Param Parser

Parameters

Human Detection

Method NN

Prepare frame for human tracking Edited Frame → **Classify humans in frame**

Frame →

Position Estimator

Estimate position in camera frame pos in camera frame

Transform

Estimate human pos in robot frame

pos in px →

Vision API

Communicate w/ planner/controller x, y distance ← **Calculate floor distance to human** ← Human (x,y,z) in robot frame

<struct>
Detection
+ x: int
+ y: int
+ width: int

```
                              + height: int
                        ─────────────────────────────────
                        + operator=(const Detection&): Detection&
                        + operator+(const Detection&): Detection
                        + operator-(const Detection&): Detection
                        + operator+=(const Detection&): Detection&
```

<class>
Human Detector

- img_dim: std::array<int, 2>
- detection_classes: vector<cv::String>
- classes: vector<string>
- detection_probability_threshold: double
- nms_threshold: double
- score_threshold: double
- coco_path: const string
- net: cv::dnn::net

- parse_dnn_output(const vector<cv::Mat>&, cv::Mat*, bool):
        shared_ptr<<vector<Detection>>>
- draw_pred(int, float, int, int, int, int, cv::Mat*): void
+ prep_frame(const cv::Mat&): shared_ptr<cv::Mat>
+ detect(const cv::Mat&, bool=false): shared_ptr<vector<Detection>>
+ get_img_dims(): array<int, 2>

<class>
Position Estimator

- avg_human_height: double [m]
- cam_focal_len: double [m]
- cam_pix_density: double [px/m]
- img_center: array<double, 2>
- cam2robot_transform: Eigen::Matrix<double, 4, 4>

- set_values(double, double, double,double, double, double, double, double, double)
- compute_transform_from_xyzp(double, double, double, double)
+ approximate_camera_z(const Detection&): double [m]
+ estimate_xyz(const Detection&): array<double, 3> [m]
+ estimate_all_xyz(const vector<Detection>&): shared_ptr<vector<array<double, 3>>>
+ get_cam2robot_transform(): Eigen::Matrix<double, 4, 4>

<struct>
Var

+ name: string
+ default_unit: string

Vision API

- robot_params: const unordered_map<string, double>&
- detector: HumanDetector- estimator: PositionEstimator
- alert_thresholds: std::array<double, 2> [m]

+ get_xyz(const cv::Mat&): shared_ptr<vector<array<double, 3>>> [m]
+ calculate_distance(array<double, 3> xyz): double
+ print_alerts(const vector<array<double, 3>>): void

<class>
Param Parser

- _var_list: const vector<Var>

- isnot_alnum(char): bool
+ get_unit(const string&): array<string, 2>
+ split_variable(const string&): array<string, 3>
+ set_variable(const array<string, 3>&): double
+ parse_robot_params(string): unordered_map<string, double>