

Unity Virtual Home README

San Jose State University, Department of Computer Engineering

Author: Adam Goldstein

Project Advisor: Dr. Harry Li

03/01/2023	Created Document	Adam Goldstein

Github Source Link: https://github.com/xavierpuigf/virtualhome_unity

Github Fork: <https://github.com/adkap2/virtualhome>

Website: <http://virtual-home.org/>

Table of Contents

I.	Introduction
II.	Installation
III.	Python to Unity Communication
IV.	In-home tasks

I. Introduction

The Unity VirtualHome Readme discusses the steps necessary to install and run the VirtualHome program within Unity. This allows for the simulation of multiagent tasks within numerous home-like environments.

Agents in this environment are represented as human-like aviators which interact with the environment through high level instructions. The virtualhome contains a knowledge base allowing for an extensive set of instructions to perform various tasks.

This Readme is comprised of three parts

1. Installation instructions
2. Initial Unity to Python communication and VirtualHome rendering
3. Performing basic in-home tasks with the VirtualHome Agent

II. VirtualHome Installation Instructions

Note: This code was performed and tested using Ubuntu 20.04 and UnityHub 3.4.1

1. Install UnitySimulator

a. Clone the Github Repository

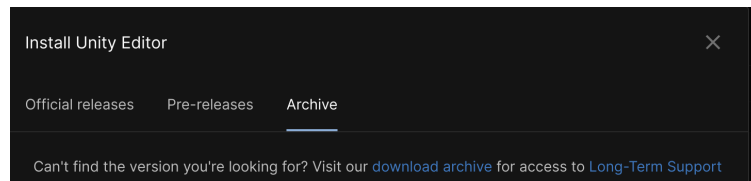
```
cd ~/Project directory
git clone https://github.com/adkap2/virtualhome.git
```

b. Install Unity version 2021.32.19f1

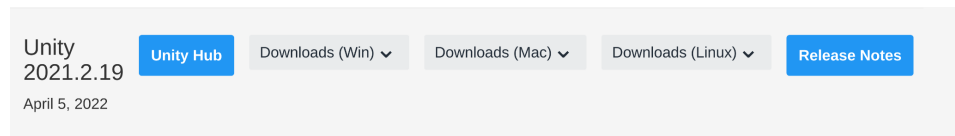
i. From UnityHub, go to installs, install editor



ii. Go to Archive, Download Archive



iii. Go to Unity 2021.2.19 and select UnityHub



c. Install Necessary Assets

i. Purchase Complete Home Interior Pack

<https://assetstore.unity.com/packages/3d/props/interior/complete-home-interior-pack-31049>

ii. Purchase Final IK

<https://assetstore.unity.com/packages/tools/animation/final-ik-14290>

iii. Purchase DunGen

<https://assetstore.unity.com/packages/tools/utilities/dungen-15682>

d. Add Project

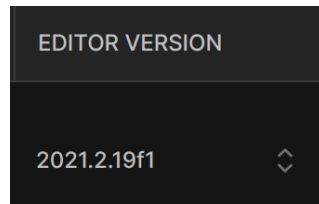
i. Go to Projects, Open



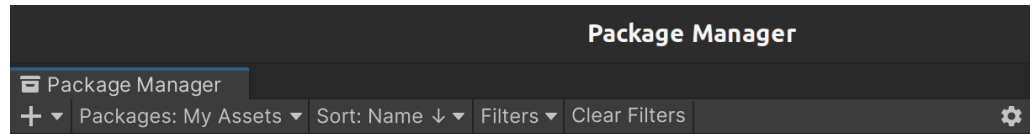
- ii. Select Add project from disk



- iii. Select the cloned github directory from 1.
- iv. Under Projects, Editor Version, Select 2021.2.1.19f1



- e. Add assets to project
 - i. In the Unity Editor, go to Package Manager
 - ii. Select My Assets



- iii. You should see the 3 assets you purchased.
- iv. For each Asset, click Download
- v. Click Import
- f. Open a scene in VirtualHome
 - i. File > Open Scene > Assets/Story Generator/Scene/Scene_0.unity

2. Install VirtualHome Python API

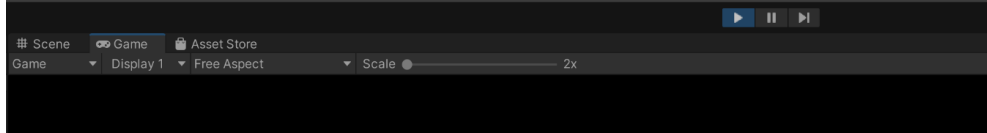
- a. Create Python 3.7 Environment

```
conda create --name unityvhome python=3.7
conda activate unityvhome
```

```
pip install virtualhome
```

III. Python to Unity Communication

1. From Virtualhome Unity, Select Play mode



2. Open a new Jupyter Notebook

```
code virtualhomescript.ipynb
```

3. Import Virtualhome

```
%matplotlib notebook
import virtualhome
from unity_simulator.comm_unity import UnityCommunication
```

4. Create a communication object while Unity is in play mode

```
# Connect to virtual home simulator
comm = UnityCommunication(timeout_wait=1000)
```

IV. Virtual Agent Basic In-home Tasks

1. Reset the scene and retrieve scene 3

```
comm.reset(3)
s, g = comm.environment_graph()
```

2. Add Agent and get nodes of common Objects

```
# Add a agent
comm.add_character('Chars/female2', initial_room='kitchen')

# Get nodes for apple, desk, kitchen
apple = [node['id'] for node in g['nodes'] if node['class_name'] == 'apple'][1]
desk = [node['id'] for node in g['nodes'] if node['class_name'] == 'desk'][1]
kitchen = [node['id'] for node in g['nodes'] if node['class_name'] == 'kitchen'][0]
fridge = [node['id'] for node in g['nodes'] if node['class_name'] == 'fridge'][0]
tv = [node['id'] for node in g['nodes'] if node['class_name'] == 'tv'][0]
table = [node['id'] for node in g['nodes'] if node['class_name'] == 'kitchentable'][0]
microwave = [node['id'] for node in g['nodes'] if node['class_name'] == 'microwave'][0]
faucet_id = [node['id'] for node in g['nodes'] if node['class_name'] == 'faucet'][-1]

# get the light nodes in the scene
light_nodes = []

for light_node in [node for node in g['nodes'] if node['class_name'] == 'lightswitch']:
    light_nodes.append(light_node)
light_node = [x for x in g['nodes'] if x['class_name'] == 'lightswitch']
```

3. Generate and Run the Agent's Script to complete household tasks

```
script = [
    # Turn on all lights
    '<char0> [switchon] <lightswitch> {}'.format(light_nodes[3]['id']),
    '<char0> [switchon] <lightswitch> {}'.format(light_nodes[0]['id']),
    '<char0> [switchon] <lightswitch> {}'.format(light_nodes[1]['id']),
    '<char0> [switchon] <lightswitch> {}'.format(light_nodes[2]['id']),
    # Move the apple to the desk
    '<char0> [grab] <apple> {}'.format(apple),
    '<char0> [put] <apple> {} <desk> {}'.format(apple, desk),
    # open fridge
    '<char0> [walk] <fridge> {}'.format(fridge),
```

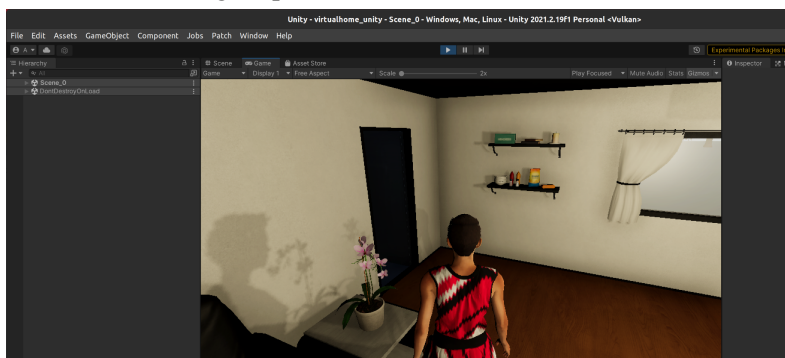
```

'<char0> [open] <fridge> ({}).format(fridge),
# close fridge
'<char0> [close] <fridge> ({}).format(fridge),
# turn on faucet
'<char0> [walk] <faucet> ({}).format(faucet_id),
'<char0> [switchon] <faucet> ({}).format(faucet_id),
# Walk to microwave
'<char0> [walk] <microwave> ({}).format(microwave),
# Open microwave
'<char0> [open] <microwave> ({}).format(microwave),
# close microwave
'<char0> [close] <microwave> ({}).format(microwave),
# Turn on Microwave
'<char0> [switchon] <microwave> ({}).format(microwave),
'<char0> [walk] <tv> ({}).format(tv),
'<char0> [switchon] <tv> ({}).format(tv),
# walk to faucet
'<char0> [walk] <faucet> ({}).format(faucet_id),
# turn off faucet
'<char0> [switchoff] <faucet> ({}).format(faucet_id),
'<char0> [walk] <kitchentable> ({}).format(table)
]

success, message = comm.render_script(script=script[0:20],
                                     find_solution=True,
                                     processing_time_limit=200,
                                     frame_rate=15,
                                     image_width=640,
                                     image_height=480,
                                     skip_animation=False,
                                     image_synthesis=['normal'],
                                     camera_mode=['PERSON_FROM_BACK'],
                                     recording=True)

```

4. View Humanoid agent perform tasks in real-time



Citation

- [1] Puig, X., Ra, K., Boben, M., Li, J., Wang, T., Fidler, S., & Torralba, A. (2018). Virtualhome: Simulating household activities via programs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 8494-8502).