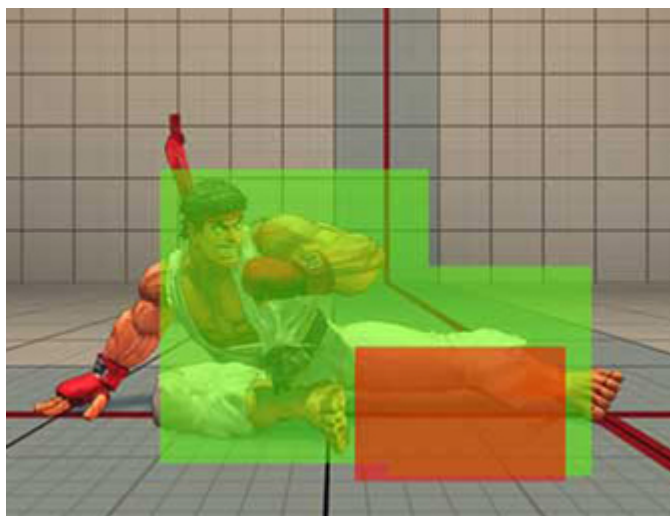


Hit Box Setup

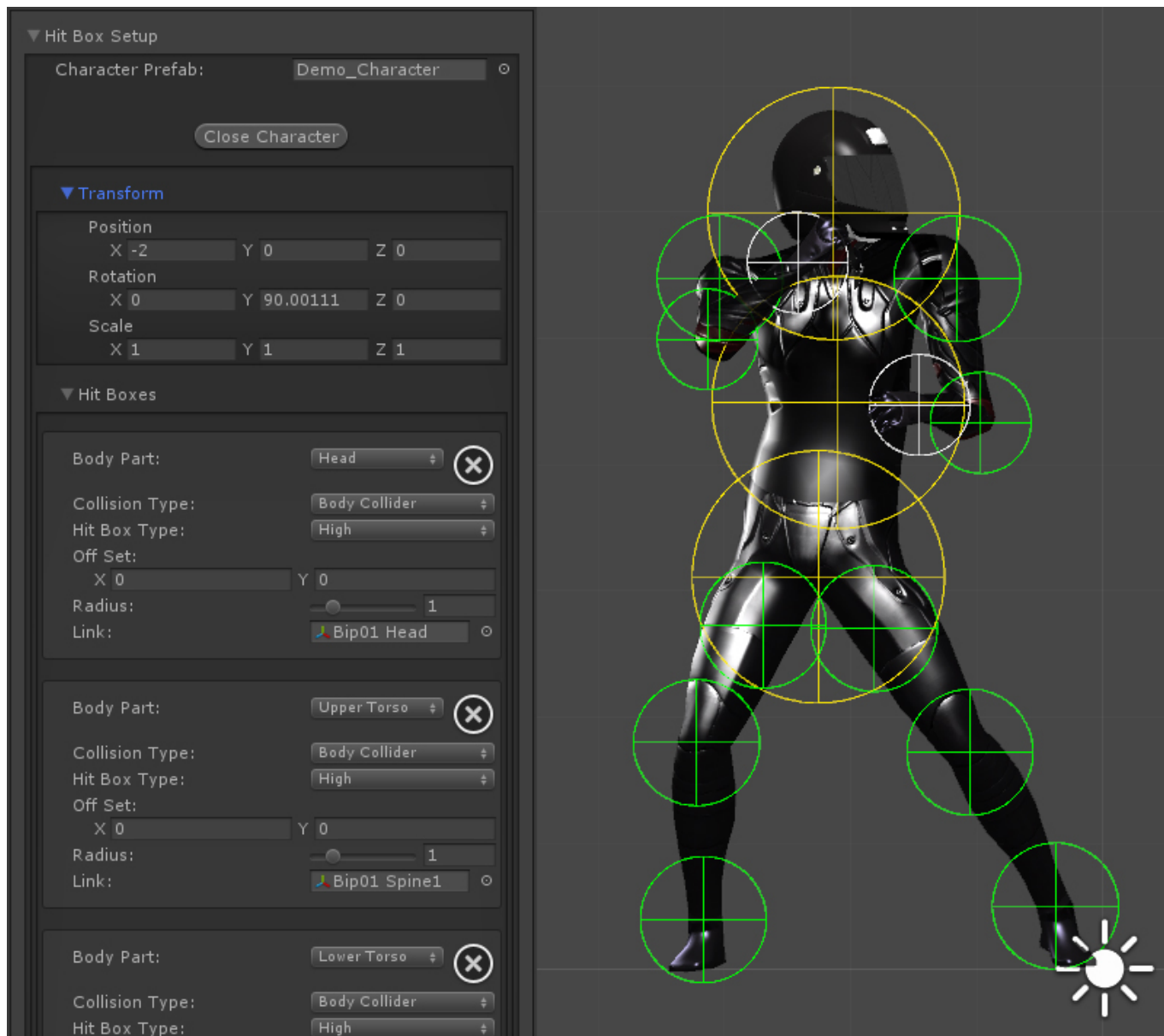
One of the key ingredients of any fighting game are hit boxes. Different than other action games, 2.5D fighting games rely on precision hits in a more “mechanical” way, as it emulates the sensation of playing a 2D game with the precision (and impact) of well adjusted invisible hitboxes.

There are many types of hitboxes, around, and thanks to the recent use of 3D technology to emulate 2D game play, hitboxes are commonly known for following body joints. Here are some examples:



UFE uses a similar technology, in which it detects the distance between 2 points in a projected 2D environment and based on the defined radius of the hitbox, determine rather or not a hit should be confirmed.

Similar to Marvel vs Capcom 3 series, UFE uses circles as hitboxes as it uses less processing power and gives you a wider variety of game play to work with.



Character Prefab

Choose a character prefab: A prefab containing the following components

- Animation
- (SOURCE VERSION) Hit Boxes Script (.\\UFE\\Scripts\\HitBoxesScript.cs)
- (OTHER VERSIONS) Hit Boxes Script (.\\UFE\\Plugin\\UFE.dll\\HitBoxesScript)

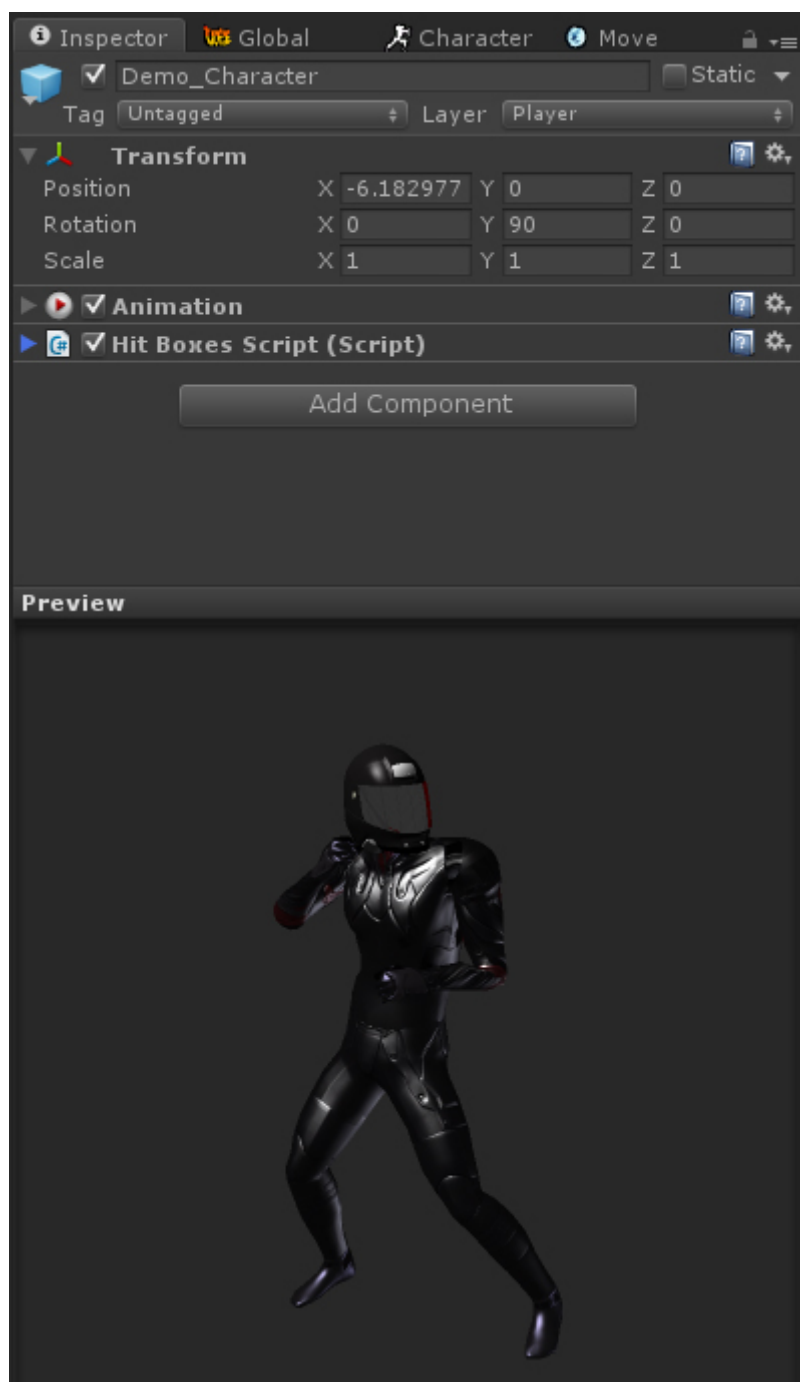
or

Create a character prefab: Create a character prefab from your character model

- Drag your character model from your project folder to the **Hierarchy**
- If it doesn't have the animation component yet, click on **Add Component**, than search for Animation. No further configuration is necessary.
- Using the Project window, navigate to .\\UFE\\Scripts\\HitBoxesScript.cs ¹⁾ and, with the character still selected, drag the script to the Inspector window. No further configuration is

necessary.

- Once you're done, drag the character back from Hierarchy to your desired folder in the Project window. Unity will automatically create a prefab.



Now just drag your nearly created prefab to the *Character Prefab* field under the Character Editor. Once you have your character in, you can click on "Preview" to see the character. Now lets configure the hitboxes.

Transform

Works similar to Unity's default Transform component. Adjust your character's rotation so it looks like

its standing on the left side of the screen.

Notes:

- Avoid changing the character's position as it will save its "local position" and it might cause some visual bugs in the future.
- Even though you can use the standard t-pose as the default animation when importing your character, its recommended that you use the idle animation as your character's default pose/animation. It makes it easier to work with hitboxes, specially if you adjust the character rotation.
- Make sure you adjust the characters rotation as you would like to see them on screen. 2.5D Fighting games tend to have more "theatrical" pose, in a sense that their body language are constantly facing the camera (or "audience"), while their heads are facing the opponent.
- Don't worry about mirroring your animations. UFE will take care of that for you.

Hit Boxes

Tag your character's body parts to the hit boxes, set its radius and adjust its behavior.

- **Body Part:** A tag representing the body part inside UFE.
- **Collision Type:**
 - Body Collider (yellow hitboxes): Movement and hit collision. Opposing yellow hitboxes cannot overlap.
 - Hit Collider (green hitboxes): Hit collision only. Opposing green hitboxes can overlap.
 - No Collider (white hitboxes): No collision. Use these for body parts that are not considered hitboxes but can be a [hurtbox](#) if needed.
- **Hit Box Type:**
 - High: If a non-knockdown attack hits this hitbox, Get Hit High animation group is triggered.
 - Low: If a non-knockdown attack hits this hitbox, Get Hit Low animation group is triggered.
- **Off Set:** Off set the hit boxes around the character body if needed. Be careful with high values, as different animations can cause unexpected positioning.
- **Radius:** How big the hitbox is. Its recommended that yellow hitboxes cover as much ground as possible to avoid unexpected cross-overs during battle.
- **Link:** The body part related to the character on screen. Navigate through the mesh and select the bone that best cover the selected body part.

Note: After the last step if you can't see the hitbox in scene view, click on Gizmos (top right corner of scene view) and toggle "HitBoxesScript".

Code example:

```
void OnHit(HitBox strokeHitBox, MoveInfo move, CharacterInfo hitter){  
    if (strokeHitBox.type == HitBoxType.low) Debug.Log ("low hit!");  
}
```

Related Video Tutorial:



Video

[< Back to Character Editor](#)

1)

.\UFE\Plugin\UFE.dll\HitBoxesScript if you are not using the source version.

From:

<http://www.ufe3d.com/> - **Universal Fighting Engine**

Permanent link:

<http://www.ufe3d.com/doku.php/character:hitbox?rev=1391060773>

Last update: **2014/01/30 00:46**

