- Main Page
- Data Structures
- Files

- File List
- Globals

param.h File Reference

ARToolkit global structure (parameters) subroutines. More...

```
#include <AR/config.h>
```

Data Structures

struct	**ARParam** *camera intrinsic parameters. More...*
struct	**ARSParam**

Functions

int	**arParamGet** (double global[][3], double screen[][2], int data_num, double mat[3][4]) *XXXBK.*
int	**arParamDecomp** (**ARParam** *source, **ARParam** *icpara, double trans[3][4]) *XXXBK.*
int	**arParamDecompMat** (double source[3][4], double cpara[3][4], double trans[3][4]) *XXXBK.*
int	**arParamIdeal2Observ** (const double dist_factor[4], const double ix, const double iy, double *ox, double *oy) *Convert ideal screen coordinates of a vertex to observed ones.*
int	**arParamObserv2Ideal** (const double dist_factor[4], const double ox, const double oy, double *ix, double *iy) *Convert observed screen coordinates of a vertex to ideal ones.*
int	**arParamChangeSize** (**ARParam** *source, int xsize, int ysize, **ARParam** *newparam) *change the camera size parameters.*
int	**arParamSave** (char *filename, int num, **ARParam** *param,...) *save a camera intrinsic parameters.*
int	**arParamLoad** (const char *filename, int num, **ARParam** *param,...) *load the camera intrinsic parameters.*
int	**arParamDisp** (**ARParam** *param) *display parameters.*
int	**arsParamChangeSize** (ARSParam *source, int xsize, int ysize, ARSParam *newparam)
int	**arsParamSave** (char *filename, ARSParam *sparam)
int	**arsParamLoad** (char *filename, ARSParam *sparam)
int	**arsParamDisp** (ARSParam *sparam)
int	**arsParamGetMat** (double matL[3][4], double matR[3][4], double cparaL[3][4], double cparaR[3][4], double matL2R[3][4])

Detailed Description

ARToolkit global structure (parameters) subroutines.

This file contains principal routines for loading, saving, and modify camera parameters for of ARToolkit library. Different structures are used for modify in run-time this parameters in the library. A file structure is use for input/output.

Remarks:

History :

Author:

Takeshi Mita tmita@inolab.sys.es.osaka-u.ac.jp

Shinsaku Hiura shinsaku@sys.es.osaka-u.ac.jp

Hirokazu Kato kato@sys.im.hiroshima-cu.ac.jp

Version:

4.1

Date:

01/12/07

Function Documentation

```
int arParamChangeSize( ARParam * source,
                       int       xsize,
                       int       ysize,
                                  ARParam * newparam
                                )
```

change the camera size parameters.

Change the size variable in camera intrinsic parameters.

Parameters:

source	name of the source parameters structure
xsize	new length size
ysize	new height size
newparam	name of the destination parameters structure.

Returns:
0

```
int arParamDecomp( ARParam * source,
                   ARParam * icpara,
                   double    trans[3][4]
                 )
```

XXXBK.

XXXBK

Parameters:

source	XXXBK
icpara	XXXBK
trans	XXXBK

Returns:
XXXBK

```
int arParamDecompMat( double source[3][4],
                      double cpara[3][4],
                      double trans[3][4]
                    )
```

XXXBK.

XXXBK

Parameters:

source	input camera matrix
cpara	camera parameter to be set
trans	XXXBK

Returns:
XXXBK

```
int arParamDisp( ARParam * param )
```

display parameters.

Display the structure of the camera instrinsic parameters argument.

Parameters:

param	structure to display

Returns:
0

```
int arParamGet( double global[][3],
                double screen[][2],
                int    data_num,
                double mat[3][4]
              )
```

XXXBK.

XXXBK

Parameters:

global	XXXBK

screen	XXXBK
data_num	XXXBK
mat	XXXBK

Returns:
XXXBK

```
int int arParamIdeal2Observ( const double dist_factor[4],
                             const double ix,
                             const double iy,
                             double *     ox,
                             double *     oy
                           )
```

Convert ideal screen coordinates of a vertex to observed ones.

Ideal coordinates mean that the distortion of the camera is compensated (so a straight line looks straight). In observed coordinates the camera-distortion is not compensated and thus a straight line is not shown really straight.

Parameters:

dist_factor	distorsion factors of used camera
ix	x in ideal screen coordinates
iy	y in ideal screen coordinates
ox	resulted x in observed screen coordinates
oy	resulted y in observed screen coordinates

Returns:
0 if success, -1 otherwise

```
int arParamLoad( const char * filename,
                 int          num,
                 ARParam *    param,
                              ...
               )
```

load the camera intrinsic parameters.

Load camera intrinsic parameters in the ARToolkit Library from a file (itself, an output of the calibration step).

Parameters:

filename	name of the parameters file.
num	number of variable arguments
param	result of the loaded parameters

Returns:
0 if success, -1 if Error (file not found, file structure problem)

```
int arParamObserv2Ideal( const double dist_factor[4],
                         const double ox,
                         const double oy,
                         double *     ix,
                         double *     iy
                       )
```

Convert observed screen coordinates of a vertex to ideal ones.

Ideal coordinates mean that the distortion of the camera is compensated (so a straight line looks straight). In observed coordinates the camera-distortion is not compensated and thus a straight line is not shown really straight.

Parameters:

dist_factor	distorsion factors of used camera
ox	x in observed screen coordinates
oy	y in observed screen coordinates
ix	resulted x in ideal screen coordinates
iy	resulted y in ideal screen coordinates

Returns:
0 if success, -1 otherwise

```
int arParamSave( char *    filename,
                 int       num,
                 ARParam * param,
                           ...
```

)

save a camera intrinsic parameters.

Save manipulated camera intrinsic parameters in a file.

Parameters:

filename	name of the parameters file.
num	number of variable arguments
param	parameters to save

Returns:

0 if success, -1 if Error (file not found, file structure problem)

Generated with Doxygen