High Dynamic Range Imaging: Towards the Limits of the Human Visual Perception

Rafał Mantiuk
Max-Planck-Institut für Informatik
Saarbrücken

1 Introduction

Vast majority of digital images and video material stored today can capture only a fraction of visual information visible to the human eye and does not offer sufficient quality to reproduce them on the future generation of display devices. The limitating factor is not the resolution, since most consumer level digital cameras can take images of higher number of pixels than most of displays can offer. The problem is a limited color gamut and even more limited dynamic range (contrast) that cameras can capture and that majority of image and video formats can store.

For instance, each pixel value in the JPEG image encoding is represented using three 8-bit integer numbers (0-255) using the YC_rC_b color space. Such color space is able to store only a small part of visible color gamut (although containing the colors most often encountered in the real world), as illustrated in Figure 1-left, and even smaller part of luminance range that can be perceived by our eyes, as illustrated in Figure 1-right. The reason for this is that the JPEG format was designed to store as much information as can be displayed on the majority of displays, which were Cathode Ray Tube (CRT)

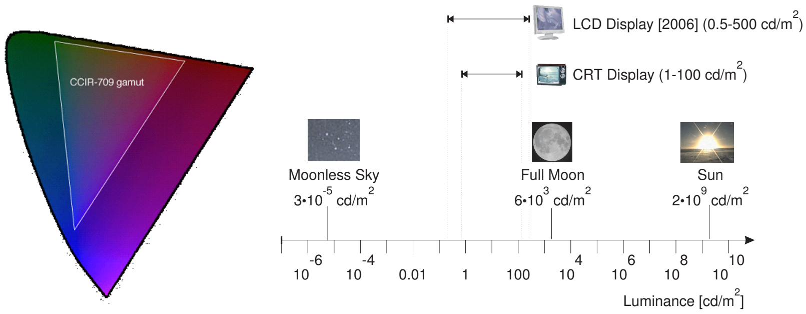

Fig. 1: Left: color gamut frequently used in traditional imaging (CCIR-705), compared to the full visible color gamut. Right: real-world luminance values compared with the range of luminance that can be displayed on CRT and LDR monitors.

monitors at the time when the JPEG compression was developed. This assumption is no longer valid, as the new generations of LCD and Plasma displays can visualize much broader color gamut and dynamic range than their CRT ancestors. Moreover, as new display devices become available, there is a need for higher precision of image and video content. The traditional low-dynamic range and limited color gamut imaging, which is confined to three 8-bit integer color channels, cannot offer the precision that is needed for the further developments in image capture and display technologies.

The High Dynamic Range Imaging (HDRI) overcomes the limitation of traditional imaging by using much higher precision when performing operations on color. Pixel colors are specified in HDR images as a triple of floating point values (usually 32-bit per color channel), providing the accuracy that is far below the visibility threshold of the human eye. Moreover, HDRI operates on colors of original scenes, instead of their renderings on a particular display medium, as is the case of the traditional imaging. By its inherent colorimetric precision, HDRI can represent all colors that can be found in real world and can be perceived by the human eye.

HDRI, which originated from the computer graphics field, has been recently gaining momentum and revolutionizing almost all fields of digital imaging. One of the breakthroughs of the HDR revolution was the development of an HDR display, which proved that the visualization of color and the luminance range close to real scenes is possible (Seetzen, Heidrich, Stuerzlinger, Ward, Whitehead, Trentacoste, Ghosh & Vorozcovs 2004). One of the first to adopt HDRI were video game developers together with graphics card vendors. Today most of the state-of-the art video game engines perform rendering using HDR precision to deliver more believable and appealing virtual reality worlds. A computer generated imagery used in the special effect production strongly depends on the HDR techniques. High-end cinematographic cameras, both analog and digital, already provide significantly higher dynamic range than most of the displays today. Their quality can be retained after digitalization only if a form of HDR representation is used. HDRI is also a strong trend in digital photography, mostly due to the multi-exposure techniques, which can be used to take an HDR image even with a consumer level digital camera. To catch up with the HDR trend, many software vendors announce their support of the HDR image formats, taking Adobe® Photoshop® CS2 as an example.

Besides its significant impact on existing imaging technologies that we can observe today, HDRI has potential to radically change the methods in which imaging data is processed, displayed and preserved in several fields of science. Computer vision algorithms can greatly benefit from the increased precision of HDR images, which lack over- or under-exposed regions, which are often the cause of the algorithms failure. Medical imaging has already developed image formats (DICOM format) that can partly cope with shortcomings of traditional images, however they are supported only by specialized hardware and software. HDRI gives the sufficient precision for medical imaging and therefore its capture, processing and rendering techniques can be used also in this field. For instance, HDR displays can show even better contrast than high-end medical displays and therefore facilitate diagnosing based on CT scans. HDR techniques can also find applications in astronomical imaging, remote sensing, industrial design and scientific visualization.

HDRI does not only provide higher precision, but also enables to synthesize, store and visualize a range of perceptual cues, which are not achievable with the traditional imaging. Most of the imaging standards and color spaces have been developed to match the needs of office or display illumination conditions. When viewing such scenes or images in such conditions, our visual system operates in a mixture of day-light and dim-light vision state, so called the mesopic vision. When viewing out-door scenes, we use day-light perception of colors, so called the photopic vision. This distinction is important for digital imaging as both types of vision shows different performance and result in different perception of colors. HDRI can represent images of luminance range fully covering both the photopic and the mesopic vision, thus making distinction between them possible. One of the differences between mesopic and photopic vision is the impression of colorfulness of objects. We tend to regard objects more colorful when they are brightly illuminated, which is the phenomena that is called Hunt's effect. To render enhanced colorfulness properly, digital images must preserve information about the actual level of luminance of the original scene, which is not possible in the case of the traditional imaging. Real-world scenes are not only brighter and more colorful than their digital reproductions, but also contain much higher contrast, both local between neighboring objects, and global between distant objects. The eye has evolved to cope with such high contrast and its presence in a scene evokes important perceptual cues. The traditional imaging, unlike HDRI, is not able to represent such high-contrast scenes. Similarly, the traditional images can hardly represent such common visual phenomena as self-luminous surfaces (sun, shining lamps) and bright specular highlights. They also do not contain enough information to reproduce visual glare (brightening of the areas surrounding shining objects) and a short-time dazzle due to sudden raise of light level (e.g. when exposed to the sunlight after staying indoors). To faithfully represent, store and then reproduce all these effects, the original scene must be stored and treated using high fidelity HDR techniques.

Despite its advantages, the inception of HDRI in various fields of digital imaging poses serious problems. The biggest is the lack of well standardized color spaces and image formats, of which traditional imaging is abundant. Such color spaces and image formats would facilitate exchange of information between HDR applications. Due to the different treatment of color, introduction of HDRI also requires redesigning entire imaging pipeline, including acquisition (cameras, computer graphics synthesis), storage (formats, compression algorithms) and display (HDR display devices and display algorithms).

This paper summarizes the work we have done to make the transition from the traditional imaging to HDRI smoother. In the next section we describe our implementation of HDR image and video processing framework, which we created for the purpose of our research projects and which we made available as an Open Source project. Section 3 describes our contributions in the field of HDR image and video encoding. These include a perceptually motivated color space for efficient encoding of HDR pixels and two extensions of MPEG standard that allow to store movies containing full color gamut and luminance range visible to the human eye.

2 HDR Imaging Framework

Most of the traditional image processing libraries store each pixel using limited-precision integer numbers. Moreover, they offer restricted means of colorimetric calibration. To overcome these problems, we have implemented HDR imaging framework as a package of several command line programs for reading, writing, manipulating and viewing high-dynamic range (HDR) images and video frames. The package was intended to solve our current research problems, therefore simplicity and flexibility were priorities in its design. Since we found the software very useful in numerous projects, we decided to make it available for the research community as an Open Source project licensed under the GPL. The software is distributed under the name *pfstools* and its home page can be found at `http://pfstools.sourceforge.net/`.

The major role of the software is the integration of several imaging and image format libraries, such as *ImageMagick*, *OpenEXR* and *NetPBM*, into a single framework for processing high precision images. To provide enough flexibility for a broad range of applications, we have build *pfstools* on the following concepts:

– Images/frames should hold an arbitrary number of channels (layers), which can represent not only color, but also depth, alpha-channel, and texture attributes;
– Each channel should be stored with high precision, using floating point numbers. If possible, the data should be colorimetrically calibrated and provide the precision that exceeds the performance of the human visual system.
– Luminance should be stored using physical units of cd/m^2 to distinguish between the night- and the day-light vision.
– There should be user-defined data entries for storing additional, application specific information (e.g. colorimetric coordinates of the white point).

pfstools are built around a generic and simple format of storing images, which requires only a few lines of code to read or write. The format offers arbitrary number of channels, each represented as a 2-D array of 32-bit floating point numbers. There is no compression as the files in this format are intended to be transferred internally between applications without writing them to a disk. A few channels have a predefined function. For example, channels with the IDs 'X', 'Y' and 'Z' are used to store color data in the CIE XYZ (absolute) color space. This is different to most imaging frameworks that operate on RGB channels. The advantage of the CIE XYZ color space is that it is precisely defined in terms of spectral radiance and the full visible color gamut can be represented using only positive values of color components. The file format also offers a way to include in an image any number of user *tags* (name and value pairs), which can contain any application dependent data. A sequence of images is interpreted by all "pfs-compliant" applications as consequtive frames of an animation, so that video can be processed in the same way as images. The format is described in detail in a separate specification[1].

pfstools are a set of command line tools with almost no graphical user interface. This greatly facilitates scripting and lessens the amount of work needed to program and maintain a user interface. The exception is a viewer

[1] Specification of the *pfs* format can be found at:
`http://www.mpi-sb.mpg.de/resources/pfstools/pfs_format_spec.pdf`

of HDR images. The main components of *pfstools* are: programs for reading and writing images in all major HDR and LDR formats (e.g. OpenEXR, Radiance's RGBE, logLuv TIFF, 16-bit TIFF, PFM, JPEG, PNG, etc.), programs for basic image manipulation (rotation, scaling, cropping, etc.), an HDR image viewer, and a library that simplifies file format reading and writing in C++. The package includes also an interface for *matlab* and *GNU Octave*. The *pfstools* framework does not impose any restrictions on the programming language. All programs that exchange data with *pfstools* must read or write the file format, but there is no need to use any particular library. The typical usage of *pfstools* involves executing several programs joined by UNIX pipes. The first program transmits the current frame or image to the next one in the chain. The final program should either display an image or write it to a disk. Such pipeline architecture improves flexibility of the software but also gives straightforward means for parallel execution of the pipeline components on multiprocessor computers. Some examples of command lines are given below:

```
pfsin␣input.exr␣|␣pfsfilter␣|␣pfsout␣output.exr
```

Read the image `input.exr`, apply the filter `pfsfilter` and write the output to `output.exr`.

```
pfsin␣input.exr␣|␣pfsfilter␣|␣pfsview
```

Read the image `input.exr`, apply the filter `pfsfilter` and show the result in an HDR image viewer.

```
pfsin␣in%04d.exr␣--frames␣100:2:200␣\
␣|␣pfsfilter␣|␣pfsout␣out%04d.hdr
```

Read the sequence of OpenEXR frames `in0100.exr`, `in0102.exr`, .., `in0200.exr`, apply the filter `pfsfilter` and write the result in Radiance's RGBE format to `out0000.hdr`, `out0001.hdr`, ...

---

pfstools is only a base set of tools which can be easily extended and integrated with other software. For example, *pfstools* is used to read, write and convert images and video frames for the prototype implementation of our image and video compression algorithms. HDR images can be rendered on existing displays using one of the several implemented tone mapping algorithms from the *pfstmo* package[2], which is build on top of *pfstools*. Using the software from the *pfscalibration* package[3], which is also based on *pfstools*,

[2] *pfstmo* home page: `http://www.mpii.mpg.de/resources/tmo/`

[3] *pfscalibration* home page: `http://www.mpii.mpg.de/resources/hdr/calibration/pfs.html`

cameras can be calibrated and images rescaled in physical or colorimetrical units. A computational model of the human visual system – *HDR-VDP*[4] – uses *pfstools* to read its input from multitude of image formats.

We created *pfstools* to fill the gap in the imaging software, which can seldom handle HDR images. We have found from the e-mails we received and the discussion group contacts that *pfstools* is used for high definition HDR video encoding, medical imaging, variety of tone mapping projects, texture manipulations and quality evaluation of CG rendering.

3 HDR Image and Video Compression

Wide acceptance of new imaging technology is hardly possible if there is no image and video content that the users could benefit from. The distribution of digital content is strongly limited if there is no efficient image and video compression and no standard file formats that software and hardware could recognize and read. In this section we propose several solutions to the problem of HDR image and video compression, including a color space for HDR pixels that is used as an extension to the MPEG-4 standard, and a backward-compatible HDR MPEG compression algorithm.

3.1 Color Space for HDR Pixels

Although the most natural representation of HDR images is a triple of floating point numbers, such representation does not lead to the best image or video compression ratios and adds complexity to compression algorithms. Moreover, since the existing image and video formats, such as MPEG-4 or JPEG2000, can encode only integer numbers, HDR pixels must be represented as integers in order to encode them using these formats. Therefore, it is highly desirable to convert HDR pixels from a triple of 32-bit floating point values, to integer numbers. Such integer encoding of luminance should take into account the limitations of human perception and the fact that the eye can see only limited numbers of luminance levels and colors. This section gives an overview of the color space that can efficiently represent HDR pixel values using only integer numbers and the minimal number of bits. More information on this color space can be found in (Mantiuk, Myszkowski & Seidel 2006).

Different applications may require different precision of the visual data. For example satellite imaging may require multi-spectral techniques to capture information that is not even visible to the human eye. However, for a

[4]*HDR-VDP* home page: `http://www.mpii.mpg.de/resources/hdr/vdp/index.html`

large number of applications it is sufficient if the human eye cannot notice any encoding artifacts. It is important to note that low dynamic range formats, like JPEG or a simple profile MPEG, can not represent the full range of colors that the eye can see. Although the quantization artifacts due to 8-bit discretization in those formats are hardly visible to our eyes, those encoding can represent only the fraction of the dynamic range and the color gamut that the eye can see.

Choice of the color space used for image or video compression has a great impact on the compression performance and the capabilities of the encoding format. To offer the best trade-off between compression efficiency and visual quality without imposing any assumptions on the display technology, we propose that the color space used for compression has the following properties:

1. The color space can encode the full color gamut and the full range of luminance that is visible to the human eye. This way the human eye, instead of the current imaging technology, defines the limits of such encoding.
2. A unit distance in the color space correlates with the Just Noticeable Difference (JND). This offers a more uniform distribution of distortions across an image and simplifies control over distortions for lossy compression algorithms.
3. Only positive integer values are used to encode luminance and color. Integer representation simplifies and improves image and video compression.
4. A half-unit distance in the color space is below 1 JND. If this condition is met, the quantization errors due to rounding to integer numbers are not visible.
5. The correlation between color channels should be minimal. If color channels are correlated, the same information is encoded twice, which worsens the compression performance.
6. There is a direct relation between the encoded integer values and the photometrically calibrated XYZ color values.

There are several color spaces that already meet some of the above requirements, but there is no color space that accommodates them all. For example, the Euclidean distance in the *CIE* $L^*u^*v^*$ color space correlates with the JND (Property 2), but this color space does not generalize to the full range of visible luminance levels, ranging from scotopic light levels, to very bright photopic conditions. Several perceptually uniform quantization strategies have been proposed (Sezan, Yip & Daly 1987, Lubin & Pica 1991), including the grayscale standard display function from the DICOM standard (DICOM PS 3-2004 2004). However, none of these take into account as broad dynamic range and diversified luminance conditions as required by Property 1.

Most of the traditional image or video formats use so called *gamma correction* to convert luminance or RGB tristimulus values into integer numbers, which can be latter encoded. Gamma correction is usually given in a form of a power function $intensity = signal^{\gamma}$ (or $signal = intensity^{(1/\gamma)}$ for an inverse gamma correction), where the value of γ is typically around 2.2. Gamma correction was originally intended to reduce camera noise and to control the current of the electron beam in CRT monitors. Further details on gamma correction can be found in (Poynton 2003). Accidentally, light *intensity* values, after being converted into *signal* using the inverse gamma correction formula, correspond usually well with our perception of lightness. Therefore such values are also well suited for image encoding since the distortions caused by image compression are equally distributed across the whole scale of *signal* values. In other words, altering *signal* by the same amount for both small values and large values of a signal should result in the same magnitude of visible changes. Unfortunately, this is only true for a limited range of luminance values, usually within a range from 0.1 to 100 cd/m^2. This is because the response characteristics of the human visual system (HVS) to luminance[5] changes considerably above 100 cd/m^2. This is especially noticeable for HDR images, which can span the luminance range from 10^{-5} to 10^{10} cd/m^2. An ordinary gamma correction is not sufficient in such case and a more elaborate model of luminance perception is needed. This problem is solved by the *JND* encoding, described in this section.

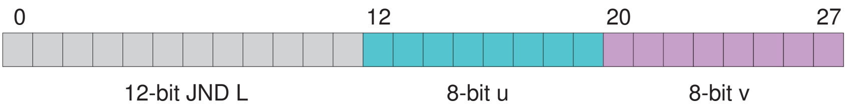

Fig. 2: 28-bit per pixel JND encoding

JND encoding can be regarded as an extension of gamma correction to HDR pixel values. The name *JND* encoding is motivated by its design, which makes the encoded values correspond to the Just Noticeable Differences (JND) of luminance.

JND encoding requires two bytes to represent color and 12 bits to encode luminance (see Figure 2). Chroma (hue and saturation) is represented using u' and v' chromacities as recommended by CIE 1976 Uniform Chromacity Scales (UCS) diagram and defined by equations:

$$u' = \frac{4X}{X + 15Y + 3Z} \tag{1}$$

$$v' = \frac{9Y}{X + 15Y + 3Z} \tag{2}$$

[5] HVS use both types of photoreceptors, cones and rods, in the range of luminance aproximately from 0.1 to 100 cd/m^2. Above 100 cd/m^2 only cones contribute to the visual response.

Luma, l, is found from absolute luminance values, y $[cd/m^2]$, using the formula:

$$l_{hdr}(y) = \begin{cases} a \cdot y & \text{if } y < y_l \\ b \cdot y^c + d & \text{if } y_l \leq y < y_h \\ e \cdot \log(y) + f & \text{if } y \geq y_h \end{cases} \qquad (3)$$

There is also a formula for the inverse conversion, from 12-bit luma to luminance:

$$y(l_{hdr}) = \begin{cases} a' \cdot l_{hdr} & \text{if } l_{hdr} < l_l \\ b'(l_{hdr} + d')^{c'} & \text{if } l_l \leq l_{hdr} < l_h \\ e' \cdot \exp(f' \cdot l_{hdr}) & \text{if } l_{hdr} \geq l_h \end{cases} \qquad (4)$$

The constants are given in the table below:

$a = 17.554$	$e = 209.16$	$a' = 0.056968$	$e' = 32.994$
$b = 826.81$	$f = -731.28$	$b' = 7.3014e-30$	$f' = 0.0047811$
$c = 0.10013$	$y_l = 5.6046$	$c' = 9.9872$	$l_l = 98.381$
$d = -884.17$	$y_h = 10469$	$d' = 884.17$	$l_h = 1204.7$

The above formulas have been derived from the psychophysical measurements of the luminance detection thresholds[6]. To meet our initial requirements for HDR color space, in particular Property 4, the derived formulas guarantee that the same difference of values l, regardless whether in bright or in dark region, corresponds to the same visible difference. Neither luminance nor the logarithm of luminance has this property, since the response of the human visual system to luminance is complex and non-linear. The values of l lay in the range from 0 to 4095 (12 bit integer) for the corresponding luminance values from 10^{-5} to 10^{10} cd/m^2, which is the range of luminance that the human eye can effectively see (although the values above 10^6 can be damaging to the eye and would mostly be useful for representing the luminance of bright light sources).

Function $l(y)$ (Equation 3) is plotted in Figure 3 and labelled "*JND* encoding". Note that both the formula and the shape of the *JND* encoding is very similar to the nonlinearity (gamma correction) used in the sRGB color space. Both *JND* encoding and sRGB nonlinearity follow similar curve on the plot, but the *JND* encoding is more conservative (a steeper curve means that a luminance range is projected on a larger number of discrete luma values, V, thus lowering quantization errors). However, the sRGB non-linearity results in a too steep function for luminance above 100 cd/m^2, which requires too many bits to encode real-world luminance values.

The color space described in this section can be directly used for many existing image and video compression formats, such as JPEG-2000 and MPEG-4.

[6]The full derivation of this function can be found in (Mantiuk, Myszkowski & Seidel 2006). The formulas are derived from the threshold versus intensity characteristic measured for human subjects and fitted to the analytical model (CIE 1981).

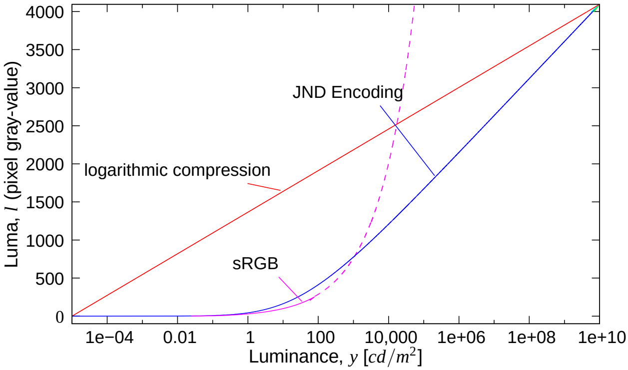

Fig. 3: Functions mapping physical luminance y to encoded luma values l. JND Encoding – perceptual encoding of luminance; sRGB – nonlinearity (gamma correction) used for the sRGB color space; logarithmic compression – logarithm of luminance, rescaled to 12-bit integer range. Note that encoding high luminance values using the sRGB nonlinearity (dashed line) would require significantly larger number of bits than the perceptual encoding.

Both these formats can encode luminance with 12 or more bits, which make them fully capable of representing HDR pixel values. As a proof of concept we extended an MPEG-4 compression algorithm to use the proposed color space. The modified video encoder achieved good compression performance, offering the ability to store the full color gamut and the range of luminance that is visible to the human eye (Mantiuk, Krawczyk, Myszkowski & Seidel 2004), as demonstrated in Figure 4. Moreover, the advanced HDR video player, which we created for the purpose of playback of HDR movies, can play video and apply one from several available tone-mapping algorithms in real-time (Krawczyk, Myszkowski & Seidel 2005). The additional advantage of HDR content is the possibility to simulate on traditional displays the perceptual effects that are normally only evoked when observing scenes of large contrast and luminance range. An examples of such effects are the night vision and an optically accurate motion blur, demonstrated in Figure 5. More examples can be found at the project page: `http://www.mpi-inf.mpg.de/resources/hdrvideo/index.html`.

The application of the proposed color space is not limited to image and video encoding. Since the color space is approximately perceptually uniform (Property 2), it can be used as a color difference metric for HDR images, similarly as the *CIE* $L^*u^*v^*$ color space is commonly used for traditional images. The luminance coding can also approximate photoreceptor response to light in the computational models of the human visual system (Mantiuk, Myszkowski & Seidel 2006). Since the proposed color encoding minimizes the number of bits required to represent color and at the same time does not

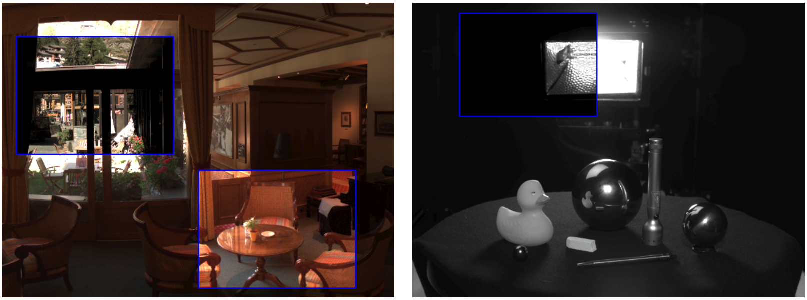

Fig. 4: Two screenshots from the advanced HDR video player, showing an extreme dynamic range captured withing HDR video sequences. Blue frames represent virtual filters that adjust exposure in the selected regions.

Fig. 5: Screenshots demonstrating simulation of perceptual and optical effects, possible only for HDR content. Left: simulation of night vision, resulting in a limited color vision and bluish cast of colors. Right: simulation of physically accurate motion blur (right side) compared with the motion blur computed from the traditional video material (left side).

compromise visual quality, it can be an attractive method of encoding data transmitted digitally from the CPU to a graphics card or from the graphics card to a display device.

3.2 Backward-compatible HDR Video Compression

Since the traditional, low-dynamic range (LDR) file formats for images and video, such as JPEG or MPEG, have become widely adapted standards, supported by almost all software and hardware equipment dealing with digital imaging, it cannot be expected that these formats will be immediately replaced with their HDR counterparts. To facilitate transition from the traditional to HDR imaging, there is a need for backward compatible HDR formats, that would be fully compatible with existing LDR formats and at the same time would support enhanced dynamic range and color gamut.

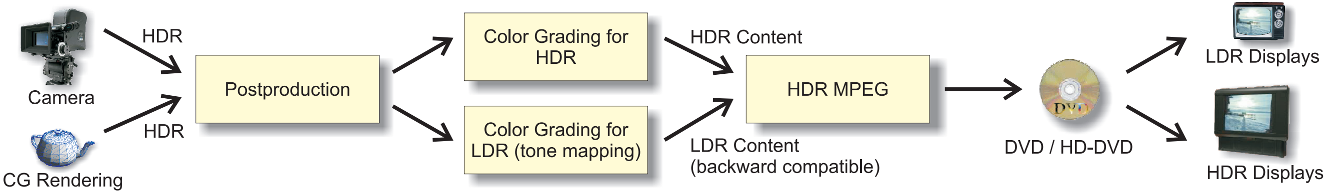

Fig. 6: The proposed backward compatible HDR DVD movie processing pipeline. The high dynamic range content, provided by advanced cameras and CG rendering, is encoded in addition to the low dynamic range (LDR) content in the video stream. The files compressed with the proposed HDR MPEG method can play on traditional LDR and future generation HDR displays.

Encoding movies in HDR format is attractive for cinematography, especially that movies are already shot with high-end cameras, both analog and digital, that can capture much higher dynamic range than typical MPEG compression can store. To encode cinema movies using traditional MPEG compression, the movie must undergo processing called color grading. Part of this process is the adjustment of tones (tone-mapping) and colors (gamut-mapping), so that they can be displayed on majority of TV sets (refer to Figure 6). Although such processing can produce high quality content for typical CRT and LCD displays, the high quality information, from which advanced HDR displays could benefit, is lost. To address this problem, the proposed HDR-MPEG encoding can compress both LDR and HDR into the same backward compatible movie file (see Figure 6). Depending on the capabilities of the display and playback hardware or software, either LDR or HDR content is displayed. This way HDR content can be added to the video stream at the moderate cost of about 30% of the LDR stream size. Because of such small overhead, both standard-definition and high-definition (HD) movies can fit in their original storage medium when encoded with HDR information.

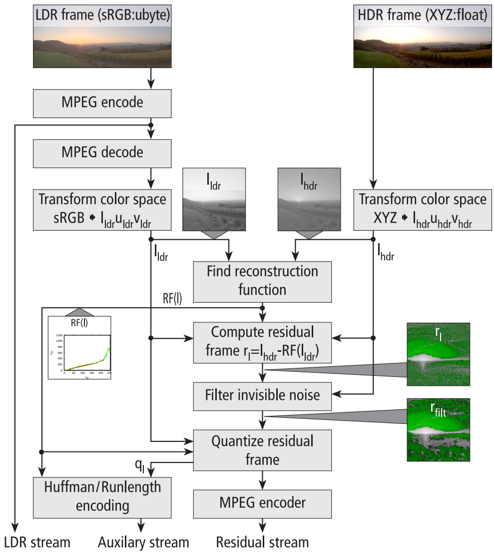

Fig. 7: A data flow of the backward compatible HDR MPEG encoding.

The complete data flow of the proposed backward compatible HDR video compression algorithm is shown in Figure 7. The encoder takes two sequences of HDR and LDR frames as input. The LDR frames, intended for LDR devices, usually contain a tone mapped or gamut mapped version of the original HDR sequence. The LDR frames are compressed using a standard MPEG encoder (*MPEG encode* in Figure 7) to produce a backward compatible LDR stream. The LDR frames are then decoded to obtain a distorted (due to lossy compression) LDR sequence, which is later used as a reference for the HDR frames (see *MPEG decode* in Figure 7). Both the LDR and HDR frames are then converted to compatible color spaces, which minimize differences between LDR and HDR colors. The reconstruction function (see *Find reconstruction function* in Figure 7) reduces the correlation between LDR and HDR pixels by giving the best prediction of HDR pixels based on the values of LDR pixels. The residual frame is introduced to store a difference between the original HDR values and the values predicted by the reconstruction function. To further improve compression, invisible luminance and chrominance variations are removed from the residual frame (see *Filter invisible noise* in Figure 7). Such filtering simulates the visual processing that is performed by the retina in order to estimate the contrast detection threshold at which the eye does not see any differences. The contrast magnitudes that are below this threshold are set to zero. Finally, the pixel values of a residual frame are quantized (see *Quantize residual frame* in Figure 7) and compressed using a standard MPEG encoder into a residual stream. Both the reconstruction function and the quantization factors are compressed using a lossless arithmetic encoding and stored in an auxiliary stream.

This subsection is intended to give only an overview of the compression algorithm. Further details can be found in (Mantiuk, Efremov, Myszkowski & Seidel 2006a) or (Mantiuk, Efremov, Myszkowski & Seidel 2006b) and on the project web page: `http://www.mpii.mpg.de/resources/hdr/hdrmpeg/`.

We implemented and tested a dual video stream encoding for the purpose of a backward compatible HDR encoding, however, we believe that other applications that require encoding multiple streams can partly or fully benefit from the proposed method. For example, a movie could contain a separate video stream for color blind people. Such a stream could be efficiently encoded because of its high correlation with the original color stream. Movie producers commonly target different audiences with different color appearance (for example *Kill Bill 2* was screened with a different color stylization in Japan). The proposed algorithm could be easily extended so that several color stylized movies could be stored on a single DVD. This work is also a step towards an efficient encoding of multiple viewpoint video, required for 3D video (Matusik & Pfister 2004).

4 Conclusions

In this paper we introduce the concept of HDR imaging, pointing out its advantages over the traditional digital imaging. We describe our implementation of the image processing software that operates on HDR images and offers flexibility necessary for research purposes. We believe that the key issue that needs to be resolved to enable wide acceptance of HDRI is efficient image and video compression of HDR content. We address the compression issues by deriving a perceptually-motivated HDR color space capable of encoding the entire dynamic range and color gamut visible to the human eye. We propose also two compression algorithms, one being a straightforward extension of the existing MPEG standard, and the other offering backward compatibility with traditional video content and equipment. The proposed backward-compatible algorithm facilitates a smooth transition from the traditional to high-fidelity HDR DVD content.

In our work we try to realize the concept of an imaging framework that would not be restricted by any particular imaging technology and, if storage efficiency is required, be limited only by the capabilities of the human visual system. If the traditional imaging is strongly dependent on the particular technology (e.g. primaries of color spaces based on the red, green and blue phosphor in CRT displays), HDRI can offer an image-independent representation of images and video. However, redesigning existing imaging software and hardware to work with HDR content requires a lot of effort and definition of new imaging standards. Our mission is to popularize the concept of HDR imaging, develop standard tools and algorithms for processing HDR content and research the aspects of human perception that have key influence on digital imaging.

Acknowledgements

I would like to thank my advisors, Karol Myszkowski and Hans-Peter Seidel, for supporting my work on HDRI. Special thanks go to Grzegorz Krawczyk and Alexander Efremov for their work on the HDR video compression projects.

References

CIE (1981). *An Analytical Model for Describing the Influence of Lighting Parameters Upon Visual Performance*, Vol. 1. Technical Foundations, CIE 19/2.1, International Organization for Standardization.

DICOM PS 3-2004 (2004). Part 14: Grayscale standard display function, *Digital Imaging and Communications in Medicine (DICOM)*, National Electrical Manufacturers Association. **URL:** *http://medical.nema.org/dicom/2004.html*

Krawczyk, G., Myszkowski, K. & Seidel, H.-P. (2005). Perceptual effects in real-time tone mapping, *SCCG '05: Proc. of the 21st Spring Conference on Computer Graphics*, pp. 195–202.

Lubin, J. & Pica, A. (1991). A non-uniform quantizer matched to the human visual performance, *Society of Information Display Int. Symposium Technical Digest of Papers* (22): 619–622.

Mantiuk, R., Efremov, A., Myszkowski, K. & Seidel, H.-P. (2006a). Backward compatible high dynamic range mpeg video compression, *ACM Transactions on Graphics* **25**(3).

Mantiuk, R., Efremov, A., Myszkowski, K. & Seidel, H.-P. (2006b). Design and evaluation of backward compatible high dynamic range video compression, *MPI Technical Report MPI-I-2006-4-001*, Max Planck Institute für Informatik.

Mantiuk, R., Krawczyk, G., Myszkowski, K. & Seidel, H.-P. (2004). Perception-motivated high dynamic range video encoding, *ACM Transactions on Graphics* **23**(3): 730–738.

Mantiuk, R., Myszkowski, K. & Seidel, H.-P. (2006). Lossy compression of high dynamic range images and video, *Proc. of Human Vision and Electronic Imaging XI*, Vol. 6057 of *Proceedings of SPIE*, SPIE, San Jose, USA, p. 60570V.

Matusik, W. & Pfister, H. (2004). 3D TV: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes, *ACM Trans. on Graph.* **23**(3): 814–824.

Poynton, C. (2003). *Digital Video and HDTV: Algorithms and Interfaces*, Morgan Kaufmann.

Seetzen, H., Heidrich, W., Stuerzlinger, W., Ward, G., Whitehead, L., Trentacoste, M., Ghosh, A. & Vorozcovs, A. (2004). High dynamic range display systems, *ACM Trans. on Graph.* **23**(3): 757–765.

Sezan, M., Yip, K. & Daly, S. (1987). Uniform perceptual quantization: Applications to digital radiography, *IEEE Transactions on Systems, Man, and Cybernetics* **17**(4): 622–634.