Reflect the 3D face on unity

1. Take a photo of three faces from your webcam

(Python from here)

1. Make a Mediapipe UV map for each photo
2. For the UV map in the front photo, apply a mask to the center of the face and get only the middle UV map. Also, in the photo of the left face, only the UV map where the left face is reflected is acquired from the mask. The same applies to the right face.
3. Combine each UVmap obtained by applying the mask as one UVmap.
4. Perform skeleton estimation using the frontal photograph used in the above step, and save the result (x, y, z coordinates) in csv.

(Blender from here)

1. Load from Scripting in blender using the saved csv
2. Transform the basic mesh of the canonical\_face model object of the mediapipe with the x, y, z coordinates of each landmark point read from the Csv file.
3. Read the UVmap created for the face object that reflects the result of skeleton estimation, read it with texture, and apply it.

Run on mediapipe

1. Import the created face object in the FaceMesh Scene of Mediapipe UnityPlugin.

2. Get the Landmark Point from Mediapipe UnityPlugin.

3. Since the acquired landmark points are normalized numerical values, determine the scale as appropriate to obtain the coordinate values.

4. Link the coordinate values with the mesh vertices of the face object to reflect the result of skeleton estimation from the mediapipe.

Setup video

MediapipeUnityPlugin setup

Windows reference

url: [Installation Guide · homuler/MediaPipeUnityPlugin Wiki · GitHub](https://github.com/homuler/MediaPipeUnityPlugin/wiki/Installation-Guide)

Mediapipe python version setup

Command input with cmd

pip install mediapipe

pip install opence-python

Run from sample code

url: [Face Mesh - mediapipe (google.github.io)](https://google.github.io/mediapipe/solutions/face_mesh.html)

Blender setup

Download from homepage

url: [Download — blender.org](https://www.blender.org/download/)

Setup development environment document

Spec requirement : Works with cpu only

Machine specs :

Programing language : python, C#

HDD storages

3. Software

mediapipeUnityPlugin

Note: gpu mode is not available on windows and mac

Version: v0.8.2 (confirmed 2022 / 04/15, latest version: v0.9.0)

Unity 2020.03.30f1, Mediapipe 0.8.9 is used in Plugin

See url for details

url: [Installation Guide · homuler/MediaPipeUnityPlugin Wiki · GitHub](https://github.com/homuler/MediaPipeUnityPlugin/wiki/Installation-Guide)

Mediapipe

Version: 0.8.9.1 (confirmed on 2022/04/15)

Dependent library is url

url: [Installation - mediapipe (google.github.io)](https://google.github.io/mediapipe/getting_started/install.html)

blender

I used 3.1.0, but other versions look good

1. Problem, Issue, Challenge, Progress

* In Mediapipeunity v0.8.3, unityEngine.Color and mediapipe.Color conflict, so I think it is better to use v0.8.2 or v0.9.0.
* (It looks good because the license of the original code was free for confirmation, but I would like other people to confirm it)
* When importing Obj or fbx 3D object files into Unity, there is a problem that the index in the file changes automatically. From now on, the landmark point of the mediapipe and the vertex of the face object cannot be matched, and it becomes a mess.
  + Solution (there may be other ways, but now as a solution)

1. Import the 3D model file once on unity, and if the index changes, export it on unity and save it.

2. Import the 3D model file saved with the index changed and the original 3D model file into blender together, load the Geometry Nodes, and save the index and coordinate values.

3. Create a program that associates indexes with the same coordinate values ​​and obtain the corresponding dictionary.

Finished task

At present, at the stage, it is now possible to create a photo uvmap with Mediapipe one by one. However, there is code (load\_obj, write\_obj) in this code that may catch the license, so you need to rewrite that part.

We also demonstrated that you can create a face by transforming the vertices of a 3D face object from the landmark points of the mediapipe.

Current task

After this, rigging with your face on unity

Prepare to do task

It is assumed that the face part information will be dynamically acquired from the media pipe to create a more accurate face.

1. List the reference video or resource

See Uvmap generated code

[GitHub - apple2373/mediapipe-facemesh](https://github.com/apple2373/mediapipe-facemesh)

See face and rigging on Mediapipe and unity

[UnityでMediaPipeを実行してVTuberになる - Qiita](https://qiita.com/yamatohkd/items/b2d9a6055761b0b2c369)