HDR SYSTEM README

May 2015

ASC Technology Committee, ASC UHD Committee

Single Master HDR

It is believed by our group that it possible to have a single mastered appearance for High Dynamic Range (HDR), and thus a single master. It is recognized that there will be some differing ranges (both bright and dark) and color gamuts in receiving displays and projectors. Despite these differences, it seems possible to create a single master containing the aesthetic intent, and convey a large portion of that mastered intent to every properly-calibrated display.

A key to doing this is felt to be the division of calibration and aesthetic adjustment. Calibration should rely on spectral radiometers, such that the entire end-to-end production chain (from camera to final viewer) can lock down colors (as chromaticities and associated spectra).

It is realized that aesthetic appearance adjustment is required in many places in the production chain. These are for good reason. However, each such stage of aesthetic adjustment should be a delta from the radiometric calibration, such that its nature, purpose, and behavior, can be easily understood.

In this way, we believe that a Single Master HDR architecture is eminently feasible, and that it should be top priority going forward.

HDR System Intended Range, Gamut, and Purpose

The HDR System is a test and exploration code suite intended to directly address aesthetic and technical issues. The initial aim is to gain aesthetic control over HDR masters, and provide tools to preserve the HDR image appearance over a population of displays having varying capabilities. The initial intended brightness range spans 200nits to 1000nits. Explorations below 200nits are of interest, but felt to be overly challenging at this stage. Explorations above 1000nits are also of interest, but there is some concern that such high brightnesses raise significant worries with respect to viewing discomfort. Both of higher and lower brightness extensions are deemed best left to a future time when/if mastery has been gained over the 200-1000nit range.

In a similar way, the initial HDR System gamut will be limited to P3 color primaries (the DCinema "minimum color gamut").

It is felt that by limiting range and gamut that the likelihood of success is greatly improved.

However, despite these current limitations (for practicality and expedience), the HDR System is not limited technically in gamut nor range by its present architecture.

Location of the HDR Master

In the HDR System, the HDR master resides prior to the rgb-ratio-preserving rendering "nugget". The HDR master is logically scene-light-referred, even if graded and aesthetically adjusted. The HDR System ACES-RGB exr example files can be used directly at this location, without any adjustment, to yield a high quality appearance on supported display types. The choice of a scene-light-referred master is an open topic, with the possibility of the output of the rendering nugget also being a candidate as the HDR master. However, the function of the rendering nugget is to apply the "system gamma boost" that is typical in imaging systems (including film). It is felt that such a system gamma boost is not as well understood for HDR as is the meaning and integrity of scene-referred light. Essentially any camera that can output ACES-rgb exr files is directly providing scene-referred light (usually with substantial dynamic range). This provides a direct connection to radiometric principles, which are the basis of this HDR system.

Display-type masters, such as Rec709 gamma 2.4, P3 with PQ_gamma, and similar, are display-type specific, and are thus not by nature display-type independent. Further, the HDR System Master, although it may be range limited to such display types in the actual range and gamut that is used, should not by its architecture have such a limitation built into its architectural design. Tools for transforming display-type masters into the HDR System Master are provided (the inversion tool), but such masters are by nature limited to the source display-type's range and gamut.

It is worth noting that the example scenes have colors well outside of P3 gamut. This can help test and evaluate such issues with respect to the HDR System tools. Such colors are preserved by the rendering nugget, but are lost with P3 or Rec709 display type outputs. It is the intent of the rendering nugget architecture to be without range limitation in range or gamut. Thus, the stage is properly set for adding new device types beyond P3 gamut in the future.

Relationship to ACES

The Academy Color Encoding System (ACES) was originally named the Image Interchange Framework (IIF). The Academy Color Encoding Space (ACES) was the original meaning of the ACES acronym. This name change, and duplicate meaning to the term ACES, requires some clarification. The Academy Color Encoding Space will be referred to herein as ACES RGB. It is supported as OpenExr half-float files (which pre-date ACES and IIF). The half-float was originally developed by NVidia and ILM, and has been available for over a decade in the OpenExr system (see openexr.com). ACES RGB uses CIE 1931 xy chromaticity-defined red green and blue primaries (with green and blue being “virtual” non-physical primaries).

Although the HDR System has its development history interlinked with ACES/IIF development, and owes many of its ideas to the work of ACES participants, the HDR System is not ACES, differing from ACES 1.0 in substantial ways. However, ACES 1.0 interoperability is available through display-type interoperability (e.g. P3_D60 gamma 2.6), as with any workflow process.

HDR System Primaries

The HDR System rendering nugget preserves R, G, and B relative ratios without requiring that RGB be ACES primaries. Although the HDR System currently uses ACES primaries for interoperation with ACES RGB exr files, the rendering nugget could just as easily use XYZ, BT2020 RGB linear, or other sets of primaries. However, the display-type transforms require specific knowledge of the color primaries for their color transform matrix. Further, optional aesthetic-rendering preambles have been “tuned” to work with ACES RGB. Such preambles would require the addition of to/from ACES RGB matrix transforms in order to function properly is alternate primaries (e.g. XYZ or BT2020 RGB). It is not recommended that P3 primaries be used in the rendering nugget, since crossing the P3 gamut boundary yields unwieldy negative numbers which are problematic with some display-type matrices. To summarize, the HDR System currently uses ACES RGB, but could be re-architected to use XYZ, BT2020 RGB linear, or other similar wide-gamut linear representations. With XYZ there should be no loss of any physically-realizable colors. With BT2020 RGB linear, the BT2020 gamut boundary remains, especially for cyan colors.

HDR System Whitepoint

It should be noted that ACES RGB has set R=G=B=D60 white. To many people, D60 is aesthetically preferable to D65. The HDR System is currently architected at the D60 whitepoint. However, use of D60 limits display-type top-end range of D65 system blue by 10% (much less than 10% in gamma 2.4 values, being about 4%). This should not be an issue in distributing to HDR displays having a D65 whitepoint. However, it is an issue of mastering to an HDR display in P3 or Rec709 having a D65 whitepoint, with the intent of using that master in the HDR System (via inversion) with the D60 whitepoint. In order to preserve the colors having bright blue in this scenario, care must be taken not to use the top 10% (linear light) of the blue range when mastering on D65 displays in this way. Mastering on D60 displays has no such limitation. Repurposing and interoperation are affected by this issue, but looking forward to a new HDR system, this should not be an issue (since viable HDR mastering displays will all support D60). The luminance reduction from a D60 whitepoint for HDR is less than 1% (10% of the blue component of luminance) on D65 whitepoint HDR displays.

Note that the HDR System could be re-architected to R=G=B=D65 using ACES primary chromaticities, or BT2020 R=G=B=D65, or to X=Y=Z=E (chromaticity .3333 .3333), or other useful whitepoints. However, at present, it is felt best to retain the HDR System whitepoint of R=G=B=D60 using ACES RGB primaries.

It is worth noting that there is nothing within the HDR System itself that is limited by any whitepoint, since the HDR System processing is entirely floating point. However, any display-type transform, as just described, is affected.

There is an inherent tension between R=G=B maximum white and aesthetic white. As an example, consider the R=G=B=white as a calibration of the “minimum P3 gamut” for DCinema. This greenish whitepoint was chosen over nearby D60 and D61, despite many objections to the greenish offset. Despite assurances that this greenish whitepoint would not be a problem, in practice it has been continuously problematic, and remains so. Thus, the aesthetic white and the calibration R=G=B whitepoint are intertwined in practice, even if not in theory.

It is currently unclear whether laser primaries, or other narrow-spectrum primaries, are a significant consideration for the HDR System. If so, it should be noted that existing definitions of whitepoint, and most tools for measuring whitepoint, are not specified properly for calibration. Further, even if properly calibrated, narrow-spectrum primaries do not lend themselves to color appearance accuracy. It may be best to defer HDR System exploration of narrow-spectrum primary display types (although the HDR System currently encompasses the gamut from a technical standpoint by using ACES RGB).

A Note About HDR Mastering

When mastering, the “colorfulness” of brighter portions of the scene, both saturated and desaturated, will be a function of absolute brightness. Many displays that are useful for HDR mastering, including the Dolby PRM in “dynamic” mode, and the Sony BVM-X300, will decrease overall brightness based upon the proportion of the screen area that is bright. The maximum brightness will be reduced for 100% screen area being white, vs. 1%, or 10% of the screen area being white. This reduction can be a factor of two or more. This may be aesthetically desirable, but the exact behavior must somehow be characterized if so. It may be necessary to check a mastering grade against a display-type that does not have the brightness reduction (such as a DLP projection on a very small area screen, thus being very bright). It also should be noted that dimmer displays may not have this large-bright-region darkening behavior. A 200nit display at the low end of the range may be sufficient to judge appearance for bright regions of the image.

It is also a good practice to check the color saturation appearance of bright screen regions on the dimmest display in the supported range. For example, a display set to 200nits might serve as such a reference display for bright-region "colorfulness" appearance. If the 200nit display has properly adjusted "colorfulness" for bright regions (both saturated and desaturated colorfulness), then brighter displays (e.g. 600nits) will appear too colorful (even if at the same measured chromaticity). This is the correct approach, although some discipline is required to work this way (which has no historical precedent). Some amount of bright region desaturation can be added to the bright display's bright regions for judging the appearance (e.g. using a low value of the `MDR_RADIOMETRIC_PROPORTION or HDR_RADIOMETRIC_PROPORTION`, as appropriate). The `MDR_RADIOMETRIC_PROPORTION or HDR_RADIOMETRIC_PROPORTION,` as appropriate, should be set to 1.0 when viewing, judging, and setting the 200nit bright region "colorfulness". When bright scene regions have proper appearance on the 200nit display, then the additional desaturation required for brighter displays will function properly.

To summarize, a calibrated display at the dim end of the HDR range (e.g. 200nits) should be available to serve as a reference for bright-scene-region "colorfulness" and appearance, using a `MDR_RADIOMETRIC_PROPORTION or HDR_RADIOMETRIC_PROPORTION,` as appropriate, set to 1.0 (thus yielding a radiometrically-calibrated HDR master). Brighter HDR displays can be desaturated from this bright region behavior (e.g. by reducing the radiometric proportion to a small value such as 0.1 or 0.2) for judging appearance. Some exploration of how best to do this will help work out the details.

Note that the principle of "colorfulness" requires that any single HDR master not attempt to achieve maximum color saturation of bright regions that may be possible on the brightest HDR display. The single HDR master concept requires that the bright region colorfulness be judged on the dimmest (e.g. 200nit) calibrated display, and desaturated as required to match equivalent "colorfulness" appearance on the brightest display (e.g. 1000nits), and proportionally at all brightnesses in between (e.g. 300-800nits).

An HDR master using full "colorfulness" on the brightest HDR display will necessarily appear less colorful on dimmer displays, and thus is not consistent with the creation of a single HDR master.

HDR System Contents

Optional Available Test Scenes (4k ACES RGB exr half-float):

Note that these scenes have been pre-processed with noise reduction and sharpening using Image Essence LLC HDR preprocessing tools. JKP_Alps involved exposure and rgb gain balance and cross dissolves. ICAS scenes are not adjusted for exposure nor rgb balance.

• JKP_Alps, NikonD800E timelapse, and F55 (Courtesy of Joe Kane; DP: Florian Friedrich)

• ICAS Diner F65 (Courtesy of the ASC and Revelation Entertainment; DP: Bob Primes ASC)

• ICAS Night F65 (Courtesy of the ASC and Revelation Entertainment; DP: Fred Goodich ASC)

There is a plan to add scenes.

Code Modules:

• HDR_CPU.cpp: file-in/file-out code. Input files are exr half-float ACES RGB. Output files are exr or dpx, either with display-ready values. OpenExr must be installed to build. This code is casually supported (no guarantees) by Gary Demos. This code is an example only (i.e. use at your own risk, do not use for production, do not put this code into any products).

• HDR_CPU_INVERSE.cpp: file-in/file-out code. Input files are dpx display-ready values. Output files are exr half-float ACES RGB. This inversion code is experimental and unsupported (i.e. use at your own risk, do not use for production, do not put this code into any products).

• hdr_macros.h: Contains macros to process the rendering nugget and MDR Rec709 or HDR P3_D60. This file can be optionally included by OpenCL at runtime to process in the GPU (see next), or included in the .cpp files above during compile. The rendering nugget and MDR/HDR display-type transform code is casually supported (no guarantees) by Gary Demos. All aesthetic rendering preambles and interaction code is unsupported. This code is an example only (i.e. use at your own risk, do not use for production, do not put this code into any products).

• HDR_GPU_CL_GL_LINUX: Read exr half-float ACES RGB into memory, and display to the X11 screen using CL/GL interop. Binary-only optionally available on linux systems (requires also .cl file) in the GPU. Must be hand-installed by Gary Demos as a binary, source code not provided. The hand-install may not be successful on some linux and gpu configurations, no guarantee. Note, on NVidia GPU's, the cl file must continuously change to force re-interpret (stale cache workaround) of the hdr_macros.h file if making changes. This code is unsupported (i.e. use at your own risk, do not use for production, do not put this code into any products).

NOTE: No information is used in dpx nor exr headers beyond that sufficient to indicate resolution, pixel locations, and endian-ness. Such information is ignored when reading, and no attempt is made to place meaningful information into headers during writing. File types are implicit due to use context, but not due to any additional nor header information. See code for details with respect to dpx, or see OpenExr *documentation with respect to exr header defaults (which are ignored, no reliance should be placed on any meaning within any OpenExr file header).*

Use and Exploration of HDR System

Rendering Nugget

The purpose of the rendering nugget is to apply a "system gamma" while retaining the relative ratios of r, g, and b. By preserving the relative ratios, chromaticity is maintained (independently of how chromaticity is defined). The way that this is done is by utilizing a norm function of r, g, and b to create a single number that can be utilized to achieve the system gamma. The system gamma is not an exponent, but rather a curve function. It is commonly called a "tone curve", although the word "tone" isn't very descriptive. In motion picture film, there is a mid-brightness (gray) gamma boost, centered around the Laboratory Aim Density (LAD). LAD is defined as 18% grey in the scene, which maps to 10% transmittance (1.0 density) on the print film. This is the key design point for the tone curve, wherein .18 maps to .10. The mid-grey .10 will likely need to lift for higher brightness presentations. Thus, this mapping is not strict.

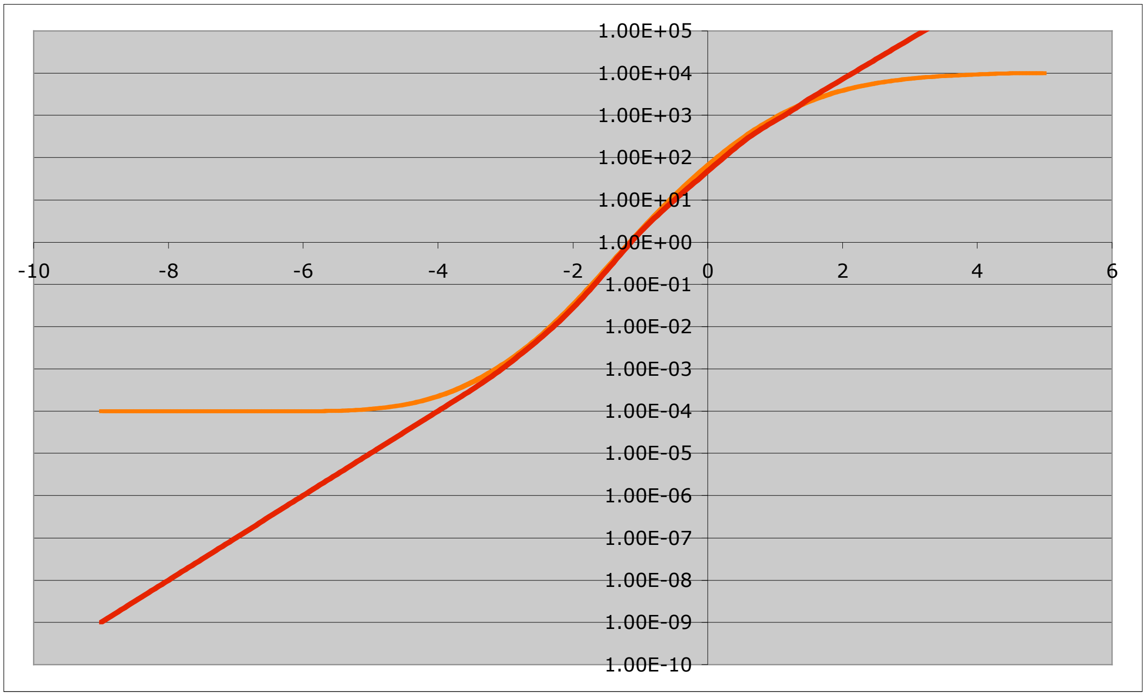
On film there is an "S-Curve" on both print and negative. Negative captures a broader dynamic range than is reproduced on the print, but the print magnifies the scene contrast via the system gamma. The scene-dark behavior of film is such that colors desaturate. This is undesirable as a forced system adjustment for HDR, but may be desirable as an aesthetic grading attribute. Due to this, the bottom part of the tone curve in the rendering nugget has S-Curve behavior to some degree, but does not desaturate. The slope flattens currently to .02, which goes through 0.0 on through to negative values with the same .02 slope. Negative values are a bit experimental, and are not smoothly continuous near zero (there are slope discontinuities in computation of the norm). RGB values are clipped up to zero after the rendering nugget process.

The top of the S-Curve on film is a desaturation as the print film approaches clear. If R, G, and B separately "ride" the S-Curve top brightness, then red will flatten out first, then green, and then blue. Thus, bright saturated reds will gradually go orange, then yellow, then white. Although this is common behavior, it is clearly not always desirable, and there is no direct analogy with HDR.

Thus, the top of the tone curve in the rendering nugget is slope 1.0, such that out_value = in_value for bright norm values. Thus, there is no S-Curve top whatsoever in the rendering nugget.

Using the norm lookup/interpolate through the tone curve, a value is obtained that is used in ratio to the norm to scale R, G, and B. Thus, this single scale factor will inherently preserve R, G, and B ratios (and thus chromaticities).

Here is a logarithmic(horizontal) vs linear(vertical) plot of the current HDR System tone curve (in red) vs. ACES 0.2's rrt tone curve (in orange).


1.00E+05
1.00E+04
1.00E+03
1.00E+02
1.00E+01
1.00E+00
1.00E-01
1.00E-02
1.00E-03
1.00E-04
1.00E-05
1.00E-06
1.00E-07
1.00E-08
1.00E-09
1.00E-10
-10
-8
-6
-4
-2
0
2
4
6

Aesthetic Rendering Preambles

The Aesthetic Rendering Preambles change appearance, and do not preserve R, G, B relative ratios (and thus do not preserve chromaticity). These various optional preambles attempt to modify the look of the image to create a pleasing appearance. However, there are a number of configurations of these preambles to fit different scene types. There are also varying levels of complexity, as reflected in the names of the preambles. The mastered appearance in the HDR System is the output of any one (or none) of these preambles, but is logically prior to the rendering nugget.

The aesthetic rendering preambles are for testing use by way of example, and are not supported, and are not intended to be nailed down in the HDR System. Note that color grading can be in series with these preambles, or use none of them. It makes no difference. It is the net result, in the logical position prior to the rendering nugget, that represents the intended HDR appearance. The nugget is required, as is some display-type transform, for mastering to judge appearance. Limitations in the mastering display (or projector) must be considered, but there are several high quality mastering-grade displays available in the 600nit to 1000nit range.

The configurations are controlled by the following:

```
/* select one of the following: */
#define BYPASS_LMT /* this selects solely the rendering nugget */
//#define SIMPLE_LMT
//#define GAMMA_AND_MAT
//#define MODERATE_LMT
//#define FULL_LMT
//#define DOUBLE_LMT


#define BRIGHT_HIGHLIGHTS_IN_TONE_CURVE /* if defined: use bright
highlights, if not defined: use lower highlights */
//#define FACES_HIGHLIGHTS /* if defined: desaturate highlights for
faces, if not defined: more color saturation in bright colors (does not
apply to BYPASS_LMT nor GAMMA_AND_MAT) */
```

The most useful configurations have been found to be:

1) the above (rendering nugget, full rgb relative ratio preservation)
2) FULL_LMT aesthetic rendering preamble to boost color saturation (complex)
3) FULL_LMT plus turn off `BRIGHT_HIGHLIGHTS_IN_TONE_CURVE but enable FACES_HIGHLIGHTS`. This configuration is useful when faces are prominent in the scene.

A useful nomenclature for including in tool names set in these three configurations are "ngt" for nugget, nothing for full, and nobr_fc for full+no_bright_highlights+faces.

Display-Type Transforms

The HDR System supports two example display-type transforms. These are selected on the command line (display type 1 for MDR Rec709 300nits and display type 2 for HDR P3_D60 700nits). The processing is based upon an asymptote, and is algorithmically identical for MDR and HDR. An asymptote such as x/(1+x) can be seen to leave x unchanged for low values, to be at .5 when x=1, and gradually approach 1.0 as x goes to infinity. The parameters vary from MDR to HDR, providing a template that any specific can use to match these parameters to the specific range. However, MDR and HDR can be varied with simple scaling within a range of about +- one stop (+- a factor of two) and still remain reasonable.

There are two sets of these transforms. One is the long-tested (five years HDR, two years MDR) set. This set desaturates fully at the bright end, like film's S-Curve top. The other set is activated by the flag `RADIOMETRIC_ODTS`. This flag switches to an alternate implementation. The alternate implementation removes a small saturation adjustment matrix at the start (probably not needed in this revised version of MDR HDR). However, otherwise the main difference is that `RADIOMETRIC_ODTS` activates an RGB-relative-ratio preserving process using a norm and the asymptote. The blend factors `MDR_RADIOMETRIC_PROPORTION and HDR_RADIOMETRIC_PROPORTION` control the amount that the RGB values desaturate at the bright end, vs. the amount that the RGB relative ratios are preserved. At 0.0, the asymptote desaturates fully (like film). This is similar to the MDR and HDR processing when `RADIOMETRIC_ODTS` is off. When `MDR_RADIOMETRIC_PROPORTION and HDR_RADIOMETRIC_PROPORTION` are set to 1.0, the RGB relative ratios are fully preserved. The blend factor allows some amount of each.

For system calibration, and for mastering, the `MDR_RADIOMETRIC_PROPORTION and HDR_RADIOMETRIC_PROPORTION` should be set to 1.0. See the note about HDR mastering above.

For use of HDR_CPU_INVERSE, the `MDR_RADIOMETRIC_PROPORTION and HDR_RADIOMETRIC_PROPORTION` should also be set to 1.0. However, any blend value from 0.0 to 1.0 is useful, with values near 0.5 being the most useful to preserve bright color saturation (both saturated and desaturated color saturation). More normal behavior on bright displays is to desaturate fully (0.0 for the blend proportion), or `MDR_RADIOMETRIC_PROPORTION and HDR_RADIOMETRIC_PROPORTION` both set off.

Note that once bright color (both saturated and desaturated colors) is desaturated, it usually cannot be recovered. If it is the intent that bright color always be desaturated, then this should be done as an aesthetic adjustment and baked into the master (the master is prior to the rendering nugget). However, brighter displays (e.g. 1000nits) will appear more colorful (the "Hunt effect") than dimmer displays (e.g. 200nits) even with the same measured chromaticities. Thus, the `MDR_RADIOMETRIC_PROPORTION and HDR_RADIOMETRIC_PROPORTION` are inherently tied to the specific presentation display, and its settings (due to the resulting absolute brightness only being known locally at that display).

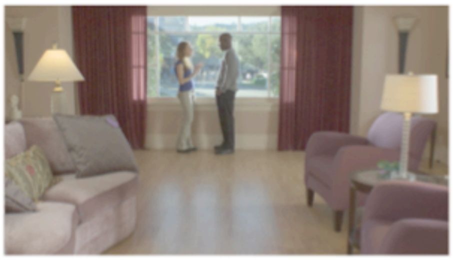
The other key parameters enabled by `RADIOMETRIC_ODTS` is the `ROOM_BRIGHTENING` parameter. This parameter is intended to adjust the image appearance for ambient room surround brightness. At 0.0, `ROOM_BRIGHTENING` is set for dark surround viewing. At 1.0, `ROOM_BRIGHTENING` is set for high room ambient light. Since `ROOM_BRIGHTENING` is only applied to the rgb-ratio preserving path, the rgb ratios are unaltered through this adjustment. Note that dark parts of a scene (including dark faces) appear darker in higher ambient surround vs dark surround. Maintaining appearance requires brightening those darker regions.

Since `ROOM_BRIGHTENING` is only applied to the rgb-ratio preserving section, its affect is scaled by the `MDR_RADIOMETRIC_PROPORTION and HDR_RADIOMETRIC_PROPORTION` . If these are 0.0, there will be no affect, and if they are 1.0, there will be a large affect. The intent is that some setting with `MDR_RADIOMETRIC_PROPORTION and HDR_RADIOMETRIC_PROPORTION` in the .1 to .5 range may be found that properly adjusts to viewing room ambient light.

As with colorfulness as a function of absolute brightness, the ambient surround is only knowable at each viewing display, and thus must be processed accordingly within each such display.

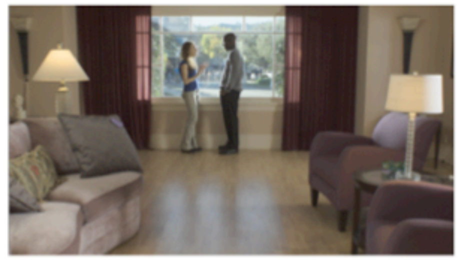
note: colors not accurately reproduced on slides (shown for approximate comparison)

Aesthetic Rendering Embodies Overall Gamma Boost



Scene-referred
Without Aesthetic Rendering
With P3 Gamma/Gamut
And Limited Display Range S-Curve

Similar to ACES P3_D60_ODT Only (no RRT)



Scene-referred
Through Faces Aesthetic Rendering
Then Via P3 Gamma/Gamut
and Limited Display Range S-Curve

Similar to ACES RRT+P3_D60_ODT

Faces Rendering
Minus Two Stops (Scene)

Faces Rendering
Nominal Exposure (Scene)

Faces Rendering
Plus Two Stops (Scene)

note: colors not accurately reproduced on slides (shown for approximate comparison)

Nugget (basic) Rendering
Nominal Exposure (Scene)

Nominal (non-face) Rendering
Nominal Exposure (Scene)

Alternate Display-Type Configurations

Comments in the code show potentially useful variations to the initial HDR and MDR display-type configurations. Example commented-out matrices are present for D60 to D65 conversion. There is currently no conversion matrix to BT2020 at either D60 nor D65, but hopefully that will be worked out in the near future, and added as a comment.

The gamma values are set at 2.4, but may be changed to any useful value (e.g. P3_D60 at gamma 2.6).

The flag `PQ_GAMMA` alternately selects pq_gamma instead of gamma 2.4. Note that `PQ_GAMMA` is a bit of a mystery, with unusual operations such as an exponent of `78.84375`. The use of `PQ_GAMMA` is not recommended without expert advice from someone who understands its function, and can advise as to whether `PQ_GAMMA` is being applied properly. It is not known whether there are adjustments in these formulae which are needed, nor their meaning and use if so. The `PQ_GAMMA` formulae were cribbed blindly from an ACES 1.0 HDR ODT.

Note: Display-type transforms are currently implemented to use full range. When using a gamma (vs pq_gamma), the 2.22 gamma with linear black toe is sometimes used instead of gamma 2.4 (although this distinction is confusing in BT1881). Reference monitors have long used 2.4, as measured by Adv Imaging with the BVM HD CRT. If SMPTE range were to be utilized, it should be noted that negative numbers are *potentially directly supportable, as are numbers above logical 1.0 (e.g. above 940 in 10bits). However, in such case care should be taken to ensure that such numbers have a clear meaning (via a corresponding precise specification of the numerical meaning of all values used).*

Check List For HDR_CPU

• Check that the hdr_macros.h file is set up as intended. In particular check flags such as `RADIOMETRIC_ODTS, PQ_GAMMA, MDR_DISPLAY_GAMMA, HDR_DISPLAY_GAMMA`, `MDR_RADIOMETRIC_PROPORTION and/or HDR_RADIOMETRIC_PROPORTION`, and `ROOM_BRIGHTENING`.

Also check that `BYPASS_LMT` is set, or one of the five optional preambles (which always implicitly include `BYPASS_LMT`) is set as intended.

• Compile HDR_CPU.cpp using by copy and paste of the build line at the beginning of the code. A gnu C++ compiler must be on the system, and OpenExr must be installed for this to build. There is no makefile.

• Make sure the input files are exr half-float in ACES RGB primaries (extension .exr)

• Select either dpx or exr half-float output (extensions .dpx or .exr)

• Use %07d in the name (for 7 digits with leading zeros) for the range of frames. If you are unfamiliar with this, ask a C programmer how this frame number structure is typically specified.

• Select the range of frames to process (start to end frame numbers)

• Select either 1 for MDR or 2 for HDR as the display-type

• Debug the path and file names if it doesn't start right up. There is little or no startup time needed. The files are 16-way multithreaded, may be problematic on processors having a small number of cores. Memory use is not excessive, but memory should have sufficient size for 4k workflows if working on 4k files (the test ACES files are all 4k).

• The output .exr or .dpx files have pixels ready to put up on the display_type specified.

Use and Exploration of the HDR_CPU_INVERSE

NOTE: The CPU Inverse is experimental and completely unsupported. It is meant to serve only as a possible example, and to enable testing providing some potentially useful interoperability at the current stage. The use of any dpx input to HDR_CPU_INVERSE will be limited in gamut, dynamic range, and bit depth due to *current practice limitations. These limitations will be present in the ACES RGB exr file that results (and may not have acceptable appearance).*

The HDR_CPU_INVERSE program is a fairly exotic and experimental piece of code. It operates by scattering random ACES RGB values forward through the rendering nugget and display-type transform. The resulting values then form an inverse map with holes in it. An ad-hoc algorithm is used to sweep paths of hole-filling using neighbors (especially darker neighbors). Although the result of this is a fully populated 3-Dimensional inverse lookup table (dpx to exr), there is no guarantee that the values at the bright, dark, and gamut extremes of this table are reasonable. In general there will be hard-clipping of all of these. Whether the limitations are acceptable is dependent upon the use case (i.e. dependent upon the contents of the images being processed).

Using the JKP test charts that cover full saturation, it has been experimentally found (for HDR P3_D60 display-type) that reasonable behavior falls under 1014/1023. Because of this, currently all dpx input values are hard-clipped to 1014 maximum. It has also been experimentally found that the brightest pure colors require some desaturation. This is due to the ACES to P3_D60 diagonal matrix terms in the HDR display-type transform. The amount of bright red desaturation required is substantial. The desaturation is achieved by adding some green and blue for large pure red. The desaturation required for other saturated colors is small (but necessary). Note that this desaturation is proportional to the magnitude, such that there is no desaturation at low values of pure colors.

The lookup table is 256 cubed (8bits RGB being 24bits of table). The low order 2bits of the 10-bit dpx RGB values are interpolated. Note that such interpolation could be extended usefully to higher bit-depths, although table accuracy limitations and noise-like properties may not warrant such extension.

Note that this algorithmic method does not care what configuration the rendering nugget, optional preambles, or display-type parameters are set. However, only one configuration of some of the parameters is the intended use. Parameters which may be varied and yet still remain within the intended uses include `PQ_GAMMA` , gamma value, and colors space matrix (e.g. D60 or D65, Rec709 or P3).

The parameters which are not intended to vary are `BYPASS_LMT` (thus RGB relative ratio preservation), `RADIOMETRIC_ODTS` (again RGB relative ratio preservation), `MDR_RADIOMETRIC_PROPORTION` or `HDR_RADIOMETRIC_PROPORTION` (as appropriate) set to 1.0 (again RGB relative ratio preservation), and `ROOM_BRIGHTENING` set to 0.0 (dark surround viewing during mastering).

With these settings, the RGB relative ratios, and thus chromaticities are preserved through the inversion process. Thus, the chromaticities of the resulting ACES RGB are identical to the chromaticities of the dpx file that was thus inverted (assuming the parameters matched its colors space and gamma parameters).

The intention is that bright color desaturation, and room brightening, should be applied in subsequent display presentation via a subsequent forward rendering nugget and device-type transform. The device-type transform might use a possible different setting of `ROOM_BRIGHTENING` if viewed in a bright room. It might use a different setting for `MDR_RADIOMETRIC_PROPORTION` or `HDR_RADIOMETRIC_PROPORTION` (as appropriate) , or even turn off `RADIOMETRIC_ODTS` altogether (for full desaturation of bright colors.

Another way to think about this is that all bright color saturation in the inverted source is preserved, whether or not it is intended to be desaturated. If the intent is to desaturate it further, then this is easily done. If not, then the bright color desaturation already present in the inversion source file will be exactly preserved.

Once the inverted ACES RGB source has been obtained via inversion, then other display-type transforms can be used. For example, a P3_D60 pq_gamma HDR dpx source can be inverted into ACES RGB. That ACES RGB can use a different display-type transform to yield Rec709 D65 at 300nits (MDR) at gamma 2.4.

Note that dpx files having R=G=B=D65 will be converted via the device-type forward matrix to invert back into ACES R=G=B=D60. Thus, interoperation between D60 and D65 is a matter of which matrix is used in the display-type transform both for inversion and for subsequent presentation. Of course, the issues described above concerning the top 10% of blue (1% of luminance) remain.

Checklist for HDR_CPU_INVERSE

• Know what colorspace and gamma is used in the dpx file to be inverted into ACES RGB. Also know whether the file is HDR or MDR.

• Check that the hdr_macros.h file is set up to match. In particular check flags `PQ_GAMMA, MDR_DISPLAY_GAMMA, HDR_DISPLAY_GAMMA (if not PQ_GAMMA and HDR)`. Use the appropriate matrix if D65. Default matrices are D60 (including MDR rec709).

• Check that the hdr_macros.h file is set up as required for inversion. In particular check flag `RADIOMETRIC_ODTS is on`, `MDR_RADIOMETRIC_PROPORTION and/or HDR_RADIOMETRIC_PROPORTION (as appropriate) is set at 1.0`, and `ROOM_BRIGHTENING is set at 0.0`.

• Also check that `BYPASS_LMT` is set.

• Compile HDR_CPU_INVERSE.cpp using by copy and paste of the build line at the beginning of the code. A gnu C++ compiler must be on the system, and OpenExr must be installed for this to build. There is no makefile.

• Invoke the command line using HDR or MDR to match the dynamic range of the input dpx file.

• Make sure the input files are dpx 10-bit rgb files in the appropriate primaries (Rec709 and or P3 in D60 or D65 whitepoints current specified or available in commented matrices). Input file extension .dpx.

• Select exr half-float output files (extension .exr). These will be ACES RGB files with R=G=B=D60. These files could be considered as "scene referred", but clearly they are only indirectly so.

• Use %07d in the name (for 7 digits with leading zeros) for the range of frames. If you are unfamiliar with this, ask a C programmer how this frame number structure is typically specified.

• Select the range of frames to process (start to end frame numbers)

• Select either 1 for MDR or 2 for HDR as the display-type represented by the input dpx file.

• This code is not multi-threaded since atomic operations would be needed are not available in pthreads. Memory use is not excessive, but memory should have sufficient size for 4k workflows if working on 4k files (the test ACES files are all 4k). There is a long-duration computation before frames are processed, wherein the inverse is populated by the random number process. The process is set at 32 batches, but 128batches may be needed to best fill extremes of bright, dark, and saturated colors. Computation may be several minutes for 32 batches, and four times as long for 128 batches.

• The output .exr files have pixels in ACES RGB. These can be used as input files identically to the uses that ICAS and JKP Alps scenes are used as input to HDR_CPU (or HDR_GPU_CL_GL). The display-type configuration of HDR or MDR may be used on such additional forward processing, with any parameter altered as appropriate. In other words, interoperability has been achieved across whitepoints, gammas, pq_gamma, Rec709, P3, and MDR and HDR ranges.

Summary

It is hoped that these relatively simple code modules (the inverse excepted) form the basic tools that could be leveraged to achieve the single master goal for an HDR system. This is certainly early work on this topic, but the hope is that we can build on this initial set of tools and refine them to the level required.

Clearly many parts of the HDR image chain will need active participants. We thus seek additional contributors, to help with the testing, evaluation, and refinement of the HDR System.

GOOD LUCK!

Contact:
Gary Demos
Image Essence LLC
Chair of Adv Imaging, ASC Technology Committee
garyd@alumni.caltech.edu
(310) 383 2201 cellphone