Detection

- confThreshold_ : float
- nmsThreshold_ : float
- inpWidth_ : float
- inpHeight_ : float
- modelWeightsFile_ : string
- modelConfigFile_ : string
- modelClassFile_ : string
- frame_ : cv::Mat
- classes : vector<string>
- detections : vector<cv::Rect>
- confidenceDetection : vector<float>

- drawRedBoundingBox(coordinates: vector<int>, classID: int, conf: float): void
- postProcess(outs : const vector<cv::Mat>): vector<cv::Rect>
- getOutputsNames(net : const cv::dnn::Net) : :vector<String>
+ Detection(): void
+ initializeParams(confThreshold: float, nmsThreshold: float, inpWidth: float, inpHeight: float): void
+ loadModelandLabelClasses(modelWeightsFile: string,modelConfigFile: string, modelClassFile: string) : void
+ getDetections(): vector<cv::Rect>
+ processFrameforHuman(): vector<cv::Rect>
+ getConfidence(): vector<float>
+ setFrame(frame : cv::Mat)
+ ~Detection(): void

Track

- multiTracker: cv::Ptr<cv::MultiTracker>
- frame_ : cv::Mat

- getCoordinatesInCameraFrame(coordinates: vector<float>): vector<float>
+ Track(): void
+ setFrame(frame: cv::Mat): void
+ initializeTracker(): void
+ runTrackerAlgorithm(detections: vector<cv::Rect>): void
+ drawGreenBoundingBox(): cv::Mat
+ resizeBoxes(box: cv::Rect): void
+ updateTracker():void
+ ~Track(): void

DataLoader

- method_ : string
- frame_ : cv::Mat
- path: string
- ouputFile : string

+ DataLoader(): void
+ DataLoader(path: string, method: string): void
+ setInputStreamMethod(method: string) : void
+ getInputStreamMethod(): string
+ setPath(path: string) : void
+ checkParser(cv::CommandLineParser) : int
+ processInput(cv:: CommandLineParser): void
+ ~DataLoader(): void