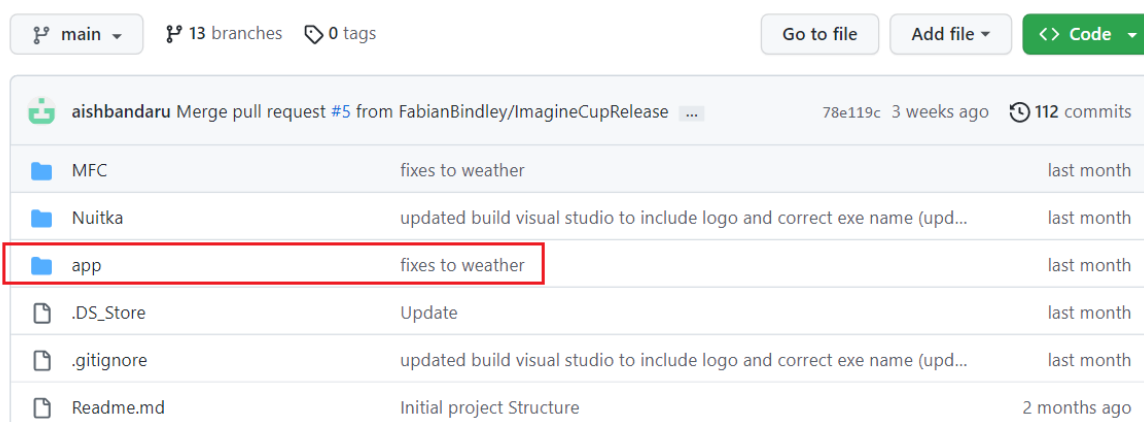


UCL Open-Illumiroom V2 Deployment Manual

Original Author: Fabian Bindley

System Development

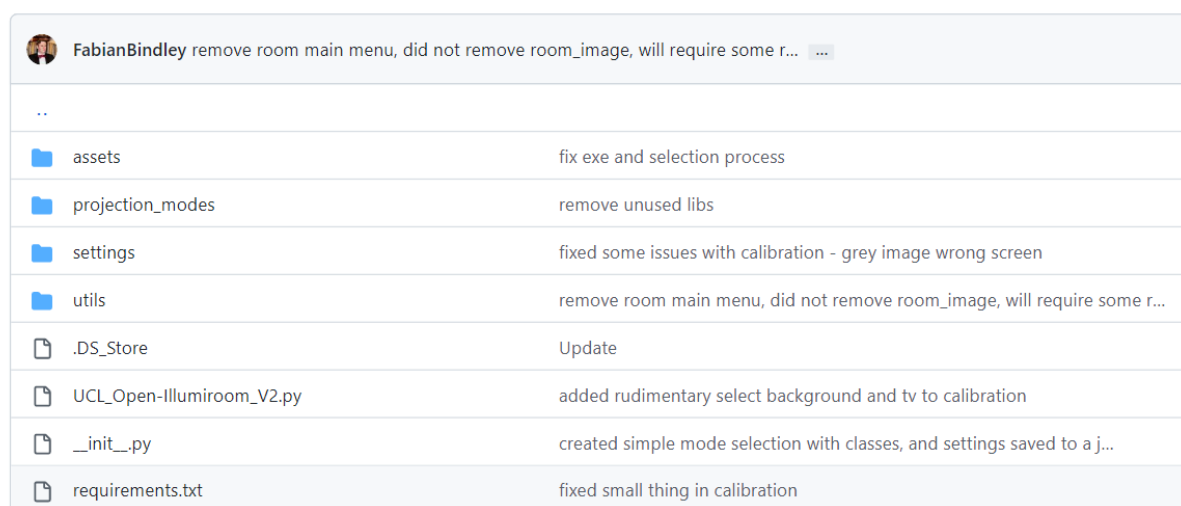
The UCL Open-Illumiroom V2 system source code is contained within the app folder in the root directory of the project's repository.



The screenshot shows a GitHub repository interface. At the top, there are buttons for 'main', '13 branches', and '0 tags'. There are also buttons for 'Go to file', 'Add file', and 'Code'. Below this is a table of files and folders. The 'app' folder is highlighted with a red box.

File/Folder	Description	Last Modified
MFC	fixes to weather	last month
Nuitka	updated build visual studio to include logo and correct exe name (upd...	last month
app	fixes to weather	last month
.DS_Store	Update	last month
.gitignore	updated build visual studio to include logo and correct exe name (upd...	last month
Readme.md	Initial project Structure	2 months ago

The *main* file for the app is the UCL_Open-Illumiroom_V2.py file, it contains the app's initialisation functions, and main loop. Projection mode classes, util classes, assets and settings are separated into 4 folders.



The screenshot shows a GitHub commit diff by FabianBindley. The commit message is 'remove room main menu, did not remove room_image, will require some r...'. The diff shows changes to several files and folders. The 'utils' folder is highlighted with a red box.

File/Folder	Description
assets	fix exe and selection process
projection_modes	remove unused libs
settings	fixed some issues with calibration - grey image wrong screen
utils	remove room main menu, did not remove room_image, will require some r...
.DS_Store	Update
UCL_Open-Illumiroom_V2.py	added rudimentary select background and tv to calibration
__init__.py	created simple mode selection with classes, and settings saved to a j...
requirements.txt	fixed small thing in calibration

The required Python libraries are included in the requirements.txt.

Compilation

To compile UCL Open-Illumiroom V2, the Nuitka library is used. Nuitka compiles Python source-code to C source code on a platform dependent basis.

To compile UCL Open-Illumiroom V2 to a standalone file that can be distributed, run the following bash command:

- `python -m nuitka --standalone --enable-plugin=tk-inter --enable-plugin=pyside2 --output-dir=release --remove-output --disable-console app/UCL_Open-Illumiroom_V2.py; cp -r app/settings release/UCL_Open-Illumiroom_V2.dist; cp -r app/assets release/UCL_Open-Illumiroom_V2.dist`

This will create a release folder in the root of the directory and place a UCL_Open-Illumiroom_V2.dist folder there with the compiled build, as well as an executable: UCL_Open-Illumiroom_V2.exe.

The assets and settings folders are also copied over to the .dist folder.

To compile a build with a console output for debugging, you may run the following bash command:

- `python -m nuitka --standalone --enable-plugin=tk-inter --enable-plugin=pyside2 --output-dir=release --remove-output app/UCL_Open-Illumiroom_V2.py; cp -r app/settings release/UCL_Open-Illumiroom_V2.dist; cp -r app/assets release/UCL_Open-Illumiroom_V2.dist`

The difference between the 2 commands is that the `--disable-console` flag is not present in the debugging command which allows the console output to show.

Additional notes on compilation:

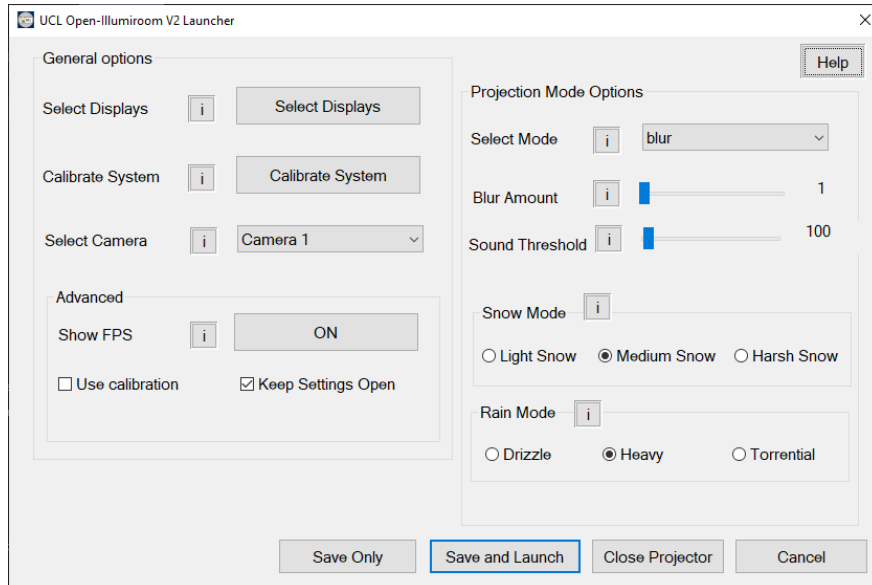
- Currently, the Wobble mode and Audio Capture containing the machine learning based audio detection cannot be compiled due to Nuitka not supporting the library Librosa (which is used to extract audio features). As a result, an older version of Wobble is compiled that does not perform automated audio classification. Therefore, instead of Wobble being triggered for targeted sounds like gunfire, it is triggered for any noise above a pre-defined threshold volume.
- The Nuitka folder in the repository contains more information about compilation, as well as the 2 bash commands.

MFC Development and Compilation

MFC C++ Settings application designed for Facial Navigation on Windows.

Original Author: Anelia Gaydardzhieva

Adapted for UCL Open-Illumiroom V2: Fabian Bindley



To Run

Please navigate to the MFC directory in the repo, then double click on the UCL Open-Illumiroom V2 Launcher.exe file

Structure

The structure of the MFC section mimics that of MotionInput v3.2 (MI_v3.2). The essential components from MI_v3.2 have been used to build our own MFC application files and folders. This allows the MFC application to run in isolation and ensures intentional performance when integrated with MI_v3.2.

- The UCL_Open-Illumiroom_V2.dist/settings folder stores general_settings.json and mode_settings.json. These are the settings files for UCL Open-Illumiroom V2.
- packages folder contains nlohmann.json.3.10.5 which is the JSON library used to read and write into the JSON files.
- The UCL_Open_Illumiroom_V2.dist folder contains the compiled code to run the system, including UCL_Open-Illumiroom_V2.exe, the main application executable.

Setup

1. In case you have not done so already, install Visual Studio (preferably v2022).

2. Open Visual Studio Installer.
3. Select Modify. If you have more than one version of VS installed, select Modify on the version you wish to access the code from.
4. Select Individual Components.
5. Find the box with the latest C++ MFC (e.g. C++ MFC for latest v142 build tools (x86 & x64)).
6. If the box is not marked with a tick, select it and save.
7. To modify the MFC code simply open *YourGithubRepo\MFC\data\UCL Open-Illumiroom V2\MFC-UCL-MI3-Settings.sln* file.

The MFC is called MFC-UCL-MI3 in places because this was the template used to create the UCL Open-Illumiroom V2 MFC. If possible, the naming should be updated in future versions, and old unused code and resources removed.

View/Modify MFC

Steps in VS:

View -> Resource View -> Dialog

Manual file navigation:

MFC-UCL-MI3-Settings.sln -> MFCUCLMI3Settings.rc -> Dialog

Access main file

MFC-UCL-MI3-Settings.sln -> MFC-UCL-MI3-SettingsDlg.cpp

Compile

1. Open MFC-UCL-MI3-Settings.sln
2. Set Solution Configurations to Debug or Release
3. Set option x64 or x86
4. Build Solution (Click green run icon on top bar)
5. The compiled .exe file is in root directory

Use the MFC .exe

1. Copy the .exe file from its position in the repo
2. Paste the .exe in the same directory as UCL Open-Illumiroom V2.dist