Usage Tips

⋮

To help you make the most effective use of the Dynamic Mesh Combiner (DMC) and avoid potential issues, here are some useful tips

Object destroy

When you remove a baked animated mesh using the `Destroy()` function, there can be a delay between the `Destroy()` and `OnDisabled()` calls in the Dynamic Mesh Combiner (DMC) system. During this time, the Skinned Mesh Renderer may still attempt to render the baked mesh with its associated bones. However, if a bone is missing, the vertices of that bone will be rendered at the zero coordinates until the mesh is re-baked.

To address this issue, there are a couple of approaches you can take:

1. Hide the object and wait for the mesh to be rebaked: By hiding the object, you can prevent the incorrect rendering of the vertices associated with missing bones. Once the mesh is rebaked, you can then safely remove the object without any rendering artifacts.

2. Use the `obj.gameObject.LazyDestroy()` function: This function hides the object and schedules its removal after a 10-second delay. By using this approach, you allow the mesh to be rebaked during the waiting period, ensuring that the rendering is correct before the object is finally removed. It's important to note that if you're using `OnDestroy()` callbacks, they will also be called with the same 10-second delay.

Consider these options to address the issue of rendering artifacts when removing baked animated meshes in the DMC system. Choose the approach that best fits your specific requirements and ensures the proper rendering and removal of objects in your Unity project.