
RoomAlive Toolkit README

The RoomAlive Toolkit enables creation of dynamic projection mapping experiences. This toolkit has been in internal use at Microsoft Research for several years and has been used in a variety of interactive projection mapping projects such as RoomAlive, IllumiRoom, ManoAMano, Beamatron and Room2Room.

The toolkit consists of two separate projects:

- **ProCamCalibration** - This C# project can be used to calibrate multiple projectors and Kinect cameras in a room to enable immersive, dynamic projection mapping experiences. The code-base also includes a simple projection mapping sample using Direct3D.
- **RoomAlive Toolkit for Unity** - RoomAlive Toolkit for Unity contains a set of Unity scripts and tools that enable immersive, dynamic projection mapping experiences, based on the projection-camera calibration from ProCamCalibration. This project also includes a tool to stream and render Kinect depth data to Unity.

Here is an example scene from our *RoomAlive* project to illustrate what is possible (this one uses 6 projectors and 6 Kinect cameras):



Development Status

This project is under development. The current release is in beta and all APIs are subject to change. The next major features might include compute shaders for handling all depth meshes as a single unified vertex buffer, radiometric compensation across dynamic scenes, support for Unity and Unreal Engines. We welcome contributions!

Citations

The RoomAlive Project was started in the summer of 2013 with a group of superstar interns. If you are looking for a reference to that original work, please cite:

```
1 @inproceedings{Jones:2014:RME:2642918.2647383,
2   author = {Jones, Brett and Sodhi, Rajinder and Murdock, Michael and
3     Mehra, Ravish and Benko, Hrvoje and Wilson, Andrew and Ofek, Eyal
4     and MacIntyre, Blair and Raghuvanshi, Nikunj and Shapira, Lior},
5   title = {RoomAlive: Magical Experiences Enabled by Scalable, Adaptive
6     Projector-camera Units},
7   booktitle = {Proceedings of the 27th Annual ACM Symposium on User
8     Interface Software and Technology},
9   series = {UIST '14},
10  year = {2014},
11  isbn = {978-1-4503-3069-5},
12  location = {Honolulu, Hawaii, USA},
13  pages = {637--644},
14  numpages = {8},
15  url = {http://doi.acm.org/10.1145/2642918.2647383},
16  doi = {10.1145/2642918.2647383},
17  acmid = {2647383},
18  publisher = {ACM},
19  address = {New York, NY, USA},
20  keywords = {projection mapping, projector-camera system, spatial
21    augmented reality},
22 }
```

Contribute

We welcome contributions to help advance projection mapping research frontier!

- File an issue first so we are aware about change you want to make and possibly guide you. Please include these log files when you report an issue.
- Use usual steps to make changes just like other GitHub projects.
- Clean compile your changes on Windows and test basic operations.

-
- When your pull request is created, you might get prompted to one-time sign Contributor License Agreement (CLA) unless changes are minor. It's very simple and takes less than a minute.
 - If your pull request gets a conflict, please resolve it.
 - Watch for any comments on your pull request.
 - Please try and limit your changes to small number of files. We need to review every line of your change and we can't reasonably do that if you make requests with huge number of changes.
 - Do not make just cosmetic changes. We will generally reject pull requests with only cosmetic changes.

License

This project is licensed under MIT license.