<<External>>
<<Interface>>
OpenCV

<<External>>
<<Interface>>
Sklearn

Detector

- image : Array
- dims : (int, int)

+ load_image(path): void
+ select_roi(frame) : void
+ draw_bounds(frame, rect, c1, c2) : void

<<External>>
<<Interface>>
Numpy

Tracker

- sequence : cv.VideoCapture
- dims : (int, int)
- roi : Array

+ load_sequence(path) : void
+ select_roi(frame) : void
+ draw_bounds(frame, rect, c1, c2) : void

CH Tracker

^- seqeuence : cv.VideoCapture
^- roi : Array
^- dims: (int, int)
^- nc : int
^- nd : int
^- alpha

^- load_sequence(path) : void
^- select_roi(frame) : void
^- draw_bounds(frame, rec, c1)
^- rgb_euclid (pixel) : (float)
^- rgb_quant_m(roi, nc) : (KMeans, Array)
^- rbg_quant_c(kmeans, roi, nc) : Array
^- CH_vert(qlabels) : Array
^- CH_hori(qlabels) : Array
^- CH_pdiag(qlabels) : Array
^- CH_ndiag(qlabels) : Array
^- CH_model(roi) : (KMeans, Array)
^- CH_candidate(kmeans, roi) : Array
^- CH_intersection(ch1, ch2) : int
^- search(kmeans, frame, chm, h, w, a) : void
^- localise(frame, roi, nc, nd, sf) : void
+ track()

CH Detector

^- seqeuence : cv.VideoCapture
^- roi : Array
^- dims: (int, int)
- nc : int
- nd : int
- alpha

^- load_image(path) : void
^- select_roi(frame) : void
^- draw_bounds(frame, rect, c1, c2) : void
- rgb_euclid (pixel) : (float)
- rgb_quant_m(roi, nc) : (KMeans, Array)
- rbg_quant_c(kmeans, roi, nc) : Array
- CH_vert(qlabels) : Array
- CH_hori(qlabels) : Array
- CH_pdiag(qlabels) : Array
- CH_ndiag(qlabels) : Array
- CH_model(roi) : (KMeans, Array)
- CH_candidate(kmeans, roi) : Array
- CH_intersection(ch1, ch2) : int
- search(kmeans, frame, chm, h, w, a) : void
- localise(frame, roi, nc, nd, sf) : void
+ track()

Mean Shift Tracker

^- seqeuence : cv.VideoCapture
^- roi : Array
^- dims: (int, int)
- eps : int

^+ load_sequence(path) : void
^+ select_roi(frame) : (int, int, int, int)
^+ draw_bounds(frame, rect) : void
- get_elliptical_mask(dims) : Array
- get_euclidean(x1, x2) : float
- get_epanech_weight(x) : float
- get_epanech_kernel(dims) : Array
- get_bhatt_coeff(q, p) : float
- get_weights(roi, q, p, m) : Array
- get_msv(roi, weights) : (int, int)
- ms_loop(frame, q, c0, dims) : (int, int)
+track()

Simple Template Matching Tracker

^- seqeuence : cv.VideoCapture
^- roi : Array
^- dims: (int, int)

^+ load_sequence(path) : void
^+ select_roi(frame) : void
^+ draw_bounds(frame, rect, c1, c2) : void
- simple_temp_track(temp) : (int, int)
+ detect_template(temp) : void
+ track()

Adaptive Template Matching Tracker

^- seqeuence : cv.VideoCapture
^- roi : Array
^- dims: (int, int)

^+ load_sequence(path) : void
^+ select_roi(frame) : void
^+ draw_bounds(frame, rect, c1, c2) : void
- adaptive_temp_track(temp) : (int, int)
+ detect_template(temp) : void
+ track()

API

<<External>>
<<Interface>>
PyQt

GUI

- sequence_path : string

- menu() : void
- layout() : void
- API() : void
- run_sequence(path) : void

Use
Extends

Front End

Back End