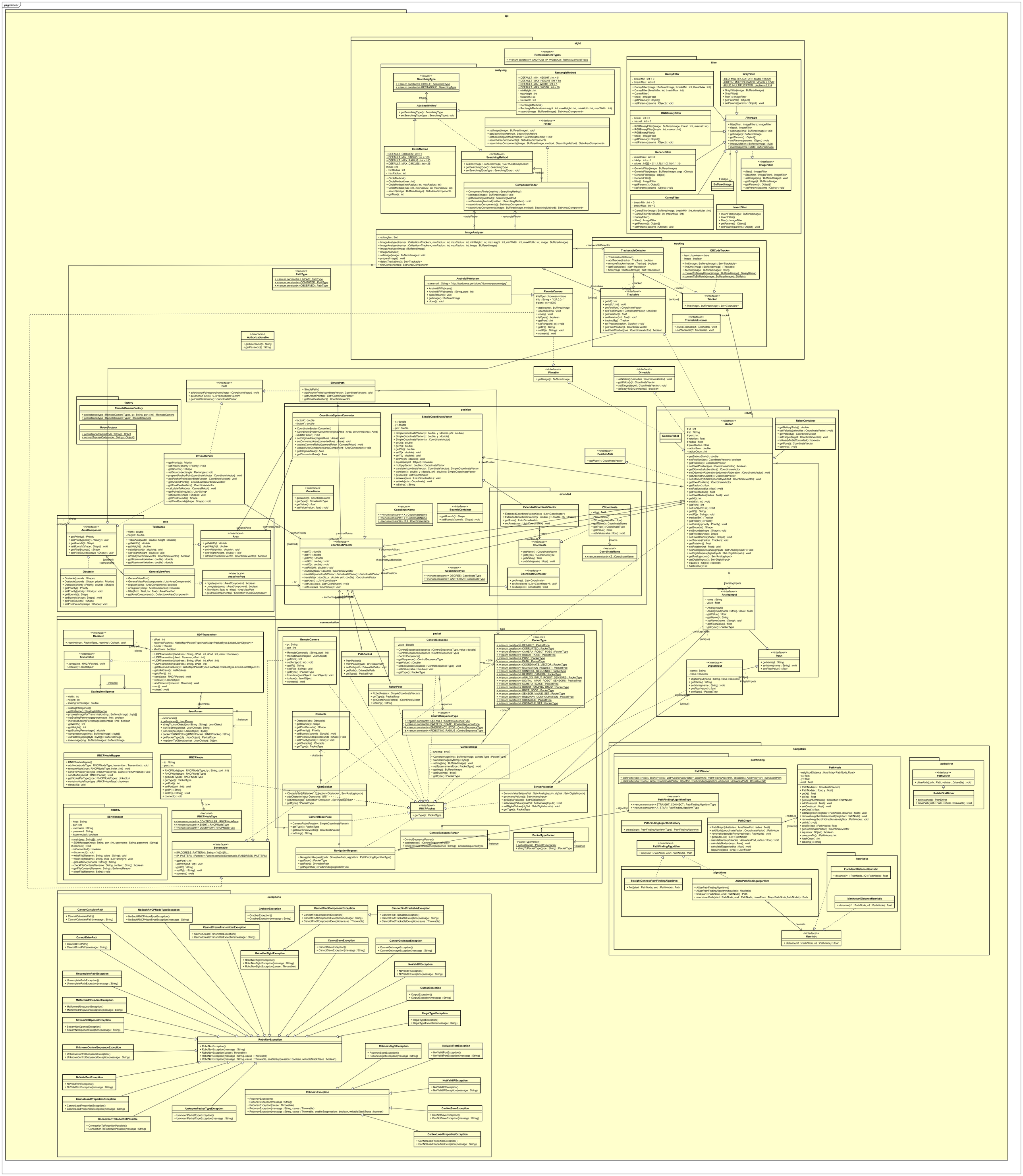
pkg robonav
api
sight
<<enum>> RemoteCameraTypes
+ <<enum constant>> ANDROID_IP_WEBCAM : RemoteCameraTypes
analysing
<<enum>> SearchingType
+ <<enum constant>> CIRCLE : SearchingType
+ <<enum constant>> RECTANGLE : SearchingType
# type
AbstractMethod
+ getSearchingType() : SearchingType
+ setSearchingType(type : SearchingType) : void
RectangleMethod
+ DEFAULT_MIN_HEIGHT : int = 5
+ DEFAULT_MAX_HEIGHT : int = 50
+ DEFAULT_MIN_WIDTH : int = 5
+ DEFAULT_MAX_WIDTH : int = 30
- minHeight : int
- maxHeight : int
- minWidth : int
- maxWidth : int
+ RectangleMethod()
+ RectangleMethod(minHeight : int, maxHeight : int, minWidth : int, maxWidth : int)
+ search(image : BufferedImage) : Set<AreaComponent>
<<interface>> Finder
+ setImage(image : BufferedImage) : void
+ getSearchingMethod() : SearchingMethod
+ setSearchingMethod(method : SearchingMethod) : void
+ searchAreaComponents() : Set<AreaComponent>
+ searchAreaComponents(image : BufferedImage, method : SearchingMethod) : Set<AreaComponent>
CircleMethod
+ DEFAULT_CIRCLES : int = 1
+ DEFAULT_MIN_RADIUS : int = 100
+ DEFAULT_MAX_RADIUS : int = 150
+ DEFAULT_MAX_CIRCLES : int = 25
# max : int
- minRadius : int
- maxRadius : int
+ CircleMethod()
+ CircleMethod(max : int)
+ CircleMethod(minRadius : int, maxRadius : int)
+ CircleMethod(max : int, minRadius : int, maxRadius : int)
+ search(image : BufferedImage) : Set<AreaComponent>
+ getMax() : int
<<interface>> SearchingMethod
+ search(image : BufferedImage) : Set<AreaComponent>
+ getSearchingType() : SearchingType
+ setSearchingType(type : SearchingType) : void
# method
ComponentFinder
+ ComponentFinder(method : SearchingMethod)
+ setImage(image : BufferedImage) : void
+ getSearchingMethod() : SearchingMethod
+ setSearchingMethod(method : SearchingMethod) : void
+ searchAreaComponents() : Set<AreaComponent>
+ searchAreaComponents(image : BufferedImage, method : SearchingMethod) : Set<AreaComponent>
- circleFinder
- rectangleFinder
ImageAnalyser
- rectangles : Set
+ ImageAnalyser(tracker : Collection<Tracker>, minRadius : int, maxRadius : int, minHeight : int, maxHeight : int, minWidth : int, maxWidth : int, image : BufferedImage)
+ ImageAnalyser(tracker : Collection<Tracker>, minRadius : int, maxRadius : int, image : BufferedImage)
+ ImageAnalyser(image : BufferedImage)
+ ImageAnalyser()
+ setImage(image : BufferedImage) : void
+ prepareImage() : void
+ detectTrackables() : Set<Trackable>
+ findComponents() : Set<AreaComponent>
- trackerableDetector
filter
CannyFilter
- threshMin : int = 0
- threshMax : int = 0
+ CannyFilter(image : BufferedImage, threshMin : int, threshMax : int)
+ CannyFilter(threshMin : int, threshMax : int)
+ CannyFilter()
+ filter() : ImageFilter
+ getParams() : Object[]
+ setParams(params : Object) : void
GrayFilter
- RED_MULTIPLICATOR : double = 0.299
- GREEN_MULTIPLICATOR : double = 0.587
- BLUE_MULTIPLICATOR : double = 0.114
+ GrayFilter(image : BufferedImage)
+ GrayFilter()
+ filter() : ImageFilter
+ getParams() : Object[]
+ setParams(params : Object) : void
RGBBinaryFilter
- thresh : int = 0
- maxval : int = 0
+ RGBBinaryFilter(image : BufferedImage, thresh : int, maxval : int)
+ RGBBinaryFilter(thresh : int, maxval : int)
+ RGBBinaryFilter()
+ filter() : ImageFilter
+ getParams() : Object[]
+ setParams(params : Object) : void
Filterpipe
+ filter(filter : ImageFilter) : ImageFilter
+ filter() : ImageFilter
+ setImage(img : BufferedImage) : void
+ getImage() : BufferedImage
+ getParams() : Object[]
+ setParams(params : Object) : void
+ image2Mat(im : BufferedImage) : Mat
+ mat2Image(ma : Mat) : BufferedImage
GenericFilter
- kernelSize : int = 3
- ddepth : int = -1
- values : int[][] = {{-1,-1,1},{-1,-2,1},{-1,-1,1}}
+ GenericFilter(image : BufferedImage)
+ GenericFilter(image : BufferedImage, args : Object)
+ GenericFilter(args : Object)
+ GenericFilter()
+ filter() : ImageFilter
+ getParams() : Object[]
+ setParams(params : Object) : void
# image
BufferedImage
<<interface>> ImageFilter
+ filter() : ImageFilter
+ filter(filter : ImageFilter) : ImageFilter
+ setImage(img : BufferedImage) : void
+ getImage() : BufferedImage
+ getParams() : Object[]
+ setParams(params : Object) : void
CannyFilter
- threshMin : int = 0
- threshMax : int = 0
+ CannyFilter(image : BufferedImage, threshMin : int, threshMax : int)
+ CannyFilter(threshMin : int, threshMax : int)
+ CannyFilter()
+ filter() : ImageFilter
+ getParams() : Object[]
+ setParams(params : Object) : void
InvertFilter
+ InvertFilter(image : BufferedImage)
+ InvertFilter()
+ filter() : ImageFilter
+ getParams() : Object[]
+ setParams(params : Object) : void
tracking
TrackerableDetector
+ TrackerableDetector()
+ addTracker(tracker : Tracker) : boolean
+ removeTracker(tracker : Tracker) : boolean
+ getTrackables() : Set<Trackable>
+ find(image : BufferedImage) : Set<Trackable>
QRCodeTracker
- least : boolean = false
- image : boolean
+ find(image : BufferedImage) : Set<Trackable>
+ findOne(image : BufferedImage) : Trackable
+ decode(image : BufferedImage) : String
+ convertToBinaryBitmap(image : BufferedImage) : BinaryBitmap
+ convertToBitMatrix(image : BufferedImage) : BitMatrix
- tracked
- trackables
- tracker
(unique)
<<interface>> Trackable
+ getId() : int
+ setId(id : int) : void
+ getPosition() : CoordinateVector
+ setPosition(pos : CoordinateVector) : boolean
+ getRotation() : float
+ setRotation(rot : float) : void
+ trackedBy() : Tracker
+ setTracker(tracker : Tracker) : void
+ getPixelPosition() : CoordinateVector
+ setPixelPosition(pos : CoordinateVector) : boolean
<<interface>> Tracker
+ find(image : BufferedImage) : Set<Trackable>
<<interface>> TrackableListener
+ foundTrackable(t : Trackable) : void
+ lostTrackable(t : Trackable) : void
- tracker
<<enum>> PathType
+ <<enum constant>> LINEAR : PathType
+ <<enum constant>> COMPUTED : PathType
+ <<enum constant>> OBSERVED : PathType
AndroidIPWebcam
- streamurl : String = "http://ipaddress:port/video?dummy=param.mjpg"
+ AndroidIPWebcam()
+ AndroidIPWebcam(ip : String, port : int)
+ openStream() : void
+ getImage() : BufferedImage
+ close() : void
RemoteCamera
isOpen : boolean = false
ip : String = "127.0.0.1"
port : int = 8080
+ getImage() : BufferedImage
+ openStream() : void
+ close() : void
+ isOpen() : boolean
+ getPort() : int
+ setPort(port : int) : void
+ getIP() : String
+ setIP(ip : String) : void
+ connect() : void
<<interface>> Authorizationable
+ getUsername() : String
+ getPassword() : String
<<interface>> Filmable
+ getImage() : BufferedImage
<<interface>> Driveable
+ setVelocity(velocities : CoordinateVector) : void
+ getVelocity() : CoordinateVector
+ setTarget(target : CoordinateVector) : void
+ isReadyToBeControlled() : boolean
<<interface>> Path
+ addAnchorPoint(coordinateVector : CoordinateVector) : void
+ getAnchorPoints() : List<CoordinateVector>
+ getFinalDestination() : CoordinateVector
SimplePath
+ SimplePath()
+ addAnchorPoint(coordinateVector : CoordinateVector) : void
+ getAnchorPoints() : List<CoordinateVector>
+ getFinalDestination() : CoordinateVector
factory
RemoteCameraFactory
+ getInstance(type : RemoteCameraTypes, ip : String, port : int) : RemoteCamera
+ getInstance(type : RemoteCameraTypes) : RemoteCamera
RobotFactory
+ getInstance(trackerCode : String) : Robot
+ convertTrackerCode(code : String) : Object[]
DriveablePath
+ getPriority() : Priority
+ setPriority(priority : Priority) : void
+ getBounds() : Shape
+ setBounds(rectangle : Rectangle) : void
+ prependAnchorPoint(coordinateVector : CoordinateVector) : void
+ addAnchorPoint(coordinateVector : CoordinateVector) : void
+ getAnchorPoints() : LinkedList<CoordinateVector>
+ getFinalDestination() : CoordinateVector
+ calculateToRobot(r : CameraRobot) : void
+ getPointsStringList() : List<String>
+ setBounds(shape : Shape) : void
+ getPixelBounds() : Shape
+ setPixelBounds(shape : Shape) : void
position
CoordinateSystemConverter
- factorX : double
- factorY : double
+ CoordinateSystemConverter()
+ CoordinateSystemConverter(originalArea : Area, convertedArea : Area)
- updateFactor() : void
+ setOrginalArea(originalArea : Area) : void
+ setConvertedArea(convertedArea : Area) : void
+ updateCameraRobot(cameraRobot : CameraRobot) : void
+ updateAreaComponent(areaComponent : AreaComponent) : void
+ getOrginalArea() : Area
+ getConvertedArea() : Area
SimpleCoordinateVector
- x : double
- y : double
- phi : double
+ SimpleCoordinateVector(x : double, y : double, phi : double)
+ SimpleCoordinateVector(cv : double, y : double)
+ SimpleCoordinateVector(cv : CoordinateVector)
+ getX() : double
+ getY() : double
+ getPhi() : double
+ setX(x : double) : void
+ setY(y : double) : void
+ setPhi(phi : double) : void
+ equals(object : Object) : boolean
+ multiply(factor : double) : CoordinateVector
+ translate(coordinateVector : CoordinateVector) : SimpleCoordinateVector
+ translate(x : double, y : double, phi : double) : SimpleCoordinateVector
+ getAxes() : List<Coordinate>
+ setAxes(axes : List<Coordinate>) : void
+ setAxes(axes : Coordinate) : void
+ toString() : String
# pixelPosition
<<interface>> PositionAble
+ getPose() : CoordinateVector
<<interface>> Coordinate
+ getName() : CoordinateName
+ getType() : CoordinateType
+ getValue() : float
+ setValue(value : float) : void
<<enum>> CoordinateName
+ <<enum constant>> X : CoordinateName
+ <<enum constant>> Y : CoordinateName
+ <<enum constant>> PHI : CoordinateName
<<interface>> BoundsContainer
+ getBounds() : Shape
+ setBounds(bounds : Shape) : void
extended
ExtendedCoordinateVector
+ ExtendedCoordinateVector(axes : List<Coordinate>)
+ ExtendedCoordinateVector(x : double, y : double, phi : double)
+ getAxes() : List<Coordinate>
+ setAxes(axes : List<Coordinate>) : void
ZCoordinate
- value : float
+ ZCoordinate()
+ ZCoordinate(value : float)
+ getName() : CoordinateName
+ getType() : CoordinateType
+ getValue() : float
+ setValue(value : float) : void
- axes
- anchorPoints
(ordered)
<<interface>> Coordinate
+ getName() : CoordinateName
+ getType() : CoordinateType
+ getValue() : float
+ setValue(value : float) : void
<<enum>> CoordinateName
+ <<enum constant>> Z : CoordinateName
<<interface>> CoordinateContainer
+ getAxes() : List<Coordinate>
+ setAxes(axes : List<Coordinate>) : void
+ setAxes(axes : Coordinate) : void
area
AreaComponent
+ getPriority() : Priority
+ setPriority(priority : Priority) : void
+ getBounds() : Shape
+ setBounds(shape : Shape) : void
+ getPixelBounds() : Shape
+ setPixelBounds(shape : Shape) : void
TableArea
- width : double
- height : double
+ TableArea(width : double, height : double)
+ getWidth() : double
+ getHeight() : double
+ setWidth(width : double) : void
+ setHeight(height : double) : void
+ isValid(coordinateVector : CoordinateVector) : boolean
+ getAbsoluteX(relative : double) : double
+ getAbsoluteY(relative : double) : double
<<interface>> Area
+ getWidth() : double
+ getHeight() : double
+ setWidth(width : double) : void
+ setHeight(height : double) : void
+ isValid(coordinateVector : CoordinateVector) : boolean
- originalArea
- convertedArea
Obstacle
+ Obstacle(bounds : Shape)
+ Obstacle(bounds : Shape, priority : Priority)
+ Obstacle(priority : Priority, bounds : Shape)
+ getPriority() : Priority
+ setPriority(priority : Priority) : void
+ getBounds() : Shape
+ setBounds(shape : Shape) : void
+ getPixelBounds() : Shape
+ setPixelBounds(shape : Shape) : void
GeneralViewPort
+ GeneralViewPort()
+ GeneralViewPort(components : List<AreaComponent>)
+ register(comp : AreaComponent) : boolean
+ unregister(comp : AreaComponent) : boolean
+ filter(from : float, to : float) : AreaViewPort
+ getAreaComponents() : Collection<AreaComponent>
<<interface>> AreaViewPort
+ register(comp : AreaComponent) : boolean
+ unregister(comp : AreaComponent) : boolean
+ filter(from : float, to : float) : AreaViewPort
+ getAreaComponents() : Collection<AreaComponent>
CoordinateVector
+ getX() : double
+ getY() : double
+ getPhi() : double
+ setX(x : double) : void
+ setY(y : double) : void
+ setPhi(phi : double) : void
+ multiply(factor : double) : CoordinateVector
+ translate(coordinateVector : CoordinateVector) : CoordinateVector
+ translate(x : double, y : double, phi : double) : CoordinateVector
+ getAxes() : List<Coordinate>
+ setAxes(axes : List<Coordinate>) : void
+ setAxes(axes : Coordinate) : void
# odometryAtStart
# odometryAbberation
# realPosition
- cv
- anchorPoints
- originalArea
- convertedArea
<<enum>> CoordinateType
+ <<enum constant>> DEGREE : CoordinateType
+ <<enum constant>> CARTESIAN : CoordinateType
robot
CameraRobot
<<abstract>> Robot
id : int
ip : String
port : int
rotation : float
radius : float
pixelRadius : float
- radiusSum : double
- radiusCount : int
+ getBatteryState() : double
+ setPosition(pos : CoordinateVector) : boolean
+ getPosition() : CoordinateVector
+ setPixelPosition(pos : CoordinateVector) : boolean
+ getOdometryAbberation() : CoordinateVector
+ setOdometryAbberation(odometryAbberation : CoordinateVector) : void
+ getOdometryAtStart() : CoordinateVector
+ setOdometryAtStart(odometryAtStart : CoordinateVector) : void
+ getPixelPosition() : CoordinateVector
+ getRadius() : float
+ setRadius(radius : float) : void
+ getPixelRadius() : float
+ setPixelRadius(radius : float) : void
+ getId() : int
+ setId(id : int) : void
+ getPort() : int
+ setPort(port : int) : void
+ getIP() : String
+ setIP(ip : String) : void
+ trackedBy() : Tracker
+ getPriority() : Priority
+ setPriority(priority : Priority) : void
+ getBounds() : Shape
+ setBounds(shape : Shape) : void
+ getPixelBounds() : Shape
+ setPixelBounds(shape : Shape) : void
+ setTracker(tracker : Tracker) : void
+ getRotation() : float
+ setRotation(rot : float) : void
+ setAnalogInputs(analogInputs : Set<AnalogInput>) : void
+ setDigitalInputs(digitalInputs : Set<DigitalInput>) : void
+ getAnalogInputs() : Set<AnalogInput>
+ getDigitalInputs() : Set<DigitalInput>
+ equals(o : Object) : boolean
+ hashCode() : int
RobotContainer
+ getBatteryState() : double
+ setVelocity(velocities : CoordinateVector) : void
+ getVelocity() : CoordinateVector
+ setTarget(target : CoordinateVector) : void
+ isReadyToBeControlled() : boolean
+ getPose() : CoordinateVector
+ connect() : void
# analogInputs
(unique)
<<interface>> AnalogInput
- name : String
- value : float
+ AnalogInput()
+ AnalogInput(name : String, value : float)
+ getValue() : float
+ getName() : String
+ setName(name : String) : void
+ getFloatValue() : float
+ getType() : PacketType
- analogValues
<<interface>> Input
+ getName() : String
+ setName(name : String) : void
+ getFloatValue() : float
<<interface>> DigitalInput
- name : String
- value : boolean
+ DigitalInput(name : String, value : boolean)
+ getName() : String
+ setName(name : String) : void
+ getFloatValue() : float
+ getType() : PacketType
# digitalInputs
- digitalValues
communication
<<interface>> Receiver
+ receive(type : PacketType, received : Object) : void
<<interface>> Transmitter
+ send(data : RNCPPacket) : void
+ receive() : JsonObject
- clients
UDPTransmitter
- dPort : int
- receivedPackets : HashMap<PacketType,HashMap<PacketType,LinkedList<Object>>>
- runner : Thread
- shutdown : boolean
+ UDPTransmitter(dAddress : String, dPort : int, sPort : int, client : Receiver)
+ UDPTransmitter(client : Receiver, sPort : int)
+ UDPTransmitter(dAddress : String, dPort : int, sPort : int)
+ UDPTransmitter(dAddress : String, dPort : int)
+ getReceivedPackets() : HashMap<PacketType,HashMap<PacketType,LinkedList<Object>>>
+ getdAddress() : InetAddress
+ getdPort() : int
+ send(data : RNCPPacket) : void
+ receive() : JsonObject
+ addReceiver(receiver : Receiver) : void
+ run() : void
+ close() : void
- instance
ScalingIntelligence
- width : int
- height : int
- scalingPercentage : double
- ScalingIntelligence()
+ getInstance() : ScalingIntelligence
+ processImageForTransmission(img : BufferedImage) : byte[]
+ setScalingPercentage(percentage : int) : boolean
+ increaseScalingPercentage(percentage : int) : boolean
+ getWidth() : int
+ getHeight() : int
+ getScalingPercentage() : double
+ compressImage(img : BufferedImage) : byte[]
+ extractImage(imgByte : byte[]) : BufferedImage
- scaleImage(img : BufferedImage) : BufferedImage
- jsonParse
JsonParser
- JsonParser()
+ getInstance() : JsonParser
+ stringToJsonObject(jsonString : String) : JsonObject
+ jsonToString(object : JsonObject) : String
+ jsonToByte(object : JsonObject) : byte[]
+ packetToRNCPString(RNCPPacket : RNCPPacket) : String
+ getPacketType(obj : JsonObject) : PacketType
+ mcpJsonToObject(packet : JsonObject) : Object
RNCPNodeMapper
+ RNCPNodeMapper()
+ addNode(nodeType : RNCPNodeType, transmitter : Transmitter) : void
+ removeNode(type : RNCPNodeType, index : int) : void
+ sendPerNodeType(type : RNCPNodeType, packet : RNCPPacket) : void
+ sendToAll(packet : RNCPPacket) : void
+ getNodesPerType(type : RNCPNodeType) : LinkedList
+ containsNodeType(type : RNCPNodeType) : boolean
+ closeAll() : void
RNCPNode
- ip : String
- port : int
+ RNCPNode(type : RNCPNodeType, ip : String, port : int)
+ RNCPNode(type : RNCPNodeType)
+ getNodeType() : RNCPNodeType
+ getType() : PacketType
+ getPort() : int
+ setPort(port : int) : void
+ getIP() : String
+ setIP(ip : String) : void
+ connect() : void
- type
<<enum>> RNCPNodeType
+ <<enum constant>> CONTROLLER : RNCPNodeType
+ <<enum constant>> SIGHT : RNCPNodeType
+ <<enum constant>> OVERVIEW : RNCPNodeType
SSHFile
SSHManager
- host : String
- port : int
- username : String
- password : String
- isconnected : boolean
+ main(arg : String[]) : void
+ SSHManager(host : String, port : int, username : String, password : String)
# connect() : void
+ isConnected() : boolean
+ disconnect() : void
+ writeFile(filename : String, value : String) : void
+ writeFile(filename : String, lines : List<String>) : void
+ getLastLine(filename : String) : String
+ checkFileContent(filename : String, content : String) : boolean
+ getFileContent(filename : String) : BufferedReader
+ clearFile(filename : String) : void
<<interface>> Streamable
+ IPADDRESS_PATTERN : String = "^([0127]\...
+ IP_PATTERN : Pattern = Pattern.compile(Streamable.IPADDRESS_PATTERN)
+ getPort() : int
+ setPort(port : int) : void
+ getIP() : String
+ setIP(ip : String) : void
+ connect() : void
packet
RemoteCamera
- ip : String
- port : int
+ RemoteCamera(ip : String, port : int)
+ RemoteCamera(json : JsonObject)
+ getPort() : int
+ setPort(port : int) : void
+ getIP() : String
+ setIP(ip : String) : void
+ getType() : PacketType
+ getPacket() : DriveablePath
+ toJson() : JsonObject
+ connect() : void
PathPacket
+ PathPacket()
+ PathPacket(path : DriveablePath)
+ setPath(path : DriveablePath) : void
+ getPath() : DriveablePath
+ getType() : PacketType
ControlSequence
- value : Double
+ ControlSequence(sequence : ControlSequenceType, value : double)
+ ControlSequence(sequence : ControlSequenceType)
+ ControlSequence()
+ getSequence() : ControlSequenceType
+ getValue() : Double
+ setSequence(sequence : ControlSequenceType) : void
+ setValue(value : Double) : void
+ getType() : PacketType
- type
<<enum>> PacketType
+ <<enum constant>> DEFAULT : PacketType
+ <<enum constant>> CORRUPTED : PacketType
+ <<enum constant>> CAMERA_ROBOT_POSE : PacketType
+ <<enum constant>> ROBOT_POSE : PacketType
+ <<enum constant>> POSE : PacketType
+ <<enum constant>> PATH : PacketType
+ <<enum constant>> COORDINATE_VECTOR : PacketType
+ <<enum constant>> NAVIGATION_REQUEST : PacketType
+ <<enum constant>> CONTROL_SEQUENCE : PacketType
+ <<enum constant>> REMOTE_CAMERA : PacketType
+ <<enum constant>> ANALOG_INPUT_ROBOT_SENSORS : PacketType
+ <<enum constant>> DIGITAL_INPUT_ROBOT_SENSORS : PacketType
+ <<enum constant>> CAMERA_IMAGE : PacketType
+ <<enum constant>> ROBOT_CAMERA_IMAGE : PacketType
+ <<enum constant>> RNCP_NODE : PacketType
+ <<enum constant>> SENSOR_VALUE_SET : PacketType
+ <<enum constant>> ROBONAV_CONFIGURATION : PacketType
+ <<enum constant>> OBSTACLE : PacketType
+ <<enum constant>> OBSTACLE_SET : PacketType
RobotPose
+ RobotPose(cv : SimpleCoordinateVector)
+ getType() : PacketType
+ getCoordinateVector() : CoordinateVector
+ toString() : String
- sequence
<<enum>> ControlSequenceType
+ <<enum constant>> DEFAULT : ControlSequenceType
+ <<enum constant>> BATTERY_STATE : ControlSequenceType
+ <<enum constant>> EMERGENCY_STOP : ControlSequenceType
+ <<enum constant>> ROBOTINO_RADIUS : ControlSequenceType
Obstacle
+ Obstacle(obs : Obstacle)
+ getBounds() : Shape
+ getPixelBounds() : Shape
+ getPriority() : Priority
+ setBounds(bounds : Double) : void
+ setPixelBounds(pixelBounds : Shape) : void
+ setPriority(priority : Priority) : void
+ getObstacle() : Obstacle
+ getType() : PacketType
- obstacles
CameraImage
- byteImg : byte[]
+ CameraImage(img : BufferedImage, cameraType : PacketType)
+ CameraImage(byteImg : byte[])
+ setImg(img : BufferedImage) : void
+ setType(cameraType : PacketType) : void
+ getImg() : BufferedImage
+ getByteImg() : byte[]
+ getType() : PacketType
ObstacleSet
+ ObstacleSet(obstacles : Collection<Obstacle>, Set<AnalogInput>)
+ addObstacle(obj : Obstacle) : void
+ getObstacles() : Collection<Obstacle>, Set<AnalogInput>
+ getType() : PacketType
SensorValueSet
+ SensorValueSet(anaVal : Set<AnalogInput>, digVal : Set<DigitalInput>)
+ getAnalogValues() : Set<AnalogInput>
+ getDigitalValues() : Set<DigitalInput>
+ setAnalogValues(anaVal : Set<AnalogInput>) : void
+ setDigitalValues(digVal : Set<DigitalInput>) : void
+ getType() : PacketType
<<interface>> RNCPPacket
+ getType() : PacketType
CameraRobotPose
+ CameraRobotPose(cv : SimpleCoordinateVector)
+ getType() : PacketType
+ getCoordinateVector() : CoordinateVector
+ toString() : String
ControlSequenceParser
- ControlSequenceParser()
+ getInstance() : ControlSequenceParser
+ stringToControlSequence(sequence : String) : ControlSequenceType
- instance
PacketTypeParser
- PacketTypeParser()
+ getInstance() : PacketTypeParser
+ stringToPacketType(type : String) : PacketType
- instance
NavigationRequest
+ NavigationRequest(path : DriveablePath, algorithm : PathFindingAlgorithmType)
+ getType() : PacketType
+ getPath() : DriveablePath
+ getAlgorithm() : PathFindingAlgorithmType
navigation
pathfinding
PathPlanner
+ planPath(robot : Robot, anchorPoints : List<CoordinateVector>, algorithm : PathFindingAlgorithm, obstacles : AreaViewPort) : DriveablePath
+ planPath(robot : Robot, target : CoordinateVector, algorithm : PathFindingAlgorithm, obstacles : AreaViewPort) : DriveablePath
<<enum>> PathFindingAlgorithmType
+ <<enum constant>> STRAIGHT_CONNECT : PathFindingAlgorithmType
+ <<enum constant>> A_STAR : PathFindingAlgorithmType
- algorithm
PathFindingAlgorithmFactory
+ create(type : PathFindingAlgorithmType) : PathFindingAlgorithm
<<interface>> PathFindingAlgorithm
+ find(start : PathNode, end : PathNode) : Path
PathGraph
+ PathGraph(obstacles : AreaViewPort, radius : float)
+ addNode(coordinateVector : CoordinateVector) : PathNode
+ removeNode(toBeRemovedNode : PathNode) : void
+ getNodeList() : List<PathNode>
- calculateArea(obstacles : AreaViewPort, radius : float) : void
- calculateNodes(area : Area) : void
- calculateEdges(radius : float) : void
- loopLines(area : Area) : List<Float>
- nodeList
(ordered)
PathNode
- neighborDistance : HashMap<PathNode,Float>
- x : float
- y : float
- cost : float
+ PathNode(cv : CoordinateVector)
+ PathNode(x : float, y : float)
+ getX() : float
+ getY() : float
+ getNeighborNodes() : Collection<PathNode>
+ addCost(cost : float) : void
+ setCost(cost : float) : void
+ getCost() : float
+ addNeighbor(neighbor : PathNode, distance : float) : void
+ removeNeighborBidirectional(neighbor : PathNode) : void
+ removeNeighborUnidirectional(neighbor : PathNode) : void
+ unlink() : void
+ costTo(next : PathNode) : float
+ getCoordinateVector() : CoordinateVector
+ equals(o : Object) : boolean
+ compareTo(o : PathNode) : int
+ hashCode() : int
+ toString() : String
algorithms
StraightConnectPathFindingAlgorithm
+ find(start : PathNode, end : PathNode) : Path
AStarPathFindingAlgorithm
+ AStarPathFindingAlgorithm()
+ AStarPathFindingAlgorithm(heuristic : Heuristic)
+ find(start : PathNode, end : PathNode) : Path
- reconstructPath(start : PathNode, end : PathNode, cameFrom : Map<PathNode,PathNode>) : Path
- heuristic
heuristics
EuclideanDistanceHeuristic
+ distance(n1 : PathNode, n2 : PathNode) : float
ManhattanDistanceHeuristic
+ distance(n1 : PathNode, n2 : PathNode) : float
<<interface>> Heuristic
+ distance(n1 : PathNode, n2 : PathNode) : float
pathdriver
<<interface>> PathDriver
+ drivePath(path : Path, vehicle : Driveable) : void
RotateFirstDriver
+ getInstance() : PathDriver
+ drivePath(path : Path, vehicle : Driveable) : void
exceptions
CannotCalculatePath
+ CannotCalculatePath()
+ CannotCalculatePath(message : String)
NoSuchRNCPNodeTypeException
+ NoSuchRNCPNodeTypeException()
+ NoSuchRNCPNodeTypeException(message : String)
GrabberException
+ GrabberException()
+ GrabberException(message : String)
CannotFindComponentException
+ CannotFindComponentException()
+ CannotFindComponentException(message : String)
+ CannotFindComponentException(cause : Throwable)
CannotFindTrackableException
+ CannotFindTrackableException()
+ CannotFindTrackableException(message : String)
+ CannotFindTrackableException(cause : Throwable)
CannotCreateTransmitterException
+ CannotCreateTransmitterException()
+ CannotCreateTransmitterException(message : String)
CannotDrivePath
+ CannotDrivePath()
+ CannotDrivePath(message : String)
CannotSaveException
+ CannotSaveException()
+ CannotSaveException(message : String)
CannotGetImageException
+ CannotGetImageException()
+ CannotGetImageException(message : String)
RoboNavSightException
+ RoboNavSightException()
+ RoboNavSightException(message : String)
+ RoboNavSightException(cause : Throwable)
NoValidIPException
+ NoValidIPException()
+ NoValidIPException(message : String)
UncompletePathException
+ UncompletePathException()
+ UncompletePathException(message : String)
OutputException
+ OutputException()
+ OutputException(message : String)
MalformedRncpJsonException
+ MalformedRncpJsonException()
+ MalformedRncpJsonException(message : String)
IllegalTypeException
+ IllegalTypeException()
+ IllegalTypeException(message : String)
StreamNotOpenedException
+ StreamNotOpenedException()
+ StreamNotOpenedException(message : String)
RoboNavException
+ RoboNavException()
+ RoboNavException(message : String)
+ RoboNavException(cause : Throwable)
+ RoboNavException(message : String, cause : Throwable)
+ RoboNavException(message : String, cause : Throwable, enableSuppression : boolean, writableStackTrace : boolean)
UnknownControlSequenceException
+ UnknownControlSequenceException()
+ UnknownControlSequenceException(message : String)
RobonavSightException
+ RobonavSightException()
+ RobonavSightException(message : String)
NotValidPortException
+ NotValidPortException()
+ NotValidPortException(message : String)
NoValidPortException
+ NoValidPortException()
+ NoValidPortException(message : String)
NotValidIPException
+ NotValidIPException()
+ NotValidIPException(message : String)
CannotLoadPropertiesException
+ CannotLoadPropertiesException()
+ CannotLoadPropertiesException(message : String)
RobonavException
+ RobonavException()
+ RobonavException(message : String)
+ RobonavException(cause : Throwable)
+ RobonavException(message : String, cause : Throwable)
+ RobonavException(message : String, cause : Throwable, enableSuppression : boolean, writableStackTrace : boolean)
CanNotSaveException
+ CanNotSaveException()
+ CanNotSaveException(message : String)
UnknownPacketTypeException
+ UnknownPacketTypeException()
+ UnknownPacketTypeException(message : String)
ConnectionToRobotNotPossible
+ ConnectionToRobotNotPossible()
+ ConnectionToRobotNotPossible(message : String)
CanNotLoadPropertiesException
+ CanNotLoadPropertiesException()
+ CanNotLoadPropertiesException(message : String)