DeLDRify: single-image LDR to HDR augmentation

Clement Mauget (20236399) - Oskar Joergensen (20236269)

Abstract

In this paper, we present DeLDRify, a novel approach for single-image Low Dynamic Range (LDR) to High Dynamic Range (HDR) augmentation. We discuss the challenges associated with this conversion process and introduce our model, which leverages Generative Adversarial Networks (GANs) and a modified established network for super-resolution, to achieve impressive results. Our method demonstrates promising performance, even when trained on a smaller dataset, showcasing its potential for real-world applications. Additionally, we provide both quantitative and qualitative evaluations of our model, highlighting its ability to recover information from shadows and highlights. Overall, DeLDRify offers a compelling solution for enhancing LDR images, paving the way for improved visual experiences in the era of HDR content. Our code is available here: https://github.com/oskarjor/DeLDRify

1. Introduction

The widespread availability of Ultra High Definition (UHD) displays, such as modern televisions, has prompted the need for a continuous supply of UHD images for viewing. To address this challenge, various solutions for image upsampling and super-resolution have emerged. Presently, most high-end UHD TVs can perform real-time upsampling, converting low-resolution images into high-resolution ones. We are now seeing the same phenomenon happen with High Dynamic Range (HDR) displays. While HDR capabilities are becoming increasingly common, HDR content remains somewhat scarce. Like with converting low resolution images to high resolution images, LDR to HDR conversion is an interesting and non-trivial problem. It requires preserving much of the information from the original input, while also hallucinating in some new details. In response to this, we introduce a method similar to the upsampling techniques used for Low Dynamic Range (LDR) to High Dynamic Range (HDR) image generation, employing a Generative Adversarial Network (GAN): DeLDRify.

1.1. LDR and HDR Images

But what are exactly LDR and HDR images? HDR images are often represented with at least 16 bits per pixels, whereas LDR can be stuck at 8 bits. This difference in stored information, coupled with a generally better sensor technologies, allows HDR images to contain more dynamic range. Dynamic range is the range between the lowest light value represented by something other than 0, and the highest light value represented by something other than 255. The difference between LDR and HDR images can, in some sense, be seen as a clipping issue.

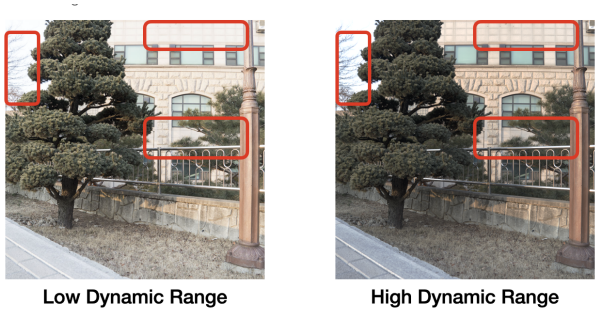

Figure 1. Comparison between and LDR and HDR image, relevant differences are denoted with a red rectangle

2. Related work

The task of reconstructing an HDR image from LDR images can be split into two broad categories. First, we have the problem of constructing a single HDR image from several LDR images, usually at different exposures. Second, we have the task of reconstructing an HDR image from a single LDR image. While the first problem has been studied for decades [3], the second problem has historically been more difficult. We will now take a look at some relevant work related to the task of converting LDR images to HDR.

2.1. Current methods

The first attempt at solving the problem of converting a single LDR image to HDR, used a deep convolutional network [4, 10], and since then several other approaches have been tried. Some use the UNet architecture. There has

also been attempts to introduce a network that is spatially variant (as opposed to conventional CNNs), and a third network that weights the input image, to pay more attention to over-/under-exposed areas as these provide the most trouble when reconstructing [2]. Others apply several convolutional neural networks, in an attempt to capture both local and global information about the LDR image, and use it for the reconstruction [11]. A newer approach utilizes a modified transformer architecture to image features, and a separate network for learning the tone mapping [16]. There have also been approaches using Generative Adversarial Networks, both for the multi-image reconstruction problem [9], and the single-image problem [14].

To our knowledge, HDR-cGAN [14] is the newest attempt at single-image LDR to HDR conversion using a GAN architecture. As such, it is interesting to take a closer look at what they did. Due to the complexity of the task, Raipurkar et al. opted to use a three stage pipeline. The first stage is the Linearization module. The first module converts the non-linear, fix ranged LDR image data, into linear image irradiance. The authors claim that this module is largely responsible for preserving and enhancing information in under-exposed and normally-exposed regions of the image. The next module is the Over-Exposed Region Correction (OERC). This hallucinates details from the over-exposed regions of the image. The authors use a mask of the over-exposed regions to guide the network to where it should hallucinate. Finally, we have the Refinement module. This module is responsible for adjusting color, gamma, white balance and doing other corrections that modern cameras automatically apply.

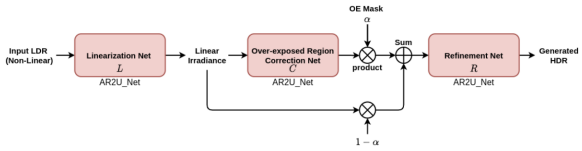

Figure 2. Architecture of the HDR-cGAN

Each module's architecture is based on a convolutional recurrent neural network (CRNN), with attention [1, 12]. The model is trained end-to-end using adversarial loss (hence the GAN name), as well as reconstruction loss and perceptual loss.

3. Proposed Method

Extensive research has already been made in super-resolution methods using GANs [5, 15]. Those methods provide state-of-the-art models and impressive performance. An intuitive next step is to investigate the most effective approaches, and then pivot from the Low-Resolution (LR) to High-Resolution (HR) problem to a LDR-to-HDR challenge.

3.1. LR-to-HR methods

One of the most efficient method for super-resolution leveraging GANs is ESRGAN, or Enhanced Super-Resolution Generative Adversarial Network [17]. ESRGAN builds upon the traditional SRGAN (Super-Resolution Generative Adversarial Network) [8] by introducing architectural improvements and more sophisticated training techniques, more specifically, it uses a simplified 16 blocks deep ResNet (fig 3). ESRGAN's training process leverages a combination of adversarial training with perceptual loss to enhance image quality, allowing it to generate high-resolution, realistic images with enough high-frequency information. The loss for the generator (LG) (6) is a combination of the standard adversial loss (5), the proposed perceptual loss L_{percep}, and a simple L_1 loss:

$$L_{Ra}^{G} = - \mathbb{E}_{x_r}[\log(1 - D_{Ra}(x_r, x_f))] \quad (1)$$

$$- \mathbb{E}_{x_f}[\log(D_{Ra}(x_f, x_r))] \quad (2)$$

$$LG = L_{percep} + \lambda L_{Ra}^{G} + \eta L_1 \quad (3)$$

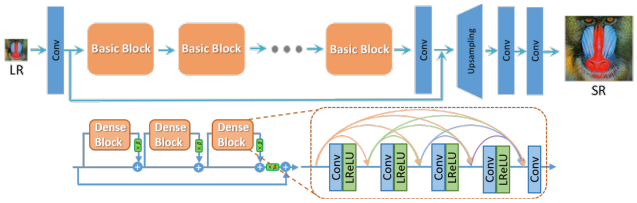

Figure 3. ESRGAN Model Architecture

3.2. LR-to-HR into LDR-to-HDR

LR-to-HR problems have a lot in common with LDR-to-HDR problems. Both problems deal with image-to-image problems, and infer unseen data from low information spaces. Since GANs network have been historically good at deriving the goal from the data, we believe that leveraging a modified version of the SRResNet and modifying the perceptual loss will lead to a solid foundation to iterate on.
The goal of this project is then two-fold:

- Setting a New Benchmark for LDR-to-HDR Challenges: Our first goal is to establish a novel state-of-the-art standard for addressing Low Dynamic Range (LDR) to High Dynamic Range (HDR) conversion problems. By doing so, we aspire to push the boundaries of image enhancement and the creation of realistic HDR content from initially limited data.

- Exploring the Versatility of Generative Adversarial Networks (GANs) in Image Enhancement: This study delves into the broad applicability of GANs beyond their conventional use cases by examining their effectiveness in addressing LDR-to-HDR image enhancement challenges. Our goal is to understand how GANs, originally designed for other image processing generative tasks, perform when confronted with the task of elevating the dynamic range of images, shedding light on their versatility in diverse image processing scenarios.

3.3. Model Architecture

More precisely, we leverage the ESRGAN model architecture (fig 3) with the upsampling layer removed. Following ESRGAN idea, the main building block of DeLDRify is the residual-in-residual dense block (RRDB) (fig 4), which combines multi-level residual connections and dense block.

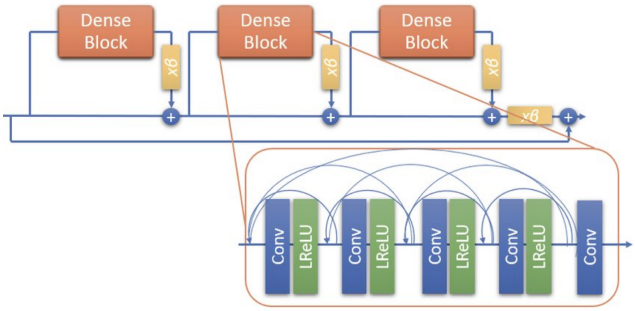

Figure 4. Residual-in-Residual Dense Block

In each dense block, we find multiple convolutional layers, where each takes as an input the combination of all previous layers. Multiple dense block are then added in a serial manner, with residual connection in-between. Finally, a residual connection by-passing all the dense block is added. One key elements of RRDB is the use of residual scaling instead of batch normalization. For each residual connection, the scaling factor is a free learnable parameter. By stepping away from batch normalization, it can, to some degree, reduce the inherent "hallucinations" in GANs models. RRDB blocks allow for an easy and stable training of very deep-networks.

As for the discriminator, the one provided with ESRGAN proved to be too powerful, resulting in the generator not being able to learn. To tackle this issue, we used a modified discriminator, a simple CNN trained on classification tasks. Throughout our experiments, we implemented from 6 to 10 convolutional layers (the depth) with different filter size (the width). It is important to note that, whereas the generator is given the LDR image as an input, the discriminator only receives the output of the generator.

3.4. HDR Image Representation

A Low Dynamic Range image is normally represented by 3 channels, one for each color **R**ed, **G**reen and **B**lue (RGB). Each pixel in each channel takes a value from 0 to 255, representing how present each color is at that exact pixel. A pixel that is completely green might therefore be represented by the tuple (0, 255, 0), while a pixel that is equal parts red and blue (purple) might be represented by the tuple (146, 0, 146). Since each color can take 256 values, they can be represented by 8 bits. A pixel (which consist of 3 values) might therefore be represented by 24 bits. For an HDR image, each color can take 2^{32} different values. The most naïve way to represent this is to use 32 bits for every channel, for a total of 96 bits per pixel. This is called the extended RGB format. This can lead to quite large images. Therefore, other formats have been created. One of those is the RGBE format. Here, each color channel is represented by 8 bits. Furthermore, we have a 4th channel called the exponent channel. For each pixel, a shared exponent is stored in this channel using 8 bits. This allows each color to still take 2^{32} values, but uses only 32 bits per pixel [20]. While the two formats can store the same information, the way in which they store it might mean that one is better suited when training a neural network. We will therefore consider both options.

4. Experiments

4.1. Training Details

Our model was implemented in PyTorch, with code adapted from [17]. We have a dataset of matching LDR and HDR pairs. Each image is downscaled to 128×128. This is to make it feasible to train the model. The model was trained on an Apple M1 Pro GPU with 16 GB of RAM for 3 hours. With a better GPU, it would be possible to adopt the architecture to train on larger images. The model was trained for 400 epochs using the Adam optimizer [7] with a $lr = 0.0002$ and $\beta_1 = 0.9, \beta_2 = 0.999$. The generator follows the ESRGAN implementation, but with the upsampling layers removed. The discriminator also follows ESRGAN, and is relativistic [17]. For training the generator, we use the reconstruction loss and the adversarial loss, which is scaled by a hyperparameter λ (usually set to 0.02). The total loss is then given by:

$$loss_G = loss_{pixel} + \lambda \cdot loss^G_{adv} \tag{4}$$

$$loss_{pixel} = \sum_p ||HDR_F - HDR_R|\forall p \tag{5}$$

$$loss^G_{adv} = -\mathbb{E}_{HDR_R}[\log(1 - D_{Ra}(HDR_R, HDR_F))] \tag{6}$$

$$\tag{7}$$

where HDR_{fake} is the generated HDR image, HDR_{real} is the ground truth HDR image, p is a pixel in the image and D_{Ra} is the relativistic discriminator describe in [17]. For the discriminator, we only use the adversarial loss in its symmetrical form:

$$loss_D = -\mathbb{E}_{HDR_R}[\log(D_{Ra}(HDR_R, HDR_F))] \quad (8)$$
$$- \mathbb{E}_{HDR_F}[\log(1 - D_{Ra}(HDR_F, HDR_R))] \quad (9)$$
$$(10)$$

All the code is available, along with suggested hyperparameters [1].

We use the LDR-HDR-pair_Dataset available here [6]. This dataset consists of 176 HDR images, with three corresponding LDR images. The LDR images have different exposure values. For this study, we are using the LDR images with a medium exposure setting. We use this dataset since it is publicly available, and easily accessible. This ensures that our results are reproducible, and are easy to compare to.

4.2. Evaluation Metrics

For evaluating our performance, we will use metrics commonly used to quantify the reconstruction quality. Primarily, we will use the Peak Signal-to-Noise ratio, or *PSNR* for short. Given an input image I and our approximation K, it is defined as follows (in dB):

$$PSNR = 10 \cdot log_{10}(\frac{MAX_I^2}{MSE}) \quad (11)$$
$$= 20 \cdot log_{10}(\frac{MAX_I}{\sqrt{MSE}}) \quad (12)$$
$$= 20 \cdot log_{10}(MAX_I) - 10 \cdot log_{10}(MSE) \quad (13)$$

where MAX_I is the maximum possible pixel value for an image (i.e., 255 for 8-bit images) and the MSE is defined as follows:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2 \quad (14)$$

where m, n and are the image dimensions [19].

We also employ Structural Similarity, or SSIM for short [18]. This is a more robust measurement, as it does not measure the absolute difference between the images at a pixel level, but rather if the same *structural information* is present. It is calculated on various windows of an image. Given two windows of size $N \times N$ x, y from our two images, we calculate SSIM as follows:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (15)$$

where μ_x, μ_y is the pixel sample mean of windows x, y and $\sigma_x, \sigma_y, \sigma_{xy}$ is the variance of windows x, y and their covariance. c_1, c_2 are hyperparameters, mainly included for stability when elements in the denominator is close to zero.

4.3. Evaluation Environment

To test our model, we split the LDR-HDR pair dataset into a training and validation set with a 0.2 ratio. We use the validation set for testing. However, and with some fear of small data leakage between the training and validation set, we chose to only report the worst results encountered. As such, we essentially create a lower-bound of the model performance. We believe that this offer a strong enough base to compare our different models.

Furthermore, several of the state-of-the-art models originated from the NTIRE 2021 challenge on High Dynamic Range imaging [13], for which a dataset was created. It consists of roughly 1500 HDR-LDR image-pairs, where the LDR image is synthesized from the HDR image. As other SOTA models have been trained and evaluated on this dataset, we find it natural to evaluate our model on it. It's crucial to emphasize that the images in this dataset differ significantly from the ones used for training, featuring more indoor settings and high-contrast scenes. While comparing the extended RGB and the RGBE format, we quickly realized that it was very hard for the model to predict exponent channel for our images, as all values ranged from 0 to 4 in the HDR images. This lead to very low changes in the exponent having drastic outcomes. We therefore decided to abandon this format.

4.4. Quantitative Results

We compare three model variant, each with a different discriminator width and depth, on the validation dataset and the NTIRE dataset. For the validation dataset, we only report the average metrics of the poorest performing examples.

We further compare those models with the top 5 models from the NTIRE 2021 challenge, with the values reported from this paper [2].

We report our results in the following table (fig 5). Reinforcing our impression of an unbalanced discriminator, we observe that the model using the smallest discriminator displays the best performance.

Furthermore, the results are not far away from the top 5 of the NTIRE challenge. Indeed, those models show an improvement of only 5 to 10% compared to ours. For a fair comparison, we have to consider two factors. First, our

[1] https://github.com/oskarjor/DeLDRify

	NTIRE		Validation	
	PSNR	SSIM	PSNR	SSIM
NOAHTCV	32.867	-	-	-
XPixel	32.285	-	-	-
BOE-IOT-AIBD	33.490	-	-	-
CET CVLab	32.060	-	-	-
CVRG	31.021	-	-	-
DeLDRify (Thin - Shallow Discriminator)	**29.292**	**0.827**	**25.363**	**0.880**
DeLDRify (Wide - Shallow Discriminator)	28.410	0.715	24.673	0.871
DeLDRify (Wide - Deep Discriminator)	27.112	0.591	25.154	0.858

Figure 5. Quantitative Results

model is trained on a significantly smaller and widely different dataset than from the NTIRE one. Second, the top 5 models are trained on a high-end GPU for weeks, whereas our model is trained for 3 hours on a laptop. Considering those circumstances, we believe that DeLDRify shows promising results.

4.5. Qualitative Results

In addition to our quantitative analysis, we provide a qualitative assessment of our results (fig 6, 7, 8). Please note that the HDR and Fake HDR pictures have been tone mapped for view on an SDR display.

For the outdoor scene (fig 6), we observe that we indeed have information recovered from the shadows, with the grill on the bottom left, and the highlights, with the top right. We do see that the model struggles with the light streaks on the gutter. Indeed, the model interprets it as the beginning of an overexposed region rather than a single light region.

For the first indoor scene (fig 7), we observe that we do recover information in the highlights. The reflection on the ground also carries more information.

For the second indoor scene from the NTIRE dataset (fig 8), we observe that we are able to recover over-exposed area from the bottle and the tale. However, the model does not perform well to recover huge areas of shadows. We believe this is due to the non-representation of those type of examples in the LDR-HDR pair dataset.

Overall, DeLDRify provides good qualitative results and shows good generalization.

5. Conclusion

Throughout this paper, we presented our model for single image LDR to HDR conversion, DeLDRify. We showed that, despite unfair comparison, our model achieves close results with other state-of-the-art models. We strongly believe that, with more resources and time, we could achieve SOTA performance.

Figure 6. Qualitative results for an outdoor scene from the LDR-HDR pair dataset.

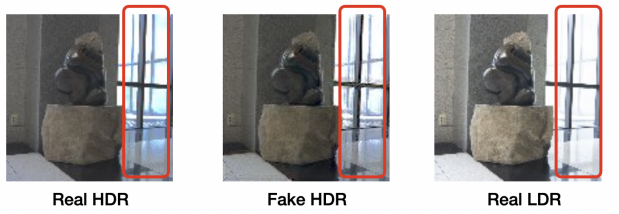

Figure 7. Qualitative results for an indoor scene from the LDR-HDR pair dataset.

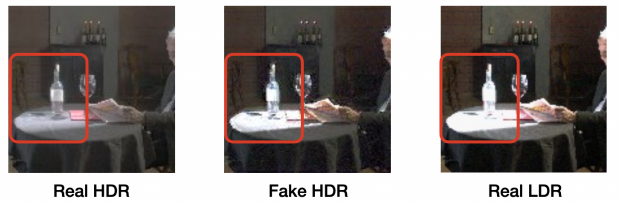

Figure 8. Qualitative results for an indoor scene from the NTIRE dataset.

This paper also shows to a certain extent the versatility of GANs network. Indeed, the performance of a modified ESRGAN on new tasks is impressive.

For future work, two paths open. First, we could delve deeper into the generalizability of the RRDB blocks and the ESRGAN networks. An extensive weight and generalization study could be performed. Second, we could focus on improving the performance of DeLDRify. To do so, we could train our model on larger, more diverse datasets and with higher quality images. We could further modify the ESRGAN for this task by, for example, implementing an attention mechanism shifting the focus on more over- and under-exposed regions of the images.

We are eager to further this project and refine our method, should the opportunity arise.

References

[1] Md Zahangir Alom, Mahmudul Hasan, Chris Yakopcic, Tarek M. Taha, and Vijayan K. Asari. Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation, 2018. 2

[2] Xiangyu Chen, Yihao Liu, Zhengwen Zhang, Yu Qiao, and Chao Dong. Hdrunet: Single image hdr reconstruction with denoising and dequantization, 2021. 2, 4

[3] Paul E Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs, 1997. 1

[4] Gabriel Eilertsen, Joel Kronander, Gyorgy Denes, Rafał K. Mantiuk, and Jonas Unger. HDR image reconstruction from a single exposure using deep CNNs. *ACM Transactions on Graphics*, 36(6):1–15, nov 2017. 1

[5] Kui Fu, Jiansheng Peng, Hanxiao Zhang, Xiaoliang Wang, and Frank Jiang. Image super-resolution based on generative adversarial networks: a brief review. 2020. 2

[6] Hanbyol Jang, Kihun Bang, Jinseong Jang, and Dosik Hwang. Dynamic range expansion using cumulative histogram learning for high dynamic range image generation. *IEEE Access*, 8:38554–38567, 2020. 4

[7] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 3

[8] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 2

[9] Siyeong Lee, Gwon Hwan An, and Suk-Ju Kang. Deep recursive hdri: Inverse tone mapping using generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 596–611, 2018. 2

[10] S. Mann and R.W. Picard. Being 'undigital' with digital cameras: Extending dynamic range by combining differently exposed pictures. Technical Report 323, M.I.T. Media Lab Perceptual Computing Section, Boston, Massachusetts, 1994. Also appears, IS&T's 48th annual conference, Cambridge, Massachusetts, May 1995. 1

[11] Demetris Marnerides, Thomas Bashford-Rogers, Jonathan Hatchett, and Kurt Debattista. Expandnet: A deep convolutional neural network for high dynamic range expansion from low dynamic range content, 2019. 2

[12] Ozan Oktay, Jo Schlemper, Loic Le Folgoc, Matthew Lee, Mattias Heinrich, Kazunari Misawa, Kensaku Mori, Steven McDonagh, Nils Y Hammerla, Bernhard Kainz, Ben Glocker, and Daniel Rueckert. Attention u-net: Learning where to look for the pancreas, 2018. 2

[13] Eduardo Pérez-Pellitero, Sibi Catley-Chandar, Aleš Leonardis, and Radu Timofte. Ntire 2021 challenge on high dynamic range imaging: Dataset, methods and results, 2021. 4

[14] Prarabdh Raipurkar, Rohil Pal, and Shanmuganathan Raman. Hdr-cgan: Single ldr to hdr image translation using conditional gan, 2021. 2

[15] Khushboo Singla, Rajoo Pandey, and Umesh Ghanekar. A review on single image super resolution techniques using generative adversarial network. *Optik*, page 169607, 2022. 2

[16] Jiaqi Tang, Xiaogang Xu, Sixing Hu, and Ying-Cong Chen. High dynamic range image reconstruction via deep explicit polynomial curve estimation. In *Frontiers in Artificial Intelligence and Applications*. IOS Press, sep 2023. 2

[17] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0, 2018. 2, 3, 4

[18] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 4

[19] Wikipedia contributors. Peak signal-to-noise ratio — Wikipedia, the free encyclopedia, 2023. [Online; accessed 24-October-2023]. 4

[20] Wikipedia contributors. Rgbe image format — Wikipedia, the free encyclopedia, 2023. [Online; accessed 11-December-2023]. 3