Code documentation

Generated by Doxygen 1.9.1

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

Chapter 2

Class Index

2.1 Class List

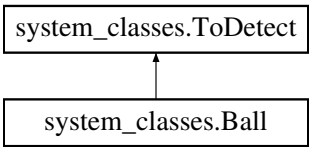
Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

Class Documentation

3.1 system_classes.Ball Class Reference

Inheritance diagram for system_classes.Ball:



Public Member Functions

- def **__init__** (self, color)

Additional Inherited Members

3.1.1 Detailed Description

```
balle à détecter
```

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

- system_classes.py

3.2 vision.Color Class Reference

Public Member Functions

- def **__init__** (self, name, BGR)

Public Attributes

- **name**
- **BGR**

3.2.1 Detailed Description

```
couleur

Parameters
----------
name : string
    nom de la couleur
BGR : type
    code en BGR de la couleur

Attributes
----------
name
BGR
```

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

- vision.py

3.3 skynet_control.CommandeSkynet Class Reference

Public Member Functions

- def **__init__** (self, robot_index, angle=0, is_clockwise=True, grandeur=0, is_foward=True, kick=False)
- def get_command_intensity (self)

Public Attributes

- **robot_index**
- **angle**
- **is_clockwise**
- **grandeur**
- **is_foward**
- **kick**

3.3.1 Detailed Description

```
commandes créés par la décision pour chaque robot

Parameters
----------
robot_index : RobotIndex
    identifiant du robot
angle : float
    erreur sur l'angle à combler
is_clockwise : bool
    le sens de l'angle (référentiel de l'image)
grandeur : float
    erreur sur la grandeur à combler
is_foward : bool
    sens de la grandeur
kick : bool
    est-ce que le robot doit frapper la balle

Attributes
----------
robot_index
angle
is_clockwise
grandeur
is_foward
kick
```

3.3.2 Member Function Documentation

3.3.2.1 get_command_intensity()

```
def skynet_control.CommandeSkynet.get_command_intensity (
              self )

applique les PID aux commandes pour générer les données à envoyer aux robots

Returns
-------
CommandIntensity
    commande en intensité pour les robots
```

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

- skynet_control.py

3.4 skynet_control.CommandIntensity Class Reference

Public Member Functions

- def **__init__** (self, clockwise_intensity=0, foward_intensity=0)

Public Attributes

- **clockwise_intensity**
- **foward_intensity**

3.4.1 Detailed Description

```
 objet pour contenir les commandes en intensité

Parameters
----------
clockwise_intensity : int
    intensité dans le sens horaire
foward_intensity : int
    intensité vers l'avant

Attributes
----------
clockwise_intensity
foward_intensity
```

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

- skynet_control.py

3.5 config_loader.Configs Class Reference

Public Member Functions

- def **__init__** (self)

Static Public Member Functions

- def get ()
- def save (path='')

Static Public Attributes

- string **config_path** = 'config.yaml'

3.5.1 Detailed Description

```
singleton charger et accéder aux configurations du projet

Attributes
----------
config_path : string
    Chemin du fichier de configuration
__instance : dictionary
    Dictionnaire des valeurs contenue dans le fichier de configs
```

3.5.2 Member Function Documentation

3.5.2.1 get()

```
def config_loader.Configs.get ( )  [static]

renvoie crée instance ou le renvoie

Returns
-------
dictionary
    Dictionnaire des configs
```

3.5.2.2 save()

```
def config_loader.Configs.save (
              path = '' )  [static]

sauvegarde les configs dans un fichier yaml

Parameters
----------
path : string
    chemin de sauvegarde, prend celui de config_path si aucun défini
```

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

- config_loader.py

3.6 configurator.Configurator Class Reference

Public Member Functions

- def **__init__** (self, parent)
- def get_param_value (self, name)
- def save (self)
- def refresh_label (self)
- def on_closing (self)

Public Attributes

- **vc**
- **w**
- **h**
- **parent**
- **panel**
- **calibration_params**
- **image**
- **imgtk**

3.6.1 Detailed Description

```
ajoute un configurateur à une fenêtre Tkinter

Parameters
----------
parent : Tk
    fenêtre Tkinter parent

Attributes
----------
vc : VideoCapture
    capture de la caméra
w : int
    largeur de l'image
h : int
    hauteur de l'image
on_closing : func
    fonction de fermeture de la fenêtre
panel : Label
    label pour contenir l'image
calibration_params : array of TunableParam
    tableau des paramètres à calibrer
save : func
    fonction de sauvegarde des paramètres
refresh_label : func
    fonction à lancer pour rafraichir l'image du panel
parent
```

3.6.2 Member Function Documentation

3.6.2.1 get_param_value()

```
def configurator.Configurator.get_param_value (
              self,
              name )

renvoie la valeur du paramètre ayant comme nom name

Parameters
----------
name : string
    nom du paramètre

Returns
-------
float
    valeur actuel du paramètre
```

3.6.2.2 on_closing()

```
def configurator.Configurator.on_closing (
              self )

détruit la fenêtre principale
```

3.6.2.3 refresh_label()

```
def configurator.Configurator.refresh_label (
              self )
```

```
refraichi le panel avec une nouvelle image
```

3.6.2.4 save()

```
def configurator.Configurator.save (
              self )
```

```
sauvegarde les param dans calibration_params
```

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

- configurator.py

3.7 wrap.distortionRemover Class Reference

Public Member Functions

- def **__init__** (self, width, height, k1=-8.2e-6, k2=0, p1=0.0, p2=0.0)
- def **__call__** (self, img)

Public Attributes

- **distCoeff**
- **cam**

3.7.1 Detailed Description

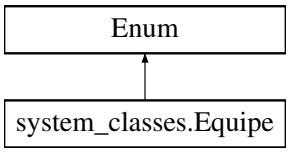
```
retire la distortion d'une image
```

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

- wrap.py

3.8 system_classes.Equipe Class Reference

Inheritance diagram for system_classes.Equipe:



Static Public Attributes

- int **HUMANITY** = 0
- int **SKYNET** = 1

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

- system_classes.py

3.9 system_classes.InfoVision Class Reference

Public Member Functions

- def **__init__** (self)
- def calculate_situation (self)
- def print (self)

Public Attributes

- **robots_info**
- **position_balle**

3.9.1 Detailed Description

```
informations provenant de la vision

Attributes
----------
robots_info : array of RobotInfo
    tableau des informations des robots
position_balle : array of float
    position de la balle
```

3.9.2 Member Function Documentation

3.9.2.1 calculate_situation()

```
def system_classes.InfoVision.calculate_situation (
              self )

calcule la situation de chaque robot
```

3.9.2.2 print()

```
def system_classes.InfoVision.print (
              self )

print les informations de la vision
```

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

- system_classes.py

3.10 skynet_control.pidList Class Reference

Static Public Member Functions

- def get (num)

Static Public Attributes

- dictionary **pids** = {}

3.10.1 Detailed Description

```
presque Singleton contenant une liste de PID

Attributes
----------
pids : dictionary of tuple of simple_pid.PID
    PID pour chaque robot de skynet en position et angle
```

3.10.2 Member Function Documentation

3.10.2.1 get()

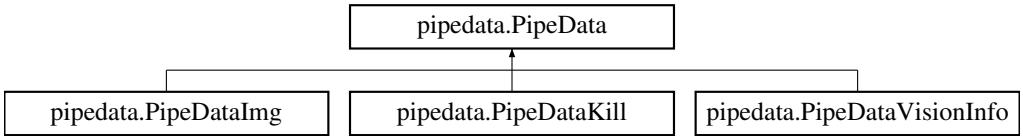
```
def skynet_control.pidList.get (
              num )  [static]

renvoie les PID de num

Parameters
----------
num : int
    indice des PID

Returns
-------
tuple of PID
    PID en angle et position
```

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

- skynet_control.py

3.11 pipedata.PipeData Class Reference

Inheritance diagram for pipedata.PipeData:



Public Member Functions

- def **__init__** (self, kill=False)

Public Attributes

- **kill**

3.11.1 Detailed Description

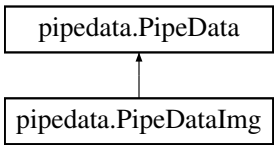
```
données à mettre entre les étages du pipeline

Parameters
----------
kill : bool
    est-ce que l'étage doit arrêter son exécution

Attributes
----------
kill
```

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

- pipedata.py

3.12 pipedata.PipeDataImg Class Reference

Inheritance diagram for pipedata.PipeDataImg:



Public Member Functions

- def **__init__** (self, img=[ ])

Public Attributes

- **img**

3.12.1 Detailed Description

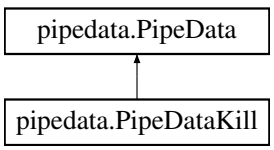
```
PipeData avec image

Parameters
----------
img : Image
    image à envoyer

Attributes
----------
img
```

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

- pipedata.py

3.13 pipedata.PipeDataKill Class Reference

Inheritance diagram for pipedata.PipeDataKill:



Public Member Functions

- def **__init__** (self)

Additional Inherited Members

3.13.1 Detailed Description

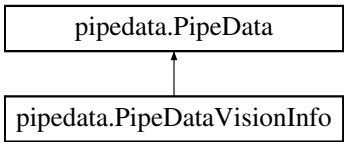
```
données contenant seulement le signal pour arrêter l'étage
```

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

- pipedata.py

3.14 pipedata.PipeDataVisionInfo Class Reference

Inheritance diagram for pipedata.PipeDataVisionInfo:



Public Member Functions

- def **__init__** (self, vision_info)

Public Attributes

- **vision_info**

3.14.1 Detailed Description

```
PipeData avec VisionInfo

Parameters
----------
vision_info : VisionInfo
    informations provenants de la vision

Attributes
----------
vision_info
```

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

- pipedata.py

3.15 pipeline.Pipeline Class Reference

Public Member Functions

- def **__init__** (self, to_detects, mqtt_client)
- def **start** (self)
- def **kill** (self)

Public Attributes

- **q_to_stage_camera**
- **q_to_display**
- **q_to_stage_vision**
- **q_to_stage_decision**
- **stage_camera**
- **stage_vision**
- **stage_dec_and_pub**

3.15.1 Detailed Description

```
pipeline de traitement, fait l'acquisition d'image, le traitement et la publiation

Parameters
----------
to_detects : array of ToDetect
    tableau des objets à détecter
mqtt_client : MQTT.Client
    client MQTT

Attributes
----------
q_to_stage_camera : Queue
    queue se randant à la caméra
q_to_display : Queue
    queue entre la vision et l'affichage
q_to_stage_vision : Queue
    queue entre la caméra et la vision
q_to_stage_decision : Queue
    queue entre la vision et la décision
stage_camera : Process
    process de la caméra
stage_vision : Process
    process de la vision
stage_dec_and_pub : Process
    process de la décision et de la publication
```

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

- pipeline.py

3.16 pipeline.pipelineAndDisplay Class Reference

Public Member Functions

- def **__init__** (self, parent, to_detect, mqtt_client)
- def **refresh_label** (self)
- def **on_closing** (self)

Public Attributes

- **parent**
- **panel**
- **pipeline**
- **queue_display**
- **image**
- **imgtk**

3.16.1 Detailed Description

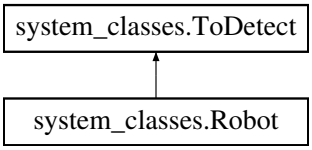
```
class contenant le pipeline et un affichage TKinter

Parameters
----------
parent : TK
    fenêtre parent
to_detect : array of ToDetect
    objets à détecter
mqtt_client : MQTT.Client
    client MQTT

Attributes
----------
on_closing : Func
    fonction de fermeture du système
panel : Label
    panneau pour afficher la sortie de la vision
pipeline : Pipeline
    pipeline de traitement
queue_display : Queue
    queue contenant les images à afficher
refresh_label : Func
    fonction de rafraichiment de l'image
parent
```

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

- pipeline.py

3.17 system_classes.Robot Class Reference

Inheritance diagram for system_classes.Robot:



Public Member Functions

- def **__init__** (self, color, robot_index)

Public Attributes

- **robot_index**

3.17.1 Detailed Description

```
robot à détecter

Parameters
----------
color : Color
    couleur du robot à détecter
robot_index : RobotIndex
    identifiant du robot

Attributes
----------
robot_index
```

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

- system_classes.py

3.18 system_classes.RobotIndex Class Reference

Public Member Functions

- def **__init__** (self, equipe, num)
- def to_string (self)

Static Public Member Functions

- def **init_from_dict** (dict)

Public Attributes

- **equipe**
- **num**

3.18.1 Detailed Description

```
identifiant d'un robot

Parameters
----------
equipe : Equipe
    équipe du robot
num : int
    numéro du robot

Attributes
----------
equipe
num
```

3.18.2 Member Function Documentation

3.18.2.1 to_string()

```
def system_classes.RobotIndex.to_string (
              self )

renvoie l'identifiant en string
```

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

- system_classes.py

3.19 system_classes.RobotInfo Class Reference

Public Member Functions

- def **__init__** (self, position, direction_vec, robot_index)
- def calculate_situation (self, position_balle)
- def **get_distance_balle_robot** (self)
- def get_num (self)
- def get_equ (self)
- def get_name (self)
- def print (self)

Public Attributes

- **vecangle_direction**
- **robot_index**
- **position**
- **possession_balle**
- **vecangle_robot_balle**
- **diff_vecangle**

3.19.1 Detailed Description

```
informations relatives à un robot

Parameters
----------
position : array of float
    position du robot
direction_vec : array of float
    vecteur de direction du robot
robot_index : RobotIndex
    identifiant du robot

Attributes
----------
vecangle_direction : VecAngle
    direction du robot
possession_balle : bool
    si le robot possède la balle
vecangle_robot_balle : VecAngle
    direction entre le robot et la balle
diff_vecangle : VecAngleDiff
    difference entre la diretion du robot et de la balle
robot_index
position
```

3.19.2 Member Function Documentation

3.19.2.1 calculate_situation()

```
def system_classes.RobotInfo.calculate_situation (
              self,
              position_balle )
```

remplie les informations manquante relative à la balle

```
Parameters
----------
position_balle : array of float
    position de la balle
```

3.19.2.2 get_equ()

```
def system_classes.RobotInfo.get_equ (
              self )
```

renvoie léquipe de l'identifiant

3.19.2.3 get_name()

```
def system_classes.RobotInfo.get_name (
              self )
```

renvoie l'identifiant en string

3.19.2.4 get_num()

```
def system_classes.RobotInfo.get_num (
              self )
```

renvoie le numéro de l'identifiant

3.19.2.5 print()

```
def system_classes.RobotInfo.print (
              self )


print les commandes et l'identifiant
```

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

- system_classes.py

3.20 vision.Square Class Reference

Public Member Functions

- def **__init__** (self, top_left, side_length, color, line_width=1)
- def draw (self, img)

Public Attributes

- **color**
- **top_left**
- **bottom_right**
- **line_width**

3.20.1 Detailed Description

```
un carré à dessiner sur une image

Parameters
----------
top_left : array of float
    coin haut gauche du carré
side_length : int
    longueur des coté du carré
color : Color
    couleur du carré à dessiner
line_width : int
    épaisseur de la ligne

Attributes
----------
bottom_right : array of float
    position du coin bas droit du carré
color
top_left
line_width
```

3.20.2 Member Function Documentation

3.20.2.1 draw()

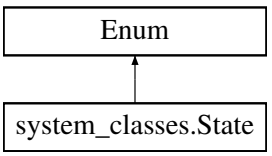
```
def vision.Square.draw (
              self,
              img )

dessine le carré sur img
```

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

- vision.py

3.21 system_classes.State Class Reference

Inheritance diagram for system_classes.State:



Static Public Attributes

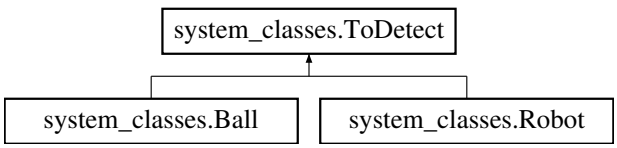
- int **SKYNET_POSSESSION** = 0
- int **HUMANITY_POSSESSION** = 1
- int **NO_ONE_POSSESSION** = 2

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

- system_classes.py

3.22 system_classes.ToDetect Class Reference

Inheritance diagram for system_classes.ToDetect:



Public Member Functions

- def **__init__** (self, color)

Public Attributes

- **color**

3.22.1 Detailed Description

```
objet à détecter par la vision

Parameters
----------
color : Color
    couleur de l'objet à détecter

Attributes
----------
color
```

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

- system_classes.py

3.23 configurator.TunableParam Class Reference

Public Member Functions

- def **__init__** (self, parent, name, default_delta)
- def plus (self)
- def minus (self)
- def print (self)
- def update_config (self)

Public Attributes

- **name**
- **value**
- **entry_delta**

3.23.1 Detailed Description

```
paramètre pouvant être modifié par le configurateur
Parameters
----------
parent : Tk
    conteneur tkinter du paramètre
name : string
    nom du paramètre dans le fichier de configuration
default_delta : float
    variation par défault du paramètre

Attributes
----------
value : float
    valeur du paramètre
plus : func
    fonction d'incrémentation
minus : type
    function de décrémentation
entry_delta : Entry
    champ modifiable de la valeur du delta
name
```

3.23.2 Member Function Documentation

3.23.2.1 minus()

```
def configurator.TunableParam.minus (
              self )
```

```
fonction de décrémentation de la valeur
```

3.23.2.2 plus()

```
def configurator.TunableParam.plus (
              self )
```

```
fonction d'incrémentation de la valeur
```

3.23.2.3 print()

```
def configurator.TunableParam.print (
              self )
```

```
affiche la valeur actuelle du paramètre
```

3.23.2.4 update_config()

```
def configurator.TunableParam.update_config (
              self )
```

```
actualise la configuration dans les configs

Returns
-------
type
    Description of returned object.
```

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

- configurator.py

3.24 vecangle.VecAngle Class Reference

Public Member Functions

- def **__init__** (self, vec)
- def **get_norme** (self)
- def **orth_projection_norme** (self, vecangle)

Public Attributes

- **vec**
- **angle**

3.24.1 Detailed Description

```
Short summary.

Parameters
----------
vec : array of float
    vecteur

Attributes
----------
angle : int
    angle du vecteur
vec
```

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

- vecangle.py

3.25 vecangle.VecAngleDiff Class Reference

Public Member Functions

- def **__init__** (self, a, b)

Public Attributes

- **is_clockwise**

3.25.1 Detailed Description

```
difference entre deux VecAngle

Parameters
----------
a : VecAngle
    vecteur a
b : VecAngle
    vecteur b

Attributes
----------
angle : int
    angle entre les deux vecteurs
is_clockwise : bool
    si le sens de a à b est horaire
```

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

- vecangle.py

3.26 wrap.Warp Class Reference

Public Member Functions

- def **__init__** (self, w, h, k1, k2, p1, p2, crop_top, crop_bottom, crop_left, crop_right, theta)
- def **__call__** (self, img, draw=False)

Static Public Member Functions

- def **init_from_configs** ()

Public Attributes

- **distortion_remover**
- **w**
- **h**
- **crop_top**
- **crop_bottom**
- **crop_left**
- **crop_right**
- **theta**
- **M_rot**

3.26.1 Detailed Description

```
retire la distortion d'une image et ajuste l'angle
```

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

- wrap.py

Index