

Adaptive Debanding Filter

Zhengzhong Tu , Jessie Lin , Yilin Wang, Balu Adsumilli, and Alan C. Bovik, *Fellow, IEEE*

Abstract—Banding artifacts, which manifest as staircase-like color bands on pictures or video frames, is a common distortion caused by compression of low-textured smooth regions. These false contours can be very noticeable even on high-quality videos, especially when displayed on high-definition screens. Yet, relatively little attention has been applied to this problem. Here we consider banding artifact removal as a visual enhancement problem, and accordingly, we solve it by applying a form of content-adaptive smoothing filtering followed by dithered quantization, as a post-processing module. The proposed debanding filter is able to adaptively smooth banded regions while preserving image edges and details, yielding perceptually enhanced gradient rendering with limited bit-depths. Experimental results show that our proposed debanding filter outperforms state-of-the-art false contour removing algorithms both visually and quantitatively.

Index Terms—Debanding, false contour, compression artifact, dithering, post-processing.

I. INTRODUCTION

RECENT years have witnessed significant advancements of video compression technologies. The implementation of modern video coding standards, such as H.264/AVC [1], HEVC [2], VP9 [3], and AV1 [4] have greatly benefited high-quality video streaming applications over bandwidth-constrained networks. Despite the improved capabilities of video codecs, banding artifacts remain a dominant visual impairment of high-quality, high-definition compressed videos. Banding artifacts, which present as sharply defined color bands on otherwise smoothly-varying regions, are often quite noticeable, in part because of the tendency of visual apparatus to enhance sharp gradients, as exemplified by the Mach bands illusion [5]. To further optimize the perceptual quality of compressed user-generated videos [6], [7], developing ways to detect and reduce banding artifacts is a problem of pressing interest.

Prior approaches to banding reduction may be categorized in three ways. If debanding is attempted on the source content before encoding, it is a pre-processing step [8]–[11]. However, these methods have generally been developed for decontouring heavily quantized pictures rather than on compressed/banded video content. A second approach is to apply in-loop processing, whereby the quantization process is adjusted inside the encoder to reduce banding effects [12], [13]. The third approach, post-filtering, has been most widely studied, since it offers maximum

Manuscript received June 19, 2020; revised August 8, 2020; accepted September 7, 2020. Date of publication September 21, 2020; date of current version October 8, 2020. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Shiwen He. (*Corresponding author: Zhengzhong Tu.*)

Zhengzhong Tu and Alan C. Bovik are with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: zhengzhong.tu@utexas.edu; bovik@ece.utexas.edu).

Jessie Lin, Yilin Wang, and Balu Adsumilli are with Google Inc., Mountain View, CA 94043 USA (e-mail: jessielin@google.com; yilin@google.com; balsumilli@google.com).

Digital Object Identifier 10.1109/LSP.2020.3024985

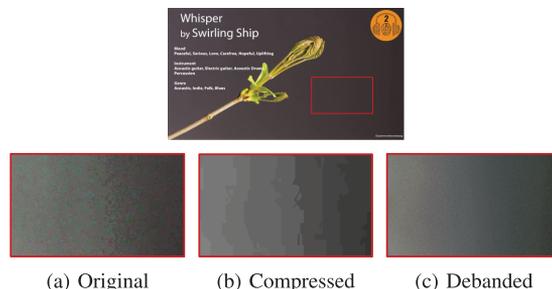


Fig. 1. Examples of banding artifacts introduced by video encoding/transcoding: the first row shows an example of banded videos, while the second row shows contrast-enhanced zoom-in of the box-marked banding area in (a) original, and (b) VP9-compressed video, respectively. Our proposed AdaDeband filter effectively processes the banded regions, yielding perceptually enhanced smoothness rendering, as shown in (c).

freedom for decoder implementations, i.e., the design of post-filters can be relatively unconstrained and flexible. Most banding removal algorithms follow a two-step procedure: first, banding regions are detected and located in the source video frame; then, spatially local filtering is applied to reduce the banding artifacts with dithering sometimes incorporated. For the banding detection stage, some methods [10], [14]–[18] exploit local features, either pixel- or block-wise, such as the image gradient, contrast, or entropy, to measure potential banding statistics. Other methods utilize image segmentation techniques [19], [20]. Either way, banding artifacts are subsequently suppressed by applying low-pass smoothing filters [10], [15], [16], dithering techniques [14], [18], [21], [22], or combinations of these [17], [20].

We deem post-filtering a better approach to handle banding artifacts, since it can be performed outside of the codec loop, e.g., in the display buffer, hence offering maximum freedom of design. Moreover, state-of-the-art adaptive loop filters like those implemented in VP9 or HEVC have not been observed to suppress or have any effect on banding artifacts [17], [19]. Here we propose a new adaptive debanding filter, which we dub the AdaDeband method, as a post-processing solution to reduce perceived banding artifacts in compressed videos. Unlike many prior debanding/decontouring models, we recast the problem differently, as a reconstruction-requantization problem, where we first employ an adaptive interpolation filter along each banded region to estimate the “ideal” plane at a higher bit-depth, then we re-quantize the signal to 8-bit via a dithering technique to suppress quantization error, yielding a visually pleasant enhancement of the banding region. Fig. 1 exemplifies VP9-compressed banding artifacts and the corresponding AdaDeband-filtered output. We demonstrate that our proposed method outperforms other recent debanding methods, both qualitatively and quantitatively.

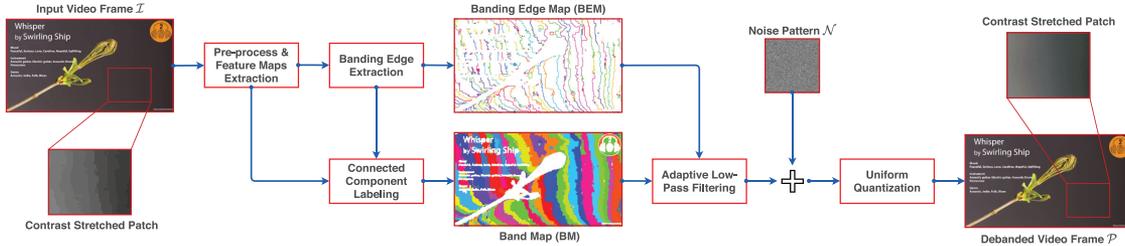


Fig. 2. Schematic overview of the proposed adaptive debanding filter (AdaDeband).

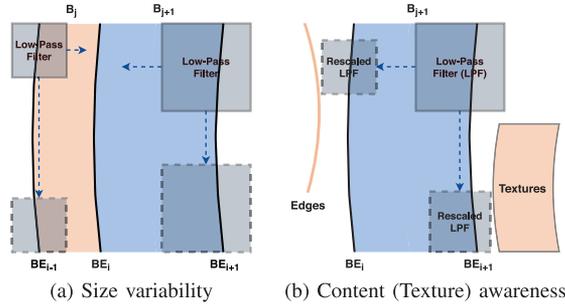


Fig. 3. The size of the low-pass filter window that is applied on different bands depends on both (a) the band widths and (b) contextual edges/textures.

The rest of the letter is structured as follows. We present the details of the proposed adaptive debanding filter in Section II, while evaluative results are given in Section III. Finally, Section IV concludes the letter.

II. ADAPTIVE DEBANDING FILTER

Fig. 2 illustrates a schematic of AdaDeband, which is comprised of three modules. First, a banding detector is deployed to localize the banding edges and band segments with pixel precision. A content-aware size-varying smoothing filter is then applied to reconstruct the gradients within band segments at a higher bit-depth, while preserving image texture details. The final step involves re-quantizing the reconstructed smooth regions to SDR 8-bit resolution using a dithering process to reduce quantization error, yielding a perceptually pleasant debanded image.

A. Banding Region Detection

We utilize the Canny-inspired blind banding detector with same parameters proposed in [23] for banding edge extraction. Only the first two modules, pre-processing and banding edge extraction, are performed to obtain a banding edge map (BEM). For convenience, we restate some definitions from [23]: after self-guided pre-filtering, pixels having gradient magnitudes less than T_1 are labelled as flat pixels (FP); pixels with gradient magnitudes exceeding T_2 are marked as textured pixels (TP). The remaining pixels are grouped into a candidate banding pixel (CBP) set, based on which the BEM is extracted. Connected-component (CC) labeling is applied on the set of non-textured pixels ($FP \cup CBP$), thereby generating a band map (BM). Band edges (B) define the boundaries of adjacent bands (B); i.e., the bands (B) are framed by band edges (BE), as shown in Fig. 3. In this way we define and extract two component sets, BEM and BM, which together compose all the banded regions of a given video frame.

B. Banding Segment Reconstruction

Banding artifacts usually occur on regions of small (but non-zeros) gradients, usually appearing as bands separated by steps of color and/or luminance. As depicted in Fig. 3, pixels lying within a band have very similar colors and luminances. A simple method of restoring an ‘original’ smooth local image is to apply a low-pass filter (LPF) to interpolate across bands. A traditional 2D-LPF may be formulated as:

$$\mathcal{J}(i, j) = \sum_{k=-K}^K \sum_{\ell=-L}^L w_{k, \ell} \mathcal{I}(i - k, j - \ell), \quad (1)$$

where $\mathcal{I}(i, j)$ are input pixel (luminance or color) values at spatial locations (i, j) , and $w_{k, \ell}$ are the filter coefficients, where $\sum_{k, \ell} w_{k, \ell} = 1$. Here we use the simple moving average filter ($w_{k, \ell} = 1/[(2K + 1)(2L + 1)]$) as the smoothing LPF, although one may use any other smoothing filter, such as a Gaussian-shaped filter, or even a nonlinear device such as a median filter.

To effectively smooth band discontinuities, the span of LPF should be adequately wide relative to the band width. In Fig. 3(a), for example, it is only necessary to apply a small LPF on the pink band, whereas a larger LPF would be required for the blue segment. Thus, we apply a dynamic way of determining the filter size for each band, or segment of a band, if the width of the band varies along its axis, based on the extracted BEM and BM. Assume B_j to be an exemplar band, which is framed by detected band edges, BE_{i-1} and BE_i , as depicted in Fig. 3(a). For all the pixels $\mathcal{I}(m, n)$ located within band B_j , the spatial extent of the LPF is defined in terms of the ratios of the band area to the lengths of the adjacent BEs. In the unusual instance where a band is enclosed by a single BE, then the ratio is scaled by four. Both cases are expressed here:

$$l(m, n) = \begin{cases} 4 \times |B_j| / |BE_k|, & k \in \mathcal{N}_{B_j} \text{ if } |\mathcal{N}_{B_j}| = 1 \\ \max_k |B_j| / |BE_k|, & k \in \mathcal{N}_{B_j} \text{ if } |\mathcal{N}_{B_j}| > 1 \end{cases}, \quad (2)$$

where $|S|$ is the cardinality of the pixel set S (area of band, or length of band edge), and \mathcal{N}_{B_j} denotes the set of band edges that enclose B_j . Finally, define the space-varying radius of the LPF window at (m, n) to be half of $l(m, n)$:

$$h(m, n) = \max\{1, \lfloor (l(m, n) - 1) / 2 \rfloor\}. \quad (3)$$

We have ensured that the span of the LPF adapts to the local geometries of banded areas, as shown in Fig. 3(a). Nevertheless, it must also be recognized that the LPFs may process visually important textures or object boundaries while smoothing large banding regions. This kind of content blurring is not acceptable. Accordingly, we constrain the LPF not to include any TP in the sampling window, which is achieved by recursively halving the

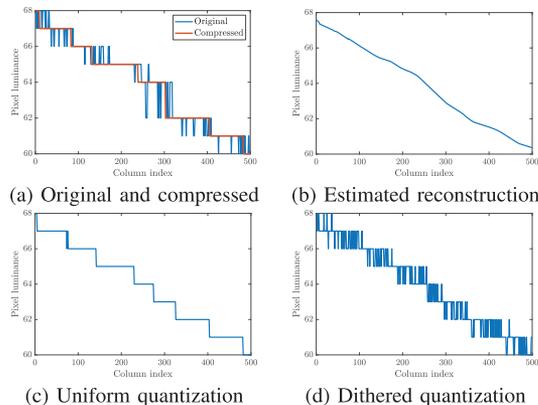


Fig. 4. One-dimensional visualization of the effects of dithered quantization, using row 200 of the exemplary patch in Fig. 2. The uniform quantizer on the (b) reconstructed plane still yields banding, as shown in (c), while dithered quantization generates noisy patterns (as in (d)), similar to the original in (a).

filter size (as in Fig. 3(b)):

$$\begin{aligned} & \text{repeat } h(m, n) := \max\{1, \lfloor h(m, n)/2 \rfloor\} \\ & \text{until } \forall (i, j) \in \mathcal{H}^h(m, n), (i, j) \notin \text{TP} \end{aligned} \quad (4)$$

where $\mathcal{H}^h(m, n) = \{(m+x, n+y)\}$, $x = -h, \dots, h$, $y = -h, \dots, h$, is the set of indices of the (square) filter window centered at (m, n) , with linear dimensions $2h + 1$.

The above filter-size-determining process is performed on every banded pixel in set CBP to generate a window-size map for further deployment of size-varying LPFs. We observed that, however, the estimated window-size map tends to be noisy due to the potential unrobustness of banding detector. Thus, a median filter is applied to further “denoise” the estimated window-size map, based on which LPFs are then conducted via Eq. (1).

C. Requantization With Dithering

Dithering techniques are used in a variety of ways in the design of debanding algorithms. Some methods, for example, only apply filtering without dithering to remove false contours [10], [15], [16]. Other methods apply only a very small amount of dither, either by adding random noise (noise-shaping) [12], [18], [21], [22], or by stochastic shuffling of pixels [14], [17], but without any smoothing filter. Among those that combine filtering with dithering, Baugh [20] proposed to apply dithering after smoothing on banded regions, while the authors of [17] suggested probabilistic dithering prior to average filtering. Here we will show that dithered re-quantization very effectively ameliorates banding artifacts arising from compression.

Fig. 4 shows an example of the effects of processing banded areas with and without dithering, respectively. It may be observed that dithered quantization on the reconstructed banded regions is able to reduce re-quantization error, yielding a pixel distribution similar to the original, while direct quantization without dithering still retains bands. Formally, randomized (dithered) quantization [24] may be expressed as:

$$\mathcal{P}(i, j) = \mathbf{Q}[\mathcal{J}(i, j) + \mathcal{N}(i, j)], \quad (5)$$

where $\mathcal{J}(i, j)$ is the filtered image calculated via Eq. (1), $\mathcal{N}(i, j)$ is a 2D noise image, and $\mathbf{Q}[\cdot]$ is a N-to-8-bit quantizer.

It should be noted that the pattern of the noise image $\mathcal{N}(i, j)$ used for dithering will shape the textures of the resulting debanded regions, hence the type of noise must be carefully selected. We demonstrate several well-known noise patterns and their corresponding dithered outcomes, both visually and quantitatively, in Fig. 5. Among commonly used methods [27]–[29], we chose to employ Gaussian blurred uniform white noise $\mathcal{U}(-2, +2)$, as shown in Fig. 5(d); other recommended and effective options include uniform and $1/f^{1/2}$ noise (Fig. 5(c) (e)). It may also be observed from Fig. 5 that AdaDeband outperforms FCDR and FFmpeg-deband when smoothing the banded staircase, yielding a more pleasant visual enhancement.

III. EXPERIMENTS

We compared our proposed adaptive debanding filter (dubbed AdaDeband) against two recent debanding/decontouring methods proposed for compressed videos: FCDR [17] and the FFmpeg-deband filter [25], on ten¹ selected videos from the YouTube UGC dataset [30]. All the test sequences were scaled to 720p for computational convenience, and we compressed the videos using VP9 constrained quality (CQ) mode with `-crf 39` to generate noticeable banding artifacts. In our implementations, only the luma channel was filtered since much less banding was observed on the Cb/Cr channels, but the proposed AdaDeband could also be applied to each color or chroma component.

Visual comparisons of the debanding methods are shown in Fig. 6. We may see that AdaDeband effectively smoothed the banding, leaving edges/textures well preserved. FFmpeg-deband, in contrast, tended to over-smooth weak textures, and under-smooth relatively large banded regions. Another advantage of AdaDeband is its adaptiveness, which can remove bands of any scale/shape, whereas FFmpeg-deband and FCDR require specification of a set of filter parameters, which may affect performance on scenarios it has not been exposed to.

To further verify the adaptiveness of AdaDeband against different quantization parameters, we generated different levels of banding effects using the VP9 CQ mode, with `-crf` ranging from 1 to 51 on the exemplary video in Fig. 1. We may see in Fig. 7 that our proposed debanding filter produced robust BBAND scores against different levels of quantization without any further tuning of filter parameters.

In addition to the visual results, we also quantitatively compared the debanded videos using several common video quality models. However, it has been shown that traditional video quality metrics like PSNR and even SSIM family [31]–[33] do not align very well with human perception of banding [19]. Moreover, if the original video already contains banding artifacts, which is often the case for high-resolution videos, it is less reliable to rely on full-reference quality models, since “more similarity” with respect to the original is not necessarily indicating “perceptually better.” Therefore, we also compare on a blind banding assessment model (the BBAND index) [23], which has been shown to deliver predictions consistent with subjective judgments of banding. It may be seen in Table I that AdaDeband outperformed

¹Filenames: Gaming_1080P-71a5, NewsClip_1080P-2eb0, Sports_1080P-19d8, Vlog_720P-60f8, LyricVideo_1080P-3b60, LyricVideo_1080P-5a1f, MusicVideo_720P-44c1, MusicVideo_720P-3698, Sports_720P-058f, Vlog_720P-32b2

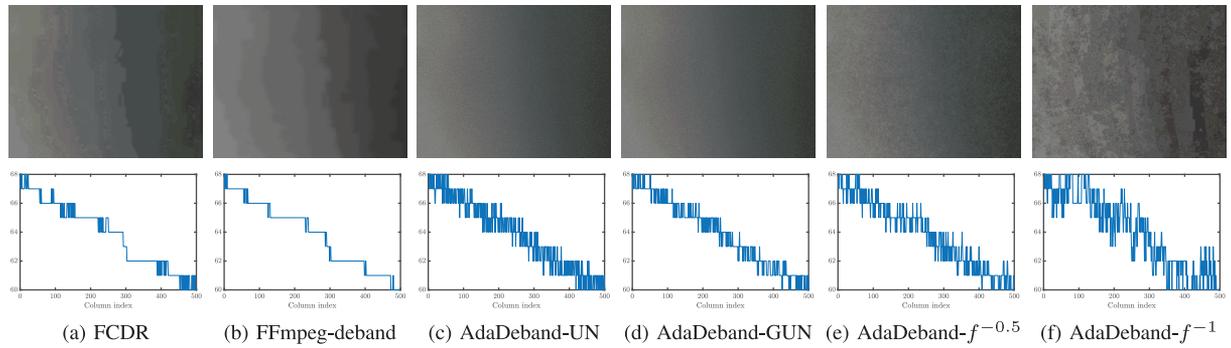


Fig. 5. Visualizations (1st row) and corresponding pixel value distributions (2nd row) of compared debanding outputs on the exemplary patch in Fig. 2. The compared methods are (a) FCDR [17], (b) FFmpeg-deband [25], (c–f) AdaDeband with (c) uniform noise (UN), (d) Gaussian-blurred uniform noise (GUN), (e) $1/f^{1/2}$ noise, and (f) pink noise ($1/f$). The top row of figures has been contrast-stretched for better visualization.

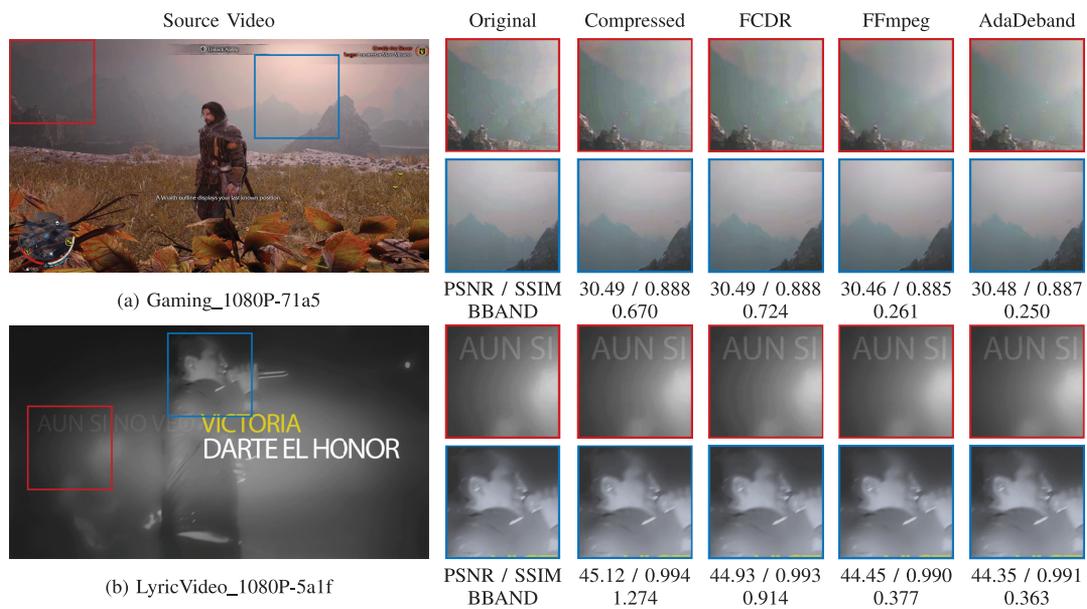


Fig. 6. Qualitative and quantitative comparisons of debanded frames produced by different debanding algorithms. Frame crops (contrast-stretched for better visualization) from left to right: original, compressed, FCDR, FFmpeg-deband, and AdaDeband. We follow the style in [26].

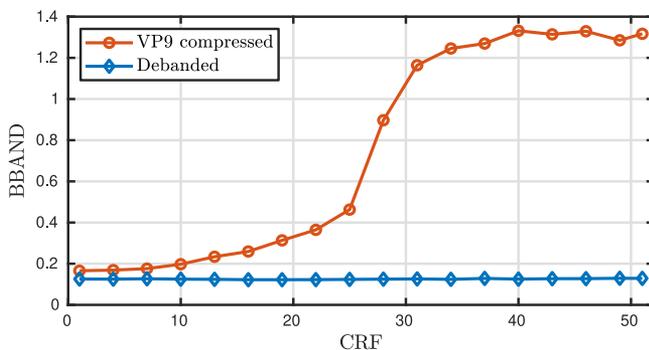


Fig. 7. AdaDeband is robust against different levels of quantization.

FCDR and is on par with FFmpeg-deband in terms of BBAND scores, indicating that it produces perceptually favorable results. Moreover, AdaDeband yields slightly better PSNR and SSIM scores than FFmpeg-deband, indicating less distortion compared to the original.

TABLE I
PERFORMANCE COMPARISON OF DEBANDING METHODS. EACH CELL SHOWS THE EVALUATION RESULTS FORMATTED AS MEAN (\pm STD)

Method	PSNR \uparrow	SSIM \uparrow	BBAND \downarrow
FCDR	39.08 (\pm 5.10)	0.9709 (\pm 0.0308)	0.5790 (\pm 0.2494)
FFmpeg	38.84 (\pm 4.96)	0.9677 (\pm 0.0311)	0.2264 (\pm 0.0903)
AdaDeband	38.97 (\pm 4.97)	0.9699 (\pm 0.0309)	0.2206 (\pm 0.0895)

IV. CONCLUSION

We proposed an adaptive post-debanding filter to remove banding artifacts resulting from coarse video compression. The efficacy of the algorithm can be attributed to the accurate detection of banding regions, content-aware band reconstruction, and a dithered re-quantization. Both visual and quantitative results demonstrate significant performance improvement over prior debanding methods. Actually, it can be regarded as a visual enhancement algorithm, whose effects could be accounted for when performing rate-distortion optimized video encoding. Since it is a post-processing model, it may be optimized for efficient, low-power implementations in real-time applications.

REFERENCES

- [1] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H. 264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.
- [2] 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.
- [3] D. Mukherjee *et al.*, "The latest open-source video codec VP9—An overview and preliminary results," in *Proc. Picture Coding Symp.*, 2013, pp. 390–393.
- [4] Y. Chen *et al.*, "An overview of core coding tools in the AV1 video codec," in *Proc. Picture Coding Symp.*, 2018, pp. 41–45.
- [5] F. Ratliff, *Mach Bands: Quantitative Studies on Neural Networks in the Retina*, ser. Holden-Day Series in Psychology. San Francisco, CA, USA: Holden-Day, 1965. [Online]. Available: <https://books.google.com/books?id=iwwpAQAAMAAJ>
- [6] Z. Tu, Y. Wang, N. Birkbeck, B. Adsumilli, and A. C. Bovik, "UGC-VQA: Benchmarking blind video quality assessment for user generated content," 2020, *arXiv:2005.14354*.
- [7] Z. Tu, C.-J. Chen, L.-H. Chen, N. Birkbeck, B. Adsumilli, and A. C. Bovik, "A comparative evaluation of temporal pooling methods for blind video quality assessment," in *Proc. IEEE Int. Conf. Image Process.*, 2020, *arXiv:2002.10651*.
- [8] L. Roberts, "Picture coding using pseudo-random noise," *IRE Trans. Inf. Theory*, vol. IT-8, no. 2, pp. 145–154, Feb. 1962.
- [9] G. Joy and Z. Xiang, "Reducing false contours in quantized color images," *Comput. Graph.*, vol. 20, no. 2, pp. 231–242, 1996.
- [10] S. J. Daly and X. Feng, "Decontouring: Prevention and removal of false contour artifacts," in *Proc. SPIE Human Vis. Electron. Imag.*, vol. 5292, 2004, pp. 130–149.
- [11] S. J. Daly and X. Feng, "Bit-depth extension using spatiotemporal microdither based on models of the equivalent input noise of the visual system," in *Proc. Color Imag. VIII: Process., Hardcopy, Appl.*, vol. 5008, 2003, pp. 455–466.
- [12] K. Yoo, H. Song, and K. Sohn, "In-loop selective processing for contour artefact reduction in video coding," *Electron. Lett.*, vol. 45, no. 20, pp. 1020–1022, 2009.
- [13] N. Casali, M. Naccari, M. Mrak, and R. Leonardi, "Adaptive quantisation in HEVC for contouring artefacts removal in UHD content," in *Proc. IEEE Int. Conf. Image Process.*, 2015, pp. 2577–2581.
- [14] W. Ahn and J.-S. Kim, "Flat-region detection and false contour removal in the digital TV display," in *Proc. IEEE Int. Conf. Multimedia Expo.*, 2005, pp. 1338–1341.
- [15] H.-R. Choi, J. W. Lee, R.-H. Park, and J.-S. Kim, "False contour reduction using directional dilation and edge-preserving filtering," *IEEE Trans. Consum. Electron.*, vol. 52, no. 3, pp. 1099–1106, 2006.
- [16] J. W. Lee, B. R. Lim, R.-H. Park, J.-S. Kim, and W. Ahn, "Two-stage false contour detection using directional contrast and its application to adaptive false contour reduction," *IEEE Trans. Consum. Electron.*, vol. 52, no. 1, pp. 179–188, Feb. 2006.
- [17] Q. Huang, H. Y. Kim, W.-J. Tsai, S. Y. Jeong, J. S. Choi, and C.-C. J. Kuo, "Understanding and removal of false contour in HEVC compressed images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 2, pp. 378–391, Feb. 2018.
- [18] S. Bhagavathy, J. Llach, and J. Zhai, "Multiscale probabilistic dithering for suppressing contour artifacts in digital images," *IEEE Trans. Image Process.*, vol. 18, no. 9, pp. 1936–1945, Sep. 2009.
- [19] Y. Wang, S.-U. Kum, C. Chen, and A. Kokaram, "A perceptual visibility metric for banding artifacts," in *Proc. IEEE Int. Conf. Image Process.*, 2016, pp. 2067–2071.
- [20] G. Baugh, A. Kokaram, and F. Pitié, "Advanced video debanding," in *Proc. Eur. Conf. Vis. Media Prod.*, 2014, pp. 1–10.
- [21] Y. Wang, C. Abhayaratne, R. Weerakkody, and M. Mrak, "Multi-scale dithering for contouring artefacts removal in compressed UHD video sequences," in *Proc. IEEE Global Conf. Signal Inf. Process.*, 2014, pp. 1014–1018.
- [22] X. Jin, S. Goto, and K. N. Ngan, "Composite model-based DC dithering for suppressing contour artifacts in decompressed video," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2110–2121, Aug. 2011.
- [23] Z. Tu, J. Lin, Y. Wang, B. Adsumilli, and A. C. Bovik, "Bband index: A no-reference banding artifact predictor," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2020, pp. 2712–2716.
- [24] R. A. Wannamaker, S. P. Lipshitz, J. Vanderkooy, and J. N. Wright, "A theory of nonsubtractive dither," *IEEE Trans. Signal Process.*, vol. 48, no. 2, pp. 499–516, Feb. 2000.
- [25] "FFmpeg filters - deband," 2020. Accessed: May 5, 2020. [Online]. Available: <https://ffmpeg.org/ffmpeg-filters.html#deband>
- [26] L.-H. Chen, C. G. Bampis, Z. Li, A. Norkin, and A. C. Bovik, "ProxIQA: A proxy approach to perceptual optimization of learned image compression," 2019, *arXiv:1910.08845*.
- [27] R. A. Ulichney, "Dithering with blue noise," *Proc. IEEE*, vol. 76, no. 1, pp. 56–79, Jan. 1988.
- [28] B. Lippel and M. Kurland, "The effect of dither on luminance quantization of pictures," *IEEE Trans. Commun. Technol.*, vol. CT-19, no. 6, pp. 879–888, Dec. 1971.
- [29] R. A. Ulichney, "Review of halftoning techniques," in *Proc. SPIE Color Imag. Device-Ind. Color Hardcopy Graph. Arts V*, vol. 3963, 1999, pp. 378–391.
- [30] Y. Wang, S. Inguva, and B. Adsumilli, "YouTube UGC dataset for video compression research," in *Proc. IEEE Int. Workshop Multimedia Signal Process.*, 2019, pp. 1–5.
- [31] Z. Wang *et al.*, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [32] S.-C. Pei and L.-H. Chen, "Image quality assessment using human visual DOG model fused with random forest," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3282–3292, Nov. 2015.
- [33] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. IEEE Asilomar Conf. Signals, Syst. Comput.*, vol. 2, 2003, pp. 1398–1402.