**Name**: 3DText

**Version**: 1.1

**Purpose**:

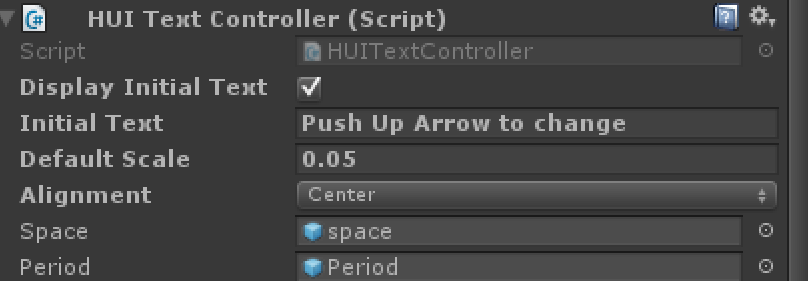
A package for dynamically setting 3D text.

**Description**:

This package is mostly a collection of models for characters in the alphabet. The primary object is the HUIText (HUI standing for Holographic User Interface) prefab which has references to all character models. It’s script, HUITextController has the public method SetText to create the necessary text.

**Setup Instructions**:

Place the HUIText prefab where you want it. In the HUI Text Controller properties, either set the Initial Text property and set the Display Initial Text flag set to True:



Or have your scripts find its HUITextController script of an object and call the SetText method of the HUITextController:

public class ExampleHUITextSceneController : MonoBehaviour

{

public GameObject HUITextGameObject;

private HUITextController huiTextController;

// Use this for initialization

void Start()

{

huiTextController =

HUITextGameObject.GetComponent<HUITextController>();

}

// Update is called once per frame

void Update()

{

if (Input.GetKeyDown(KeyCode.UpArrow))

{

huiTextController.SetText("Push Down Arrow to change");

}

else if (Input.GetKeyDown(KeyCode.DownArrow))

{

huiTextController.SetText("Push Up Arrow to change");

}

}

}

Scale can be changed through calling SetText with a scale or through setting Default Scale in the HUITextController properties.

**Example Scenes**:

ExampleScene.unity shows using the HUIText prefab both with initial text and with dynamically changing the text in the ExampleHUITextSceneController script.

**Sources**:

* Custom Maya Models
  + Alphabet

**Version Changes**:

* 1.1:
  + Vertical Alignment
  + Scale types:
    - Bounded meaning the letters won’t extend beyond this total size
      * MaxScale is the biggest it will scale