End-to-End Learned ROI Image Compression

Hiroaki Akutsu Takahiro Naruko
Research & Development Group
Hitachi, Ltd., Japan
hiroaki.akutsu.cs@hitachi.com

Abstract

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 use road images used to check damaged parts in the road. Our evaluation reveals that BPG does not provide adequate quality for the road damaged parts at a low bit rate (1.0 bpp or less). 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. Experimental results show that the proposed method implemented for CAE-based compression from F. Mentzer et al. learns the characteristics of the road damaged parts by end-to-end training with the weighted loss function and 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.

1. Introduction

Image data generated by digital devices is enormous and is generated at every moment. To transfer and store this increasing data, technology that provides a high compression ratio for the data is needed. In this paper, we present the effectiveness of image compression based on a convolutional auto encoder (CAE) with region of interest (ROI) in a use case in which images taken by on-vehicle cameras are used to check damaged regions on the roads for maintenance work.

2. Related works

2.1. Convolutional autoencoder based image compression

Leading research [4, 6, 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 [9] (which qualifies structural similarities) are used as the loss function. CAE-based image compression methods such as [4, 6] automatically learn to adjust to the bit rate necessary for each part by using a technique called an "importance map" with end-to-end learning.

Selective generative compression [1] generates portions of images by a generative adversary network (GAN) to improve a compression rate further. This method can dramatically improve a compression rate up to 0.1 bpp or less 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) [3]. 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 impor-

$q = 32$	$q = 36$	$q = 40$	$q = 44$
(bpp: 1.164)	(bpp: 0.754)	(bpp: 0.457)	(bpp: 0.255)

Figure 1. Road damage images at different qualities encoded by BPG (using Adachi_20170906093840 [5]).

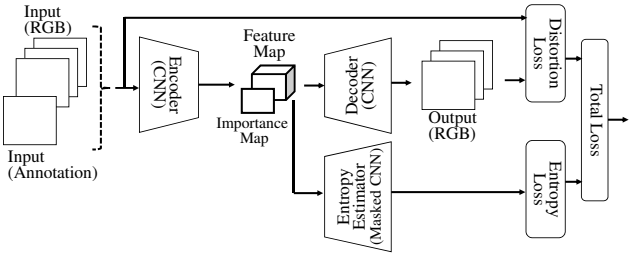


Figure 2. Network Architecture Overview.

tant part in the end-to-end manner from the training data with annotation.

BPG [2] 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 to the road images. We assumed that end-to-end learning gives the CAE-based methods better compression rates compared to conventional methods.

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 that changes parameters of an image quality metric for each area according to the annotation information.

2. Append a new encoder input channel and feed it annotation information for manual quality control (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.

3.2. Loss function

We defined weighted MS-SSIM (wSSIM), image quality metrics that reflect annotation information. They are used to evaluate image quality 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 [9].

Let $\mathbf{x}_{j,i}$ and $\mathbf{y}_{j,i}$ 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 ($wSSIM$) is computed as

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

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. 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 $\mathbf{a}_{i,j}$.

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 [8] that uses information content weight with MS-SSIM. 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 [4, 6, 7].

$$\mathcal{L} = \mathcal{L}_e + \lambda \mathcal{L}_d \quad (2)$$

$\mathcal{L}_e$ 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.

$$\mathcal{L}_d = \max(1 - wSSIM_p, T_p) + \max(1 - wSSIM_{np}, T_{np}) \quad (3)$$

$wSSIM_p$ and $wSSIM_{np}$ represent prioritized parts and non-prioritized parts of image quality calculated by Eqn 1. Target distortion of priority parts T_p and non-priority parts T_{np} are given by quality settings. In our experiment, A has a constant positive value c for the priority area and 0 for the non-priority area. $wSSIM_p$ is calculated with A, and $wSSIM_{np}$ is calculated with the inverse of A. However, if a priority or non-priority area does not exist, $wSSIM$ cannot be calculated because $\sum_i a_{i,j}$ equals 0. To avoid this problem, a sufficiently small coefficient is added to A during training.

4. Experiments

We evaluated the effectiveness of our method with the RoadDamageDataset [5], which is a dataset of images that contain damaged parts of a road.

4.1. Experimental conditions

The RoadDamageDataset contains annotation information and image data, which are downsampled to 256 x 256 pixels in this evaluation. 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 when evaluating BPG.

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	6,925 files from [5] (Width:256, Height:256)
Test data	1,811 files from [5] (exclude images with no damaged parts) (Width:256, Height:256)
Quality settings	$T_p = 0.03$ ($wSSIM_p = 0.97$), $T_{np} = 0.05$ ($wSSIM_{np} = 0.95$)

Table 1. Experimental conditions.

Method	bpp	$wSSIM_p$	$wSSIM_{np}$
Proposal train (w input A)	0.251	0.970 (26.78)	0.952 (22.06)
Proposal train (w/o input A)	0.263	0.970 (27.04)	0.953 (21.98)
Normal train	0.382	0.970 (27.81)	0.973 (23.55)
BPG ($q = 32$)	1.183	0.970 (32.93)	0.985 (33.39)

Table 2. Experimental results (averages of test data; PSNR in parentheses).

4.2. Results

The experimental results are shown in Figure 3 (a)-(g). In the conventional codecs like JPEG (b) and BPG (c), the damaged portion of the road on the lower left disappears. With the CAE-based compression [6] trained by our methods ((d) and (e)), the portion has a higher quality compared to the conventional method under the same level bpp conditions. Compared to method [6] without our methods (f) and BPG (g) with same level quality conditions of the prioritized portion, our methods ((d) and (e)) reduces bpp.

Table 2 shows the results of the average bpp of test image data under the same quality level conditions in the priority parts ($wSSIM_p$). In Table 2, 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 wSSIMs 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.

4.3. Annotation effects

Figures 4 (a)–(c) show visualized importance maps of the reference image, in which the black 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 annotation of the image where black represents priority parts. Compared to (a), (b) show that the damaged parts have a lot of bit allocation and the parts without damage are do not. This means that by the proposed method, the network learns the characteristics of the damaged parts and it realizes automatic control of prioritized bit allocation. (c) show a stronger correlation with

(a) PNG (ground truth)

(b) JPEG ($q = 6$)
bpp: 0.280
$wSSIM_p$: 0.808 (26.02)
$wSSIM_{np}$: 0.898 (21.96)

(c) BPG (4:4:4, $q = 43$)
bpp: 0.291
$wSSIM_p$: 0.845 (27.56)
$wSSIM_{np}$: 0.943 (25.34)

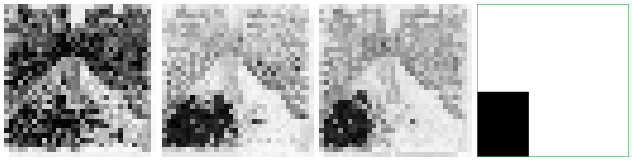
(d) Proposal train with [6] (w/o input A)
bpp: 0.263
$wSSIM_p$: 0.976 (27.20)
$wSSIM_{np}$: 0.957 (21.35)

(e) Proposal train with [6] (w input A)
bpp: 0.252
$wSSIM_p$: 0.976 (27.42)
$wSSIM_{np}$: 0.956 (21.31)

(f) Normal MS-SSIM train with [6]
bpp: 0.384
$wSSIM_p$: 0.971 (27.97)
$wSSIM_{np}$: 0.975 (22.89)

(g) BPG (4:4:4, $q = 31$)
bpp: 1.286
$wSSIM_p$: 0.971 (34.02)
$wSSIM_{np}$: 0.986 (34.39)

Figure 3. Experimental results. (using Adachi_20170906093840, PSNR in parentheses).



(a) (b) (c) (d)

Figure 4. Importance maps and an annotation information (using Adachi_20170906093840).

the input annotation, which indicates that more specific bit allocation with manually annotation input.

5. Conclusion

We evaluated image compression methods using images of roads that had damaged parts. The results revealed that the image quality and compression rate of the damaged parts improved by neural network-based compression techniques that use MS-SSIM as a loss function over the conventional image compression (BPG).

We proposed a method to improve the compression rate while maintaining the given quality of the parts using annotation information that expresses the importance of each part of the image.

Combining our method with CAE-based compression [6] learns the characteristics of the road damaged parts by end-to-end training with the weighted loss function and reduces bpp by 31% compared to the original method [6] while maintaining the predetermined image quality in the parts.

Acknowledgement

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

References

[1] Eirikur Agustsson, Michael Tschannen, Fabian Mentzer, Radu Timofte, and Luc Van Gool. Generative adversarial

networks for extreme learned image compression. *CoRR*, abs/1804.02958, 2018.

[2] Fabrice Bellard. Bpg image format. https://bellard.org/bpg/.

[3] Charilaos Christopoulos, Joel Askelof, and Mathias Larsson. Efficient methods for encoding regions of interest in the upcoming jpeg2000 still image coding standard. *IEEE Signal Processing Letters*, 7(9):247–249, Sep. 2000.

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

[5] Hiroya Maeda, Yoshihide Sekimoto, Toshikazu Seto, Takehiro Kashiyama, and Hiroshi Omata. Road damage detection and classification using deep neural networks with smartphone images. *Computer-Aided Civil and Infrastructure Engineering*, 2018.

[6] Fabian Mentzer, Eirikur Agustsson, Michael Tschannen, Radu Timofte, and Luc Van Gool. Conditional probability models for deep image compression. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 4394–4402, 2018.

[7] David Minnen, Johannes Ballé, and George D Toderici. Joint autoregressive and hierarchical priors for learned image compression. In *Thirty-second Conference on Neural Information Processing Systems, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 10794–10803, 2018.

[8] Zhou Wang and Qiang Li. Information content weighting for perceptual image quality assessment. *IEEE Transactions on Image Processing*, 20(5):1185–1198, May 2011.

[9] Zhou Wang, Eero P. Simoncelli, and Alan C. Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*, volume 2, pages 1398–1402 Vol.2, Nov 2003.