3D room mapping with a hand-held ToF camera
User Guide

Tobias Grundström
tobgr602@student.liu.se

Anna Hjelmberg
annhj876@student.liu.se

Mats Nilsson
matni403@student.liu.se

Fredrik Olsson
freol454@student.liu.se

2017-12-08

Contents

1 Installation

This section describes how to download and install the software and required dependencies. The software is developed and tested on Linux Mint 18.3 but should work on most modern Unix distributions though some configuration may be necessary.

1.1 Folder Structure

The intended folder structure to use our CMake-files is as follows:

```
~/TSBB11/
  fatman
  flann
  boost
  eigen
  vtk
  pcl
```

1.2 CMake

The program requires CMake[1] to build the software, if running on LiU systems use these commands.

```
module initadd prog/cmake/3.5.1
module add prog/cmake/3.5.1
```

Otherwise make sure the system has CMake version 3.5.1 or above.

1.3 FLANN

Fast Library for Aproximate Nearest Neighbour, FLANN[2], provides a library for NN search in high dimensional space. [1].

```
git clone https://github.com/mariusmuja/flann.git
cd flann && mkdir build && cd build
cmake .. -DCMAKE_BUILD_TYPE=release
make
```

[1] https://cmake.org/
[2] http://www.cs.ubc.ca/research/flann/

1.4 Boost

Boost library[3] is used by PCL for handling pointers.

```
wget https://dl.bintray.com/boostorg/release/1.65.1/source/boost_1_65_1.
    ↪ tar.gz
tar xvf boost_1_65_1.tar.gz
```

1.5 Eigen

Eigen is used by FGR and PCL to handle matrices. [2]

```
wget http://bitbucket.org/eigen/eigen/get/3.3.4.tar.bz2
tar xvf 3.3.4.tar.bz2
cd eigen-eigen-5a0156e40feb && mkdir build && cd build
cmake .. -DCMAKE_INSTALL_PREFIX=/home/matni403/TSBB11/eigen-eigen-5
    ↪ a0156e40feb/build -DINCLUDE_INSTALL_DIR=eigen3
make install
```

1.6 VTK

```
wget http://www.vtk.org/files/release/8.0/VTK-8.0.1.tar.gz
tar xvf VTK-8.0.1.tar.gz
cd VTK-8.0.1 && mkdir build && cd build
cmake ..
make
```

1.7 PCL

```
wget https://github.com/PointCloudLibrary/pcl/archive/pcl-1.8.1.tar.gz
tar xvf pcl-1.8.1.tar.gz
cd pcl-pcl-1.8.1 && mkdir build && cd build
ccmake ..
```

In the console, press [c] to start the configuration and [t] to see more options. When you get an error, press [e] to exit the warnings and go back to parameters. Change the following parameters, all at the same time or one library at a time, reconfiguring ([c],[e]) between each library/parameter. Ignore all warnings for developers.

Lastly press [g] to generate the makefiles after changing all the parameters.

[3] http://www.boost.org/

```
BUILD_visualization             ON
BUILD_surface                   ON

EIGEN_INCLUDE_DIR               ~/TSBB11/eigen-eigen-5a0156e40feb/

FLANN_INCLUDE_DIR               ~/TSBB11/flann/src/cpp
FLANN_LIBRARY                   ~/TSBB11/flann/build/lib/libflann.so
FLANN_LIBRARY_DEBUG             ~/TSBB11/flann/build/lib/libflann.so

VTK_DIR                         ~/TSBB11/VTK-8.0.1/build
```

After setting the directories run the following command:

```
make
```

1.8 FATMAN (Fredrik Anna Tobias Mats Are Nerds)

If the above mentioned folder structure is set up as intended and all dependencies are installed correctly, clone the Git-repository and build the program.

```
git clone https://gitlab.ida.liu.se/tsbb11-2017/fatman.git
cd build
cmake ..
make
```

1.9 Other packages

Some more packages may be required to be installed on the system for the program to work, for example the GNU C++ compiler and libgl.

2 Connecting the Camera

The camera connects to an RJ45 ethernet port which should connect to the same system which runs the software. This can either be done using a router and connecting the system to that router or connect the ethernet cable directly to the computer in which case a DHCP server[4] should run on the computer.

[4] http://www.dhcpserver.de/cms/

3 Running the Software

The software is run in command line and uses a few flags that have to be set when running in order to determine how the program should run, e.g. from pre-taken images or directly from the camera.

```
-n <2,...>       The Number of images to use
-f <cam,disk>    Images from camera or disk
-w <0,1>         Write new images to disk
```

For example to run the program by taking 10 images directly with the camera and then saving the images to file the correct line should be like this:

```
./main -n 10 -f cam -w 1
```

As soon as the program is run it will connect to the camera and capture the first image as soon as a connection is established, so be sure to direct the camera against the scene before running the program. The prompt will then ask you to press enter to continue before capturing each successive image.

References

[1] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *International Conference on Computer Vision Theory and Application VISSAPP'09)*, INSTICC Press, 2009, pp. 331–340.

[2] G. Guennebaud, B. Jacob, *et al.*, *Eigen v3*, http://eigen.tuxfamily.org, 2010.