is a distortion term that qualifies an image quality. $\mathcal{L}_d$ is defined by the following equation in our method.

$$\mathcal{L}_d = \max(1 - wMSSSIM_p, T_p) + \max(1 - wMSSSIM_{np}, T_{np}) \tag{3}$$



(a) PNG (ground truth)

(b) JPEG (q = 6)
bpp: 0.280
$wMSSSIM_p$: 0.808
$wMSSSIM_{np}$: 0.898
$wPSNR_p$: 26.02
$wPSNR_{np}$: 21.96
$MS-SSIM$: 0.887
$PSNR$: 22.36

(c) BPG (4:4:4, q = 43)
bpp: 0.291
$wMSSSIM_p$: 0.845
$wMSSSIM_{np}$: 0.943
$wPSNR_p$: 27.56
$wPSNR_{np}$: 25.34
$MS-SSIM$: 0.931
$PSNR$: 25.60

(d) Proposal train with [6] (w/o input A)
bpp: 0.263
$wMSSSIM_p$: 0.976
$wMSSSIM_{np}$: 0.957
$wPSNR_p$: 27.20
$wPSNR_{np}$: 21.35
$MS-SSIM$: 0.960
$PSNR$: 21.84

(e) Proposal train with [6] (w input A)
bpp: 0.252
$wMSSSIM_p$: 0.976
$wMSSSIM_{np}$: 0.956
$wPSNR_p$: 27.42
$wPSNR_{np}$: 21.31
$MS-SSIM$: 0.959
$PSNR$: 21.82

(f) Normal MS-SSIM train with [6]
bpp: 0.384
$wMSSSIM_p$: 0.971
$wMSSSIM_{np}$: 0.975
$wPSNR_p$: 27.97
$wPSNR_{np}$: 22.89
$MS-SSIM$: 0.974
$PSNR$: 23.34

(g) BPG (4:4:4, q = 31)
bpp: 1.286
$wMSSSIM_p$: 0.971
$wMSSSIM_{np}$: 0.986
$wPSNR_p$: 34.02
$wPSNR_{np}$: 34.39
$MS-SSIM$: 0.984
$PSNR$: 34.33

Fig. 4 Experimental results (using Adachi_20170906093840).

$wMSSSIM_p$ and $wMSSSIM_{np}$ represent image quality calculated by Eq. (1) in prioritized parts and non-prioritized parts. Target distortion of priority parts T_p and non-priority parts T_{np} are given by quality settings. Max operations are used because it is prioritized to reduce bpp once the target quality levels of the parts are satisfied. $wMSSSIM_p$ is calculated with A, and $wMSSSIM_{np}$ is calculated with the inverse of A, i.e. $c-A$. However, if a priority area does not exist, $wMSSSIM_p$ cannot be calculated because A is a zero matrix and $\sum_i a_{i,j}$ equals 0, which leads to 0-division in Eq. (1). This problem also occur when the non-prioritized area does not exist when calculate $wMSSSIM_{np}$. To avoid this problem, a sufficiently small coefficient is added to A during training.

4. Experiments

4.1 Experimental Conditions

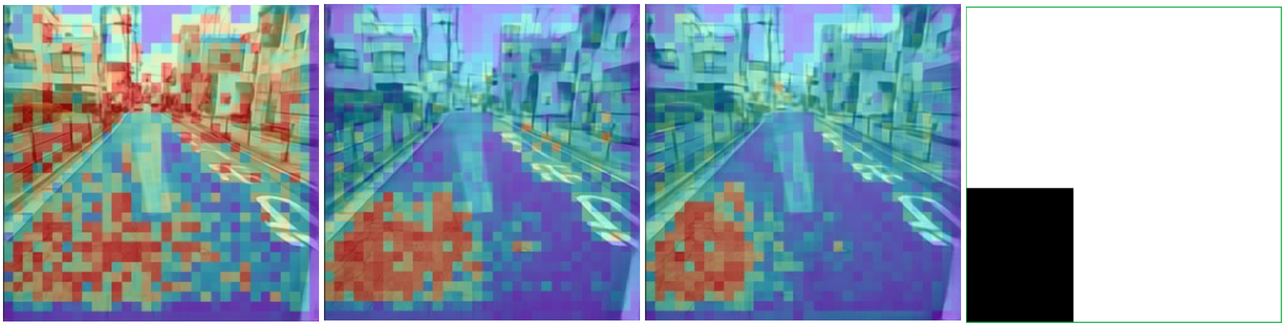
We evaluated the effectiveness of our method with the RoadDamageDataset [4], which is a dataset of images that contain damaged parts of a road. The RoadDamageDataset contains annotation information and image data, which are downsampled to 256×256 pixels in this evaluation. We also evaluated the proposed method with ADE20K dataset which contains segmentation information and set area with cars as the prioritized parts. ADE20K segmentation information and image data are downsampled and clipped to 256 × 256 pixels for evaluations. For details on the network architecture and implementation that our method employs as a base, refer to paper [6]. The settings of this experiment are summarized in Table 1. We set the chroma format to 4:4:4 and 4:2:0, and we set compression level to 9 (maximum) when evaluating BPG. Quantizer parameter settings are described in each figure and tables. The other parameters are set to default values. We used BPG version 0.9.8 for the evaluations. We used ImageMagick 6.8.9-9 for the JPEG compression evaluations. JPEG quality level settings are described in each figure and the other parameters are set to default values. (the chroma format is 4:2:0).

Table 2 Experimental results (averages of RoadDamageDataset test data).

Method	bpp	$wMSSSIM_p$	$wMSSSIM_{np}$	$wPSNR_p$	$wPSNR_{np}$	$MS-SSIM$	$PSNR$
Proposal train (w input A)	0.251	0.970	0.952	26.78	22.06	0.955	22.51
Proposal train (w/o input A)	0.263	0.970	0.953	27.04	21.98	0.956	22.45
Normal train	0.382	0.970	0.973	27.81	23.55	0.973	23.95
BPG 4:4:4 (q = 32)	1.183	0.970	0.985	32.93	33.39	0.982	33.28
BPG 4:2:0 (q = 32)	1.122	0.968	0.982	32.67	32.63	0.979	32.61

Table 3 Experimental results (averages of car images in ADE20K).

Method	bpp	$wMSSSIM_p$	$wMSSSIM_{np}$	$wPSNR_p$	$wPSNR_{np}$	$MS-SSIM$	$PSNR$
Proposal train (w input A)	0.249	0.979	0.957	23.90	24.50	0.960	24.48
Proposal train (w/o input A)	0.251	0.977	0.957	23.88	24.65	0.959	24.62
Normal train	0.312	0.973	0.974	23.00	25.93	0.974	25.68
BPG 4:4:4 (q = 40)	0.338	0.975	0.961	26.43	28.83	0.962	28.65
BPG 4:2:0 (q = 40)	0.323	0.972	0.957	26.08	28.51	0.958	28.33



(a) Normal MS-SSIM train with [6]

(b) Proposal train with [6] (w/o input A)

(c) Proposal train with [6] (w input A)

(d) Annotation

Fig. 5 Importance maps and an annotation information (using Adachi_20170906093840).

Two networks are independently constructed and trained for case (i) annotation data is input to the encoder as a hint and (ii) the annotation data is not input. The main reason for comparing the two cases is to show that the proposed method can automatically learn important parts without manually inputting annotation data and achieve almost the same bit rate.

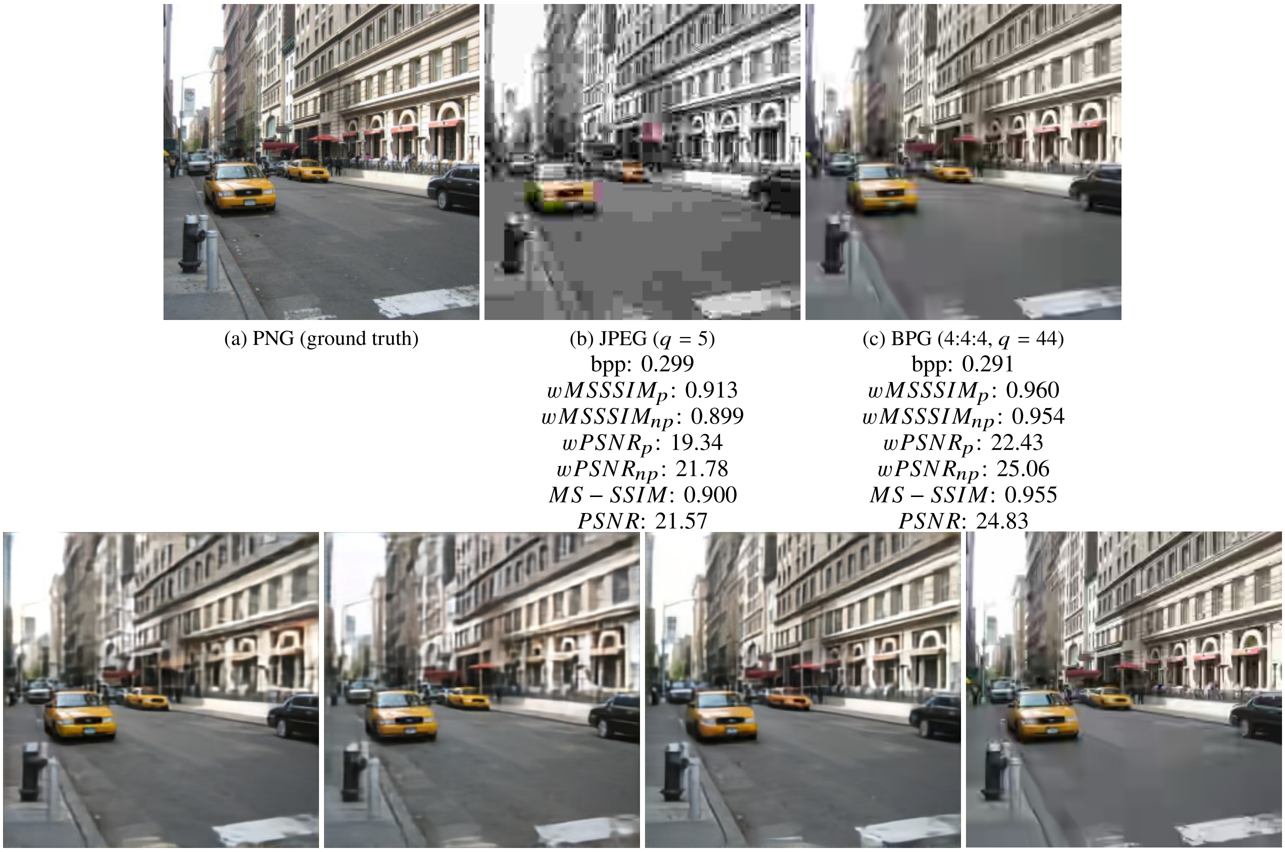
4.2 Results

The experimental results are shown in Figs. 4 and 6 (a)–(g). We define wPSNRp and wPSNRnp as PSNR value in the prioritized and non-prioritized parts respectively and use them as a quality metric in addition to wMSSSIM. With the CAE-based compression [6] trained by our methods ((d) and (e)), the portion has a higher quality compared to the conventional codecs like JPEG (b) and BPG (c) under the same level bpp conditions. Compared to method [6] without our methods (f) and BPG (g) our methods ((d) and (e)) reduces bpp under the same level or even lower quality conditions of the prioritized portion. Our method ((d) and (e)) works because the quality of the important area (wMSSSIMp) is higher than the whole picture quality (MS-SSIM) compared to the original (f). According to the RoadDamageDataset results in Figs. 4 (c) and (d), the wPSNR of the damaged part of the BPG is 27.56, and the wPSNR of the proposed method is lower than that (27.20). Despite the low wPSNR of the proposed method, the damaged parts can be visually confirmed (they disappear in BPG). Therefore, it can be seen that wMSSSIM is more suitable in this case.

Table 2 shows the results of the average bpp of test image data under the same quality level conditions in the prioritized parts ($wMSSSIM_p$). The bpps are theoretical value calculated by the entropy estimator. Note that the theoretical values include small errors that are less than 0.1% in most image data compared to actual values. Compared to the method [6] without the proposed method (normal train), the method [6] with the proposed method reduces the amount of data by 31% on average even without receiving the annotation as the encoder input while the wMSSSIMs in the damaged parts are on the same level. The proposed method with the annotation input for the encoder reduces the amount of data by 34% on average. Table 3 shows results on ADE20K. As a result, we confirmed that the compression rate was improved by 19% than [6] with even wMSSSIMps are on the better condition. In the case of the car, even with a compression method such as BPG, the area of the car tends to have high wMSSSIM. However, even in this case, the proposed method provides a higher compression ratio than BPG with wMSSSIMps are on the same level.

4.3 Annotation Effects

Figures 5 (a)–(c) and Figs. 7 (a)–(c) show visualized impor-



(a) PNG (ground truth)

(b) JPEG (q = 5)
bpp: 0.299
$wMSSSIM_p$: 0.913
$wMSSSIM_{np}$: 0.899
$wPSNR_p$: 19.34
$wPSNR_{np}$: 21.78
$MS - SSIM$: 0.900
$PSNR$: 21.57

(c) BPG (4:4:4, q = 44)
bpp: 0.291
$wMSSSIM_p$: 0.960
$wMSSSIM_{np}$: 0.954
$wPSNR_p$: 22.43
$wPSNR_{np}$: 25.06
$MS - SSIM$: 0.955
$PSNR$: 24.83

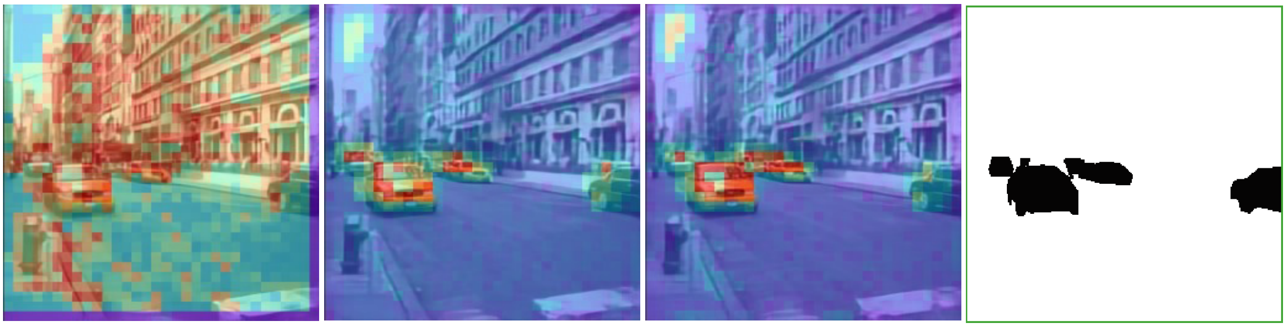
(d) Proposal train with [6]
(w/o input A)
bpp: 0.265
$wMSSSIM_p$: 0.981
$wMSSSIM_{np}$: 0.959
$wPSNR_p$: 23.29
$wPSNR_{np}$: 21.34
$MS - SSIM$: 0.961
$PSNR$: 21.45

(e) Proposal train with [6]
(w input A)
bpp: 0.265
$wMSSSIM_p$: 0.981
$wMSSSIM_{np}$: 0.959
$wPSNR_p$: 23.43
$wPSNR_{np}$: 21.36
$MS - SSIM$: 0.961
$PSNR$: 21.47

(f) Normal MS-SSIM train
with [6]
bpp: 0.337
$wMSSSIM_p$: 0.971
$wMSSSIM_{np}$: 0.977
$wPSNR_p$: 21.82
$wPSNR_{np}$: 22.78
$MS - SSIM$: 0.977
$PSNR$: 22.71

(g) BPG (4:4:4, q = 41)
bpp: 0.444
$wMSSSIM_p$: 0.978
$wMSSSIM_{np}$: 0.967
$wPSNR_p$: 24.88
$wPSNR_{np}$: 27.11
$MS - SSIM$: 0.968
$PSNR$: 26.93

Fig. 6 Experimental results (using ADE_val_00000855).



(a) Normal MS-SSIM train with [6]

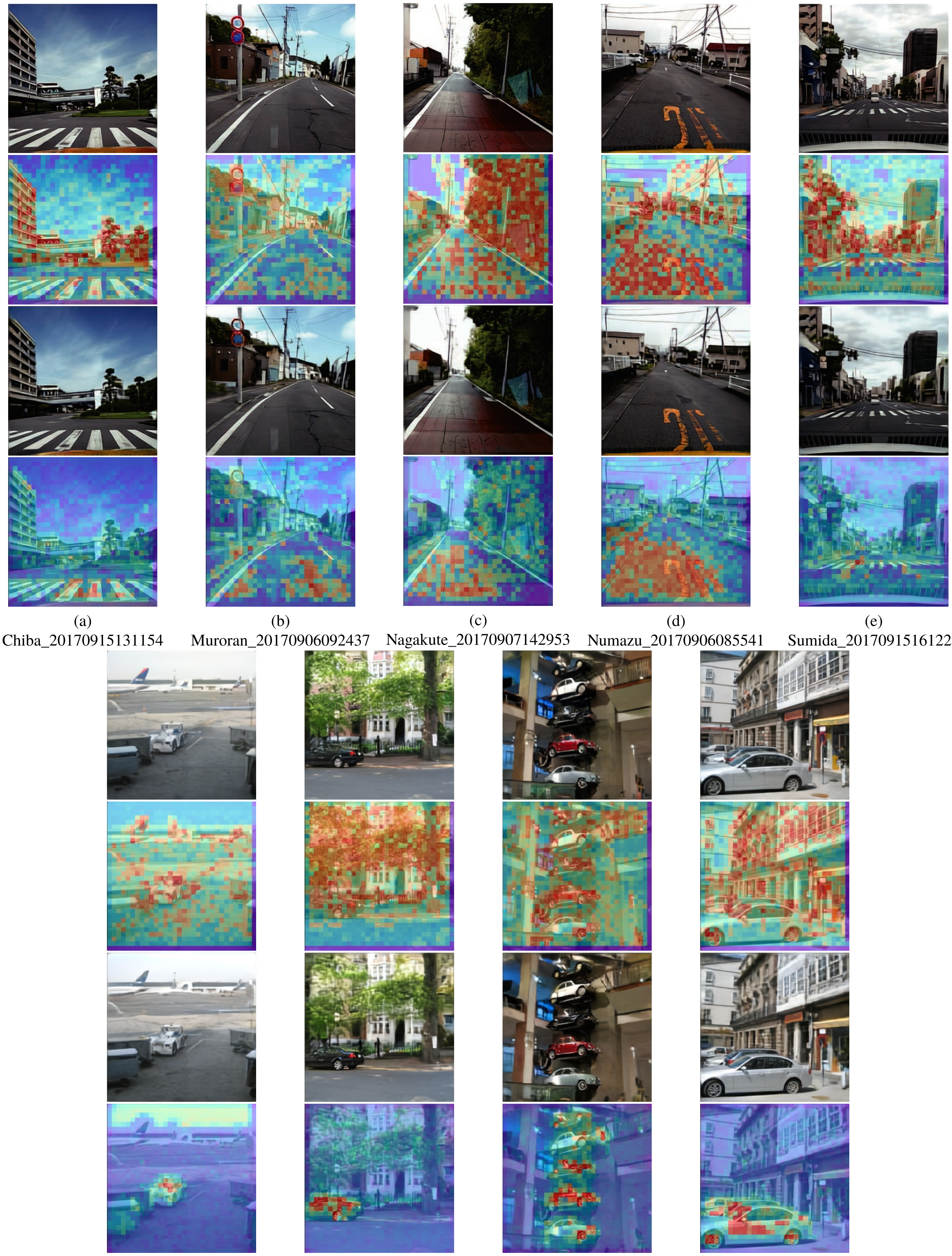
(b) Proposal train with [6] (w/o input A)

(c) Proposal train with [6] (w input A)

(d) Annotation

Fig. 7 Importance maps and an annotation information (using ADE_val_00000855).

tance maps of the reference image presented in Figs. 4 and 6, in which the red parts represent larger amount of bits. (a) is an importance map without a proposal method. (b) is an importance map when the network is trained by the proposed loss function without receiving the annotation as the encoder input. (c) is an importance map with the proposed method with the encoder annotation input. (d) is the ground truth annotation of the image where black represents priority parts. Compared to (a), (b) shows that the damaged parts have a lot of bit allocation and the parts without damage do



(a)
Chiba_20170915131154

(b)
Muroran_20170906092437

(c)
Nagakute_20170907142953

(d)
Numazu_20170906085541

(e)
Sumida_20170915161222

(a)
ADE_val_00000032

(b)
ADE_val_00000780

(c)
ADE_val_00001743

(d)
ADE_val_00001993

Fig. 8 Encoded images and importance maps (upper is normal MS-SSIM train with [6] and lower is proposal train with [6] w/o input A).

not. This means that by the proposed method, the network learns the characteristics of the damaged parts and it realizes automatic control of bit allocation. (c) shows a stronger correlation with the input annotation, which indicates that more specific bit allocation is possible with manual annotation input.

Another images of the experimental results are shown in Fig. 8. Upper is normal MS-SSIM train with [6] and lower is proposal train with [6] without input A. The results show that by the proposed method, the network learns the characteristics of the multiple types of damaged parts (e.g. paint damage, road crack) and car parts and realizes automatic control of bit allocation.

5. Conclusion

We proposed a method to improve the compression rate of CAE-based compression method while maintaining the given quality of the prioritized parts using annotation information that expresses the importance of each part of the image. We show experimental results using a road damage image dataset and ADE20K dataset.

The experimental results show that our weighted loss function enables CAE-based compression [6] to learn the characteristics and preferred bit allocations of the prioritized parts by end-to-end training. In the case of using road damage image dataset, our method reduces bpp by 31% compared to the original method [6] while maintaining the predetermined image quality in the parts.

Acknowledgments

We gratefully acknowledge Prof. Kiyoharu Aizawa (The University of Tokyo) for his expertise that greatly assisted our research and encouragement to write this paper.

References

[1] H. Akutsu and T. Naruko, “End-to-end learned roi image compression,” The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2019.

[2] ITU, “Harnessing the internet of things for global development,” 2016.

[3] IDC, “Data age 2025.” https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf, 2018.

[4] H. Maeda, Y. Sekimoto, T. Seto, T. Kashiyama, and H. Omata, “Road damage detection and classification using deep neural networks with smartphone images,” Computer-Aided Civil and Infrastructure Engineering, vol.33, no.12, pp.1127–1141, 2018.

[5] M. Li, W. Zuo, S. Gu, D. Zhao, and D. Zhang, “Learning convolutional networks for content-weighted image compression,” 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.3214–3223, June 2018.

[6] F. Mentzer, E. Agustsson, M. Tschannen, R. Timofte, and L. Van Gool, “Conditional probability models for deep image compression,” 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, pp.4394–4402, 2018.

[7] D. Minnen, J. Ballé, and G.D. Toderici, “Joint autoregressive and hierarchical priors for learned image compression,” Thirty-second Conference on Neural Information Processing Systems, NeurIPS 2018, 3-8 Dec. 2018, Montréal, Canada, pp.10794–10803, 2018.

[8] Z. Wang, E.P. Simoncelli, and A.C. Bovik, “Multiscale structural similarity for image quality assessment,” Conference Record of the Asilomar Conference on Signals, Systems and Computers, pp.1398–1402, 2003.

[9] E. Agustsson, M. Tschannen, F. Mentzer, R. Timofte, and L.V. Gool, “Generative adversarial networks for extreme learned image compression,” CoRR, vol.abs/1804.02958, 2018.

[10] C. Christopoulos, J. Askelof, and M. Larsson, “Efficient methods for encoding regions of interest in the upcoming jpeg2000 still image coding standard,” IEEE Signal Process. Lett., vol.7, no.9, pp.247–249, Sep. 2000.

[11] F. Bellard, “Bpg image format.” https://bellard.org/bpg/.

[12] Z. Wang and Q. Li, “Information content weighting for perceptual image quality assessment,” IEEE Trans. Image Process., vol.20, no.5, pp.1185–1198, May 2011.

[13] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, “Scene parsing through ade20k dataset,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp.5122–5130, 2017.

Hiroaki Akutsu rceived his M.Eng. and Dr.Eng. degrees from Waseda University in 2005 and 2017, respectively. He now with Hitachi,Ltd. Research & Development Group, Data Storage Research Dept. working as a Senior Researcher.

Takahiro Naruko rceived his M.S. degree in Computer Science from the University of Tokyo in 2016. He now with Hitachi,Ltd. Research & Development Group, Data Storage Research Dept. working as a Researcher.