Max-imize Your Color: **Jul-2010**
Expert Tips from our Color Scientist Max Derhak

Going In-Depth On Image Sample Bit Depth

In this month's installment of *Max-imize Your Color* we will discuss image sample bit depth and some of the roles it can play in getting great output.

Color pixels in an image represent a recipe for defining a color. Generally, photographic images are saved as 24-bit RGB images with each pixel being made up of three 8-bit samples containing values for red, green, and blue. Each pixel sample represents the relative amount of excitation of each pixel's phosphors on a CRT monitor. Thus RGB values can be thought of as percentages of the maximum amount of phosphor excitation ranging from no excitation (0%) to maximum excitation (100%) - or expressed numerically as a factor going from 0.0 to 1.0.

The range of values going from 0.0 to 1.0 can be represented as unsigned integers with the precision being determined by the number of bits that are used. Having more bits allows for more unique levels to describe the range going from minimum to maximum values.

To help explain this concept let's consider the simple case of encoding sixteen levels using various bit depths. The following table shows how sixteen evenly spaced values (x) ranging from 0.0 to 1.0 are encoded as unsigned integers of various bit depths.

x	2bit	4bit	8bit	16bit
0.000	0	*0*	0	0
0.067	0	*1*	17	4369
0.133	0	*2*	34	8738
0.200	0	*3*	51	13107
0.267	1	*4*	68	17476
0.333	1	*5*	85	21845
0.400	1	*6*	102	26214
0.467	1	*7*	119	30583
0.533	2	*8*	136	34952
0.600	2	*9*	153	39321
0.667	2	*10*	170	43690
0.733	2	*11*	187	48059
0.800	3	*12*	204	52428
0.867	3	*13*	221	56797
0.933	3	*14*	238	61166
1.000	3	*15*	255	65535
Effective levels	4	**16**	16	16

Notice that when encoding using two bits, only four levels are available and therefore multiple values of **x** end up being represented by the same level value This is due to the lack of precision available for representing each of the sixteen levels.

If an image only has 16 levels, then only four bits are needed to uniquely represent each value of **x** and accurately keep track of levels in the image. 16-level pixels can therefore be stored, loaded, duplicated, and displayed using four bits without any loss of information.

A problem can occur though if a color operation is performed where the pixel values can be modified in a way that results in intermediate levels being needed. An example of this is performing a partial scaling operation (such as in linearization or a general tone control operation). The following table shows what happens if a gamma factor is applied to the 0.0 to 1.0 range (similar to applying a linearization).

$y=x^{1.8}$	2bit	4bit	8bit	16bit
0.000	0	*0*	0	0
0.008	0	*0*	2	501
0.027	0	*0*	7	1743
0.055	0	*1*	14	3617
0.093	0	*1*	24	6070
0.138	0	*2*	35	9071
0.192	1	*3*	49	12595
0.254	1	*4*	65	16622
0.323	1	*5*	82	21138
0.399	1	*6*	102	26130
0.482	1	*7*	123	31587
0.572	2	*9*	146	37499
0.669	2	*10*	171	43857
0.773	2	*12*	197	50653
0.883	3	*13*	225	57881
1.000	3	*15*	255	65535
Effective levels	4	**13**	16	16

In this case the gamma operation results in a remapping of the values of x that are not exactly aligned with the levels associated with the 4-bit integer encoding. The results are therefore rounded resulting in a loss of three effective levels being used after the gamma operation is applied. For the purposes of discussion we will call this “effective level degradation”, which can result in potential posterization (where multiple different colors are converted to the same color) of image content.

Having more bits available for encoding levels provides additional precision to encode partial level results. Thus, the effective 16 levels available after a scaling operation remain the same for the 8-bit and 16-bit cases.

The amount of effective level degradation that occurs is determined by how many partial levels are defined and rounded by the results. (For example: scaling all values by 0.2 results in 1/5 the number of effective levels). For practical purposes, in most cases with 8-bit per sample images the amount of scaling is relatively low and therefore the ability to notice any effective level degradation occurring is minimal.

What it means for you

What this means is that tone mapping (linearization) operations often result in fewer effective levels than you start with. It is therefore desirable to have more levels available to store the results.

Digital camera and scanner companies are well aware of this issue. Because of this they often internally capture images at higher bit depths in the device to ensure that the internal color correction operations to get to standard RGB color spaces result in 256 effective levels that are passed along with 8-bit sample data.

Additionally, many digital cameras now support the ability to save images using 16 bits per channel. This provides the ability to perform repeated image edits (many, many times) in photo editing tools without effective level degradation becoming visible. Once all edits are performed and the image is ready to print, then it can generally be saved and/or processed as an 8-bit per sample image, and printed without any additional loss in quality due to effective level degradation if appropriate measures are undertaken by the image reproduction system.

New to X10

With the introduction of **X10**, two processing paths are now available to reproduce images in RIP-Queue. The legacy mode involves using mostly 8-bit processing until the very end of the output chain. The second processing path introduced with X10 provides the ability to process using a complete 16-bit processing path.

Let's consider how effective level degradation applies to processing 8-bit per channel image files in RIP-Queue using the legacy 8-bit per sample processing mode. To improve throughput while minimizing effective level degradation, RIP-Queue employs varying bit depths in the processing pipeline. The following diagram shows a general description of the stages of operations performed by RIP-Queue to process an 8-bit per sample image for printing. Between each stage is indicated the bit depth of the data passed between the operations.

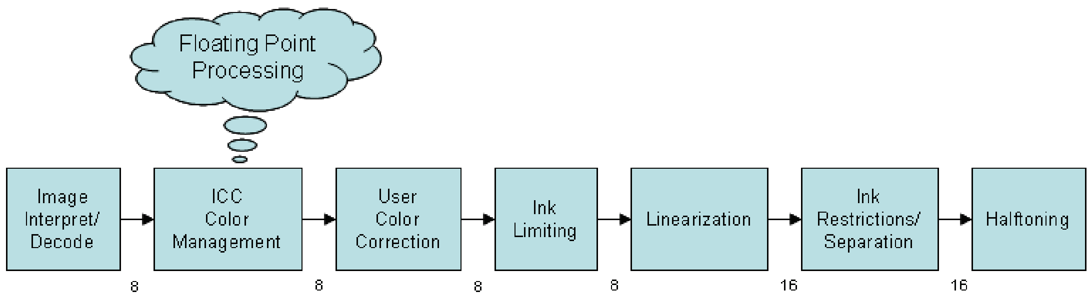


Figure 1 – 8-bit per sample processing path

When 8-bit per sample processing is used, images are interpreted/decoded in the first stage and color pixel data is converted to 8 bits per sample. Images that do not store eight bits per sample are supported by the software through sample conversion. For instance, in the case where a 16-bit per sample image is processed using the 8-bit path, the pixel sample data is simply decoded and converted to eight bits per sample for further processing.

(Note 1: In practice few images that are printed actually use or need 16 bits per sample. As such, it has not been observed that initially converting such images to eight bits per sample for the purposes of processing to be a significant issue as long as the system employs techniques to minimize effective level reduction).

(Note 2: Rendered image formats (like PDF/PS) can also contain vector data that can be rendered at varying bit depths. In this case the use of eight bits for rendering limits the maximum number of steps available in some gradient blends to 256 steps. Using the new 16-bit per sample processing path in X10 and the enhancements that we made to the PDF/PS RIP Engine ensure that blends are rendered with more than 256 steps, which can result in better looking output of gradient blends).

ICC color management is then applied in the second stage. The ONYX ICC color engine internally utilizes 32-bit floating point for all mathematical operations. Effective level degradation is therefore not a problem with ICC color management in ONYX, since the conversion between two different color spaces is much more involved than simple channel scaling. With the high accuracy of the floating point processing in the ONYX color engine, this results in 256 levels per channel (thus utilizing the full eight bits) coming out of color management without any effective level degradation.

User color correction has the opportunity for some effective level degradation in an 8-bit per sample pipeline. If user color correction is not used then no effective level degradation will occur. The extent to which it might be noticeable depends upon the operations that are performed by the user color correction edits. Significant scaling of output levels can result in the greatest effective level degradation.

Ink limiting only has the potential to introduce effective level degradation in an 8-bit per sample pipeline in areas where significant ink limiting occurs (I.E. ink limits below 200%). Even in these cases since advanced ink limiting is selective in how ink limits are applied the extent to which degradation can occur is minimized.

ONYX has long recognized that it is crucial to prevent level degradation when performing calibration, applying linearization, and performing ink restrictions. As such **the calibration process always converts 8-bit data into 16-bit** data as part of the linearization process. This ensures that linearization and the remaining stages will always provide the ability to output 256 levels for 8-bit data coming into the system.

After linearization, ink restrictions and light ink separation are performed using 16-bit data values. This means that scaling can occur without incurring visible penalties from effective level degradation.

Halftoning is the final stage which is also performed using 16-bit values.

Support for a completely 16-bit processing path was added in the Version X10 RIP-Queue. In this case 16-bit sample data is passed throughout the rendering pipeline. Image data is converted to 16-bits after images are interpreted/decoded in the first stage. If an image has less than 8-bits per sample encoding it is converted to 16-bit per samples and passed along to the next stages in the processing pipeline.

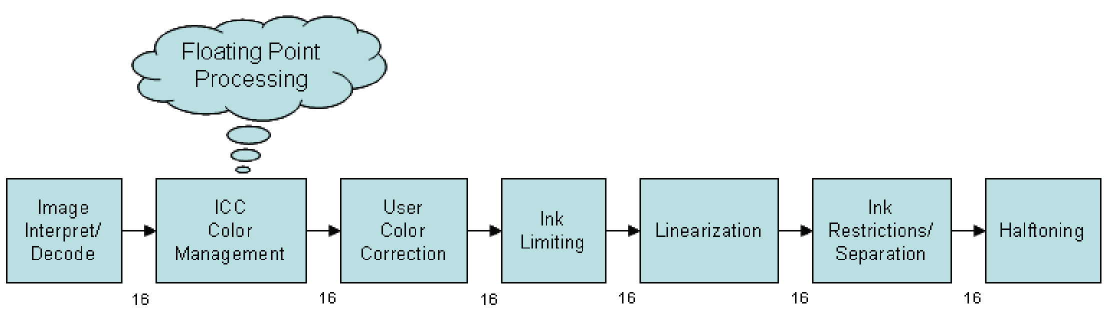


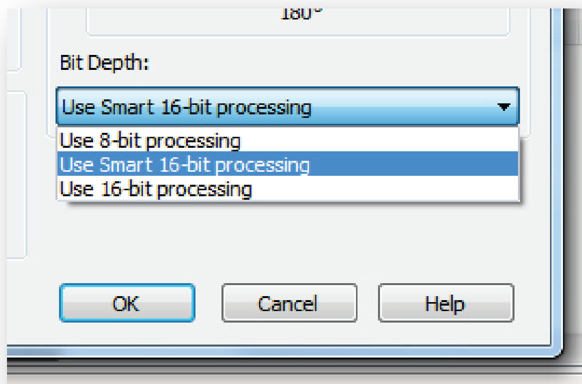
Figure 2 – 16-bit per sample processing path in Version X10

Processing with 16-bit data involves extra overhead resulting in potentially longer processing times. Calculations are performed with twice the precision and twice as much temporary internal storage is required resulting in more data access to memory.

Generally, little difference will be seen when using a 16-bit processing pipeline with 8-bit per sample image data. Some small differences may result by ensuring that 256 levels will be used when user color correction or ink limiting are performed.

(Note 3: Nearly all contone print modes for printers only support 8-bit data as input to the printer. Since the bit depth is limited by the printer, these print modes use the 8-bit per sample processing path even if a 16-bit processing path is selected in RIP-Queue).

Smart 16-bit processing feature



At ONYX we believe in giving you optimal color output as efficiently as possible, which is why we created the Smart 16-bit processing feature. By selecting the "Smart 16-bit processing" option the system intelligently chooses what processing path to use depending upon the image content and output size. If 16-bit per sample image data is present or blends in EPS/PS/PDF files that could most effectively use more than 256 levels are present then 16-bit per sample processing is used. Otherwise, the quicker 8-bit per sample processing path is selected.

The larger the output the more likely that EPS/PS/PDF blends will benefit from using 16-bit processing. However, as the output gets larger the processing time can also substantially increases and using 16-bit processing might not necessarily make sense if the output will be viewed at a large distance.

In summary, the 8-bit per sample processing path is recommended to ensure the quickest processing time with only a slight degradation of 16-bit images and the limitation of 256 steps for gradient blends in EPS/PS/PDF images. The 16-bit per sample processing path is recommended to ensure that you will always avoid bit depth degradation and get the best results for blends while potentially taking longer to process. The "Smart 16-bit" selective processing option is recommended when you want to strike a balance between processing faster and getting better output quality when it makes the most sense.

In conclusion

ONYX's application of the strategic use of varying levels of bit depth – moving between 8-bit and 16-bit as needed – is designed to minimize level degradation while still maintaining maximum processing speed and efficiency.

Learn more about "Smart 16-bit Processing" >>

(Note: This document is relative to printing images that are meant to be printed or displayed on a monitor. This document does not cover High Dynamic Range (HDR) images which usually have more than 8 bits per sample to enable the encoding of the greater dynamic ranges resulting from the combination of the capture of multiple exposures of the same scene. Such images require careful tone mapping or complex processing to be properly displayed and/or printed out. In such cases it is recommended that photo editing tools be employed that provide the ability to render such images into a form that is appropriate for low dynamic range output devices like printers or typical computer displays).