

C - Particle Filter

0.1.0

Generated by Doxygen 1.8.6

Sun Feb 7 2016 19:54:40

Contents

1	Class Index	1
1.1	Class List	1
2	File Index	3
2.1	File List	3
3	Class Documentation	5
3.1	measurement Struct Reference	5
3.1.1	Member Data Documentation	5
3.1.1.1	x	5
3.1.1.2	y	5
3.2	particle Struct Reference	5
3.2.1	Member Data Documentation	5
3.2.1.1	w	5
3.2.1.2	x	5
3.2.1.3	y	6
4	File Documentation	7
4.1	src/gauss_sample.c File Reference	7
4.1.1	Function Documentation	7
4.1.1.1	box_muller	7
4.1.1.2	main	7
4.2	src/particle_filter.c File Reference	7
4.2.1	Macro Definition Documentation	8
4.2.1.1	PI	8
4.2.2	Function Documentation	8
4.2.2.1	create_fake_measurements	8
4.2.2.2	init_particles	8
4.2.2.3	main	8
4.2.2.4	max	8
4.2.2.5	normpdf	8
4.2.2.6	particle_filter	8

4.2.2.7	resampling_wheel	8
4.2.2.8	weighted_average	8
4.2.2.9	weighted_sample	8
4.2.2.10	zs_to_csv	8
4.3	src/particle_filter.h File Reference	8
4.3.1	Macro Definition Documentation	9
4.3.1.1	M	9
4.3.1.2	max_x	9
4.3.1.3	max_y	9
4.3.1.4	N	9
4.3.2	Function Documentation	9
4.3.2.1	init_particles	9
4.3.2.2	normpdf	9
4.3.2.3	particle_filter	9
4.4	src/random_number_gen.c File Reference	9
4.4.1	Function Documentation	10
4.4.1.1	init	10
4.4.1.2	randn	10
4.4.1.3	randu	10
4.5	src/random_number_gen.h File Reference	10
4.5.1	Function Documentation	10
4.5.1.1	randn	10
4.5.1.2	randu	10
Index		11

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

measurement	5
particle	5

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

src/ gauss_sample.c	7
src/ particle_filter.c	7
src/ particle_filter.h	8
src/ random_number_gen.c	9
src/ random_number_gen.h	10

Chapter 3

Class Documentation

3.1 measurement Struct Reference

```
#include <particle_filter.h>
```

Public Attributes

- double [x](#)
- double [y](#)

3.1.1 Member Data Documentation

3.1.1.1 double measurement::x

3.1.1.2 double measurement::y

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

- [src/particle_filter.h](#)

3.2 particle Struct Reference

```
#include <particle_filter.h>
```

Public Attributes

- double [x](#)
- double [y](#)
- double [w](#)

3.2.1 Member Data Documentation

3.2.1.1 double particle::w

3.2.1.2 double particle::x

3.2.1.3 double particle::y

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

- [src/particle_filter.h](#)

Chapter 4

File Documentation

4.1 src/gauss_sample.c File Reference

```
#include <stdio.h>
#include <math.h>
```

Functions

- float [box_muller](#) (float *m*, float *s*)
- int [main](#) (int *argc*, char **argv*[])

4.1.1 Function Documentation

4.1.1.1 float [box_muller](#) (float *m*, float *s*)

4.1.1.2 int [main](#) (int *argc*, char * *argv*[])

4.2 src/particle_filter.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <particle_filter.h>
#include <random_number_gen.h>
#include <math.h>
#include <time.h>
```

Macros

- #define [PI](#) 3.14159265358979323846

Functions

- void [init_particles](#) (struct [particle](#) *particles*[*N*])
- double [normpdf](#) (double *x*, double *mu*, double *sigma*)

Calculates the value of x of a normal distribution with mean μ and standard deviation σ .

- void [weighted_sample](#) (struct [particle](#) ps[], struct [particle](#) res[], double weights[], int samples)
- double [max](#) (double arr[], int size)
- void [resampling_wheel](#) (struct [particle](#) ps[], struct [particle](#) res[], double weights[], int samples)
- void [particle_filter](#) (struct [particle](#) xs[N], struct [measurement](#) *z)
- void [create_fake_measurements](#) (struct [measurement](#) zs[M])
- void [zs_to_csv](#) (struct [measurement](#) zs[M], char filename[])
- struct [particle](#) [weighted_average](#) (struct [particle](#) ps[], int size)
- int [main](#) (int argc, char *argv[])

4.2.1 Macro Definition Documentation

4.2.1.1 `#define PI 3.14159265358979323846`

4.2.2 Function Documentation

4.2.2.1 void [create_fake_measurements](#) (struct [measurement](#) zs[M])

4.2.2.2 void [init_particles](#) (struct [particle](#) particles[N])

4.2.2.3 int [main](#) (int argc, char * argv[])

4.2.2.4 double [max](#) (double arr[], int size)

4.2.2.5 double [normpdf](#) (double x, double mu, double sigma)

Calculates the value of x of a normal distribution with mean mu and standard deviation sigma.

Parameters

x	
mu	Mean of the normal distribution
sigma	Standard deviation of the normal distribution

Returns

Value of the normal distribution at position x

4.2.2.6 void [particle_filter](#) (struct [particle](#) xs[N], struct [measurement](#) * z)

4.2.2.7 void [resampling_wheel](#) (struct [particle](#) ps[], struct [particle](#) res[], double weights[], int samples)

4.2.2.8 struct [particle](#) [weighted_average](#) (struct [particle](#) ps[], int size)

4.2.2.9 void [weighted_sample](#) (struct [particle](#) ps[], struct [particle](#) res[], double weights[], int samples)

4.2.2.10 void [zs_to_csv](#) (struct [measurement](#) zs[M], char filename[])

4.3 src/particle_filter.h File Reference

Classes

- struct [particle](#)
- struct [measurement](#)

Macros

- #define `N` 100 /* Total number of particles */
- #define `M` 200 /* Number of fake measurements */
- #define `max_x` 200
- #define `max_y` 200

Functions

- double `normpdf` (double `x`, double `mu`, double `sigma`)
Calculates the value of x of a normal distribution with mean mu and standard deviation sigma.
- void `init_particles` (struct `particle` `particles[N]`)
- void `particle_filter` (struct `particle` `particles[N]`, struct `measurement` *`m`)

4.3.1 Macro Definition Documentation

4.3.1.1 #define `M` 200 /* Number of fake measurements */

4.3.1.2 #define `max_x` 200

4.3.1.3 #define `max_y` 200

4.3.1.4 #define `N` 100 /* Total number of particles */

4.3.2 Function Documentation

4.3.2.1 void `init_particles` (struct `particle` *particles[N]*)

4.3.2.2 double `normpdf` (double *x*, double *mu*, double *sigma*)

Calculates the value of x of a normal distribution with mean mu and standard deviation sigma.

Parameters

<i>x</i>	
<i>mu</i>	Mean of the normal distribution
<i>sigma</i>	Standard deviation of the normal distribution

Returns

Value of the normal distribution at position x

4.3.2.3 void `particle_filter` (struct `particle` *particles[N]*, struct `measurement` * *m*)

4.4 src/random_number_gen.c File Reference

```
#include <math.h>
#include <stdlib.h>
#include <time.h>
```

Functions

- void `init` ()
- double `randn` (double *mu*, double *sigma*)
- double `randu` (double *a*, double *b*)

4.4.1 Function Documentation

4.4.1.1 void `init` ()

4.4.1.2 double `randn` (double *mu*, double *sigma*)

4.4.1.3 double `randu` (double *a*, double *b*)

4.5 `src/random_number_gen.h` File Reference

Functions

- double `randn` (double *mu*, double *sigma*)
- double `randu` (double *a*, double *b*)

4.5.1 Function Documentation

4.5.1.1 double `randn` (double *mu*, double *sigma*)

4.5.1.2 double `randu` (double *a*, double *b*)

Index

- box_muller
 - gauss_sample.c, 7
- create_fake_measurements
 - particle_filter.c, 8
- gauss_sample.c
 - box_muller, 7
 - main, 7
- init
 - random_number_gen.c, 10
- init_particles
 - particle_filter.c, 8
 - particle_filter.h, 9
- M
 - particle_filter.h, 9
- main
 - gauss_sample.c, 7
 - particle_filter.c, 8
- max
 - particle_filter.c, 8
- max_x
 - particle_filter.h, 9
- max_y
 - particle_filter.h, 9
- measurement, 5
 - x, 5
 - y, 5
- N
 - particle_filter.h, 9
- normpdf
 - particle_filter.c, 8
 - particle_filter.h, 9
- PI
 - particle_filter.c, 8
- particle, 5
 - w, 5
 - x, 5
 - y, 5
- particle_filter
 - particle_filter.c, 8
 - particle_filter.h, 9
- particle_filter.c
 - create_fake_measurements, 8
 - init_particles, 8
 - main, 8
 - max, 8
 - normpdf, 8
 - PI, 8
 - particle_filter, 8
 - resampling_wheel, 8
 - weighted_average, 8
 - weighted_sample, 8
 - zs_to_csv, 8
- particle_filter.h
 - init_particles, 9
 - M, 9
 - max_x, 9
 - max_y, 9
 - N, 9
 - normpdf, 9
 - particle_filter, 9
- randn
 - random_number_gen.c, 10
 - random_number_gen.h, 10
- random_number_gen.c
 - init, 10
 - randn, 10
 - randu, 10
- random_number_gen.h
 - randn, 10
 - randu, 10
- randu
 - random_number_gen.c, 10
 - random_number_gen.h, 10
- resampling_wheel
 - particle_filter.c, 8
- src/gauss_sample.c, 7
- src/particle_filter.c, 7
- src/particle_filter.h, 8
- src/random_number_gen.c, 9
- src/random_number_gen.h, 10
- w
 - particle, 5
- weighted_average
 - particle_filter.c, 8
- weighted_sample
 - particle_filter.c, 8
- x
 - measurement, 5
 - particle, 5
- y
 - measurement, 5

particle, [5](#)

zs_to_csv

particle_filter.c, [8](#)