# TASK-ORIENTED NEAR-LOSSLESS BURST COMPRESSION

*Weixin Jiang*[1,⋆], *Gang Wu*[2,⋆], *Vishy Swaminathan*[2], *Stefano Petrangeli*[2], *Haoliang Wang*[2], *Ryan A. Rossi*[2] *and Nedim Lipka*[2]

[1]Department of Computer Science, Northwestern University, USA
[2]Adobe Research, Adobe Inc., USA
⋆These authors are considered joint first authors

## ABSTRACT

Unlike single images, capturing bursts enable many possible downstream tasks (e.g. superresolution, HDR enhancement) due to the rich information preserved in the consecutive frames. Efficient compression of these bursts is therefore essential given the additional frames to store. In this paper, we propose a novel near-lossless compression method that can preserve the most relevant information in the burst to enable multiple downstream image enhancement tasks, while at the same time reducing the file size. Specifically, we propose a two-bitstream near-lossless compression pipeline that controls the image-space distortion at frame level, and introduce the Lipschitz condition to bound the task-space distortion at burst level. Experiments conducted on a real-world burst dataset confirm the benefit of the proposed solution in terms of rate-distortion both in the burst frame space and the superresolution task space, a popular downstream task in burst processing.

***Index Terms***— Burst, near-lossless image compression, Lipschitz condition, Bayer raw images

## 1. INTRODUCTION

Burst-based imaging is one of the key techniques in modern computational photography to overcome the limitations of mobile devices' cameras. Capturing a group of images temporally close to each other, a so-called *burst*, enables very complex processing like superresolution and low-light photography with visual results unreachable with a single input image. This happens because burst frames present sub-pixel shifts with respect to each other, e.g., due to camera motion, thus providing different samplings of the captured scene.

However, the increased number of captured images represents a challenge in terms of storage. For this reason, most mobile devices today simply discard the original burst once it is processed at capture time. However, storing the original burst could be useful to enable further downstream tasks at a later stage, like super-resolution, focus switch, lighting adjustment, denoising, or other image enhancement tasks.

In this paper, we therefore propose a burst-specific compression method that is able to consistently reduce the overall burst size while preserving the necessary information to enable multiple high-quality downstream image enhancement tasks. Traditional compression techniques focus on either lossless compression or lossy compression optimized for human perception. In our problem instead, the compressed output is to be consumed by a downstream image processing algorithm, which requires us to selectively preserve information in the burst that is relevant for the task. We exploit this aspect to design a compression scheme that results in better performance compared to state-of-the-art methods both in terms of compression and distortion in the downstream task space. Particularly, we want the output of a downstream image enhancement task starting from our compressed burst to be as close as possible to the output starting from the uncompressed burst. Our proposed approach is based on a two-bitstream technique that allows us to reach near-lossless performance without compromising on compression. Moreover, we design the Lipschitz condition to predict the distortion of the compressed burst in the downstream task by simply considering the distortion of the burst frames compared to their uncompressed version, i.e., without actually processing the burst.

We summarize the contributions of our work as follows:

- We introduce the Lipschitz condition for our problem to bound the distortion in the downstream task space based on the distortion in the burst images space;
- We propose a two-bitstream near-lossless burst compression pipeline, which contains a lossy compression module and a residual coding module. This design allows us to trade-off between compression ratio and distortion of the lossy reconstruction via a hyperparameter and reach near-lossless performance without compromising on compression;
- Our approach is the first solution for near-lossless compression of Bayer RAW images, the most common uncompressed image format produced by digital cameras;
- We evaluate our network considering superresolution as a downstream task, and show the effectiveness of our model in both image space and the task space.

The work was done when Weixin Jiang was an intern at Adobe Research.

## 2. RELATED WORK

Image compression algorithms have been studied for decades. The key idea behind compression is to exploit the redundancy in the input image. Usually, images are transformed to a feature space by a linear or nonlinear transformation, followed by an entropy coding process, such as arithmetic coding [1] and asymmetric numeral system [2]. Existing approaches can be broadly divided into lossless or lossy.

Lossless methods aim at reducing the storage size without any alteration or loss in the original image. JPEG2000 [3] applies the discrete wavelet transform to capture the local image statistics. WebP [4] improves the compression ratio by using the intra-frame coding of the VP8 video format [5]. FLIF [6] is built on the MANIAC entropy coding algorithm and is the current state-of-the-art for lossless compression. Several recent works use Convolutional Neural Networks (CNN) to better capture the hidden statistics of the image and achieve higher compression. Pixel CNN and its variants [7, 8, 9] aim to predict the distribution of unseen pixel value conditioned on previously predicted pixels. Despite good compression performance, these methods are slow in practice.

While lossless methods do not introduce any distortion, the compression ratio that can be obtained using lossy methods is much higher, which makes them the preferred choice for constrained settings like mobile devices. By adding a quantization step into a lossless pipeline, it is possible to increase the compression ratio, at the cost of increased distortion compared to the original image. CNN-based lossy image compression algorithms adopt an autoencoder architecture, in which the key step to maximize compression is the entropy estimation techniques. The hyperprior model [10] extracts an image-specific prior from the given image and uses the prior to estimate the marginal distribution of the latent representation. The joint models [11, 12, 13, 14] introduce the context modeling to the hyperprior model for a better marginal distribution estimation. However, such CNN-based autoencoder architecture is usually not invertible, making it difficult to explicitly control the distortion of the decompressed image.

Even though existing lossy image compression methods can obtain competitive compression ratio, they are designed to optimize between compression ratio and distortion compared to the original input image. On the other hand, in our burst compression case, the compressed burst is to be further processed by a downstream image processing algorithm. Therefore, in our work, we propose a novel compression technique that is able to find the right trade-off between compression ratio of the burst and reconstruction quality of the output of a given downstream processing task (e.g., superresolution). Our approach is designed to reach near-lossless performance, meaning that the absolute difference between the original and compressed image can be controlled and be bounded by a single value referred as $\tau$.

## 3. TASK-ORIENTED NEAR-LOSSLESS BURST COMPRESSION

In this paper, we propose a two-stage framework for the task-oriented near-lossless compression of bursts, as shown in Fig. 1. Assume $I_j$ denotes a set of burst frames, where $j$ denotes the id of the frame. The image compression model is then denoted as $(E, D)$, where $E, D$ denote the encoding and decoding modules, respectively. The encoding module transforms the image data into a feature vector, while the decoding module transforms the feature vector back to the image space. For conventional image compression algorithms, distortion is measured directly in the image space, i.e., $d_{img} = \|D(E(\{I_j\})) - \{I_j\}\|_p$, where $\|\cdot\|_p$ denotes the $L^p$ norm. In this case, it is easy to bound the distortion by pixel-wise manipulation. In the task-oriented case instead, a downstream task $T$ is introduced to transform the burst images from the image space to the task space, which entails that the distortion is measured in the task space, i.e., $d_{task} = \|T(D(E(\{I_j\}))) - T(\{I_j\})\|_p$. Bounding the distortion in the task space by operating on the pixels in the image space is a challenging problem given that the downstream task is often complex and highly non-linear (e.g., a neural network). The goal of our approach is to overcome this limitation and be able to control the task space distortion from the image space. We first show how to bound the distortion in the task space, and later on describe our near-lossless pipeline.

### 3.1. Image-space Distortion to Task-space Distortion

In this section, we show how to estimate the error bound of the task-space distortion based on the error bound of the image-space distortion. If the downstream task $T$ is a linear transformation, it is possible to find the closed form of the correlation between the per-pixel variation in the image space with that in the task space. Without loss of generality, we first assume the downstream task is the bilinear interpolation, in which each pixel in the task space is interpolated by its adjacent pixels in the image space. We denote by $(x, y)$ and $(u, v)$ the coordinates of the pixels of the image in the image space and in the task space, respectively. In bilinear interpolation, the pixel $(u_{2t}, v_{2t})$ in the task space, of which the pixel value is $P_{2t,2t}$, is interpolated by the pixels $\{(x_t, y_t), (x_{t+1}, y_t), (x_t, y_{t+1}), (x_{t+1}, y_{t+1})\}$ in the image space, of which the pixel values are $\{Q_{t,t}, Q_{t+1,t}, Q_{t,t+1}, Q_{t+1,t+1}\}$. More formally:

$$\frac{\partial P_{2t,2t}}{\partial Q_{t+k,t+l}} = \frac{(-1)^{k+l}(x_{t+1-k} - u_{2t})(y_{t+1-l} - v_{2t})}{(x_{t+1} - x_t)(y_{t+1} - y_t)} \qquad (1)$$

where $k, l \in \{0, 1\}$. From Eq. 1, it is possible to see that the variation of the pixel value in the task space is linearly related to the variations of the pixel values in the image space. With $u_{2t} \in [x_t, x_{t+1}]$, $v_{2t} \in [y_t, y_{t+1}]$, we have $\left|\frac{\partial P_{2t,2t}}{\partial Q_{t+k,t+l}}\right| \leq 1$. If the distortion in the image space is bounded by $\epsilon$, i.e., $d_{img} \leq \epsilon$, then the distortion in the task space is bounded by $4\epsilon$, i.e., $d_{task} = \sum_{k,l} \frac{\partial P_{2t,2t}}{\partial Q_{t+k,t+l}} \epsilon \leq 4\epsilon$. Therefore, if the downstream task is a linear transformation, it is possible to bound the per-pixel variation in the task space by bounding the per-pixel variation in the image space.

In the burst processing domain though, the downstream task is often a non-linear transformation (e.g., a neural network), which entails that $d_{task} = \max_{u,v} \sum_{x,y} \frac{\partial P_{u,v}}{\partial Q_{x,y}} \epsilon$ (where $P_{u,v}$ and $Q_{x,y}$ denote the pixel value in the task and the image space, respectively). Given the complexity of the downstream task, it might not be possible to compute this partial derivative, which indicates the pixel-wise correlation between the image and task space. Inspired by previous works on the robustness of neural networks [15, 16, 17], we introduce the Lipschitz continuity of a neural network, and bound the per-pixel variation in the task space with the estimated tight bound of the Lipschitz constant. Particularly, for a given function $f : \mathbb{R}^n \to \mathbb{R}^m$, if there exists a non-negative constant $L \geq 0$ such that:

$$\|f(X) - f(Y)\|_p \leq L\|X - Y\|_p, \quad \forall X, Y \in \mathbb{R}^n \tag{2}$$

where the function $f$ is Lipschitz continuous on $\mathbb{R}^n$, and the smallest such $L$ is called the Lipschitz Constant (LP) of $f$. The Lipschitz constant is the maximum ratio between variations in the output space and the variations in the input space and thus is a measure of sensitivity of the function with respect to input perturbations. In linear transformations, the Lipschitz constant is easy to estimate. Neural networks can be divided into linear operators (e.g., convolutions) and nonlinear operators (e.g., activation functions). The difficulty of estimating the Lipschitz constant of a neural network lies therefore in the non-linear activation functions. Previous works found that even though activation functions are nonlinear in nature, they are usually slope restricted:

$$\alpha \leq \frac{\varphi(X) - \varphi(Y)}{X - Y} \leq \beta, \forall X, Y \in \mathbb{R}, \tag{3}$$

where $\varphi : \mathbb{R} \to \mathbb{R}$ can be a nonlinear function, and $0 \leq \alpha < \beta < \infty$. In particular, the activation functions ReLU, tanh and sigmoid are all slope restricted with $\alpha = 0$, $\beta = 1$. Such slope-restricted non-linearities enable us to estimate a tight bound on the Lipschitz constant [17]. However, computing the Lipschitz constant of a neural network would require quadratic time, according to the number of neurons in the network, which is infeasible in practice. For this reason, in this work, we estimate the Lipschitz constant of the downstream task in a numerical manner. In particular, we add uniform noise to the input of the network with different means and measure the corresponding variations in the task space. Therefore, if the Lipschitz continuity holds for the downstream task, with corresponding Lipschitz constant $L$, we can bound the task-space distortion by manipulating image-space pixel-wise distortion, i.e., $d_{task} \leq L d_{img}$.

### 3.2. A Near-lossless Image Compression Framework

We now introduce our near-lossless image compression framework, which targets the following objective function:

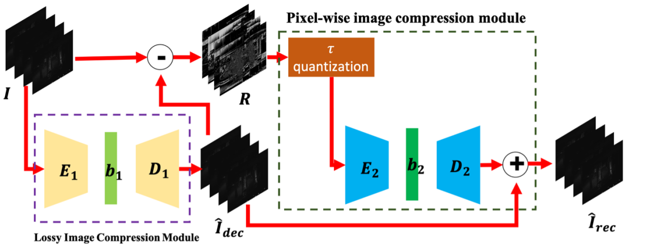

**Fig. 1**. Overview of proposed near-lossless image compression system.

$$\begin{aligned} &\min_E H(E(\{I_j\})) \\ &\text{s.t. } \|T(D(E(\{I_j\}))) - T(\{I_j\})\|_p < \varepsilon \end{aligned} \tag{4}$$

where $H$ denotes encoding costs. Assuming the Lipschitz continuity holds for the downstream task, we can rewrite the objective function as:

$$\begin{aligned} &\min_E H(E(\{I_j\})) \\ &\text{s.t. } \|D(E(\{I_j\})) - \{I_j\}\|_p < \frac{\varepsilon}{L} \end{aligned} \tag{5}$$

Inspired by previous works on ”lossy plus residual” coding schemes [18, 19], we propose a two-stage framework for our compression pipeline. As shown in Fig. 1, the first stage utilizes a lossy image compression module to reduce the redundancy in the source image, in which we may apply any flexible lossy compression algorithms (e.g. JPEG, lossy FLIF etc.). The second stage is a pixel-wise image compression that focuses on encoding the residuals between the source image and the decoded image from the first stage. Particularly, the residual map is first fed into the $\tau$ quantization module, in which a binning process is implemented for error control, and then the quantized residual map is fed into a lossless image compression module. Thus, in the end, we use two bitstreams to encode the source image. The advantages of using this two-stage framework for near lossless image compression are two-fold. First, any lossy compression scheme (either traditional or CNN-based) can be employed in the first stage, which enables the framework to be applied in multiple compression scenarios. Second, unlike existing learned lossy compression methods, our framework introduces invertibility, allowing for controllable distortion without need for retraining.

In particular, the $\tau$ quantization module is formulated as:

$$\hat{R} = \operatorname{sgn}(R)(2\tau + 1)\left\lfloor \frac{|R| + \tau}{2\tau + 1} \right\rfloor, \tag{6}$$

where $R$ and $\hat{R}$ denote the residual map before and after quantization, respectively, $\operatorname{sgn}()$ denotes the sign function and $\lfloor\rfloor$ the maximum integer that is less than or equal to the given variable. With the $\tau$ quantization module, the per-pixel difference between $R$ and $\hat{R}$ is bounded by $\tau$. To ensure the compression ratio of the second stage increases with increased tolerance on the distortion error, we apply a PMF quantization along with the $\tau$ quantization module. Assume $r_{x,y}$ and $\hat{r}_{x,y}$ denote the pixel values of the residual map $R$ and $\hat{R}$ at position $(x, y)$, respectively. According to the quantization scheme at Eq. 6, we have $\|\hat{r}_{x,y} - r_{x,y}\| \leq \tau$. Similar to [8], the discretized logistic mixture model is introduced to estimate the distribution of each residual. Note that each burst frame is in the RAW format instead of standard RGB format, which contains four RGGB channels. Thus, we may factorize the distribution of each pixel (i.e. $Prob(\hat{r}_{x,y})$) as the product of the distribution of each sub-pixel.

$$\begin{aligned} Prob(\hat{r}_{x,y}) = &\, Prob(\hat{r}_{x,y}^{c_r} | \mu^{c_r}(C_{x,y}), s^{c_r}(C_{x,y})) \\ &\times Prob(\hat{r}_{x,y}^{c_{g_1}} | \mu^{c_{g_1}}(C_{x,y}, \mu^{c_r}), s^{c_{g_1}}(C_{x,y})) \times ... \end{aligned} \tag{7}$$

where $c_r, c_{g_1}, c_{g_2}, c_b$ denote the four RGGB channels. $C_{x,y}$ denotes the context information. Then we adapt the cross-channel auto-regression model for parameters estimation,

$$\begin{aligned} \mu^{c_{g_1}} &= \mu^{c_{g_1}}(C_{x,y}) + \alpha_1(C_{x,y})\hat{r}_{x,y}^{c_r} \\ \mu^{c_{g_2}} &= \mu^{c_{g_2}}(C_{x,y}) + \alpha_2(C_{x,y})\hat{r}_{x,y}^{c_r} + \beta_1(C_{x,y})\hat{r}_{x,y}^{c_{g_1}} \\ \mu^{c_b} &= \mu^{c_b}(C_{x,y}) + \alpha_3(C_{x,y})\hat{r}_{x,y}^{c_r} + \beta_2(C_{x,y})\hat{r}_{x,y}^{c_{g_1}} \\ &+ \gamma(C_{x,y})\hat{r}_{x,y}^{c_{g_2}}. \end{aligned} \tag{8}$$

## 4. EXPERIMENTAL RESULTS

To evaluate the benefits of the proposed burst compression approach, we choose super-resolution as a downstream task, given its popularity. Specifically, we choose the Deep Burst SR network from Bhat et al. [20], which takes a set of burst frames and generates a single super-resoluted frame.

First, we aim to investigate the impact of distortion on the individual burst frames on the final distortion of the super-resolute image generated by Deep Burst SR. Following similar convention in the near-lossless compression domain, we compute distortion $\tau$ as the largest pixel-level difference between the original and the compressed image (i.e., the H-infinity norm). As shown in Fig. 2(a), given the same level of distortion in the image space, the distortion in the task space may vary. Next, in Fig. 2(b), we compute the maximum ratio between the variations in the task space and those in the image space at different distortion levels. Notably, the sensitivity of the given neural network decreases when the distortion level increases as shown in Fig. 2(b).

To implement our method, we use the lossy FLIF algorithm (with quality setting = 25) as the lossy compression module in the first stage and adapt the PixelCNN++ algorithm as the lossless image compression model for the generation of the residual stream in the second stage. Particularly, we extended the original Pixel CNN++, which is designed to work on 8-bit RGB images, to 10-bit Bayer RAW RGGB images[1], the standard uncompressed output of modern digital cameras. We use the public HDR+ dataset [21] for our experiments, which contains 3640 10-bit RGGB bursts. 80% of the bursts are used for training and 20% for validation and testing.

For evaluation, we choose JPEG LS [22] as our baseline method, where we use its near-lossless mode to control the maximum pixel-level distoration, Tau. We evaluate the two methods on a set of bursts consists of 10bit Bayer raw images. The results of this analysis are as shown in Fig. 3, where compression rate is measured with bis per sub-pixel (BPSP) and distoration is measured in terms of Tau. As we can see, our proposed method still under-performs the highly-optimized hand-crafted codec JPEG LS. However, our proposed solution is the first learning-based solution for near-lossless compression of Bayer raw images.

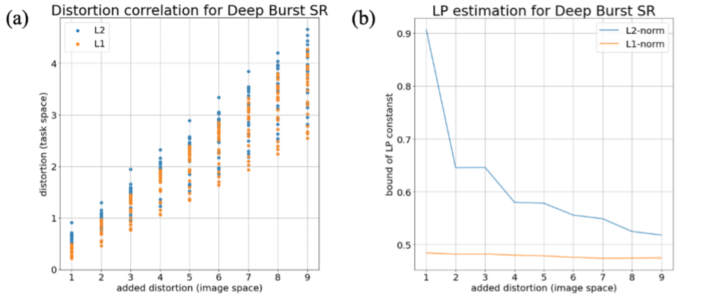

**Fig. 2**. Lipschitz constant estimation when considering super-resolution as the burst downstream processing task.

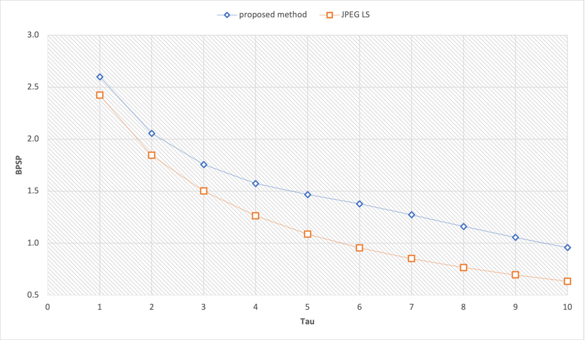

**Fig. 3**. Proposed Nearlossless Compression method's BPSP over Tau on 10bit Burst images.

## 5. CONCLUSION

In this work, we presented a novel approach for the near-lossless compression of bursts. Since bursts are meant to be processed by a downstream image processing algorithm, we design a two-stage pipeline that controls the image-space distortion of the individual burst frames while guaranteeing a specific level of distortion in the task space. This is obtained by introducing the Lipschitz condition for our problem that relates task space distortion to image space distortion. Moreover, our approach represents the first attempt at the near-lossless compression of Bayer RAW images, the most common uncompressed output format of digital cameras. Experiments on the HDR+ burst dataset confirms the effectiveness of our scheme.

[1] https://en.wikipedia.org/wiki/Bayer_filter

## 6. REFERENCES

[1] Ian H Witten, Radford M Neal, and John G Cleary, "Arithmetic coding for data compression," *Communications of the ACM*, vol. 30, no. 6, pp. 520–540, 1987.

[2] Jarek Duda, "Asymmetric numeral systems," *arXiv preprint arXiv:0902.0271*, 2009.

[3] Athanassios Skodras, Charilaos Christopoulos, and Touradj Ebrahimi, "The jpeg 2000 still image compression standard," *IEEE Signal processing magazine*, vol. 18, no. 5, pp. 36–58, 2001.

[4] Google Developers, "A new image format for the web," *https://developers.google.com/speed/webp/*, 2010.

[5] Google Inc., "Vp8 data format and decoding guide," *https://datatracker.ietf.org/doc/html/rfc6386*, 2011.

[6] Jon Sneyers and Pieter Wuille, "Flif: Free lossless image format based on maniac compression," in *2016 IEEE international conference on image processing (ICIP)*. IEEE, 2016, pp. 66–70.

[7] Aaron Van Oord, Nal Kalchbrenner, and Koray Kavukcuoglu, "Pixel recurrent neural networks," in *International conference on machine learning*. PMLR, 2016, pp. 1747–1756.

[8] Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P Kingma, "Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications," *arXiv preprint arXiv:1701.05517*, 2017.

[9] Alexander Kolesnikov and Christoph H Lampert, "Pixelcnn models with auxiliary variables for natural image modeling," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1905–1914.

[10] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston, "Variational image compression with a scale hyperprior," *arXiv preprint arXiv:1802.01436*, 2018.

[11] David Minnen, Johannes Ballé, and George D Toderici, "Joint autoregressive and hierarchical priors for learned image compression," *Advances in neural information processing systems*, vol. 31, 2018.

[12] Jooyoung Lee, Seunghyun Cho, and Seung-Kwon Beack, "Context-adaptive entropy model for end-to-end optimized image compression," *arXiv preprint arXiv:1809.10452*, 2018.

[13] Fabian Mentzer, Eirikur Agustsson, Michael Tschannen, Radu Timofte, and Luc Van Gool, "Conditional probability models for deep image compression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4394–4402.

[14] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto, "Learned image compression with discretized gaussian mixture likelihoods and attention modules," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7939–7948.

[15] Aladin Virmaux and Kevin Scaman, "Lipschitz regularity of deep neural networks: analysis and efficient estimation," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[16] Dongmian Zou, Radu Balan, and Maneesh Singh, "On lipschitz bounds of general convolutional neural networks," *IEEE Transactions on Information Theory*, vol. 66, no. 3, pp. 1738–1759, 2019.

[17] Mahyar Fazlyab, Alexander Robey, Hamed Hassani, Manfred Morari, and George Pappas, "Efficient and accurate estimation of lipschitz constants for deep neural networks," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[18] Paul W Melnychuck and Majid Rabbani, "Survey of lossless image coding techniques," in *Digital Image Processing Applications*. SPIE, 1989, vol. 1075, pp. 92–100.

[19] Yuanchao Bai, Xianming Liu, Wangmeng Zuo, Yaowei Wang, and Xiangyang Ji, "Learning scalable ly=-constrained near-lossless image compression via joint lossy image and residual compression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11946–11955.

[20] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte, "Deep burst super-resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9209–9218.

[21] Samuel W. Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan T. Barron, Florian Kainz, Jiawen Chen, and Marc Levoy, "Burst photography for high dynamic range and low-light imaging on mobile cameras," *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, vol. 35, no. 6, 2016.

[22] Marcelo J Weinberger, Gadiel Seroussi, and Guillermo Sapiro, "The loco-i lossless image compression algorithm: Principles and standardization into jpeg-ls," *IEEE Transactions on Image processing*, vol. 9, no. 8, pp. 1309–1324, 2000.