

Tennis Assistant - Computer Vision

Generated by Doxygen 1.8.9.1

Fri Mar 27 2015 22:20:43

Contents

1	Namespace Index	1
1.1	Packages	1
2	Hierarchical Index	3
2.1	Class Hierarchy	3
3	Class Index	5
3.1	Class List	5
4	File Index	7
4.1	File List	7
5	Namespace Documentation	9
5.1	basket Namespace Reference	9
5.1.1	Function Documentation	9
5.1.1.1	is_basket_middle	9
5.1.1.2	run	9
5.1.1.3	run_middle	10
5.1.2	Variable Documentation	10
5.1.2.1	base_filename	10
5.1.2.2	image_half_size	10
5.1.2.3	particle_filter	10
5.1.2.4	save_count	10
5.2	basket_runner Namespace Reference	10
5.3	basket_test Namespace Reference	10
5.3.1	Function Documentation	10
5.3.1.1	basketMissing	10
5.3.1.2	basketPresent	10
5.3.1.3	unitTest	10
5.4	experiment Namespace Reference	10
5.4.1	Detailed Description	10
5.4.2	Function Documentation	11
5.4.2.1	binary_mask	11

5.4.2.2	blobs_by_mask	11
5.4.2.3	dilation_and_blur	11
5.4.2.4	experiment	11
5.4.2.5	hard_threshold	11
5.5	image_support Namespace Reference	11
5.5.1	Function Documentation	11
5.5.1.1	external_init_particle_filter	11
5.5.1.2	get_best_blob	11
5.5.1.3	get_hue_blobs	12
5.5.1.4	image_hue_filter	13
5.5.1.5	is_blob_in_middle_helper	13
5.6	particlefilter Namespace Reference	13
5.7	prquadtree Namespace Reference	13
5.7.1	Detailed Description	14
5.8	prquadtree_test Namespace Reference	14
5.9	prquadtree_test_example Namespace Reference	14
5.9.1	Detailed Description	14
5.9.2	Variable Documentation	14
5.9.2.1	b	14
5.9.2.2	b2	14
5.9.2.3	c	14
5.9.2.4	nearby	14
5.9.2.5	pt	14
5.9.2.6	qt	14
5.10	tennis_ball Namespace Reference	14
5.10.1	Detailed Description	15
5.10.2	Function Documentation	15
5.10.2.1	is_ball_middle	15
5.10.2.2	run	15
5.10.3	Variable Documentation	15
5.10.3.1	particle_filter	15
5.11	tennis_ball_runner Namespace Reference	15
5.12	tennis_ball_test Namespace Reference	15
5.12.1	Function Documentation	16
5.12.1.1	ballMissing	16
5.12.1.2	ballPresent	16
5.12.1.3	unitTest	16
6	Class Documentation	17
6.1	prquadtree.Box Class Reference	17

6.1.1	Detailed Description	17
6.1.2	Constructor & Destructor Documentation	17
6.1.2.1	__init__	17
6.1.3	Member Function Documentation	18
6.1.3.1	contains_point	18
6.1.3.2	intersect	19
6.1.4	Member Data Documentation	19
6.1.4.1	center	19
6.1.4.2	half_size	19
6.2	prquadtree.Particle Class Reference	19
6.2.1	Detailed Description	20
6.2.2	Constructor & Destructor Documentation	20
6.2.2.1	__init__	20
6.2.3	Member Data Documentation	20
6.2.3.1	score	20
6.2.3.2	x	20
6.2.3.3	y	20
6.3	particlefilter.ParticleFilter Class Reference	20
6.3.1	Constructor & Destructor Documentation	20
6.3.1.1	__init__	20
6.3.2	Member Function Documentation	21
6.3.2.1	clear_scores	21
6.3.2.2	iterate	21
6.3.2.3	score	21
6.3.3	Member Data Documentation	21
6.3.3.1	image_box	21
6.3.3.2	iterations	21
6.3.3.3	iterations_before_clearing	21
6.3.3.4	pr_tree	21
6.4	prquadtree.Point Class Reference	21
6.4.1	Detailed Description	22
6.4.2	Constructor & Destructor Documentation	22
6.4.2.1	__init__	22
6.4.3	Member Function Documentation	22
6.4.3.1	__repr__	22
6.4.3.2	__str__	22
6.4.4	Member Data Documentation	23
6.4.4.1	x	23
6.4.4.2	y	23
6.5	prquadtree.PRQuadTree Class Reference	23

6.5.1	Detailed Description	23
6.5.2	Constructor & Destructor Documentation	24
6.5.2.1	__init__	24
6.5.3	Member Function Documentation	24
6.5.3.1	__str__	24
6.5.3.2	insert	24
6.5.3.3	print_all_points	24
6.5.3.4	query_k_nearest	25
6.5.3.5	query_range	26
6.5.3.6	size	26
6.5.4	Member Data Documentation	26
6.5.4.1	box	26
6.5.4.2	ne	26
6.5.4.3	nw	26
6.5.4.4	points	26
6.5.4.5	QT_NODE_CAPACITY	27
6.5.4.6	se	27
6.5.4.7	sw	27
6.6	prquadtree_test.TestBox Class Reference	27
6.6.1	Member Function Documentation	27
6.6.1.1	test_box_contains	27
6.6.1.2	test_box_insert	27
6.7	prquadtree_test.TestParticle Class Reference	27
6.7.1	Member Function Documentation	28
6.7.1.1	test_particle_insert	28
6.8	prquadtree_test.TestPoint Class Reference	28
6.8.1	Member Function Documentation	28
6.8.1.1	test_point_insert	28
6.9	prquadtree_test.TestPrQuadTree Class Reference	28
6.9.1	Member Function Documentation	28
6.9.1.1	test_insert	28
6.9.1.2	test_nearby	29
7	File Documentation	31
7.1	basket.py File Reference	31
7.2	basket_runner.py File Reference	31
7.3	basket_test.py File Reference	31
7.4	experiment.py File Reference	32
7.5	image_support.py File Reference	32
7.6	particlefilter.py File Reference	32

7.7	prquadtree.py File Reference	33
7.8	prquadtree_test.py File Reference	33
7.9	prquadtree_test_example.py File Reference	33
7.10	tennis_ball.py File Reference	33
7.11	tennis_ball_runner.py File Reference	34
7.12	tennis_ball_test.py File Reference	34
7.13	visual.h File Reference	34
7.13.1	Function Documentation	34
7.13.1.1	get_objects	34
7.13.1.2	set_objects	34
7.13.1.3	start_visual	34
7.13.1.4	stop_visual	34
Index		35

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

basket	9
basket_runner	10
basket_test	10
experiment	10
image_support	11
particlefilter	13
prquadtree	13
prquadtree_test	14
prquadtree_test_example	14
tennis_ball	14
tennis_ball_runner	15
tennis_ball_test	15

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

prquadtree.Box	17
particlefilter.ParticleFilter	20
prquadtree.Point	21
prquadtree.Particle	19
prquadtree.PRQuadTree	23
TestCase	
prquadtree_test.TestBox	27
prquadtree_test.TestParticle	27
prquadtree_test.TestPoint	28
prquadtree_test.TestPrQuadTree	28

Chapter 3

Class Index

3.1 Class List

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

prquadtree.Box	Class defining a square on the coordinate system via a center point and half of square width	17
prquadtree.Particle	Represents particle point	19
particlefilter.ParticleFilter	20
prquadtree.Point	Represents an (x,y) coordinate point on a grid	21
prquadtree.PRQuadTree	Class representing a Point Range Quadtree	23
prquadtree_test.TestBox	27
prquadtree_test.TestParticle	27
prquadtree_test.TestPoint	28
prquadtree_test.TestPrQuadTree	28

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

basket.py	31
basket_runner.py	31
basket_test.py	31
experiment.py	32
image_support.py	32
particlefilter.py	32
prquadtree.py	33
prquadtree_test.py	33
prquadtree_test_example.py	33
tennis_ball.py	33
tennis_ball_runner.py	34
tennis_ball_test.py	34
visual.h	34

Chapter 5

Namespace Documentation

5.1 basket Namespace Reference

Functions

- def `is_basket_middle` (`img`)
Single entry function returning True/False if basket is in the middle of the screen.
- def `run_middle` ()
Runs continuously and prints if the best detected blob is in the middle.
- def `run`
Runs continuously outlines best matched blob if it is in the middle.

Variables

- `particle_filter` = None
- int `image_half_size` = -1
- int `save_count` = 1
- tuple `base_filename` = datetime.now()

5.1.1 Function Documentation

5.1.1.1 `def basket.is_basket_middle (img)`

Single entry function returning True/False if basket is in the middle of the screen.

Parameters

<i>img</i>	SimpleCV.Image The image to test
------------	----------------------------------

5.1.1.2 `def basket.run (bestBlobCallback = False)`

Runs continuously outlines best matched blob if it is in the middle.

Parameters

<i>bestBlob↔ Callback</i>	function Callback called passing the best blob found
-------------------------------	--

5.1.1.3 def basket.run_middle ()

Runs continuously and prints if the best detected blob is in the middle.

5.1.2 Variable Documentation

5.1.2.1 tuple basket.base_filename = datetime.now()

5.1.2.2 int basket.image_half_size = -1

5.1.2.3 basket.particle_filter = None

5.1.2.4 int basket.save_count = 1

5.2 basket_runner Namespace Reference

5.3 basket_test Namespace Reference

Functions

- def [unitTest](#) (actual, expected, name)
- def [basketPresent](#) ()
- def [basketMissing](#) ()

5.3.1 Function Documentation

5.3.1.1 def basket_test.basketMissing ()

5.3.1.2 def basket_test.basketPresent ()

5.3.1.3 def basket_test.unitTest (*actual, expected, name*)

5.4 experiment Namespace Reference

Functions

- def [experiment](#)
- def [hard_threshold](#) (img)
- def [binary_mask](#) (img)
- def [dilation_and_blur](#) (img)
- def [blobs_by_mask](#) (img)

5.4.1 Detailed Description

A utility file for testing out computer vision techniques on preset images.
The purpose of this is to avoid using the webcam, and test on consistent test cases.

5.4.2 Function Documentation

5.4.2.1 `def experiment.binary_mask (img)`

5.4.2.2 `def experiment.blobs_by_mask (img)`

5.4.2.3 `def experiment.dilation_and_blur (img)`

5.4.2.4 `def experiment.experiment (image_function = None, blob_function = None, directory = " . / ")`

5.4.2.5 `def experiment.hard_threshold (img)`

5.5 image_support Namespace Reference

Functions

- `def external_init_particle_filter (img)`
Initializes particle filter.
- `def image_hue_filter`
Converts given image to HSV based on the given color.
- `def get_hue_blobs (img)`
Gets basket blobs after hue distance filtering.
- `def get_best_blob (blobs, particle_filter)`
Returns the best blob out of the provided set and particle filter.
- `def is_blob_in_middle_helper (img, blob)`
Determines whether the given blob is in center of image.

5.5.1 Function Documentation

5.5.1.1 `def image_support.external_init_particle_filter (img)`

Initializes particle filter.

Parameters

<i>img</i>	SimpleCV.Image captured image
------------	-------------------------------

Returns

A ParticleFilter object

5.5.1.2 `def image_support.get_best_blob (blobs, particle_filter)`

Returns the best blob out of the provided set and particle filter.

Parameters

<i>blobs</i>	list of potential HSV blobs
<i>particle_filter</i>	initialized ParticleFilter object

Returns

The largest blob found or None.

5.5.1.3 `def image_support.get_hue_blobs (img)`

Gets basket blobs after hue distance filtering.

Parameters

<i>img</i>	SimpleCV.Image captured image.
------------	--------------------------------

Returns

Set of 'black' potential blobs.

5.5.1.4 def image_support.image_hue_filter (*img*, *ball* = True)

Converts given image to HSV based on the given color.

Parameters

<i>img</i>	SimpleCV.Image captured image
<i>color</i>	tuple of RGB values of single 'H' value of HSV

Returns

HSV converted image

5.5.1.5 def image_support.is_blob_in_middle_helper (*img*, *blob*)

Determines whether the given blob is in center of image.

Parameters

<i>img</i>	SimpleCV.Image captured image
<i>blob</i>	SimpleCV.Blob Blob object

Returns

True if blob in middle of image, false otherwise.

5.6 particlefilter Namespace Reference

Classes

- class [ParticleFilter](#)

5.7 prquadtree Namespace Reference

Classes

- class [Box](#)
Class defining a square on the coordinate system via a center point and half of square width.
- class [Particle](#)
Represents particle point.
- class [Point](#)
Represents an (x,y) coordinate point on a grid.
- class [PRQuadTree](#)
Class representing a [Point](#) Range Quadtree.

5.7.1 Detailed Description

Implementation of a Point Range Quadtree.

Author: Pawel Szczurko

5.8 prquadtree_test Namespace Reference

Classes

- class [TestBox](#)
- class [TestParticle](#)
- class [TestPoint](#)
- class [TestPrQuadTree](#)

5.9 prquadtree_test_example Namespace Reference

Variables

- tuple [b](#) = [Box](#)([Point](#)(5,5), 50)
- tuple [b2](#) = [Box](#)([Point](#)(50,50), 50)
- tuple [qt](#) = [PRQuadTree](#)([b2](#))
- tuple [pt](#) = [Point](#)(2,2)
- tuple [nearby](#) = [qt.query_k_nearest](#)([pt](#), 20)
- int [c](#) = 1

5.9.1 Detailed Description

File testing the capabilities of the PRQuadTree.

Author: Pawel Szczurko

5.9.2 Variable Documentation

5.9.2.1 tuple [prquadtree_test_example.b](#) = [Box](#)([Point](#)(5,5), 50)

5.9.2.2 tuple [prquadtree_test_example.b2](#) = [Box](#)([Point](#)(50,50), 50)

5.9.2.3 int [prquadtree_test_example.c](#) = 1

5.9.2.4 tuple [prquadtree_test_example.nearby](#) = [qt.query_k_nearest](#)([pt](#), 20)

5.9.2.5 tuple [prquadtree_test_example.pt](#) = [Point](#)(2,2)

5.9.2.6 tuple [prquadtree_test_example.qt](#) = [PRQuadTree](#)([b2](#))

5.10 tennis_ball Namespace Reference

Functions

- def [is_ball_middle](#) (img)

Entry point for module which determines whether tennis ball is in the middle of the image.

- def `run` ()

Continuously captures image from computer camera and feeds it to the `is_ball_middle` method to detect whether tennis ball is in the middle of the screen.

Variables

- `particle_filter` = None

5.10.1 Detailed Description

Simple detection of ball using SimpleCV (much easier than OpenCV). The `run` method identifies a tennis ball in the camera stream image. '`is_ball_middle`' function can be used to determine whether a ball is horizontally centered based on a specified threshold.

—Pawel Szczurko

5.10.2 Function Documentation

5.10.2.1 def tennis_ball.is_ball_middle (`img`)

Entry point for module which determines whether tennis ball is in the middle of the image.

Parameters

<code>img</code>	SimpleCV.Image
------------------	----------------

Returns

boolean. True if tennis ball is in middle, false otherwise.

5.10.2.2 def tennis_ball.run ()

Continuously captures image from computer camera and feeds it to the `is_ball_middle` method to detect whether tennis ball is in the middle of the screen.

5.10.3 Variable Documentation

5.10.3.1 tennis_ball.particle_filter = None

5.11 tennis_ball_runner Namespace Reference

5.12 tennis_ball_test Namespace Reference

Functions

- def `unitTest` (actual, expected, name)
- def `ballPresent` ()
- def `ballMissing` ()

5.12.1 Function Documentation

5.12.1.1 `def tennis_ball_test.ballMissing ()`

5.12.1.2 `def tennis_ball_test.ballPresent ()`

5.12.1.3 `def tennis_ball_test.unitTest (actual, expected, name)`

Chapter 6

Class Documentation

6.1 prquadtree.Box Class Reference

Class defining a square on the coordinate system via a center point and half of square width.

Public Member Functions

- def `__init__` (self, `center`, `half_size`)
Construct a `Box` object.
- def `contains_point` (self, point)
Verifies that the given point is within this square.
- def `intersect` (self, other_box)
Checks if the provided box/square intersects with this square.

Public Attributes

- `center`
- `half_size`

6.1.1 Detailed Description

Class defining a square on the coordinate system via a center point and half of square width.

6.1.2 Constructor & Destructor Documentation

6.1.2.1 def prquadtree.Box.__init__ (self, center, half_size)

Construct a `Box` object.

Parameters

<code>center</code>	<code>Point</code> type specifying the center of the square
<code>half_size</code>	int half the length of the square

6.1.3 Member Function Documentation

6.1.3.1 `def prquadtree.Box.contains_point (self, point)`

Verifies that the given point is within this square.

Parameters

<i>point</i>	Point type to check if it's in the square
--------------	---

Returns

boolean indicating whether the point is within the square

6.1.3.2 def prquadtree.Box.intersect (*self*, *other_box*)

Checks if the provided box/square intersects with this square.

Parameters

<i>other_box</i>	Box object
------------------	----------------------------

Returns

Boolean indicating if the two intersect anywhere

6.1.4 Member Data Documentation

6.1.4.1 prquadtree.Box.center

6.1.4.2 prquadtree.Box.half_size

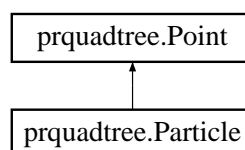
The documentation for this class was generated from the following file:

- [prquadtree.py](#)

6.2 prquadtree.Particle Class Reference

Represents particle point.

Inheritance diagram for prquadtree.Particle:



Public Member Functions

- def [__init__](#) (self, x, y)
Constructs a [Particle](#).

Public Attributes

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

6.2.1 Detailed Description

Represents particle point.

6.2.2 Constructor & Destructor Documentation

6.2.2.1 `def prquadtree.Particle.__init__(self, x, y)`

Constructs a [Particle](#).

Parameters

<code>x</code>	float/int x-position
<code>y</code>	float/int y-position

6.2.3 Member Data Documentation

6.2.3.1 `prquadtree.Particle.score`

6.2.3.2 `prquadtree.Particle.x`

6.2.3.3 `prquadtree.Particle.y`

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

- [prquadtree.py](#)

6.3 particlefilter.ParticleFilter Class Reference

Public Member Functions

- `def __init__(self, box)`
Constructor.
- `def iterate(self, blobs)`
For each blob, it updates the points in the tree increasing the score of those which are within the bounding square of the blob.
- `def score(self, blob)`
Returns the sum of the scores of the points found within this blob by querying the quadtree.
- `def clear_scores(self)`
Resets all scores of blobs This should be used when changing the webcam view.

Public Attributes

- [pr_tree](#)
- [image_box](#)
- [iterations](#)
- [iterations_before_clearing](#)

6.3.1 Constructor & Destructor Documentation

6.3.1.1 `def particlefilter.ParticleFilter.__init__(self, box)`

Constructor.

Parameters

<i>box</i>	Box the box representing the web cam view
------------	---

6.3.2 Member Function Documentation

6.3.2.1 `def particlefilter.ParticleFilter.clear_scores (self)`

Resets all scores of blobs This should be used when changing the webcam view.

6.3.2.2 `def particlefilter.ParticleFilter.iterate (self, blobs)`

For each blob, it updates the points in the tree increasing the score of those which are within the bounding square of the blob.

Parameters

<i>blobs</i>	array An array of blob objects which were matched
--------------	---

6.3.2.3 `def particlefilter.ParticleFilter.score (self, blob)`

Returns the sum of the scores of the points found within this blob by querying the quadtree.

Parameters

<i>blob</i>	Blob A single blob
-------------	--------------------

Returns

int The sum of the scores of the points contained in the passed blob

6.3.3 Member Data Documentation

6.3.3.1 `particlefilter.ParticleFilter.image_box`6.3.3.2 `particlefilter.ParticleFilter.iterations`6.3.3.3 `particlefilter.ParticleFilter.iterations_before_clearing`6.3.3.4 `particlefilter.ParticleFilter.pr_tree`

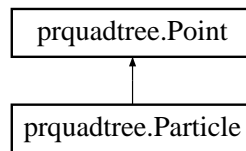
The documentation for this class was generated from the following file:

- [particlefilter.py](#)

6.4 prquadtree.Point Class Reference

Represents an (x,y) coordinate point on a grid.

Inheritance diagram for prquadtree.Point:



Public Member Functions

- `def __init__(self, x, y)`
Constructs a coordinate [Point](#).
- `def __str__(self)`
Overwriting the default to string method of the [Point](#) class.
- `def __repr__(self)`
Needed for printing via 'print'.

Public Attributes

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

6.4.1 Detailed Description

Represents an (x,y) coordinate point on a grid.

6.4.2 Constructor & Destructor Documentation

6.4.2.1 `def prquadtree.Point.__init__(self, x, y)`

Constructs a coordinate [Point](#).

Parameters

x	float/int x-position
y	float/int y-position

6.4.3 Member Function Documentation

6.4.3.1 `def prquadtree.Point.__repr__(self)`

Needed for printing via 'print'.

6.4.3.2 `def prquadtree.Point.__str__(self)`

Overwriting the default to string method of the [Point](#) class.

Returns

String representation of [Point](#)

6.4.4 Member Data Documentation

6.4.4.1 prquadtree.Point.x

6.4.4.2 prquadtree.Point.y

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

- [prquadtree.py](#)

6.5 prquadtree.PRQuadTree Class Reference

Class representing a [Point](#) Range Quadtree.

Public Member Functions

- def [__init__](#) (self, [box](#))
Constructs a PR Quadtree given an initial square.
- def [insert](#) (self, point)
Inserts a point into the PRQuadtree.
- def [query_range](#) (self, rng)
Returns the points in the provided range.
- def [query_k_nearest](#) (self, point, k)
Returns k points closest to the provided point.
- def [print_all_points](#) (self, root)
Prints all points stored in the PRQuadtree.
- def [__str__](#) (self)
Prints the points of the nw,ne,sw,se blocks of the given [PRQuadTree](#) node.

Static Public Member Functions

- def [size](#) (prtree)
Static method that determines the size of the given tree.

Public Attributes

- [box](#)
- [points](#)
- [nw](#)
- [ne](#)
- [sw](#)
- [se](#)

Static Public Attributes

- int [QT_NODE_CAPACITY](#) = 20

6.5.1 Detailed Description

Class representing a [Point](#) Range Quadtree.

6.5.2 Constructor & Destructor Documentation

6.5.2.1 `def prquadtree.PRQuadTree.__init__(self, box)`

Constructs a PR Quadtree given an initial square.

Parameters

<i>box</i>	Box representing initial square
------------	---

6.5.3 Member Function Documentation

6.5.3.1 `def prquadtree.PRQuadTree.__str__(self)`

Prints the points of the nw,ne,sw,se blocks of the given [PRQuadTree](#) node.

Returns

String A string of points in the blocks Generates string based on the number of points stored in the provided node.

Parameters

<i>loc</i>	PRQuadTree a PRQuadTree node (ie nw,ne,sw,se)
<i>name</i>	String

Returns

String A string with point and name

6.5.3.2 `def prquadtree.PRQuadTree.insert(self, point)`

Inserts a point into the PRQuadtree.

Parameters

<i>point</i>	Point
--------------	-----------------------

Returns

A boolean returning true on success, false on failure.

6.5.3.3 `def prquadtree.PRQuadTree.print_all_points(self, root)`

Prints all points stored in the PRQuadtree.

Parameters

<i>root</i>	PRQuadTree start point, or the root of the Quadtree
-------------	---

Returns

String a string with coordinates

6.5.3.4 `def prquadtree.PRQuadTree.query_k_nearest (self, point, k)`

Returns k points closest to the provided point.

Parameters

<i>point</i>	Point a Point from which to search for other points.
<i>k</i>	int number of closest points to return

Returns

array A coordinate distance between the search point and the provided point Internal method used to provide python method with a key (coordinate distance) on which to sort.

Parameters

<i>p</i>	Point
----------	-----------------------

Returns

float

6.5.3.5 `def prquadtree.PRQuadTree.query_range (self, rng)`

Returns the points in the provided range.

Parameters

<i>rng</i>	Box a Box range from which to retrieve points
------------	---

Returns

A list of points within the provided range

6.5.3.6 `def prquadtree.PRQuadTree.size (prtree) [static]`

Static method that determines the size of the given tree.

Keeping an insertion count in the client code would be preferred to this due to heavy recursion.

Parameters

<i>prtree</i>	PRQuadTree
---------------	----------------------------

Returns

int An integer representing the number of points in the given tree.

6.5.4 Member Data Documentation

6.5.4.1 `prquadtree.PRQuadTree.box`6.5.4.2 `prquadtree.PRQuadTree.ne`6.5.4.3 `prquadtree.PRQuadTree.nw`6.5.4.4 `prquadtree.PRQuadTree.points`

6.5.4.5 `int prquadtree.PRQuadTree.QT_NODE_CAPACITY = 20` `[static]`

6.5.4.6 `prquadtree.PRQuadTree.se`

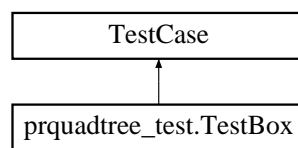
6.5.4.7 `prquadtree.PRQuadTree.sw`

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

- [prquadtree.py](#)

6.6 prquadtree_test.TestBox Class Reference

Inheritance diagram for `prquadtree_test.TestBox`:



Public Member Functions

- `def test_box_insert (self)`
- `def test_box_contains (self)`

6.6.1 Member Function Documentation

6.6.1.1 `def prquadtree_test.TestBox.test_box_contains (self)`

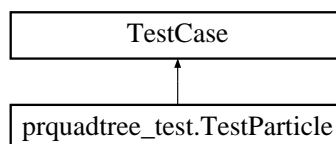
6.6.1.2 `def prquadtree_test.TestBox.test_box_insert (self)`

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

- [prquadtree_test.py](#)

6.7 prquadtree_test.TestParticle Class Reference

Inheritance diagram for `prquadtree_test.TestParticle`:



Public Member Functions

- `def test_particle_insert (self)`

6.7.1 Member Function Documentation

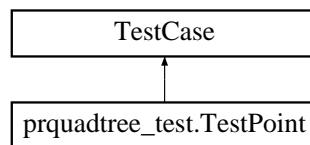
6.7.1.1 `def prquadtree_test.TestParticle.test_particle_insert (self)`

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

- [prquadtree_test.py](#)

6.8 prquadtree_test.TestPoint Class Reference

Inheritance diagram for prquadtree_test.TestPoint:



Public Member Functions

- `def test_point_insert (self)`

6.8.1 Member Function Documentation

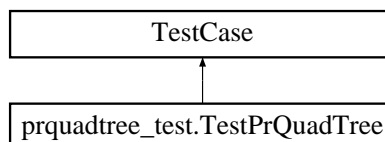
6.8.1.1 `def prquadtree_test.TestPoint.test_point_insert (self)`

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

- [prquadtree_test.py](#)

6.9 prquadtree_test.TestPrQuadTree Class Reference

Inheritance diagram for prquadtree_test.TestPrQuadTree:



Public Member Functions

- `def test_insert (self)`
- `def test_nearby (self)`

6.9.1 Member Function Documentation

6.9.1.1 `def prquadtree_test.TestPrQuadTree.test_insert (self)`

6.9.1.2 `def prquadtree_test.TestPrQuadTree.test_nearby (self)`

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

- [prquadtree_test.py](#)

Chapter 7

File Documentation

7.1 basket.py File Reference

Namespaces

- `basket`

Functions

- `def basket.is_basket_middle (img)`
Single entry function returning True/False if basket is in the middle of the screen.
- `def basket.run_middle ()`
Runs continuously and prints if the best detected blob is in the middle.
- `def basket.run`
Runs continuously outlines best matched blob if it is in the middle.

Variables

- `basket.particle_filter = None`
- `int basket.image_half_size = -1`
- `int basket.save_count = 1`
- `tuple basket.base_filename = datetime.now()`

7.2 basket_runner.py File Reference

Namespaces

- `basket_runner`

7.3 basket_test.py File Reference

Namespaces

- `basket_test`

Functions

- def [basket_test.unitTest](#) (actual, expected, name)
- def [basket_test.basketPresent](#) ()
- def [basket_test.basketMissing](#) ()

7.4 experiment.py File Reference

Namespaces

- [experiment](#)

Functions

- def [experiment.experiment](#)
- def [experiment.hard_threshold](#) (img)
- def [experiment.binary_mask](#) (img)
- def [experiment.dilation_and_blur](#) (img)
- def [experiment.blobs_by_mask](#) (img)

7.5 image_support.py File Reference

Namespaces

- [image_support](#)

Functions

- def [image_support.external_init_particle_filter](#) (img)
Initializes particle filter.
- def [image_support.image_hue_filter](#)
Converts given image to HSV based on the given color.
- def [image_support.get_hue_blobs](#) (img)
Gets basket blobs after hue distance filtering.
- def [image_support.get_best_blob](#) (blobs, particle_filter)
Returns the best blob out of the provided set and particle filter.
- def [image_support.is_blob_in_middle_helper](#) (img, blob)
Determines whether the given blob is in ceter of image.

7.6 particlefilter.py File Reference

Classes

- class [particlefilter.ParticleFilter](#)

Namespaces

- [particlefilter](#)

7.7 prquadtree.py File Reference

Classes

- class [prquadtree.Point](#)
Represents an (x,y) coordinate point on a grid.
- class [prquadtree.Particle](#)
Represents particle point.
- class [prquadtree.Box](#)
Class defining a square on the coordinate system via a center point and half of square width.
- class [prquadtree.PRQuadTree](#)
Class representing a [Point](#) Range Quadtree.

Namespaces

- [prquadtree](#)

7.8 prquadtree_test.py File Reference

Classes

- class [prquadtree_test.TestPoint](#)
- class [prquadtree_test.TestParticle](#)
- class [prquadtree_test.TestBox](#)
- class [prquadtree_test.TestPrQuadTree](#)

Namespaces

- [prquadtree_test](#)

7.9 prquadtree_test_example.py File Reference

Namespaces

- [prquadtree_test_example](#)

Variables

- tuple [prquadtree_test_example.b](#) = [Box](#)([Point](#)(5,5), 50)
- tuple [prquadtree_test_example.b2](#) = [Box](#)([Point](#)(50,50), 50)
- tuple [prquadtree_test_example.qt](#) = [PRQuadTree](#)(b2)
- tuple [prquadtree_test_example.pt](#) = [Point](#)(2,2)
- tuple [prquadtree_test_example.nearby](#) = [qt.query_k_nearest](#)(pt, 20)
- int [prquadtree_test_example.c](#) = 1

7.10 tennis_ball.py File Reference

Namespaces

- [tennis_ball](#)

Functions

- def [tennis_ball.is_ball_middle](#) (img)
Entry point for module which determines whether tennis ball is in the middle of the image.
- def [tennis_ball.run](#) ()
Continuously captures image from computer camera and feeds it to the [is_ball_middle](#) method to detect whether tennis ball is in the middle of the screen.

Variables

- [tennis_ball.particle_filter](#) = None

7.11 tennis_ball_runner.py File Reference

Namespaces

- [tennis_ball_runner](#)

7.12 tennis_ball_test.py File Reference

Namespaces

- [tennis_ball_test](#)

Functions

- def [tennis_ball_test.unitTest](#) (actual, expected, name)
- def [tennis_ball_test.ballPresent](#) ()
- def [tennis_ball_test.ballMissing](#) ()

7.13 visual.h File Reference

Functions

- int [start_visual](#) (void)
- void [set_objects](#) (object_t *objs)
- void [get_objects](#) (object_t *objs, point_t *locations)
- void [stop_visual](#) (void)

7.13.1 Function Documentation

7.13.1.1 void [get_objects](#) (object_t * *objs*, point_t * *locations*)

7.13.1.2 void [set_objects](#) (object_t * *objs*)

7.13.1.3 int [start_visual](#) (void)

7.13.1.4 void [stop_visual](#) (void)

Index

- `__init__`
 - `particlefilter::ParticleFilter`, 20
 - `prquadtree::Box`, 17
 - `prquadtree::PRQuadTree`, 24
 - `prquadtree::Particle`, 20
 - `prquadtree::Point`, 22
 - `__repr__`
 - `prquadtree::Point`, 22
 - `__str__`
 - `prquadtree::PRQuadTree`, 24
 - `prquadtree::Point`, 22
- b
 - `prquadtree_test_example`, 14
- b2
 - `prquadtree_test_example`, 14
- `ballMissing`
 - `tennis_ball_test`, 16
- `ballPresent`
 - `tennis_ball_test`, 16
- `base_filename`
 - `basket`, 10
- `basket`, 9
 - `base_filename`, 10
 - `image_half_size`, 10
 - `is_basket_middle`, 9
 - `particle_filter`, 10
 - `run`, 9
 - `run_middle`, 10
 - `save_count`, 10
- `basket.py`, 31
- `basket_runner`, 10
- `basket_runner.py`, 31
- `basket_test`, 10
 - `basketMissing`, 10
 - `basketPresent`, 10
 - `unitTest`, 10
- `basket_test.py`, 31
- `basketMissing`
 - `basket_test`, 10
- `basketPresent`
 - `basket_test`, 10
- `binary_mask`
 - `experiment`, 11
- `blobs_by_mask`
 - `experiment`, 11
- `box`
 - `prquadtree::PRQuadTree`, 26

- c
 - `prquadtree_test_example`, 14
- `center`
 - `prquadtree::Box`, 19
- `clear_scores`
 - `particlefilter::ParticleFilter`, 21
- `contains_point`
 - `prquadtree::Box`, 18
- `dilation_and_blur`
 - `experiment`, 11
- `experiment`, 10
 - `binary_mask`, 11
 - `blobs_by_mask`, 11
 - `dilation_and_blur`, 11
 - `experiment`, 11
 - `hard_threshold`, 11
- `experiment.py`, 32
- `external_init_particle_filter`
 - `image_support`, 11
- `get_best_blob`
 - `image_support`, 11
- `get_hue_blobs`
 - `image_support`, 11
- `get_objects`
 - `visual.h`, 34
- `half_size`
 - `prquadtree::Box`, 19
- `hard_threshold`
 - `experiment`, 11
- `image_box`
 - `particlefilter::ParticleFilter`, 21
- `image_half_size`
 - `basket`, 10
- `image_hue_filter`
 - `image_support`, 13
- `image_support`, 11
 - `external_init_particle_filter`, 11
 - `get_best_blob`, 11
 - `get_hue_blobs`, 11
 - `image_hue_filter`, 13
 - `is_blob_in_middle_helper`, 13
- `image_support.py`, 32
- `insert`
 - `prquadtree::PRQuadTree`, 24
- `intersect`
 - `prquadtree::Box`, 19
- `is_ball_middle`

- tennis_ball, 15
- is_basket_middle
 - basket, 9
- is_blob_in_middle_helper
 - image_support, 13
- iterate
 - particlefilter::ParticleFilter, 21
- iterations
 - particlefilter::ParticleFilter, 21
- iterations_before_clearing
 - particlefilter::ParticleFilter, 21
- ne
 - prquadtree::PRQuadTree, 26
- nearby
 - prquadtree_test_example, 14
- nw
 - prquadtree::PRQuadTree, 26
- particle_filter
 - basket, 10
 - tennis_ball, 15
- particlefilter, 13
- particlefilter.ParticleFilter, 20
- particlefilter.py, 32
- particlefilter::ParticleFilter
 - __init__, 20
 - clear_scores, 21
 - image_box, 21
 - iterate, 21
 - iterations, 21
 - iterations_before_clearing, 21
 - pr_tree, 21
 - score, 21
- points
 - prquadtree::PRQuadTree, 26
- pr_tree
 - particlefilter::ParticleFilter, 21
- print_all_points
 - prquadtree::PRQuadTree, 24
- prquadtree, 13
- prquadtree.Box, 17
- prquadtree.PRQuadTree, 23
- prquadtree.Particle, 19
- prquadtree.Point, 21
- prquadtree.py, 33
- prquadtree::Box
 - __init__, 17
 - center, 19
 - contains_point, 18
 - half_size, 19
 - intersect, 19
- prquadtree::PRQuadTree
 - __init__, 24
 - __str__, 24
 - box, 26
 - insert, 24
 - ne, 26
 - nw, 26
 - points, 26
 - print_all_points, 24
 - QT_NODE_CAPACITY, 26
 - query_k_nearest, 24
 - query_range, 26
 - se, 27
 - size, 26
 - sw, 27
- prquadtree::Particle
 - __init__, 20
 - score, 20
 - x, 20
 - y, 20
- prquadtree::Point
 - __init__, 22
 - __repr__, 22
 - __str__, 22
 - x, 23
 - y, 23
- prquadtree_test, 14
- prquadtree_test.py, 33
- prquadtree_test.TestBox, 27
- prquadtree_test.TestParticle, 27
- prquadtree_test.TestPoint, 28
- prquadtree_test.TestPrQuadTree, 28
- prquadtree_test::TestBox
 - test_box_contains, 27
 - test_box_insert, 27
- prquadtree_test::TestParticle
 - test_particle_insert, 28
- prquadtree_test::TestPoint
 - test_point_insert, 28
- prquadtree_test::TestPrQuadTree
 - test_insert, 28
 - test_nearby, 28
- prquadtree_test_example, 14
 - b, 14
 - b2, 14
 - c, 14
 - nearby, 14
 - pt, 14
 - qt, 14
- prquadtree_test_example.py, 33
- pt
 - prquadtree_test_example, 14
- QT_NODE_CAPACITY
 - prquadtree::PRQuadTree, 26
- qt
 - prquadtree_test_example, 14
- query_k_nearest
 - prquadtree::PRQuadTree, 24
- query_range
 - prquadtree::PRQuadTree, 26
- run
 - basket, 9
 - tennis_ball, 15
- run_middle

- basket, [10](#)
- save_count
 - basket, [10](#)
- score
 - particlefilter::ParticleFilter, [21](#)
 - prquadtree::Particle, [20](#)
- se
 - prquadtree::PRQuadTree, [27](#)
- set_objects
 - visual.h, [34](#)
- size
 - prquadtree::PRQuadTree, [26](#)
- start_visual
 - visual.h, [34](#)
- stop_visual
 - visual.h, [34](#)
- sw
 - prquadtree::PRQuadTree, [27](#)
- tennis_ball, [14](#)
 - is_ball_middle, [15](#)
 - particle_filter, [15](#)
 - run, [15](#)
- tennis_ball.py, [33](#)
- tennis_ball_runner, [15](#)
- tennis_ball_runner.py, [34](#)
- tennis_ball_test, [15](#)
 - ballMissing, [16](#)
 - ballPresent, [16](#)
 - unitTest, [16](#)
- tennis_ball_test.py, [34](#)
- test_box_contains
 - prquadtree_test::TestBox, [27](#)
- test_box_insert
 - prquadtree_test::TestBox, [27](#)
- test_insert
 - prquadtree_test::TestPrQuadTree, [28](#)
- test_nearby
 - prquadtree_test::TestPrQuadTree, [28](#)
- test_particle_insert
 - prquadtree_test::TestParticle, [28](#)
- test_point_insert
 - prquadtree_test::TestPoint, [28](#)
- unitTest
 - basket_test, [10](#)
 - tennis_ball_test, [16](#)
- visual.h, [34](#)
 - get_objects, [34](#)
 - set_objects, [34](#)
 - start_visual, [34](#)
 - stop_visual, [34](#)
- x
 - prquadtree::Particle, [20](#)
 - prquadtree::Point, [23](#)
- y