Vis Comput (2018) 34:167–176
https://doi.org/10.1007/s00371-016-1322-0

ORIGINAL ARTICLE

An evaluation of power transfer functions for HDR video compression

Jonathan Hatchett[1] · Kurt Debattista[1] · Ratnajit Mukherjee[1] · Thomas Bashford-Rogers[1] · Alan Chalmers[1]

Published online: 6 October 2016

Abstract High dynamic range (HDR) imaging enables the full range of light in a scene to be captured, transmitted and displayed. However, uncompressed 32-bit HDR is four times larger than traditional low dynamic range (LDR) imagery. If HDR is to fulfil its potential for use in live broadcasts and interactive remote gaming, fast, efficient compression is necessary for HDR video to be manageable on existing communications infrastructure. A number of methods have been put forward for HDR video compression. However, these can be relatively complex and frequently require the use of multiple video streams. In this paper, we propose the use of a straightforward Power Transfer Function (PTF) as a practical, computationally fast, HDR video compression solution. The use of PTF is presented and evaluated against four other HDR video compression methods. An objective evaluation shows that PTF exhibits improved quality at a range of bitrates and, due to its straightforward nature, is highly suited for real-time HDR video applications.

Keywords HDR video compression · Power Transfer Function · Video compression metrics

1 Introduction

High dynamic range (HDR) video provides a significant difference in visual quality compared to traditional low dynamic range (LDR) video. With up to 96 bits per pixel (BPP), compared to a standard image of 24 BPP, a single uncompressed HDR frame of 1920 × 1080 resolution requires 24 MB, and a minute of data at 30 fps is 42 GB [5]. To cope effectively with this large amount of data, efficient compression is required. Moreover, if HDR is to gain wide acceptance, and find use in broadcast, internet streaming, remote gaming, etc., it is crucial that computationally efficient encoding and decoding is possible.

HDR video compression may be classified as either a one-stream or two-stream approach [11]. A two-stream method separates the single HDR video input stream into base and detail streams which are then compressed separately according to their individual characteristics. One-stream methods, on the other hand, take advantage of the higher bit-depth available in modern video codecs. A transfer function (TF) is used to map the HDR video input stream to a single, high bit-depth stream and optionally some metadata to aid the post-processing before display. A number of the proposed one-stream methods [9,25] use complex TFs, requiring many floating-point operations for both compression and decompression.

In this paper we evaluate whether straightforward power functions, with their associated computational benefits, can be used to efficiently compress HDR video. We propose a HDR video compression method, the Power Transfer Function (PTF), which aims to provide real-time HDR video encoding without a loss in quality or compression performance.
The key contributions of this work are:

- The presentation of PTF, a straightforward HDR transfer function, with an objective evaluation demonstrating that the method is a highly performant HDR video compression technique.

Electronic supplementary material The online version of this article (doi:10.1007/s00371-016-1322-0) contains supplementary material, which is available to authorized users.

✉ Jonathan Hatchett
j.p.hatchett@warwick.ac.uk

[1] WMG, University of Warwick, Coventry, UK

- An evaluation of the performance of PTF, showing that an analytic implementation of the method exceeds 380 fps for decoding video on commodity hardware, outperforming an analytic implementation of another leading transfer function by over an order of magnitude and a pre-calculated look-up table implementation by a factor of approximately 1.5.

2 Related work

HDR video compression methods can be split into two broad categories: one-stream and two-stream. Two-stream methods have the advantage that they can work well on existing 8-bit infrastructure. One-stream methods, on the other hand, require at least 10-bit infrastructure. The advantage of one-stream methods is that they follow a similar pipeline to those used for LDR video, without the need for secondary streams to be transmitted or combined before display.

Two-stream methods can be considered either backwards compatible or non-backwards compatible based on whether one of the streams can be presented using a non-HDR aware video player. Mantiuk et al. [21,24] presented a method which, following the overall method proposed by [30], tone maps the HDR data to create a backward compatible image [21]. This image is then restored to a colour space compatible with the original and the difference in luminance between the reconstructed frame and the original, is taken and stored a residual data stream. The decoding is performed by reconstructing the tone mapped image and then applying the residuals previously created. The method proposed by Lee and Kim [20] also follows the structure proposed by [30]. In this method the backwards compatible frames are generated using a temporally coherent tone mapper and the residual is created by taking the logarithm of the division of the reconstructed image and the original HDR image. To reduce noise, the residual stream is cross-bilaterally filtered [13]. Other proposed two-stream methods include goHDR [6] and optimal exposure [12].

Several one-stream HDR video compression methods have been proposed in the last 10 years. One of the earliest was by Mantiuk et al. [23] that extended the existing MPEG-4 encoder and attempted to preserve colour and luminance levels visible to the human eye [23]. This mapped the real-world luminances from linear RGB to an 11-bit perceptually uniform luma space and chrominance into an 8-bit uniform chromaticity scale similar to that used in LogLUV encoding [19]. We will refer to this method as HDRV for the remainder of this paper. Garbas and Thoma [17] presented a temporally coherent extension of the Adaptive LogLUV function [26] suitable for HDR video compression [17]. The proposed method maps real-world luminance into a 12-bit luma space and preserves chrominance in 8-bit $u'v'$ chroma channels similar to LogLUV [19]. We will refer to this method as Fraunhofer for the remainder of this paper. Zhang et al. [33] developed a method that converts HDR data to a 32-bit LogLUV colour space [19], after which the 16-bit luminance channel is converted to 14-bit by non-linear quantisation, similar to Lloyd–Max optimisation [28].

The Perceptual Quantizer (PQ) method is based on the fitting of a polynomial function to the peaks in the Barten model of visual perception [7]. Compression is provided by means of a closer fit to a human visual response curve [25]. This method has recently been included in a SMPTE standard, ST2084 [1].

More recently, Borer [9] proposed a compression method based on the log and gamma segments of Mantiuk's analytic model [24] that increases the dynamic range that can be distributed by a factor of 50. This Hybrid–Log–Gamma (HLG) method has been developed to provide support for a display independent television system [10], and has also been included in the Arib STD-B67 standard [3].

3 Power Transfer Function

The human visual system (HVS) has greater sensitivity to relative differences in darker areas of a scene than brighter areas [14,31]. This non-linear response can be generalised by a straightforward power function. The Power Transfer Function (PTF) weights the use of the values available to preserve detail in the areas of the HDR content in which the HVS is more sensitive. PTF therefore allocates more values to the dark regions than to the light regions. The theoretical properties of the power functions used in PTF will be presented in Sect. 3.3.

3.1 Motivation

The recent addition of higher bit-depth support to commonly used video encoding standards such as advanced video coding (AVC) [32], high efficiency video coding (HEVC) [29] and methods such as VP9 has diminished the need for two-stream methods. Instead, this support has motivated an investigation into the efficient mapping of HDR data into 10 and 12 bits. For this purpose, PQ [25] uses a perceptual encoding to map the contrast sensitivity of the HVS to the values available in the video stream. This perceptual encoding, however, relies on a complex transfer function.

In this paper, we investigate whether a transfer function implemented using straightforward power functions can provide an efficient mapping. Power functions also provide computational benefits, particularly for lower integer powers. To perform the PQ mapping [25] requires many calculations, however, a power function can be computed with a single calculation.

Algorithm 1 Power Transfer Function Encoding

```
procedure PTF$_\gamma$($frames_{in}$, $\mathfrak{N}$)
    for $i \leftarrow 1$,LENGTH($frames_{in}$) do
        $S \leftarrow frames_{in}[i]$
        $L \leftarrow S/\mathfrak{N}$
        $V \leftarrow L^{1/\gamma}$
        $Q \leftarrow$QUANTISE($V$)
        $frames_{out}[i] \leftarrow Q$
    end for
    return $frames_{out}$, $\mathfrak{N}$
end procedure
```

Algorithm 2 Power Transfer Function Decoding

```
procedure PTF$'_\gamma$($frames_{in}$, $\mathfrak{N}$)
    for $i \leftarrow 1$,LENGTH($frames_{in}$) do
        $Q \leftarrow frames_{in}[i]$
        $V \leftarrow$DEQUANTISE($Q$)
        $L \leftarrow V^{\gamma}$
        $S \leftarrow L \cdot \mathfrak{N}$
        $frames_{out}[i] \leftarrow S$
    end for
    return $frames_{out}$
end procedure
```

3.2 Method

PTF is a single stream method, converting HDR input into a single set of compressed output frames. To achieve this compression, PTF utilises the power function: $f(x) = Ax^{\gamma}$ where: A is a constant, x is normalised image data contained by the set $[0, 1] \subset \mathbb{R}$ and $\gamma \in \mathbb{R}^{+}$.

The straightforward nature of the PTF method is shown in Fig. 1a, b which present the general pipeline into which PTF is used, and from Algorithms 1 and 2 which detail the compression and decompression procedures, PTF$_\gamma$ and PTF$'_\gamma$, respectively.

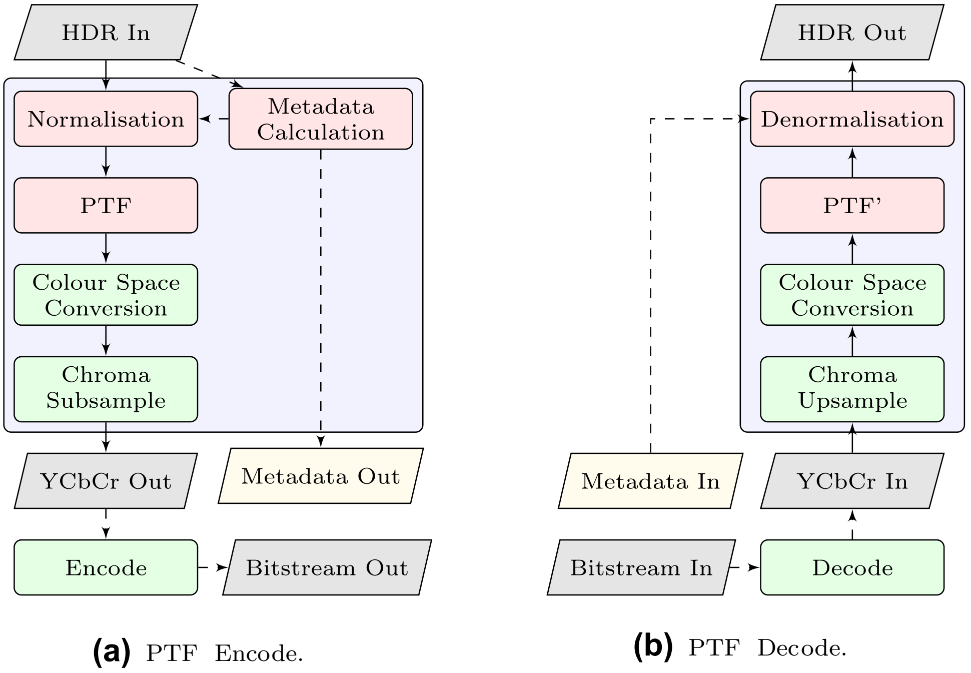
Before a HDR video is compressed using PTF, it is normalised to the range [0, 1] with a normalisation factor $\mathfrak{N}$ using the relation $L = S/\mathfrak{N}$ where: S is full range HDR data. If the footage is of an unknown range then it can be analysed to determine the correct $\mathfrak{N}$ for encoding, or for live broadcast, $\mathfrak{N}$ can be set to the peak brightness the camera is capable of capturing or the display is capable of presenting.

If the normalisation factor is variable, then it can be stored as metadata along with the video data to correctly rescale the footage for display. Each input frame may be normalised independently, however, this may introduce artefacts as the scaling and nonlinearity can interact and lead to the accumulation of errors when using predicted frames. More often a global or temporal normalisation factor is used. The metadata can either be passed at the bitstream level, i.e. with supplemental enhancement information (SEI) messages, or at the container level, i.e. MPEG-4 Part 14 (MP4) data streams.

Following compression with PTF, the data must be converted into the output colour space to be passed to the video encoder, and if chroma subsampling is to be used, reduced to the correct format.

3.3 Theoretical analysis

Figure 2 presents a comparison of just noticeable difference (JND) characteristics from various methods and standards. The Greyscale Display Function (GDF) is an



(a) PTF Encode.

(b) PTF Decode.

Fig. 1 Example pipelines used for encoding and decoding HDR using the PTF. **a** Takes in HDR video frames in either scene or display referred scale and outputs YCbCr for encoding with a standard encoder. **b** Takes as input the encoded bitstream and outputs HDR frames at the initial scale. The *dashed lines* denote optional processing **a** PTF encode. **b** PTF decode

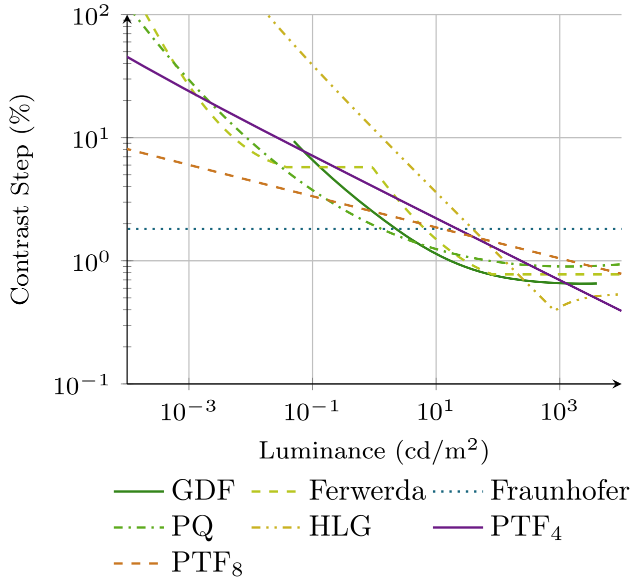


Fig. 2 Contrast sensitivity plots showing GDF as implemented by the DICOM standard, Ferwerda TVI [15] as used by HDRV, Adaptive LogLUV used by the Fraunhofer method, Dolby PQ, BBC HLG and PTF$_4$ and PTF$_8$. Luminance and Contrast are shown on logarithmic scales

implementation of the Barten contrast sensitivity function (CSF) [7] developed for the digital imaging and communications in medicine (DICOM) standard [2]. This CSF plots a relationship between luminance and luma such that the contrast steps between each consecutive luma value are not perceptible. Methods with contrast steps larger than that of the GDF are likely perceptible at that luminance. The DICOM standard GDF is defined with a lower bound of 0.5 × 10^{-1}. As the Fraunhofer method is also based on log luminance it exhibits a purely linear plot on Fig. 2.

To understand how power functions could be adapted for HDR video compression, we investigated the JND characteristics of PTF with the γ values 4 and 8. We chose integer values as we expected them to exhibit reduced computational cost over non-integer values. In Sect. 4.2 we investigate the role of γ in PTF compression. Figure 2 shows how PTF$_4$ compares against other methods. PTF$_8$ expresses too few values for the brighter regions of the image along with reserving a large proportion of the available luma values for a region very close to the lower bound. However, this does provide PTF$_8$ the ability to store a very high dynamic range.

The power function used in PTF is similar to the well-known Gamma function used in LDR video adapted instead to provide HDR video compression. Figure 3 presents a comparison of the shape of the proposed TFs in a normalised space. As a linear plot would express no compression, we can see that PTF$_{2.2}$ provides a small amount of compression.

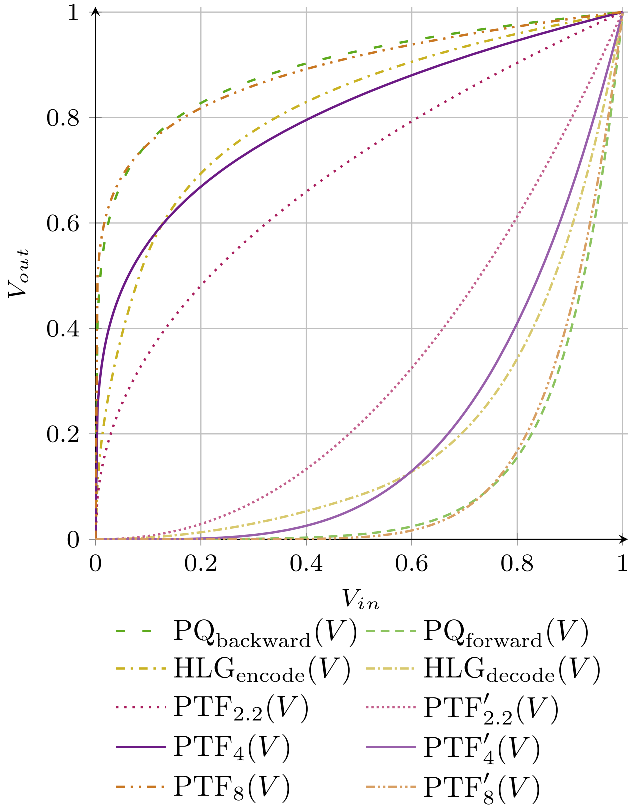
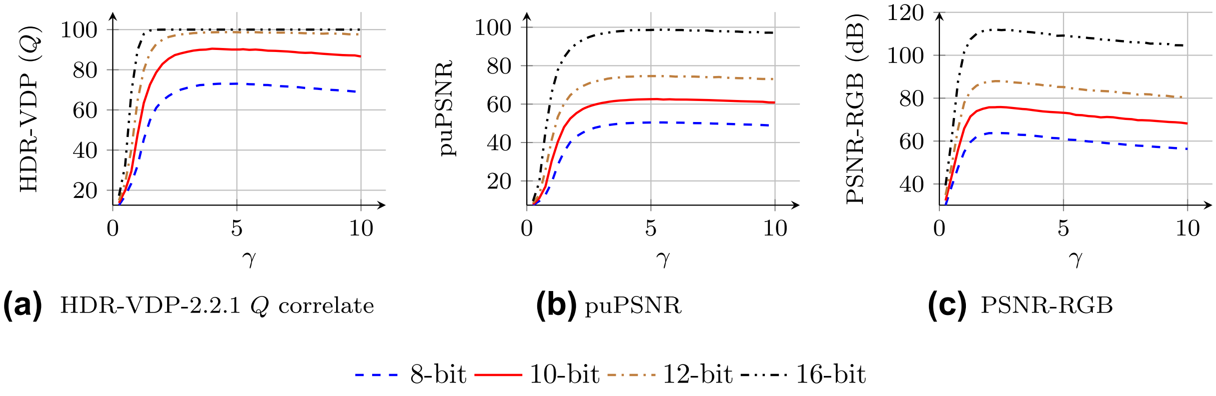


Fig. 3 Graph showing encoding and decoding transfer functions. Presented are PTF$_4$ and PTF$_8$ alongside the PQ and HLG curves. PTF$_{2.2}$ is presented for comparison with an example LDR Gamma function. HLG has been rescaled to the [0, 1] range for comparison with other TFs

4 Results

To evaluate how the efficiency of PTF compares with other proposed methods it has been compared with the following four state-of-the-art one-stream methods (described in more detail in Sect. 2): HDRV [23], Fraunhofer [17], PQ [25], and HLG [9]. For fairness, HDRV and Fraunhofer were adapted from their original presentation for use with a 10-bit video encoder. HDRV was implemented with the luminance range which is reduced 1 × 10^{-5} to 1 × 10^4 such that the TVI curve [15] could provide a mapping from luminance to 10-bit luma. The Fraunhofer implementation uses Adaptive LogLUV [26] which provides mappings for a flexible number of bits.

These methods will be compared on an objective basis using the metrics presented in this section. Subsequently, an analysis of the effect of γ on the coding error introduced by compression is provided. The results of the objective evaluation performed on the compression methods are then presented. Finally, the computational performance of PTF in contrast with PQ and look-up tables is addressed.



(a) HDR-VDP-2.2.1 Q correlate **(b)** puPSNR **(c)** PSNR-RGB

Fig. 4 The relationship between γ and coding error for PTF created at different bit-depths across a range of metrics. The results are the average distortion introduced by PTF for the selection of HDR images in Online Resource **a** HDR-VDP-2.2.1 Q correlate **b** puPSNR **c** PSNR-RGB

4.1 Metrics

The following three metrics are used to provide results for the evaluation.

Peak signal to noise ratio (PSNR) is one of the most widely used metrics for comparing processed image quality. To adapt the method for HDR imaging, L_{peak} was fixed at 10,000 cd/m^2 and the result was taken as the mean of the channel results.

$$\text{PSNR}_\lambda = 20 \log_{10} \left(\frac{L_{\text{peak}}}{\sqrt{\text{MSE}_\lambda}} \right) \tag{1}$$

Perceptually Uniform PSNR (puPSNR) was proposed as an extension to PSNR such that it is capable of handling real-world luminance levels without affecting the results for existing displays [4]. The proposed metric maps the range 1×10^{-5} to 1×10^{-8} cd/m^2 in real-world luminance to values that approximate perceptually uniform values derived from a CSF. It is from the remapped luminance that the PSNR is calculated.

HDR-VDP-2.2.1 (HDR Visual Difference Predictor) is an objective metric based on a detailed model of human vision [22]. The metric estimates the probability at which an average human observer will detect differences between a pair of images in a psychophysical evaluation. The visual model used by this metric takes several aspects of the human visual system into account such intra-ocular light scatter, photo-receptor spectral sensitivities and contrast sensitivity. HDR-VDP-2.2.1 has been shown to be the objective metric that correlates most highly with subjective studies [18,27].

The metrics were calculated for every frame, except HDR-VDP-2.2.1 which was every 10th frame due to its computational expense, and averaged to produce a final figure for the sequence.

4.2 Analysis of power functions

Figure 4a–c show the motivation for the selection of γ by comparing the average distortion introduced by PTF over a range of γ values. These figures suggest that the different metrics favour certain γ values over others. A dataset of 20 HDR images were used for computing the results (these are shown in Online Resource 1).

The pipeline used for this analysis is shown in Fig. 5. After compression and colour conversion the images were not passed through the video encoder and were instead immediately decompressed to ascertain just the coding errors introduced by each γ value. The γ values used in the evaluation ranged from 0.25 to 10 and increased in steps of 0.25. The evaluation was performed at four bit-depths: 8, 10, 12 and 16. PSNR-RGB suggests that a γ of 2.2 will give the best results, and as it is also used for LDR video. HDR-VDP-2.2.1 Q correlate indicates that a γ of around 4 will perform best and puPSNR a γ of around 6. Figure 3 shows that the PQ TF proposed by Miller et al. [25] can be closely approximated by a γ value of 8 and hence the value was also tested. Integer values are favoured as the operations required to decode are significantly faster than non-integers, as discussed in Sect. 4.4.

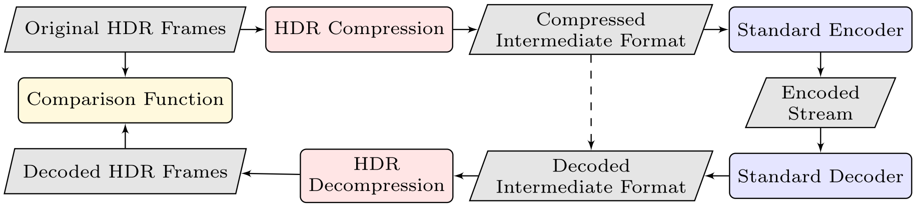


Fig. 5 The evaluation pipeline used for comparing compression methods. The *dashed line* denotes comparison of coding errors only

Based on the peaks of the graph, and similarities to the GDF and PQ (see Sect. 3.3), the four implementations of PTF chosen for testing were: PTF$_{2.2}$, PTF$_4$, PTF$_6$ and PTF$_8$.

Also of note in Fig. 4 is the how the peak in quality does not shift greatly as the bit-depth is increased. This suggests that γ will not need to be changed in an environment of 12 and above bits. This will be explored further in future work.

4.3 Quality

The approach used for quality comparison is outlined in Fig. 5. For each of the compression methods the pipeline is executed in its entirety. The content is provided as individual HDR frames in OpenEXR format. The compression method's encoding process is run on each of the ten sequences of frames, presented in Table 1, to produce 10-bit files in YCbCr format. These sequences were chosen as they cover a wide range of content types, such as computer graphics renderings, video captured by a SphereonVR HDR Video Camera or an ARRI Alexa. Each scene consisted of 150 frames and was encoded at 24 frames per second. The encoding was conducted with the HEVC encoder x265, due its computational efficiency, and 4:2:0 chroma subsampling with the quantisation parameters QP $\in$ [5, 10, 15, 20, 25, 30, 35]. The Group Of Pictures (GOP) structure contained both bi-directional (B) and predicted (P) frames and the pattern used was (I)BBBP where the intra (I) frame period was 30 frames. The encoded bitstreams were then decoded using the HEVC Test Model (HM) [29] reference decoder, and subsequently using the individual compression method's decoding process.

4.3.1 Analysis

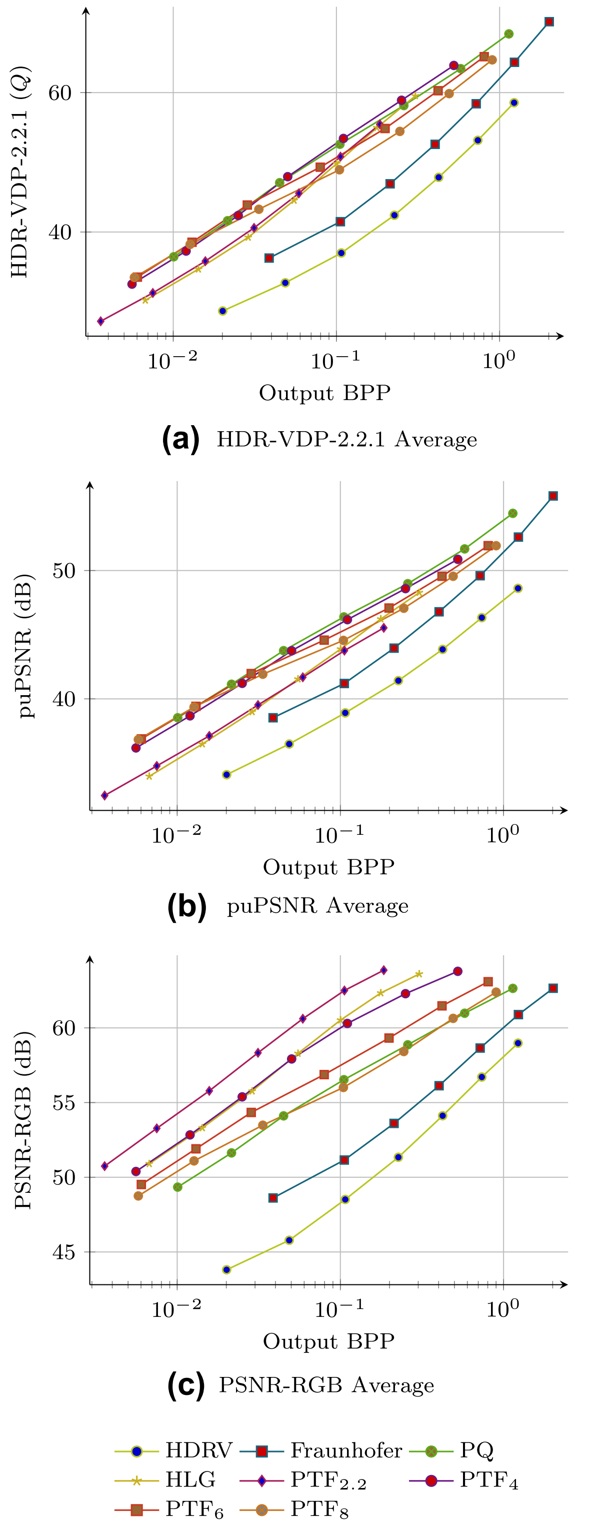
Figure 6a–c show the results for each of the tested methods for the three quality metrics presented in Sect. 4.1. On each of the figures an increase on the Y axis indicates improved objective quality, and a decrease on the X axis indicates reduced bit-rate. Therefore, results closest to the top-left corner are preferred. For each method at each QP, the average BPP of the encoded bitstreams across all sequences is calculated and plotted against the average quality measured. The ten HDR video sequences used to test the compression methods are shown in Table 1. Results for individual sequences are presented in Online Resource 3.

4.3.2 Discussion

The rate-distortion plots shown in Fig. 6 present the trade-off between bit-rate and quality for each method. If a plotted

Table 1 The ten HDR video sequences used to evaluate the methods, showing resolution and dynamic range

Thumbnail	**Name** **Resolution** **Dynamic Range**
	Welding 1920×1080 20.54
	Jaguar 1920×1080 25.30
	Seine 1920×1080 20.54
	Tears of Steel 1920×800 20.35
	Mercedes 1920×1080 19.85
	Beer Festival 4 1900×1060 22.11
	Carousel Fireworks 9 1900×1060 22.38
	Bistro 3 1900×1060 22.13
	Fireplace 1 1900×1060 22.45
	Showgirl 1 1900×1060 22.73



(a) HDR-VDP-2.2.1 Average

(b) puPSNR Average

(c) PSNR-RGB Average

Fig. 6 Rate-distortion characteristics showing the results of each method averaged over the ten sequences and with three metrics, HDR-VDP-2.2.1, PSNR and puPSNR. The Output BPP is shown on a logarithmic scale to improve clarity. **a** HDR-VDP-2.2.1 Average **b** puPSNR Average **c** PSNR-RGB Average

line maintains a position above another, this indicates that improved quality can be consistently obtained from a method even with a reduction in bit-rate.

These figures show that $PTF_{2.2}$ achieves the highest average PSNR followed by HLG then PTF_4. As PSNR does not perceptually weight the error encountered, $PTF_{2.2}$ is rated highly. This is because the close to linear mapping provided by $PTF_{2.2}$ reduces error in the bright regions while failing to preserving detail in the dark regions. The reduced error on the relatively large values found in the bright regions, therefore, favour $PTF_{2.2}$ when tested with PSNR.

HDR-VDP-2.2.1 and puPSNR [4,22] use perceptual weightings that recognise that error in the dark regions is more noticeable to the HVS than the error in the bright regions. These metrics show that on average PTF_4 exhibits the least error for a given bit-rate than the other methods, although for certain sequences, such as Beer Festival 4, PTF_6 achieves the highest quality. PTF_4 weights error in the dark regions more highly than $PTF_{2.2}$ but less highly than PTF_6 or PTF_8.

The Bjøntegaard delta metric [8] calculates the average difference in quality between pairs of methods encoding sequences at the same bit-rate. Using this metric we can determine the average HDR-VDP-2.2.1 Q correlate gain over the range of bit-rates achieved by PTF when compared with the other methods evaluated. From Table 2 it can be seen that PTF_4 gained 0.32 over PQ, 2.90 over HLG, 7.28 over Fraunhofer and 13.35 over HDRV. We can also see that PTF_4 gained 0.96 over PTF_6, 2.24 over PTF_8 and 2.39 over $PTF_{2.2}$. A table showing Bjøntegaard delta bit-rate metric results is available in Online Resource 4. A useful feature of PTF is its adaptability which enables the use of different γ values to provide the best performance for particular sequences.

4.4 Computational performance

High performance is essential for real-world encoding and decoding. With that in mind we compared PTF against an analytical implementation of PQ [25] and against look-up tables (LUTs).

Table 3 shows the decoding performance of PTF'_4 and PQ and their LUT equivalents, PTF'_4 LUT and PQ LUT, for the scenes presented in Table 1. The 1D LUTs were generated by storing the result of each transfer function for every 10-bit input value in a floating-point array. The scaling required to reconstruct the full HDR frame was also included in the table to improve performance resulting in a mapping from 10-bit compressed RGB to full HDR floating-point. The results were produced by a single-threaded C++ implementation compiled with the Intel C++ Compiler v16.0. Only the inner

Table 2 Bjøntegaard delta VDP results showing the average improvement in HDR-VDP-2.2.1 Q correlate results between pairs of methods over ten sequences

	$\mathbf{PTF_{2.2}}$	$\mathbf{PTF_4}$	$\mathbf{PTF_6}$	$\mathbf{PTF_8}$	**HDRV**	**Fraun.**	**PQ**	**HLG**
$\mathbf{PTF_{2.2}}$	0.0	−2.39	−1.11	0.14	11.32	5.65	−1.18	0.42
$\mathbf{PTF_4}$	2.39	0.0	0.96	2.24	13.35	7.28	0.32	2.9
$\mathbf{PTF_6}$	1.11	−0.96	0.0	1.3	12.52	6.74	−0.62	1.58
$\mathbf{PTF_8}$	−0.14	−2.24	−1.3	0.0	11.18	5.42	−1.91	0.2
HDRV	−11.32	−13.35	−12.52	−11.18	0.0	−5.1	−13.64	−11.39
Fraunhofer	−5.65	−7.28	−6.74	−5.42	5.1	0.0	−7.95	−5.87
PQ	1.18	−0.32	0.62	1.91	13.64	7.95	0.0	1.93
HLG	−0.42	−2.9	−1.58	−0.2	11.39	5.87	−1.93	0.0

Positive numbers denote a HDR-VDP-2.2.1 Q correlate improvement on average over the range of bit-rates exhibited by the row method on the left verses the column method above

Table 3 Difference in decoding time per frame between PTF$'_4$, PQ$_{\text{forward}}$ and their LUT equivalents across a range of sequences and averaged over five tests per sequence performed on a workstation PC

Name	Time per frame (ms)				Speed up (ratio)	
	Analytic		LUT		PTF$'_4$	
	PTF$'_4$	PQ	PTF$'_4$	PQ	PQ	LUT
Welding	2.57	66.37	4.13	3.95	25.85	1.61
Jaguar	2.73	66.78	3.92	3.87	24.47	1.44
Seine	2.58	64.01	3.92	3.92	24.86	1.52
Tears of Steel	2.69	98.08	3.95	3.91	36.49	1.47
Mercedes	2.72	73.57	3.80	3.95	27.00	1.39
Beer festival 4	2.61	65.16	3.73	3.81	24.92	1.43
Carousel Fireworks 9	2.56	65.91	3.77	3.93	25.79	1.48
Bistro 3	2.63	65.85	3.82	3.95	25.00	1.45
Fireplace 2	2.31	129.84	3.66	3.86	56.22	1.58
Showgirl 1	2.70	69.39	3.89	3.99	25.69	1.44
Average	2.61	76.50	3.86	3.91	29.63	1.48

Speed up is ratio between and PQ$_{\text{forward}}$ and between PTF$'_4$, and PTF$'_4$ and the LUT implementation of PTF$_4$. A graphical version of this table is shown in Online Resource 2

loop was timed so disk read and write speeds are not taken into account. Each result was taken as the average of five tests per method on each sequence to reduce the variance associated with CPU timing. The software was compiled with the AVX2 instruction set with automatic loop-unrolling, O3 optimisations and fast floating-point calculations. The machine used to run the performance tests was an Intel Xeon E3-1245v3 running at 3.4 GHz with 16 GB of RAM running the Microsoft Windows 8.1 x86-64 operating system.

The encoding performance was also evaluated for the methods. In this case the mapping was from full HDR floating-point to 10-bit output and hence the LUT implementations could not include scaling in the table. The sequences, resolution and sequence lengths were the same as above. PTF$_4$ encoding was achieved on average per frame in 4.37 ms, PQ encoding in 72.59 ms, PTF$_4$ LUT in 4.02 ms and PQ LUT in 4.21 ms.

The results demonstrate that the straightforward floating-point calculations required to decode PTF$_4$ can outperform the floating-point calculations required to decode PQ by a factor of 29.63 times and even the indexing needed to use a look-up table by 1.48 times. The high performance of PTF$'_4$ is due to its compilation into only a few instructions, in this case three multiplies, that can have high performance SIMD implementations. PTF also avoids any branching, improving performance on pipelined architectures. Encoding PTF$_4$ can be achieved at a speed comparable to the use of LUT and greatly in excess of an analytic implementation of PQ.

5 Conclusion and future work

In this paper, we have introduced and evaluated a straightforward method of compressing HDR video streams. We have shown that a transfer function based on power functions is capable of producing high quality compressed HDR video and that the compression can be achieved using straightforward techniques which lend themselves to implementation in real-time and low-power environments. On a commodity desktop machine, PTF is able to be decoded at over 380 fps and outperforms an analytic implementation of PQ by a factor of over 29.5 and a look-up implementation by a factor of nearly 1.5. Encoding performance outperforms PQ by a factor of 16.6 and is only slightly slower than a LUT. Thanks to its straightforward nature, PTF is amiable to acceleration through the use of hardware such as FPGAs and GPUs. We intend to develop an implementation on such platforms in the future. As a continuation of this work we would like to confirm the objective results with a subjective evaluation. This could also serve as further confirmation of the correlation between HDR-VDP-2.1.1 results and experiments involving human participants.

Acknowledgements We would like to thank Edouard Francois for the Seine content which has been kindly provided by Technicolor. For the Cars Longshot, Beerfest Lightshow, Carousel Fireworks, Bistro, Fireplace and Showgirl sequences we would like to thank the University of Stuttgart [16]. We would like to thank the Blender Foundation for Tears of Steel which is made available under the Creative Commons. Alan Chalmers and Kurt Debattista are Royal Society Industrial Fellows. We would also like to thank Scott Daly and Pete Shirley for their comments on the paper.

References

1. ST 2084:2014: High dynamic range electro-optical transfer function of mastering reference displays. Society of Motion Picture and Television Engineers (SMPTE) (2014)
2. Association, N.E.M., of Radiology, A.C., et al.: Digital imaging and communications in medicine (DICOM), Part 14: Grayscale Standard Display Function. National Electrical Manufacturers Association (1998)
3. Association of Radio Industries and Businesses (ARIB): STD-B67: essential parameter values for the extended image dynamic range television (EIDRTV) system for programme production (2015)
4. Aydın, T.O., Mantiuk, R., Seidel, H.P.: Extending quality metrics to full luminance range images. In: Electronic Imaging 2008, pp. 68,060B–68,060B (2008)
5. Banterle, F., Artusi, A., Debattista, K., Chalmers, A.: Advanced High Dynamic Range Imaging: Theory and Practice. CRC Press, Boca Raton (2011)
6. Banterle, F., Artusi, A., Debattista, K., Ledda, P., Chalmers, A., Edwards, G.J., Bonnet, G.: HDR video data compression devices and methods (2010). EP Patent 2,144,444
7. Barten, P.G.: Physical model for the contrast sensitivity of the human eye. SPIE/IS&T **1992**, 57–72 (1992)
8. Bjøntegaard, G.: Calculation of average psnr differences between rd-curves. Doc. VCEG-M33 ITU-T Q6/16, Austin, TX, USA, 2-4 April 2001 (2001)
9. Borer, T.: Non-linear opto-electrical transfer functions for high dynamic range television (2014)
10. Borer, T., Cotton, A.: A display independent high dynamic range television system (2015)
11. Chalmers, A., McNamee, J., Hatchett, J., Mukherjee, R., Olaizola, I., Debattista, K.: 12 Bits is Simply not Enough for HDR Video!. BEC, NAB (2015)
12. Debattista, K., Bashford-Rogers, T., Selmanović, E., Mukherjee, R., Chalmers, A.: Optimal exposure compression for high dynamic range content. Vis. Comput. **31**(6), 1089–1099 (2015)
13. Eisemann, E., Durand, F.: Flash photography enhancement via intrinsic relighting. In: ACM transactions on graphics (TOG), vol. 23, pp. 673–678. ACM (2004)
14. Fechner, G.T.: Ueber eine scheibe zur erzeugung subjectiver farben. Annalen der Physik **121**(10), 227–232 (1838)
15. Ferwerda, J.A., Pattanaik, S.N., Shirley, P., Greenberg, D.P.: A model of visual adaptation for realistic image synthesis. In: SIGGRAPH, pp. 249–258. ACM (1996)
16. Froehlich, J., Grandinetti, S., Eberhardt, B., Walter, S., Schilling, A., Brendel, H.: Creating cinematic wide gamut hdr-video for the evaluation of tone mapping operators and hdr-displays (2014)
17. Garbas, J.U., Thoma, H.: Temporally coherent luminance-to-luma mapping for high dynamic range video coding with H.264/AVC. In: ICASSP, pp. 829–832. IEEE (2011)
18. Hanhart, P., Bernardo, M.V., Pereira, M., Pinheiro, A.M., Ebrahimi, T.: Benchmarking of objective quality metrics for hdr image quality assessment. EURASIP J. Image Video Process. **2015**(1), 1–18 (2015)
19. Larson, G.W.: Logluv encoding for full-gamut, high-dynamic range images. J. Graphics Tool. **3**(1), 15–31 (1998)
20. Lee, C., Kim, C.S.: Rate-distortion optimized compression of high dynamic range videos. In: Proceedings of the 16th European Signal Processing Conference (2008)
21. Mantiuk, R., Efremov, A., Myszkowski, K., Seidel, H.P.: Backward compatible high dynamic range mpeg video compression. In: ACM Transactions on Graphics (TOG), vol. 25, pp. 713–723. ACM (2006a)
22. Mantiuk, R., Kim, K.J., Rempel, A.G., Heidrich, W.: HDR-VDP-2: a calibrated visual metric for visibility and quality predictions in all luminance conditions. In: ACM Transactions on Graphics (TOG), vol. 30, p. 40. ACM (2011)
23. Mantiuk, R., Krawczyk, G., Myszkowski, K., Seidel, H.P.: Perception-motivated high dynamic range video encoding. In: ACM Transactions on Graphics (TOG), vol. 23, pp. 733–741. ACM (2004)
24. Mantiuk, R., Myszkowski, K., Seidel, H.P.: Lossy compression of high dynamic range images and video. In: Electronic Imaging 2006, pp. 60,570V–60,570V. International Society for Optics and Photonics (2006b)
25. Miller, S., Nezamabadi, M., Daly, S.: Perceptual signal coding for more efficient usage of bit codes. SMPTE Conf. **2012**, 1–9 (2012)
26. Motra, A., Thoma, H.: An adaptive logluv transform for high dynamic range video compression. In: Image Processing (ICIP), pp. 2061–2064. IEEE (2010)
27. Mukherjee, R., Debattista, K., Bashford-Rogers, T., Vangorp, P., Mantiuk, R., Bessa, M., Waterfield, B., Chalmers, A.: Objective and subjective evaluation of high dynamic range video compression. Signal Processing: Image Communication **47**, 426–437 (2016). doi:10.1016/j.image.2016.08.001. http://www.sciencedirect.com/science/article/pii/S0923596516301084
28. Scheunders, P.: A genetic lloyd-max image quantization algorithm. Pattern Recogn. Lett. **17**(5), 547–556 (1996)
29. Sullivan, G.J., Ohm, J.R., Han, W.J., Wiegand, T.: Overview of the high efficiency video coding (hevc) standard. IEEE TCSVT **22**(12), 1649–1668 (2012)
30. Ward, G., Simmons, M.: Subband encoding of high dynamic range imagery. In: Proceedings of the 1st Symposium on Applied Perception in Graphics and Visualization, pp. 83–90. ACM (2004)
31. Weber, E.H.: De Pulsu, resorptione, auditu et tactu: Annotationes anatomicae et physiologicae. CF Koehler (1834)
32. Wiegand, T., Sullivan, G.J., Bjøntegaard, G., Luthra, A.: Overview of the h. 264/avc video coding standard. IEEE TCSVT **13**(7), 560–576 (2003)
33. Zhang, Y., Reinhard, E., Bull, D.: Perception-based high dynamic range video compression with optimal bit-depth transformation. In: Image Processing (ICIP), pp. 1321–1324. IEEE (2011)

Jonathan Hatchett is currently undertaking a Ph.D. in High Dynamic Range video compression at the University of Warwick following a Batchelors in Computer Science at the same institution. His research interests include HDR, image and video processing and GPGPU techniques.

Kurt Debattista is an Associate Professor at the University of Warwick. He has a Ph.D. from the University of Bristol, an M.Sc. in Psychology, an M.Sc. in Computer Science and a B.Sc. in Mathematics and Computer Science. His research interests include HDR imaging, high-fidelity computer graphics, serious games and parallel computing.

Ratnajit Mukherjee is a Ph.D. student in Visualisation Group at the University of Warwick. His research interests include HDR, video compression, signal and image processing. He completed his Bachelor of Technology degree, with honours in Information Technology from West Bengal University of Technology, WB, India, and conducted research and development of Embedded Assistive Technology devices at the Indian Institute of Technology, Kharagpur, India.

Thomas Bashford-Rogers has a degree in Computer Science from the University of Bristol and a doctorate in Computer Graphics from the University of Warwick. His research interests include global illumination, raytracing and Monte Carlo methods. He is currently a Research Fellow at the University of Warwick.

Alan Chalmers is Professor of Visualisation at the University of Warwick. He has a Ph.D. in Computer Science from the University of Bristol, 1991 and an M.Sc. with distinction from Rhodes University, SA, 1984. He has published over 210 papers in journals and international conferences on highfidelity virtual environments, multi-sensory perception, and HDR imaging.