GIMME

Battle Transitions

Manual

Contents

1 Dependencies

The package will only work with URP! Otherwise no additional packages or dependencies are needed. The minimum tested Unity Version number is 2022.3!

2 Setup

In most cases, all that is required is to put the **BattleTransitions** prefab under your main camera as a child. The script will automatically search for a camera component in its parent(s).

Afterwards, the transitions have to be triggered via code - in particular in the **BattleTransitions Component**, which is on the prefab with the same name. The name of the method to be called is **DoBattleTransition**.

It only requires three inputs:

- Battle Transition Type: The type of transition to use - see 3
- Duration: The number of seconds, how long the transition should last
- Keep Active: If true, the screen quad remains active after the transition has finished

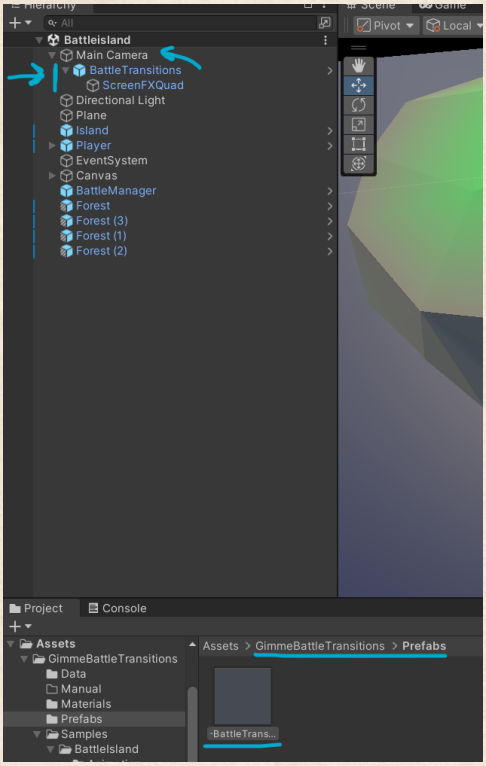

Figure 1: Prefab Setup

Starting the coroutine will render the screen texture into a render texture that is then displayed on the screen. The game will visibly pause, yet the game continues! I.e. you have to pause the game yourself before calling the method if required - this is possible, since all the effects are using unscaled delta time (the effects are not affected by the time scale)

You can find examples on how to call the method in question in the provided sample scene.

3 Effects

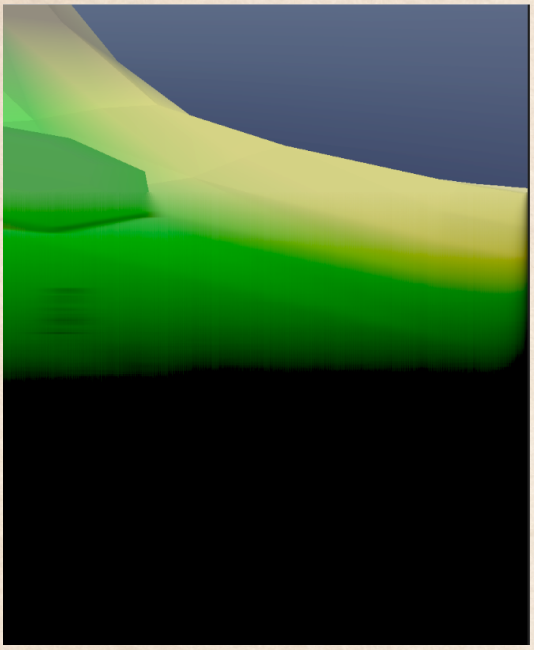

Figure 2: Horizontal

In total, four effects are included. If you use the **BattleTransitions** component provided in the asset, you will find default parameters for each effect on it. However, if you provide the parameters yourself in code, these are ignored.

The effects will only be visible, if the near offset on the battle transitions component is slightly larger than the near-clip plane of the main camera currently in use.

3.1 Horizontal

A scanline moving from the bottom of the screen to the top, inspired by **The Legend Of Dragoon**.

The parameters are:

- **Fade** - Amount of color that is removed per second once the scanline passes a pixel
- **Scan Duration** - The time necessary for the scanline to move from the bottom of the screen to the top
- **Left Smooth Offset** - Additional sample to the left of the pixel for smoothing and a washed out look (in UV Space)
- **Right Smooth Offset** - Additional sample to the right of the pixel for smoothing and a washed out look (in UV Space)
- **Offset** - Sample to the top, from which the main pixel color is sampled (in UV Space)

3.2 Vertical

A scanline with additional noise, moving from left to right. The effect was inspired by **Final Fantasy VIII**.

The parameters are:

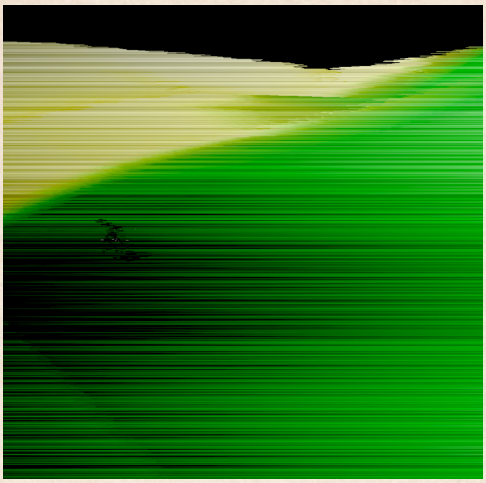

Figure 3: Vertical

- **Fade** - Amount of color that is removed per second once the scanline passes a pixel
- **Scan Start** - Number of seconds to wait before the scanline effect should start (negative number)
- **Scan Speed** - Speed of the vertical scan (in UV coordinates per second)
- **Noisiness** - Noisiness of the color fade
- **Noise Scale** - Scale of the noise used in the color fade
- **Saturate Increase** - Saturation increase per second at the beginning of the effect
- **Saturate Duration** - Time of the saturation increase in seconds

3.3 Spiral

The effect warps the screen UVs into a spiral, inspired by **Final Fantasy IX**.

The parameters are:

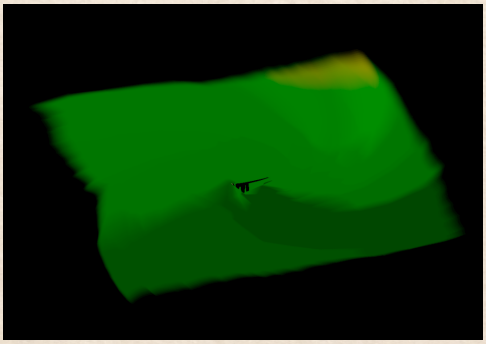

Figure 4: Spiral

- **Fade** - Amount of color that is removed per second from each pixel
- **Angle Speed** - Speed of turning
- **Center Speed** - Speed at which pixels are drawn to the center of the screen

- **Offset** - Offset (in time) from which pixels are sampled

3.4 Diamond

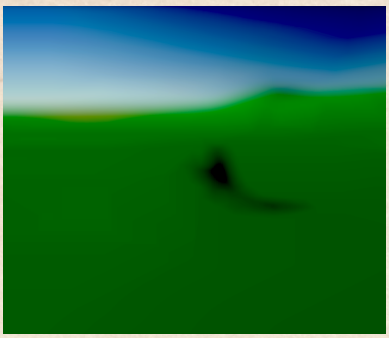

Figure 5: Diamond

A simple smoothing effect, sampling the render texture in a diamond fashion that can be used for various transitions (not only battles).

The parameters are:

- **Fade** - Amount of color that is removed per second
- **Offset** - Offset from where pixels are sampled for the diamond effect - bigger values equate to faster smoothing of the screen texture

4 Contact

For any questions or suggestions, you can reach me anytime by the following email-adress:

blenderfan@gmx.at

There is also a discord server, which is usually the fastest way to reach me:

Parable Games - Discord

Alternatively, you can also find some social media links and contact information on my website:

https://parable-games.com

Thank You!

Your download of **Gimme Battle Transitions** enables me to continue developing code and techniques for games in an independent way!