

Installation und Nutzung der LTILib2 unter GNU/Linux für Robotvision

John Janus john.janus@haw-hamburg.de

11. April 2013

1 Installation der LTILib2

Der Quellcode findet sich unter <http://www.ie.itcr.ac.cr/palvarado/ltilib-2/homepage/index.html>. Dort lässt sich ein .tar.bz2 Archiv herunterladen. Dieses wird nun mittels `tar -xvf <archiv>` entpackt. Für openSUSE, debian und ubuntu finden sich unter ltilib-2/linux Skripte, welche die Dependencies installieren. Als root ausgeführt installieren diese die zum Bauen der Bibliothek benötigten Pakete. Falls für openSUSE 12.3 gebaut werden soll, muss man vorher eine Datei kopieren. Ab hier wird angenommen, dass man sich in dem Verzeichnis ltilib-2/linux befindet. Dies geht mittels `mkdir -p .distros/openSUSE__project/Dartmouth` zum erstellen des Verzeichnisses und `cp .distros/SUSE_Linux/Caledon/packages .distros/openSUSE__project/dartmouth` um die Datei packages in das richtige Verzeichnis zu kopieren. Nun kann mit `make -f Makefile.cvs` das configure script erstellt werden. Ab hier beginnt der normale Dreisprung:

```
1 ./configure
2 make
3 sudo make install
```

So wird die Bibliothek nach /usr/local installiert.

2 Änderung der includes und Methoden

Bisher konnte ich mit folgenden Änderungen die Lösung der ersten Aufgabe übersetzen:

- bei allen lti und gtk includes die "" in <> ändern
- <gtk/gtk.h> statt "gtk.h"
- <ltiViewer2D.h> statt "ltiViewer.h"
- <ltiIOImage.h> statt "ltiBMPFunctor.h"
- <ltiSplitImageToHSI.h> statt "ltiSplitImg.h"
- <ltiGuiServer.h> statt "ltiGtkServer.h"

- `<ltiChannel8.h>` statt `"ltiImage"`
- `gtkServer server` und `server.start()` entfernen
- `loadBMP` zu `ioImage` ändern
- `Viewer` zu `Viewer2D` ändern
- den Aufruf der `channel8::resize` Methode von `resize (rowSize,columnSize,0,false,true)` zu `resize (rowSize,columnSize,0,lti :: Init)` ändern

3 Kompilierung eines Projektes

Da der Compiler ziemlich viele Bibliotheken inkludieren muss, gibt es ein nützliches Tool, welches die benötigten Flags ausgibt: `lti-config`. Um nun ein Programm zu übersetzen nutzt man folgenden Befehl:

```
1  g++ -Wall -std=c++11 -o <Programmname> <Quelle 1> <Quelle 2>... 'lti-config --libs --cxxflags'
```

Bei dem Compileraufruf kommt es auf die Reihenfolge an! `lti-config` muss nach den Quelldateien aufgerufen werden.