

Contents:

- [Standard Color Chart](#)
- [Steps](#)
- [Exporting the New Lookup Table](#)
- [Loading the Lookup Table in CRYENGINE](#)
- [Important Facts](#)
- [Best Practices](#)
 - [Photoshop Best Practices](#)
- [Troubleshooting](#)
- [Video Tutorial](#)

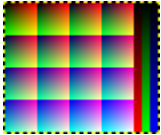
This tutorial covers how to fine tune the tones and colors in the final rendered image of your level using the [Environment Editor](#) and Photoshop to provide a custom lookup table.

The procedure is to use Photoshop to modify the standard color chart below while consulting reference images of your game, to export the modified color chart as a CryTIFF file, and finally to activate the color grade in the Engine. CRYENGINE compares the expected values with the modified colors you create, and adjusts the game image in real time.

Color grading should be the **last step** in the configuration of your environment preset. Please see the [Photoshop Best Practices](#) section below for more information.

Standard Color Chart

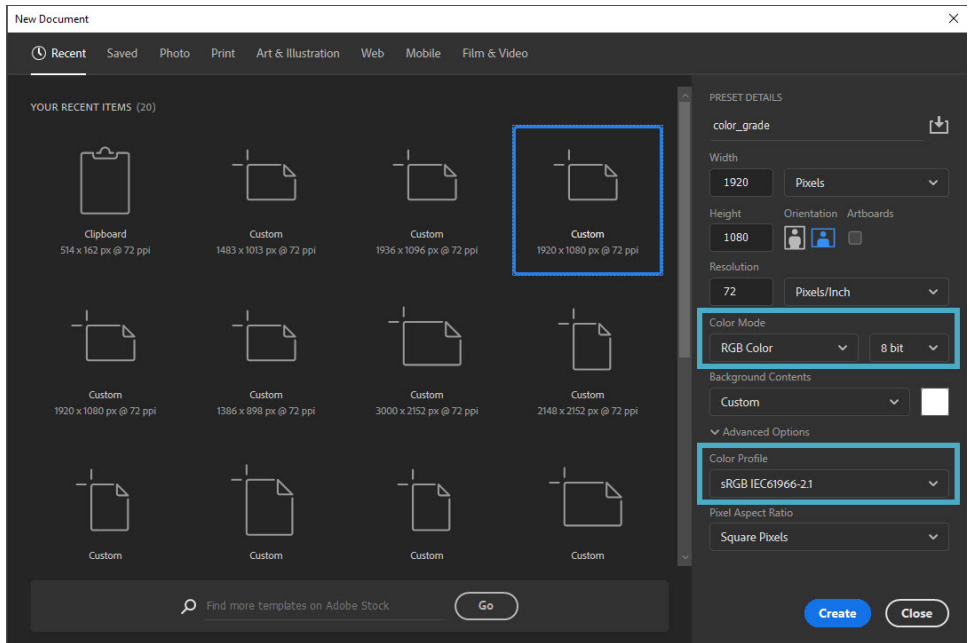
The entire basis for this workflow is the color chart image below, which you need to download by right-clicking on it and saving it.



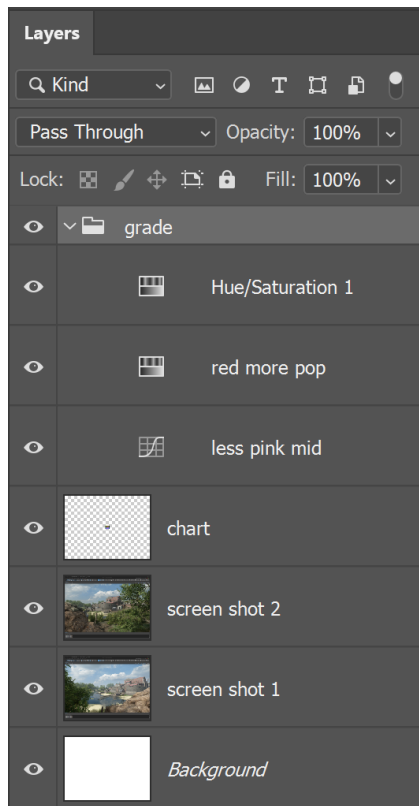
Do not modify this image! You'll use this to create a lookup table in a Photoshop document, detailed in the following steps.

Steps

- Download the standard lookup table above.
- Create a new Photoshop (.PSD) document with **Color Mode** set to **RGB Color** and **8 bit**, large enough in resolution to accommodate the screen shots you plan to capture (HD, 4K, etc.). Make sure you assign the **sRGB color profile** to your file to insure consistent color throughout the workflow.



Photoshop - New document



- Drag the lookup table into the Photoshop document. Do not resize, transform or modify it in any way. Name its layer *chart* or something of that nature.
- Grab screen captures of representative scenes in your game and paste them into your Photoshop document below the color chart layer.
- Above all of these layers, we suggest that you make a new layer group (**Layer New Group**) called *color grade* or something like that. You may end up with several versions of your color grade, so you may want to name this *color grade version 1* in anticipation of iteration and to make comparison easy by toggling layer group visibility on and off.
- Within that layer group, add **adjustment layers** to perform the desired tonal and color changes. Do not use layer masks, as your adjustments will always be applied globally.
- Once you're satisfied with your color grade, save your Photoshop file.

Exporting the New Lookup Table

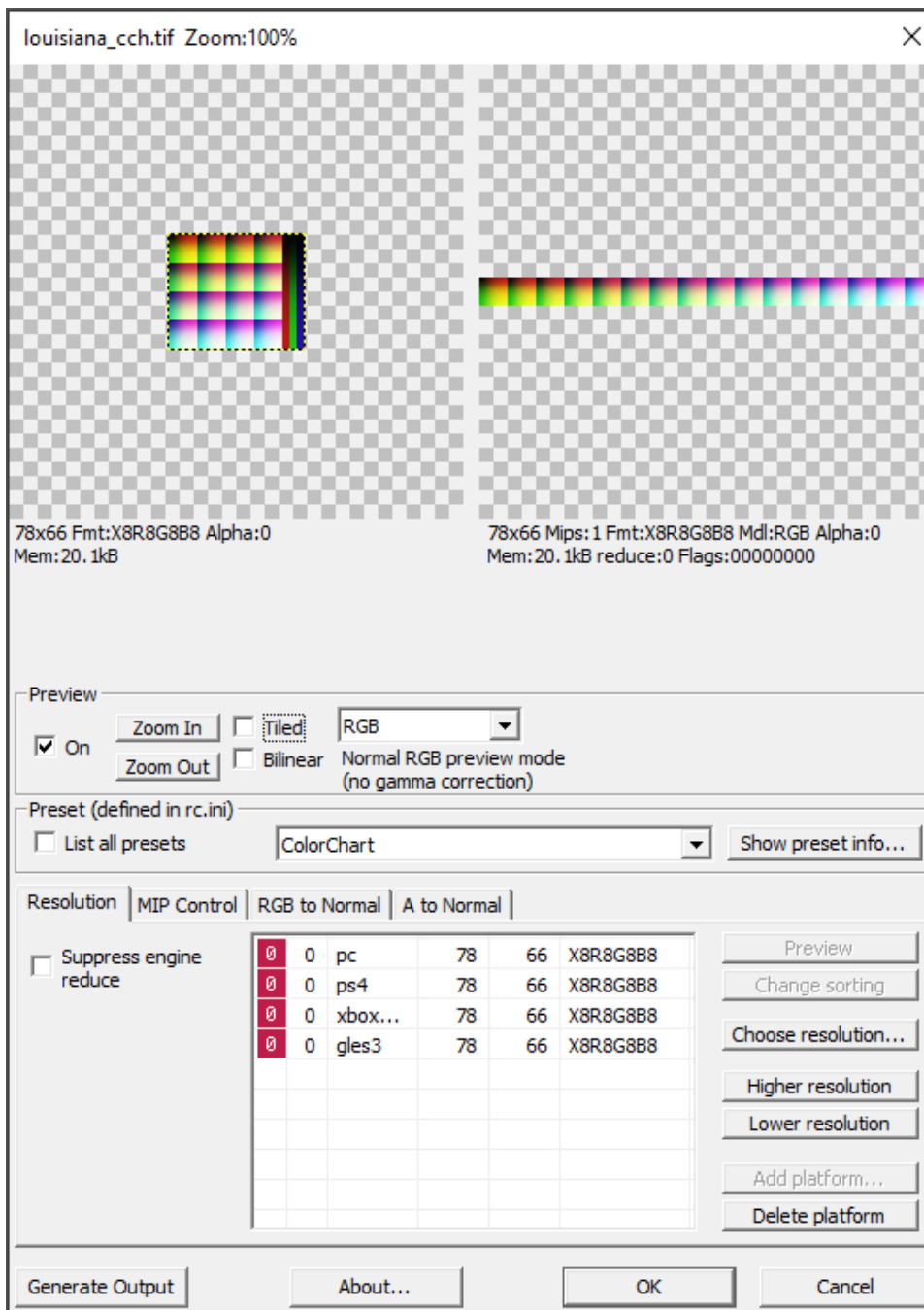
All CRYENGINE needs is the color chart whose colors have been changed through your adjustments. You can either crop your image down to only include the lookup table and then export it (this is the recommended method, as it will be a smaller, faster loading file), or export it as-is; CRYENGINE will auto-detect the lookup table and ignore any image outside of it.

- Make sure your CRYENGINE project is open so the required **.cryasset** file for your lookup table will be generated automatically.
- In Photoshop, flatten your layers. (**Layers Flatten Image**).

Tip

Don't forget to flatten your Photoshop layers before exporting the CryTif file. TIFF files with layers will **not** be readable by the Engine.

- Save your image in CryTif format (**File Save As**, and set CryTif as the file format). Append **_cch.tif** to the filename so the [Resource Compiler](#) will automatically choose the correct shader. Set the path as follows in your project folder: *Assets\textures\colorcharts*. Note that you may have chosen a custom name for your assets folder. Also, if these folders don't already exist, you may need to create them.
- In the Resource Compiler dialog box, make sure your image is **not tiled** and that **Preset** is set to **ColorChart**. Click on **Generate Output** (not the OK button) to export your file.



Resource Compiler

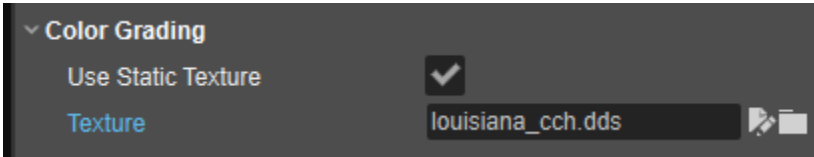
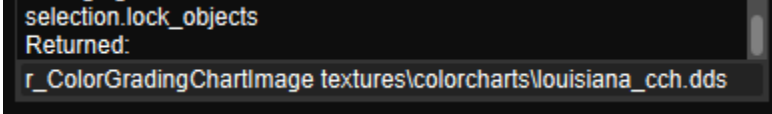
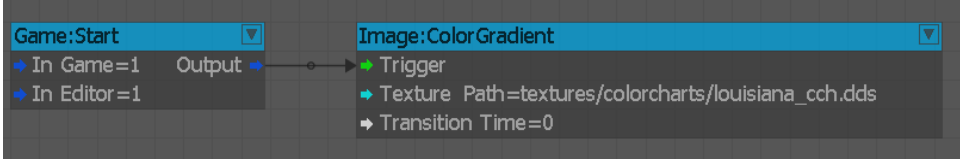
- Open your project's `Assets\textures\colorcharts` folder and verify that the file exported with a `.dds` extension name and that a corresponding `.cryasset` file was created. The lookup table is ready to use.

Loading the Lookup Table in CRYENGINE

Now you're ready to apply the new lookup table to your game:

First, insure that color grading is **enabled** (it is by default) by typing `r_ColorGrading 2` (or `1`) in the Console.

There are several ways to activate your custom color grade:

	<p>Add it to an environment preset in the Environment Editor, under Constants Color Grading by enabling Use Static Texture and browsing to the location of your lookup table in the Texture property. Be sure to save your modified environment preset. The color grade will be applied whenever the environment preset is loaded. Once a color grade has been assigned in the Color Grading Texture property, you can compare how things look with and without it simply by toggling the Color Grading Use Static Texture property on/off.</p>
	<p>Type <code>r_ColorGradingChartImage</code> followed by the relative path and filename of your color grade file in the Console (E.g., <code>textures/colorcharts/yourLookupTable_cch.dds</code>). If you plan to use this method instead of assigning it to an environment preset and you wish this to load when your game loads, be sure to add this to one of your <code>.cfg</code> files or <code>.cryproject</code> file. Just be aware that with this method, there is no guarantee that the correct environment preset on which your color grade is based is currently loaded.</p>
	<p>Use Flow Graph's Image:ColorGradient node. If you plan to switch between multiple color grades in game, it's a good idea to also assign a default color grade with the Image:ColorGradientDefaults node so Flow Graph knows which color grade to revert to after temporary switches. The Transition Time property can be used to blend from one grade to another over a selected time interval.</p>
	<p>Use code or visual scripting.</p>

Important Facts

Color grading is the last step in the render pipeline.

Since you can modify colors and tones in any way you like in a lookup table using Photoshop's sophisticated tools, it's possible for lookup tables to perform more precise color and tonal adjustments to the overall image than what can be done with **Variables HDR Color Balance** and **Saturation** properties, even though their capabilities overlap.

For example, you could;

- just desaturate deep shadows to get better blacks;
- split tone an image with cool shadows and warm highlights;
- or create a color grade to desaturate all colors and change all shadows and highlights past a threshold you choose to a colorful overlay; this helps identifying clipping or extreme tones that you might want to adjust lighting before beginning the actual color grade process.

You can see examples of these debugging color grades in the free [GameSDK's textures.pak](#) file in the color charts folder. You don't need to create a GameSDK project; just open the `.pak` file in a compression utility like 7Zip and extract the color grades that interest you into your own project's `textures/colorcharts` folder, then assign them using one of the methods described above.

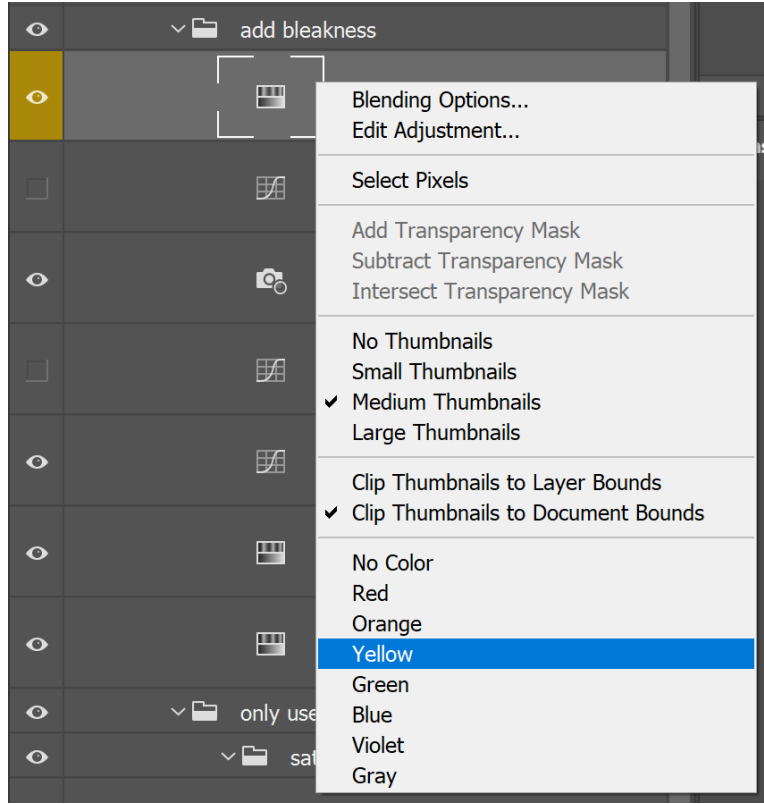
Best Practices

- In general, we recommend lighting and configuring your environment preset for a somewhat flat, **low-contrast** look, then develop a color grade that **adds** contrast to bring the image to its final desired look. Adding contrast works better than trying to reduce it at the color grading stage. If your lighting ends up producing a very high-contrast image, you risk losing highlight and shadow detail that will be difficult or impossible to recover. Trying to do so through color grading will often reveal undesirable banding (posterization) artifacts.
- Don't forget that color grading will also be applied to your game's UI and menu, so be sure to include screen captures of the relevant screens in your Photoshop file to check with your final grade - or create a separate color grade and load it when menus appear.
- Don't forget to **disable** any existing color grade **before** capturing screen shots for use in a new color grading process.
- Make sure that you capture a wide range of images from your game that represent the full range of lighting conditions, place them into your Photoshop file, and view the effect of your color grade on each of them individually.
- Be especially sure to include screen shots of your highest contrast scenes where HDR is trying to balance interior and exterior lighting levels to insure that you aren't clipping shadows or highlights through your color grade.

Photoshop Best Practices

Since color grading often goes through a long iteration process with input from art directors and many stakeholders, it's common for your Photoshop layers to grow and grow, to get disabled, the layer opacity adjusted, etc. Thus keeping your Photoshop layers well-organized will save you a lot of trouble. Here are some suggestions to help save you time and confusion:

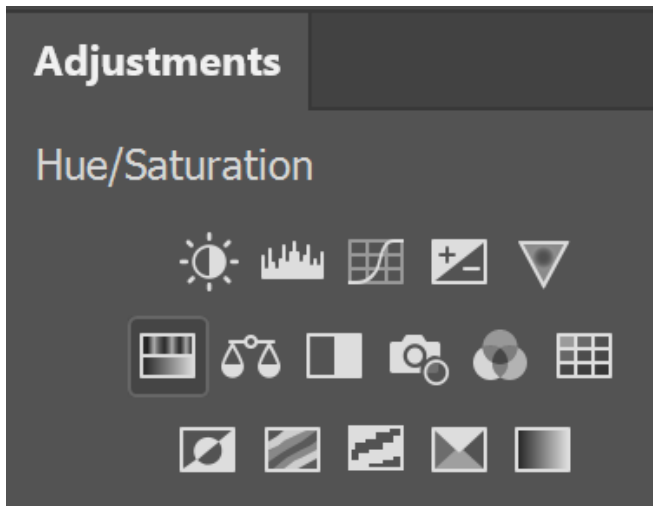
- Use adjustment layers (**Layer New Adjustment Layer**) in Photoshop (**not Image Adjustments**) so you affect the look of all screen capture layers (as well as the color chart) rather than making permanent changes to an individual image layer.
- Be sure **not** to use layer masks in your Photoshop adjustment layers, as your color grade will be applied globally, not locally.



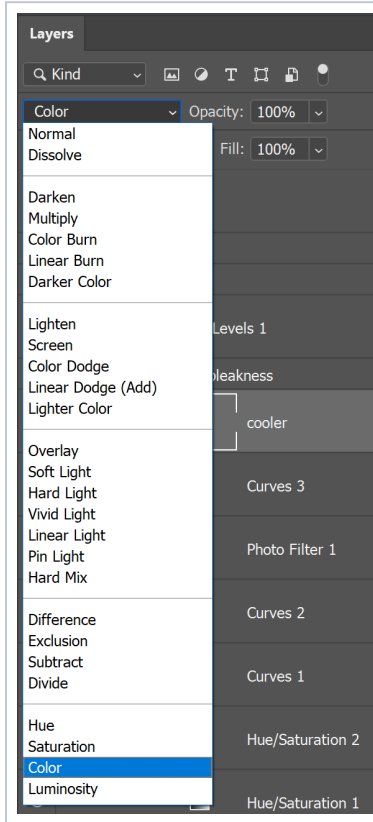
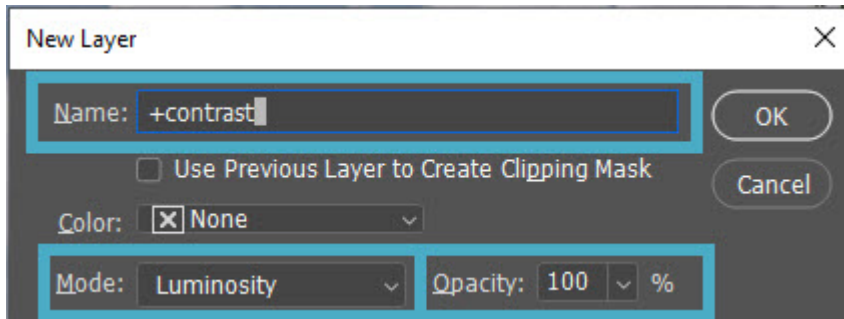
You can standardize on a system of using Photoshop's layer colors to tag adjustments with colors whose meaning your studio has agreed upon: deprecated, not used, in progress, alternative, etc.

An easy way to fine tune the strength of adjustments whose influence you like but whose effect is too strong is simply to reduce the **layer opacity** of adjustment layers. You can apply the same approach to a group of adjustment layers by putting them into a **layer group** and varying its opacity to alter the strength of the entire color grade. This is especially helpful when you want to vary the intensity of complex adjustments, like curves that have many points across the RGB channels, without having to modify every point while maintaining numerical relationships.

Naming your adjustment layers to identify their intended purpose (*desaturate reds* or *increase contrast*, etc.) is a simple way to facilitate communication across teams and saves time spent trying to figure out what each layer is doing, especially when you come back to update it some months or years later. If you develop a habit of **clicking** on an adjustment button in the **Adjustments** palette while holding the **Alt** key:



- you'll be prompted to choose a **name** for the layer, a **blending mode**, and **layer opacity**:



Photoshop's blending modes provide independent control over hue, saturation, and tonal changes. For example, adding contrast in a **Normal** blending mode will also unavoidably increase saturation, since the RGB color model has no separate channels for luminosity, hue, and saturation. Using the **Luminosity** blending mode allows you to modify tonal contrast without affecting colors. Similarly, using one of the color modes - **Color**, **Hue**, or **Saturation** - allows colors to be changed without affecting tones.

Keep in mind that color grading is the **last** step performed in the render pipeline (in 8-bit sRGB color space), **after** HDR settings are applied (in 10-bit color space from 16 bit calculations). Therefore the screen captures that you use in the color grading process have already been adjusted by any HDR settings, and a color grade isn't affected by HDR once it's activated. However, this also means that if you go back and make changes to your HDR settings in the environment preset (or anything else that affects the final image), you'll essentially invalidate your color grade and have to start all over again. Thus color grading should be the **last step** in the configuration of your environment preset. (Note that there are non-physical post-render effects that happen after the image is rendered and color grading is applied, like **Variables Sun Rays Effect Sun Rays Custom Color**.)

You can create and activate color grades for special purposes, such as momentarily switching to a stylized look when a player is struck and injured, for menus, special view modes, or for troubleshooting purposes, such as applying a bright, saturated colors to very dark and very bright RGB values to visually identify shadows and highlights that are darker or lighter than a threshold you have chosen. If you need to switch between color grades very rapidly, you may need to adjust the CVar **r_ColorGradingChartsCache**, which sets how many frames pass between updates to the color grade. The default is every four frames, but you can force it every other frame with the value **1**, or even to every frame with **0**. Forcing the Engine to refresh this cache more often than your color grade is changing will only degrade performance with no gain.

Troubleshooting

If you encounter visual glitches on the PC, please ensure that you haven't overwritten 3D settings for either the Editor or game launcher in the control panel of your video driver. This is a major source of visual artifacts.

In particular, enforcing **multi-sampling** or **anisotropic texture filtering** can break post processing effects, deferred shading, and other aspects of the final image.

Video Tutorial

This tutorial is also available in video form on our YouTube channel here: