

# Compression for Multiple Reconstructions

Yehuda Dar, Michael Elad, and Alfred M. Bruckstein  
Computer Science Department, Technion – Israel Institute of Technology

**Abstract**—In this work we propose a method for optimizing the lossy compression for a network of diverse reconstruction systems. We focus on adapting a standard image compression method to a set of candidate displays, presenting the decompressed signals to viewers. Each display is modeled as a linear operator applied after decompression, and its probability to serve a network user. We formulate a complicated operational rate-distortion optimization trading-off the network’s expected mean-squared reconstruction error and the compression bit-cost. Using the alternating direction method of multipliers (ADMM) we develop an iterative procedure where the network structure is separated from the compression method, enabling the reliance on standard compression techniques. We present experimental results showing our method to be the best approach for adjusting high bit-rate image compression (using the state-of-the-art HEVC standard) to a set of displays modeled as blur degradations.

**Index Terms**—Rate-distortion optimization, signal compression, image compression, image deblurring, alternating direction method of multipliers (ADMM).

## I. INTRODUCTION

Multimedia content is often distributed using broadcast and “on-demand” services reaching consumers with various display devices. Therefore, rendering the image/video can widely differ due to various technical aspects such as the specific display technology, different screen resolutions, etc. Such multimedia distribution systems fundamentally rely on lossy compression in order to meet storage and transmission-bandwidth limitations. However, while the displayed signals are the important outcomes of the flow, the compression is usually optimized only with respect to the decompressed signal, ignoring the subsequent processing and degradations occurring at the different displays. In this work we study the problem of optimizing signal compression to a known set of display settings having different usage probabilities.

We recently [1] proposed an optimization methodology to adjust standard image/video compression to a known type of display presenting the decompressed signal to the viewer. Our framework essentially pre-compensates the display degradation from the compression standpoint in a rate-distortion optimized manner. Here we extend the problem settings of [1] to optimize the compression with respect to a set of display devices, described by several linear rendering models and their probabilities to be in use by consumers. One can interpret the display models and their usage probabilities as a characterization of a multimedia distribution network.

We formulate a rate-distortion optimization to trade-off the compression bit-cost and the expected mean-squared error of

the displayed signal. Similar to our previous works [1], [2], we address the computationally hard optimization using the alternating direction method of multipliers (ADMM) [3] translating the task to sequentially applying standard compressions (that are network independent!) and  $\ell_2$ -constrained deconvolutions expressing the network structure. This procedure can be generically adapted to various network layouts and to any standard compression technique, providing network-optimized binary data that is compatible with desired standard decompression processes.

The problem settings we address here resemble the framework of (lossy) compression for computations applied on the decompressed data (see, e.g., [4], [5]). Yet, the post-decompression processing we consider here is mainly an unwanted degradation, in contrast to a desired computation. However, the ability of our method to adapt standard compression to post-decompression processing may be utilized also for compression tasks involving computations necessary to be carried out on the decompressed data.

The important problem of various display devices is treated also from the perspective of scalable image/video coding methods (see, e.g., the extension of the HEVC standard in [6]), where the signal is coded in layers of increasing quality/resolution to be peeled by the network or the user device. In contrast, we take here another viewpoint on the problem, optimizing a single (non-layered) compression of a signal to a given collection of rendering models.

We demonstrate our general approach for image compression using the state-of-the-art HEVC standard coupled with various simplified display models in the form of linear blur operators following the decompression. While another recent method [7] optimizes image rendering with respect to a perceptual quality metric, we present here (and in [1]) a method to globally optimize the flow of compression, decompression and rendering. Since our optimization goal and the distortion type differ from those in [7], the two methods cannot be quantitatively compared. In our experiments we compared our approach to regular HEVC compression, and to preceding the compression with the Expected Patch Log Likelihood (EPLL) deblurring method [8] adapted to the same fidelity term as we use in our method. The rate-distortion performance of the various methods clearly exhibit our method as the leading approach at high bit-rate compression.

## II. THE PROPOSED METHOD

### A. Problem Formulation

We consider the network structure described in Fig. 1, starting with an input signal in the form of an  $N$ -length

<sup>0</sup>Authors’ E-mail addresses: {ydar, elad, freddy}@cs.technion.ac.il.

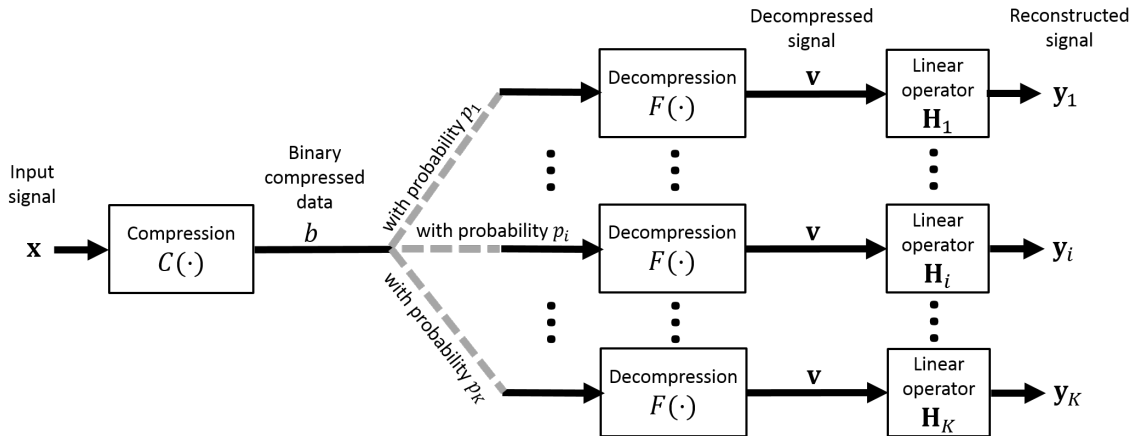


Fig. 1. The general network structure considered in this paper.

column-vector  $\mathbf{x} \in \mathbb{R}^N$  that is compressed and distributed over the network to users having various reconstruction systems. We describe the lossy compression procedure using the function  $C : \mathbb{R}^N \rightarrow \mathcal{B}$ , mapping the  $N$ -dimensional input-signal domain to the discrete set  $\mathcal{B}$  of compressed representations in the form of variable-length binary descriptions. The compression of  $\mathbf{x}$  is denoted by  $b = C(\mathbf{x})$ , where  $b \in \mathcal{B}$  is the compressed data to transmit over the network to an arbitrary number of users. The users have reconstruction systems that first decompress the data via  $\mathbf{v} = F(b)$ , where  $F : \mathcal{B} \rightarrow \mathcal{S}$  maps the binary compressed representations in  $\mathcal{B}$  to the respective decompressed signals in the discrete set  $\mathcal{S} \subset \mathbb{R}^N$ . The decompressed signal  $\mathbf{v}$  (which is an  $N$ -length column vector) further goes through a linear operation, associated with a processing/degradation stage, that produces the reconstruction available to the user. In the case of visual signals, the post-decompression component may be a display rendering the viewed image. We assume a user may have one of  $K$  reconstruction systems (where  $K$  is a positive finite integer), differing in the linear operator applied after decompression. The post-decompression linear operator of the  $k^{\text{th}}$  system type ( $k = 1, \dots, K$ ) is denoted by the  $N \times N$  matrix  $\mathbf{H}_k$ , producing a corresponding reconstructed (output) signal

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{v}. \quad (1)$$

We assume the portions of using each of the  $K$  reconstruction systems are known and denoted by  $p_1, \dots, p_K \geq 0$ , where  $\sum_{k=1}^K p_k = 1$ . Accordingly, a network user can be modeled to have a reconstruction system of a type corresponding to a discrete random variable over the values  $\{1, \dots, K\}$  with the respective probabilities  $p_1, \dots, p_K$ . Then, by (1) the reconstructed signal is a random vector  $\mathbf{y}$  having the value  $\mathbf{y}_k$  with probability  $p_k$  for  $k = 1, \dots, K$ . For a given (deterministic) input signal  $\mathbf{x}$  and its decompressed version  $\mathbf{v}$ , and by the network structure, we quantify the expected mean-squared-error (MSE) of the reconstruction as

$$D(\mathbf{x}, \mathbf{v}) \triangleq \frac{1}{N} \sum_{k=1}^K p_k \|\mathbf{x} - \mathbf{H}_k \mathbf{v}\|_2^2. \quad (2)$$

Our goal here is to optimize the rate-distortion performance of the network for a given input signal  $\mathbf{x}$ . Accordingly, we formulate the task as the minimization of the compression bit-cost under constrained expected distortion (2), namely,

$$\begin{aligned} \hat{\mathbf{v}} &= \underset{\mathbf{v} \in \mathcal{S}}{\operatorname{argmin}} R(\mathbf{v}) \\ \text{subject to} \quad & \frac{1}{N} \sum_{k=1}^K p_k \|\mathbf{x} - \mathbf{H}_k \mathbf{v}\|_2^2 \leq d \end{aligned} \quad (3)$$

where  $R(\mathbf{v})$  evaluates the length of the binary compressed description  $b \in \mathcal{B}$  matched to the decompressed signal  $\mathbf{v}$ , and  $d \geq 0$  is the allowed distortion.

Similar to contemporary compression tasks (see, e.g., [9], [10]), we turn our optimization (3) into its unconstrained Lagrangian form

$$\hat{\mathbf{v}} = \underset{\mathbf{v} \in \mathcal{S}}{\operatorname{argmin}} R(\mathbf{v}) + \lambda \frac{1}{N} \sum_{k=1}^K p_k \|\mathbf{x} - \mathbf{H}_k \mathbf{v}\|_2^2 \quad (4)$$

where  $\lambda \geq 0$  is a Lagrange multiplier matching to a distortion constraint  $d_\lambda \geq 0$  (such coding without a specified distortion constraint is prevalent, for instance, in video coding [10]). Since we consider the compression of high-dimensional signals (i.e.,  $N$  is large) the discrete set  $\mathcal{S}$  is prohibitively large, meaning that a direct solution of the Lagrangian form in (4) is impractical for arbitrarily structured matrices  $\{\mathbf{H}_k\}_{k=1}^K$ . Note that when  $\mathbf{H}_k = \mathbf{I}$  for  $k = 1, \dots, K$ , the Lagrangian optimization in (4) reduces to the standard compression form [9], [11], disregarding the network-oriented problem, and practically solvable using block-based architectures that decompose the problem to a sequence of block-level optimizations of sufficiently low dimensions.

## B. Practical Iterative Procedure

We employ the alternating direction method of multipliers (ADMM) technique [3] to resolve the computationally challenging problem (4) when the post-decompression operators  $\{\mathbf{H}_k\}_{k=1}^K$  are arbitrarily structured. We begin by splitting the

optimization variable such that (4) becomes

$$\hat{\mathbf{v}} = \underset{\mathbf{v} \in \mathcal{S}, \mathbf{z} \in \mathbb{R}^N}{\operatorname{argmin}} R(\mathbf{v}) + \lambda \frac{1}{N} \sum_{k=1}^K p_k \|\mathbf{x} - \mathbf{H}_k \mathbf{z}\|_2^2 \quad (5)$$

subject to  $\mathbf{v} = \mathbf{z}$

where  $\mathbf{z} \in \mathbb{R}^N$  is an auxiliary variable that is not limited to the discrete set  $\mathcal{S}$ . Applying the scaled form of the augmented Lagrangian and the method of multipliers [3, Ch. 2] on (5) yields an iterative process formulated as

$$\left( \hat{\mathbf{v}}^{(t)}, \hat{\mathbf{z}}^{(t)} \right) = \quad (6)$$

$$\underset{\mathbf{v} \in \mathcal{S}, \mathbf{z} \in \mathbb{R}^N}{\operatorname{argmin}} R(\mathbf{v}) + \frac{\lambda}{N} \sum_{k=1}^K p_k \|\mathbf{x} - \mathbf{H}_k \mathbf{z}\|_2^2 + \frac{\beta}{2} \|\mathbf{v} - \mathbf{z} + \mathbf{u}^{(t)}\|_2^2$$

$$\mathbf{u}^{(t+1)} = \mathbf{u}^{(t)} + \left( \hat{\mathbf{v}}^{(t)} - \hat{\mathbf{z}}^{(t)} \right), \quad (7)$$

where  $t$  denotes the iteration index,  $\mathbf{u}^{(t)} \in \mathbb{R}^N$  is the scaled dual variable, and  $\beta$  is an auxiliary parameter introduced by the augmented Lagrangian. We get the ADMM form of the problem by applying one iteration of alternating minimization on (6), leading to the following sequence of easier optimizations

$$\hat{\mathbf{v}}^{(t)} = \underset{\mathbf{v} \in \mathcal{S}}{\operatorname{argmin}} R(\mathbf{v}) + \frac{\beta}{2} \|\mathbf{v} - \tilde{\mathbf{z}}^{(t)}\|_2^2 \quad (8)$$

$$\hat{\mathbf{z}}^{(t)} = \underset{\mathbf{z} \in \mathbb{R}^N}{\operatorname{argmin}} \lambda \frac{1}{N} \sum_{k=1}^K p_k \|\mathbf{x} - \mathbf{H}_k \mathbf{z}\|_2^2 + \frac{\beta}{2} \|\mathbf{z} - \tilde{\mathbf{v}}^{(t)}\|_2^2 \quad (9)$$

$$\mathbf{u}^{(t+1)} = \mathbf{u}^{(t)} + \left( \hat{\mathbf{v}}^{(t)} - \hat{\mathbf{z}}^{(t)} \right). \quad (10)$$

where  $\tilde{\mathbf{z}}^{(t)} = \hat{\mathbf{z}}^{(t-1)} - \mathbf{u}^{(t)}$  and  $\tilde{\mathbf{v}}^{(t)} = \hat{\mathbf{v}}^{(t)} + \mathbf{u}^{(t)}$ . Nicely, the compression architecture  $\{\mathcal{S}, R\}$  and the network layout described by  $\{\mathbf{H}_k, p_k\}_{k=1}^K$  were separated by the ADMM to distinct (and simpler) optimization tasks.

The optimization formulation in (8) coincides with the Lagrangian rate-distortion optimization utilized for standard compression tasks employing the usual (network independent) MSE distortion metric (here the effective Lagrange multiplier is  $\tilde{\lambda} = \frac{\beta N}{2}$ ). Hence, we propose to replace the solution of (8) with a standard compression (and decompression) method – even one that does not exactly follow the Lagrangian optimization in (8). We refer to the standard compression and decompression as

$$b^{(t)} = \operatorname{StandardCompress} \left( \tilde{\mathbf{z}}^{(t)}, \theta \right) \quad (11)$$

$$\hat{\mathbf{v}}^{(t)} = \operatorname{StandardDecompress} \left( b^{(t)} \right) \quad (12)$$

where  $\theta$  is a parameter generalizing the Lagrange multiplier part in regulating the rate-distortion tradeoff (see Algorithm 1). The last generalizations establish the proposed procedure as a generic methodology for optimizing any compression method to particular network layouts.

The optimization in (9) can be interpreted as an extended  $\ell_2$ -constrained deconvolution problem, here including a combination of several fidelity terms associated with the degradation

operators  $\{\mathbf{H}_k\}_{k=1}^K$ . The analytic solution of (9) is

$$\hat{\mathbf{z}}^{(t)} = \left( \sum_{k=1}^K p_k \mathbf{H}_k^T \mathbf{H}_k + \frac{\beta N}{2\lambda} \mathbf{I} \right)^{-1} \left( \sum_{k=1}^K p_k \mathbf{H}_k^T \mathbf{x} + \frac{\beta N}{2\lambda} \tilde{\mathbf{v}}^{(t)} \right)$$

exhibiting it as a linear combination of  $\mathbf{x}$  and  $\tilde{\mathbf{v}}^{(t)}$ . We define the parameter  $\tilde{\beta} \triangleq \frac{\beta N}{2\lambda}$  and use it in the generic method summarized in Algorithm 1.

---

#### Algorithm 1 Generic Network-Optimized Compression

---

- 1: Inputs:  $\mathbf{x}$ ,  $\theta$ ,  $\tilde{\beta}$ .
  - 2: Initialize  $t = 0$ ,  $\hat{\mathbf{z}}^{(0)} = \mathbf{x}$ ,  $\mathbf{u}^{(1)} = \mathbf{0}$ .
  - 3: **repeat**
  - 4:    $t \leftarrow t + 1$
  - 5:    $\tilde{\mathbf{z}}^{(t)} = \hat{\mathbf{z}}^{(t-1)} - \mathbf{u}^{(t)}$
  - 6:    $b^{(t)} = \operatorname{StandardCompress} \left( \tilde{\mathbf{z}}^{(t)}, \theta \right)$
  - 7:    $\hat{\mathbf{v}}^{(t)} = \operatorname{StandardDecompress} \left( b^{(t)} \right)$
  - 8:    $\tilde{\mathbf{v}}^{(t)} = \hat{\mathbf{v}}^{(t)} + \mathbf{u}^{(t)}$
  - 9:    $\hat{\mathbf{z}}^{(t)} = \left( \sum_{k=1}^K p_k \mathbf{H}_k^T \mathbf{H}_k + \tilde{\beta} \mathbf{I} \right)^{-1} \left( \sum_{k=1}^K p_k \mathbf{H}_k^T \mathbf{x} + \tilde{\beta} \tilde{\mathbf{v}}^{(t)} \right)$
  - 10:    $\mathbf{u}^{(t+1)} = \mathbf{u}^{(t)} + \left( \hat{\mathbf{v}}^{(t)} - \hat{\mathbf{z}}^{(t)} \right)$
  - 11: **until** stopping criterion is satisfied
  - 12: Output:  $b^{(t)}$ , which is the binary compressed data obtained in the last iteration.
- 

### III. EXPERIMENTAL RESULTS

Let us demonstrate our method for optimizing the HEVC still-image compression standard (implemented in the software in [12]) to three possible blur operators degrading the decompressed image. The post-decompression linear operators  $\mathbf{H}_1$ ,  $\mathbf{H}_2$ , and  $\mathbf{H}_3$  correspond to shift-invariant Gaussian blur kernels (of  $15 \times 15$  pixels size) having standard deviations 0.6, 0.8, and 1, respectively, and usage probabilities of  $p_1 = 0.6$ ,  $p_2 = 0.3$ , and  $p_3 = 0.1$ .

To evaluate our method we constructed three competing techniques also using HEVC image compression, and compared them to our method<sup>1</sup>. The PSNR-bitrate curves of the examined methods (see, e.g., Fig. 3b) were created for each of the 12 examined images (see Table I) by applying their HEVC component for 9 quality parameter (QP) values equally-spaced between 1 to 41. The first competing approach is to regularly compress without any pre/post processing (while the decompression is still followed by the inevitable deterioration). As expected, this naive method performs poorly. The second competing procedure precedes the compression with deconvolution using the Expected Patch Log Likelihood (EPLL) method relying on a Gaussian Mixture Model (GMM) prior learned for natural images (see [8]). The EPLL implementation used here is with respect to a fidelity term corresponding to (2) and suitable parameter settings. The third competing method

<sup>1</sup>The Peak Signal-to-Noise Ratio (PSNR) here relies on the expected reconstruction MSE given in (2), i.e.,  $PSNR = 10 \log_{10} (P^2/D(\mathbf{x}, \mathbf{v}))$  where  $\mathbf{x}$  and  $\mathbf{v}$  are the input and the decompressed signals, respectively, and  $P$  is the maximal signal-value generally possible.

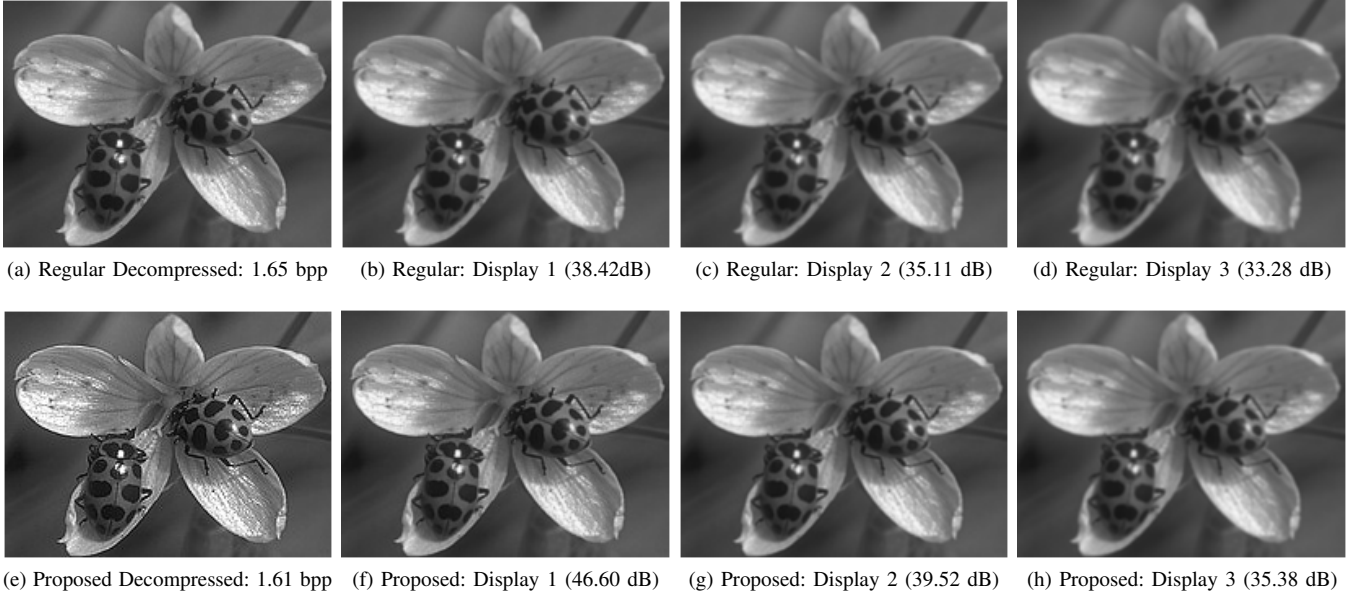


Fig. 2. The regular and the proposed method applied for the 'Flower and Bugs' image. The denoted PSNR values in this figure are for the individual reconstructions, i.e., using the regular MSE and not the expected one from (2) that is used in the rest of the paper.

is our pre-compensating compression from [1], optimized only for a single display (corresponding to the highest probability).

The implementation of the proposed method (Algorithm 1) uses a  $\tilde{\beta}$  value based on the HEVC quality parameter (the  $\tilde{\beta}$  value here is 10 times the value formulated in [1]). The stopping criterion was defined to a maximal number of 40 iterations or to end earlier when  $\hat{v}^{(t)}$  and  $\hat{z}^{(t)}$  converge or diverge (as described in [1]).

The evaluation of the PSNR-bitrate curves summarized in Table I and exemplified for one image in Fig. 3b, showing that our method outperforms the other techniques at high bit-rates, where we achieve significant PSNR gains compared to the regular, the EPLL-based, and the single display optimization procedures. The average PSNR gains in Table I were computed based on the BD-PSNR metric [13], [14] for the high bit-rate range (here defined by QP values 1,6,11,16).

In Figure 2 we present visual results for the compression of the 'Flower and Bugs' image (see Fig. 3a, where only a portion of the input image is presented due to lack of space). Figures 2a and 2e exhibit the decompressed images (before degradation) using the regular approach and the proposed method, respectively. Figures 2b-2d and 2f-2h show the three simulated displayed versions of the decompressed images. Evidently, we get significantly higher PSNR values (that correspond to the regular MSE measure) at a similar (slightly lower) bit-rate. Our method produces an overly-sharpened decompressed image (Fig. 2e) that is later balanced with the rendering blur, leading to better displayed images (Figs. 2f-2h).

#### IV. CONCLUSION

We presented a method for optimizing compression to a set of reconstruction systems, each has a different linear processing/degradation after decompression. Using ADMM

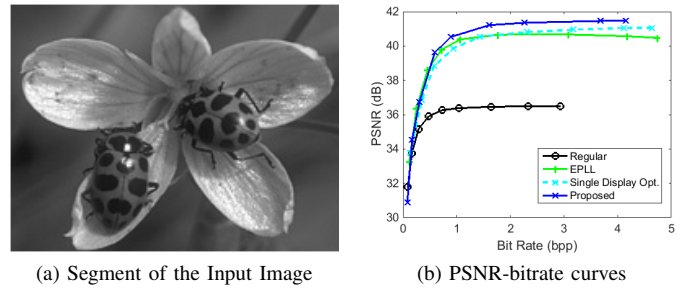


Fig. 3. Methods evaluation for the image 'Flower and Bugs'.

TABLE I  
METHOD EVALUATION FOR 12 IMAGES

Dataset	Image	Average PSNR Gains at High Bit-Rates		
		Proposed over Regular	Proposed over EPLL	Proposed over Single Display
[15] TEST IMAGES 300x300	Almonds	5.25	0.47	0.49
	Cards	4.83	0.56	0.35
	Duck toys	4.83	0.56	0.49
	Garden table	4.70	0.48	0.35
[16] UCID 384x512	House & lawn	3.53	1.36	0.25
	Tree	3.83	1.18	0.20
	Garden	3.26	1.44	0.17
	Teddy bear	4.96	0.73	0.48
[17] Berkeley 481x321	Bears	4.73	0.64	0.35
	Boats	4.05	0.90	0.28
	Butterfly	4.85	0.67	0.39
	Flower & Bugs	4.83	0.74	0.49

we established a generic compression procedure relying on a standard compression technique. Experiments for adjusting image compression (using the HEVC standard) to a set of blur operators modeling display degradations showed our method as the leading approach for high bit-rate compression.

#### REFERENCES

- [1] Y. Dar, M. Elad, and A. M. Bruckstein, "Optimized pre-compensating compression," *Submitted to IEEE Trans. Image Process., arXiv preprint arXiv:1711.07901*, 2017.
- [2] —, "System-aware compression," *arXiv preprint arXiv:1801.04853*, 2018.
- [3] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [4] V. Misra, V. K. Goyal, and L. R. Varshney, "Distributed scalar quantization for computing: High-resolution analysis and extensions," *IEEE Trans. Inf. Theory*, vol. 57, no. 8, pp. 5298–5325, 2011.
- [5] J. Z. Sun, V. Misra, and V. K. Goyal, "Distributed functional scalar quantization simplified," *IEEE Trans. Signal Process.*, vol. 61, no. 14, pp. 3495–3508, 2013.
- [6] J. M. Boyce, Y. Ye, J. Chen, and A. K. Ramasubramonian, "Overview of SHVC: Scalable extensions of the high efficiency video coding standard," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 20–34, 2016.
- [7] V. Laparra, A. Berardino, J. Ballé, and E. P. Simoncelli, "Perceptually optimized image rendering," *Journal of the Optical Society of America A*, vol. 34, p. 1511, 2017.
- [8] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *IEEE ICCV*, 2011, pp. 479–486.
- [9] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 23–50, 1998.
- [10] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.
- [11] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 9, pp. 1445–1453, 1988.
- [12] F. Bellard, "BPG 0.9.6." [Online]. Available: <http://bellard.org/bpg/>
- [13] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," in *ITU-T Q. 6/SG16 VCEG, 15th Meeting, Austin, Texas, USA, April, 2001*.
- [14] G. Valenzise, "Bjontegaard metric (Matlab function)." [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/27798-bjontegaard-metric>
- [15] N. Asuni and A. Giachetti, "TESTIMAGES: a large-scale archive for testing visual devices and basic image processing algorithms." in *Eurographics Italian Chapter Conference*, 2014, pp. 63–70.
- [16] G. Schaefer and M. Stich, "UCID: an uncompressed color image database," in *Storage and Retrieval Methods and Applications for Multimedia*, vol. 5307. International Society for Optics and Photonics, 2003, pp. 472–481.
- [17] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *IEEE ICCV*, 2001, pp. 416–423.