## UML class diagram

Rohan Singh, Ariun Gupta | October 21, 2019

## VisionModule - confidenceThreshold : float - nmsThreshold : float + applyGaussianFilter(cv::Mat, cv::Size, float) : cv::Mat + applyFilter(cv::Mat, cv::Size) : cv::Mat + applyMedianFilter(cv::Mat, int) : cv::Mat + reshape(cv::Mat, cv::Size) : cv::Mat + nonMaximalSuppression(cv::Mat&, std::vector<cv::Rect>&, std::vector<float>): std::vector<std::vector<int>> DetectionModule - network : Network - tf : Transformation - io : IOHandler - detectedObjects : std::vector<cv::Mat> - inputChoice : int - videoWriter : cv::VideoWriter - videoFrame : cv::VideoCapture - finalDetections : std::vector<std::vector<int>> + DetectionModule(): void + ~DetectionModule(): void + detectObjects(cv::Mat) : int + preProcessImage(cv::Mat, char) : cv::Mat + postProcessImage(cv::Mat, int) : cv::Mat + getInput() : void + getFrame(std::string, int, std::string, int) : int Network - configurationFilePath : std::string - voloNetwork : cv::dnn::Net

## Transformation

baseFrame : cv::MatendFrame : cv::Mat

+ Transformation(): void

+ Transformation(cv::Mat, cv::Mat) : void

+ ~Transformation(): void

+ endToBase(cv::Mat) : cv::Mat

+ baseToEnd(cv::Mat) : cv::Mat + imageToCamera(cv::Mat, cv::Mat) : cv::Mat

+ cameraToImage(cv::Mat, cv::Mat) : cv::Mat

+ getBaseFrame(): cv::Mat

+ getEndFrame(): cv::Mat

- confidenceThreshold : float

- nmsThreshold : float

 $\hbox{- imageWidth}: int\\$ 

- imageHeight : int

- blob : cv::Mat

Camera

- cameralD:int

+ Camera(): void

+ ~Camera(): void

+ isActive(): bool

+ getID(): int

+ displayInfo(): void

- calibrationMatrix:cv::Mat

- resolution:std::pair<int,int>

+ Camera(int, std::pair<int,int>): void

+ getCalibrationMatrix() : cv::Mat

+ getResolution() : std::pair<int, int>

+ Camera(int, std::pair<int.int>, cv::Mat) : void

+ Network(): void

+ ~Network() : void

+ createNetworkInput(cv::Mat) : int

+ applyYOLONetwork(): std::vector<cv::Mat>