Demos

There are two demo scenes which demonstrate how to use this effect in a single- and multi-camera setups. Check them out before doing your own experiments.

Both can be run on a desktop machine or on a mobile device (or as a WebGL application!). Controls for both are:

- W/S (or tapping/clicking the bottom or top part of the screen) - changes current predefined model
- A/D (or tapping/clicking the left or right part of the screen) - changes currently displayed image
- Space (or tapping/clicking the central part of the screen) - switches the effect on and off

If you want to see what this effect looks like with older games (which were designed to be run on a CRT and are a good lesson on how your assets should be created), download these screenshots and put them in the **UltimateCRT/Demo/Resources** directory:

https://drive.google.com/file/d/0B8h8HnACVIK_eDFWSkQ3ZGN4TWc/view?usp=sharing

Keep in mind these are not part of the project, cannot be distributed along with your game and are provided only for instructional and demonstrational purposes.

What is Ultimate CRT?

Ultimate CRT is an Image Effect for Unity3D which simulates the sort of picture old cathode ray tube (CRT) displays produced. Once added to a camera, each frame it processes that camera's output by applying various image transformations, making the image look and feel more "analogue" or "retro".

How does it work?

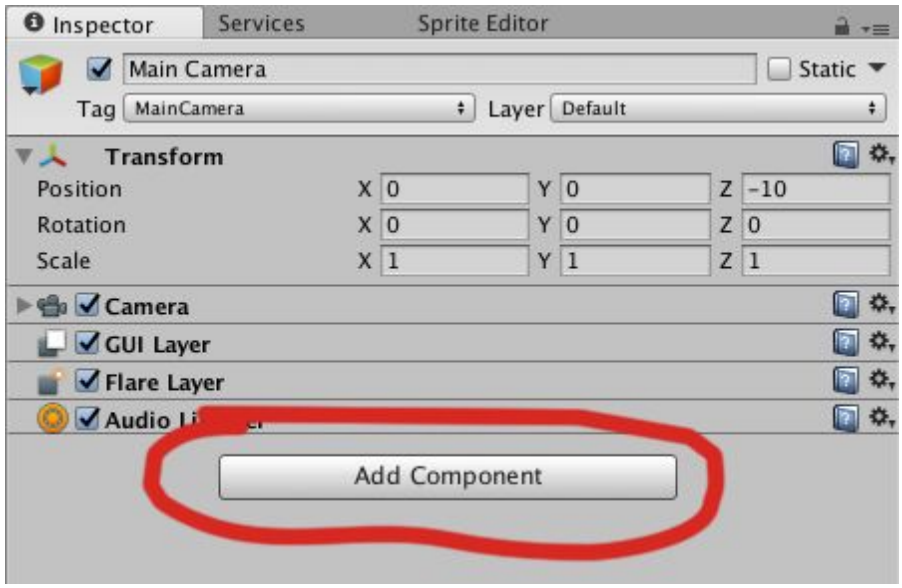
It works on a per pixel basis - each pixels of your input image will be treated as a subpixel. There are three subpixels per each color channel (for red, green and blue channels), so a perfect input image should have each virtual pixel made of a grid of 3x3 pixels (or, speaking more meaningfully, input image should be enlarged three times). This means that in general using upscaled low resolution images will work better, but it doesn't mean you can't use the high-res ones - just remember that this effect "sees" your image as made of not individual pixels, but 3x3 pixel groups.
Also make sure, if you're enlarging your input image (e.g. your sprites are drawn 3 times larger than their textures really are) to use the nearest neighbour filtering method (it's called Point in Unity3D) - this will make the output image less blurry (don't worry, it won't be blocky or LCD-like sharp, the effect will take care for it).

Standard or standalone?

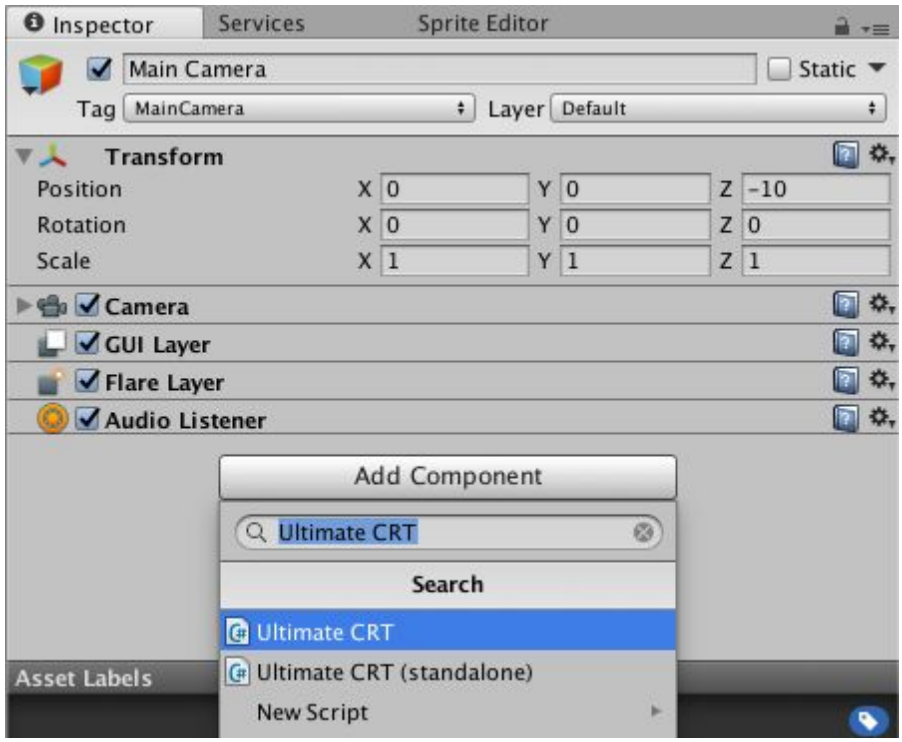
It comes in two versions. A standard version works like a normal Image Effect and can be combined with other effects, e.g. from the Standard Effect package.
A standalone version is, as the name suggest, a standalone effect. It's optimized to be a couple FPS faster than the standard one, but at an expense of not being able to be combined with other Image Effects.

Setup

To add Ultimate CRT to a camera, select one of your cameras in the current stage and then press **Add Component** button in the Inspector:



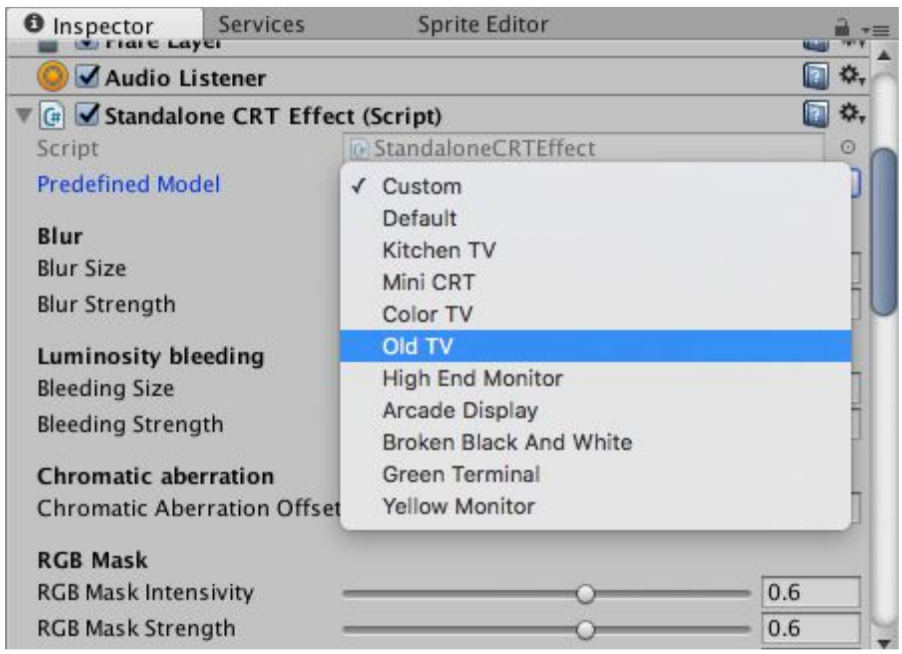
A popup window will appear. To locate the Ultimate CRT component, simply type in "**Ultimate CRT**" in the text box and select a standard or standalone version:



After that you should see the camera's output with the effect applied to it.

Predefined models

The effect itself has over 30 parameters which have to be adjusted to achieve the look you're after, so to make things a bit easier a number of predefined setups have been provided. To access these simply click on the **Predefined Model** combo-box:



and select any of the displayed options. All parameters will be adjusted accordingly.

You can use these predefined models as a base for your own setups. Just remember, after selecting the one you'd like to modify, to **switch back Predefined Model setting to Custom**. This way your changes won't be overridden after you hit play or launch your game.

Basic parameters

These are, in general, self explanatory and the best way to learn what they do is to simply play with them a bit. Also once you hold the mouse pointer over each param, you'll see a popup describing the param's function.

But for better understanding, here's what each group does.

Blur

CRTs were nowhere near as sharp as modern day LCDs, so blur's primary function is to reduce the overall image sharpness a bit. Blurred version of the input image is also used for computing **Chromatic Aberration** and **Luminosity Bleeding**, so it's good to remember that you should almost always have some blur in your setup (you can reduce its visibility, by adjusting the Blur Strength instead of Blur Size).

Luminosity Bleeding

Because of the way CRTs were built, brighter pixels were somehow prioritized over the darker ones. These settings let you adjust how much adjacent pixels will bleed over their darker neighbours, making brighter areas slightly bigger and darker ones slightly slimmer.

Chromatic Aberration

Older displays had this flaw which created colorful halos on the edges of the displayed features, called chromatic aberration. Ultimate CRT has a simple but effective way of simulating that flaw by splitting the R and B channels and rendering them slightly away from the pixel's center.

RGB Mask

To make the whole CRT effect look even more profound a RGB mask has to be added. What it does is letting only one color channel per pixel through, which simulates a part of what shadow mask did in the original CRTs. Keep in mind this is happening on a per pixel basis, so each pixel will let through only one color channel (either red, green or blue), so the first column of pixels will be all red, the second one all green, the third one all blue and the next one all red again, etc. Of course you can adjust that (because having only red, green and blue pixels on screen would simply look dreadful) and make the first column be just a bit less green-blueish instead of being completely red and so on.

White and color noise

If you're going for a truly analogue look, you need to add noise. Huge amount of noise can make the output look bad, like on a broken TV (which might also be something you're after), but adding just the right amount can actually make the image look better, by making edges less sharp, color transitions more gradual etc. When adjusting the noise level always make sure you're looking at an animated image (e.g. press play in the player), never think of it as a static effect.

Color adjustments

Here you can set the darkest and the brightest color in the output image - input pixels will be adjusted accordingly. You can use this setting to cut off some color values (e.g. not show the darkest values and make black color a bit greyish instead), increase or reduce contrast, brightness and saturation or to create a monochromatic display (like in the black and white, green and yellow predefined models).

Horizontal interference

Old or broken displays often had this horizontal bars going over the screen and that is exactly what this setting lets you set up. It can also be used (if you set the bar's length big enough and the speed high enough) to simulate screen flickering (screen will go continuously from darker to brighter in a split of a second). However I highly advice warning your users that a flickering effect might appear as it is really, really tiring for the eyes and in some cases even dangerous (may cause epilepsy).

Various CRT related settings

Here you can adjust how noticeable other flaws of old CRTs are going to be. Mask lets you increase the visibility of the shadow mask (similar to RGB mask, but it only darkens certain pixels of the image), curvature makes the image more round closer to the edges (which can be modified for both axes - if you're targeting both 4:3 and 16:9 monitors, make sure to adjust the X axis accordingly for each type) and vignetting makes corners of the image darker than the central part.

Camera's texture size

Modern LCDs can display really high resolutions which old CRT couldn't. You can (and really should) simulate that by rendering the input image onto a smaller texture and use this texture as an input for further processing. This is, of course, all automatized and only requires you to set: how the internal textures size should be computed (it has to be proportionally sized to the camera's display size) - should it be resized according to display's width or height; what's the scaling policy - should the internal texture always be created or used only for downscaling; and what's the internal texture's size (width or height, according to the previous setting).
The last setting will effectively set your simulated display's resolution, **given in subpixels**, so a texture of 600 pixels height will simulate a display which can effectively display about

200 horizontal lines (because each output virtual pixel consist of a 3x3 pixel matrix). By default this is set to 768, which is perfect for simulating old PAL displays with resolution of 320x256 pixels.

Multi-camera configuration

Ultimate CRT can only be added to one camera, which should be sufficient for most use cases. But if you use more than one camera to render your scene, you can do so and still be able to use the same effect for all of your cameras.

Setup

First, add Ultimate CRT to the last camera which renders to the screen. Then add each other camera to the **Other Cameras** property in the Ultimate CRT (the order doesn't matter). Finally make sure that only the first camera which renders the scene is the only one which clears the background (in the Inspector, **Camera** > **Clear Flags**), so the output will only be cleared once per frame. It works! :)

Troubleshooting

- **Image becomes more blurry with each frame**
  No camera clears the output buffer. Find the first camera that renders the scene and set its Clear Flags property to Solid Color.
- **No everything is rendered**
  Make sure you added the effect to the last camera rendering the scene. Also make sure than no camera except the last one uses image filters, which makes it impossible to capture the output of such cameras.