TrackingSystem::ClassDiagram

package TrackingSystem

Logger
- +static getLogger(): Logger&
- +setLogLever(logLevel)
- +log(logType): ofstream&

ImageProcessingMngmt
- -cameras: std_vector<ImageProcessing>
- +ImageProcessingMngmt(file: sysConfig.yml, intMsg: IntraSystemMessaging)
- -factoryCamera(camProps: CameraProperties)
- +evaluate()

CameraProperties
- +cameraID: int
- +videoSrc: std_string
- +configFile: std_string

ClusterMngmt
- -cameras: std_vector<CameraProperties>
- +ClusterMngmt(file: sysConfig.yml, intMsg: IntraSystemMessaging*)
- +evaluate()

means:
operator>>(...)

cv_videoCapture
- +operator_stream()

ImageProcessing
- -cameraID: int
- +ImageProcessing(camID, src, cam, objDet, intMsg)
- +evaluate()

1..*

-internalCom

IntraSystemMessaging
- +send(msg: IntraSysMsg)
- +recv(out msg: IntraSysMsg): Status

internalCom

IntraDirect
- -msg: IntraSysMsg

Triangulation
- +calculatePosition(messages: vector<IntraSysMsg>, out positions: vector<Pos3D>, out distances: vector<int>)
- -triangulate(ray1: VectRay, ray2: VectRay, out pos: Pos3D): int

ComInterface
- +ComInterface(socketServer: UdpSocketServer)
- +evaluate()
- +sendData(position: Pos3D, errorCode: char)

-externalCom

-capture

-camParams

ImageSource
- +operator_stream()

overload operator>>
to hide color space
conversion and
reset to first frame
for video files

CameraParams
- +globalMaskSet: bool
- -cameraMatrix: cv_Mat
- -distCoeffs: cv_Mat
- -globalMask_Rect: cv_Rect
- -positionVector: cv_Point3i
- -rotationMatrix: cv_Mat
- -globalMask: cv_Mat
- +CameraParams(file: camConfig.yml)
- +undistort(src: PxPos, out dst: PxPos): Status
- +calcObjRay(pos: PxPos, objectRay: VectRay): Status
- +initGlobalMask(frame: cv_Mat)
- +addGlobalMaskToFrame(frame: cv_Mat)
- -parseConfig(configFile: std_string)

-objectDet

«interface»
ObjectDetection
- -referenceFrame: cv_Mat
- +detect(frame: cv_Mat, out positions: PxPosList): Status
- +setReference(frame: cv_Mat)

ObjDetSimple

ObjDetAdvanced

ComBachelorprojekt

ComArttrack

-socket

UdpSocket
- +clientAdress
- +evaluate()
- +receiveMessage(): std_string
- +sendMessage(msg[]: char, lenght: int)

«dataType»
PxPosList

«dataType»
VectRayList
- +deepCopyCtor()

«dataType»
IntraSysMsg
- +camID: int
- +rayList: VectRayList

«dataType»
Pos3D
- +x: int
- +y: int
- +z: int

«dataType»
Status

«dataType»
PxPos
- +x: float
- +y: float

«dataType»
VectRay
- +pos: Pos3D
- +dir: Pos3D

camConfig.yml

-timestamp
-datetime: std_string
+cameraMatrix: cv_Mat
+distCoeffs: cv_Mat
+globalMask: cv_Rect
+positionVector: cv_Point3i
+rotationMatrix: cv_Mat
-viewingCenter: cv_Point3i
-viewingRight: cv_Point3i

legend:
private: used by calibration tools
public: used by tracking system

sysConfig.yml

+objectDetectionUsed: std_string
+cameras[1..*]: CameraProperties
+comInterfaceUsed: std_string
+port: int

1..*
+camera

CameraProperties

+cameraID: int
+videoSrc: std_string
+configFile: std_string

cameraID: for intra system messaging
videoSrc: for cv::videoCapture
configFile: some relative file path

using the openCV FileStorage class to write
xml and yaml
cameraConfig will be converted to yaml,
because the file content is incompatible to
the previous system anyway.