

# Package: panstarrs (via r-universe)

October 9, 2024

**Title** Interface to the Pan-STARRS API

**Version** 0.2.2

**Description** An interface to the API for 'Pan-STARRS1', a data archive of the PS1 wide-field astronomical survey. The package allows access to the PS1 catalog and to the PS1 images. (see <https://outerspace.stsci.edu/display/PANSTARRS/> for more information). You can use it to plan astronomical observations, make guidance pictures, find magnitudes in five broadband filters (g, r, i, z, y) and more.

**License** MIT + file LICENSE

**URL** <https://uskovgs.github.io/PanSTARRS/>

**BugReports** <https://github.com/uskovgs/PanSTARRS/issues>

**Depends** R (>= 3.5)

**Imports** bit64, checkmate, curl, data.table, httr, jsonlite

**Suggests** dplyr, knitr, magick, rmarkdown, testthat (>= 3.0.0)

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Encoding** UTF-8

**RoxygenNote** 7.3.1

**Repository** <https://uskovgs.r-universe.dev>

**RemoteUrl** <https://github.com/uskovgs/panstarrs>

**RemoteRef** HEAD

**RemoteSha** b4d88002ddaafdcc33fdabc11726332fad1ffff

## Contents

|                           |   |
|---------------------------|---|
| ps1_cone . . . . .        | 2 |
| ps1_crossmatch . . . . .  | 3 |
| ps1_image_color . . . . . | 4 |
| ps1_image_gray . . . . .  | 5 |

|                            |   |
|----------------------------|---|
| ps1_image_list . . . . .   | 6 |
| ps1_image_url . . . . .    | 6 |
| ps1_mast_resolve . . . . . | 8 |
| ps1_metadata . . . . .     | 8 |
| ps1_resolve . . . . .      | 9 |
| ps1_search . . . . .       | 9 |

|              |           |
|--------------|-----------|
| <b>Index</b> | <b>11</b> |
|--------------|-----------|

---

|          |  |
|----------|--|
| ps1_cone | <i>Do a cone search of the PS1 catalog</i> |
|----------|--|

---

## Description

Do a cone search of the PS1 catalog

## Usage

```
ps1_cone(
  ra,
  dec,
  r_arcmin = 0.05,
  table = c("mean", "stack", "detection", "forced_mean"),
  release = c("dr2", "dr1"),
  columns = NULL,
  verbose = FALSE,
  ...
)
```

## Arguments

|          |   |
|----------|---|
| ra       | (degrees) J2000 Right Ascension                           |
| dec      | (degrees) J2000 Declination                               |
| r_arcmin | (arcmins) Search radius (<= 30 arcmins)                   |
| table    | "mean"(default), "stack", "detection" or "forced_mean"    |
| release  | "dr1" or "dr2"(default)                                   |
| columns  | list of column names to include (NULL means use defaults) |
| verbose  | print info about request                                  |
| ...      | other parameters (e.g., nDetections.min = 2)              |

## Value

data.frame

## Examples

```
## Not run:
ps1_cone(ra = 139.334, dec = 68.635, r_arcmin = 0.05, nDetections.gt = 1)

## End(Not run)
```

---

|                |  |
|----------------|--|
| ps1_crossmatch | <i>Do a cross-match with PS1 catalog</i> |
|----------------|--|

---

## Description

Do a cross-match with PS1 catalog

## Usage

```
ps1_crossmatch(
  ra,
  dec,
  r_arcmin = 0.05,
  table = c("mean", "stack", "detection", "forced_mean"),
  release = c("dr2", "dr1"),
  verbose = FALSE
)
```

## Arguments

|          |  |
|----------|--|
| ra       | (degrees) numeric vector of J2000 Right Ascension    |
| dec      | (degrees) numeric vector of J2000 Declination        |
| r_arcmin | (arcmins) Search radius ( $\leq 30$ arcmins)         |
| table    | "mean"(default), "stack", "detection", "forced_mean" |
| release  | "dr1" or "dr2"(default)                              |
| verbose  | print info about request                             |

## Value

data.frame

## Examples

```
## Not run:
ps1_crossmatch(ra = c(268.70342, 168.87258), dec = c(71.54292, 60.75153))

## End(Not run)
```

---

ps1\_image\_color      *Get color image at a sky position*

---

### Description

Get color image at a sky position

### Usage

```
ps1_image_color(  
    ra,  
    dec,  
    size = 240,  
    output_size = NULL,  
    filters = "grizy",  
    format = "jpg"  
)
```

### Arguments

|             |   |
|-------------|---|
| ra          | ra position in degrees  |
| dec         | dec position in degrees   |
| size        | extracted image size in pixels (0.25 arcsec/pixel)  |
| output_size | output (display) image size in pixels (default = size). output_size has no effect for fits format images. |
| filters     | string with filters to include  |
| format      | data format (options are "jpg", "png")  |

### Value

the image url

### Examples

```
## Not run:  
ps1_image_color(ra = 83.633210, dec = 22.014460, size = 1280, filters="grz")  
  
## End(Not run)
```

---

|                |  |
|----------------|--|
| ps1_image_gray | <i>Get grayscale image at a sky position</i> |
|----------------|--|

---

### Description

Get grayscale image at a sky position

### Usage

```
ps1_image_gray(  
    ra,  
    dec,  
    size = 240,  
    output_size = NULL,  
    filter = "g",  
    format = "jpg"  
)
```

### Arguments

|             |   |
|-------------|---|
| ra          | ra position in degrees  |
| dec         | dec position in degrees   |
| size        | extracted image size in pixels (0.25 arcsec/pixel)  |
| output_size | output (display) image size in pixels (default = size). output_size has no effect for fits format images. |
| filter      | string with filter to extract (one of grizy)  |
| format      | data format (options are "jpg", "png")  |

### Value

the image

### Examples

```
## Not run:  
ps1_image_gray(ra = 83.633210, dec = 22.014460, size = 1280, filter = "i")  
  
## End(Not run)
```

---

|                |                           |
|----------------|---------------------------|
| ps1_image_list | <i>Get list of images</i> |
|----------------|---------------------------|

---

**Description**

Query ps1filenames.py service to get a list of images.

**Usage**

```
ps1_image_list(ra, dec, size = 240, filters = "grizy")
```

**Arguments**

|         |  |
|---------|--|
| ra      | ra position in degrees                   |
| dec     | dec position in degrees                  |
| size    | image size in pixels (0.25 arcsec/pixel) |
| filters | string with filters to include           |

**Details**

src: <https://ps1images.stsci.edu/ps1image.html>

**Value**

table with the results

**Examples**

```
## Not run:
# Crab nebulae image
ps1_image_list(ra = 83.633210, dec = 22.014460, size = 1280, filters = "grz")

## End(Not run)
```

---

|               |                          |
|---------------|--------------------------|
| ps1_image_url | <i>Get URL of images</i> |
|---------------|--------------------------|

---

**Description**

Get URL of images

### Usage

```
ps1_image_url(  
  ra,  
  dec,  
  size = 240,  
  output_size = NULL,  
  filters = "grizy",  
  format = "jpg",  
  color = FALSE  
)
```

### Arguments

|             |   |
|-------------|---|
| ra          | ra position in degrees  |
| dec         | dec position in degrees   |
| size        | extracted image size in pixels (0.25 arcsec/pixel)  |
| output_size | output (display) image size in pixels (default = size). output_size has no effect for fits format images.                         |
| filters     | string with filters to include  |
| format      | data format (options are "jpg", "png" or "fits")  |
| color       | if TRUE, creates a color image (only for jpg or png format). Default is return a list of URLs for single-filter grayscale images. |

### Value

string with the URL

### Examples

```
## Not run:  
ps1_image_url(  
  ra = 83.633210,  
  dec = 22.014460,  
  size = 1280,  
  format = "jpg",  
  filters = "grz",  
  color = T)  
  
## End(Not run)
```

---

ps1\_mast\_resolve      *Get the RA and Dec for an object using the MAST name resolver*

---

**Description**

Get the RA and Dec for an object using the MAST name resolver

**Usage**

```
ps1_mast_resolve(name)
```

**Arguments**

name                  Name of object

**Value**

list of ra, decl

**Examples**

```
## Not run:  
ps1_mast_resolve('Acrux')  
  
## End(Not run)
```

---

ps1\_metadata      *Metadata from PSI*

---

**Description**

Return metadata for the specified catalog and table

**Usage**

```
ps1_metadata(table = "mean", release = "dr2")
```

**Arguments**

table                  "mean", "stack", "forced\_mean" or "detection"  
release                "dr1" or "dr2"(default)

**Value**

Returns data.frame with columns: name, type, description



**Examples**

```
## Not run:
ps1_metadata()

## End(Not run)
```

---

ps1\_resolve

*Get the RA and Dec for objects from PanSTARRS catalog.*


---

**Description**

Only works for "north" objects with decl > -30. For all objects see function 'ps1\_mast\_resolve'.

**Usage**

```
ps1_resolve(target_names, verbose = FALSE)
```

**Arguments**

```
target_names    character vector of target names (see example)
verbose         print info about request
```

**Value**

```
data.frame
```

**Examples**

```
## Not run:
ps1_resolve(c('Andromeda', "SN 2005D", 'Antennae', 'ANTENNAE'))

## End(Not run)
```

---

ps1\_search

*Do a general search of the PS1 catalog (possibly without ra/dec/radius)*


---

**Description**

Do a general search of the PS1 catalog (possibly without ra/dec/radius)

**Usage**

```
ps1_search(  
  table = c("mean", "stack", "detection", "forced_mean"),  
  release = c("dr2", "dr1"),  
  columns = NULL,  
  verbose = FALSE,  
  ...  
)
```

**Arguments**

|         |   |
|---------|---|
| table   | "mean", "stack", "detection" or "forced_mean"             |
| release | "dr1" or "dr2"(default)                                   |
| columns | list of column names to include (NULL means use defaults) |
| verbose | print info about request                                  |
| ...     | other parameters (e.g., nDetections.min = 2).             |

**Value**

data.frame

**Examples**

```
## Not run:  
ps1_search(  
  table='detection',  
  release='dr2',  
  objid = '190361393344112894')  
  
ps1_search(  
  table='mean',  
  release='dr2',  
  objid = '190361393344112894',  
  columns = c('objName', 'raMean', 'decMean', 'rMeanPSFMag'))  
  
## End(Not run)
```

# Index

ps1\_cone, [2](#)  
ps1\_crossmatch, [3](#)  
ps1\_image\_color, [4](#)  
ps1\_image\_gray, [5](#)  
ps1\_image\_list, [6](#)  
ps1\_image\_url, [6](#)  
ps1\_mast\_resolve, [8](#)  
ps1\_metadata, [8](#)  
ps1\_resolve, [9](#)  
ps1\_search, [9](#)