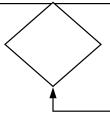
Robot

yolov3: YOLOv3isImage: boolisVideo: bool

imagePath: std::stringvideoPath: std::string

- + Robot(): void
- + setIsImage(isImageValue: bool): void
- + getIsImage(): bool
- + setIsVideo(isVideoValue: bool): void
- + getIsVideo(): bool
- + processImage(): void
- + processVideo(): void
- + checkParser(parser: cv::CommandLineParser): void



Utils

- classes: std::vector<std::string>
- classesFile: std::string
- modelConfiguration: std::string
- modelWeights: std::string
- + Utils(): void
- + getClassesFile(): std::string
- + setClassesFile(classesFileValue: std::string): void
- + getModelConfiguration(): std::string
- + setModelConfiguration(modelConfigurationValue: std::string):
- + getModelWeights(): std::string
- + setModelWeights(modelWeightsValue: std::string): void
- + addClasses(): void
- + getClasses(): std::vector<std::string>
- + drawBoundingBox(classId: int, confidence: double, left: int, top:

int, right: int, bottom: int, frame: cv::Mat): void

YOLOv3

- utils: Utils

confThreshold: floatnmsThreshold: float

inputWidth: intinputHeight: int

+ YOLOv3(): void

+ YOLOv3(confThresholdValue: float, nmsThresholdValue: float, inputWidthValue: int, inputHeightValue: int)

+ setConfThreshold(confThresholdValue: float): void

+ getConfThreshold(): float

+ setNmsThreshold(nmsThresholdValue: float): void

+ getNmsThreshold(): float

+ setInputWidth(inputWidthValue: int): void

+ getInputWidth(): int

+ setInputHeight(inputHeightValue: int): void

+ getInputHeight(): int

+ preprocess(frame: cv::Mat): cv::Mat

+ run(frame: cv::Mat): std::vector<cv::Mat>

+ postprocess(frame: cv::Mat, outputs: std::vector<cv::Mat>): void

+ getOutputLayerNames(net: cv::dnn::Net): std::vector<std::string>

