https://doi.org/10.2352/CIC.2024.32.1.10

Neural Image Compression and the Degradation of Color: An Analysis

Ian MacPherson, Trevor Canham, and Michael S. Brown
York University, Toronto, Canada

Abstract

Neural image compression employs deep neural networks and generative models to achieve impressive compression rates and reconstruction qualities compared to traditional signal-processing-based compression algorithms such as JPEG. However, color artifacts that arise in an image as the amount of compression increases have not been formally analyzed for neural-based compression algorithms. This paper provides an initial investigation into the degradation of color when images are compressed at comparable bit rates using lossy neural image compression and variants of JPEG. Our findings indicate that neural image compression degrades color more gracefully than JPEG, JPEG 2000, and JPEG XL.

Introduction

Lossy neural image compression is an emerging research area that has shown promise in achieving more optimal rate-distortion (RD) trade-offs than prominent classical compression algorithms such as JPEG. While JPEG compression's impact on image quality has been thoroughly studied [1, 13, 18], the effect on image characteristics and the type of artifacts that arise is less understood for neural compression methods. Neural compression achieves impressive compression rates through data-driven training that allows the deep network to produce plausible hallucinations to recover the image content. This neural-based approach contrasts sharply with JPEG's frequency domain quantization approach.

The impetus for this paper is to explore how color is impacted by neural compression methods as the compression rate increases (i.e., bit rates are reduced). In particular, we focus on lossy neural compression methods that are trained to optimize Shannon's RD objective [12], where distortion is evaluated using the peak signal-to-noise ratio (PSNR) in a standard RGB color space (sRGB), and the rate is measured in bits per pixel (bpp).

Specifically, we examine neural compression models provided in the CompressAI library [3] against JPEG [16], JPEG 2000 [15], and JPEG XL [11] on 100 randomly selected images from the Vimeo90k [17] test set. We analyze how the original image's color —in $L^*a^*b^*$ space— changes as the image is compressed at different bit rates by neural image compressors and JPEG variants. To provide a quantitative metric to measure the degradation in color representation, we compare the difference in *unique* a^*b^* coordinates of the images in $L^*a^*b^*$ space. This aims to provide a deeper understanding of the amount of hallucination from the neural compressors (i.e., variance in unique a^*b^* coordinates between compression levels to quantify hallucination or color degradation). Fig. 1 shows an example of the reduction in a^*b^* coordinates between JPEG and a neural compressor at comparable bit rates. In addition, our analysis includes examining each method's average Delta E 2000 (ΔE_{00}) error across different compression levels to provide an estimate of perceptual color distortion.

Figure 1: Comparison of the a^*b^* coordinates between an original image and its compressed counterparts using a neural compressor [2] and JPEG compression [16].

Our analysis found that neural compression schemes were much more graceful in their degradation of color than JPEG, JPEG 2000, and JPEG XL. Overall, neural compressors provide competitive RD trade offs and gracefully degrade color across all methods analyzed, especially at bit rates lower than those achievable by the JPEG methods tested. The remainder of this paper describes related work, the methodology for analysis, and our findings.

Related Work

JPEG remains the most used form of lossy compression for image data. Standardized in 1992, JPEG transforms non-overlapping eight-by-eight image patches using the discrete cosine transform (DCT) and achieves compression by discarding high-frequency information and quantizing the remaining information referenced to a quantization table before entropy coding.

JPEG 2000 operates similarly to JPEG, with the predominant difference being that the method opts for the discrete wavelet transform in place of DCT and provides benefits in the flexibility of the code stream. JPEG XL supports both lossy and lossless compression, offering better compression rates than JPEG, and can be applied to images with dimensions over one billion [14]. Similar to the original JPEG implementation, JPEG XL uses DCT

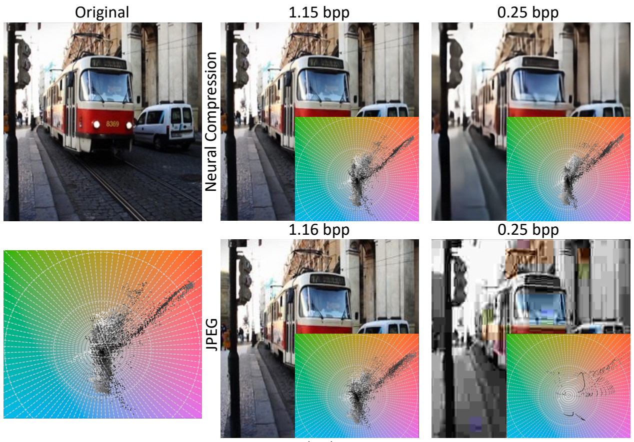

Figure 2: Comparison of the a^*b^* coordinates between an original image and its compressed counterparts using a neural compressor [2] and JPEG compression [16].

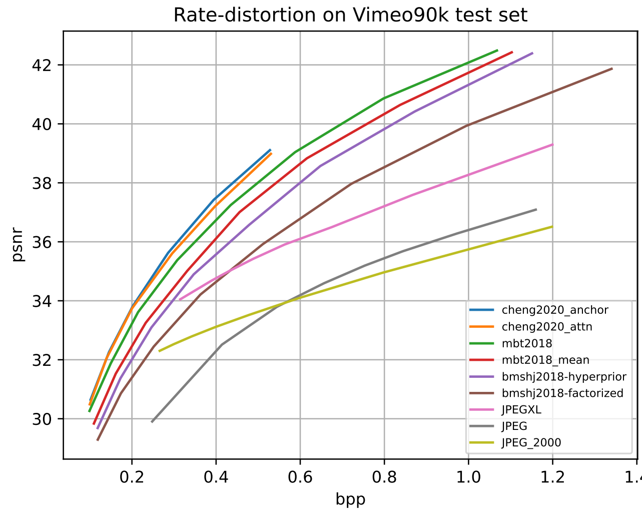

Figure 3: Rate-distortion values for each compression method in this analysis. The neural compressors perform much better concerning rate distortion at and beyond comparable bit rates to JPEG, JPEG 2000, and JPEG XL.

blocks. However, the blocks can be of varying sizes. Note that while JPEG 2000 and JPEG XL are feature-rich, they have not achieved the same adoption as the original JPEG.

Neural image compression models are commonly based on a deep learning architecture known as the autoencoder, where the first part of the network, the encoder, transforms the image from pixel space to a learned latent representation. The image is reconstructed from this latent representation back to pixel space by a mirrored set of operations known as the decoder. To effectively compress the image using neural networks, the entropy of the latent representation is used to compress the latent code into a byte string using deterministic entropy coding methods, accompanied by entropy models (i.e., because the latent representation does not guarantee a reduction in entropy). A quantization step usually accompanies this and further discards information in the compression process.

More recent works have employed generative models such as Generative Adversarial Networks (GANs) [5], variational autoencoders (VAEs) [8], and diffusion models [6] to enhance reconstruction quality [9, 7]. Generative models are particularly well positioned for lossy image compression as they are capable of hallucinating information discarded in the encoding process. However, such methods are also prone to hallucinating information that does not exist in the original image.

CompressAI is a Python library that provides access to neural image compression model architectures and sets of pre-trained weights for each model. The models provided in the library were introduced in the following works: “Variational Image Compression With A Scale Hyperprior” [2], which uses a hyperprior network along with a VAE; “Joint Autoregressive and Hierarchical Priors for Learned Image Compression” [10], which explores the use of alternate entropy models; and “Learned Image Compression with a Discretized Gaussian Mixture Likelihoods and Attention Modules” [4], which uses discretized Gaussian Mixture Likelihoods to parameterize the distributions of latent codes to achieve a more accurate entropy model. The work in [4] uses attention-based models to increase performance but requires much more training time due to the computational complexity of the transformer architecture. Due to the computational complexity of training, the CompressAI library provides a smaller set of pre-trained weights for the attention-based models.

For neural compression algorithms, a single set of weights is required for a respective compression level, which has been explicitly optimized for an RD trade-off. The CompressAI library provides the ability to select from six to eight pre-trained weights, where each set of weights is referred to as a “quality level” or a hand-selected point on the RD curve (i.e., one and eight refer to the lowest and highest bit rate and PSNR, respectively). These quality levels are ambiguous, as they are arbitrarily chosen by the developers of the CompressAI library and relate to how the optimization function for the neural compressors is weighted to permit distortion at training time.

The cost function that optimizes the neural compression models in the CompressAI library, where D denotes mean squared error (MSE), and R denotes bits per pixel (bpp), is given by :

$$L = \lambda * 255^2 * D + R, \tag{1}$$

where λ is a hyper-parameter that determines the amount of permissible distortion in the reconstruction (i.e., a lower λ value allows for more significant distortion, and a higher λ allows for less distortion). Eight different lambda parameters are associated with a particular “quality level.” In the case of CompressAI, quality level 1 refers to the most error permissible in the image reconstruction (i.e., low bit-rate, high distortion), and quality level 8 refers to the least amount of error permissible (i.e., high bit rate, low distortion) in the image reconstruction. It is worth noting that other image-quality metrics can be substituted to compute distortion D.

Method

Three unique neural image compression models and their two variants were used to compress 100 randomly selected images from the test split of the Vimeo90k dataset. Three variants of JPEG were also tested on the same set of images to provide a comparison to classical compression methods. Those methods were

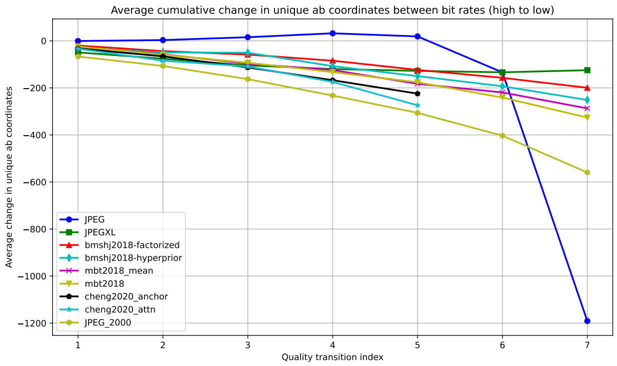

Figure 4: Average cumulative change in unique a^*b^* coordinates across 100 randomly selected images in the Vimeo90k test set between compression rates (i.e., low to high) for neural image compressors, JPEG, JPEG 2000, and JPEG XL. The x-axis, denoted "Quality transition index," determines the change between one compression "quality" level to the next (i.e., Eight quality levels, seven transitions in quality. Transition index "1" denotes the transition from compression quality eight to quality seven, "2" from seven to six, and so on). The y-axis shows the average change in unique a^*b^* coordinates across the test set from one quality to the next lowest quality.

JPEG, JPEG 2000, and JPEG XL. The 100 test images were randomly selected from the Vimeo90k test portion, as the dataset's training portion optimizes the pre-trained weights provided by the CompressAI library for each neural image compressor. Therefore, testing on this data provides a fair comparison between the neural compression algorithms and JPEG variants.

Each 100 randomly selected images from the Vimeo90k test set were compressed and reconstructed at each of the eight quality levels available for the neural compressors. For the JPEG variants, parameters were chosen to most closely match the bit rates spanned by the neural compressors as seen in Fig. 3 to provide a fair comparison. However, since the neural compression algorithms can compress at ratios greater than JPEG, there are examples of processed images from the neural compressors at much lower bit rates than achievable with JPEG-based methods.

Images compressed by each method were then converted from sRGB to $L^*a^*b^*$ and plotted for comparison. The a^*b^* coordinates were quantized from floating point representation to integers such that there are 255 unique values for a^* and b^* respectively, and the number of unique a^*b^* coordinates per image was computed to generate the quantitative results. The difference in the number of unique a^*b^* coordinates is computed among quality levels for each method and used as a metric to measure the reduction or hallucination of total unique colors between compression levels. In other words, this visualization provides an understanding of the capacity of compression methods to represent color as compression increases and image quality is reduced. The average color difference was also computed using ΔE_{00} between the original 100 images from the test dataset and their compressed counterparts for each method and compression level. ΔE_{00} is used as a proxy to quantify the perceptual color difference between the original image and its compressed version. The ΔE_{00} results can be seen in Fig. 5.

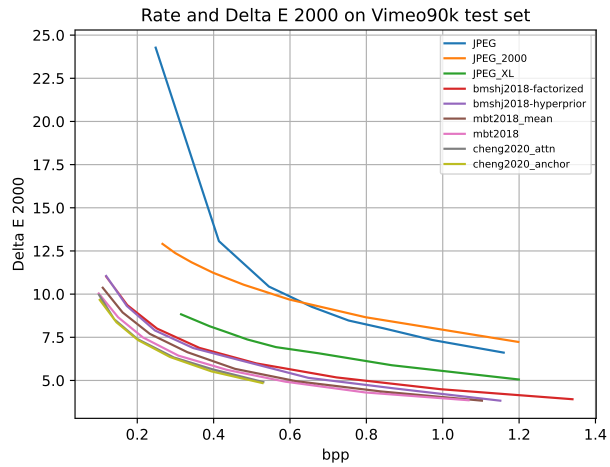

Figure 5: The rate-distortion values for each compression method in this analysis, where distortion is measured via ΔE_{00}. The neural compressors perform better at and beyond comparable bit rates to JPEG, JPEG 2000, and JPEG XL.

Results

Fig. 4 shows the average change in unique a^*b^* coordinates from the highest quality (i.e., highest bit-rate and lowest distortion for each method) to the lowest quality (i.e., lowest bit-rate and high distortion for each method). The x-axis denotes the change from one quality level to the next, and is referred to as the "Quality transition index". This means that a plot point at quality transition index "1" denotes the increase or reduction in unique a^*b^* coordinates from quality eight to quality seven for each method. The rest of the quality indices continue this pattern in descending order of compression quality. This plot further shows a cumulative change, meaning quality transition index seven denotes the total unique a^*b^* coordinates hallucinated or reduced across all compression quality levels for the respective method.

It can be observed that the standard deviation of the neural compression algorithms is much lower between quality transition levels than JPEG, especially for low-bit rates (i.e., increased compression). In the mid-bit-rate regime, JPEG distorts the unique a^*b^* coordinates beyond those in previous compression levels. Interestingly, JPEG XL performs best overall in unique a^*b^* coordinates retained by the highest compression rate (i.e., lowest bit rate, highest distortion), albeit at a worse compression ratio than the neural compressors. JPEG 2000 performs significantly better than JPEG but worse than the neural compression algorithms. In contrast, the neural image compression algorithms behave more consistently and similarly reduce the unique a^*b^* coordinates across the different methods. While this metric does not elucidate the types of hallucinations caused by the neural compressors, it shows that neural compression reduces the number of unique a^*b^* coordinates in the image less than that incurred when using JPEG or JPEG 2000 at and beyond comparable bit-rates and especially in the low-bit-rate regime.

Fig. 1 and Fig. 2, show the a^*b^* coordinates of an image compressed at comparable bit rates via neural compression and JPEG. While the neural compressor modifies the distribution and reduces the number of unique colors measured by unique a^*b^* coordinates, the degradation is far more "graceful" than JPEG, which seems to collapse to a small set of hues in the low-bit-rate regime. Fig. 5 shows the average ΔE_{00} between the original 100 images used for testing and the output of each compression method across bit rates. The plot shows that the neural compressors perform better in terms of ΔE_{00} in the low-bit-rate regime and at bit rates comparable to JPEG, JPEG 2000, and JPEG XL. Also, the JPEG color degradation at low bit rates is much more significant according to ΔE_{00} than PSNR.

Concluding Remarks

This paper analyzes the color degradation of neural image compressors compared to JPEG, JPEG 2000, and JPEG XL. Our analysis found that when compared at similar bit rates, neural image compressors degrade color more gracefully than the JPEG variants. In particular, the number of unique a^*b^* color coordinates remained significantly more consistent, and ΔE_{00} values were lower as the compression rate was increased than observed in the tested JPEG variants. JPEG XL's performance in the retention of color as bit rates reduced was impressive, albeit at higher bit rates than achieved by the neural compressors. Overall, the neural compressors behaved consistently between methods, whereas there was more variability between JPEG methods. As neural compressors become more prominent, it is important to understand the types of artifacts they introduce as compression levels vary.

We note that many of the neural compression methods used in this analysis have been superseded in favor of diffusion and GAN-based compression models that perform better with respect to RD. However, training a diffusion-based neural compressor model for a particular RD performance often requires weeks on a large GPU cluster. This need for substantial computing resources hinders the ability to effectively analyze many emerging diffusion and GAN-based neural compressors to understand their characteristics at different quality levels.

Acknowledgements

This work was supported by the Canada First Research Excellence Fund for the Vision: Science to Applications (VISTA) program, an NSERC Discovery Grant, and the Canada Research Chair program.

References

[1] Hazem Munawer Al-Otum. Qualitative and quantitative image quality assessment of vector quantization, jpeg, and JPEG2000 compressed images. *Journal Electronic Imaging*, 2003.

[2] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. "Variational image compression with a scale hyperprior". In *ICLR*, 2018.

[3] Jean Bégaint, Fabien Racapé, Simon Feltman, and Akshay Pushparaja. "Compressai: a pytorch library and evaluation platform for end-to-end compression research". *CoRR*, abs/2011.03029, 2020.

[4] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. "Learned image compression with discretized gaussian mixture likelihoods and attention modules"'. In *CVPR*, 2020.

[5] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. "Generative adversarial networks". *CoRR*, abs/1406.2661, 2014.

[6] Jonathan Ho, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models". In *NeurIPS*, 2020.

[7] Emiel Hoogeboom, Eirikur Agustsson, Fabian Mentzer, Luca Versari, George Toderici, and Lucas Theis. "High-fidelity image compression with score-based generative models". *CoRR*, abs/2305.18231, 2023.

[8] Diederik P. Kingma and Max Welling. "Auto-encoding variational bayes". In *ICLR*, 2014.

[9] Fabian Mentzer, George D Toderici, Michael Tschannen, and Eirikur Agustsson. "High-fidelity generative image compression". 2020.

[10] David Minnen, Johannes Ballé, and George Toderici. "Joint autoregressive and hierarchical priors for learned image compression". In *NeurIPS*, 2018.

[11] Alexander Rhatushnyak, Jan Wassenberg, Jon Sneyers, Jyrki Alakuijala, Lode Vandevenne, Luca Versari, Robert Obryk, Zoltan Szabadka, Evgenii Kliuchnikov, Iulia-Maria Comsa, Krzysztof Potempa, Martin Bruse, Moritz Firsching, Renata Khasanova, Ruud van Asseldonk, Sami Boukortt, Sebastian Gomez, and Thomas Fischbacher. Committee draft of JPEG XL image coding system. *CoRR*, abs/1908.03565, 2019.

[12] C. E. Shannon. Coding theorems for a discrete source with a fidelity criterion. In *IRE National Convention Record*, 1959.

[13] Hamid R. Sheikh, Zhou Wang, and Alan C. Bovik. No-reference perceptual quality assessment of JPEG compressed images. In *ICIP*, 2002.

[14] Jon Sneyers. How JPEG XL Compares to Other Image Codecs. https://cloudinary.com/blog/how_jpeg_xl_compares_to_other_image_codecs, 2020. [Accessed 12-09-2024].

[15] David S. Taubman and Michael W. Marcellin. *JPEG2000 - image compression fundamentals, standards and practice*. Kluwer, 2002.

[16] Gregory K. Wallace. "The JPEG still picture compression standard". *Communications of the ACM*, 1991.

[17] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T. Freeman. "Video enhancement with task-oriented flow". *IJCV*, 2019.

[18] Guangtao Zhai and Xiongkuo Min. Perceptual image quality assessment: a survey. *Science China Information Sciences*, 2020.

Author Biography

Ian MacPherson *is a Ph.D. student in the Department of Electrical Engineering & Computer Science at York University in Toronto, Canada. He is being co-supervised by Michael S. Brown and Marcus A. Brubaker. Previously, he completed his Master's degree under the supervision of Michael S. Brown. His research interests focus on neural image compression and the in-camera processing pipeline.*

Trevor Canham *received the BSc in Motion Picture Science from the Rochester Institute of Technology. In the following years he worked in Marcelo Bertalmío's Image Processing for Enhanced Cinematography lab in Barcelona, Spain, and later at the Spanish National Research Council. He has also worked in the cinema industry - as an intern at Company 3 NY and later as a colorist for several independent films. In 2023 he was awarded best student paper at the 31st Color & Imaging Conference and received the graduate student award from the Colour Research Society of Canada. His interests lie in the interaction between color phenomenology and imaging systems.*

Michael S. Brown *is a professor and Canada Research Chair at York University. His research focuses on in-camera color processing algorithms for photographic and scientific applications.*