Optimized Pre-Compensating Compression

Yehuda Dar, Michael Elad, *Fellow, IEEE*, and Alfred M. Bruckstein

Abstract—In imaging systems, following acquisition, an image/video is transmitted or stored and eventually presented to human observers using different and often imperfect display devices. While the resulting quality of the output image may severely be affected by the display, this degradation is usually ignored in the preceding compression. In this paper, we model the sub-optimality of the display device as a known degradation operator applied on the decompressed image/video. We assume the use of a standard compression path, and augment it with a suitable pre-processing procedure, providing a compressed signal intended to compensate the degradation without any post-filtering. Our approach originates from an intricate rate-distortion problem, optimizing the modifications to the input image/video for reaching best end-to-end performance. We address this seemingly computationally intractable problem using the alternating direction method of multipliers approach, leading to a procedure in which a standard compression technique is iteratively applied. We demonstrate the proposed method for adjusting HEVC image/video compression to compensate post-decompression visual effects due to a common type of displays. Particularly, we use our method to reduce motion-blur perceived while viewing video on LCD devices. The experiments establish our method as a leading approach for preprocessing high bit-rate compression to counterbalance a post-decompression degradation.

Index Terms—Rate-distortion optimization, signal degradation, motion blur reduction, alternating direction method of multipliers (ADMM).

I. INTRODUCTION

IMAGE and video signals have a significant, constantly growing, role in many contemporary applications. A fundamental need of image/video applications is to store and/or transmit a digital version of the signal, obeying a bit-budget constraint stemming from the available storage space or the communication channel bandwidth. This bit-budget limitation is managed by lossy compression that produces a compressed representation satisfying the bit-cost constraint at the expense of some distortion in the decompressed signal. This systematic flow (see Fig. 1) usually ends with a human user watching the image/video on a display device. Accordingly, the quality of

the viewed signal is determined by the compression, the imperfections of the display device, and the human visual system.

Lossy image and video compression methods trade-off the compressed-form bit-rate with distortion of the decompressed signal. Popular compression techniques (e.g., JPEG [1], JPEG2000 [2], HEVC [3]) substantially differ in their rate-distortion optimization strategies and the employed image/video models. However, these standard designs ignore other procedures possibly accompanying the compression, thus, may result in sub-optimal rate-distortion performance when considering the complete system.

In this work, we study an intriguing extension of the regular compression problem, where the decompressed signal is degraded by a known linear operator (see Fig. 2). Our goal is to compress by considering the squared error between the degraded decompression and the input image. The corresponding rate-distortion optimization has a challenging structure due to the degradation operator involved in the distortion term. We tackle the intricate optimization using the alternating direction method of multipliers (ADMM) approach [4], mapping the task to a sequence of easier problems including regular rate-distortion optimizations that are replaced with repeated applications of a standard compression technique. Remarkably, our iterative procedure generically adapts a regular compression method to consider the extended settings involving a post-decompression degradation.

Our approach presented here is a paradigm for managing complicated rate-distortion optimizations associated with sophisticated compression frameworks. Specifically, we recently studied the topic of complexity-regularized restoration of an image from its deteriorated version [5], [6], where the task is an intricate rate-distortion optimization that highly resembles the optimization structure exhibited in this paper for compression purposes. Indeed, we address both problems via iterative optimization procedures, emerging from variable splitting, promoting usage of standard compression techniques. In addition, one may assess the vast potential of the proposed compression paradigm for various applications by contemplating the wide use of ADMM [7]–[12] and the similar Half Quadratic Splitting method [5], [13], [14] for a diversity of challenging signal restoration tasks.

The first part of our experiments considers the adjustment of HEVC image compression to a blur operator degrading the decompressed image. Our results demonstrate the effectiveness of the proposed approach, having superior rate-distortion performance compared to a regular HEVC compression. Another alternative to accommodate post-decompression degradation is by preceding the compression with a regular deblurring of the input, using the EPLL method [14].
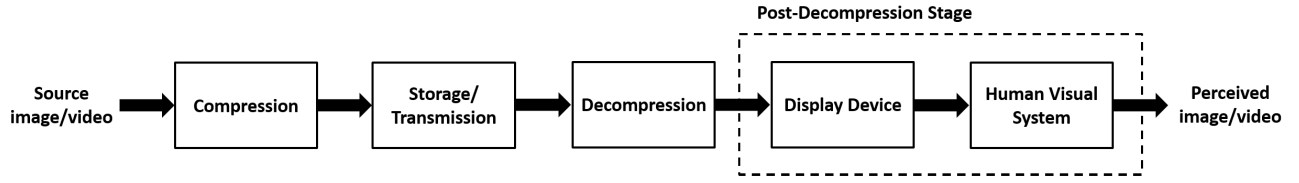
Fig. 1. The considered flow of an image/video that is first compressed and finally perceived by a human observer.
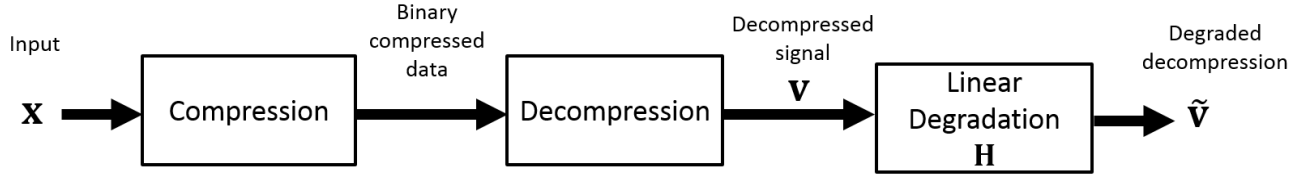

Fig. 2. Demonstration of the conceptual problem settings for compression that is oriented to post-decompression degradation.

Our method outperforms the EPLL-based approach at high bit-rate compression, reaching impressive average-PSNR (i.e., BD-PSNR [15]) gains of 2-3 dB.

As an important application of these ideas, we present a methodology for pre-compression treatment of motion-blur occurring while viewing videos on Liquid Crystal Displays (LCD). The prevalent technology of LCD devices relies on a hold-type mechanism, where each frame is constantly displayed until its replacement, resulting in delicate discontinuities of motions. The human eye tracks an object based on its smooth motion, trying to fix its location on the retina for a vivid perception. The smooth eye tracking of discontinuous motion displayed on LCD yields an unsteady positioning on the retina, causing a blurred perception of the moving object. This blur artifact is amplified for more rapid motions and/or when the video or the display frame-rates are inadequately low, implying too long constant-frame display duration. Importantly, motion blur due to the hold-type nature is still an issue of great interest in contemporary evaluations of LCD screens (for examples, see the technical reviews in [16] and the excellent experimental demonstrations therein) and considered as a crucial drawback of ultra high-definition displays [17].

Straightforward amendments for LCD motion-blur reduce the constant-frame display duration by black-frame insertion [18] that causes unwanted eye strains, or by interpolation-based frame-rate up conversion [19]–[21] that is computationally intensive and unsuited for complicated motion types. More sophisticated techniques [22]–[24] counteract the LCD motion blur by a pre-display frame filtering, designed based on blur models of the LCD hold-type behavior and the eye-tracking capability of the human visual system. These works achieved high PSNR gains using inverse filtering, [22], and the Lucy-Richardson deconvolution method [23], however, introduced subjectively annoying noise artifacts that were attenuated in [24] using spatio-temporal smoothness regularization.

While our application for LCD motion-blur reduction relates to the line of works [22]–[24], we are the first to address the problem via a pre-compression procedure suggesting computational and accuracy benefits. First, many video content types (e.g., entertainment) are compressed in offline settings rich in computation and time resources, contrasting the regular processing [22]–[24] intended for the display device. Accordingly, one can utilize our method in a video-on-demand system designed such that the display types are known and the suitable videos can be delivered to the users. Second, the blur-compensating filters make use of the current video-motion imperfectly estimated on the available data. While "on-device" methods should practically operate on decompressed frames leading to increased motion-estimation errors (especially at medium/low qualities), our approach uses the pre-compression frames for better motion estimation providing more accurate blur characterization and filtering. Nicely, our display-blur compensation is, in fact, constrained by the associated video coding procedure acting as a spatio-temporal complexity regularizer preferring smoother or other model-conforming signals costing less bits (see, e.g., in [5] and [6]). Consequently, our motion-blur reduction technique provides impressive PSNR gains (with respect to the compression bit-rates) and a pleasing subjective quality.

This paper is organized as follows. In Section II we present our method in its general form. In Section III the proposed approach is experimentally studied for adjusting HEVC image compression to balance a post-decompression blur. In Section IV we employ our method for adapting HEVC video coding to reduce motion blur occurring later on the LCD display. Section V concludes this paper.

II. THE PROPOSED METHOD

A. The Basic Rate-Distortion Optimization

We develop our method based on the system structure illustrated in Fig. 2 and explained next. First, an N-dimensional input signal, $\mathbf{x} \in \mathbb{R}^N$, goes through a lossy compression procedure resulting in a compressed binary description associated with an approximation of $\mathbf{x}$, denoted as $\mathbf{v} \in \mathbb{R}^N$, obtained after the decompression stage. However, the reconstruction $\mathbf{v}$ is further deteriorated, for instance, due to a sub-optimal display device. We consider here a linear deterioration operator, represented by the $N \times N$ real-valued matrix $\mathbf{H}$. Then, the degraded decompressed signal is defined as

$$\tilde{\mathbf{v}} \triangleq \mathbf{H}\mathbf{v}, \qquad (1)$$

the outcome of the entire process.

Our goal here is to optimize the compression procedure with respect to the squared error between the input signal $\mathbf{x}$ and the degraded decompression $\tilde{\mathbf{v}}$, that using (1) can be expressed as

$$D_H\left(\mathbf{x},\mathbf{v}\right) \triangleq \|\mathbf{x}-\mathbf{H}\mathbf{v}\|_2^2. \tag{2}$$

Without loss of generality (as will be explained later), we develop our method with respect to a block-based compression design individually operating on blocks of N_b samples defined by a non-overlapping segmentation of the signal. We refer to members belonging to the grid of non-overlapping blocks via the set of indices $\mathcal{B}$. The block-level compression procedure is modeled as a vector quantizer having a codebook $\mathcal{C}$, being a finite set of block-reconstruction candidates and their respective variable-length binary codewords. Specifically, the block-reconstruction $\mathbf{c} \in \mathcal{C}$ has a corresponding binary codeword of length $r\left(\mathbf{c}\right)$ defining the respective block bit-cost. Accordingly, the total bit-cost can be evaluated from the decompressed blocks $\{\mathbf{v}_i\}_{i\in\mathcal{B}}$ as the sum $\sum_{i\in\mathcal{B}} r\left(\mathbf{v}_i\right)$. We define the matrix $\mathbf{P}_i$ as a linear operator extracting the ith block from the complete signal by the standard multiplication $\mathbf{P}_i\mathbf{v}=\mathbf{v}_i$. Then, the bit-cost of the entire signal can be expressed as

$$R\left(\mathbf{v}\right) = \sum_{i\in\mathcal{B}} r\left(\mathbf{P}_i\mathbf{v}\right). \tag{3}$$

We use the quantities defined in (2)-(3) to formulate the rate-distortion optimization in the unconstrained Lagrangian form:

$$\hat{\mathbf{v}} = \underset{\mathbf{v}\in\mathcal{C}_{\mathcal{B}}}{\text{argmin}} \ \ \|\mathbf{x}-\mathbf{H}\mathbf{v}\|_2^2 + \lambda \sum_{i\in\mathcal{B}} r\left(\mathbf{P}_i\mathbf{v}\right) \tag{4}$$

where $\lambda \geq 0$ is the Lagrange multiplier associated with some total bit-cost constraint, and $\hat{\mathbf{v}}$ is the optimal decompressed signal among the candidates available in the effective full-signal codebook:

$$\mathcal{C}_{\mathcal{B}} = \left\{ \mathbf{c} \ \Big| \ \mathbf{c} = \sum_{i\in\mathcal{B}} \mathbf{P}_i^T\mathbf{c}_i, \{\mathbf{c}_i\}_{i\in\mathcal{B}} \in \mathcal{C} \right\} \tag{5}$$

where the linear operator $\mathbf{P}_i^T$ places a block in the ith block location in a full-signal layout. One should note that, for an arbitrarily structured $\mathbf{H}$, the optimization (4) is difficult to solve since it does not allow the commonly used block-based treatment (for examples, see its various forms in the fundamental studies on operational rate-distortion optimization [25]–[27] and also in recent works [28], [29]).

B. Practical Iterative Procedure

The structural complication of the rate-distortion optimization (4) is facilitated using the ADMM strategy [4] as explained next. Initially, we define the auxiliary variable $\mathbf{z} \in \mathbb{R}^N$ letting us to reformulate the problem (4) into

$$\left(\hat{\mathbf{v}},\hat{\mathbf{z}}\right) = \underset{\mathbf{v}\in\mathcal{C}_{\mathcal{B}},\ \mathbf{z}\in\mathbb{R}^N}{\text{argmin}} \ \|\mathbf{x}-\mathbf{H}\mathbf{z}\|_2^2 + \lambda \sum_{i\in\mathcal{B}} r\left(\mathbf{P}_i\mathbf{v}\right) \tag{6}$$

$$\text{s.t. } \mathbf{z}=\mathbf{v}. \tag{7}$$

Then, considering (6) via its augmented Lagrangian (in its scaled version [4, Ch. 2]) leads to an iterative procedure, where the tth iteration is

$$\left(\hat{\mathbf{v}}^{(t)},\hat{\mathbf{z}}^{(t)}\right) = \underset{\mathbf{v}\in\mathcal{C}_{\mathcal{B}},\mathbf{z}\in\mathbb{R}^N}{\text{argmin}} \ \|\mathbf{x}-\mathbf{H}\mathbf{z}\|_2^2$$
$$+ \lambda \sum_{i\in\mathcal{B}} r\left(\mathbf{P}_i\mathbf{v}\right) + \frac{\beta}{2}\left\|\mathbf{v}-\mathbf{z}+\mathbf{u}^{(t)}\right\|_2^2 \tag{8}$$

$$\mathbf{u}^{(t+1)} = \mathbf{u}^{(t)} + \left(\hat{\mathbf{v}}^{(t)}-\hat{\mathbf{z}}^{(t)}\right), \tag{9}$$

where $\mathbf{u}^{(t)} \in \mathbb{R}^N$ is the scaled dual variable and β is an auxiliary parameter originating at the Lagrangian.

Since each of the optimization variables in (8) participates only in two of the three terms in the cost function and, therefore, one iteration of alternating minimization provides us the ADMM form that iterates over the following manageable optimizations:

$$\hat{\mathbf{v}}^{(t)} = \underset{\mathbf{v}\in\mathcal{C}_{\mathcal{B}}}{\text{argmin}} \ \frac{\beta}{2}\left\|\tilde{\mathbf{z}}^{(t)}-\mathbf{v}\right\|_2^2 + \lambda \sum_{i\in\mathcal{B}} r\left(\mathbf{P}_i\mathbf{v}\right) \tag{10}$$

$$\hat{\mathbf{z}}^{(t)} = \underset{\mathbf{z}\in\mathbb{R}^N}{\text{argmin}} \ \|\mathbf{x}-\mathbf{H}\mathbf{z}\|_2^2 + \frac{\beta}{2}\left\|\mathbf{z}-\tilde{\mathbf{v}}^{(t)}\right\|_2^2 \tag{11}$$

$$\mathbf{u}^{(t+1)} = \mathbf{u}^{(t)} + \left(\hat{\mathbf{v}}^{(t)}-\hat{\mathbf{z}}^{(t)}\right). \tag{12}$$

where $\tilde{\mathbf{z}}^{(t)} = \hat{\mathbf{z}}^{(t-1)} - \mathbf{u}^{(t)}$ and $\tilde{\mathbf{v}}^{(t)} = \hat{\mathbf{v}}^{(t)} + \mathbf{u}^{(t)}$. The analytic solution of the second-stage problem in (11) is

$$\hat{\mathbf{z}}^{(t)} = \left(\mathbf{H}^T\mathbf{H} + \frac{\beta}{2}\mathbf{I}\right)^{-1}\left(\mathbf{H}^T\mathbf{x}+\frac{\beta}{2}\tilde{\mathbf{v}}^{(t)}\right), \tag{13}$$

thus, exhibiting optimization (11) as a weighted averaging operation.

Importantly, the first stage (10) is a rate-distortion optimization compatible with a block-based treatment and considering the regular squared-error metric for the compression of $\tilde{\mathbf{z}}^{(t)}$, obtained in the second stage of the former iteration. Moreover, this full-image rate-distortion optimization is done for a Lagrange multiplier of value $\tilde{\lambda} = \frac{2\lambda}{\beta}$. We denote the compression-decompression procedure associated with (10) as

$$\hat{\mathbf{v}}^{(t)} = CompressDecompress_\lambda\left(\tilde{\mathbf{z}}^{(t)}\right). \tag{14}$$

We further suggest using a standard compression method as the compression-decompression operator (14). While many compression methods do not follow the exact rate-distortion optimization we got in our mathematical development (10), we still suggest using such techniques as replacements for (10). Additionally, since various compression methods do not rely on Lagrangian optimization, their operating parameters may differ (for example, quality parameters, compression ratios, or output bit-rates). Accordingly, we present the suggested algorithm with respect to a general compression procedure that its output bit-cost is directly or indirectly affected by a parameter denoted as θ. This generalization is used in Algorithm 1.

The replacement of the rate-distortion optimization formulation in (10) with a standard compression-decompression process (14) is motivated by a similar development step used

Algorithm 1 Proposed Method: Compression Adjusted to Post-Decompression Degradation

1: Inputs: $\mathbf{x}$, θ, β.
2: Initialize $\hat{\mathbf{z}}^{(0)} = \mathbf{x}$, $\mathbf{u}^{(1)} = \mathbf{0}$.
3: $t = 1$
4: **repeat**
5:     $\tilde{\mathbf{z}}^{(t)} = \hat{\mathbf{z}}^{(t-1)} - \mathbf{u}^{(t)}$
6:     $\hat{\mathbf{v}}^{(t)} = CompressDecompress_\theta \left(\tilde{\mathbf{z}}^{(t)} \right)$.
7:     $\tilde{\mathbf{v}}^{(t)} = \hat{\mathbf{v}}^{(t)} + \mathbf{u}^{(t)}$
8:     $\hat{\mathbf{z}}^{(t)} = \left(\mathbf{H}^T \mathbf{H} + \frac{\beta}{2} \mathbf{I} \right)^{-1} \left(\mathbf{H}^T \mathbf{x} + \frac{\beta}{2} \tilde{\mathbf{v}}^{(t)} \right)$
9:     $\mathbf{u}^{(t+1)} = \mathbf{u}^{(t)} + \left(\hat{\mathbf{v}}^{(t)} - \hat{\mathbf{z}}^{(t)} \right)$
10:    $t \leftarrow t + 1$
11: **until** stopping criterion is satisfied
12: Output: Binary compressed data obtained in the last application of Stage 6.
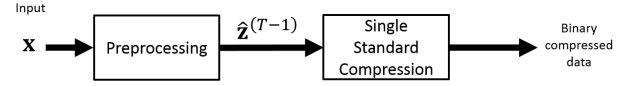

Fig. 3. Interpretation of the proposed compression method as a preprocessing stage followed by a single standard compression. The demonstration here assumes that our procedure runs T iterations.

in the Plug-and-Play Priors method [8] for image restoration, where an optimization stage corresponding to a Maximum A-Posteriori (MAP) Gaussian denoising problem is replaced with the application of an existing denoiser (such as BM3D [30]). In both cases (ours and in [8]), the application of an arbitrary compression/denoising method means that the convexity of the optimization problem cannot be guaranteed and, therefore, in some cases the optimization may not converge. Accordingly, the implementations we present in Sections III-IV include a divergence detection mechanism as part of the stopping criterion, this feature is explained later in this paper. Studying the convergence/divergence properties of our framework (in the spirit of the analysis in [31]) is left to a future work.

We can further interpret the proposed iterative compression approach as a preprocessing stage coupled with a single standard compression, being the one applied in the last iteration as the determining stage outputting the compressed binary data (see Fig. 3). Remarkably, our compression output is compatible with a standard decompression process. The overall quality improvement suggested by our method obviously entails an increased computational cost that, nevertheless, is distributed between the encoder and the decoder stages in the following attractive structure: the decoder complexity remains as in the standard form, while the encoder has the increased computational load of repeatedly applying standard compressions (14) and the ℓ_2-constrained deconvolutions (13). This system layout is beneficial for applications where the compression can be carried out offline in environments rich in computational and time resources, whereas the decompression on the display devices should be of a low computational cost due to run-time and energy-consumption limitations.

III. ADJUSTING HEVC IMAGE COMPRESSION TO BLURRY DECOMPRESSION

In this section we demonstrate our approach for adapting HEVC's still-image compression (in its version implemented in the BPG image format [32]) to a blur deteriorating the

decompressed image. The experiment goal here is to study our method with respect to alternative processing strategies. Our ideal settings here, considering a known Gaussian blur kernel, serve as a preliminary stage to the intricate application presented in Section IV. The degradation operator $\mathbf{H}$ is associated with a Gaussian blur kernel of standard deviation 0.6 and 15×15 pixels size. We consider a shift-invariant degradation, thus, efficiently degrade an image using a two-dimensional convolution with the blur kernel.

The Peak Signal-to-Noise Ratio (PSNR) is defined here based on the squared-error distortion (2) and can be written as

$$PSNR = 10 \log_{10} \left(\frac{P^2}{\frac{1}{N} \|\mathbf{x} - \tilde{\mathbf{v}}\|_2^2} \right) \qquad (15)$$

where $\mathbf{x}$ and $\tilde{\mathbf{v}}$ are the input and the degraded decompressed signals, respectively, and N is the signal dimension. The maximal value attainable by the examined signals is denoted as P that, e.g., equals to 255 for grayscale images with pixel values in the range $[0, 255]$. In the PSNR computation we ignore margins of 35 pixels along the borders of the image to exclude effects of specific boundary conditions used in the applied convolutions.

A. Competing Methods

We compare our approach to three competing strategies also considering HEVC image compression. In Figure 4 we compare the rate-distortion curves of the various methods corresponding to operating their HEVC component using quantization parameter (QP) values between 1 to 49 in jumps of 3. The examined compression procedures are:

1) Regular Compression Without Any Pre/Post Processing: This is the baseline approach where a regular compression-decompression application is followed by deterioration. Obviously, since this procedure ignores the degradation, it is the cheapest in computations and provides an inferior performance (see the solid-line black curves in Fig. 4).

2) Pre-Compression Pseudoinverse Filtering of the Input Image: We consider here an ideal pseudoinverse filter, matched to the known degradation operator $\mathbf{H}$, employed as a pre-compression filter. The numerical crudity of the pseudoinverse filter yields a very large dynamic range of pixel values, hence, requiring shifting and scaling before the compression and the inverse adaptations after decompression (before degradation). Since the pseudoinverse filtered image is far from obeying natural-image characteristics (e.g., smoothness), it is inefficiently compressed by a standard compression technique. This drawback results in performance inferior even to regular compression without any processing (see Fig. 4).
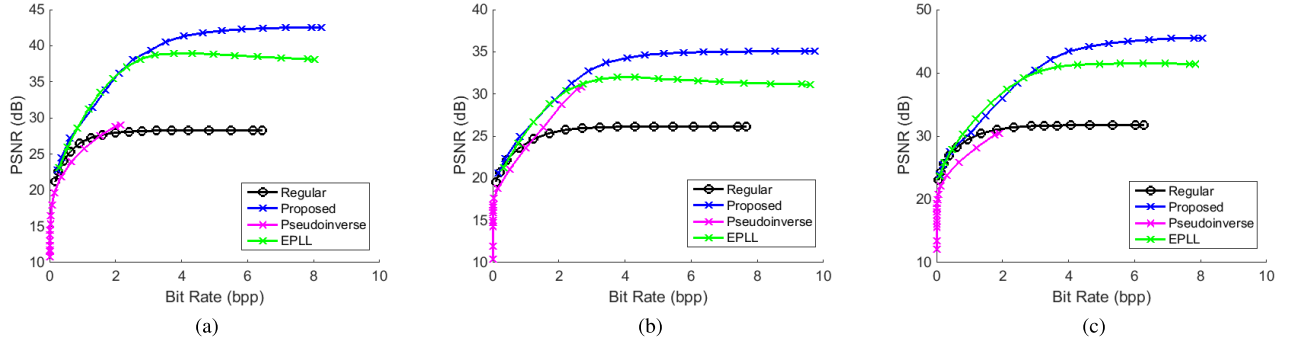
Fig. 4. PSNR-bitrate curves comparing our approach to competing methods for three grayscale images (see also Table I). The post-decompression deterioration is a 15×15 Gaussian blur kernel (standard deviation 0.6). (a) Cards. (b) Tree. (c) Bears.
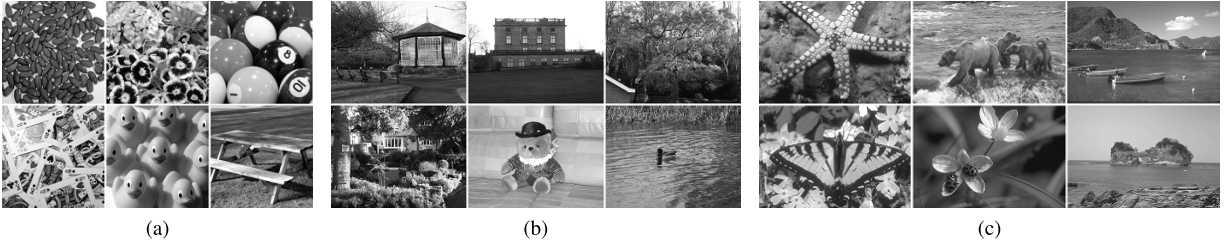

Fig. 5. The images used in the experiments. (a) TESTIMAGES. (b) UCID. (c) Berkeley.

Moreover, the unusual signals provided by the pseudoinverse filter can be compressed using HEVC to a limited range of bit-rates, for examples, observe the rightmost working-points of the magenta curves obtained using HEVC compression at the very high quality corresponding to $QP = 1$. This exemplary approach exhibits the challenges in pre-compression processing.

3) Pre-Compression Filtering via the Expected Patch Log Likelihood (EPLL) Method: We define the main competing method to employ a pre-compression filtering in the form of the EPLL deblurring method relying on a Gaussian Mixture Model (GMM) prior learned for natural images (see [14]). Indeed, the processed image conforms with natural-image attributes, thus, efficiently compressed by HEVC leading to a good rate-distortion performance considering the degraded decompressed image (see the solid-line green curves in Fig. 4). In the EPLL experiments we used the implementation published by Zoran and Weiss [14] with parameters we found to improve the rate-distortion performance considered here.

B. Our Method: Experiment Settings

We now turn to evaluate our method with respect to the above three reference techniques. In the implementation of Algorithm 1 we set β to a value depending on the specific quantization parameter (QP) given to the HEVC compression:

$$\beta = \begin{cases} 0.03, & \text{for } 0 \leq QP \leq 20 \\ 0.05, & \text{for } 20 < QP \leq 30 \\ 0.10, & \text{for } 30 < QP \leq 40 \\ 0.35, & \text{for } 40 < QP \leq 45 \\ 0.45, & \text{for } 45 < QP \leq 51 \end{cases} \quad (16)$$

Recall that HEVC QP values are integers between 0 to 51, where a lower value yields a higher quality. The stopping

criterion was defined to a maximal number of 40 iterations or to end earlier when $\hat{\mathbf{v}}^{(t)}$ and $\hat{\mathbf{z}}^{(t)}$ are detected to converge or diverge. The convergence/divergence detection relies on the total absolute difference between $\hat{\mathbf{v}}^{(t)}$ and $\hat{\mathbf{z}}^{(t)}$ in each iteration, namely, $w^{(t)} \triangleq \left\| \hat{\mathbf{v}}^{(t)} - \hat{\mathbf{z}}^{(t)} \right\|_1$. Accordingly, we determine convergence when $|w^{(t)} - w^{(t-1)}| < 0.2$ for three consecutive iterations. Divergence is identified when $w^{(t)} - w^{(t-1)} > 50$ and, in that case, the algorithm output is taken from the preceding iteration. Note that the threshold values given here for convergence/divergence detection depend on the signal dimension and the typical value-range (the thresholds specified here are for signal with values in the range [0,1]).

The rate-distortion curves of our method, presented in Fig. 4 as the blue solid-lines, outperform the other pre-compression techniques at the high bit-rate range. The PSNR gains at high bit-rates are significant (note the wide PSNR range of the graphs that may visually mislead), reaching improvements of several dBs. These impressive PSNR gains at high bit-rates were further established by examining 18 images (see Fig. 5 and Table I) collected from three different datasets (TESTIMAGES [33], UCID [34], and Berkeley [35]). The comparison in Table I considers the average PSNR difference between performance curves (e.g., see Fig. 4) of the proposed, the EPLL-based, and the regular methods. The average PSNR differences between curves were calculated using the BD-PSNR metric [15], [36], for the entire bit-rate range (i.e., the complete curves generated for QP values between 1 to 49 in jumps of 3) and for curve segments corresponding to high bit-rates (defined by QP values 1, 7, 13 and 19).

In Figure 6 we present visual results for the 'Starfish' image. First, we examine the regular compression procedure where the input image (Fig. 6a) is compressed using HEVC at a bit-rate of 5.061 bits per pixel (bpp), leading to the decompressed

TABLE I

IMAGE COMPRESSION CONSIDERING POST-DECOMPRESSION DETERIORATION OF A GAUSSIAN BLUR
AVERAGE PSNR GAINS (MEASURED USING BD-PSNR)

| Dataset	Image	All Bit-Rates		High Bit-Rates	
		Proposed over Regular	Proposed over EPLL	Proposed over Regular	Proposed over EPLL
TESTIMAGES 300x300	Almonds	5.49	0.22	14.59	3.22
	Flowers	5.57	0.41	14.96	3.21
	Billiard balls	5.57	-0.08	12.60	2.00
	Cards	6.08	0.90	13.55	3.67
	Duck toys	3.62	-0.62	10.42	1.08
	Garden table	3.78	0.29	12.61	2.69
UCID 384x512	Garden house	3.21	1.14	7.48	3.39
	House and lawn	2.41	0.50	6.41	2.97
	Tree	3.57	1.16	8.79	3.64
	Garden	3.16	1.49	6.55	3.32
	Teddy bear	3.65	-0.39	10.99	2.37
	Duck	5.25	0.06	15.64	2.62
Berkeley 481x321	Starfish	4.72	-0.36	15.42	2.02
	Bears	3.78	0.42	12.69	3.46
	Boats	3.29	0.43	8.94	2.65
	Butterfly	4.76	0.42	11.80	3.33
	Flower and Bugs	4.18	0.02	11.00	2.39
	Sea	4.58	0.21	12.87	2.49
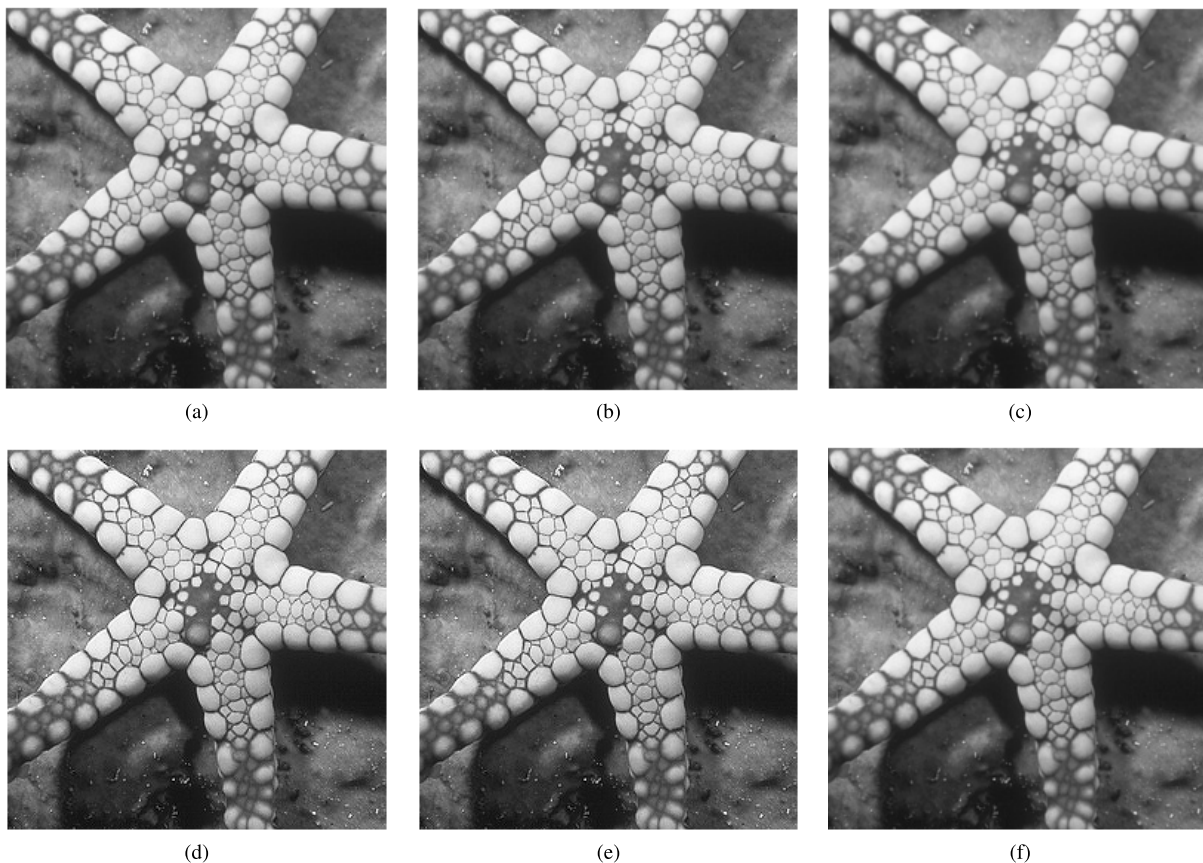
Fig. 6. Demonstrating the intermediate and resultant images of the regular and the proposed compression methods for a post-decompression deterioration of a Gaussian blur kernel. The image is 'Starfish' (a segment of 256×256 pixels is shown). (a) Input. (b) Regular Decompression (5.061 bpp). (c) Regular Degraded Decompression (34.33 dB). (d) Our: Input to Last Iteration Compression. (e) Our: Decompression (4.296 bpp). (f) Our: Degraded Decompression (49.58 dB).

image in Fig. 6b (note that this is the pre-degradation image). Then, obviously, the post-degradation decompressed image (Fig. 6c) suffers from tremendous blur affecting also the PSNR (measured with respect to the precompression image).

In contrast, our approach processes the input image such that the compression in the last iteration gets a sharpened version (see Fig. 6d) adjusted to the specifically known blur operator, then, the compressed image at bit-rate 4.296 bpp leads to a

degraded decompression with moderate blur effects (Fig. 6f) and PSNR improvement of 15.25 dB with respect to the regular compression at a higher bit-rate.

The experiments show that the current implementation of our approach is significantly better than the considered alternatives for compression at the high bit-rate range. We consider this behavior to emerge from the following two facts. First, the details of the pre-compression processing are preserved better when high bit-rate compression is applied. Second, at high bit-rate compression the employed quantization is finer, thus, the discrete optimization problem imitates more closely an optimization over a continuous domain which is more suitable for the ADMM optimization technique.

IV. APPLICATION TO LCD MOTION-BLUR REDUCTION

A. The LCD Motion Blur and Its Modeling

A prominent type of post-decompression degradations is defined as the inevitable artifacts arising due to various display device technologies. For instance, the formerly prevalent Cathode Ray Tube (CRT) displays employ an impulse-type mechanism where video frames are instantaneously presented, producing a good perceptual motion-continuity and unpleasing flickering artifacts. Here we focus on the current Liquid Crystal Display (LCD) technology, the ultimate successor of CRT, being a hold-type display where each frame is constantly presented for a duration of $(frame\ rate)^{-1}$ seconds, referred to as the hold time (e.g., for a rate of 60 frames per second the hold time is 16.6 milliseconds). While this hold-type architecture is flickering free, it suffers from a non-smooth presentation of motions that cause blur in the image perceived by the viewer. Specifically, the human eye pursues constant motion of an object to fix its image location on the retina for a detailed perception. While motion presented on an LCD device has delicate discontinuities, the eye still tracks it as if it was continuous and, thus, suffers from corresponding spatial displacements on the retina that blur the perceived image.

Additional LCD blur stems from the response time, which is the duration taking a pixel to change its intensity, that despite its reduction along the years still introducing some amount of blur (see, e.g., [16]). As in [22]–[24], we consider motion blur arising only from the hold-type method of the LCD. The reader is referred to [17], [22]–[24], and [37] for additional discussions on the above described CRT and LCD motion artifacts. In this section we will rely on existing models and problem settings addressing the LCD motion blur, and utilize our method from Section II for adjusting HEVC video coding to pre-compensate the perceived motion blur.

The two prominent signal-processing models of LCD motion blur were developed in the frequency [22] and signal [37] domains, considering the display impulse response and the human visual system (HVS) mechanisms of motion tracking and spatio-temporal low-pass filtering. A later model [24] interpreted the former analyses to the case of discrete video signals, and approximated the temporal blur operator as an intra-frame spatial degradation determined by the current motion. This spatio-temporal equivalence of motion-blur degradation due to the hold-type nature of the

---

Algorithm 2 Proposed Method: Video Coding Adjusted to Compensate LCD Motion Blur

---

1: Inputs: $\mathbf{x}$, θ, β.
2: Set frame blur operators $\{\mathbf{H}_i\}_{i=1}^T$ based on motion estimation.
3: Initialize $\hat{\mathbf{z}}^{(0)} = \mathbf{x}$, $\mathbf{u}^{(1)} = \mathbf{0}$, $t = 1$.
4: **repeat**
5:     $\tilde{\mathbf{z}}^{(t)} = \hat{\mathbf{z}}^{(t-1)} - \mathbf{u}^{(t)}$
6:     $\hat{\mathbf{v}}^{(t)} = CompressDecompress_\theta \left(\tilde{\mathbf{z}}^{(t)} \right)$.
7:     $\tilde{\mathbf{v}}^{(t)} = \hat{\mathbf{v}}^{(t)} + \mathbf{u}^{(t)}$
8:     Form $\hat{\mathbf{z}}^{(t)}$ via frame-level solutions for $i = 1, ..., T$:
       $\hat{\mathbf{z}}_i^{(t)} = \left(\mathbf{H}_i^T \mathbf{H}_i + \frac{\beta}{2}\mathbf{I} \right)^{-1} \left(\mathbf{H}_i^T \mathbf{x}_i + \frac{\beta}{2}\tilde{\mathbf{v}}_i^{(t)} \right)$
9:     $\mathbf{u}^{(t+1)} = \mathbf{u}^{(t)} + \left(\hat{\mathbf{v}}^{(t)} - \hat{\mathbf{z}}^{(t)} \right)$
10:     $t \leftarrow t + 1$
11: **until** stopping criterion is satisfied
12: Output: Binary compressed data obtained in the last application of Stage 6.

LCD was used in various forms in [38]–[41]. We here follow the model for LCD motion-blur given in [24] to be aligned with the problem settings defined therein.

The kth frame of the displayed video is a $W \times H$ two-dimensional discrete signal, comprised of $N_f = W \cdot H$ pixels that their column-stack form is denoted here as $\mathbf{v}_k \in \mathbb{R}^{N_f}$. The perceived image corresponding to the kth frame is

$$\tilde{\mathbf{v}}_k = \mathbf{H}_k \mathbf{v}_k \tag{17}$$

where $\mathbf{H}_k$ is a $N_f \times N_f$ matrix representing the motion-blur as a spatial operator. The rth row of $\mathbf{H}_k$ specifies the blur operation producing the rth pixel of the degraded frame. The local blur operation is determined by the associated motion vector that may vary for different pixels. For example, assume the rth pixel corresponds to the motion vector $(0, -3)$ describing a vertical motion upwards in 3 pixels with respect to the previous frame, accordingly, (assuming r corresponds to a coordinate sufficiently distant from the frame boundaries in its 2D arrangement) the rth row of $\mathbf{H}_k$ should be formed from the following entries

$$\mathbf{H}_k \left[r, c \right] = \begin{cases} \frac{1}{3}, & \text{for } c = r, \ r - 1, \ r - 2 \\ 0, & \text{otherwise.} \end{cases} \tag{18}$$

A detailed numerical method for defining the blur kernel given a motion vector was described in [24]. An important particular case occurs when the frame motion is global, leading to a block-circulant matrix $\mathbf{H}_k$.

B. The Proposed Method for Motion-Blur Reduction

We now turn to translate our general method given in Section II to the specific degradation of LCD motion blur. The considered video signal is a sequence of T frames, each of $W \times H$ pixels. The column-stack form of the video is denoted as $\mathbf{x} \in \mathbb{R}^N$ where $N = T \cdot W \cdot H$ is the total amount of pixels. The signal $\mathbf{x}$ is comprised from a concatenation of the
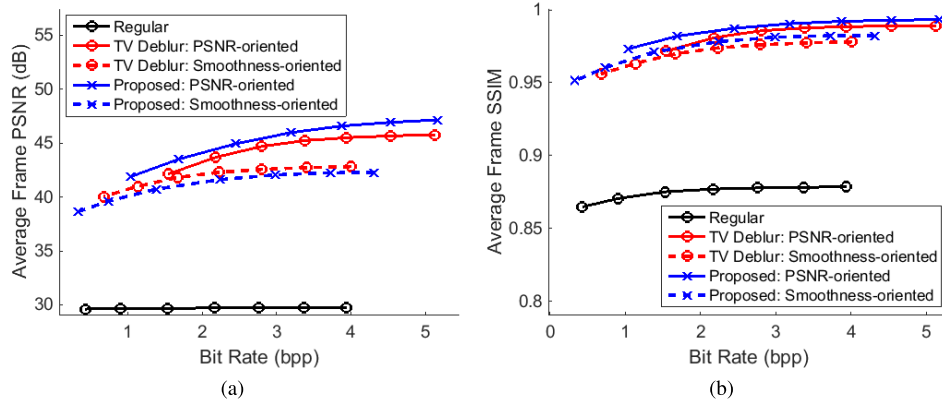
Fig. 7. LCD motion blur experiment for the 'Shields' sequence. The performance of our method in its PSNR-oriented (solid blue line) and smoothness-oriented (dashed blue line) modes is compared to preprocessing using total-variation deblurring [45] in its PSNR-oriented (solid red line) and smoothness-oriented (dashed red line) parameter settings, and to a regular compression-decompression procedure without any additional processing (solid black line). The average frame-PSNR and average frame-SSIM are evaluated in (a) and (b), respectively.
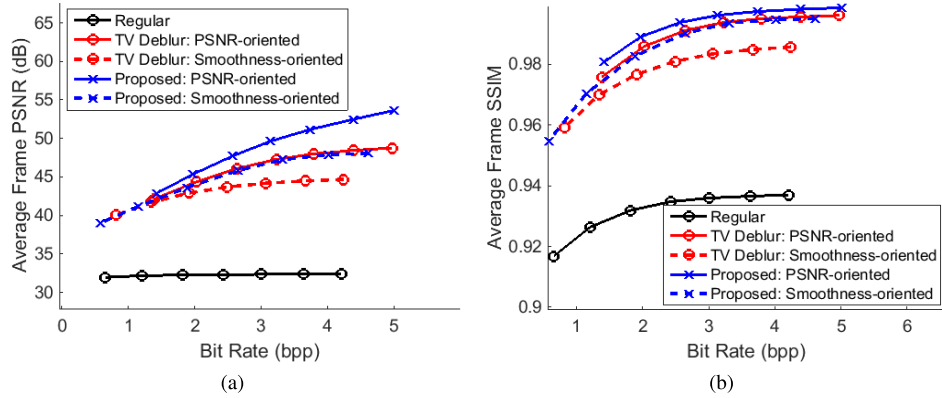

Fig. 8. LCD motion blur experiment for the 'Stockholm' sequence. The performance of our method in its PSNR-oriented (solid blue line) and smoothness-oriented (dashed blue line) modes is compared to preprocessing using total-variation deblurring [45] in its PSNR-oriented (solid red line) and smoothness-oriented (dashed red line) parameter settings, and to a regular compression-decompression procedure without any additional processing (solid black line). The average frame-PSNR and average frame-SSIM are evaluated in (a) and (b), respectively.

T frames, i.e.,

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_T \end{bmatrix} \qquad (19)$$

where $\mathbf{x}_i \in \mathbb{R}^{N_f}$ is the column-stack form of the ith frame, and $N_f = W \cdot H$ is the number of pixels in a frame.

The degradation considered here is modeled as independent spatial operations on each of the frames. Accordingly, the full signal degradation operator is the following block-diagonal matrix:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_2 & \mathbf{0} & \vdots \\ \vdots & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{H}_T \end{bmatrix} \qquad (20)$$

where the $N_f \times N_f$ matrix $\mathbf{H}_i$ is the spatial blur operator of the ith frame.

Then, the block-diagonal structure (20) lets us to decompose the optimization (11), which is an intermediate stage in the ADMM iteration, into the following frame-level optimizations:

$$\hat{\mathbf{z}}_i^{(t)} = \underset{\mathbf{z}_i \in \mathbb{R}^{N_f}}{\operatorname{argmin}} \ \|\mathbf{x}_i - \mathbf{H}_i \mathbf{z}_i\|_2^2 + \frac{\beta}{2} \left\| \mathbf{z}_i - \tilde{\mathbf{v}}_i^{(t)} \right\|_2^2, \quad i = 1, \ldots, T \qquad (21)$$

where $\hat{\mathbf{z}}_i^{(t)}$ and $\tilde{\mathbf{v}}_i^{(t)}$ are the column-vector forms of the ith frames of the video signals $\hat{\mathbf{z}}^{(t)}$ and $\tilde{\mathbf{v}}^{(t)}$, respectively. The analytic solution of the ith frame optimization from (21) is

$$\hat{\mathbf{z}}_i^{(t)} = \left(\mathbf{H}_i^T \mathbf{H}_i + \frac{\beta}{2}\mathbf{I} \right)^{-1} \left(\mathbf{H}_i^T \mathbf{x}_i + \frac{\beta}{2}\tilde{\mathbf{v}}_i^{(t)} \right). \qquad (22)$$

This computationally important update of Algorithm 1 is employed in Algorithm 2 describing the video compression method compensating a post-decompression LCD motion blur.

(a) Original

(b) Regular: Displayed 3.940 bpp

(c) TV Smoothness-Oriented: Displayed 3.411 bpp

(d) TV PSNR-Oriented: Displayed 3.377 bpp

(e) Proposed Smoothness-Oriented: Displayed 3.248 bpp

(f) Proposed PSNR-Oriented: Displayed 3.192 bpp

(g) Regular: Perceived PSNR 30.72 dB SSIM 0.8884

(h) TV Smoothness-Oriented: Perceived PSNR 43.39 dB SSIM 0.9796

(i) TV PSNR-Oriented: Perceived PSNR 45.49 dB SSIM 0.9873

(j) Proposed Smoothness-Oriented: Perceived PSNR 42.99 dB SSIM 0.9820

(k) Proposed PSNR-Oriented: Perceived **PSNR 46.02 dB SSIM 0.9894**

Fig. 9.   LCD motion blur experiment for the sequence 'Shields' (grayscale, 120 frames at 60fps, 480 × 480 pixels) where a global horizontal motion of −3 pixels/frame causes the blur. An exemplary frame segment is presented. (a) the original frame segment. (b)-(f) are the displayed frame segments using each of the five examined methods. (g)-(k) are the simulated perceived frame segments corresponding to each of the displayed frames. The presented PSNR and SSIM evaluations are for the complete perceived frame, and the bit-rates are those measured for the compression of the complete sequence of 120 frames.

C. Experimental Results

We evaluated our method by adjusting the HEVC video coding standard to compensate the perceptual motion-blur caused by LCD devices. As in previous works for LCD motion-blur reduction [23], [24] we considered the 'Shields' and 'Stockholm' sequences (60 frames per second) [42], having

(a) Regular   (b) TV Smoothness-Oriented   (c) TV PSNR-Oriented   (d) Proposed Smoothness-Oriented   (e) Proposed PSNR-Oriented
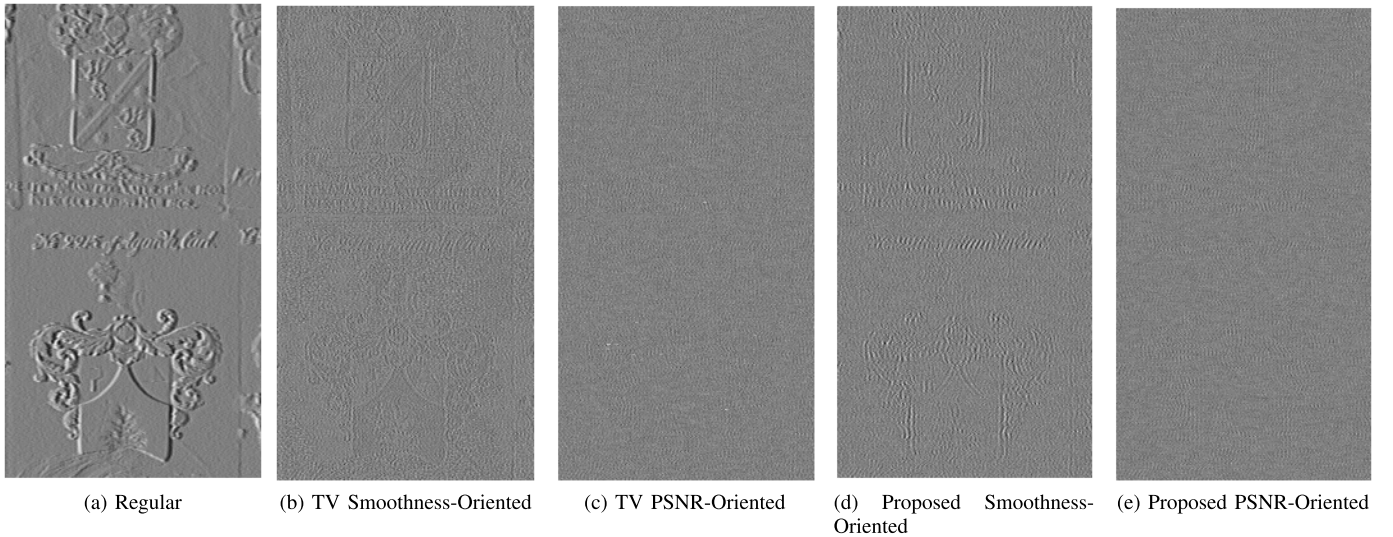
Fig. 10. LCD motion blur experiment for the sequence 'Shields'. Here we present the difference images between the perceived and the original frame segments, for each of the methods presented in Fig. 9: (a) regular, (b) TV smoothness-oriented, (c) TV PSNR-oriented, (d) proposed smoothness-oriented, and (e) proposed PSNR-oriented. The difference images are presented in grayscale by scaling the value range of $[-50, 50]$ for the regular approach, and the value range of $[-15, 15]$ for the other methods.

global horizontal camera motions of -3 pixels/frame and 2 pixels/frame, respectively. The considered video segments were defined as 120 frames of 480×480 pixels taken from the 720p sequences mentioned above.

In the experiments of this section we use the HEVC implementation given in the reference software HM 15.0 [43] set to the 'random access' profile, where powerful motion-compensation procedures together with P and B frame types are employed. The presented comparison (Figs. 7-8) consider HEVC compression operated for QP values between 1 and 19 in jumps of 3. The performance evaluations in Figs. 7a, 8a rely on the average PSNR of the frames. The comparisons in Figs. 7b, 8b consider the value of the SSIM metric [44] averaged over all the frames considered. The basic reference performance is the regular compression where no pre or post processing is done and, consequently, the respective performance is inferior to the other processing-aided methods (see black curves in Figs. 7-8).

The main competing approach is to precede the compression with a video deblurring method addressing the motion blur using spatio-temporal total-variation (TV) regularization [45] (note that the deblurring technique in [45] extends and improves upon the LCD motion-blur reduction method in [24]). Importantly, the methods in [24] and [25] consider the video motion deblurring without any aspect of compression, thus, we employ them in the problem settings considered here as a pre-compression stage. Accordingly, we optimized the parameters of [45] to provide a high average frame-PSNR in our settings (see red solid-line performance curves in Figs. 7-8). In addition, high PSNR is not necessarily coupled with high visual quality, as the perceived video may look noisy and/or flickery (see, e.g., [24]). Therefore, we also define an additional parameter setting of [45] to provide more visually pleasing results at the expense of the PSNR (the performance curves of this smoothness-oriented setting appear as red dashed-line curves in Figs. 7-8. Moreover, we noticed

that the total-variation deblurring method produces artifacts along the vertical borders of the frames (the horizontal borders are artifact free because the global motion is horizontal). Accordingly, we gave the total-variation deblurring method a larger portion of the frame with margins of 100 pixels in each side, then, these margins are removed before given to the compression. This procedure was carried out only for the total-variation deblurring approach.

We evaluated our method in two modes: the first aims to a high PSNR by the setting $\beta = 10\tilde{\beta}$ where $\tilde{\beta}$ is defined by the QP-dependent rules in (16). The second version, referred to as smoothness-oriented, is determined by setting $\beta = 50\tilde{\beta}$ that leads to an increased spatio-temporal smoothness (a visually pleasing property) at the expense of the PSNR. Both of these settings employed a maximal number of 10 iterations, or stopped earlier if convergence or divergence are detected. The rules defining convergence/divergence are as in the image experiment presented in Section III-B, but with the different threshold values of $0.5 \cdot T$ and $50/T$ for convergence and divergence, respectively (also note the dependency on the number of frames T). The performance curves of the PSNR-oriented and the smoothness-oriented modes of our method appear in Figs. 7-8 as blue solid-lines and blue dashed-lines, respectively.

Figures 7-8 show that our method greatly outperforms the regular compression procedure: a BD-PSNR gain of 13.90 dB was achieved for the 'Shields' sequence (Fig. 7a), and a gain of 13.28 dB was obtained for the 'Stockholm' sequence (Fig. 8a). Figures 7b, 8b present also significant gains in SSIM terms.

Let us examine the performance of our approach with respect to the total-variation deblurring technique, both in their PSNR-oriented settings (the blue and red solid-line curves in Figs. 7-8). Considering the 'Shields' (Fig. 7a) and 'Stockholm' (Fig. 8a) sequences, our method achieved respective BD-PSNR gains of 1.06 dB and 2.16 dB over the total-

variation deblurring technique. Figures 7b, 8b exhibit that our PSNR-oriented method is better than the TV deblurring approach also with respect to the SSIM quality metric.

The third comparison considers our method with respect to the total-variation deblurring approach, both in their smoothness-oriented settings (the blue and red dashed-line curves in Figs. 7-8). Considering the 'Shields' (Fig. 7a) and 'Stockholm' (Fig. 8a) sequences, our method differs from the TV-deblurring technique in BD-PSNR values of -0.62 dB and 0.98 dB. One should recall that this is the smoothness-oriented settings, thus, visual quality is preferred over optimizing the PSNR. Indeed, examining the SSIM-bitrate curves of the 'Shields' (Fig. 7b) and 'Stockholm' (Fig. 8b) sequences, exhibit that our smoothness-oriented method obtains the respective average SSIM gains of 3.4×10^{-3} and 5.8×10^{-3} over the smoothness-oriented TV deblurring technique. These results point on the good visual quality offered by the smoothness-oriented settings of our method.

In Figure 9 we provide a visual demonstration of the results obtained for a frame segment from the sequence 'Shields' (Fig. 9a). As the various methods do not produce equal bit-rates, the comparison is for relatively close bit-rates – specifically, the results presented for our method obtained using lower bit-rates than the other techniques. Figures 9b-9f show the frames given to the display using the various methods and their settings. Our method (Fig. 9e-9f) as well as the other deblurring-based approach (Fig. 9c-9d) provide sharpened images to display. Figures 9g-9k exhibit the simulated perceived image (i.e., the displayed frame after the blur degradation modeled above). Evidently, the perceived images corresponding to the pre-compensating techniques highly resemble the original frame. Moreover, Fig. 10 shows for each of the methods the difference between the simulated-perceived and the original images, suggesting that our PSNR-oriented method avoids noticeable image-detail loss, at the expense of some textural-noise that can be attenuated using the smoothness-oriented settings. Importantly, the PSNR-oriented settings of our method achieve the highest PSNR and SSIM values for the presented frame, while using a lower bit-rate than the regular and the total-variation approaches.

V. CONCLUSION

In this paper we studied an image/video compression problem where a linear degradation affects the decompressed signal. We handled the difficult rate-distortion optimization using the ADMM approach, resulting in an iterative procedure relying on a standard compression technique. Our method was shown to be effective in experiments of adjusting HEVC's image and video compression to post-decompression blur degradations. We consider the proposed method as a promising paradigm for addressing complicated rate-distortion optimizations arising in intricate systems and applications. Future work may extend our approach for optimizing systems where several display types are possible. Another interesting process to examine includes a degradation that precedes the compression (e.g., an acquisition stage) in addition to the post-decompression degradation (e.g., rendering). Other intriguing research directions may explore the utilization of our idea

for optimizing compression with respect to various perceptual distortion metrics or to nonlinear systems.

REFERENCES

[1] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Trans. Consum. Electron.*, vol. 38, no. 1, pp. 18–34, Feb. 1992.

[2] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: An overview," *IEEE Trans. Consum. Electron.*, vol. 46, no. 4, pp. 1103–1127, Nov. 2000.

[3] G. J. Sullivan, J.-R. Ohm, W.-J. 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.

[4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[5] Y. Dar, A. M. Bruckstein, and M. Elad, "Image restoration via successive compression," in *Proc. Picture Coding Symp. (PCS)*, Dec. 2016, pp. 1–5.

[6] Y. Dar, M. Elad, and A. M. Bruckstein. (2017). "Restoration by compression." [Online]. Available: https://arxiv.org/abs/1711.05147

[7] M. V. Afonso, J.-M. Bioucas-Dias, and M. A. T. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2345–2356, Sep. 2010.

[8] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg, "Plug-and-play priors for model based reconstruction," in *Proc. IEEE GlobalSIP*, Dec. 2013, pp. 945–948.

[9] Y. Romano, M. Elad, and P. Milanfar, "The little engine that could: Regularization by denoising (RED)," *SIAM J. Imag. Sci.*, vol. 10, no. 4, pp. 1804–1844, 2017.

[10] Y. Dar, A. M. Bruckstein, M. Elad, and R. Giryes, "Postprocessing of compressed images via sequential denoising," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3044–3058, Jul. 2016.

[11] Y. Dar, A. M. Bruckstein, M. Elad, and R. Giryes, "Reducing artifacts of intra-frame video coding via sequential denoising," in *Proc. IEEE Int. Conf. Sci. Elect. Eng. (ICSEE)*, Nov. 2016, pp. 1–5.

[12] A. Rond, R. Giryes, and M. Elad, "Poisson inverse problems by the plug-and-play scheme," *J. Vis. Commun. Image Represent.*, vol. 41, pp. 96–108, Nov. 2016.

[13] D. Geman and C. Yang, "Nonlinear image recovery with half-quadratic regularization," *IEEE Trans. Signal Process.*, vol. 4, no. 7, pp. 932–946, Jul. 1995.

[14] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *Proc. IEEE ICCV*, Nov. 2011, pp. 479–486.

[15] G. Bjontegaard, *Calculation of Average PSNR Differences Between RD-Curves*, document ITU-T Q. 6/SG16 VCEG, 15th Meeting, Austin, TX, USA, Apr. 2001.

[16] RTINGS.com. *Motion Blur of TVs*. Accessed: Oct. 20, 2017. [Online]. Available: http://www.rtings.com/tv/tests/motion/motion-blur-and-response-time

[17] B. Masia, G. Wetzstein, P. Didyk, and D. Gutierrez, "A survey on computational displays: Pushing the boundaries of optics, computation, and perception," *Comput. Graph.*, vol. 37, no. 8, pp. 1012–1038, 2013.

[18] S. Hong, B. Berkeley, and S. S. Kim, "Motion image enhancement of LCDs," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, vol. 2, Sep. 2005, p. II-17-20.

[19] N. Mishima and G. Itoh, "Novel frame interpolation method for hold-type displays," in *Proc. Int. Conf. Image Process. (ICIP)*, vol. 3, Oct. 2004, pp. 1473–1476.

[20] H. Chen, S.-S. Kim, S.-H. Lee, O.-J. Kwon, and J.-H. Sung, "Non-linearity compensated smooth frame insertion for motion-blur reduction in LCD," in *Proc. IEEE Workshop Multimedia Signal Process.*, Oct./Nov. 2005, pp. 1–4.

[21] Y. Dar and A. M. Bruckstein, "Motion-compensated coding and frame rate up-conversion: Models and analysis," *IEEE Trans. Image Process.*, vol. 24, no. 7, pp. 2051–2066, Jul. 2015.

[22] M. A. Klompenhouwer and L. J. Velthoven, "Motion blur reduction for liquid crystal displays: Motion-compensated inverse filtering," *Proc. SPIE*, vol. 5308, pp. 690–699, Jan. 2004.

[23] S. Har-Noy and T. Q. Nguyen, "LCD motion blur reduction: A signal processing approach," *IEEE Trans. Image Process.*, vol. 17, no. 2, pp. 117–125, Feb. 2008.

[24] S. H. Chan and T. Q. Nguyen, "LCD motion blur: Modeling, analysis, and algorithm," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2352–2365, Aug. 2011.

[25] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-36, no. 9, pp. 1445–1453, Sep. 1988.

[26] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 23–50, Nov. 1998.

[27] G. J. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Process. Mag.*, vol. 15, no. 6, pp. 74–90, Nov. 1998.

[28] B. Li, H. Li, L. Li, and J. Zhang, "λ domain rate control algorithm for high efficiency video coding," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 3841–3854, Sep. 2014.

[29] S. Li, M. Xu, Z. Wang, and X. Sun, "Optimal bit allocation for CTU level rate control in HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 11, pp. 2409–2424, Nov. 2016.

[30] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[31] S. H. Chan, X. Wang, and O. A. Elgendy, "Plug-and-play admm for image restoration: Fixed-point convergence and applications," *IEEE Trans. Comput. Imag.*, vol. 3, no. 1, pp. 84–98, Jan. 2017.

[32] F. Bellard. *BPG 0.9.6*. Accessed: Oct. 15, 2017. [Online]. Available: http://bellard.org/bpg/

[33] N. Asuni and A. Giachetti, "TESTIMAGES: A large-scale archive for testing visual devices and basic image processing algorithms," in *Proc. Eurograph. Italian Chapter Conf.*, 2014, pp. 63–70.

[34] G. Schaefer and M. Stich, "UCID: An uncompressed color image database," *Proc. SPIE*, vol. 5307, pp. 472–481, Dec. 2003.

[35] 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 *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 2, Jul. 2001, pp. 416–423.

[36] G. Valenzise. *Bjontegaard Metric (MATLAB Function)*. [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/27798-bjontegaard-metric

[37] H. Pan, X.-F. Feng, and S. Daly, "LCD motion blur modeling and analysis," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, vol. 2, Sep. 2005, p. II-21-4.

[38] T. Kurita, "Moving picture quality improvement for hold-type AM-LCDs," in *SID Symp. Dig. Tech. Papers*, vol. 32, no. 1, 2001, pp. 986–989.

[39] M. A. Klompenhouwer, "Temporal impulse response and bandwidth of displays in relation to motion blur," in *SID Symp. Dig. Tech. Papers*, vol. 36, no. 1, 2005, pp. 1578–1581.

[40] M. A. Klompenhouwer, "Comparison of LCD motion blur reduction methods using temporal impulse response and MPRT," in *SID Symp. Dig. Tech. Papers*, vol. 37, no. 1, 2006, pp. 1700–1703.

[41] S. Tourancheau, K. Brunnström, B. Andrén, and P. Le Callet, "LCD motion-blur estimation using different measurement methods," *J. Soc. Inf. Display*, vol. 17, no. 3, pp. 239–249, 2009.

[42] *Xiph.Org Video Test Media*. Accessed: Oct. 22, 2017. [Online]. Available: http://media.xiph.org/video/derf/

[43] *HEVC Reference Software HM 15.0*, document, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11. Accessed: Oct. 22, 2017. [Online]. Available: http://hevc.hhi.fraunhofer.de/

[44] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[45] S. H. Chan, R. Khoshabeh, K. B. Gibson, P. E. Gill, and T. Q. Nguyen, "An augmented Lagrangian method for total variation video restoration," *IEEE Trans. Image Process.*, vol. 20, no. 11, pp. 3097–3111, Nov. 2011.

Yehuda Dar received the B.Sc. degree in computer engineering and the M.Sc. degree in electrical engineering from the Technion–Israel Institute of Technology in 2008 and 2014, respectively, where he is currently pursuing the Ph.D. degree with the Computer Science Department. His research interests are mainly in signal and image/video processing and their information-theoretic aspects, in particular, in signal compression and restoration. He received the 2017 Jacobs Fellowship for Excellence.

Michael Elad (F'12) received the B.Sc., M.Sc., and D.Sc. degrees from the Department of Electrical Engineering, Technion–Israel Institute of Technology, Israel, in 1986, 1988, and 1997, respectively. Since 2003, he has been a Faculty Member with the Computer Science Department, Technion–Israel Institute of Technology, and since 2010, he has been a Full Professor. He works in the field of signal and image processing, specializing in inverse problems, and sparse representations. He received numerous teaching awards, the 2008 and 2015 Henri Taub Prizes for Academic Excellence, and the 2010 Hershel-Rich prize for innovation. He is a SIAM Fellow (2018). He has been serving as the Editor-in-Chief for the *SIAM Journal on Imaging Sciences* since 2016.

Alfred M. Bruckstein received the B.Sc. and M.Sc. degrees from Technion–Israel Institute of Technology, Haifa, in 1976 and 1980, respectively, and the Ph.D. degree in electrical engineering from Stanford University, California, in 1984. In 1984, he was with the Technion–Israel Institute of Technology, where he holds the Ollendorff Chair in science. His present research interests are in swarm/ant robotics, image and signal processing, analysis and synthesis, pattern recognition, and various aspects of applied geometry. He has authored and co-authored over one hundred and fifty journal papers in the fields of interest mentioned. He is a SIAM Fellow for contributions to signal processing, image analysis, and ant robotics, and received the 2014 SIAG/Imaging Science Prize (for the From Sparse Solutions of Systems of Equations to Sparse Modeling of Signals and Images with D. Donoho and M. Elad).