

3) Library installation

3.1) Installation

Download the IPDSLlib archive from the website and extract it within a folder (this folder will be defined as /home/courbon/IPDS/IPDSCodes/IPDSLlib/ in the sequel).

```
$ cd /home/courbon/IPDS/IPDSCodes/IPDSLlib/  
$ mkdir build  
$ cd build  
$ cmake ..
```

3.2) Summary

The IPDSLlib is the core library. Among other, it provides tools for:

- retrieving and synchronizing multiple sensor data (refer to Main methods, ConfigTable, Unit)
- data visualisation (ImageViewer_Qt, ImageLoader, LidarLoader, ImageCombination, QThreadExec)

Warning:

This piece of code is only built if Qt4 has been detected

- omnidirectional image unwarping (unwrapper)

Warning:

This piece of code is only built if OpenCV has been detected

- Data manipulation and usefull methods to reproject points from one sensor data to another (HomogeneousTransformation, HomogeneousVector, Camera_PinholeModel, Camera_UnifiedModel, ImagePoint, TransformationTool, LaserTool, LidarLoader)

Warning:

This piece of code is only built if Qt4 has been detected

To resume, most of the provided methods are available when Qt4 has been detected.

3.3) Creating the documentation

To build the HTML documentation using Doxygen, run , within build directory:

```
$make html-doc
```

3.4) Using libIPDS in your own project

Compiling and running the project

Using command line CMake

Make a directory called `build`, in which the compilation will be done. Do:

```
$ cd /PATH/TO/MY/GRAND/PROJECT
$ mkdir build
$ cd build
$ cmake ..
```

Using CMake gui (e.g. Windows)

Run CMake GUI, and fill these fields :

- **Where is the source code** : this is the folder containing the `CMakeLists.txt` file and the sources.
- **Where to build the binaries** : this is where the Visual Studio project files will be generated

Then, click **Configure**. You will be prompted for a generator/compiler. Then click the **Generate** button. If there is no errors, the project files will be generated into the **Where to build the binaries** folder.

Open the `sln` file, and build your project!