International Journal of Internet, Broadcasting and Communication Vol.16 No.2 22-30 (2024)
http://dx.doi.org/10.7236/IJIBC.2024.16.2.22

IJIBC 24-2-3

Denoising Diffusion Null-space Model and Colorization based Image Compression

Indra Imanuel †, Dae-Ki Kang††, and Suk-Ho Lee†††

†*Doctoral Degree Candidate, Dept. Computer Engineering, Dongseo University, Korea*
††*Professor, Dept. Computer Engineering, Dongseo University, Korea*
†††*Professor, Dept. Computer Engineering, Dongseo University, Korea*
Corresponding Author: Sukho Lee
E-mail petrasuk@gmail.com

Abstract

Image compression-decompression methods have become increasingly crucial in modern times, facilitating the transfer of high-quality images while minimizing file size and internet traffic. Historically, early image compression relied on rudimentary codecs, aiming to compress and decompress data with minimal loss of image quality. Recently, a novel compression framework leveraging colorization techniques has emerged. These methods, originally developed for infusing grayscale images with color, have found application in image compression, leading to colorization-based coding. Within this framework, the encoder plays a crucial role in automatically extracting representative pixels—referred to as color seeds—and transmitting them to the decoder. The decoder, utilizing colorization methods, reconstructs color information for the remaining pixels based on the transmitted data.

In this paper, we propose a novel approach to image compression, wherein we decompose the compression task into grayscale image compression and colorization tasks. Unlike conventional colorization-based coding, our method focuses on the colorization process rather than the extraction of color seeds. Moreover, we employ the Denoising Diffusion Null-Space Model (DDNM) for colorization, ensuring high-quality color restoration and contributing to superior compression rates. Experimental results demonstrate that our method achieves higher-quality decompressed images compared to standard JPEG and JPEG2000 compression schemes, particularly in high compression rate scenarios.

Keywords: *Colorization, Image Compression, Deep Learning, Denoising Diffusion, Null Space*

1. Introduction

Modern advancements have elevated the significance of image compression-decompression, allowing users to transfer high-quality images while minimizing the file size or internet traffic. In the initial era of the image

Manuscript Received: March. 5, 2024 / Revised: March. 13, 2024 / Accepted: March. 19, 2024
Corresponding Author: petrasuk@gmail.com
Tel: +82-51-320-1744, Fax: +82-51-327-8955
Professor, Department of Computer Engineering, General graduate school, Dongseo University, Korea

compression, rudimentary codecs were employed for image compression[1][2][3]. The compression-decompression process entails compressing data, transmitting them with minimal internet traffic consumption, and subsequently decompressing them. The primary objective is to achieve a negligible disparity between the original and decompressed images, ensuring that the image quality post-compression-decompression remains consistent with that before data transfer. The JPEG format which was adopted by the Joint Photographic Experts Group is a standard image format designed for lossy and compressed image data. Originating in the early '90s, JPEG has since become the preeminent image compression standard worldwide [4]. At the core of JPEG's lossy compression algorithm lies the discrete cosine transform, a mathematical operation converting each frame/field of the video source from the 2D spatial domain into the frequency domain. The JPEG standard delineates the codec, dictating how an image is compressed into a byte stream and subsequently decompressed back into its original form.

Recently, a new compression framework, utilizing colorization techniques have emerged[5][6][7]. Colorization methods [8] are dedicated to infusing grayscale images with color based on a sparse set of user-provided representative pixels. These colorization techniques have found application in the realm of image compression, giving rise to what is known as colorization-based coding.

In the context of colorization-based compression, a key task involves the automatic extraction of these representative pixels within the encoder. Effectively, the encoder is tasked with identifying and selecting the pixels crucial for the colorization process—referred to as color seeds in [8]. Subsequently, the encoder retains color information exclusively for this set of color seeds. The position vectors and chrominance values for these color seeds are transmitted to the decoder, along with the compressed luminance channel utilizing conventional compression techniques. In this process, the decoder then employs colorization methods to reconstruct the color information for the remaining pixels, completing the restoration process initiated by the encoder's selective retention of color seeds.

Meanwhile, recently, new image restoration frameworks using denoising diffusion models have been emerged[14]. For example, in [9], the authors introduced an innovative approach termed the Denoising Diffusion Null-Space Model (DDNM) as a zero-shot solution for diverse image restoration tasks. The uniqueness of their methodology lies in the refinement solely of the null-space contents during the reverse diffusion sampling. Remarkably, their solution achieves realistic and data-consistent outcomes without necessitating additional training, optimization, or any modifications to network structures, relying solely on an off-the-shelf diffusion model.

The Range-Null space decomposition [10] introduces a new tool for the interplay between realness and data consistency. Notably, data consistency is exclusively tied to the range-space contents, and these can be precisely computed through analytical methods. Consequently, the data term can be ensured, shifting the primary challenge towards identifying suitable null-space contents that ensure the resultant output aligns with real-world expectations. An intriguing observation is that emerging diffusion models [11][12] serve as optimal tools for generating ideal null-space contents. This is attributed to their capacity for explicit control over the generation process, making them conducive to achieving desired outcomes.

In this paper, we utilize the concept of colorization based coding and decompose the compression task into the gray-scale image compression task and the colorization task as is usual in colorization based coding. The main difference with conventional colorization based coding is that we do not concentrate on the extraction of the color seed but rather on the colorization process, and that we use for the colorization the Denoising Diffusion Null-Space Model (DDNM) proposed in [9]. By utilizing the DDNM, we get a high quality of color restoration, we again contributes to a good compression rate of the compression. Experimental results show that the proposed method can achieve decompressed images of higher quality than the standard JPEG and JPEG2000 compression schemes in high compression rate situations.

2. Denoising Diffusion Null-Space Model(DDNM)

The DDNM is based on the DDPM(Denoising Diffusion Probabilistic Model). The DDPM consists of a forward process and a reverse process. In the forward process, the image undergoes a gradual transformation into Gaussian noise, while the reverse process is dedicated to reconstructing Gaussian noise back into an image.

Specifically, starting from a clean image $\mathbf{x}_0$, the forward process gradually adds noise to the image $\mathbf{x}_t$ at every time step t, so that $\mathbf{x}_t$ gradually transforms into a gaussian noise as $t \rightarrow T$, where T is the total iteration number of the forward process. Then, in the reverse process, a gaussian noise is sampled at t=T, then, $\mathbf{x}_t$ is gradually denoised at every time step t so that as $t \rightarrow 0$, $\mathbf{x}_t$ becomes cleaner and cleaner, and finally at t=0, a clean image $\mathbf{x}_0$ is generated.

The relationship between $\mathbf{x}_t$ and $\mathbf{x}_{t-1}$ in the forward process as proposed in [14] is expressed as

$$\mathbf{x}_t = \sqrt{\alpha_t}\mathbf{x}_{t\text{-}1} + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_t \tag{1}$$

Here, α_t is a predefined positive value which is smaller than 1. Therefore, the content in $\mathbf{x}_t$ will diminish as $t \rightarrow T$. Furthermore, utilizing the fact that $\mathbf{x}_{t\text{-}1} = \sqrt{\alpha_{t-1}}\mathbf{x}_{t\text{-}2} + \sqrt{1-\alpha_{t-1}}\boldsymbol{\epsilon}_{t-1}$, the relationship between $\mathbf{x}_t$ and $\mathbf{x}_{t\text{-}2}$ can written as

$$\mathbf{x}_t = \sqrt{\alpha_t\alpha_{t-1}}\mathbf{x}_{t\text{-}2} + \sqrt{1-\alpha_t\alpha_{t-1}}\bar{\boldsymbol{\epsilon}}_{t\cup t-\mathbf{1}} \tag{2}$$

where $\bar{\boldsymbol{\epsilon}}_{t\cup t-\mathbf{1}}$ denotes the gaussian noise that merges $\boldsymbol{\epsilon}_t$ and $\boldsymbol{\epsilon}_{t-1}$. Likewise, we can get the relation between $\mathbf{x}_t$ and $\mathbf{x}_0$ as

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\bar{\boldsymbol{\epsilon}}_{t\cup t-1\ldots 1\cup 0}\ , \tag{3}$$

where $\bar{\alpha}_t = \prod_{i=1}^{t}\alpha_i$. Rewritting (3), we get

$$\mathbf{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\mathbf{x}_t - \bar{\boldsymbol{\epsilon}}_t\sqrt{1-\overline{\alpha_t}}\right), \tag{4}$$

where we denoted $\bar{\boldsymbol{\epsilon}}_{t\cup t-1\ldots 1\cup 0}$ as $\bar{\boldsymbol{\epsilon}}_t$.

According to (4), a guess of $\mathbf{x}_0$ can be made at every time step t. In [14], the authors proposed the use of a neural network which predicts $\bar{\boldsymbol{\epsilon}}_t$ at every time step t. Denoting the guess of $\mathbf{x}_0$ at time t as $\mathbf{x}_{0|t}$ and using the neural network $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ which takes as the input $\mathbf{x}_t$ and t and outputs an estimate for $\bar{\boldsymbol{\epsilon}}_t$, we get from (4) the following equation:

$$\mathbf{x}_{0|t} = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\mathbf{x}_t - \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\sqrt{1-\overline{\alpha_t}}\right). \tag{5}$$

In the DDNM method, the authors try to restore $\mathbf{x}$ from the following degradation equation:

$$\mathbf{y} = \mathbf{A}\mathbf{x}, \tag{6}$$

where $\mathbf{y}$ is the degraded observation, $\mathbf{A}$ is a linear operator that performs the degradation, and $\mathbf{x}$ is the original image that we want to recover. To utilize the denoising diffusion probabilistic model for the image restoration task, the DDNM model decomposes $\mathbf{x}_{0|t}$ into its range-space part and null-space part as

$$\mathbf{x}_{0|\mathrm{t}} = \mathbf{A}^{\dagger}\mathbf{A}\mathbf{x}_{0|\mathrm{t}} + \left(\mathbf{I} - \mathbf{A}^{\dagger}\mathbf{A}\right)\mathbf{x}_{0|\mathrm{t}}, \tag{7}$$

and then replaces the range-space part $\mathbf{A}^{\dagger}\mathbf{A}\mathbf{x}_{0|t}$ by $\mathbf{A}^{\dagger}\mathrm{y}$ to obtain a rectified version of $\mathbf{x}_{0|t}$:

$$\hat{\mathbf{x}}_{0|t} = \mathbf{A}^{\dagger}\mathbf{y} + \left(\mathbf{I} - \mathbf{A}^{\dagger}\mathbf{A}\right)\mathbf{x}_{0|t}. \tag{8}$$

The image $\hat{\mathbf{x}}_{0|t}$ is the restored version of $\mathbf{y}$, which gets better and better as $t \rightarrow 0$.

In the proposed method, we utilize the DDNM model to recover the original colors from the compressed $\mathbf{y}$ information. The degradation matrix $\mathbf{A}$ becomes the operator that compresses the R, G, and B color components into a single value y, and the DDNM restores the R, G, and B components back from y. It should be taken into account that we use a pixel-wise operator **A,** as will be explained in section 3, and therefore, $\mathbf{y}$ becomes a scalar value y.

3. Proposed Method

Figure 1 shows the diagram of the proposed encoder. The original image is first decomposed into two parts: the white channel and the color seeds. The white channel is then compressed by a 1-channel compression technique and the compressed white channel is then sent to the image decoder. The extracted color seeds are also rearranged in a way that the entropy coding can compress them more efficiently, then the rearranged color seeds are compressed with a suitable entropy encoding, and the compressed color seeds are sent to the image decoder. The compressed white channel and color seeds are then individually decoded in the decoder, and rearranged as a RGBW patterned image which is then used by the DDNM method to restore the original color image.

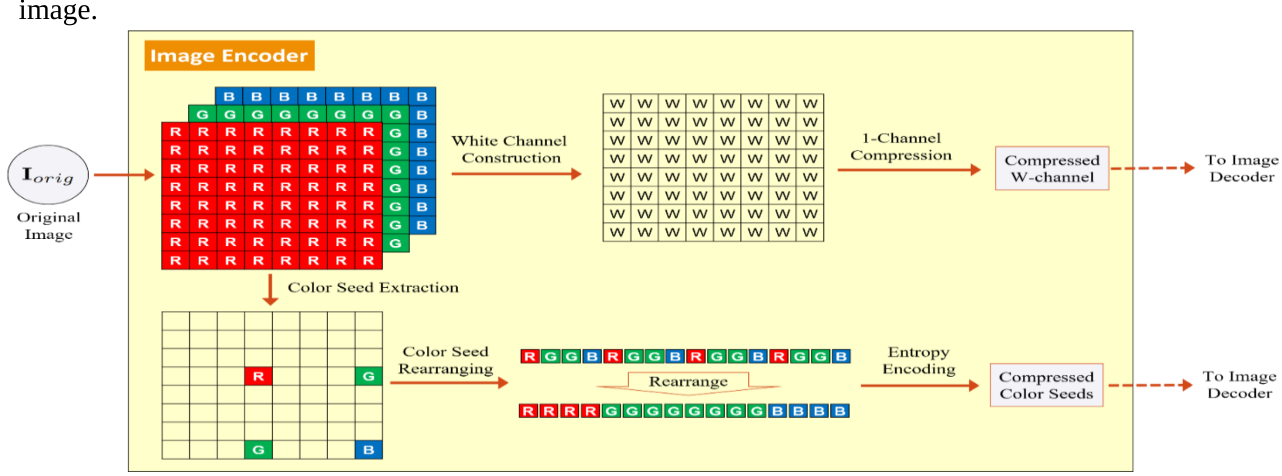

Figure 1. Diagram of the proposed Encoder

Figure 2 shows a variant of the proposed encoding method. Instead of sending the white channel and the color seeds independently to the decoder, we construct them as a single 1-channel RGBW patterned image and compress it via a 1-channel compression method and send just the compressed 1-channel image to the decoder.

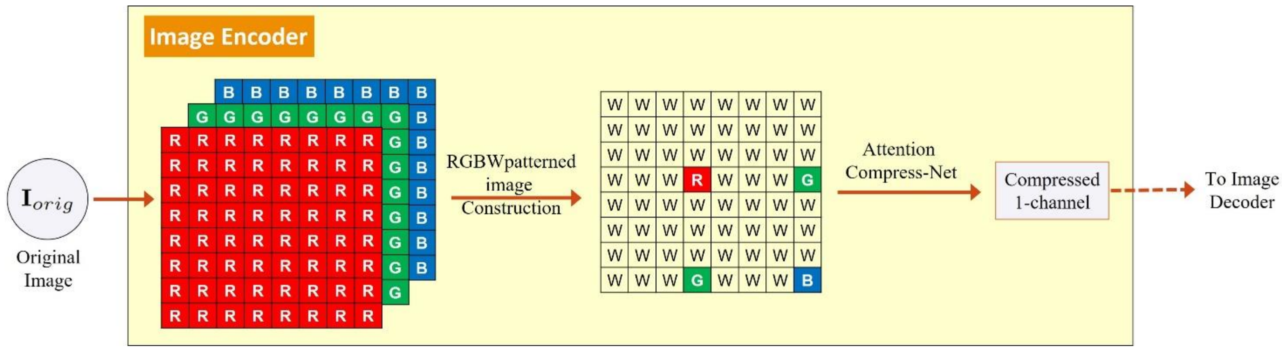

Figure 2. Diagram of a variant of the proposed Encoder

In our case, we used the Attention-CompressNet[13] for compressing the RGBW patterned image. The compressed image is then to the decoder, and the decoder reconstructs the RGBW patterned image which is then used by the DDNM to reconstruct the color image. The white channel construction is performed according to the following equation, which is the well-known RGB to luminance conversion equation obtained from the relative luminance of the three primary colors in the RGB color space[15]:

$$W = 0.299R + 0.587G + 0.114B, \tag{9}$$

where R, G, and B are the red, green, and blue channel components, and W is the converted luminance value of the pixel. After all the W values are computed for all the pixels, we get a full white channel. This single channel can be compressed using an existing 1-channel compression method. The compressed W-channel is sent to the decoder.

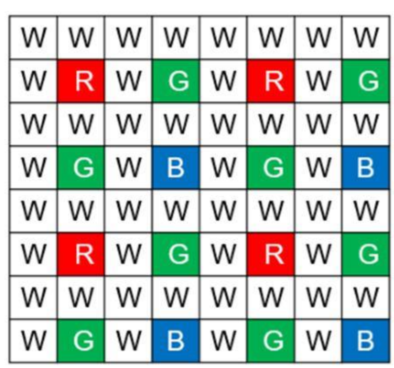

Figure 3. RGBW Pattern

Subsequently, we choose specific pixels in which we retain one of the primary color components—either the R, G, or B color component. The pixels from which to extract the R, G, B color components are predetermined. In other words, R, G, B values are extracted following a specific pattern. Figure 2 shows the pattern for which the R, G, B values are extracted. Because of the predefined pattern, there is no need to record the position of the R, G, B pixels, and we can just record the R, G, B values in a pre-defined order.

The decoder receives the compressed White channel and the compressed color seeds from the encoder. The compressed White channel is decompressed by the 1-Channel decompressor, and the color seeds are decoded by the entropy decoder. As a variant, we use the decoder in Fig. 4, which decompresses the RGBW patterned image, and then use the DDNM model to reconstruct the color image.

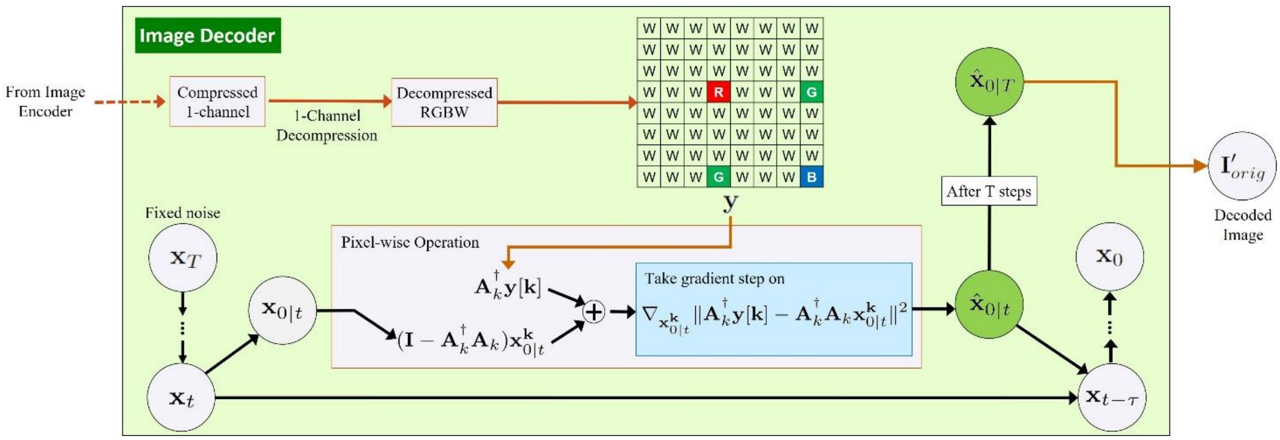

Figure 4. Diagram of the proposed Decoder

The reason for using R, G, B pixels is that the DDNM ensures accurate restoration of R, G, B values. For example, for the R pixel we have $\mathbf{A}=[1 \quad 0 \quad 0]$ and $\mathbf{A}^{\dagger}=\mathbf{A}^T(\mathbf{A}^T\mathbf{A})^{-1}=[1 \quad 0 \quad 0]^{\mathrm{T}}$ and therefore,

$$\hat{\mathbf{x}}_{0|t}=\mathbf{A}^{\dagger}\mathrm{y}+\left(\mathbf{I}-\mathbf{A}^{\dagger}\mathbf{A}\right)\mathbf{x}_{0|t}=\begin{bmatrix}\mathrm{y}\\0\\0\end{bmatrix}+\begin{bmatrix}0&0&0\\0&1&0\\0&0&1\end{bmatrix}\mathbf{x}_{0|t}=\begin{bmatrix}\mathrm{y}\\\mathbf{x}_{0|t}[2]\\\mathbf{x}_{0|t}[3]\end{bmatrix} \tag{10}$$

The values of the remaining components are determined by the diffusion model. However, the R, G, B values impose additional constraints so that the solution is further restricted to a unique solution as illustrated in Fig. 5. This constrains the DDNM to restore an image very close to the original color image.

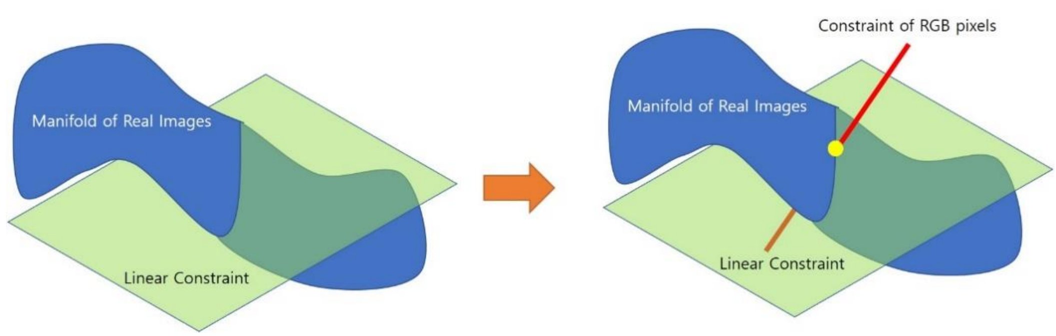

Figure 5. Constraints on the Solution Space

4. Experimental Results

For all the experiments, we used the CelebA-HQ dataset for testing. The experiments have been done on a PC with RAM of 32GB, running on a 4.2GHz CPU and using GPU RTX4090. Figure 6 compares the original color images and the decoded color images with the proposed method. The upper row in Fig. 6 shows the original colors and the bottom shows the reconstructed color image with the proposed method. The middle row shows the corresponding 1-channel RGBW patterned image that is compressed and sent to the decoder. It can be seen that the colors can be reconstructed using the proposed color filter array(CFA) pattern for encoding the original color images.

Figure 6. Decoding Results of the Proposed Method. Upper Row: Original Color Image, Middle Row: RGBW patterned Image, Bottom Row: Decoded Color Image

Figure 7 compares the results when using the All-White CFA pattern and the proposed CFA pattern to compress the original color image, and using the DDNM to reconstruct the color images. It can be seen in the upper row that the proposed method can reproduce all the original colors whereas the All-White CFA pattern fails to reconstruct the true colors. This validates the use of the RGBW patterned CFA pattern with the DDNM method.

Figure 7. Comparison of decoded results between the RGBW pattern and the All-white pattern. Top Row: RGBW pattern, Bottom Row: All-White Pattern

Figure 8 compares the decoded images between the proposed method, the JPEG algorithm, and the JPEG2000 algorithm when the BPP(Bit Per Pixel) is low. It can be seen that the proposed method achieves the best results.

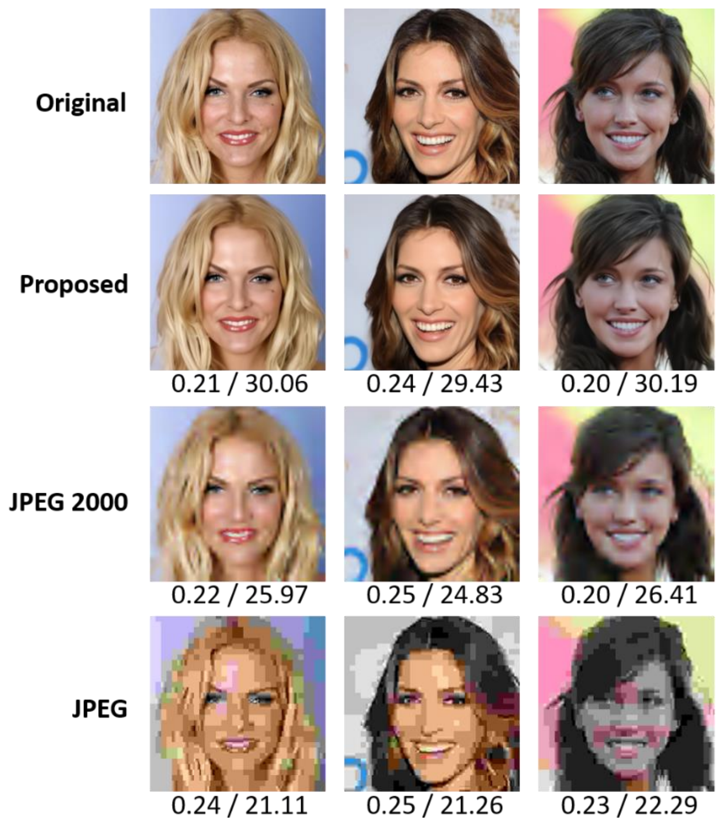

Figure 8. Comparison between the Proposed method, the JPEG and the JPEG2000 decoding results when the BBP is very low. The first values below the images are the BPP values, and the second values are the PSNR values.

5. Conclusion

In this paper, we have shown how we can compress an image based on the use of color seeds and white pixels, which are then reconstructed by the DDNM model. It has been experimentally that the proposed method achieves better compression results than the existing JPEG and JPEG2000 compression method. Furthermore, we explained why the use of the RGBW patterned image can achieve a unique solution while an all-white patterned image cannot. In future, researches can concentrate on using more diverse pattern images for the aim of compression.

Acknowledgement

This work was supported by Dongseo University, "Dongseo Cluster Project" Research Fund of 2023 (DSU- 20230004)

References

[1] L. Cheng and S. V. N. Vishwanathan, “Learning to compress images and videos,” in Proc. Int. Conf. Mach. Learn., Vol. 227, pp. 161–168, 2007. DOI: https://doi.org/10.1145/1273496.1273517
[2] X. He, M. Ji, and H. Bao, “A unified active and semi-supervised learning framework for image compression,” in Proc. IEEE Comput. Vis. Pattern Recognit., pp. 65–72, Jun. 2009. DOI: https://doi.org/10.1109/CVPR.2009.5206835
[3] L. Tan, Y. Zeng, and W. Zhang, “Research on Image Compression Coding Technology,” in Proc. 3rd International Conference on Data Mining, Communications and Information Technology (DMCIT 2019), Vol. 1284, pp.1-7, 2019. DOI: https://doi.org/10.1088/1742-6596/1284/1/012069
[4] G.K. Wallace, “The JPEG still picture compression standard,” IEEE Transactions on Consumer Electronics, Vol. 38, Issue 1, pp. 18-34, 1992, DOI: https://doi.org/10.1109/30.125072
[5] T. Miyata, Y. Komiyama, Y. Inazumi, and Y. Sakai, “Novel inverse colorization for image compression,” in Proc. Picture Coding Symp., 2009, pp. 1–4. DOI: https://doi.org/10.1109/PCS.2009.5167413
[6] S. Ono, T. Miyata, and Y. Sakai, “Colorization-based coding by focusing on characteristics of colorization bases,” in Proc. Picture Coding Symp. Dec. 2010, pp. 230–233. DOI: https://doi.org/10.1109/PCS.2010.5702473
[7] S. Lee, S.-W. Park, P. Oh, and M.-G. Kang, “Colorization-Based Compression Using Optimization,” IEEE Trans. Image Processing, Vol. 22, No. 7, pp. 2627-2635, 2013. DOI: https://doi.org/10.1109/TIP.2013.2253486
[8] A. Levin, D. Lischinski, and Y. Weiss, “Colorization using optimization,”ACM Trans. Graph., vol. 23, no. 3, pp. 689–694, Aug. 2004. DOI: https://doi.org/10.1145/1015706.1015780
[9] Y. Wang, J. Yu, J. Zhang, “Zero-Shot Image Restoration Using Denoising Diffusion Null-Space Model,” in Proc. ICML (2023), 2023. DOI: https://doi.org/10.48550/arXiv.2212.00490
[10] J. Schwab, S. Antholzer, and M. Haltmeier, “Deep null space learning for inverse problems: convergence analysis and rates,” Inverse Problems, Vol. 35, No. 2, pp.1-13, 2019. DOI: https://doi.org/10.1088/1361-6420/aaf14a
[11] B. Kawar, M. Elad, S. Ermon, and J. Song, “Denoising diffusion restoration models,” in Proc. ICML(2022), 2022. DOI: https://doi.org/10.48550/arXiv.2201.11793
[12] J. Song, C. Meng, and S. Ermon, “Denoising diffusion implicit models,” In Proc. International Conference on Learning Representations(ICLR), 2021. DOI:https://doi.org/10.48550/arXiv.2010.02502
[13] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, “Learned image compression with discretized gaussian mixture likelihoods and attention modules,” In Proc. of the IEEE/CVF conference on computer vision and pattern recognition (pp. 7939-7948). 2020. DOI:https://doi.org/10.1109/CVPR42600.2020.00796
[14] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” arXiv Preprint arXiv:2006.11239, 2020. DOI:https://doi.org/10.48550/arXiv.2006.11239
[15] International Commission on Illumination. *https://cie.co.at*.