

3D-Scan

0.1

Generated by Doxygen 1.7.1

Sun Aug 22 2010 12:32:36

Contents

1	Data Structure Index	1
1.1	Data Structures	1
2	File Index	3
2.1	File List	3
3	Data Structure Documentation	5
3.1	offset Struct Reference	5
4	File Documentation	7
4.1	server/3dscan_s.c File Reference	7
4.1.1	Detailed Description	8
4.1.2	Function Documentation	8
4.1.2.1	basic_sensor_setup	8
4.1.2.2	display_vga	8
4.1.2.3	main	9
4.1.2.4	save_imges_2_nfs	9
4.2	server/3dscan_s.h File Reference	9
4.2.1	Detailed Description	10
4.2.2	Define Documentation	10
4.2.2.1	CAL_SEARCHLEN	10
4.2.2.2	CAL_START_X	10
4.2.2.3	CAL_START_Y	10
4.2.2.4	DISPLAY_USAGE_AND_EXIT	10
4.2.3	Function Documentation	11
4.2.3.1	basic_sensor_setup	11
4.2.3.2	display_vga	11
4.2.3.3	save_imges_2_nfs	11

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

offset	5
------------------	---

Chapter 2

File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

server/ 3dscan_s.c (Server for MQ II serving for 3D- Scan Thesis)	7
server/ 3dscan_s.h (Server for MQ II serving for 3D- Scan Thesis)	9

Chapter 3

Data Structure Documentation

3.1 offset Struct Reference

Data Fields

- `int x_offs`
- `int y_offs`

The documentation for this struct was generated from the following file:

- `algorithm/algorithm.cpp`

Chapter 4

File Documentation

4.1 server/3dscan_s.c File Reference

Server for MQ II serving for 3D- Scan Thesis.

```
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <unistd.h>
#include <pthread.h>
#include <errno.h>
#include "/home/christoph/src/MiniQuadrix_II/software/include/mq2_
api.h"
#include "/home/christoph/src/MiniQuadrix_II/software/include/debug.h"
#include "../include/3ds.h"
#include "3dscan_s.h"
#include "../include/bmp.h"
#include "../include/font.h"
```

Functions

- int [main](#) (int argc, char **argv)
- int **determine_calibration** (char *path, int hei, int wid, short lower_tresh, short upper_tresh)
- int [save_images_2_nfs](#) (char *path, int hei, int wid)
Acquires an image from every can; saves them to a given path.
- void [display_vga](#) (void *_name)
Displays a cameraimage on the screen.
- int [basic_sensor_setup](#) (int ui_snr, int hei, int wid, int exp, int gain)
Does a basic setup for a sensor.

Variables

- unsigned int **ui_sensor** = 0
- unsigned short * **framebuffer1**
- unsigned short * **framebuffer2**
- pthread_t **th_display_vga**

4.1.1 Detailed Description

Server for MQ II serving for 3D- Scan Thesis.

Author

Christoph Gnip

Date

July 20th 2010

Version

1.0 This program is part of my Bachelor-Thesis. It is a Server for the MiniQuadrix II with the addition of a MiniCluster-board, which was designed during work. The idea is to generate 3D scene data exploiting only the casted shadows. The generation of the shadows is one of the tasks of the MiniCluster-Board. Therefore it holds 4 LED-arrays located on the board's edges. It is also possible to setup useful parameters, which have an influence on the acquired image.

4.1.2 Function Documentation

4.1.2.1 int basic_sensor_setup (int *ui_snr*, int *hei*, int *wid*, int *exp*, int *gain*)

Does a basic setup for a sensor.

Parameters

ui_snr Configure sensor number; set to ALL_SENSORS if you want to configure all

hei Image height

wid Image width

exp Exposure

gain Analog Gain

4.1.2.2 void display_vga (void * *_name*)

Displays a camera image on the screen.

Parameters

**_name* Name of the Thread

Use as thread to display the camera data

4.1.2.3 int main (int argc, char ** argv)

Parameters

argc Argumentcount

****argv** Given this is a commandline tool, it uses some parameters. Known are 3 main parameters

- 1 Live modus
- 2 Acquire 4 images (basicly for calibration)
- 3 Acquire sequence; 4x4 Pictures save under a path

4.1.2.4 int save_imges_2_nfs (char * path, int hei, int wid)

Acquires an image from every can; saves them to a given path.

Parameters

***path** Path where images are saved to

hei Heights of the image

wid Width of the image

4.2 server/3dscan_s.h File Reference

Server for MQ II serving for 3D- Scan Thesis.

Defines

- #define **LIVE** 1
- #define **TAKE_PICTURE** 2
- #define **RECORD_SEQUENCE** 3
- #define **CALIBRATION_RUN** 4
- #define **VGA_IMG_HEIGHT** 480
- #define **VGA_IMG_WIDTH** 640
- #define **MAX_IMG_HEIGHT** 480
- #define **MAX_IMG_WIDTH** 752
- #define **CAL_START_Y** 220
- #define **CAL_START_X** 220
- #define **CAL_SEARCHLEN** 40
- #define **DISPLAY_USAGE_AND_EXIT**

Functions

- int **basic_sensor_setup** (int ui_snr, int hei, int wid, int exp, int gain)
Does a basic setup for a sensor.
- int **save_imges_2_nfs** (char *path, int hei, int wid)
Acquires an image from every can; saves them to a given path.

- int **determine_calibration** (char *path, int hei, int wid, short lower_tresh, short upper_tresh)
- void **display_vga** (void *_name)

Displays a cameraimage on the screen.

Variables

- PB_SENSOR_CONTROL * **t_pb_sensor_control**
- PB_VGA_FIFO * **t_pb_vga_fifo** = NULL
- PB_DATAPLEXER * **t_pb_dataplexer** [4]
- PB_THRESHOLD * **t_pb_threshold** [4]
- PB_LED * **t_pb_leda** = NULL
- MQ2_MODULE_DESC * **t_mq2_module_desc**

4.2.1 Detailed Description

Server for MQ II serving for 3D- Scan Thesis.

Author

Christoph Gnip

Date _____

July 20th 2010

Version

1.0

4.2.2 Define Documentation

4.2.2.1 #define CAL_SEARCHLEN 40

Length to serch for a bright-dark change in the image [pxs]

4.2.2.2 #define CAL START X 220

X-Startoffset for the bright-dark change search [pxs]

4.2.2.3 #define CAL_START_Y 220

Y-Startoffset for the bright-dark change search [pxs]

4.2.2.4 #define DISPLAY_USAGE_AND_EXIT

Value:

```
printf("3dscan_s [Mode] [...] \n \t\t\t Mode can be:\n\t\t\t\t\t 1 ->\t Display im  
age on extern monitor\n \t\t\t\t\t 2 ->\t Take a picture of all cameras\n \t\t\t\t\t ->\t Record Image sequence\n \ \n"); \nexit(0)
```

4.2.3 Function Documentation

4.2.3.1 `int basic_sensor_setup (int ui_snr, int hei, int wid, int exp, int gain)`

Does a basic setup for a sensor.

Parameters

ui_snr Configure sensor number; set to ALL_SENSORS if you want to configure all

hei Image height

wid Image width

exp Exposure

gain Analog Gain

4.2.3.2 `void display_vga (void * _name)`

Displays a camera image on the screen.

Parameters

**_name* Name of the Thread

Use as thread to display the camdata

4.2.3.3 `int save_imges_2_nfs (char * path, int hei, int wid)`

Acquires an image from every can; saves them to a given path.

Parameters

**path* Path where images are saved to

hei Heights of the image

wid Width of the image

Index

- 3dscan_s.c
 - basic_sensor_setup, [8](#)
 - display_vga, [8](#)
 - main, [8](#)
 - save_imges_2_nfs, [9](#)
- 3dscan_s.h
 - basic_sensor_setup, [11](#)
 - CAL_SEARCHLEN, [10](#)
 - CAL_START_X, [10](#)
 - CAL_START_Y, [10](#)
 - DISPLAY_USAGE_AND_EXIT, [10](#)
 - display_vga, [11](#)
 - save_imges_2_nfs, [11](#)
- basic_sensor_setup
 - 3dscan_s.c, [8](#)
 - 3dscan_s.h, [11](#)
- CAL_SEARCHLEN
 - 3dscan_s.h, [10](#)
- CAL_START_X
 - 3dscan_s.h, [10](#)
- CAL_START_Y
 - 3dscan_s.h, [10](#)
- DISPLAY_USAGE_AND_EXIT
 - 3dscan_s.h, [10](#)
- display_vga
 - 3dscan_s.c, [8](#)
 - 3dscan_s.h, [11](#)
- main
 - 3dscan_s.c, [8](#)
- offset, [5](#)
- save_imges_2_nfs
 - 3dscan_s.c, [9](#)
 - 3dscan_s.h, [11](#)
- server/3dscan_s.c, [7](#)
- server/3dscan_s.h, [9](#)