

# Luminance-Guided Chrominance Image Enhancement for HEVC Intra Coding

Hewei Liu\*, Renwei Yang\*, Shuyuan Zhu\*, Xing Wen<sup>†</sup> and Bing Zeng\*

\*School of Information and Communication Engineering

University of Electronic Science and Technology of China, Chengdu, China

<sup>†</sup>Kuaishou, Beijing, China

**Abstract**—In this paper, we propose a luminance-guided chrominance image enhancement convolutional neural network for HEVC intra coding. Specifically, we firstly develop a gated recursive asymmetric-convolution block to restore each degraded chrominance image, which generates an intermediate output. Then, guided by the luminance image, the quality of this intermediate output is further improved, which finally produces the high-quality chrominance image. When our proposed method is adopted in the compression of color images with HEVC intra coding, it achieves 28.96% and 16.74% BD-rate gains over HEVC for the U and V images, respectively, which accordingly demonstrate its superiority. The code is available online: <https://github.com/Nickyang4900/Luminance-Guided-Chrominance-Enhancement-for-HEVC-Intra-Coding>.

**Index Terms**—Luminance, chrominance, enhancement, HEVC, intra coding.

## I. INTRODUCTION

With the rapid development of multimedia technology, there has been a huge demand for high-definition content. Currently, high efficiency video coding (HEVC) [1] is widely used in the compression of high-definition content. However, still images compressed by HEVC intra coding often suffer from the compression artifacts, especially at the low bit-rate. Recently, the new generation of video coding standard, i.e., versatile video coding (VVC) [2] has been proposed. VVC offers more excellent rate-distortion performance compared with HEVC. However, the high computational complexity of VVC limits its application in practice. Improving the performance of HEVC is still very necessary and challenged. Enhancing the quality of compressed image offers a simple but effective way to obtain a high coding efficiency. Over the past few years, the convolutional neural network (CNN) based solution is becoming more and more popular to achieve this goal.

The CNN models are normally used to compose the end-to-end schemes to reduce the compression artifacts [3], [4]. In [5], the CNN-based in-loop filter was designed to replace the sample adaptive offset (SAO) [6] in HEVC. In [7], a multi-channel long-short-term dependency residual network was proposed to enhance the quality of video frames based on the temporal correlation of frames. The CNN-based post-processing methods were proposed in [8] and [9] for the HEVC-coded images. More specifically, a variable-filter-size residue-learning was

designed in [8] to implement the quality enhancement and a deep CNN-based auto decoder was proposed in [9] to achieve the same goal. In [10], several enhancement networks were designed to improve the quality of different coding tree units which finally achieves a overall quality gain for the whole image. Meanwhile, the coding information was also adopted in the CNN-based quality enhancement solution. For example, in [11], the motion residue was utilized to improve the video quality. The coding unit mask was defined in [12] as the prior information for the CNN-based post-processing. Also, the transform unit partition map was employed in [13] to improve the quality of video frames.

Besides the coding information, the channel correlation between the luminance image (Y) and the chrominance images (U and V) may also be utilized to improve the quality of compressed images. However, most of the existing CNN-based methods process each channel independently, which ignores the channel correlation and cannot achieve a high efficiency. In this work, we propose a luminance-guided chrominance image enhancement method for HEVC intra coding. More specifically, the degraded chrominance image is firstly fed into the recursive asymmetric-convolution block (GRAB) to obtain an intermediate output. Then, the feature of luminance image is obtained via the feature extraction block (FEB). Finally, the intermediate output and the luminance feature are combined together to reconstruct the final chrominance images. The pipeline of our proposed method is illustrated in Fig. 1.

The rest of this paper is organized as follows. The motivation of our proposed method is given in Section II. Our proposed method is introduced in Section III. The experimental results are presented in Section IV and the conclusions are drawn in Section V.

## II. MOTIVATION

In color image compression, the input RGB image is normally converted into the YUV format, including one luminance image (Y) and two chrominance images, i.e., U and V. Both the U and V images are then spatially down-sampled to form the YUV-420 format. The RGB-YUV conversion is implemented as

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.2126 & 0.7152 & 0.0722 \\ -0.1146 & -0.3854 & 0.5000 \\ 0.5000 & -0.4542 & -0.0458 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix}. \quad (1)$$

This work was supported by the National Natural Science Foundation of China under Grant U20A20184 and Grant 62031009. The first and second authors contributed equally to this work.

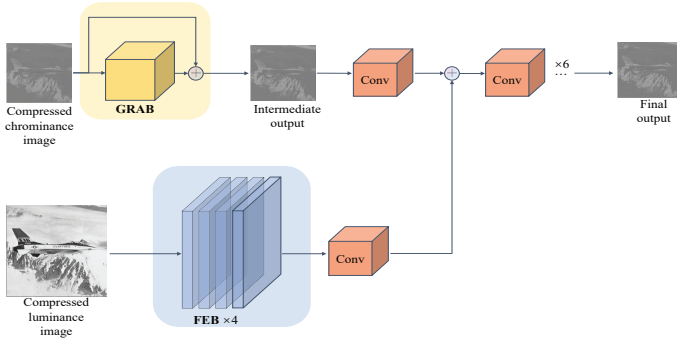


Fig. 1. Pipeline of our proposed method.

According to Eq. (1), although the luminance and chrominance images are obtained by combining the red (R), green (G) and blue (B) channels with different weighting factors, they are still highly correlated. We present some YUV-420 coding results of HEVC intra coding in Fig. 2. It is found from Fig. 2 that the compressed luminance image may potentially offer some detailed textures to enhance the compressed chrominance image. More specifically, the content of the luminance image is much clear but the texture of the chrominance image is blurred due to the spatial down-sampling and compression. The clear luminance textures may potentially provide prior information to enhance the quality of the chrominance image, which will finally offer an overall quality gain for the whole image.

### III. OUR PROPOSED METHOD

#### A. Overview

Our proposed method is designed for the YUV-420 format coding after the RGB-YUV conversion. Its pipeline is illustrated in Fig. 1, where the decoded chrominance image is firstly enhanced by GRAB, and then is further enhanced with the guidance of the decoded luminance image. We stack four feature extraction blocks (FEB) to extract the prior information from the luminance image. In our work, the intermediate chrominance output is passed through a  $3 \times 3$  convolution kernel with stride = 1. Therefore, we adopt the  $3 \times 3$  convolution kernel with stride = 2 after four FEBs to guarantee the luminance features and the chrominance features with the same size. To fuse the guided features offered by the luminance image, the features of the intermediate chrominance output and the ones of the luminance image are combined together via the element-wise adding. Moreover, we employ six convolution layers with the  $3 \times 3$  kernel size and stride = 1 in the last reconstruction procedure to yield the final chrominance output.

#### B. Gated Recursive Asymmetric-convolution Block

The proposed GRAB is used to produce the intermediate output for the chrominance image, and it is composed by six recursive units (RU). The recursive design adopted in our work aims at avoiding the dramatic increasing of model parameters when the network depth increases [14]. The architecture of GRAB is given in Fig. 3. In GRAB, each RU consists of four

asymmetric units (AU) [16], including a shallow one and three deep ones, a convolution layer, and a gate coefficient (GC) [19]. Due to the adoption of skip connection [17], RU only needs to learn the residual between the source high-quality image and the compressed low-quality input image, which leads to a simple training [18]. Except for the given values of GC, the six RUs share the same architecture and weighting parameters. In each RU, AU is designed with various branches, thus it is able to strengthen the network capability to obtain diverse features from input [15], [16]. Each AU consists of three  $1 \times 1$  convolution layers, two  $3 \times 3$  convolution layers, one average pooling and Leaky ReLU. The structure of AU is given in Fig. 4, and it is implemented as

$$z = c_1(x) + c_3(c_1(x)) + AvgPool_3(c_1(x)) + c_3(x) \quad (2)$$

where  $c_{i \in \{1,3\}}(\cdot)$  represents the convolution operation with the  $i \times i$  kernel size, and  $AvgPool_3$  denotes the average pooling with the  $3 \times 3$  slide window. In our work, the shallow and deep AUs share the same structures, but employ different weighting parameters, which can effectively reduce the parameters.

The gate coefficient was initially proposed in [19] to solve the image super resolution problem. We adopt it in this work to construct the depth-wise attention mechanism so that the network can be trained to assign different weights to the features of different depths, which potentially leads to a good reconstruction result. Moreover, we do not employ the channel attention methods [20], [21] in our work to avoid the high computational cost. The gate coefficient  $\beta$  adopted in GRAB is used as the weighting factor for each RU and it is achieved via multiplication as

$$z = \beta \cdot x. \quad (3)$$

Let us use  $s(\cdot)$  and  $d(\cdot)$  to represent the operation of the shallow and deep AUs, respectively. Then, in the  $i$ -th RU, we can obtain the output of the last AU

$$AU_4(y_{i-1}) = d(d(s(y_{i-1})) + s(y_{i-1})) + s(y_{i-1}) \quad (4)$$

where  $i \in \{1, \dots, 6\}$ , and  $y_i$  and  $\beta_i$  denote the output and GC of the  $i$ -th RU, respectively. Note that  $y_0$  is the output obtained by inputting the compressed chrominance image to a convolution layer whose kernel size is  $3 \times 3$ . Finally, the procedure of passing through the  $i$ -th RU is formulated as

$$y_i = \beta_i \cdot conv(AU(y_{i-1})) + y_{i-1}. \quad (5)$$

#### C. Feature Extraction Block

FEB is designed to extract features of the luminance image and it is designed with the skip connection structure, allowing the combination of shallow and deep convolution layer outputs. This structure not only strengthens the ability to extract features, but also makes the training more easily [14]. The architecture of the FEB is illustrated in Fig. 5. In this unit, the feature maps of all the depths directly connect to the last

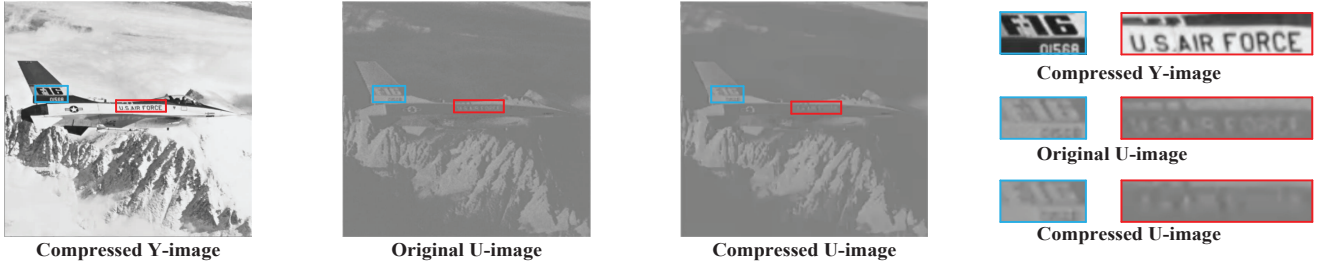


Fig. 2. An example of YUV-420 format coding by HEVC intra coding, where the quantization parameter is 22.

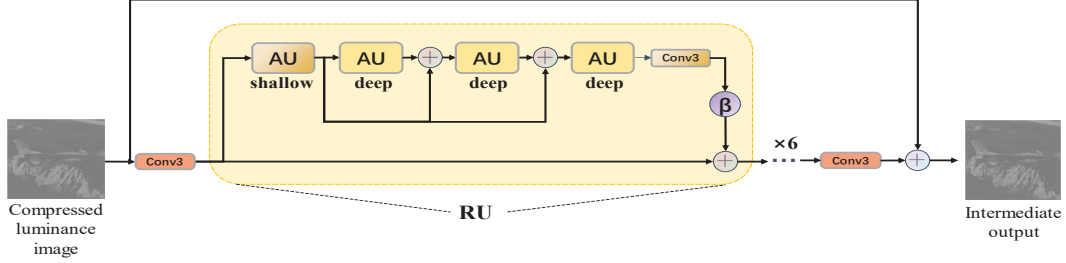


Fig. 3. Architectures of GRAB.

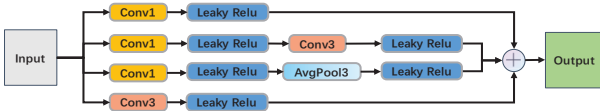


Fig. 4. Architectures of AU.

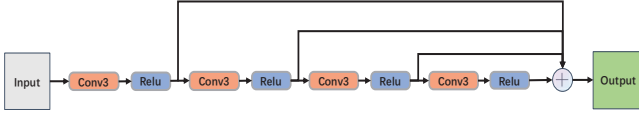


Fig. 5. Architectures of FEB.

layer, which combines the shallow and deep features from the input image. In our work, FEB is implemented as

$$f_{FEB} = \sum_{i=1}^4 \gamma_i \quad (6)$$

where  $\gamma_i$  is the output of the  $i$ -th convolution layer, i.e.,  $\gamma_i = ReLU(W_i * \gamma_{i-1})$ ,  $*$  denotes the convolution operation and  $W_i$  is the weighting parameter of the convolution layer. In this work, the ReLU activation function  $ReLU$  is employed to obtain  $\gamma_i$ . Moreover,  $\gamma_0$  should be the input image for the first FEB in in our proposed network.

#### IV. EXPERIMENTAL RESULTS

##### A. Experiment Setup

We verify the effectiveness of our proposed method by employing it to process the HEVC-compressed color images, where the HM 16.7 [23] is adopted in this work. Moreover, we compare the performance of our proposed method with the other existing methods. The first 800 color images of



Fig. 6. Classical image dataset.

Flickr2K dataset [24] are selected as the training dataset. Moreover, these images are converted into YUV-420 format and compressed by HEVC intra coding with the quantization parameters (QP) 22, 27, 32 and 37. The compressed chrominance images, including both the U and V images, are cropped into the  $32 \times 32$  non-overlapping sub-images, and the compressed luminance images are cropped into the  $64 \times 64$  non-overlapping sub-images. We train all the CNN models by using these generated sub-images.

##### B. Implementation Detail

In the network training, the batch size is set to 32 and the total epoch is set to 40. We start with a learning rate of  $10^{-4}$  and decay the learning rate with 0.1 at the 20th epochs. We first train the whole model at QP = 37 from scratch and the models at QP = 22, 27 and 32 are fine-tuned based on this trained model. Our proposed network is implemented based on Pytorch and optimized by the Adam optimizer via minimizing the  $l_1$  loss. All the models are trained with the NVIDIA GTX 2080TI GPU. Moreover, all models are only trained to enhance the quality of the U-image and they are also applied to improve the quality of the V-image.

##### C. Performance Comparison

To make a comprehensive comparison, three popular image datasets are utilized in our experiments, including 1) the classical image dataset, containing 9 color images as shown in

TABLE I  
COMPARISON OF BD-RATE.

	ASN-1		Ours-1		ASN-2		Ours-2	
	U	V	U	V	U	V	U	V
Classical	-9.06%	-8.91%	-9.60%	-9.68%	<b>-22.38%</b>	-11.39%	-21.92%	<b>-14.33%</b>
McMater	-9.77%	-12.01%	-10.23%	-12.28%	-30.02%	-15.51%	<b>-30.99%</b>	<b>-17.56%</b>
Kodak	-14.67%	-15.07%	-15.23%	-15.47%	-33.21%	-17.51%	<b>-33.98%</b>	<b>-18.33%</b>
<b>Average</b>	-11.17%	-12.00%	-11.68%	-12.47%	-28.54%	-14.80%	<b>-28.96%</b>	<b>-16.74%</b>

TABLE II  
COMPARISON OF  $\Delta$ PSNR (dB) AT QP 22.

	ASN-1		Ours-1		ASN-2		Ours-2	
	U	V	U	V	U	V	U	V
Classical	0.191	0.164	0.187	0.164	0.466	0.133	<b>0.491</b>	<b>0.203</b>
McMater	0.244	0.364	0.237	0.349	0.860	0.322	<b>0.940</b>	<b>0.427</b>
Kodak	0.371	0.460	0.365	0.441	0.919	0.455	<b>1.035</b>	<b>0.498</b>
<b>Average</b>	0.269	0.329	0.263	0.318	0.748	0.303	<b>0.822</b>	<b>0.376</b>

TABLE III  
COMPARISON OF  $\Delta$ PSNR (dB) AT QP 27.

	ASN-1		Ours-1		ASN-2		Ours-2	
	U	V	U	V	U	V	U	V
Classical	0.304	0.269	0.319	0.297	<b>0.791</b>	0.325	0.781	<b>0.444</b>
McMater	0.388	0.516	0.401	0.527	1.369	0.653	<b>1.430</b>	<b>0.752</b>
Kodak	0.515	0.567	0.530	0.582	1.304	0.632	<b>1.367</b>	<b>0.688</b>
<b>Average</b>	0.402	0.450	0.417	0.469	1.155	0.537	<b>1.193</b>	<b>0.628</b>

TABLE IV  
COMPARISON OF  $\Delta$ PSNR (dB) AT QP 32.

	ASN-1		Ours-1		ASN-2		Ours-2	
	U	V	U	V	U	V	U	V
Classical	0.337	0.360	0.360	0.384	<b>0.948</b>	0.509	0.909	<b>0.623</b>
McMater	0.469	0.568	0.503	0.589	1.641	0.802	<b>1.692</b>	<b>0.908</b>
Kodak	0.571	0.619	0.606	0.645	1.506	0.784	<b>1.528</b>	<b>0.797</b>
<b>Average</b>	0.459	0.515	0.490	0.539	1.365	0.699	<b>1.376</b>	<b>0.776</b>

TABLE V  
COMPARISON OF  $\Delta$ PSNR (dB) AT QP 37.

	ASN-1		Ours-1		ASN-2		Ours-2	
	U	V	U	V	U	V	U	V
Classical	0.349	0.392	0.404	0.467	0.946	0.620	<b>0.947</b>	<b>0.718</b>
McMater	0.493	0.589	0.522	0.595	1.576	0.828	<b>1.628</b>	<b>0.925</b>
Kodak	0.585	0.674	0.598	0.685	1.401	0.775	<b>1.453</b>	<b>0.861</b>
<b>Average</b>	0.476	0.552	0.508	0.582	1.308	0.741	<b>1.343</b>	<b>0.835</b>

Fig. 6, 2) McMater dataset [25], containing 18 color images, and 3) Kodak dataset [26], containing 24 color images. These test datasets are compressed by HEVC at QP 22, 27, 32 and 37. The default HM is set as benchmark, and we test the quality enhancement performance in terms of BD-Rate and  $\Delta$ PSNR. The proposed approach is compared with adaptive-switching neural network (ASN) [12] which also introduces the guidance image to enhance the quality of the HEVC-compressed images.

We present the average BD-rate and average  $\Delta$ PSNR results over all the test images of each dataset in Tables I-V, respectively, where Ours-1 and Ours-2 represent our proposed method without and with the guidance of the Y-image, respectively. Similarly, we use ASN-1 and ASN-2 to represent the ASN method without and with the guidance of the Y-image, respectively.

Firstly, one can see from Table I that our proposed method achieves 28.96% (on average) BD-rate gain over HEVC for the U-images and 16.74% (on average) gain for the V-images. Moreover, for most test image datasets, our approach offers higher BD-rate gain than ASN, not only for the processing of the U-images, but also for the processing of the V-images. Secondly, according to the  $\Delta$ PSNR results presented in Tables II-V, our proposed method does not perform as well as ASN when it is applied to process the chrominance images without the guidance image, especially at QP = 22. After the guidance image is introduced in our method, the significant performance gain is achieved. Meanwhile, when the images

are compressed with larger QP, our method achieves higher  $\Delta$ PSNR (on average). Thirdly, all the results given in Tables I-V demonstrate that employing the luminance image to guide the enhancement of the chrominance images can provide remarkable gain for our proposed method.

## V. CONCLUDING REMARKS

In this paper, we propose a luminance-guided chrominance image enhancement method for HEVC intra coding. We design the recursive asymmetric-convolution block to initially restore each degraded chrominance image. Then, the compressed luminance image is introduced as the guidance to further improve the quality of the chrominance image. The experimental results demonstrate that our proposed approach achieves 28.96% and 16.74% BD-rate gains over HEVC for the U- and V-images, respectively, when it is adopted in the compression of color images.

## REFERENCES

- [1] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649-1668, Dec. 2012.
- [2] B. Bross, J. Chen, J. R. Ohm, G. J. Sullivan, and Y. K. Wang, "Developments in international video coding standardization after AVC, with an overview of Versatile Video Coding (VVC)," *Proceedings of the IEEE*, vol. 109, no. 9, pp. 1463-1493, Sep. 2021.
- [3] C. Dong, Y. Deng, C. C. Loy, and X. Tang, "Compression artifacts reduction by a deep convolutional network," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015.

- [4] Y. Wang, S. Zhu, X. Meng, B. Zeng, Y. Guo, and R. Xiong, "Color image compression with transform domain down-sampling and deep convolutional reconstruction," in *Proc. IEEE Vis. Commun. Image Process.*, Dec. 2019.
- [5] W. Park and M. Kim, "CNN-based in-loop filtering for coding efficiency improvement," *IEEE Image, Video, Multidimens. Signal Process. Workshop*, Jul. 2016.
- [6] C. Fu et al., "Sample adaptive offset in the HEVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1755-1764, Dec. 2012.
- [7] X. Meng, C. Chen, S. Zhu, and B. Zeng, "A new HEVC in-loop filter based on multi-channel long-short-term dependency residual networks," in *Proc. Data Compress. Conf.*, Mar. 2018.
- [8] Y. Dai, D. Liu, and F. Wu, "A convolutional neural network approach for post-processing in HEVC intra coding," in *International Conference on Multimedia Modeling*, 2017.
- [9] T. Wang, M. Chen, and H. Chao, "A novel deep learning-based method of improving coding efficiency from the decoder-end for HEVC," in *Proc. Data Compress. Conf.*, May 2017.
- [10] C. Jia, S. Wang, X. Zhang, S. Wang, J. Liu, S. Pu, and S. Ma, "Content-aware convolutional neural network for in-loop filtering in high efficiency video coding," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3343-3356, Jul. 2019.
- [11] L. Ma, Y. Tian, P. Xing, and T. Huang, "Residual-based post-processing for HEVC," *IEEE Trans. Multimedia*, vol. 26, no. 4, pp. 67-79, Oct. 2019.
- [12] W. Lin et al., "Partition-aware adaptive switching neural networks for post-processing in HEVC," *IEEE Trans. Multimedia*, vol. 22, no. 11, pp. 2749-2763, Nov. 2020.
- [13] X. Meng, X. Deng, S. Zhu, X. Zhang, and B. Zeng, "A robust quality enhancement method based on joint spatial-temporal priors for video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 6, pp. 2401-2414, Aug. 2020.
- [14] J. Kim, J. K. Lee, and K. M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016.
- [15] X. Ding, Y. Guo, G. Ding, and J. Han, "ACNet: Strengthening the kernel skeletons for powerful CNN via asymmetric convolution blocks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Aug. 2019.
- [16] X. Ding, X. Zhang, J. Han, and G. Ding, "Diverse branch block: Building a convolution as an inception-like unit," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, Mar. 2021.
- [17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, Dec. 2015.
- [18] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: residual learning of deep CNN for image denoising," *IEEE Trans. Image Processing*, vol. 26, no. 7, pp. 3142-3155, Jul. 2017.
- [19] Y. Tai, J. Yang, X. Liu, and C. Xu, "MemNet: A persistent memory network for image restoration," in *Proc. IEEE Int. Conf. Comput. Vis.*, Aug. 2017.
- [20] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proc. Eur. Conf. Comput. Vis.*, Jul. 2018.
- [21] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011-2023, Apr. 2019.
- [22] T. Tong, G. Li, X. Liu, and Q. Gao, "Image super-resolution using dense skip connections," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017.
- [23] C. Rosewarne, B. Bross, M. Naccari, K. Sharman, and G. Sullivan, "High Efficiency Video Coding (HEVC) test model 16 (HM 16) encoder description update 7," document JCTVC-AB1002, Joint Collaborative Team on Video Coding (JCTVC), Jan. 2017.
- [24] R. Timofte et al., "NTIRE 2017 challenge on single image super-resolution: methods and results," in *Proc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jul. 2017.
- [25] L. Zhang, X. Wu, A. Buades, and X. Li, "Color demosaicking by local directional interpolation and nonlocal adaptive thresholding," *Journal of Electronic imaging*, vol. 20, no. 2, pp. 023016-023016, Apr. 2011.
- [26] R. Franzen. (1999). *Kodak Lossless True Color Image Suite*. [Online]. Available: <http://r0k.us/graphics/kodak>