

# The Toby Foliage Engine - Light

Version 1.1.0



## Documentation

**Toby Fredson**

[florianalexandru05@gmail.com](mailto:florianalexandru05@gmail.com)

# Table of Contents

---

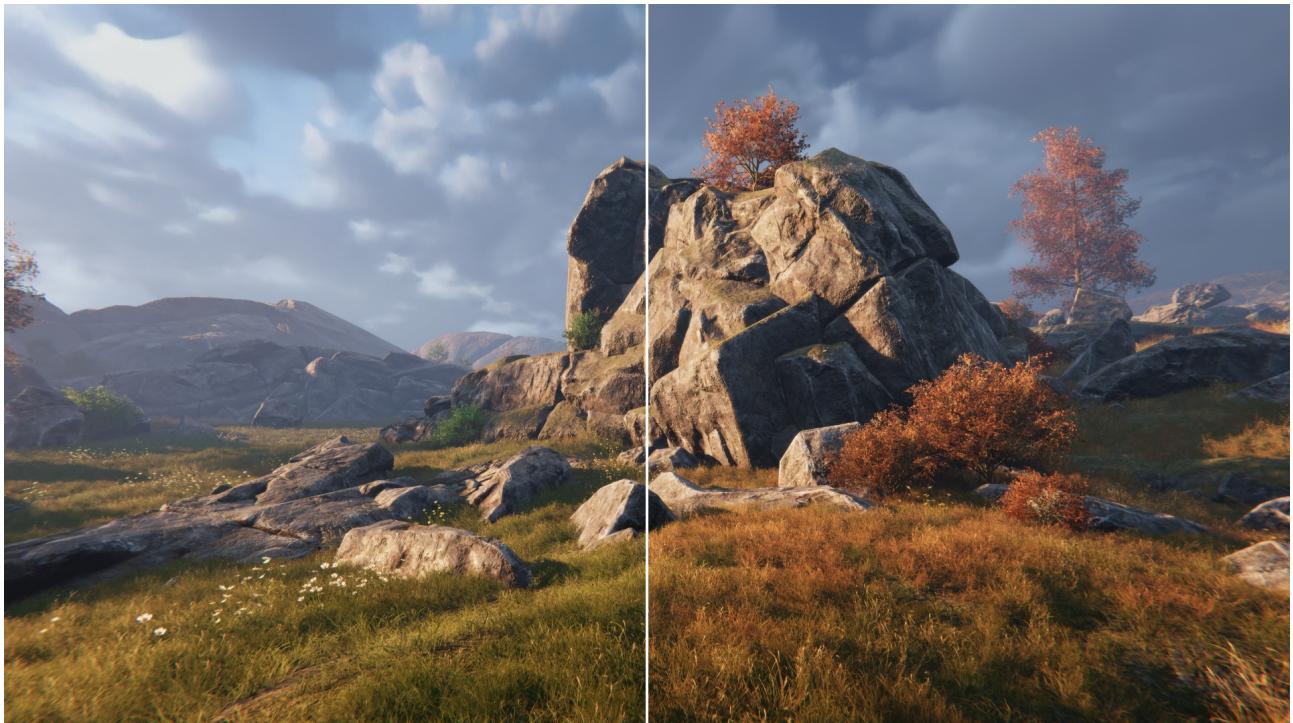
<b>Product Overview</b>	<b>3</b>
<b>(TTFE) Global Controller</b>	<b>4</b>
Global Wind Parameters	4
Known Bugs	5
Requirements for Terrain	6
<b>Shaders Overview</b>	<b>7</b>
How it Works Behind the Hood	8
Limitations	8
Vertex Colors	9
<b>Shader Settings</b>	<b>10</b>
Texture Packing	11
Texture Settings	12
TTFE Translucency	12
Limitations	12
Known Bugs	13
High Definition Render Pipeline (HDRP)	13
<b>Lighting/Shading Settings</b>	<b>14</b>
<b>Season Settings</b>	<b>15</b>
<b>Wind Settings</b>	<b>17</b>
Global Wind Settings	17
Trunk and Branch	18
Wind Mask	19
<b>Other Shaders</b>	<b>20</b>
<b>Optimization</b>	<b>21</b>
GPU Instancing	21
<b>Original Assets</b>	<b>22</b>
<b>Credits and Acknowledgement</b>	<b>23</b>

# Product Overview

Introducing the Toby Foliage Engine (TFFE), an innovative single-pivot vegetation system that simulates multiple pivots per mesh, similar to Pivot Painter. These shaders provide a comprehensive solution for achieving realistic wind and plant shading. Their seamless integration and user-friendly interface make applying it to any mesh easy. There is **no need for complex conversion steps**, as the shaders streamline the process.

TTFE is an automatic feature that works in conjunction with PBR shading. The wind physics are affected by internal mesh-based vertex position volumes. You can control the size of your volume to mask out wind in your material settings. Similarly, you can modify the behavior of your wind through the material settings.

These shaders are designed to provide optimal performance and quality for game foliage. Created with **Amplify Shader Editor**, these shaders are designed for **simplicity and efficiency**.



## Compatibility

Before using TTFE, note the following compatibility details:

\* Please note that the light version includes only the essential features and only one basic wind type is available. Amplify shader functions are not available, you can only open the shaders in code. To benefit from all features, make sure you download the full version: [The Toby Foliage Engine](#)

**\* Some features vary based on the render pipeline. Illustrations in this manual may not reflect these variations."**

\* Amplify shader functions are not available in the light version, and you can only open the shaders via code.

# (TTFE) Global Controller

The (TTFE) Global Controller gizmo adjusts material settings for all shaders simultaneously. Drag and drop the '(TTFE) GLOBAL CONTROLLER' prefab into your scene to modify material settings globally.

The prefab is located in: [Assets\Toby Fredson\The Toby Foliage Engine\TTFE\\_Core\Resources\TTFE GLOBAL CONTROLLER](#)



## Global Wind Parameters

- **Wind Type**

Choose between two types of wind phases. You can choose only one type simultaneously, as they are fundamentally different.

- **Gentle Wind**

Creates a gentle, realistic wind effect using pivot rotation. This effect mimics the subtle rotations of plants growing in clusters, such as wheat fields. It affects grass and trees, but excludes directional trunk bending for trees. However, the branches will still be affected by the wind.

\*(Most performance-friendly) [-6 FPS, 1.5 ms.](#) (Lower-end devices /Windows system)

- **Wind Off**

Turns off the wind completely via a static switch.

\*(Performance-saving) [-0 FPS](#)

- **Wind Strength**

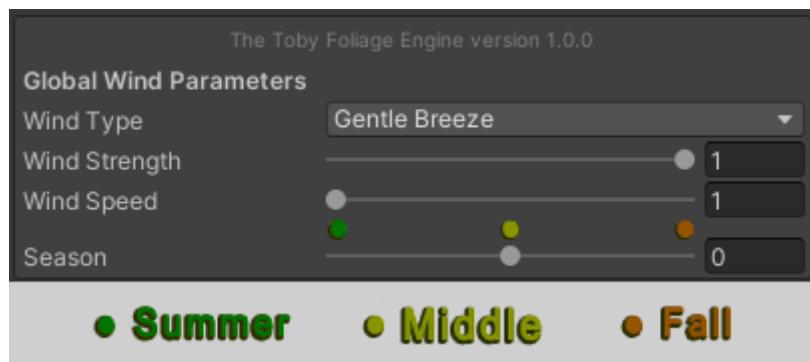
Wind intensity can be controlled and set to 0, but it's not the same as "Wind Off" and may affect performance.

- **Wind Speed**

Controls the wind speed.

## Seasons

Use the slider to adjust the season from summer to fall. You can customize more options in the material via the "Season Settings" of the shader. Adjusting these settings will largely influence the appearance of your leaves in different seasons.



The color change will be reflected in the global gizmo.



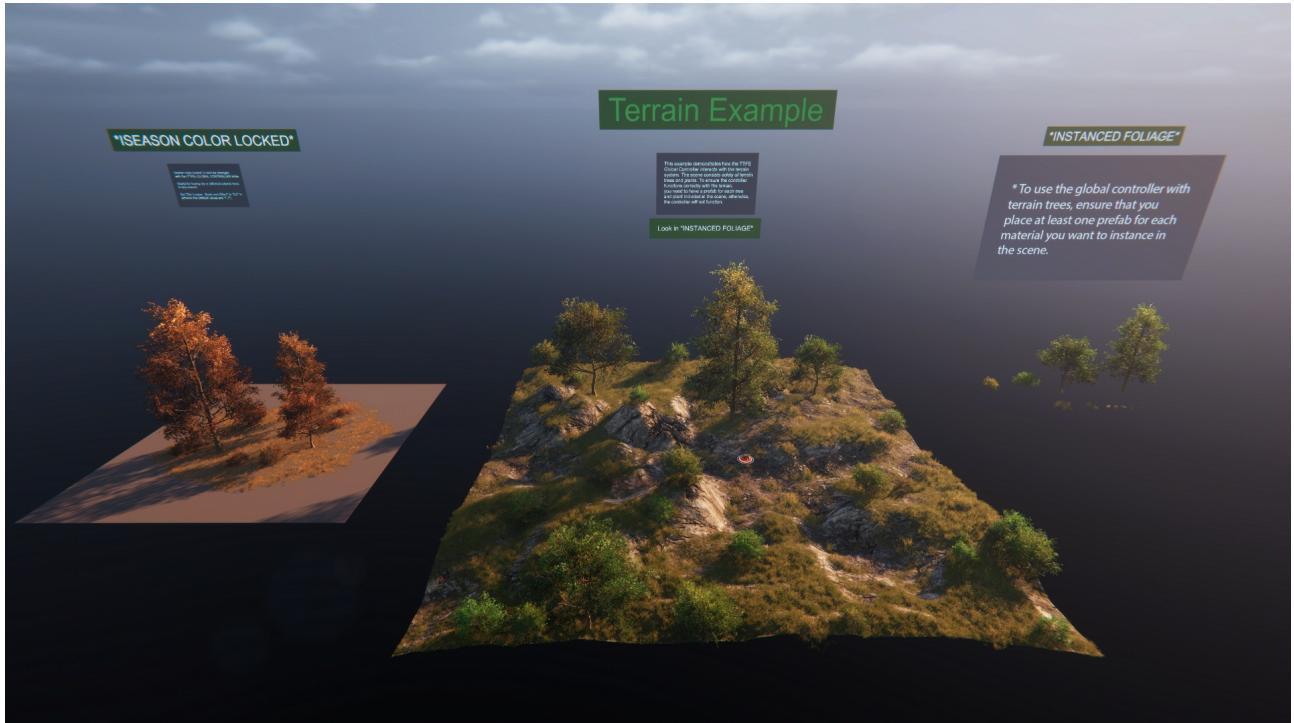
## Known Bugs

- If, after adding new vegetation prefabs to the scene, the sliders don't seem to influence the wind, This is a known issue with default behavior and can be resolved as follows: Copy and paste a new controller into the scene or reload the scene. This should help resolve the problem.
- When opening a new scene, the wind might default to one phase or another, and the leaf and bark materials will mismatch settings. To fix this, move the sliders or change the wind type.
- When saving the scene, the wind defaults to another phase. This is a glitch. Save your scene before making changes in the wind gizmo. Avoid saving afterward to preserve wind settings stored in the gizmo. The wind settings will be stored in the gizmo.

## Requirements for Terrain

**Important:** When using prefabs as terrain trees and terrain details. For the global controller to work with the default Unity terrain or TTFE terrain, at least one prefab mesh per material must be present in the scene for the terrain to instance it. Without at least one prefab mesh, the global controller will not function. This ensures the global controller can synchronize material settings across terrain instances.

Refer to the scene example provided in the package for guidance.



# Shaders Overview

To effectively adjust TTFFE shaders to fit specific requirements, it is essential to have a comprehensive understanding of their strengths and limitations. The shaders can handle various tree shapes; however, the wind effect applied to the trees will behave realistically based on their size and height. In some cases, the wind effect may not meet expectations, requiring manual adjustments.

The main tree shapes are as follows:

1. Simple tube - Tall trees and pines.
2. Round - Broccoli-like trees that can also have a thick base.
3. Irregular - The most difficult type of tree, branch bending would be impossible without Pivot Painter.



When creating a tree in your 3D application, atlas textures based on the tree's shape. Different shapes require specific settings and materials, as the wind mask must align with trunk thickness.

This section outlines the TTFE shaders' capabilities, focusing on their application to different tree shapes and wind effects.

Fig. A illustrates wind mask variations across tree types.

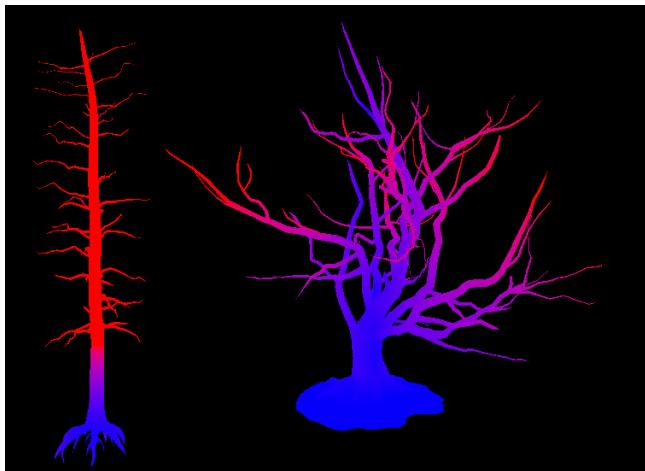


Fig. A

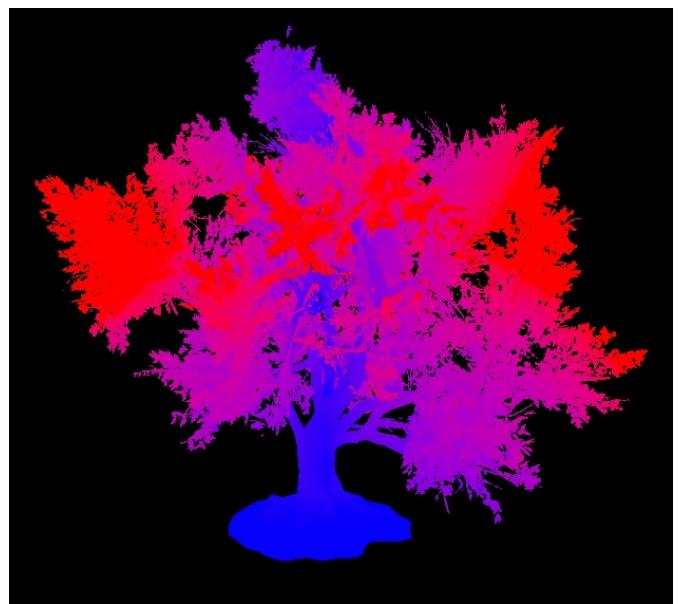


Fig. B

It's important to ensure that **bark and leaf materials** have **the same mask and wind** settings to avoid mismatch (Fig. B).

## How it Works Under the Hood

This section explains the technical mechanics of TTFE's wind system.

TTFE uses a series of vertex position nodes multiplied by UV texture coordinates, vertex normals, and tangents to rotate the entire mesh based on its point of origin. It achieves this through multiple phases, creating the illusion of complex bending.

Inside the shader, the wind system is divided into three sections: trunk wind, branch wind, and leaf wind, which includes various leaf noises for leaf cards.

*Due to the shader's method, **visible stretching** may occur in the trunk and branches. Adjust the material settings until the desired outcome is achieved.*

## Limitations

- ✕ TTFE is slightly more expensive for mobile but can be optimized by removing features.
- ✕ Shader Model 4.5+ capable devices are required (Desktop / High-End Mobile).
- ✕ While mobile devices are supported, the scenes are designed for performance testing and may not be suitable for mobile due to high performance demands.
- ✕ Only the latest LTS Unity versions are officially supported.

- ✕ Static batching is not supported.
- ✕ GPU instancing is not supported out of the box and requires third-party tools.

■ Due to the employed method, visible stretching may occur in the trunk and branches. Adjust the material settings until the desired outcome is achieved.

■ The TTFE feature is functional without the implementation of vertex colors. However, green vertex color is necessary for the "flutter feature" to work on leaves. Grass does not require any vertex colors.

■ TTFE is not a perfect alternative and may not produce results identical to those of Pivot Painter. The wind only relies on a single point of rotation!!

■ For users with lower-end devices, you can improve performance by setting the LOD bias to 1 in your Preferences under Quality Settings.

■ Setting prefabs to static will break the shader, as static batching is not supported. If you want to bake lighting, first bake and then uncheck "static" after you're finished.

## Vertex Colors

TTFE operates without vertex colors, but the green vertex color is required for the leaf flutter effect, which adds subtle movement to leaves. The grass shader does not use vertex colors. Additionally, red vertex color supports self-shading, ambient occlusion (AO), and seasonal effects, while green enhances leaf flutter.



Features like seasonal color changes and self-shading work without vertex colors but are enhanced by their use.

- ✗ Vertex color **Red\*** is not supported in the light version!
- ✗ You can't debug vertex colors in the light version!

# Shader Settings

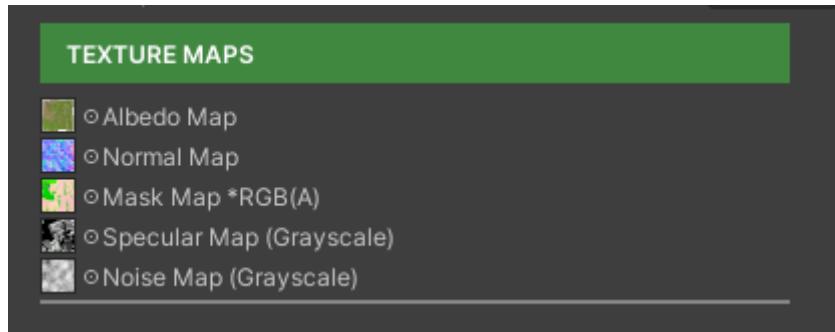
TTFE includes six shaders:

1. (TTFE) Grass Foliage
2. (TTFE) Tree Bark
3. (TTFE) Tree Foliage
4. (TTFE) Tree Billboard
5. (TTFE) Simple Cliff Masked shader.

This section covers only the most relevant settings for simplicity.

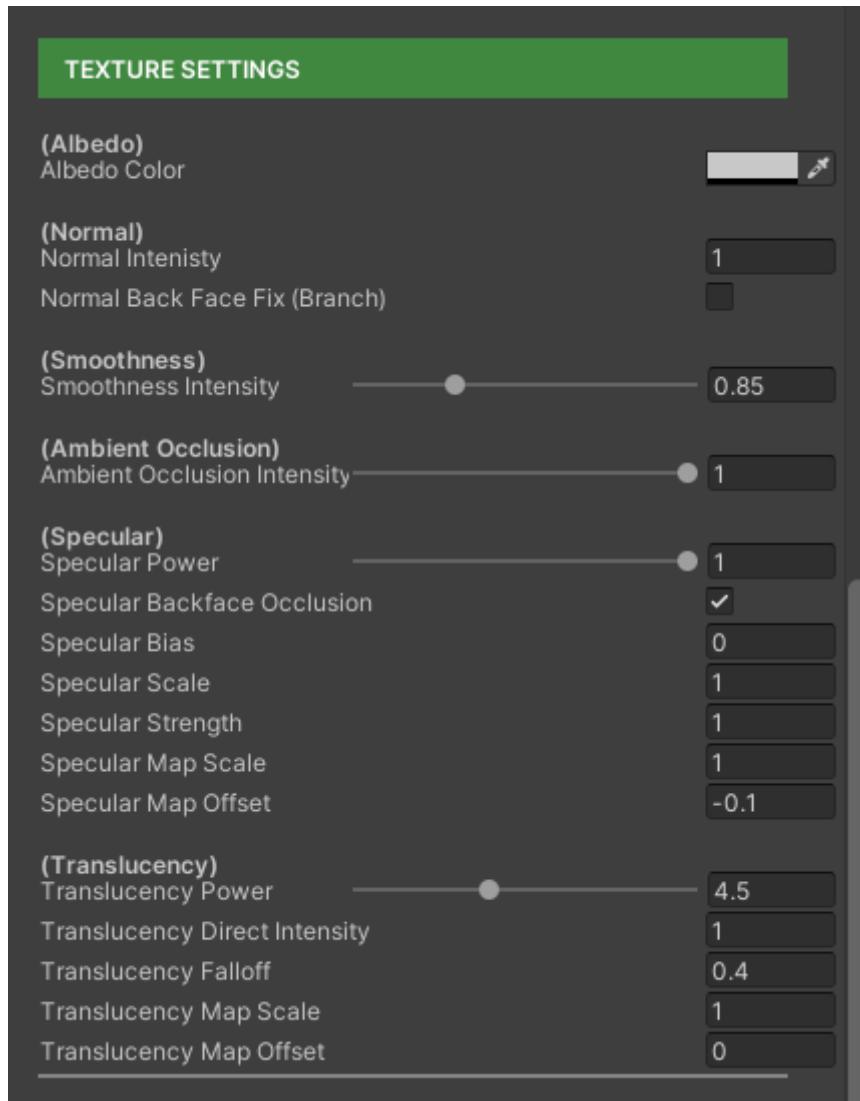
## Texture Packing

Texture packing optimizes texture usage by combining multiple maps into a single texture.



- **Mask Map \*RGB(A)**
    - R - Branch mask (masks out non-leaves in the texture and prevents them from changing color with seasonal change).
    - G - Ambient occlusion.
    - B - Translucency.
    - (A) - Smoothness.
  - **Specular Map**  
(Grayscale) - White is specular, black is not specular (Works only with "Specular Backface Occlusion" on).
- \* Only supported in URP
- **Noise Map**  
(Grayscale) - Used as a texture mask to control seasonal color changes.

## Texture Settings



- **Normal Back Face Fix (Branch)**

This branch shading is used for leaf cards with dead or dry branch textures. It corrects the normal map on the backside by flipping the Z channel.

- **Specular Backface Occlusion**

Attempts to remove the specular from the backside by recalculating vertex normals. An additional specular map can be added for more control (works with the "Specular Map" channel).

*\*In HDRP, this feature is optional or removed.*

- **TTFE Translucency (Basic)**

New and advanced deferred translucency that gets automatically occluded in shadow by a feature called "Translucency Ao". Works in Built-in, URP, and HDRP (additional to the diffusion profile In HDRP).

- **Translucency Power**

Adjust global translucent intensity.

- **Translucency Direct Intensity**

Adjust the intensity of the light dot.

- **Translucency Falloff**

Adjust the light dot falloff range.

- **Translucency Map Scale and Offset**

Adjust the intensity of the translucency map.

- **Translucency Fluffiness (HDRP Only)**

Used on grass to make it appear more lumpy.



- **Translucency Tree Tangents (HDRP Only)**

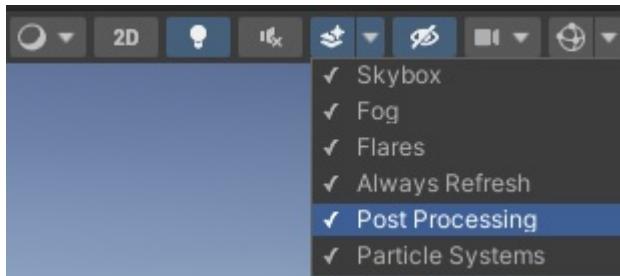
Used on trees for better and more encompassing lighting.

## **TTFE Translucency Limitations**

TTFE translucency requires one directional light to function as it is "light direction based" and is designed to work with Deferred rendering. Omni lights are partially supported, as they are locked to the sun's direction.

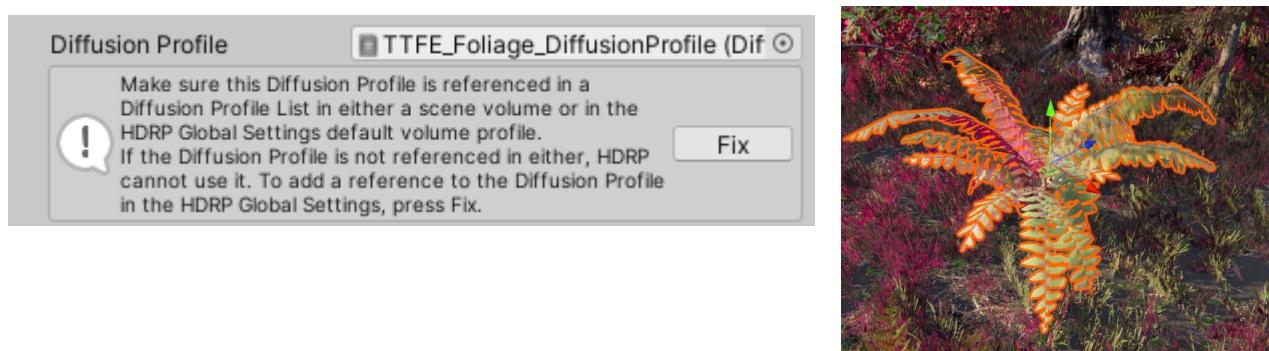
## TTFE Translucency Known Bugs

"If you encounter plant shading issues in the Unity Editor, toggle the 'Sky and Fog' button in the Scene view to resolve this known editor issue. For persistent lighting issues in the Built-in render pipeline during play or builds, install the "[TTFEL Light Glitch Fix \(Built-in\)](#)" shader or use the "[TTFEL Forward Translucency \(Built-in\)](#)" shader as an alternative."

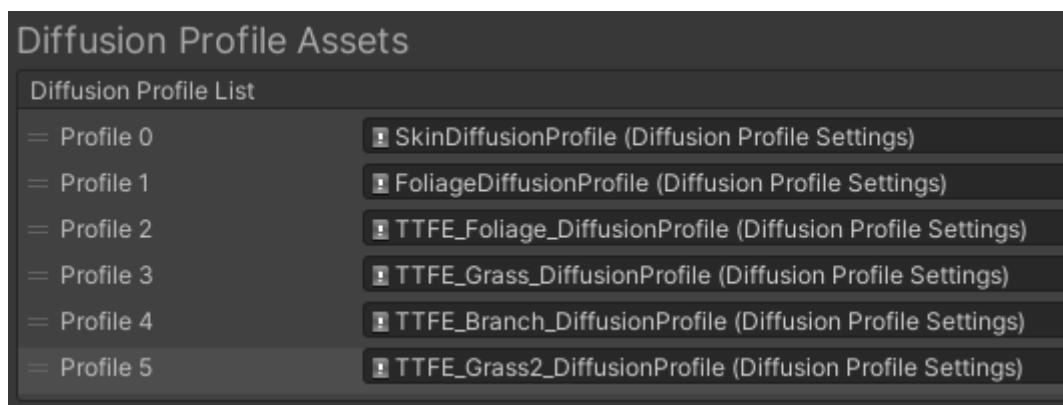


## High Definition Render Pipeline (HDRP)

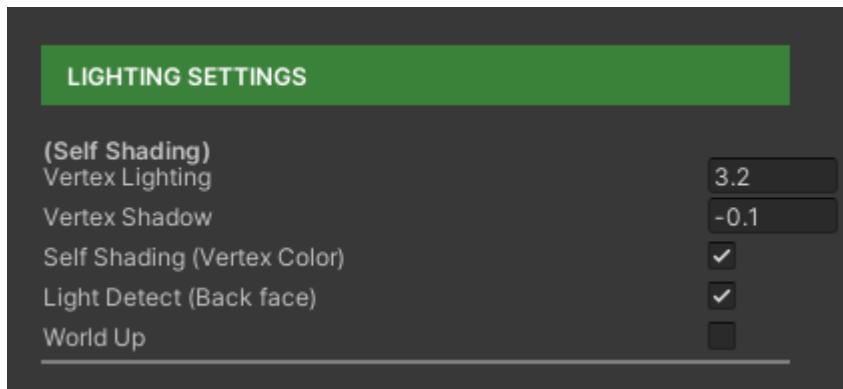
The models in the scene appear unnaturally green or pink. This issue occurs when the diffuse profile is not yet added to the HDRP asset. Select the affected material in the Unity Inspector to resolve this issue. When selected, you will see a warning window under the diffusion profile field in the inspector. Click the "fix button" to add the profiles to your project.



You can manually assign all the diffusion profiles in the "HDRP Global Settings" by going to:  
**"Edit -> Project Settings -> Graphics -> HDRP Global Settings -> Diffusion Profile Assets"**

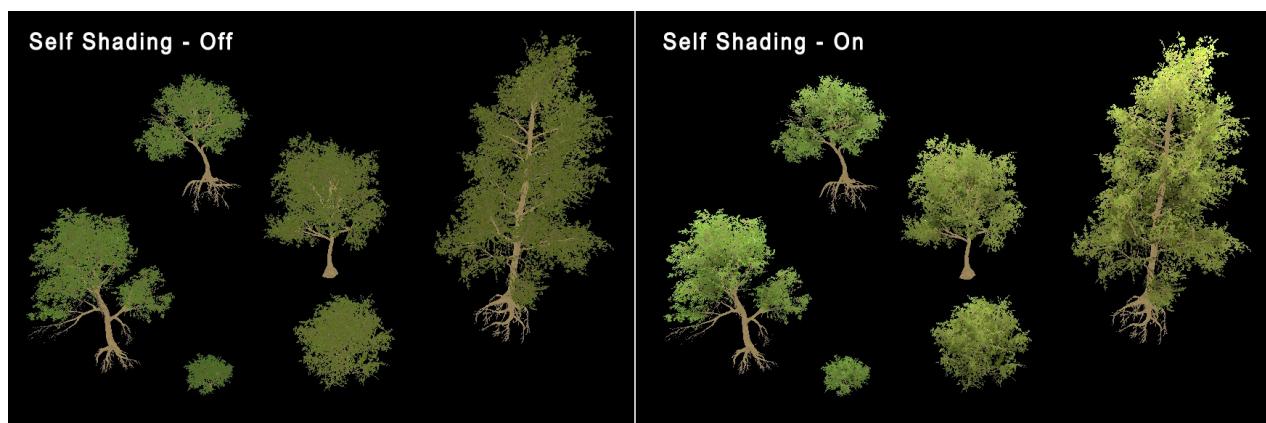


## Lighting/Shading Settings



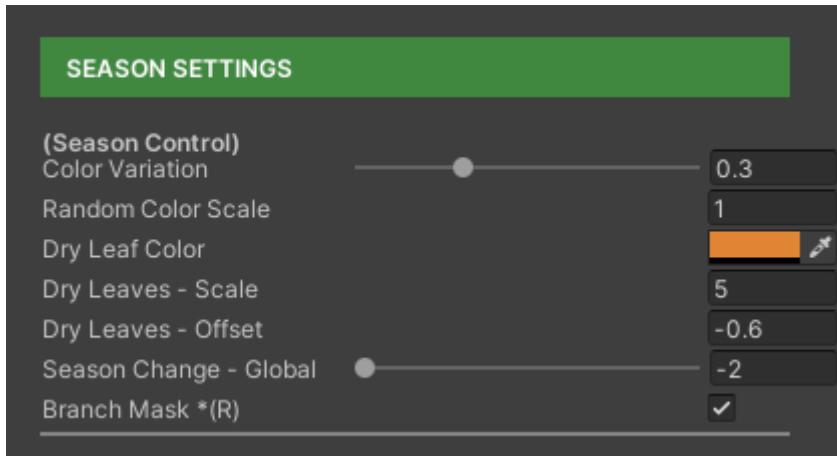
- **Self Shading**

Get some extra shading out of your foliage for a more stylized look. Vertex position scale and offset control tree shading and lighting.



- **Vertex Lighting**  
Adds vertex position-based lighting.
  - **Vertex Shadow**  
Adds vertex position-based shading.
  - **Self Shading (Vertex Color)**  
Turns the self-shading On/Off.
  - **Light Detect (Back face)**  
Addresses the common issue of dark back faces in Unity by re-calculating vertex normals. This option is enabled by default.
- \* Only available for trees in the TTFE Light version.
- **World Up (Mobile Shading)**  
Foliage with simple "vertex position Y shading" ("world up" toggle). Shading adapts to terrain surfaces for a natural look. It's usually used on grass and is more optimal for performance.

## Season Settings



- **Season Control**

Color and seasonal changes are based on vertex and world position.



- **Color Variation**

Each plant will have a random color based on world position.

- **Dry Leaf Color**

Leaf color in fall.

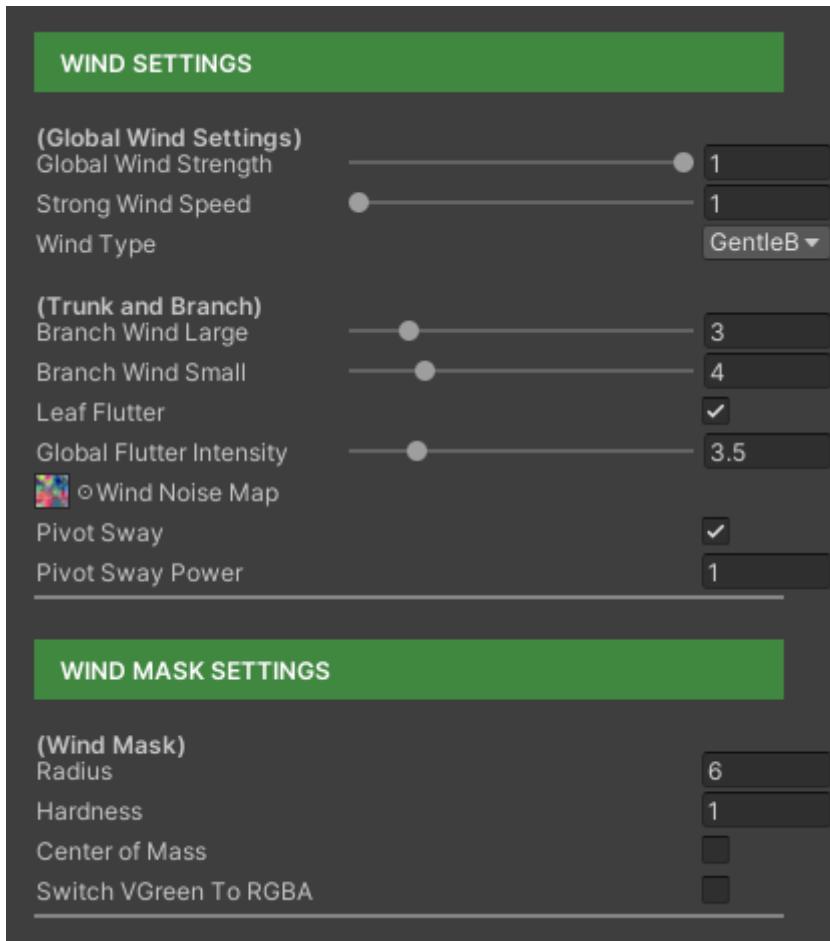
- **Dry Leaves – Scale and Offset**

Adjust the radius of the dry leaves.

To lock the season color so it can't be changed with the (TTFE) GLOBAL CONTROLLER, set "Dry Leaves - Scale and Offset" to "0,0" (default values are "1,1"). This is useful for maintaining dry or uniquely colored trees in any season.

- **Season Change - Global**  
Adjust the intensity of the seasonal change. Adjusts the 'Season' slider in the global gizmo.
- **Branch Mask \*(R)**  
Prevent seasonal color changes in texture atlas branches with masking (R-channel).

## Wind Settings



- **Global Wind Settings**

Correlates directly to the (TTFE) Global Controller sliders.

- **Global Wind Strength**  
Adjust the intensity of the main wind.
- **Strong Wind Speed**  
Adjusts the wind speed.
- **Wind Type**  
Use the dropdown to change between wind phases.

- **Trunk and Branch**

Adjust and customize branch and leaf settings.

- **Branch Wind Large and Small**

Adjust the large and small wind noise of your branches. You will need to adjust branch bending intensity as needed, as it may not be visible in some cases.

- **Simple Pivot Sway**

Best fitted for tall trees and pines with cylindrical shaped bark (prevents strange bark wobble).

- **Pivot Sway Power**

Controls the intensity of the swaying movement.

- **Grass Sway Power (Per Material) – Grass Shader**

Controls the intensity of the swaying movement of grass.

- **Leaf Flutter**

Turns on leaf fluttering.

- **Global Flutter Intensity**

Adjust the intensity of the flutter.

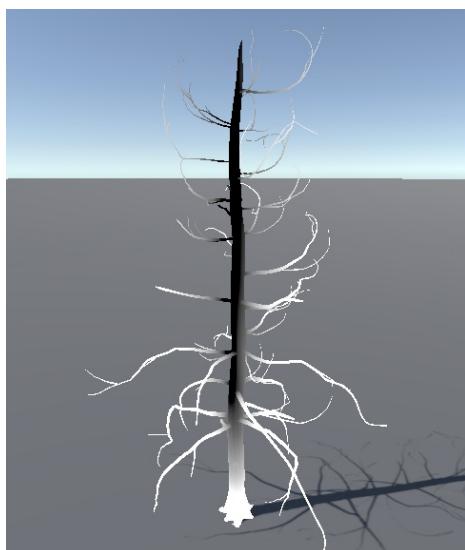
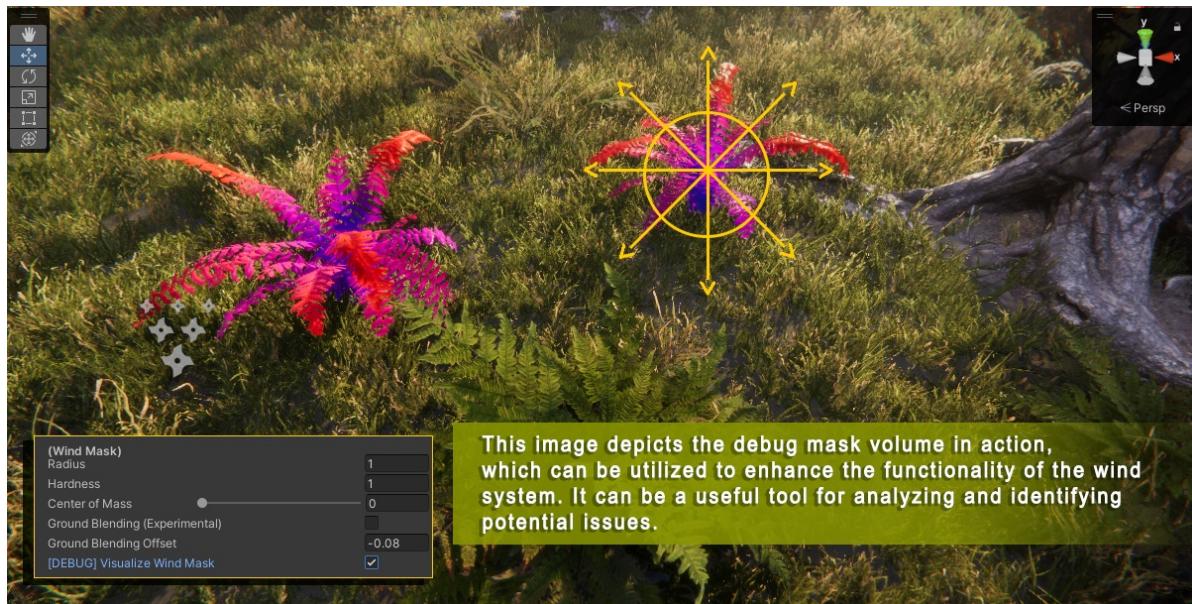
- **Wind Noise Map**

Internal 2D texture noise (3-vector RGB) generates an intense wind pulse (only available for “Strong Wind” type). The effect will be turned off if this texture is missing.

\* Only available on trees.

- **Wind Mask**

The meat and potatoes of the Toby Foliage Engine, the wind mask is essential for customizing how your trees behave in different wind conditions. A vertex-position volume sphere, masking out the wind on your trunk and leaves. Its radius can be controlled via the material settings.



\* The wind mask is rudimentary and limited, so it may not be perfect. Additional options are available.

**Visualizing wind mask is not available in the light version!**

- **Radius**  
Adjusts the volume size to mask wind in your material settings.
- **Hardness**  
Adjusts the intensity of the volume mask.
- **Center of Mass**  
Creates a center of gravity making the mesh less affected by wind in its center.
- **Switch VGreen to RGBA**  
Uses all available vertex color channels.

# Other Shaders

## (TTFE) Tree Billboard

Identical to the tree foliage shader but with fewer features.

## (TTFE) Roots

Basic root shader.

## (TTFE) Simple Cliff Masked

Basic cliff shader with grass coverage and correct edge slope angle.

It includes three material layers: ‘Rock Layer’, “Dirt Layer”, and “Coverage Layer”, plus additional “Detail Textures”.

# Optimization

All shader nodes have been thoroughly tested and optimized for performance. While the scenes and models may not be ideal for mobile devices, they serve as an excellent benchmark for testing TTFE against other shaders.

The following features may require optimization or disabling for mobile devices:

- **All Specular and Specular Backface Occlusion Features**

In Built-in and URP, this feature has a relatively light performance cost but should be avoided whenever possible. *\*To optimize, remove this feature from the shader code and set smoothness to 0.*

- **Light Detect (Back face)**

This option is enabled by default and could be expensive. “Mobile Shading (World Up)” is recommended for the best performance.

- **Translucency**

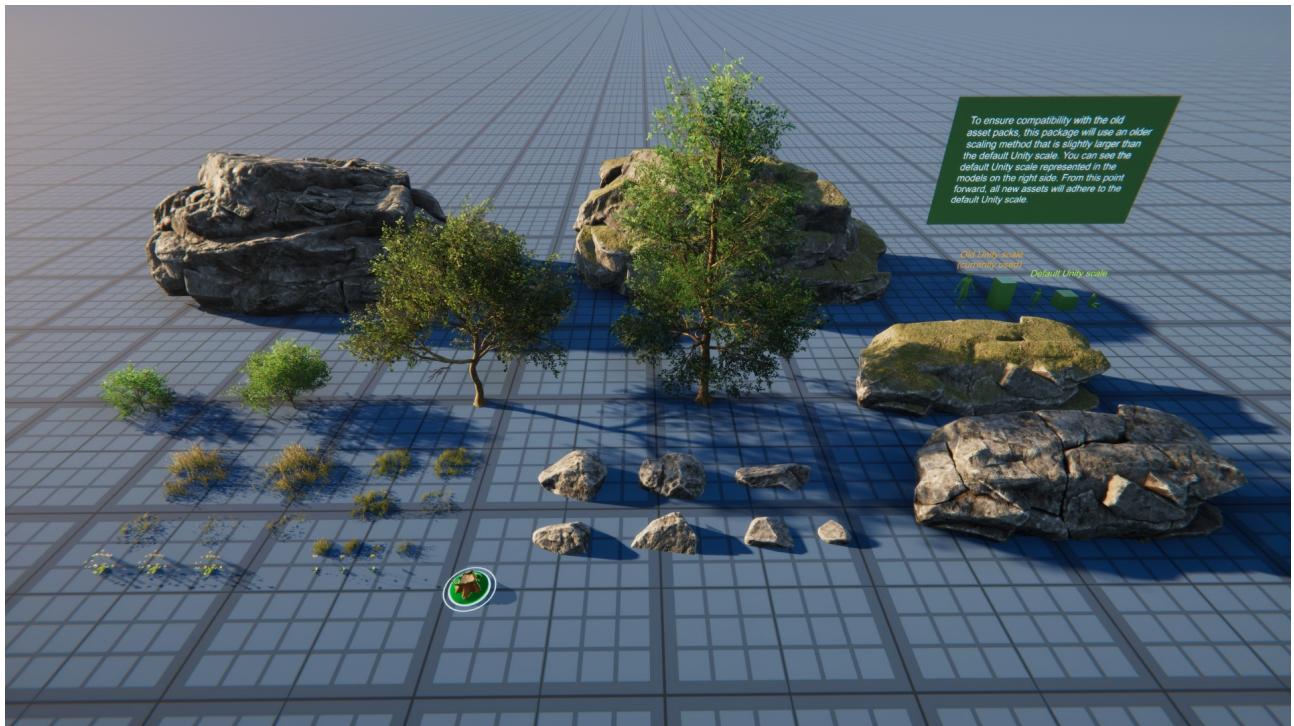
“World Up” and “Light Detect Backface” illuminate the backface, while translucency simulates subsurface scattering. In Built-in and URP this feature can be disabled on mobile devices without significantly affecting visual fidelity. *\*Remove this feature from the shader code!*

## GPU Instancing

TTFE runs very well without instancing, so manually turn on instancing on your materials as it is disabled but won't work without third-party instancers.

# Original Assets

The Toby Foliage Engine includes a range of unique art assets that are well-optimized and game-ready, all of which are original.



# Credits

The "Specular Backface Occlusion" and "Light Detect (Back face)" systems were developed with contributions from [Hybris Factory](#)

Special thanks to [Hybris Factory](#) for helping me develop the "Translucency Occlusion" system.

[https://www.artstation.com/ivan\\_janik](https://www.artstation.com/ivan_janik)

# Acknowledgement

Inspired by [The Visual Engine](#) By [BOXOPHOBIC](#).