Realistic Mesh Deformation

First of all, thank you for purchasing and using **Realistic Mesh Deformation (RMD)**!

RMD has the same deformation method as used in Realistic Car Controller. More customizable and more stable version. Can be used on any object, but wireframe topology must be at good condition.

RMD uses vertices of the meshes and changes their positions relative to the collisions. Depending on the collision impact, vertices will be moved to the collision direction. And therefore, mesh will be deformed.

Settings

RMD uses these public variables;

deformResolution: Resolution of the deformation. If your model is a high poly model, you can reduce the resolution for good performance.

deformRadius: Radius of the deformation. On higher values, bigger radius of the mesh will be deformed on collisions.

deformMutliplier: Multiplier factor of the collisions. On higher values, mesh will be deformed much easier.

randomizeVertices: Randomizes moved vertices on collisions.

maximumDeformation: Maximum deformable distance per vertex. On higher values, mesh will be deformed much more.

minimumDeformationImpulse: Collision impacts below this point will not be processed. We don't want to process minimal collisions.

recalculateNormals / recalculateBounds: On each collision, surface of the mesh will be deformed and each polygon will start to face another direction. We have to recalculate normals of the meshes for proper smooth lighting on deformed surfaces. Have effect on performance as well.

collectedMeshes: Determines which meshes of the model will be deformed or not. Clicking "Add All Meshes" button will take all meshes and submeshes of the gameobject.

overallDeform: Determines how many of the vertices are moved.

latestProcessedVertexCount: Determines how many vertices are moved on the last collision.

Adding / Removing Deformable Meshes

Each RMD has collected meshes. You can click "**Add All Meshes**" button to take all meshes and submeshes of the gameobject. And then you can remove specific meshes in the list. For example, if you don't want to deform interior mesh of the car, you can exclude it by clicking "**X**" button.

Repair State / Deforming State

RMD has two enums for repairing and deforming states. On collisions, repair state will be forced to "**Repaired**", and deforming state will be forced to "**Deforming**". On repairs, repair state will be forced ro "**Repairing**" and deforming state will be forced to "**Deformed**".When the state is repairing, it means "**UpdateRepair**" method of the RMD_Deformation script is running. Meshes in collected meshes will be stored back to the original. When the state is deforming, it means "**UpdateDeformation**" method of the RMD_Deformation.cs is running. Used for running methods and make sure they are not conflicting each other. We don't want to deform the meshes while repairing them.

Using API

API has two main static methods below;

RMD.Repair(RMD_Deformation deformedObject);

Repairs target deformed object.

RMD.RepairAll();

Repairs all deformed objects in the scene.

How To Use

All you have to do is adding **RMD_Deformation** script to your gameobject. If your gameobject has child gameobjects, it will take their meshes too. When you added the script to your gameobject, it will take all meshes at first. If you want to exclude specific objects, you can remove them in the collected meshes list. Simply clicking “**X**” button will remove the corresponding mesh from the list.

Problems

Deformation is not working on collisions;

Be sure your gameobject has proper collider. If it has child gameobjects, they must have proper colliders too.

Deformation is not strong enough;

Increase radius, multiplier, and maximum limit.

Deformation is killing performance;

Reduce resolution for high poly models. Amout of vertices has an impact on performance.

Deformation is not working on child objects / other objects

Make sure they are added in the list named collected meshes.

Deformation is not accurate, only edges are deforming;

Because your model has low amounts of vertices. If flat surface of your model doesn’t deform, it means it hasn’t vertices at that surface.

Getting errors about missing meshfilters or meshes;

Seems like one or more of collected meshes are deleted after taking them to the list. Collected meshes list must not contain any missing meshfilters or meshes. Expand collectible meshes at your inspector panel, and delete missing ones.

Support

Skype: BoneCracker Games

Gmail: bonecrackergames@gmail.com

Mail: contact@bonecrackergames.com