From Unreal Engine to Mitsuba Renderer

Aysegül Peközsoy

Apr 29, 2025

CONTENTS:

CHAPTER

ONE

BLENDER_EXPORT MODULE

Extracts lights FBX scene and converts them to XML format, readable by Mitsuba.

`blender_export.`**`blender_to_mitsuba_vec`**`(`*v*`)`

Converts a Blender vector to Mitsuba coordinate format.

Parameters
`v` (*`mathutils.Vector`*) – Blender 3D vector.

Return str
Mitsuba-formatted vector as a string.

`blender_export.`**`clear_scene`**`()`

Clears the Blender scene to default empty settings.

`blender_export.`**`get_lights`**`()`

Retrieves all light objects from the current Blender scene.

Return list
List of Blender light objects.

`blender_export.`**`import_fbx`**`(`*path*`)`

Imports an FBX file into Blender.

Parameters
`path` (*`str`*) – Path to the FBX file.

Raises
`FileNotFoundError` – If the FBX file does not exist.

`blender_export.`**`main`**`()`

Main execution function.

`blender_export.`**`write_mitsuba_xml`**`(`*xml_path*, *lights*`)`

Writes light information into a Mitsuba XML file.

Parameters

- `xml_path` (*`str`*) – Path to save the XML file.
- `lights` (*`list`*) – List of Blender light objects.

Raises
`IOError` – If the XML file cannot be written.

CHAPTER TWO

MITSUBA_RENDER_H5 MODULE

Offers functionalities to render scenes given by OBJ objects and XML lights

`mitsuba_render_h5.load_emitters(xml_path)`

Loads emitters from a Mitsuba XML scene.

Parameters
xml_path (*str*) – Path to the XML file.

Return list[dict]
List of emitter dictionaries.

Raises
FileNotFoundError – If the XML file does not exist.

`mitsuba_render_h5.main()`

Main function that parses inputs and runs render_scenes_h5()

`mitsuba_render_h5.render_scene(obj_path, xml_light_path)`

Renders a scene given by OBJ and XML lights.

Parameters

- **obj_path** (*str*) – path to the OBJ file.
- **xml_light_path** (*str*) – path to the XML light file.

Return ndarray
rendered image as np.array

`mitsuba_render_h5.render_scenes_h5(scene_dir, out_path)`

Renders all scenes, given by OBJ and XML files in a directory and saves them as PNGs and HDF5.

Parameters

- **scene_dir** (*str*) – Path to the scene directory.
- **out_path** (*str*) – Path to the output directory.

CHAPTER

THREE

TEST_MITSUBA_RENDER_H5 MODULE

Unit tests for mitsuba_render.py.

`test_mitsuba_render_h5.`**`test_check_missing_files`**`()`

`test_mitsuba_render_h5.`**`test_load_emitters_missing_file`**`()`

`test_mitsuba_render_h5.`**`test_no_scenes`**`()`

Test if a warning is raised if scenes_dir does not contain any OBJ files

CHAPTER
FOUR

README

4.1 Mitsuba Rendering from Unreal Engine Export

This project allows you to:

- Export light positions from a **Blender** scene (originally imported from Unreal Engine).
- Render the 3D scene with **Mitsuba 3** using those lights.
- Save the results as **PNG images** and **compressed HDF5** files.
- Visualize all the rendered images easily.

4.2 Installation

This project was developed in Python 3.12, the compatibility with other versions cannot be guaranteed.

To install a `requirements.txt` is provided and the modules can be installed by

```
pip install -r requirements.txt
```

4.3 Project Structure

```
.
├── README.md
├── demo.ipynb               <- demo noteboook
├── demo_scene               <- contains demo scene objs and xmls
│   ├── scene10.obj
│   ├── scene10.xml
│   ├── scene11.obj
│   └── scene11.xml
├── images
│   └── workflow.png        <- workflow image for README
├── requirements.txt        <- pip requirements to install
├── src
│   ├── blender_export.py  <- script to run in Blender to extract lights and write xml
│   └── mitsuba_render_h5.py  <- renders a directory of scenes and saves as PNGs and HDF5
└── tests
    └── test_mitsuba_render_h5.py
```

4.4 Use

The use of `mitsuba_render_h5.py` is relatively straightforward and also illustrated in `demo.ipynb`:

One can either run

```
from src.mitsuba_render_h5 import render_scene
```

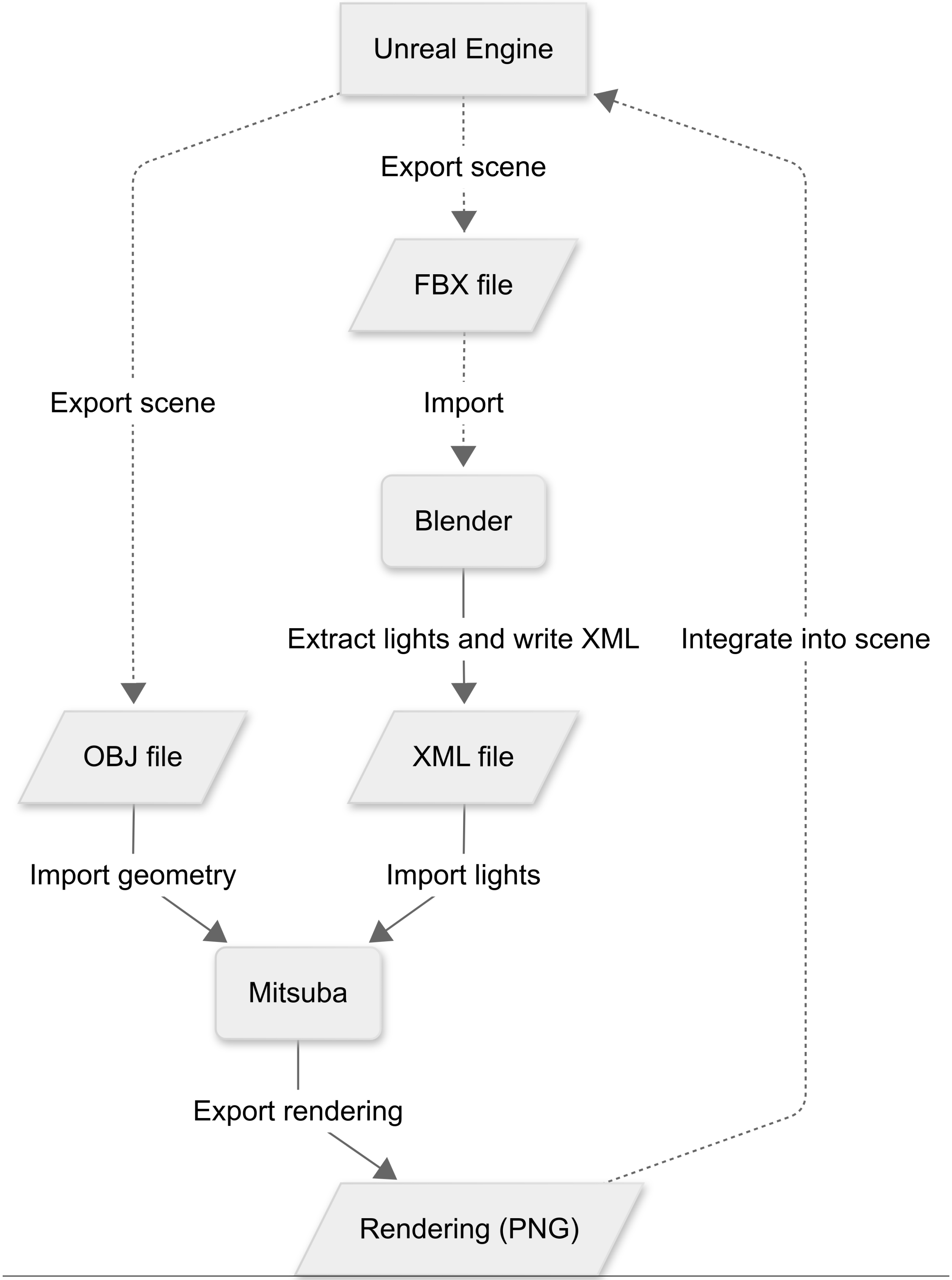
in another Python file or run `mitsuba_render_h5.py` in the command line by running e.g.

```
python src/mitsuba_render_h5.py --scene_dir demo_scene --out_path demo_scene
```

with the current working directory as defaults for the arguments `--scene_dir` and `--out_file`.

This will save the rendered png files, as well as the combined HDF5 file in the out_path directory

4.5 Workflow


Unreal Engine
Export scene
FBX file
Export scene
Import
Blender
Extract lights and write XML
Integrate into scene
OBJ file
XML file
Import geometry
Import lights
Mitsuba
Export rendering
Rendering (PNG)

PYTHON MODULE INDEX

INDEX