Overview

CRT monitors have the inherent property that they do not respond to input signals in a linear fashion. The input voltage is not directly proportional to the output luminance, so a voltage of 50% does not result in a luminance of 50%. This means that a color of RGB 0.5/0.5/0.5 will not be displayed as a mid-gray as expected but due to the exponential response curve, rather as a gray with 25% intensity.

Gamma Curve

The CRT response mapping has the property that more resolution is given to low-luminance RGB values. This is actually good since it fits the human visual system well. The RGB color space is commonly quantized to 8 bits per component. This quantization that includes only 256 shades will inevitably result in noticeable color banding when viewing a gradient. The human eye can distinguish more shades in lower luminance values than in higher values. So if a color space has more resolution in the low-luminance values, the gradient will be more perceptually uniform. This means that a constant increase of the intensity will result in a constant increase in the perceived brightness.

The transformation that monitors apply to the image can be described as an exponential function, called gamma curve. This curve is standardized as sRGB with an exponent of 2.2. sRGB is the color space that is used by Windows, the WWW (according to W3C) and most image formats including JPEG. This means that most devices like scanners and digital cameras encode the image in sRGB as well. So sRGB material is ubiquitous. It is important to note though that sRGB is not a separate image format. It just describes how the RGB color values stored in standard image files should be interpreted.

Correcting Gamma

Properly respecting gamma involves two basic steps: input and output gamma correction.

Output Gamma

Since the monitor is applying a gamma curve to the output, the input needs to be conditioned by applying the inverse. CryENGINE will automatically take care of that when sRGB support is enabled.

If no output gamma correction is applied, the final output will look too dark. The image below shows the difference for a linear gradient that is displayed without correction (top) and with proper gamma correction (bottom).

Input Gamma

As mentioned above, most color images (photos and images painted in an image editing tool) are stored in sRGB space to make best use of the limited 8 bits of precision and reduce visible banding artifacts. However, there is a big caveat when using sRGB input images for rendering. Most of the rendering equations assume linear input values and as stated before, sRGB is exponential and hence not linear. Light accumulation for example should be done in linear space.
Modern GPU's support doing the sRGB to linear conversion on the fly when sampling a texture. It is only required to flag textures as sRGB. In CryENGINE, the corresponding parameter can be set individually for each texture with the texture presets and settings.

sRGB should only be used for color textures, that is photos or textures that are painted in Photoshop. Other textures that contain computed data like normal maps should be stored with linear encoding, hence not as sRGB. Note as well that sRGB is only required for textures with 8 bits per channel. Texture formats with more bits have enough precision to store linear color values without banding artifacts.

**Using sRGB in CryENGINE**

**Enabling sRGB**

- sRGB can be enabled by setting `r_UseSRGB` to 1.
- sRGB usage should be configured in the system.cfg. Changing sRGB usage at run-time is not supported (unless you are using dx9 on PC) and may result in some textures being in sRGB and some in non-sRGB mode.
- When `r_displayinfo` is on, “SRGB” is displayed in the feature line in case sRGB rendering is enabled.

![Example of having sRGB disabled (left) and enabled (right). Note the overexposure and banding in the non-gamma-corrected image.](image)

**Monitor Calibration**

To get best results, monitor brightness and contrast should be adjusted so that the display can show a maximized color range. This helps to get more consistent color results among different displays as well.

There is a small utility integrated in the engine that can help to adjust the display. It can be accessed by setting `r_showGammaReference` to 1. The utility displays a standard gamma gradient and interlaced references/tests.

The calibration is easy to do and usually won’t take more than 2 minutes.
The first test image is an interlaced square on a gray background. Adjust your display brightness and contrast so that the middle square color blends with background color. Moving away from the monitor a bit and slightly closing your eyes helps when setting this up.

Televisions apply image rescaling by default, so results are no longer pixel-accurate which is required for the test. You might want to skip this first test for televisions.

Alternatively, there might be an option in the television to disable image rescaling. You can do that, perform the test and re-enable the scaling afterwards.

The second image contains gradients. For this test you need to adjust your display brightness and contrast settings, so that each shade in the gradient is clearly visible while at the same time trying to keep the result of the previous test. The dark shades at the end should be almost indistinguishable.

Luminance gradients in linear and sRGB space

Console Specific Notes

The Xbox 360 does an additional gamma adjustment that takes the sRGB front buffer and converts it to a gamma that is closer aligned to a TV’s response curve.

Other platforms do not necessarily perform this additional adjustment, so the same scene may look darker on the Xbox 360 compared to other platforms.

Further Reading

- Gamma Correction, Wikipedia.
- “Gamma Correction” Whitepaper, XDX Documentation.
- Gamma Through the Rendering Pipeline, Gamefest 07 Presentation.
- The Importance of Being Linear, GPU Gems 3.
- Post Processing in the Orange Box.