

●ログインパスワード

開発キットのrootおよび初期ユーザ(10moons)のパスワードは以下となります。

パスワード：**123456789**

●環境準備

Eeye開発環境を構築する場合、ご利用の環境で以下の実施をお願いします。

① Linux環境の更新(ubuntuの場合)

```
$ sudo apt update
$ sudo apt upgrade
$ sudo apt install g++
```

② 必要ライブラリのインストール

```
$ sudo apt install libssl-dev
$ sudo apt install libudev-dev
$ sudo apt install libgtk-3-dev
$ sudo apt install libjpeg9-dev
```

●環境構築

環境構築にあたっては弊社よりご提供(ダウンロード含)の開発環境ファイルの最新版を事前に開発環境内に保存してください。

ファイル名：lib2-200422BB.tar.gz (2020/4/23 現在)

① 事前に保存した開発環境ファイルを展開します。

```
$ tar -zxvf lib2-200422BB.tar.gz
```

※ 展開後のフォルダ構成は以下となります。

```
lib2-200422BB
├── fclnetsvr   : MQTTブローカサーバ
├── fclnettest  : MQTTを経由したネットワーク環境での動作サンプルアプリ
├── fcltest     : Eeyeカメラ単体で動作するサンプルアプリ
├── libfcl      : Eeyeカメラ用ライブラリ群(非ネットワーク用)
└── libfclnet   : Eeyeカメラ用ライブラリ群(ネットワーク用)
```

●ビルド方法

展開したフォルダ(lib2-200422BB)にて以下の順にビルドを実施してください。

```
$ make -C libfcl
$ make -C fclnetsvr
$ make -C libfclnet
$ make -C fcltest
$ make -C fclnettest
```

● サンプルアプリ利用手順(ネットワーク未接続)

前提条件として、実行ファイル作成済

<非ネットワーク利用版>

実行ファイル名： fcltest2

<概要>

- ① カメラ側にて解析された顔情報データを取得し、モニタへ表示します。
- ② ターミナル上にアプリ起動時に引数にて設定した顔画像(jpegファイル)と比較した際の比較結果を表示します。(0.4以上で同一人物)

<起動方法>

- ① ビルド実施後fcltestフォルダへ移動します。
\$ cd fcltest
- ② fcltestディレクトリに“face.0.jpeg”の名前で画像ファイルを配置します。
- ③ 事前にEeyeカメラとCPUボードをUSBにて接続しておきます。
- ④ 環境設定実施
\$ source setup
- ⑤ アプリ起動
\$./fcltest2 -data face.0.jpeg

● サンプルアプリ利用手順(ネットワーク接続)

前提条件として、実行ファイル作成済

<ネットワーク利用(ベータ)版>

実行ファイル名： fclnettest2

<概要>

- ① MQTTのブローカサーバを介してデータのやり取りを実施します。
- ② カメラ側にて解析された顔情報データを取得し、ディスプレイへ表示します。
- ③ 起動時に設定した顔画像(jpegファイル)と一致した顔がカメラに映ると顔を囲む枠の色が緑に変化します。
なお、ネットワーク版のアプリケーションについてはベータ版となります。

<起動方法>

※Linuxが動作するLinuxボードが2台必要

- ① 2台をそれぞれLANケーブルで接続します。
カメラ接続側はssh等でログインを実施します。
- ② ディスプレイ側Linuxボード
 - ②-1 割り付けられたアドレスを確認します。

```
$ ifconfig eth0
```


eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.100.140 netmask 255.255.255.0 broadcast 192.168.100.255
inet6 **2405:6585:c400:200:3e76:9162:890e:7dc4** prefixlen 64 scopeid 0x0<global>
inet6 fe80::568e:a271:ee8d:2e2f prefixlen 64 scopeid 0x20<link>
RX packets 3685917 bytes 1172177452 (1.0 GiB)
RX errors 1 dropped 8 overruns 0 frame 1
TX packets 691383 bytes 224759340 (214.3 MiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
 - ③ 事前にEeyeカメラとCPUボードをUSBにて接続しておきます。
 - ④ カメラ接続側Linuxボード
 - ④-1 fclnetsvrへ移動します。

```
$ cd fclnetsvr
```
 - ④-2 環境設定実施

```
$ source setup
```
 - ④-3 アプリ起動(IPアドレス部分はディスプレイ側Linuxボード側のIPアドレスを入力。)

```
$ ./fclnetsvr -id zz -url tcp://[2405:6585:c400:200:3e76:9162:890e:7dc4]:1883
```

⑤ ディスプレイ側Linuxボード(MQTTサーバ構築)

```
$ sudo apt install mosquitto  
$ cd fclnettest  
$ source setup
```

⑥ アプリ起動

```
$ ./fclnettest2 -id zz -url tcp://[::1]:1883 -data face.jpeg
```

起動時にjpegファイルを渡すと、そのjpegに一致した顔が映ると顔を囲む枠の色が緑になります。
(jpegファイルを渡さなくても起動可能)