

pmot

1.0

Generated by Doxygen 1.7.6.1

Thu Mar 12 2015 14:16:53

Contents

1 Class Index	1
1.1 Class Hierarchy	1
2 Class Index	3
2.1 Class List	3
3 File Index	5
3.1 File List	5
4 Class Documentation	7
4.1 BoundingBox Struct Reference	7
4.1.1 Member Data Documentation	7
4.1.1.1 <code>isInit</code>	7
4.1.1.2 <code>x_max</code>	7
4.1.1.3 <code>x_min</code>	7
4.1.1.4 <code>y_max</code>	7
4.1.1.5 <code>y_min</code>	7
4.1.1.6 <code>z_max</code>	7
4.1.1.7 <code>z_min</code>	8
4.2 CsFeature Class Reference	8
4.2.1 Constructor & Destructor Documentation	8
4.2.1.1 <code>CsFeature</code>	8
4.2.1.2 <code>~CsFeature</code>	8
4.2.2 Member Function Documentation	8
4.2.2.1 <code>getDist</code>	8
4.2.2.2 <code>getLongDist</code>	8

4.2.3	Member Data Documentation	8
4.2.3.1	hist	8
4.3	CsFeatureEstimation Class Reference	9
4.3.1	Constructor & Destructor Documentation	10
4.3.1.1	CsFeatureEstimation	10
4.3.1.2	~CsFeatureEstimation	10
4.3.2	Member Function Documentation	10
4.3.2.1	compute	10
4.3.2.2	evalOnePoint	10
4.3.2.3	initBox	10
4.3.3	Member Data Documentation	10
4.3.3.1	m_boxMax	10
4.3.3.2	m_boxMin	10
4.3.3.3	m_gridNr	11
4.3.3.4	m_gridsize	11
4.3.3.5	m_keypoints	11
4.3.3.6	m_keySize	11
4.3.3.7	m_xNr	11
4.3.3.8	m_yNr	11
4.3.3.9	m_zNr	11
4.3.3.10	outF	11
4.4	CSGPU Class Reference	11
4.4.1	Constructor & Destructor Documentation	12
4.4.1.1	CSGPU	12
4.4.1.2	CSGPU	12
4.4.1.3	~CSGPU	12
4.4.2	Member Function Documentation	12
4.4.2.1	compute	12
4.4.3	Member Data Documentation	12
4.4.3.1	aa	12
4.4.3.2	cloudHSV	12
4.4.3.3	cloudPos	12
4.4.3.4	cloudSize	12
4.4.3.5	gridsize	12

4.4.3.6	histsize	12
4.4.3.7	maxPt	12
4.4.3.8	minPt	12
4.4.3.9	partNr	12
4.4.3.10	partpos	12
4.4.3.11	partrot	12
4.4.3.12	refcloud	12
4.4.3.13	refcloudSize	13
4.4.3.14	refhist	13
4.4.3.15	xNr	13
4.4.3.16	yNr	13
4.4.3.17	zNr	13
4.5	PCObject::Edge Struct Reference	13
4.5.1	Member Data Documentation	13
4.5.1.1	u	14
4.5.1.2	v	14
4.5.1.3	weight	14
4.6	PCObject::Edge_spatial Struct Reference	14
4.6.1	Member Data Documentation	14
4.6.1.1	u	14
4.6.1.2	v	14
4.6.1.3	weight_pos	14
4.6.1.4	weight_vel	14
4.7	FalseObjects Struct Reference	14
4.7.1	Member Data Documentation	15
4.7.1.1	childs	15
4.7.1.2	parent	15
4.8	Frame Struct Reference	15
4.8.1	Member Data Documentation	16
4.8.1.1	time	16
4.8.1.2	trackPoints	16
4.9	Track< Object >::Frame Struct Reference	16
4.9.1	Member Data Documentation	16
4.9.1.1	id	16

4.9.1.2	object	16
4.9.1.3	time	17
4.10	Gaus Class Reference	17
4.10.1	Constructor & Destructor Documentation	17
4.10.1.1	Gaus	17
4.10.1.2	~Gaus	17
4.10.2	Member Data Documentation	17
4.10.2.1	cov_inverse	17
4.10.2.2	covariance	17
4.10.2.3	mean	17
4.10.2.4	points	17
4.11	Gaussian Class Reference	18
4.11.1	Constructor & Destructor Documentation	18
4.11.1.1	Gaussian	18
4.11.1.2	Gaussian	18
4.11.2	Member Function Documentation	19
4.11.2.1	init	19
4.11.2.2	initPrediction	19
4.11.2.3	quaternion2rotation	19
4.11.2.4	quaternion2rotation	19
4.11.2.5	updateParam	20
4.11.3	Member Data Documentation	20
4.11.3.1	cov_determinant	20
4.11.3.2	cov_inverse	20
4.11.3.3	covariance	20
4.11.3.4	dim	20
4.11.3.5	eigenvalues	20
4.11.3.6	eigenvectors	20
4.11.3.7	isEmpty	20
4.11.3.8	mean	20
4.11.3.9	nPoint	20
4.11.3.10	predictive_covariance	20
4.11.3.11	predictive_mean	21
4.11.3.12	rotation	21

4.11.3.13	translation	21
4.11.3.14	velocity	21
4.11.3.15	weight	21
4.12	GaussianTrackContainer Class Reference	21
4.12.1	Constructor & Destructor Documentation	22
4.12.1.1	GaussianTrackContainer	22
4.13	GausVcsh Class Reference	22
4.13.1	Member Function Documentation	23
4.13.1.1	generate	23
4.13.1.2	getDist	24
4.13.2	Member Data Documentation	24
4.13.2.1	color	24
4.13.2.2	points	24
4.13.2.3	shape	24
4.14	IMFT< Object > Class Template Reference	24
4.14.1	Member Typedef Documentation	26
4.14.1.1	funcWeight	26
4.14.1.2	ObjectPtr	26
4.14.1.3	Ptr	26
4.14.1.4	TrackContainerT	26
4.14.1.5	TrackT	26
4.14.1.6	V	26
4.14.1.7	VecObjectPtr	26
4.14.1.8	VecPtr	26
4.14.1.9	VecTrack	26
4.14.2	Constructor & Destructor Documentation	26
4.14.2.1	IMFT	26
4.14.2.2	~IMFT	26
4.14.3	Member Function Documentation	26
4.14.3.1	addToDGraph	26
4.14.3.2	confirmDGraph	26
4.14.3.3	deleteLastFrame	27
4.14.3.4	extension	27
4.14.3.5	extractTracks	27

4.14.3.6	getMaximumMatchedObject	27
4.14.3.7	getMaximumMatchedTrack	27
4.14.3.8	getTerminalNodes	27
4.14.3.9	getTerminalNodesLastFrame	27
4.14.3.10	getUnmatchedObjects	27
4.14.3.11	getUnmatchedTracks	27
4.14.3.12	matching	28
4.14.3.13	movingWindow	28
4.14.3.14	setFrame	28
4.14.3.15	twoFrameCorresponding	28
4.14.3.16	updateTracks	29
4.14.4	Member Data Documentation	29
4.14.4.1	cnt	29
4.14.4.2	m_currentT	29
4.14.4.3	m_g	29
4.14.4.4	m_gEdgeMap	29
4.14.4.5	m_gNodeMap	29
4.14.4.6	m_isDebug	29
4.14.4.7	m_maxWeight	29
4.14.4.8	m_newTrackID	29
4.14.4.9	m_vecOldEdge	29
4.14.4.10	m_wsum	29
4.14.4.11	maxID	29
4.14.4.12	trackContainer	29
4.14.4.13	weight	29
4.14.4.14	weight_fast	29
4.14.4.15	window_long	29
4.14.4.16	window_short	30
4.15	Track< Object >::Node Struct Reference	30
4.15.1	Member Data Documentation	30
4.15.1.1	frame	31
4.15.1.2	nodeId	31
4.15.1.3	ptr	31
4.16	ObjectNode Struct Reference	31

4.16.1 Member Data Documentation	31
4.16.1.1 nIn	32
4.16.1.2 nOut	32
4.16.1.3 object	32
4.16.1.4 time	32
4.16.1.5 type	32
4.17 Particlepose Class Reference	32
4.17.1 Constructor & Destructor Documentation	34
4.17.1.1 Particlepose	34
4.17.1.2 ~Particlepose	34
4.17.2 Member Function Documentation	34
4.17.2.1 distfeature	34
4.17.2.2 getPatchScore	34
4.17.2.3 getResult	34
4.17.2.4 initSample	34
4.17.2.5 resample	34
4.17.2.6 sampleWithVar	34
4.17.2.7 setdata	34
4.17.2.8 toCloud	34
4.17.2.9 updateModel	34
4.17.2.10 weight	34
4.17.2.11 weightByDist	35
4.17.2.12 weightGPU	35
4.17.3 Member Data Documentation	35
4.17.3.1 csest	35
4.17.3.2 m_acc	35
4.17.3.3 m_bestspeed	35
4.17.3.4 m_cnt	35
4.17.3.5 m_finalParticle	35
4.17.3.6 m_gen	35
4.17.3.7 m_internalCounter	35
4.17.3.8 m_kdtreeSceneN	35
4.17.3.9 m_lastFinalParticle	35
4.17.3.10 m_lastlastFinalParticle	35

4.17.3.11 m_lastlastlastFinalParticle	35
4.17.3.12 m_lastspeed	35
4.17.3.13 m_model	36
4.17.3.14 m_modelCentroid	36
4.17.3.15 m_modelF	36
4.17.3.16 m_modelN	36
4.17.3.17 m_modelOrig	36
4.17.3.18 m_octreegpu	36
4.17.3.19 m_particlenum	36
4.17.3.20 m_particles	36
4.17.3.21 m_scene	36
4.17.3.22 m_sceneCentroid	36
4.17.3.23 m_sceneN	36
4.17.3.24 m_vel	36
4.17.3.25 maxDist	36
4.17.3.26 minDist	36
4.17.3.27 usegpu	36
4.18 PCObject Class Reference	36
4.18.1 Member Typedef Documentation	39
4.18.1.1 funcWeightGaussian	39
4.18.2 Constructor & Destructor Documentation	39
4.18.2.1 PCObject	39
4.18.2.2 PCObject	39
4.18.2.3 ~PCObject	39
4.18.3 Member Function Documentation	39
4.18.3.1 calculateVelocity	39
4.18.3.2 componentGraph	39
4.18.3.3 evalClosestGMM	39
4.18.3.4 evalClosestNormedGMM	39
4.18.3.5 evalGMM	39
4.18.3.6 evalNormedGMM	39
4.18.3.7 filteringGMM_EM	40
4.18.3.8 filteringGMM_incrementalEM	40
4.18.3.9 gaussianTracking	41

4.18.3.10 gaussianTrackingInit	42
4.18.3.11 getCentroid	42
4.18.3.12 getid	42
4.18.3.13 initGaussian	42
4.18.3.14 initialGMM	43
4.18.3.15 insert	43
4.18.3.16 likelihood	44
4.18.3.17 likelihood_standard	44
4.18.3.18 makeTopology	44
4.18.3.19 mergeTwoGMMs	44
4.18.3.20 setid	44
4.18.3.21 setScale	44
4.18.3.22 setTransParam	44
4.18.3.23 simplify	45
4.18.3.24 toPointCloud	45
4.18.3.25 topology_posweight_rev	46
4.18.3.26 topology_velweight_rev	46
4.18.3.27 topology_weight	46
4.18.3.28 updatePredictiveParameters	46
4.18.4 Member Data Documentation	46
4.18.4.1 alpha	46
4.18.4.2 centroid	46
4.18.4.3 cnt	47
4.18.4.4 diffWeight	47
4.18.4.5 dimension	47
4.18.4.6 edges	47
4.18.4.7 filteredWeight	47
4.18.4.8 frame	47
4.18.4.9 gaussian	47
4.18.4.10 gaussianTracks	47
4.18.4.11 gmm	47
4.18.4.12 id	47
4.18.4.13 imftGaussians	47
4.18.4.14 isParamExist	47

4.18.4.15 percent	47
4.18.4.16 points	47
4.18.4.17 scale	47
4.18.4.18 state	47
4.18.4.19 th_edge	47
4.18.4.20 topology_edgeMap	47
4.18.4.21 topology_graph	47
4.18.4.22 topology_nodeMap	47
4.18.4.23 trans_param	47
4.18.4.24 window_long	47
4.18.4.25 window_short	47
4.19 PCObjectContainer Class Reference	48
4.19.1 Constructor & Destructor Documentation	48
4.19.1.1 PCObjectContainer	48
4.19.1.2 PCObjectContainer	49
4.19.1.3 ~PCObjectContainer	49
4.19.2 Member Function Documentation	49
4.19.2.1 clearAll	49
4.19.2.2 deleteObject	49
4.19.2.3 initGMM	49
4.19.2.4 makeNewObject	50
4.19.2.5 makingObjects	50
4.19.2.6 numObjects	50
4.19.3 Member Data Documentation	50
4.19.3.1 num	50
4.19.3.2 objects	51
4.19.3.3 pCloud	51
4.19.3.4 scale	51
4.20 PCTrackContainer Class Reference	51
4.20.1 Member Typedef Documentation	53
4.20.1.1 Cloud	53
4.20.1.2 CloudConstPtr	53
4.20.1.3 CloudPtr	53
4.20.1.4 PointT	53

4.20.2	Constructor & Destructor Documentation	53
4.20.2.1	PCTrackContainer	53
4.20.2.2	PCTrackContainer	53
4.20.3	Member Function Documentation	53
4.20.3.1	eigenOrdering	53
4.20.3.2	evaluate	53
4.20.3.3	iros2014	54
4.20.3.4	NORM	54
4.20.3.5	oldGaussians	54
4.20.3.6	oldMarkerIDs	54
4.20.3.7	SIGN	54
4.20.3.8	toMarkerEdges	55
4.20.3.9	toMarkerGaussians	55
4.20.3.10	toMarkerGMMs	56
4.20.3.11	toMarkerIDs	56
4.20.3.12	toPointCloudXYZI	57
4.20.4	Member Data Documentation	57
4.20.4.1	isUpdated	57
4.20.4.2	maxFrame	57
4.20.4.3	numFalsePoints	57
4.20.4.4	numTotalPoints	57
4.20.4.5	numTruePoints	57
4.20.4.6	oldGaussiansId	57
4.20.4.7	oldTrackIDs	57
4.21	PCTracking Class Reference	58
4.21.1	Constructor & Destructor Documentation	60
4.21.1.1	PCTracking	60
4.21.1.2	~PCTracking	60
4.21.2	Member Function Documentation	60
4.21.2.1	ambiguityTest	61
4.21.2.2	clearObjects	62
4.21.2.3	confirmDigraph	62
4.21.2.4	getFilteredPC	62
4.21.2.5	getLastModel	63

4.21.2.6	getParticles	63
4.21.2.7	getSegmentedPC	63
4.21.2.8	indexing	64
4.21.2.9	indvLoc	64
4.21.2.10	initdg	65
4.21.2.11	isBoundary	65
4.21.2.12	modifyFalselyMerged	65
4.21.2.13	modifyFalselySeparated	66
4.21.2.14	pointMatching	67
4.21.2.15	run	68
4.21.2.16	segmentation	69
4.21.2.17	segmentationGMM	69
4.21.2.18	trackingGaussians	69
4.21.2.19	typeClear	69
4.21.2.20	typeProblem	70
4.21.2.21	updateTracks	70
4.21.3	Member Data Documentation	70
4.21.3.1	b	70
4.21.3.2	boundary	70
4.21.3.3	boundaryMargin	70
4.21.3.4	clearArcs	71
4.21.3.5	clearNodes	71
4.21.3.6	cnt	71
4.21.3.7	dg_ambiguity	71
4.21.3.8	dg_ambiguity_arcs	71
4.21.3.9	dg_ambiguity_nodes	71
4.21.3.10	dim	71
4.21.3.11	falselyMerged	71
4.21.3.12	falselySeparated	71
4.21.3.13	frame_id	71
4.21.3.14	g	71
4.21.3.15	imft	71
4.21.3.16	imft_loc	71
4.21.3.17	isDebug	71

4.21.3.18 m_partFilters	71
4.21.3.19 maxFrame	71
4.21.3.20 maxID	71
4.21.3.21 maxProbAssociation	71
4.21.3.22 minPoints	71
4.21.3.23 nObjects	71
4.21.3.24 object_tracks	71
4.21.3.25 objects_clear	71
4.21.3.26 objects_falseMerg	71
4.21.3.27 objects_falseSep	71
4.21.3.28 objects_loc	72
4.21.3.29 objects_modified	72
4.21.3.30 objects_prev	72
4.21.3.31 observedPointsList	72
4.21.3.32 pCloud	72
4.21.3.33 pCloud_lastModel	72
4.21.3.34 pCloud_particles	72
4.21.3.35 pCloud_seg	72
4.21.3.36 percent	72
4.21.3.37 problemArcs	72
4.21.3.38 problemNodes	72
4.21.3.39 pub_model	72
4.21.3.40 pub_scene	72
4.21.3.41 r	72
4.21.3.42 scale	72
4.21.3.43 segmentation_maxSize	72
4.21.3.44 segmentation_minSize	72
4.21.3.45 segmentation_tolerance	72
4.21.3.46 thrProbScene	72
4.21.3.47 unmatchedPoints	72
4.22 Point Class Reference	73
4.22.1 Constructor & Destructor Documentation	73
4.22.1.1 Point	73
4.22.1.2 Point	73

4.22.2 Member Function Documentation	73
4.22.2.1 toPclPt	73
4.22.3 Member Data Documentation	73
4.22.3.1 dim	73
4.22.3.2 id	73
4.22.3.3 pos	73
4.22.3.4 rgb	73
4.23 Scene Struct Reference	74
4.23.1 Member Data Documentation	74
4.23.1.1 models	74
4.23.1.2 scene	74
4.23.1.3 trasformedModels	74
4.24 Track< Object > Class Template Reference	75
4.24.1 Constructor & Destructor Documentation	75
4.24.1.1 Track	75
4.24.2 Member Function Documentation	75
4.24.2.1 framePtrFromLast	75
4.24.2.2 frameSize	75
4.24.2.3 getFrameFromLast	75
4.24.2.4 insert	76
4.24.2.5 lastFrame	76
4.24.2.6 lastFramePtr	76
4.24.2.7 terminate	76
4.24.2.8 updateObjectAtFrame	77
4.24.3 Member Data Documentation	77
4.24.3.1 frames	77
4.24.3.2 id	77
4.24.3.3 maxFrame	77
4.25 TrackContainer< Object > Class Template Reference	77
4.25.1 Member Typedef Documentation	78
4.25.1.1 TrackT	78
4.25.1.2 VecTrackPtr	78
4.25.2 Constructor & Destructor Documentation	78
4.25.2.1 TrackContainer	78

4.25.2.2	<code>~TrackContainer</code>	78
4.25.3	Member Function Documentation	78
4.25.3.1	<code>createTrack</code>	79
4.25.3.2	<code>createTrack</code>	79
4.25.3.3	<code>deleteNoUpdatedTracks</code>	80
4.25.3.4	<code>isNewlyUpdated</code>	80
4.25.3.5	<code>merge</code>	80
4.25.3.6	<code>newId</code>	80
4.25.3.7	<code>numTracks</code>	81
4.25.3.8	<code>push_back_track</code>	81
4.25.3.9	<code>updateObject</code>	82
4.25.4	Member Data Documentation	82
4.25.4.1	<code>b</code>	82
4.25.4.2	<code>currentT</code>	82
4.25.4.3	<code>deletedTrackIDs</code>	82
4.25.4.4	<code>g</code>	82
4.25.4.5	<code>maxFrame</code>	82
4.25.4.6	<code>maxID</code>	82
4.25.4.7	<code>oldCnt</code>	82
4.25.4.8	<code>oldCnt_gaussians</code>	82
4.25.4.9	<code>r</code>	82
4.25.4.10	<code>tracks</code>	82
4.26	TrackPoint Struct Reference	83
4.26.1	Member Data Documentation	83
4.26.1.1	<code>pointID</code>	83
4.26.1.2	<code>trackID</code>	83
4.27	Track< Object >::V Struct Reference	83
4.27.1	Member Data Documentation	84
4.27.1.1	<code>edgeID</code>	84
4.27.1.2	<code>id</code>	84
4.27.1.3	<code>isIn</code>	84
4.27.1.4	<code>isTrack</code>	84
4.27.1.5	<code>nFrame</code>	84
4.27.1.6	<code>nTrack</code>	84

4.27.1.7	ptr	84
4.28	Vcsh Class Reference	84
4.28.1	Constructor & Destructor Documentation	85
4.28.1.1	Vcsh	85
4.28.1.2	~Vcsh	85
4.28.2	Member Function Documentation	85
4.28.2.1	addrbg	85
4.28.3	Member Data Documentation	85
4.28.3.1	hist	85
4.29	Weight Struct Reference	85
4.29.1	Member Data Documentation	85
4.29.1.1	kl	85
4.29.1.2	l2	85
4.29.1.3	type	85
5	File Documentation	87
5.1	/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/common.h File Reference	87
5.1.1	Typedef Documentation	88
5.1.1.1	Cloud	88
5.1.1.2	CloudConstPtr	88
5.1.1.3	CloudN	88
5.1.1.4	CloudNPtr	88
5.1.1.5	CloudPtr	88
5.1.1.6	PointNT	88
5.1.1.7	PointT	88
5.1.2	Function Documentation	88
5.1.2.1	colordist	88
5.1.2.2	rgbToHSV	89
5.2	/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.cpp File Reference	90
5.3	/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.h - File Reference	90
5.4	/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csgpu.h File Reference	91

5.5	/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussian.cpp File Reference	92
5.6	/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussian.h File Reference	92
5.6.1	Typedef Documentation	93
5.6.1.1	Cloud	93
5.6.1.2	CloudConstPtr	93
5.6.1.3	CloudPtr	93
5.6.1.4	Points	93
5.6.1.5	PointT	93
5.7	/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussiantrackcontainer.cpp File Reference	93
5.8	/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussiantrackcontainer.h File Reference	93
5.9	/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/imft.h File Reference	94
5.9.1	Define Documentation	95
5.9.1.1	MAXWEIGHT	95
5.10	/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/imft.hpp File Reference	96
5.11	/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/main.cpp File Reference	96
5.11.1	Function Documentation	98
5.11.1.1	cb_rgb	99
5.11.1.2	cuttingRange	99
5.11.1.3	downsampling	100
5.11.1.4	main	100
5.11.1.5	planeExtraction	100
5.11.1.6	readParameters	101
5.11.1.7	transform	101
5.11.1.8	workingspace	101
5.11.2	Variable Documentation	101
5.11.2.1	computationT	102
5.11.2.2	cont_sampling	102
5.11.2.3	cont_simplify	102
5.11.2.4	isdebug	102

5.11.2.5	lastT	102
5.11.2.6	nowT	102
5.11.2.7	param_3d6d	102
5.11.2.8	param_buffernum	102
5.11.2.9	param_filtering_range	102
5.11.2.10	param_frame_id	102
5.11.2.11	param_pub_gmmreg_model	102
5.11.2.12	param_pub_gmmreg_scene	102
5.11.2.13	param_pub_markers_edges	102
5.11.2.14	param_pub_markers_gaussians	102
5.11.2.15	param_pub_markers_gmms	102
5.11.2.16	param_pub_markers_trackID	102
5.11.2.17	param_pub_points_filtered	102
5.11.2.18	param_pub_points_segmented	102
5.11.2.19	param_pub_points_trackID	102
5.11.2.20	param_samplingRatio	102
5.11.2.21	param_segmentTolerance	102
5.11.2.22	param_simplifyRatio	102
5.11.2.23	param_sub_topic	102
5.11.2.24	param_workspace_height	102
5.11.2.25	param_workspace_topic	103
5.11.2.26	param_workspace_width	103
5.11.2.27	param_workspace_x	103
5.11.2.28	param_workspace_y	103
5.11.2.29	param_workspace_z	103
5.11.2.30	param_workspace_zheight	103
5.11.2.31	pCloud	103
5.11.2.32	pCloud_input	103
5.11.2.33	pctracking	103
5.11.2.34	pub_edges	103
5.11.2.35	pub_filtered	103
5.11.2.36	pub_gaussians	103
5.11.2.37	pub_gmms	103
5.11.2.38	pub_lastmodel	103

5.11.2.39 pub_model	103
5.11.2.40 pub_particles	103
5.11.2.41 pub_scene	103
5.11.2.42 pub_segmented	103
5.11.2.43 pub_track	103
5.11.2.44 pub_trackID	103
5.11.2.45 pub_workspace	103
5.11.2.46 sumErrorPoint	103
5.11.2.47 sumTotalPoint	103
5.11.2.48 sumTruePoint	103
5.11.2.49 tf_listener	104
5.11.2.50 workspace	104
5.12 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/particlepose.cpp File Reference	104
5.13 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/particlepose.h File Reference	104
5.14 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pcobject.cpp File Reference	105
5.14.1 Define Documentation	105
5.14.1.1 pi	105
5.14.2 Variable Documentation	106
5.14.2.1 isDebug	106
5.15 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pcobject.h File Reference	106
5.15.1 Define Documentation	107
5.15.1.1 MIN_WEIGHT_RATIO	107
5.15.2 Enumeration Type Documentation	107
5.15.2.1 SIMPLE	107
5.15.2.2 STATE	107
5.16 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pcobjectcontainer.cpp File Reference	107
5.17 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pcobjectcontainer.h File Reference	107
5.18 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctrackcontainer.cpp File Reference	108

5.19 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctrackcontainer.h File Reference	109
5.19.1 Define Documentation	109
5.19.1.1 PI	109
5.20 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking.cpp File Reference	110
5.20.1 Variable Documentation	110
5.20.1.1 isDebug_comp	110
5.21 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking.h File Reference	110
5.21.1 Define Documentation	111
5.21.1.1 FPS_CALC	111
5.21.1.2 L2CUT	112
5.21.1.3 pi	112
5.21.1.4 SQR	112
5.21.2 Typedef Documentation	112
5.21.2.1 VecObject	112
5.21.2.2 VecObjectPtr	112
5.21.3 Enumeration Type Documentation	112
5.21.3.1 Time	112
5.21.3.2 Type	112
5.22 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking_- weight.cpp File Reference	113
5.22.1 Function Documentation	113
5.22.1.1 weight_gaussian_predictive_rev	113
5.22.1.2 weight_l2_gauss	113
5.22.1.3 weight_l2_rev	113
5.22.1.4 weight_loc	113
5.22.1.5 weight_symkl_gauss	114
5.22.1.6 weight_unsymkl_gauss	114
5.23 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking_- weight.h File Reference	115
5.23.1 Define Documentation	115
5.23.1.1 pi	115
5.23.1.2 SQR	115

5.23.2 Function Documentation	116
5.23.2.1 weight_gaussian_predictive_rev	116
5.23.2.2 weight_l2_gauss	116
5.23.2.3 weight_l2_rev	116
5.23.2.4 weight_loc	116
5.23.2.5 weight_symkl_gauss	116
5.23.2.6 weight_unsymkl_gauss	117
5.24 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/track.h File - Reference	117
5.25 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/track.hpp - File Reference	118
5.26 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/trackcontainer.h File Reference	118
5.27 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/trackcontainer.hpp File Reference	119

Chapter 1

Class Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

BoundingBox	7
CsFeature	8
CsFeatureEstimation	9
CSGPU	11
PCObject::Edge	13
PCObject::Edge_spatial	14
FalseObjects	14
Frame	15
Track< Object >::Frame	16
Gaus	17
Gaussian	18
GausVcsh	22
IMFT< Object >	24
Track< Object >::Node	30
ObjectNode	31
Particlepose	32
PCObject	36
PCObjectContainer	48
PCTracking	58
Point	73
Scene	74
Track< Object >	75
TrackContainer< Object >	77
TrackContainer< Gaussian >	77
GaussianTrackContainer	21
TrackContainer< PCObject >	77
PCTrackContainer	51
TrackPoint	83
Track< Object >::V	83

Vcsh	84
Weight	85

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

BoundingBox	7
CsFeature	8
CsFeatureEstimation	9
CSGPU	11
PCObject::Edge	13
PCObject::Edge_spatial	14
FalseObjects	14
Frame	15
Track< Object >::Frame	16
Gaus	17
Gaussian	18
GaussianTrackContainer	21
GausVcsh	22
IMFT< Object >	24
Track< Object >::Node	30
ObjectNode	31
Particlepose	32
PCObject	36
PCObjectContainer	48
PCTrackContainer	51
PCTracking	58
Point	73
Scene	74
Track< Object >	75
TrackContainer< Object >	77
TrackPoint	83
Track< Object >::V	83
Vcsh	84
Weight	85

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/common.h	87
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.cpp	90
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.h	90
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csgpu.h	91
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussian.cpp	92
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussian.h	92
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussiantrackcontainer.- cpp	93
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussiantrackcontainer.- h	93
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/imft.h	94
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/imft.hpp	96
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/main.cpp	96
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/particlepose.cpp . .	104
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/particlepose.h . . .	104
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pcobject.cpp	105
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pcobject.h	106
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pcobjectcontainer.- cpp	107
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pcobjectcontainer.- h	107
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctrackcontainer.- cpp	108
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctrackcontainer.h .	109
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking.cpp . . .	110
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking.h	110
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking_- weight.cpp	113

/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/ pctracking_- weight.h	115
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/ track.h	117
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/ track.hpp	118
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/ trackcontainer.h . .	118
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/ trackcontainer.hpp .	119

Chapter 4

Class Documentation

4.1 BoundingBox Struct Reference

```
#include <pctracking.h>
```

Public Attributes

- double `x_min`
- double `x_max`
- double `y_min`
- double `y_max`
- double `z_min`
- double `z_max`
- bool `isInit`

4.1.1 Member Data Documentation

4.1.1.1 bool `BoundingBox::isInit`

4.1.1.2 double `BoundingBox::x_max`

4.1.1.3 double `BoundingBox::x_min`

4.1.1.4 double `BoundingBox::y_max`

4.1.1.5 double `BoundingBox::y_min`

4.1.1.6 double `BoundingBox::z_max`

4.1.1.7 double BoundingBox::z_min

The documentation for this struct was generated from the following file:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking.h

4.2 CsFeature Class Reference

```
#include <csfeature.h>
```

Public Member Functions

- [CsFeature \(\)](#)
- [~CsFeature \(\)](#)
- float [getDist \(CsFeature other\)](#)
- float [getLongDist \(CsFeature other\)](#)

Public Attributes

- std::vector< float > [hist](#)

4.2.1 Constructor & Destructor Documentation

4.2.1.1 CsFeature::CsFeature ()

4.2.1.2 CsFeature::~CsFeature ()

4.2.2 Member Function Documentation

4.2.2.1 float CsFeature::getDist (CsFeature other)

4.2.2.2 float CsFeature::getLongDist (CsFeature other)

4.2.3 Member Data Documentation

4.2.3.1 std::vector<float> CsFeature::hist

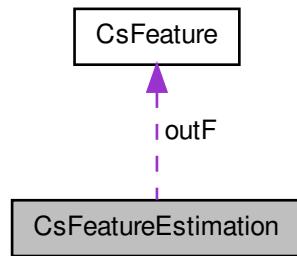
The documentation for this class was generated from the following files:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.h
- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.cpp

4.3 CsFeatureEstimation Class Reference

```
#include <csfeature.h>
```

Collaboration diagram for CsFeatureEstimation:



Public Member Functions

- `CsFeatureEstimation ()`
- `~CsFeatureEstimation ()`
- `void initBox (CloudPtr &cloud)`
- `CsFeature compute (CloudPtr &cloud)`
- `float evalOnePoint (PointT pt)`

Public Attributes

- `Vector4f m_boxMin`
- `Vector4f m_boxMax`
- `CloudPtr m_keypoints`
- `int m_keySize`
- `CsFeature outF`
- `float m_gridsize`
- `int m_xNr`
- `int m_yNr`
- `int m_zNr`
- `int m_gridNr`

4.3.1 Constructor & Destructor Documentation

4.3.1.1 `CsFeatureEstimation::CsFeatureEstimation()`

4.3.1.2 `CsFeatureEstimation::~CsFeatureEstimation()`

4.3.2 Member Function Documentation

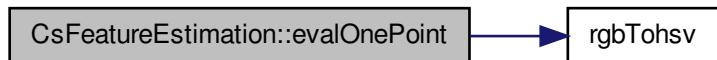
4.3.2.1 `CsFeature CsFeatureEstimation::compute(CloudPtr & cloud)`

Here is the call graph for this function:



4.3.2.2 `float CsFeatureEstimation::evalOnePoint(PointT pt)`

Here is the call graph for this function:



4.3.2.3 `void CsFeatureEstimation::initBox(CloudPtr & cloud)`

4.3.3 Member Data Documentation

4.3.3.1 `Vector4f CsFeatureEstimation::m_boxMax`

4.3.3.2 `Vector4f CsFeatureEstimation::m_boxMin`

- 4.3.3.3 int CsFeatureEstimation::m_gridNr
- 4.3.3.4 float CsFeatureEstimation::m_gridsize
- 4.3.3.5 CloudPtr CsFeatureEstimation::m_keypoints
- 4.3.3.6 int CsFeatureEstimation::m_keySize
- 4.3.3.7 int CsFeatureEstimation::m_xNr
- 4.3.3.8 int CsFeatureEstimation::m_yNr
- 4.3.3.9 int CsFeatureEstimation::m_zNr
- 4.3.3.10 **CsFeature** CsFeatureEstimation::outF

The documentation for this class was generated from the following files:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.h
- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.cpp

4.4 CSGPU Class Reference

```
#include <csgpu.h>
```

Public Member Functions

- **CSGPU** ()
- **CSGPU** (float gridsize_, int xnr_, int ynr_, int znr_, int partnr_, int cloudsize_)
- **~CSGPU** ()
- std::vector< float > **compute** ()

Public Attributes

- int **aa**
- int **histsize**
- size_t **partNr**
- size_t **cloudSize**
- size_t **refcloudSize**
- size_t **xNr**
- size_t **yNr**
- size_t **zNr**
- float3 **minPt**
- float3 **maxPt**
- float **gridsize**

- float * `refhist`
- float3 * `cloudpos`
- float3 * `cloudhsv`
- float3 * `refcloud`
- float3 * `partpos`
- float3 * `partrot`

4.4.1 Constructor & Destructor Documentation

4.4.1.1 `CSGPU::CSGPU() [inline]`

4.4.1.2 `CSGPU::CSGPU(float gridsize_, int xnr_, int ynr_, int znr_, int partnr_, int cloudsize_) [inline]`

4.4.1.3 `CSGPU::~CSGPU() [inline]`

4.4.2 Member Function Documentation

4.4.2.1 `std::vector<float> CSGPU::compute()`

4.4.3 Member Data Documentation

4.4.3.1 `int CSGPU::aa`

4.4.3.2 `float3* CSGPU::cloudhsv`

4.4.3.3 `float3* CSGPU::cloudpos`

4.4.3.4 `size_t CSGPU::cloudSize`

4.4.3.5 `float CSGPU::gridsize`

4.4.3.6 `int CSGPU::histsize`

4.4.3.7 `float3 CSGPU::maxPt`

4.4.3.8 `float3 CSGPU::minPt`

4.4.3.9 `size_t CSGPU::partNr`

4.4.3.10 `float3* CSGPU::partpos`

4.4.3.11 `float3* CSGPU::partrot`

4.4.3.12 `float3* CSGPU::refcloud`

4.4.3.13 `size_t CSGPU::refcloudSize`

4.4.3.14 `float* CSGPU::refhist`

4.4.3.15 `size_t CSGPU::xNr`

4.4.3.16 `size_t CSGPU::yNr`

4.4.3.17 `size_t CSGPU::zNr`

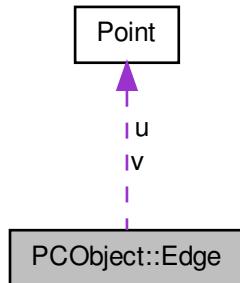
The documentation for this class was generated from the following file:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[csgpu.h](#)

4.5 PCObject::Edge Struct Reference

```
#include <pcobject.h>
```

Collaboration diagram for PCObject::Edge:



Public Attributes

- [Point u](#)
- [Point v](#)
- double [weight](#)

4.5.1 Member Data Documentation

4.5.1.1 Point PCObject::Edge::u

4.5.1.2 Point PCObject::Edge::v

4.5.1.3 double PCObject::Edge::weight

The documentation for this struct was generated from the following file:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[pcobject.h](#)

4.6 PCObject::Edge_spatial Struct Reference

```
#include <pcobject.h>
```

Public Attributes

- ListGraph::Node [u](#)
- ListGraph::Node [v](#)
- double [weight_vel](#)
- double [weight_pos](#)

4.6.1 Member Data Documentation

4.6.1.1 ListGraph::Node PCObject::Edge_spatial::u

4.6.1.2 ListGraph::Node PCObject::Edge_spatial::v

4.6.1.3 double PCObject::Edge_spatial::weight_pos

4.6.1.4 double PCObject::Edge_spatial::weight_vel

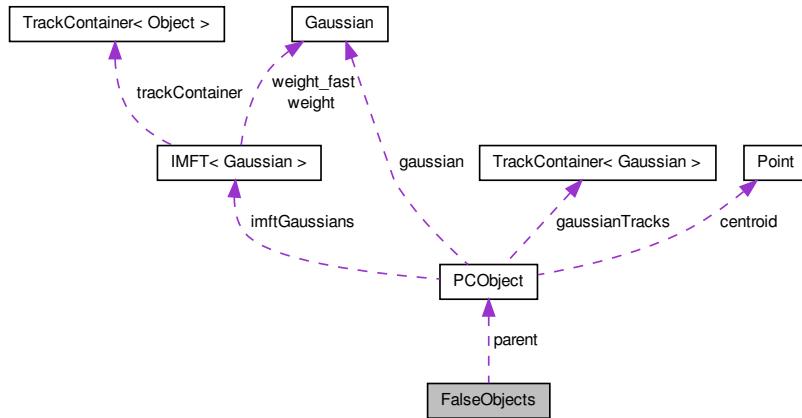
The documentation for this struct was generated from the following file:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[pcobject.h](#)

4.7 FalseObjects Struct Reference

```
#include <pctracking.h>
```

Collaboration diagram for FalseObjects:



Public Attributes

- `PCObject parent`
- `vector< PCObject > childs`

4.7.1 Member Data Documentation

4.7.1.1 `vector<PCObject> FalseObjects::childs`

4.7.1.2 `PCObject FalseObjects::parent`

The documentation for this struct was generated from the following file:

- `/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking.h`

4.8 Frame Struct Reference

```
#include <pctracking.h>
```

Public Attributes

- `vector< TrackPoint > trackPoints`
- `int time`

4.8.1 Member Data Documentation

4.8.1.1 int Frame::time

4.8.1.2 vector<TrackPoint> Frame::trackPoints

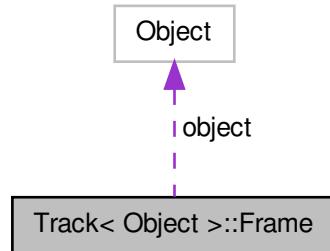
The documentation for this struct was generated from the following file:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[pctracking.h](#)

4.9 Track< Object >::Frame Struct Reference

```
#include <track.h>
```

Collaboration diagram for Track< Object >::Frame:



Public Attributes

- Object [object](#)
- int [id](#)
- int [time](#)

```
template<class Object> struct Track< Object >::Frame
```

4.9.1 Member Data Documentation

4.9.1.1 template<class Object > int Track< Object >::Frame::id

4.9.1.2 template<class Object > Object Track< Object >::Frame::object

4.9.1.3 template<class Object> int Track< Object >::Frame::time

The documentation for this struct was generated from the following file:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[track.h](#)

4.10 Gaus Class Reference

```
#include <csfeature.h>
```

Public Member Functions

- [Gaus \(\)](#)
- [~Gaus \(\)](#)

Public Attributes

- Vector3d [mean](#)
- Matrix3d [covariance](#)
- Matrix3d [cov_inverse](#)
- std::vector< Eigen::Vector3f > [points](#)

4.10.1 Constructor & Destructor Documentation

4.10.1.1 [Gaus::Gaus\(\) \[inline\]](#)

4.10.1.2 [Gaus::~Gaus\(\) \[inline\]](#)

4.10.2 Member Data Documentation

4.10.2.1 [Matrix3d Gaus::cov_inverse](#)

4.10.2.2 [Matrix3d Gaus::covariance](#)

4.10.2.3 [Vector3d Gaus::mean](#)

4.10.2.4 [std::vector<Eigen::Vector3f> Gaus::points](#)

The documentation for this class was generated from the following file:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[csfeature.h](#)

4.11 Gaussian Class Reference

```
#include <gaussian.h>
```

Public Member Functions

- `Gaussian ()`
- `Gaussian (int _dim)`
- `void init (int _dim, Points points)`
- `void updateParam (vnl_vector< double > newParam)`
- `void initPrediction ()`

Static Public Member Functions

- `static void quaternion2rotation (vnl_vector< double > q, vnl_matrix< double > &R, vnl_matrix< double > &g1, vnl_matrix< double > &g2, vnl_matrix< double > &g3, vnl_matrix< double > &g4)`
- `static void quaternion2rotation (vnl_vector< double > q, vnl_matrix< double > &R)`

Public Attributes

- `Eigen::VectorXd mean`
- `Eigen::Vector3d velocity`
- `Eigen::Vector3d eigenvalues`
- `Eigen::Matrix3d eigenvectors`
- `Eigen::MatrixXd covariance`
- `Eigen::MatrixXd cov_inverse`
- `double cov_determinant`
- `Eigen::VectorXd predictive_mean`
- `Eigen::MatrixXd predictive_covariance`
- `double weight`
- `int nPoint`
- `bool isEmpty`
- `int dim`
- `vnl_matrix< double > translation`
- `vnl_matrix< double > rotation`

4.11.1 Constructor & Destructor Documentation

4.11.1.1 Gaussian::Gaussian()

4.11.1.2 Gaussian::Gaussian(int _dim)

4.11.2 Member Function Documentation

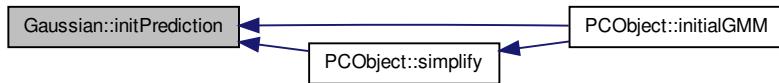
4.11.2.1 void Gaussian::init (int *dim*, Points *points*)

Here is the caller graph for this function:



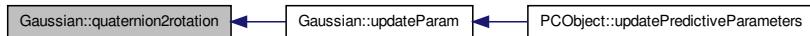
4.11.2.2 void Gaussian::initPrediction ()

Here is the caller graph for this function:



4.11.2.3 void Gaussian::quaternion2rotation (vnl_vector< double > *q*, vnl_matrix< double > & *R*, vnl_matrix< double > & *g1*, vnl_matrix< double > & *g2*, vnl_matrix< double > & *g3*, vnl_matrix< double > & *g4*) [static]

Here is the caller graph for this function:



4.11.2.4 void Gaussian::quaternion2rotation (vnl_vector< double > *q*, vnl_matrix< double > & *R*) [static]

4.11.2.5 `void Gaussian::updateParam (vnl_vector< double > newParam)`

Here is the call graph for this function:



Here is the caller graph for this function:



4.11.3 Member Data Documentation

4.11.3.1 `double Gaussian::cov_determinant`

4.11.3.2 `Eigen::MatrixXd Gaussian::cov_inverse`

4.11.3.3 `Eigen::MatrixXd Gaussian::covariance`

4.11.3.4 `int Gaussian::dim`

4.11.3.5 `Eigen::Vector3d Gaussian::eigenvalues`

4.11.3.6 `Eigen::Matrix3d Gaussian::eigenvectors`

4.11.3.7 `bool Gaussian::isEmpty`

4.11.3.8 `Eigen::VectorXd Gaussian::mean`

4.11.3.9 `int Gaussian::nPoint`

4.11.3.10 `Eigen::MatrixXd Gaussian::predictive_covariance`

4.11.3.11 Eigen::VectorXd Gaussian::predictive_mean

4.11.3.12 vnl_matrix<double> Gaussian::rotation

4.11.3.13 vnl_matrix<double> Gaussian::translation

4.11.3.14 Eigen::Vector3d Gaussian::velocity

4.11.3.15 double Gaussian::weight

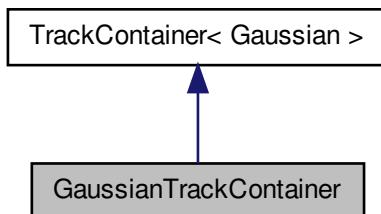
The documentation for this class was generated from the following files:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[gaussian.h](#)
- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[gaussian.cpp](#)

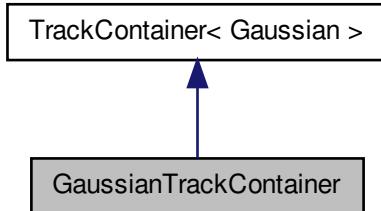
4.12 GaussianTrackContainer Class Reference

```
#include <gaussiantrackcontainer.h>
```

Inheritance diagram for GaussianTrackContainer:



Collaboration diagram for GaussianTrackContainer:



Public Member Functions

- [GaussianTrackContainer \(int _maxFrame=1000\)](#)

4.12.1 Constructor & Destructor Documentation

4.12.1.1 GaussianTrackContainer::GaussianTrackContainer (int _maxFrame = 1000)

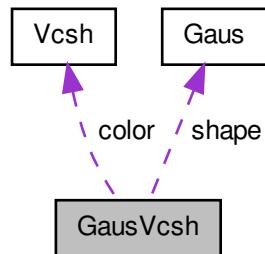
The documentation for this class was generated from the following files:

- [/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussiantrackcontainer.h](#)
- [/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussiantrackcontainer.cpp](#)

4.13 GausVcsh Class Reference

```
#include <csfeature.h>
```

Collaboration diagram for GausVcsh:



Public Member Functions

- void `generate` (float *gridsize*)
- float `getDist` (`GausVcsh` *other*)

Public Attributes

- `Gaus shape`
- `Vcsh color`
- `std::vector< PointT > points`

4.13.1 Member Function Documentation

4.13.1.1 void `GausVcsh::generate` (float *gridsize*)

Here is the call graph for this function:



4.13.1.2 float GausVcsh::getDist (GausVcsh other)

4.13.2 Member Data Documentation

4.13.2.1 Vcsh GausVcsh::color

4.13.2.2 std::vector<PointT> GausVcsh::points

4.13.2.3 Gaus GausVcsh::shape

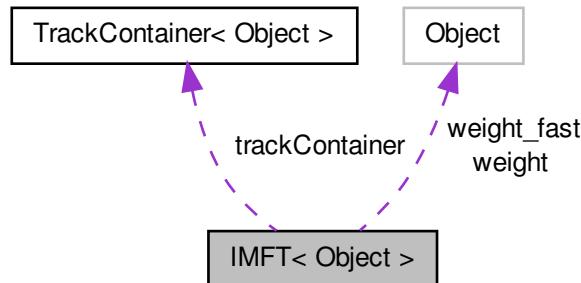
The documentation for this class was generated from the following files:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.h
- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.cpp

4.14 IMFT< Object > Class Template Reference

#include <imft.h>

Collaboration diagram for IMFT< Object >:



Public Types

- typedef TrackContainer< Object > TrackContainerT
- typedef Track< Object > TrackT
- typedef TrackT::Frame Ptr
- typedef vector< Ptr > VecPtr
- typedef TrackT::V V
- typedef vector< TrackT > VecTrack

- `typedef double(* funcWeight)(Object &o1, Object &o2)`
- `typedef Object * ObjectPtr`
- `typedef vector< Object * > VecObjectPtr`

Public Member Functions

- `IMFT (int _window_short=10, int _window_long=20, int _maxID=100, funcWeight _weight=0, funcWeight _weight_fast=0)`
- `~IMFT ()`
- `void setFrame (vector< Object * > objects, int stamp)`
- `void confirmDGraph ()`
- `void extension ()`
- `void matching ()`
- `void updateTracks ()`
- `TrackContainerT * extractTracks ()`
- `VecObjectPtr getUnmatchedObjects ()`
- `VecObjectPtr getTerminalNodes ()`
- `VecObjectPtr getTerminalNodesLastFrame ()`
- `VecObjectPtr getUnmatchedTracks ()`
- `void getMaximumMatchedTrack (ObjectPtr object, ObjectPtr &maxTrack, double &wHypothesis, ObjectPtr &objectOrigin, double &wOrigin)`
- `void getMaximumMatchedObject (ObjectPtr trackUnmatched, ObjectPtr &maxObject, double &wHypothesis, ObjectPtr &trackOrigin, double &wOrigin)`
- `bool deleteLastFrame ()`

Public Attributes

- `funcWeight weight`
- `funcWeight weight_fast`
- `TrackContainerT * trackContainer`
- `int cnt`
- `int window_short`
- `int window_long`
- `int maxID`
- `bool m_isDebug`
- `ListGraph m_g`
- `ListGraph::NodeMap< V > * m_gNodeMap`
- `ListGraph::EdgeMap< double > * m_gEdgeMap`
- `double m_maxWeight`
- `vector< int > m_vecOldEdge`
- `double m_wsum`
- `int m_currentT`
- `int m_newTrackID`

Private Member Functions

- void `movingWindow ()`
- void `addToDGraph (VecPtr ptrs)`
- void `twoFrameCorresponding (vector< ListGraph::Node > vecUFrame, vector< ListGraph::Node > vecVFrame)`

```
template<class Object> class IMFT< Object >
```

4.14.1 Member Typedef Documentation

4.14.1.1 template<class Object> `typedef double(* IMFT< Object >::funcWeight)(Object &o1, Object &o2)`

4.14.1.2 template<class Object> `typedef Object* IMFT< Object >::ObjectPtr`

4.14.1.3 template<class Object> `typedef TrackT::Frame IMFT< Object >::Ptr`

4.14.1.4 template<class Object> `typedef TrackContainer<Object> IMFT< Object >::TrackContainerT`

4.14.1.5 template<class Object> `typedef Track<Object> IMFT< Object >::TrackT`

4.14.1.6 template<class Object> `typedef TrackT::V IMFT< Object >::V`

4.14.1.7 template<class Object> `typedef vector<Object*> IMFT< Object >::VecObjectPtr`

4.14.1.8 template<class Object> `typedef vector<Ptr> IMFT< Object >::VecPtr`

4.14.1.9 template<class Object> `typedef vector<TrackT> IMFT< Object >::VecTrack`

4.14.2 Constructor & Destructor Documentation

4.14.2.1 template<class Object > `IMFT< Object >::IMFT (int _window_short = 10, int _window_long = 20, int _maxID = 100, funcWeight _weight = 0, funcWeight _weight_fast = 0)`

4.14.2.2 template<class Object > `IMFT< Object >::~IMFT ()`

4.14.3 Member Function Documentation

4.14.3.1 template<class Object > `void IMFT< Object >::addToDGraph (VecPtr ptrs) [private]`

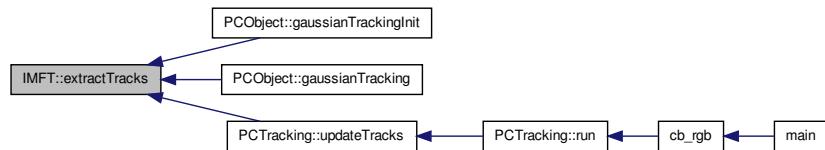
4.14.3.2 template<class Object > `void IMFT< Object >::confirmDGraph ()`

4.14.3.3 template<class Object> bool IMFT< Object >::deleteLastFrame()

4.14.3.4 template<class Object> void IMFT< Object >::extension()

4.14.3.5 template<class Object> IMFT< Object >::TrackContainerT * IMFT< Object >::extractTracks()

Here is the caller graph for this function:



4.14.3.6 template<class Object> void IMFT< Object >::getMaximumMatchedObject(ObjectPtr trackUnmatched, ObjectPtr & maxObject, double & wHypothesis, ObjectPtr & trackOrigin, double & wOrigin)

4.14.3.7 template<class Object> void IMFT< Object >::getMaximumMatchedTrack(ObjectPtr object, ObjectPtr & maxTrack, double & wHypothesis, ObjectPtr & objectOrigin, double & wOrigin)

4.14.3.8 template<class Object> IMFT< Object >::VecObjectPtr IMFT< Object >::getTerminalNodes()

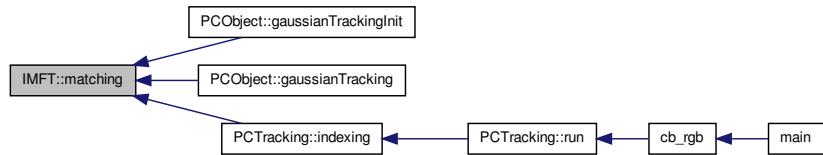
4.14.3.9 template<class Object> IMFT< Object >::VecObjectPtr IMFT< Object >::getTerminalNodesLastFrame()

4.14.3.10 template<class Object> IMFT< Object >::VecObjectPtr IMFT< Object >::getUnmatchedObjects()

4.14.3.11 template<class Object> IMFT< Object >::VecObjectPtr IMFT< Object >::getUnmatchedTracks()

4.14.3.12 template<class Object > void IMFT< Object >::matching ()

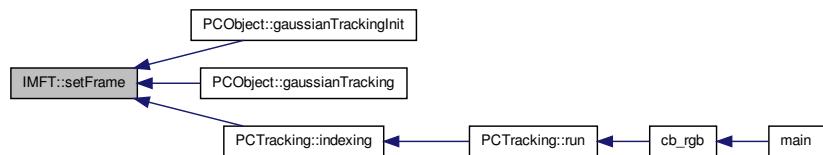
Here is the caller graph for this function:



**4.14.3.13 template<class Object > void IMFT< Object >::movingWindow ()
[private]**

**4.14.3.14 template<class Object> void IMFT< Object >::setFrame (vector< Object * >
objects, int stamp)**

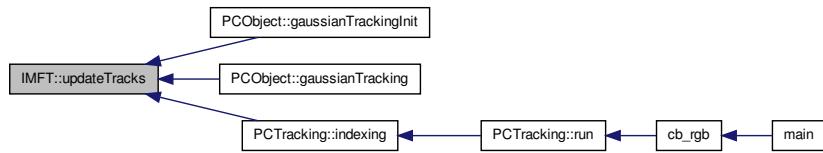
Here is the caller graph for this function:



**4.14.3.15 template<typename Object > void IMFT< Object >::twoFrameCorresponding
(vector< ListGraph::Node > vecUFrame, vector< ListGraph::Node > vecVFrame)
[private]**

4.14.3.16 template<class Object> void IMFT< Object >::updateTracks()

Here is the caller graph for this function:



4.14.4 Member Data Documentation

4.14.4.1 template<class Object> int IMFT< Object >::cnt

4.14.4.2 template<class Object> int IMFT< Object >::m_currentT

4.14.4.3 template<class Object> ListGraph IMFT< Object >::m_g

4.14.4.4 template<class Object> ListGraph::EdgeMap<double>* IMFT< Object >::m_gEdgeMap

4.14.4.5 template<class Object> ListGraph::NodeMap<V>* IMFT< Object >::m_gNodeMap

4.14.4.6 template<class Object> bool IMFT< Object >::m_isDebug

4.14.4.7 template<class Object> double IMFT< Object >::m_maxWeight

4.14.4.8 template<class Object> int IMFT< Object >::m_newTrackID

4.14.4.9 template<class Object> vector<int> IMFT< Object >::m_vecOldEdge

4.14.4.10 template<class Object> double IMFT< Object >::m_wsum

4.14.4.11 template<class Object> int IMFT< Object >::maxID

4.14.4.12 template<class Object> TrackContainerT* IMFT< Object >::trackContainer

4.14.4.13 template<class Object> funcWeight IMFT< Object >::weight

4.14.4.14 template<class Object> funcWeight IMFT< Object >::weight_fast

4.14.4.15 template<class Object> int IMFT< Object >::window_long

4.14.4.16 template<class Object> int IMFT< Object >::window_short

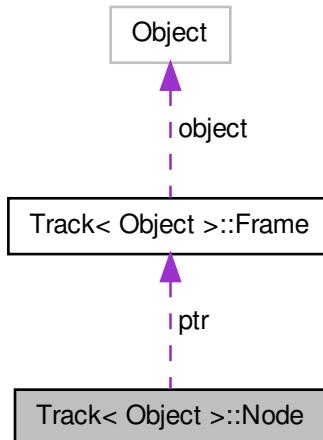
The documentation for this class was generated from the following files:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/imft.h
- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/imft.hpp

4.15 Track< Object >::Node Struct Reference

```
#include <track.h>
```

Collaboration diagram for Track< Object >::Node:



Public Attributes

- [Frame ptr](#)
- int [frame](#)
- int [nodeId](#)

```
template<class Object> struct Track< Object >::Node
```

4.15.1 Member Data Documentation

4.15.1.1 template<class Object> int Track< Object >::Node::frame

4.15.1.2 template<class Object> int Track< Object >::Node::nodeId

4.15.1.3 template<class Object> Frame Track< Object >::Node::ptr

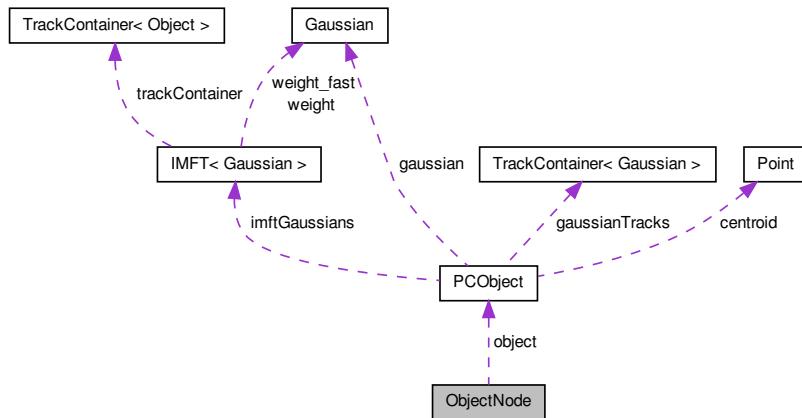
The documentation for this struct was generated from the following file:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[track.h](#)

4.16 ObjectNode Struct Reference

```
#include <pctracking.h>
```

Collaboration diagram for ObjectNode:



Public Attributes

- `Type type`
- `Time time`
- `PCObject object`
- `int nOut`
- `int nIn`

4.16.1 Member Data Documentation

- 4.16.1.1 int ObjectNode::nIn
- 4.16.1.2 int ObjectNode::nOut
- 4.16.1.3 PCObject ObjectNode::object
- 4.16.1.4 Time ObjectNode::time
- 4.16.1.5 Type ObjectNode::type

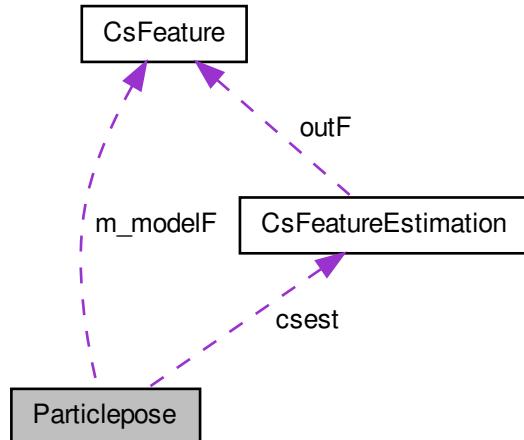
The documentation for this struct was generated from the following file:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking.h

4.17 Particlepose Class Reference

```
#include <particlepose.h>
```

Collaboration diagram for Particlepose:



Public Member Functions

- [Particlepose \(\)](#)
- [~Particlepose \(\)](#)
- void [setdata \(CloudPtr scene, CloudPtr model, int cnt\)](#)

- `CloudPtr getResult ()`
- `void initSample ()`
- `void resample (float varDist, float varAng)`
- `void weight ()`
- `CloudPtr toCloud ()`
- `void updateModel (CloudPtr cloud)`
- `std::vector< float > weightGPU (float gridsize, int xnr, int ynr, int znr, Vector4f min, Vector4f max, std::vector< float > refhist)`
- `float distfeature (VFHSignature308 f1, VFHSignature308 f2)`
- `void weightByDist ()`
- `std::vector< float > getPatchScore (CloudPtr cloud)`

Public Attributes

- `CsFeature m_modelF`
- `CsFeatureEstimation csest`

Private Member Functions

- `tracking::ParticleXYZRPY sampleWithVar (tracking::ParticleXYZRPY part, float varDist, float varAng)`

Private Attributes

- `boost::mt19937 m_gen`
- `tracking::ParticleXYZRPY m_finalParticle`
- `tracking::ParticleXYZRPY m_lastFinalParticle`
- `tracking::ParticleXYZRPY m_lastlastFinalParticle`
- `tracking::ParticleXYZRPY m_lastlastlastFinalParticle`
- `tracking::ParticleXYZRPY m_vel`
- `tracking::ParticleXYZRPY m_acc`
- `Eigen::Vector4f m_sceneCentroid`
- `Eigen::Vector4f m_modelCentroid`
- `pcl::gpu::Octree m_octreegpu`
- `CloudPtr m_scene`
- `CloudPtr m_model`
- `CloudPtr m_modelOrig`
- `CloudNPtr m_sceneN`
- `CloudNPtr m_modelN`
- `int m_particlenum`
- `std::vector < tracking::ParticleXYZRPY > m_particles`
- `std::vector < tracking::ParticleXYZRPY > m_lastspeed`
- `std::vector < tracking::ParticleXYZRPY > m_bestspeed`
- `pcl::KdTreeFLANN< PointNT >::Ptr m_kdtreeSceneN`
- `int m_cnt`

- int `m_internalCounter`
- float `minDist`
- float `maxDist`
- bool `usegpu`

4.17.1 Constructor & Destructor Documentation

4.17.1.1 `Particlepose::Particlepose()`

4.17.1.2 `Particlepose::~Particlepose()`

4.17.2 Member Function Documentation

4.17.2.1 `float Particlepose::distfeature (VFHSignature308 f1, VFHSignature308 f2)`

4.17.2.2 `std::vector< float > Particlepose::getPatchScore (CloudPtr cloud)`

4.17.2.3 `CloudPtr Particlepose::getResult ()`

4.17.2.4 `void Particlepose::initSample ()`

4.17.2.5 `void Particlepose::resample (float varDist, float varAng)`

4.17.2.6 `tracking::ParticleXYZRPY Particlepose::sampleWithVar (tracking::ParticleXYZRPY part, float varDist, float varAng) [private]`

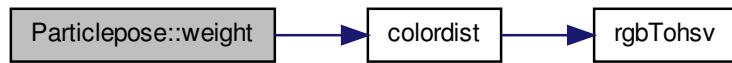
4.17.2.7 `void Particlepose::setdata (CloudPtr scene, CloudPtr model, int cnt)`

4.17.2.8 `CloudPtr Particlepose::toCloud ()`

4.17.2.9 `void Particlepose::updateModel (CloudPtr cloud)`

4.17.2.10 `void Particlepose::weight ()`

Here is the call graph for this function:



- 4.17.2.11 void Particlepose::weightByDist() [inline]
- 4.17.2.12 std::vector< float > Particlepose::weightGPU (float *gridsize*, int *xnr*, int *ynr*, int *znr*, Vector4f *min*, Vector4f *max*, std::vector< float > *refhist*)

Here is the call graph for this function:



4.17.3 Member Data Documentation

- 4.17.3.1 CsFeatureEstimation Particlepose::csest
- 4.17.3.2 tracking::ParticleXYZRPY Particlepose::m_acc [private]
- 4.17.3.3 std::vector<tracking::ParticleXYZRPY> Particlepose::m_bestspeed [private]
- 4.17.3.4 int Particlepose::m_cnt [private]
- 4.17.3.5 tracking::ParticleXYZRPY Particlepose::m_finalParticle [private]
- 4.17.3.6 boost::mt19937 Particlepose::m_gen [private]
- 4.17.3.7 int Particlepose::m_internalCounter [private]
- 4.17.3.8 pcl::KdTreeFLANN<PointNT>::Ptr Particlepose::m_kdtreeSceneN [private]
- 4.17.3.9 tracking::ParticleXYZRPY Particlepose::m_lastFinalParticle [private]
- 4.17.3.10 tracking::ParticleXYZRPY Particlepose::m_lastlastFinalParticle [private]
- 4.17.3.11 tracking::ParticleXYZRPY Particlepose::m_lastlastlastFinalParticle [private]
- 4.17.3.12 std::vector<tracking::ParticleXYZRPY> Particlepose::m_lastspeed [private]

- 4.17.3.13 **CloudPtr Particlepose::m_model** [private]
- 4.17.3.14 **Eigen::Vector4f Particlepose::m_modelCentroid** [private]
- 4.17.3.15 **CsFeature Particlepose::m_modelF**
- 4.17.3.16 **CloudNPtr Particlepose::m_modelN** [private]
- 4.17.3.17 **CloudPtr Particlepose::m_modelOrig** [private]
- 4.17.3.18 **pcl::gpu::Octree Particlepose::m_octreegpu** [private]
- 4.17.3.19 **int Particlepose::m_particlenum** [private]
- 4.17.3.20 **std::vector<tracking::ParticleXYZRPY> Particlepose::m_particles** [private]
- 4.17.3.21 **CloudPtr Particlepose::m_scene** [private]
- 4.17.3.22 **Eigen::Vector4f Particlepose::m_sceneCentroid** [private]
- 4.17.3.23 **CloudNPtr Particlepose::m_sceneN** [private]
- 4.17.3.24 **tracking::ParticleXYZRPY Particlepose::m_vel** [private]
- 4.17.3.25 **float Particlepose::maxDist** [private]
- 4.17.3.26 **float Particlepose::minDist** [private]
- 4.17.3.27 **bool Particlepose::usegpu** [private]

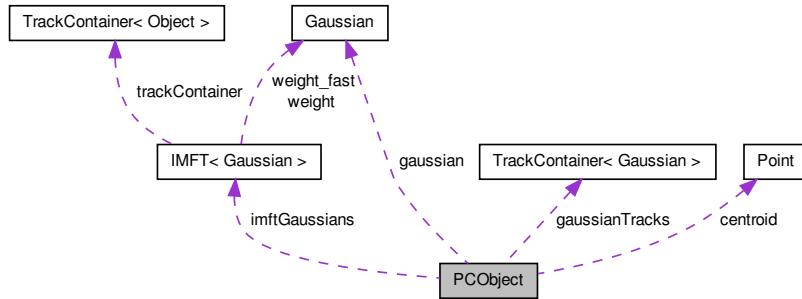
The documentation for this class was generated from the following files:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[particlepose.h](#)
- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[particlepose.cpp](#)

4.18 PCObject Class Reference

```
#include <pcobject.h>
```

Collaboration diagram for PCObject:



Classes

- struct [Edge](#)
- struct [Edge_spatial](#)

Public Types

- `typedef double(* funcWeightGaussian)(Gaussian &g1, Gaussian &g2)`

Public Member Functions

- `PCObject ()`
- `PCObject (int _id)`
- `~PCObject ()`
- `CloudPtr toPointCloud ()`
- `void insert (Point p)`
- `Point getCentroid ()`
- `void initialGMM (double _scale, double _percent)`
- `void filteringGMM_EM (PCObject *prior)`
- `void filteringGMM_incrementalEM (PCObject *prior, double percent)`
- `double likelihood (Point point, Gaussian gaussian)`
- `double likelihood_standard (Point point, Gaussian gaussian)`
- `void initGaussian (int dim)`
- `double evalGMM (Point x)`
- `double evalClosestGMM (Point x)`
- `double evalNormedGMM (Point x, double den)`
- `double evalClosestNormedGMM (Point x, double den)`
- `void simplify (int dim, SIMPLE method, double ratio, int nCluster=0)`

- void `setTransParam` (`vnl_vector< double >` param)
- void `mergeTwoGMMs` (`PCObject *gmm1, PCObject *gmm2`)
- void `setScale` (`double _scale`)
- void `gaussianTrackingInit` (`int _window_short, int _window_long, int _maxID, funcWeightGaussian _weight_gaussian, funcWeightGaussian _weight_gaussian_fast`)
- void `gaussianTracking` ()
- void `updatePredictiveParameters` ()
- void `calculateVelocity` ()
- void `makeTopology` ()
- double `topology_weight` (`Gaussian g1, Gaussian g2`)
- double `topology_posweight_rev` (`Gaussian g1, Gaussian g2`)
- double `topology_velweight_rev` (`Gaussian g1, Gaussian g2`)
- int `componentGraph` (`vector< PCObject > &newObjects`)
- void `setid` (`int _id`)
- int `getid` ()

Public Attributes

- `vector< Point > points`
- `vector< Gaussian > gmm`
- `Point centroid`
- `Gaussian gaussian`
- `STATE state`
- int `dimension`
- int `id`
- `vnl_vector< double > trans_param`
- bool `isParamExist`
- `IMFT< Gaussian > * imftGaussians`
- `TrackContainer< Gaussian > * gaussianTracks`
- int `cnt`
- `vector< Gaussian * > frame`
- `ListGraph * topology_graph`
- `ListGraph::NodeMap< Gaussian > * topology_nodeMap`
- `ListGraph::EdgeMap< double > * topology_edgeMap`
- `vector< Edge > edges`
- double `alpha`
- double `th_edge`
- double `* diffWeight`
- double `filteredWeight`
- double `scale`
- double `percent`

Private Attributes

- int `window_short`
- int `window_long`

4.18.1 Member Typedef Documentation

4.18.1.1 `typedef double(* PCObject::funcWeightGaussian)(Gaussian &g1, Gaussian &g2)`

4.18.2 Constructor & Destructor Documentation

4.18.2.1 `PCObject::PCObject()`

4.18.2.2 `PCObject::PCObject(int _id)`

4.18.2.3 `PCObject::~PCObject()`

4.18.3 Member Function Documentation

4.18.3.1 `void PCObject::calculateVelocity()`

Here is the call graph for this function:



4.18.3.2 `int PCObject::componentGraph(vector< PCObject > & newObjects)`

4.18.3.3 `double PCObject::evalClosestGMM(Point x)`

4.18.3.4 `double PCObject::evalClosestNormedGMM(Point x, double den)`

4.18.3.5 `double PCObject::evalGMM(Point x)`

4.18.3.6 `double PCObject::evalNormedGMM(Point x, double den)`

4.18.3.7 void PCObject::filteringGMM_EM (PCObject * *prior*)

Here is the call graph for this function:

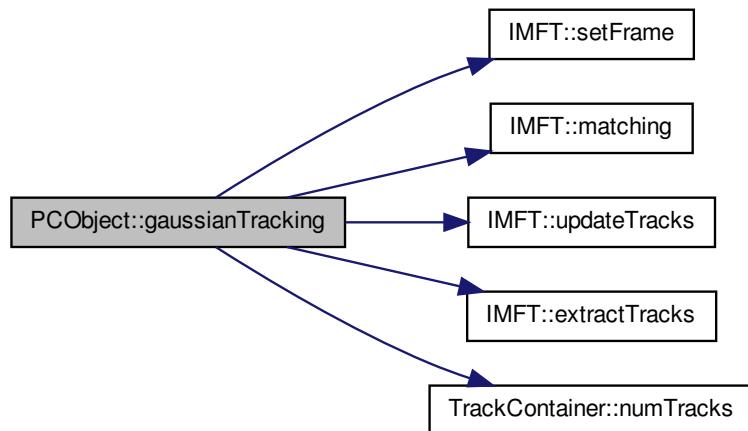
**4.18.3.8 void PCObject::filteringGMM_incrementalEM (PCObject * *prior*, double *percent*)**

Here is the call graph for this function:



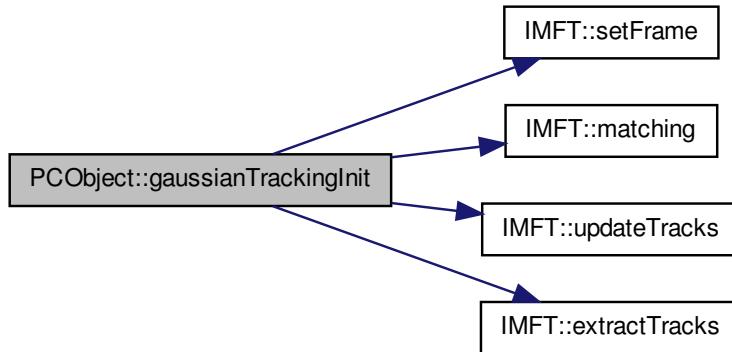
4.18.3.9 void PCObject::gaussianTracking()

Here is the call graph for this function:



```
4.18.3.10 void PCObject::gaussianTrackingInit( int _window_short, int _window_long, int  
_maxID, funcWeightGaussian _weight_gaussian, funcWeightGaussian  
_weight_gaussian_fast )
```

Here is the call graph for this function:



```
4.18.3.11 Point PCObject::getCentroid( )
```

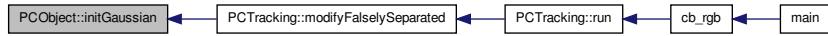
```
4.18.3.12 int PCObject::getId( ) [inline]
```

```
4.18.3.13 void PCObject::initGaussian( int dim )
```

Here is the call graph for this function:

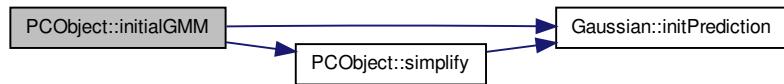


Here is the caller graph for this function:



4.18.3.14 void PCObject::initialGMM (double _scale, double _percent)

Here is the call graph for this function:



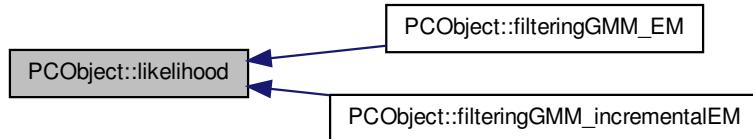
4.18.3.15 void PCObject::insert (Point p)

Here is the caller graph for this function:



4.18.3.16 double PCObject::likelihood (Point *point*, Gaussian *gaussian*)

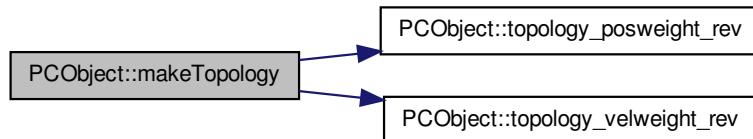
Here is the caller graph for this function:



4.18.3.17 double PCObject::likelihood_standard (Point *point*, Gaussian *gaussian*)

4.18.3.18 void PCObject::makeTopology ()

Here is the call graph for this function:



4.18.3.19 void PCObject::mergeTwoGMMs (PCObject * *gmm1*, PCObject * *gmm2*)

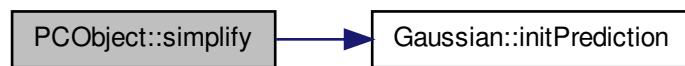
4.18.3.20 void PCObject::setId (int *id*) [inline]

4.18.3.21 void PCObject::setScale (double *scale*) [inline]

4.18.3.22 void PCObject::setTransParam (vnl_vector< double > *param*)

```
4.18.3.23 void PCObject::simplify ( int dim, SIMPLE method, double ratio, int nCluster =  
0 )
```

Here is the call graph for this function:



Here is the caller graph for this function:



```
4.18.3.24 CloudPtr PCObject::toPointCloud ( )
```

Here is the caller graph for this function:



4.18.3.25 double PCObject::topology_posweight_rev (Gaussian g1, Gaussian g2)

Here is the caller graph for this function:



4.18.3.26 double PCObject::topology_velweight_rev (Gaussian g1, Gaussian g2)

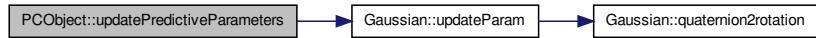
Here is the caller graph for this function:



4.18.3.27 double PCObject::topology_weight (Gaussian g1, Gaussian g2)

4.18.3.28 void PCObject::updatePredictiveParameters ()

Here is the call graph for this function:



4.18.4 Member Data Documentation

4.18.4.1 double PCObject::alpha

4.18.4.2 Point PCObject::centroid

- 4.18.4.3 int PCObject::cnt
- 4.18.4.4 double* PCObject::diffWeight
- 4.18.4.5 int PCObject::dimension
- 4.18.4.6 vector<Edge> PCObject::edges
- 4.18.4.7 double PCObject::filteredWeight
- 4.18.4.8 vector<Gaussian*> PCObject::frame
- 4.18.4.9 Gaussian PCObject::gaussian
- 4.18.4.10 TrackContainer<Gaussian>* PCObject::gaussianTracks
- 4.18.4.11 vector<Gaussian> PCObject::gmm
- 4.18.4.12 int PCObject::id
- 4.18.4.13 IMFT<Gaussian>* PCObject::imftGaussians
- 4.18.4.14 bool PCObject::isParamExist
- 4.18.4.15 double PCObject::percent
- 4.18.4.16 vector<Point> PCObject::points
- 4.18.4.17 double PCObject::scale
- 4.18.4.18 STATE PCObject::state
- 4.18.4.19 double PCObject::th_edge
- 4.18.4.20 ListGraph::EdgeMap<double>* PCObject::topology_edgeMap
- 4.18.4.21 ListGraph* PCObject::topology_graph
- 4.18.4.22 ListGraph::NodeMap<Gaussian>* PCObject::topology_nodeMap
- 4.18.4.23 vnl_vector<double> PCObject::trans_param
- 4.18.4.24 int PCObject::window_long [private]
- 4.18.4.25 int PCObject::window_short [private]

The documentation for this class was generated from the following files:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[pcobject.h](#)
- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[pcobject.cpp](#)

4.19 PCObjectContainer Class Reference

```
#include <pcobjectcontainer.h>
```

Public Member Functions

- [PCObjectContainer \(\)](#)
- [PCObjectContainer \(CloudPtr _pCloud\)](#)
- [~PCObjectContainer \(\)](#)
- int [numObjects \(\)](#)
- bool [deleteObject \(int id\)](#)
- void [makeNewObject \(PCObject &object\)](#)
- void [initGMM \(double scale, double percent\)](#)
- void [clearAll \(\)](#)

Public Attributes

- vector< [PCObject](#) > [objects](#)
- int [num](#)

Private Member Functions

- void [makingObjects \(\)](#)

Private Attributes

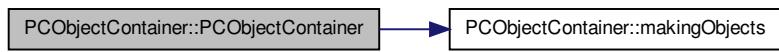
- CloudPtr [pCloud](#)
- double [scale](#)

4.19.1 Constructor & Destructor Documentation

4.19.1.1 PCObjectContainer::PCObjectContainer()

4.19.1.2 PCObjectContainer::PCObjectContainer (CloudPtr *pCloud*)

Here is the call graph for this function:



4.19.1.3 PCObjectContainer::~PCObjectContainer ()

4.19.2 Member Function Documentation

4.19.2.1 void PCObjectContainer::clearAll ()

Here is the caller graph for this function:



4.19.2.2 bool PCObjectContainer::deleteObject (int *id*)

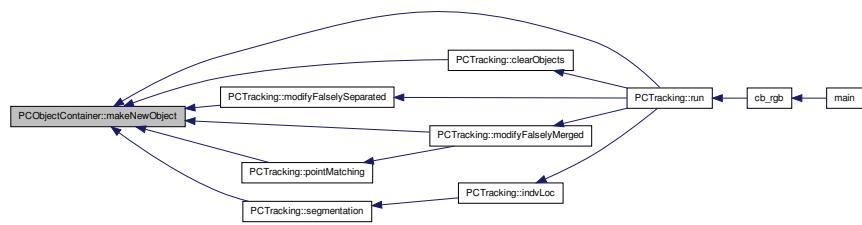
4.19.2.3 void PCObjectContainer::initGMM (double *scale*, double *percent*)

Here is the call graph for this function:



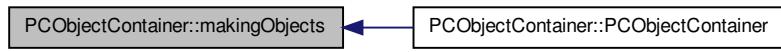
4.19.2.4 void PCObjectContainer::makeNewObject(PCObject & object)

Here is the caller graph for this function:



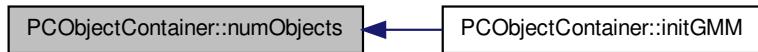
4.19.2.5 void PCObjectContainer::makingObjects() [private]

Here is the caller graph for this function:



4.19.2.6 int PCObjectContainer::numObjects() [inline]

Here is the caller graph for this function:



4.19.3 Member Data Documentation

4.19.3.1 int PCObjectContainer::num

4.19.3.2 `vector<PCObject> PCObjectContainer::objects`

4.19.3.3 `CloudPtr PCObjectContainer::pCloud` [private]

4.19.3.4 `double PCObjectContainer::scale` [private]

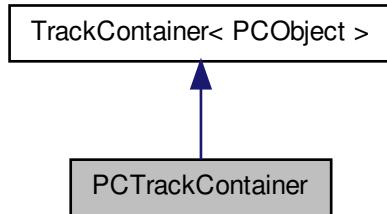
The documentation for this class was generated from the following files:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[pcobjectcontainer.h](#)
- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[pcobjectcontainer.cpp](#)

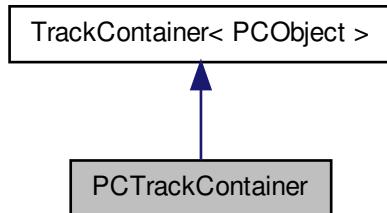
4.20 PCTrackContainer Class Reference

#include <pctrackcontainer.h>

Inheritance diagram for PCTrackContainer:



Collaboration diagram for PCTrackContainer:



Public Member Functions

- `PCTrackContainer ()`
- `PCTrackContainer (int _maxFrame=1000)`
- `void toPointCloudXYZI (Cloud &cloudOut)`
- `visualization_msgs::MarkerArray toMarkerGaussians ()`
- `visualization_msgs::MarkerArray toMarkerGMMs ()`
- `visualization_msgs::MarkerArray oldGaussians ()`
- `visualization_msgs::MarkerArray toMarkerIDs ()`
- `visualization_msgs::MarkerArray oldMarkerIDs ()`
- `visualization_msgs::Marker toMarkerEdges ()`
- `void evaluate ()`
- `void iros2014 ()`

Public Attributes

- `int numTruePoints`
- `int numFalsePoints`
- `int numTotalPoints`
- `bool isUpdated`

Private Types

- `typedef pcl::PointXYZRGB PointT`
- `typedef pcl::PointCloud< PointT > Cloud`
- `typedef Cloud::Ptr CloudPtr`
- `typedef Cloud::ConstPtr CloudConstPtr`

Private Member Functions

- `float SIGN (float x)`
- `float NORM (float a, float b, float c, float d)`
- `void eigenOrdering (const Eigen::Vector3d &values, const Eigen::Matrix3d &vectors, Eigen::Vector3d &values_ordered, Eigen::Matrix3d &vectors_ordered)`

Private Attributes

- `int maxFrame`
- `vector< int > oldGaussiansId`
- `vector< int > oldTrackIDs`

4.20.1 Member Typedef Documentation

- 4.20.1.1 `typedef pcl::PointCloud<PointT> PCTrackContainer::Cloud [private]`
- 4.20.1.2 `typedef Cloud::ConstPtr PCTrackContainer::CloudConstPtr [private]`
- 4.20.1.3 `typedef Cloud::Ptr PCTrackContainer::CloudPtr [private]`
- 4.20.1.4 `typedef pcl::PointXYZRGB PCTrackContainer::PointT [private]`

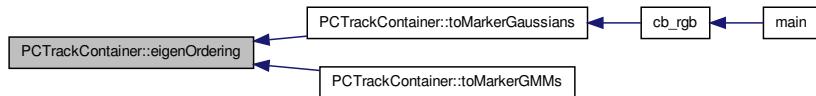
4.20.2 Constructor & Destructor Documentation

- 4.20.2.1 `PCTrackContainer::PCTrackContainer()`
- 4.20.2.2 `PCTrackContainer::PCTrackContainer(int _maxFrame = 1000)`

4.20.3 Member Function Documentation

- 4.20.3.1 `void PCTrackContainer::eigenOrdering(const Eigen::Vector3d & values, const Eigen::Matrix3d & vectors, Eigen::Vector3d & values_ordered, Eigen::Matrix3d & vectors_ordered) [private]`

Here is the caller graph for this function:



- 4.20.3.2 `void PCTrackContainer::evaluate()`

Here is the call graph for this function:



4.20.3.3 void PCTrackContainer::iros2014()

Here is the call graph for this function:

**4.20.3.4 float PCTrackContainer::NORM(float a, float b, float c, float d) [inline, private]****4.20.3.5 visualization_msgs::MarkerArray PCTrackContainer::oldGaussians()**

Here is the caller graph for this function:

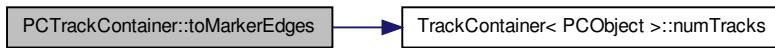
**4.20.3.6 visualization_msgs::MarkerArray PCTrackContainer::oldMarkerIDs()**

Here is the caller graph for this function:

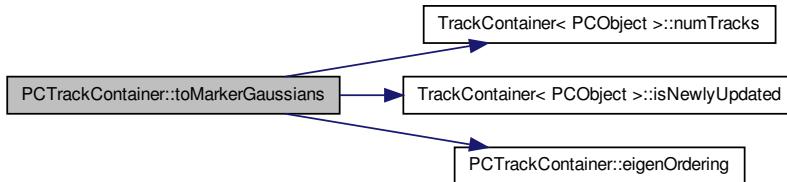
**4.20.3.7 float PCTrackContainer::SIGN(float x) [inline, private]**

4.20.3.8 visualization_msgs::Marker PCTrackContainer::toMarkerEdges()

Here is the call graph for this function:

**4.20.3.9 visualization_msgs::MarkerArray PCTrackContainer::toMarkerGaussians()**

Here is the call graph for this function:

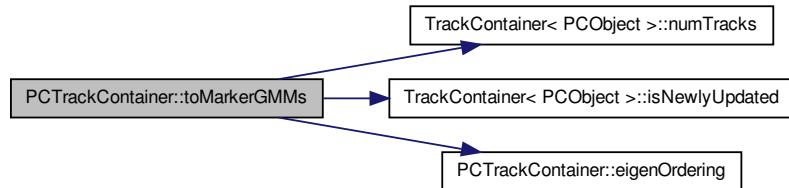


Here is the caller graph for this function:

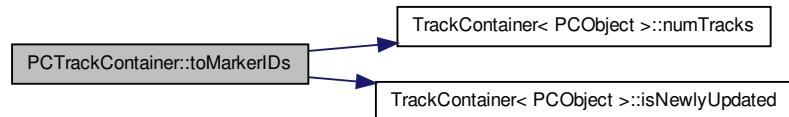


4.20.3.10 visualization_msgs::MarkerArray PCTrackContainer::toMarkerGMMs ()

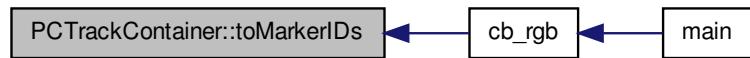
Here is the call graph for this function:

**4.20.3.11 visualization_msgs::MarkerArray PCTrackContainer::toMarkerIDs ()**

Here is the call graph for this function:

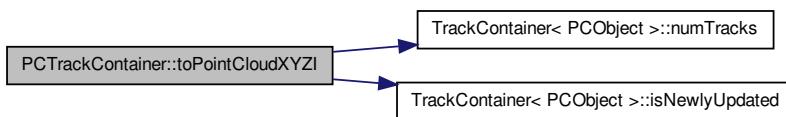


Here is the caller graph for this function:



4.20.3.12 void PCTrackContainer::toPointCloudXYZI (Cloud & *cloudOut*)

Here is the call graph for this function:



Here is the caller graph for this function:



4.20.4 Member Data Documentation

4.20.4.1 bool PCTrackContainer::isUpdated

4.20.4.2 int PCTrackContainer::maxFrame [private]

Reimplemented from [TrackContainer< PCObject >](#).

4.20.4.3 int PCTrackContainer::numFalsePoints

4.20.4.4 int PCTrackContainer::numTotalPoints

4.20.4.5 int PCTrackContainer::numTruePoints

4.20.4.6 vector<int> PCTrackContainer::oldGaussiansId [private]

4.20.4.7 vector<int> PCTrackContainer::oldTrackIDs [private]

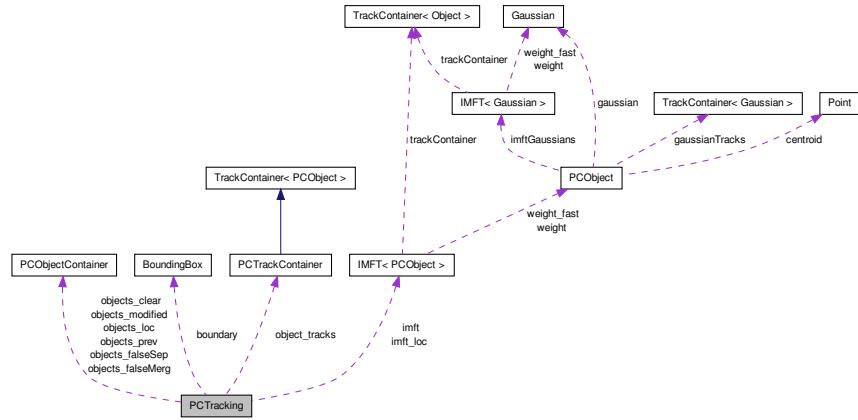
The documentation for this class was generated from the following files:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[pctrackcontainer.h](#)
- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[pctrackcontainer.cpp](#)

4.21 PCTracking Class Reference

```
#include <pctracking.h>
```

Collaboration diagram for PCTracking:



Public Member Functions

- [PCTracking \(int _isDebug, string _frame_id, ros::Publisher *_pub_scene, ros::Publisher *_pub_model, int dimension, double cont_sampling, double cont_simplify, double segTol\)](#)
- [~PCTracking \(\)](#)
- [void run \(CloudPtr _pCloud\)](#)
- [CloudPtr getFilteredPC \(\)](#)
- [CloudPtr getSegmentedPC \(\)](#)
- [CloudPtr getLastModel \(\)](#)
- [CloudPtr getParticles \(\)](#)

Public Attributes

- [IMFT< PCObject > * imft](#)
- [IMFT< PCObject > * imft_loc](#)
- [PCTrackContainer * object_tracks](#)
- [ListDigraph dg_ambiguity](#)
- [ListDigraph::NodeMap< ObjectNode > * dg_ambiguity_nodes](#)
- [ListDigraph::ArcMap< Weight > * dg_ambiguity_arcs](#)

Private Member Functions

- void `segmentationGMM` (CloudPtr pCloud, PCObjectContainer &objects)
- void `segmentation` (CloudPtr pCloud, PCObjectContainer &objects)
- void `pointMatching` (PCObjectContainer &predictiveObjects, PCObject &parentObject, CloudPtr unmatchedPoints, PCObjectContainer &objects_separated)
- void `trackingGaussians` (PCObject *object, CloudPtr observedPoints, PCObjectContainer &updatedObjects)
- void `indvLoc` ()
- void `ambiguityTest` ()
- void `typeProblem` (ListDigraph::Node &node)
- void `typeClear` (ListDigraph::Node &node)
- void `initdg` (Type type)
- void `confirmDigraph` ()
- void `clearObjects` ()
- void `modifyFalselySeparated` ()
- void `modifyFalselyMerged` ()
- bool `isBoundary` (PCObject &object)
- void `indexing` ()
- void `updateTracks` ()

Private Attributes

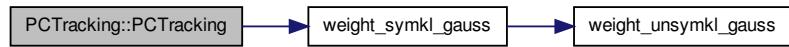
- string `frame_id`
- ros::Publisher * `pub_scene`
- ros::Publisher * `pub_model`
- CloudPtr pCloud
- CloudPtr pCloud_seg
- CloudPtr pCloud_lastModel
- CloudPtr pCloud_particles
- PCObjectContainer objects_prev
- PCObjectContainer objects_loc
- PCObjectContainer objects_modified
- PCObjectContainer objects_clear
- PCObjectContainer objects_falseSep
- PCObjectContainer objects_falseMerg
- vector< FalseObjects > falselySeparated
- vector< FalseObjects > falselyMerged
- BoundingBox boundary
- double `boundaryMargin`
- CloudPtr unmatchedPoints
- vector< CloudPtr > observedPointsList
- int `nObjects`
- int `cnt`
- bool `isDebug`
- vector< ListDigraph::Node > problemNodes

- vector< ListDigraph::Arc > `problemArcs`
- vector< ListDigraph::Node > `clearNodes`
- vector< ListDigraph::Arc > `clearArcs`
- vector< Particlepose > `m_partFilters`
- double `scale`
- double `percent`
- int `dim`
- double `segmentation_tolerance`
- double `segmentation_minSize`
- double `segmentation_maxSize`
- double `maxProbAssociation`
- double `thrProbScene`
- int `minPoints`
- int `maxID`
- int * `r`
- int * `g`
- int * `b`
- int `maxFrame`

4.21.1 Constructor & Destructor Documentation

4.21.1.1 `PCTracking::PCTracking (int _isDebug, string _frame_id, ros::Publisher * _pub_scene, ros::Publisher * _pub_model, int dimension, double cont_sampling, double cont_simplify, double segTol)`

Here is the call graph for this function:

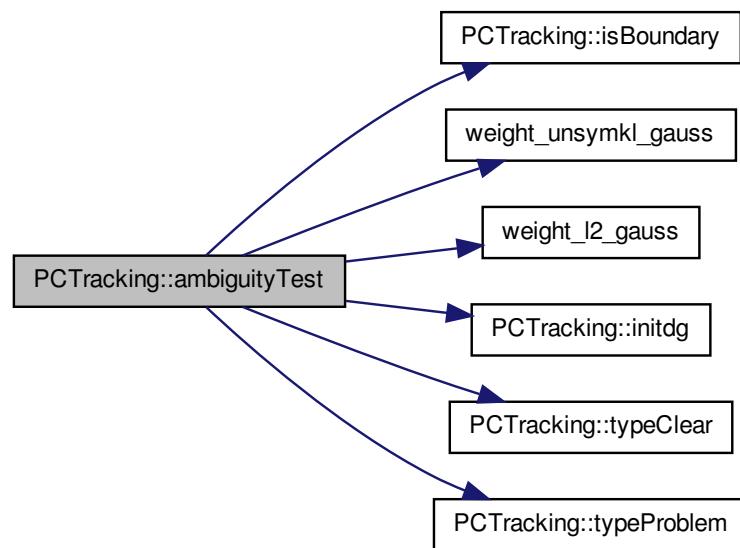


4.21.1.2 `PCTracking::~PCTracking ()`

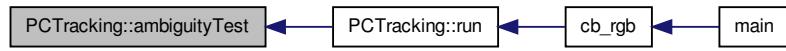
4.21.2 Member Function Documentation

4.21.2.1 void PCTracking::ambiguityTest() [private]

Here is the call graph for this function:



Here is the caller graph for this function:



4.21.2.2 void PCTracking::clearObjects() [private]

Here is the call graph for this function:



Here is the caller graph for this function:

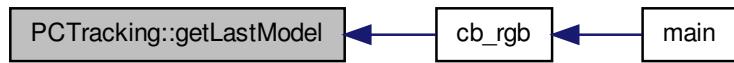
**4.21.2.3 void PCTracking::confirmDigraph() [private]****4.21.2.4 CloudPtr PCTracking::getFilteredPC() [inline]**

Here is the caller graph for this function:



4.21.2.5 CloudPtr PCTracking::getLastModel() [inline]

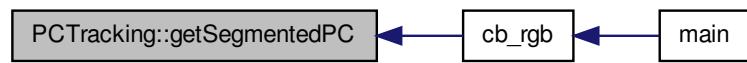
Here is the caller graph for this function:

**4.21.2.6 CloudPtr PCTracking::getParticles() [inline]**

Here is the caller graph for this function:

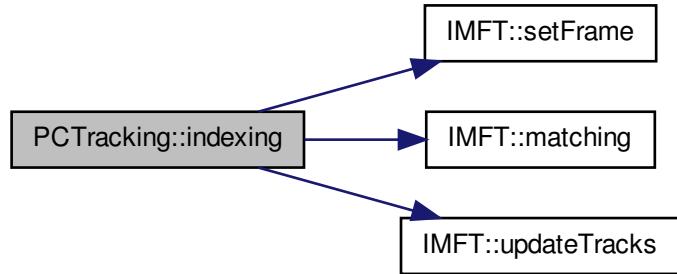
**4.21.2.7 CloudPtr PCTracking::getSegmentedPC() [inline]**

Here is the caller graph for this function:

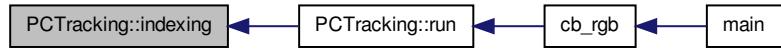


4.21.2.8 void PCTracking::indexing() [private]

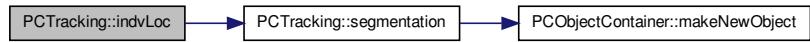
Here is the call graph for this function:



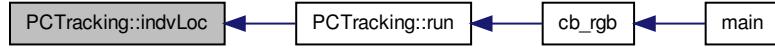
Here is the caller graph for this function:

**4.21.2.9 void PCTracking::indvLoc() [private]**

Here is the call graph for this function:



Here is the caller graph for this function:



4.21.2.10 void PCTracking::initdg (Type type) [private]

Here is the caller graph for this function:



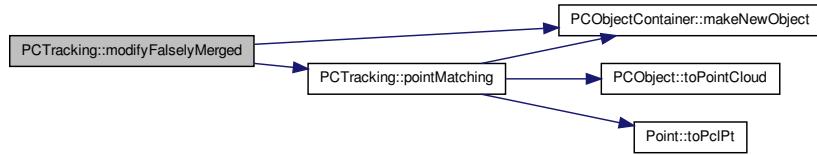
4.21.2.11 bool PCTracking::isBoundary (PCObject & object) [private]

Here is the caller graph for this function:



4.21.2.12 void PCTracking::modifyFalselyMerged () [private]

Here is the call graph for this function:

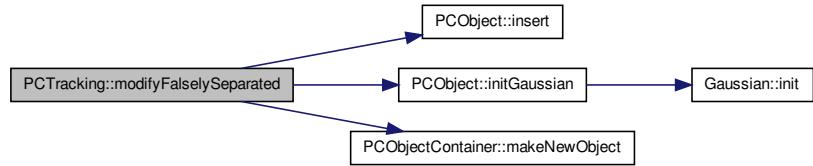


Here is the caller graph for this function:



4.21.2.13 void PCTracking::modifyFalselySeparated() [private]

Here is the call graph for this function:

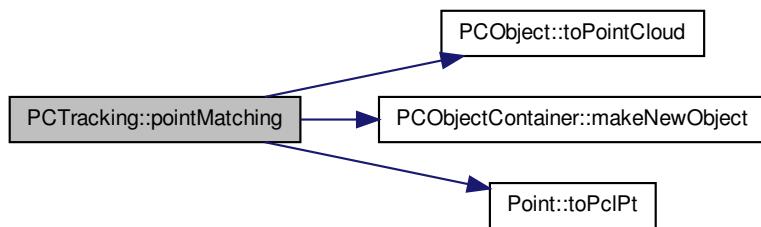


Here is the caller graph for this function:



```
4.21.2.14 void PCTracking::pointMatching ( PCObjectContainer & predictiveObjects,  
PCObject & parentObject, CloudPtr unmatchedPoints, PCObjectContainer &  
objects_separated ) [private]
```

Here is the call graph for this function:

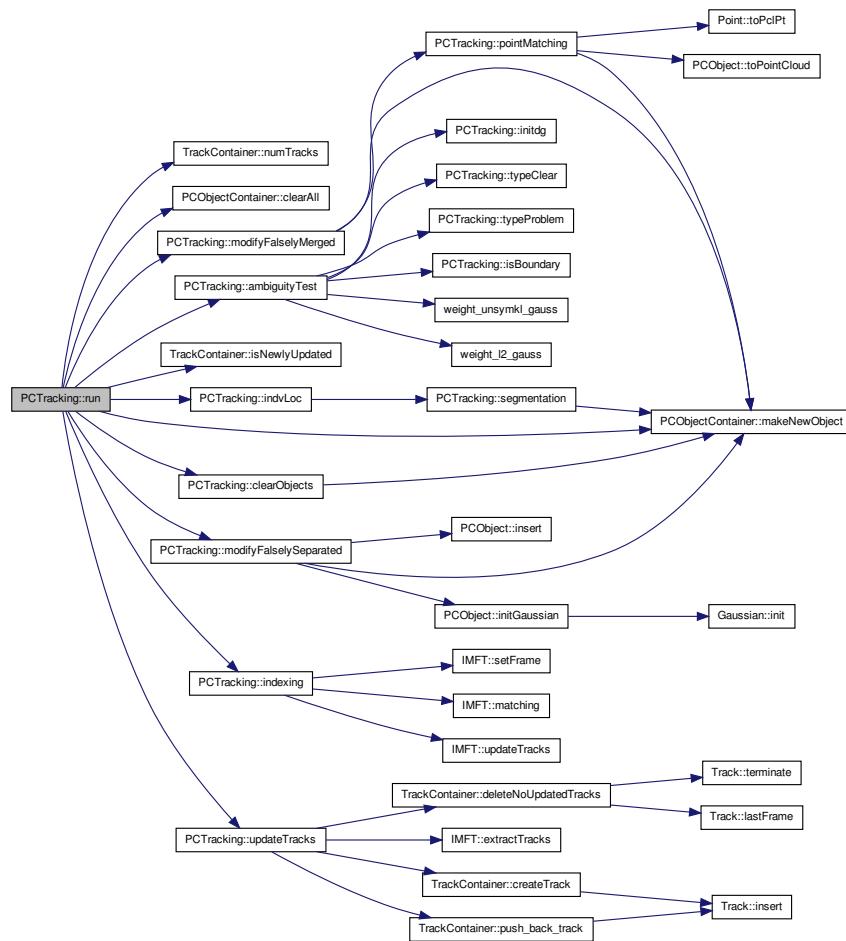


Here is the caller graph for this function:



4.21.2.15 void PCTracking::run (CloudPtr <*pCloud*>)

Here is the call graph for this function:



Here is the caller graph for this function:

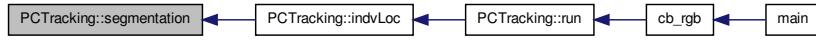


```
4.21.2.16 void PCTracking::segmentation ( CloudPtr pCloud, PCObjectContainer & objects ) [private]
```

Here is the call graph for this function:



Here is the caller graph for this function:



```
4.21.2.17 void PCTracking::segmentationGMM ( CloudPtr pCloud, PCObjectContainer & objects ) [private]
```

```
4.21.2.18 void PCTracking::trackingGaussians ( PCObject * object, CloudPtr observedPoints, PCObjectContainer & updatedObjects ) [private]
```

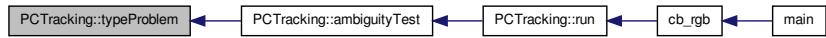
```
4.21.2.19 void PCTracking::typeClear ( ListDigraph::Node & node ) [private]
```

Here is the caller graph for this function:



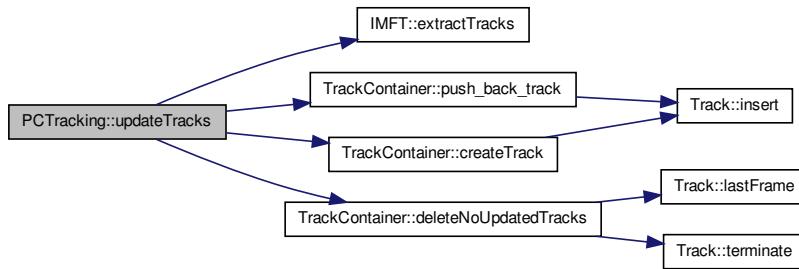
4.21.2.20 void PCTracking::typeProblem (ListDigraph::Node & node) [private]

Here is the caller graph for this function:



4.21.2.21 void PCTracking::updateTracks () [private]

Here is the call graph for this function:



Here is the caller graph for this function:



4.21.3 Member Data Documentation

4.21.3.1 int * PCTracking::b [private]

4.21.3.2 BoundingBox PCTracking::boundary [private]

4.21.3.3 double PCTracking::boundaryMargin [private]

- 4.21.3.4 `vector<ListDigraph::Arc> PCTracking::clearArcs` [private]
- 4.21.3.5 `vector<ListDigraph::Node> PCTracking::clearNodes` [private]
- 4.21.3.6 `int PCTracking::cnt` [private]
- 4.21.3.7 `ListDigraph PCTracking::dg_ambiguity`
- 4.21.3.8 `ListDigraph::ArcMap<Weight>* PCTracking::dg_ambiguity_arcs`
- 4.21.3.9 `ListDigraph::NodeMap<ObjectNode>* PCTracking::dg_ambiguity_nodes`
- 4.21.3.10 `int PCTracking::dim` [private]
- 4.21.3.11 `vector<FalseObjects> PCTracking::falselyMerged` [private]
- 4.21.3.12 `vector<FalseObjects> PCTracking::falselySeparated` [private]
- 4.21.3.13 `string PCTracking::frame_id` [private]
- 4.21.3.14 `int * PCTracking::g` [private]
- 4.21.3.15 `IMFT<PCObject>* PCTracking::imft`
- 4.21.3.16 `IMFT<PCObject>* PCTracking::imft_loc`
- 4.21.3.17 `bool PCTracking::isDebug` [private]
- 4.21.3.18 `vector<Particlepose> PCTracking::m_partFilters` [private]
- 4.21.3.19 `int PCTracking::maxFrame` [private]
- 4.21.3.20 `int PCTracking::maxID` [private]
- 4.21.3.21 `double PCTracking::maxProbAssociation` [private]
- 4.21.3.22 `int PCTracking::minPoints` [private]
- 4.21.3.23 `int PCTracking::nObjects` [private]
- 4.21.3.24 `PCTrackContainer* PCTracking::object_tracks`
- 4.21.3.25 `PCObjectContainer PCTracking::objects_clear` [private]
- 4.21.3.26 `PCObjectContainer PCTracking::objects_falseMerg` [private]
- 4.21.3.27 `PCObjectContainer PCTracking::objects_falseSep` [private]

- 4.21.3.28 **PCObjectContainer** **PCTracking::objects_loc** [private]
- 4.21.3.29 **PCObjectContainer** **PCTracking::objects_modified** [private]
- 4.21.3.30 **PCObjectContainer** **PCTracking::objects_prev** [private]
- 4.21.3.31 **vector<CloudPtr>** **PCTracking::observedPointsList** [private]
- 4.21.3.32 **CloudPtr** **PCTracking::pCloud** [private]
- 4.21.3.33 **CloudPtr** **PCTracking::pCloud_lastModel** [private]
- 4.21.3.34 **CloudPtr** **PCTracking::pCloud_particles** [private]
- 4.21.3.35 **CloudPtr** **PCTracking::pCloud_seg** [private]
- 4.21.3.36 **double** **PCTracking::percent** [private]
- 4.21.3.37 **vector<ListDigraph::Arc>** **PCTracking::problemArcs** [private]
- 4.21.3.38 **vector<ListDigraph::Node>** **PCTracking::problemNodes** [private]
- 4.21.3.39 **ros::Publisher*** **PCTracking::pub_model** [private]
- 4.21.3.40 **ros::Publisher*** **PCTracking::pub_scene** [private]
- 4.21.3.41 **int*** **PCTracking::r** [private]
- 4.21.3.42 **double** **PCTracking::scale** [private]
- 4.21.3.43 **double** **PCTracking::segmentation_maxSize** [private]
- 4.21.3.44 **double** **PCTracking::segmentation_minSize** [private]
- 4.21.3.45 **double** **PCTracking::segmentation_tolerance** [private]
- 4.21.3.46 **double** **PCTracking::thrProbScene** [private]
- 4.21.3.47 **CloudPtr** **PCTracking::unmatchedPoints** [private]

The documentation for this class was generated from the following files:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[pctracking.h](#)
- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[pctracking.cpp](#)

4.22 Point Class Reference

```
#include <gaussian.h>
```

Public Member Functions

- `Point ()`
- `Point (PointT point, int _dim)`
- `PointT toPclPt ()`

Public Attributes

- Eigen::Vector3d `pos`
- Eigen::Vector3d `rgb`
- int `id`
- int `dim`

4.22.1 Constructor & Destructor Documentation

4.22.1.1 `Point::Point ()`

4.22.1.2 `Point::Point (PointT point, int _dim)`

4.22.2 Member Function Documentation

4.22.2.1 `PointT Point::toPclPt ()`

Here is the caller graph for this function:



4.22.3 Member Data Documentation

4.22.3.1 `int Point::dim`

4.22.3.2 `int Point::id`

4.22.3.3 `Eigen::Vector3d Point::pos`

4.22.3.4 `Eigen::Vector3d Point::rgb`

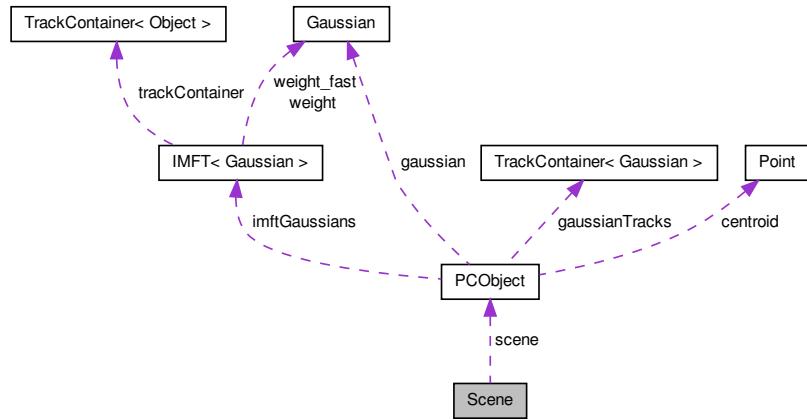
The documentation for this class was generated from the following files:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[gaussian.h](#)
- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[gaussian.cpp](#)

4.23 Scene Struct Reference

```
#include <pctracking.h>
```

Collaboration diagram for Scene:



Public Attributes

- [VecObjectPtr models](#)
- [PCObject * scene](#)
- [VecObject trasformedModels](#)

4.23.1 Member Data Documentation

4.23.1.1 [VecObjectPtr Scene::models](#)

4.23.1.2 [PCObject* Scene::scene](#)

4.23.1.3 [VecObject Scene::trasformedModels](#)

The documentation for this struct was generated from the following file:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[pctracking.h](#)

4.24 Track< Object > Class Template Reference

```
#include <track.h>
```

Classes

- struct [Frame](#)
- struct [Node](#)
- struct [V](#)

Public Member Functions

- [Track](#) (int _id, int _maxFrame=1000)
- void [insert](#) (Object &object, int time)
- [Frame lastFrame](#) ()
- [Frame * lastFramePtr](#) ()
- [Frame getFrameFromLast](#) (int n)
- [Frame * framePtrFromLast](#) (int n)
- void [updateObjectAtFrame](#) (int time, Object &object)
- void [terminate](#) ()
- int [frameSize](#) ()

Public Attributes

- vector< [Frame](#) > [frames](#)
- int [id](#)
- int [maxFrame](#)

```
template<class Object> class Track< Object >
```

4.24.1 Constructor & Destructor Documentation

4.24.1.1 template<class Object > [Track< Object >::Track](#) (int *_id*, int *_maxFrame* = 1000)

4.24.2 Member Function Documentation

4.24.2.1 template<class Object > [Frame* Track< Object >::framePtrFromLast](#) (int *n*)
[inline]

4.24.2.2 template<class Object > int [Track< Object >::frameSize](#) () [inline]

4.24.2.3 template<class Object > [Frame Track< Object >::getFrameFromLast](#) (int *n*)
[inline]

4.24.2.4 template<class Object > void Track< Object >::insert(Object & object, int time)

Here is the caller graph for this function:



4.24.2.5 template<class Object > Frame Track< Object >::lastFrame() [inline]

Here is the caller graph for this function:



4.24.2.6 template<class Object > Frame* Track< Object >::lastFramePtr() [inline]

4.24.2.7 template<class Object > void Track< Object >::terminate()

Here is the caller graph for this function:



4.24.2.8 template<class Object> void Track< Object >::updateObjectAtFrame (int time, Object & object)

Here is the caller graph for this function:



4.24.3 Member Data Documentation

4.24.3.1 template<class Object> vector<Frame> Track< Object >::frames

4.24.3.2 template<class Object> int Track< Object >::id

4.24.3.3 template<class Object> int Track< Object >::maxFrame

The documentation for this class was generated from the following files:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[track.h](#)
- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[track.hpp](#)

4.25 TrackContainer< Object > Class Template Reference

```
#include <trackcontainer.h>
```

Public Member Functions

- [TrackContainer \(int _maxFrame=1000, int _maxID=1000\)](#)
- [~TrackContainer \(\)](#)
- int [numTracks \(\)](#)
- bool [createTrack \(Object &object, int initT\)](#)
- bool [createTrack \(Object &object, int initT, int id\)](#)
- bool [merge \(TrackContainer< Object > container\)](#)
- bool [push_back_track \(int id, Object &object, int time\)](#)
- void [deleteNoUpdatedTracks \(int size\)](#)
- int [newId \(\)](#)
- bool [updateObject \(int id, int time, Object &object\)](#)
- bool [isNewlyUpdated \(int at\)](#)

Public Attributes

- vector< int > [deletedTrackIDs](#)
- [VecTrackPtr](#) [tracks](#)
- int [currentT](#)

Protected Attributes

- int [maxFrame](#)
- int [maxID](#)
- int * [r](#)
- int * [g](#)
- int * [b](#)
- int [oldCnt](#)
- int [oldCnt_gaussians](#)

Private Types

- [typedef Track< Object >](#) [TrackT](#)
- [typedef vector< TrackT * >](#) [VecTrackPtr](#)

```
template<class Object> class TrackContainer< Object >
```

4.25.1 Member Typedef Documentation

4.25.1.1 template<class Object> [typedef Track<Object>](#) [TrackContainer< Object >::TrackT](#) [private]

4.25.1.2 template<class Object> [typedef vector<TrackT*>](#) [TrackContainer< Object >::VecTrackPtr](#) [private]

4.25.2 Constructor & Destructor Documentation

4.25.2.1 template<class Object > [TrackContainer< Object >::TrackContainer](#) (int [_maxFrame = 1000](#), int [_maxID = 1000](#))

4.25.2.2 template<class Object > [TrackContainer< Object >::~TrackContainer](#) ()

4.25.3 Member Function Documentation

4.25.3.1 template<class Object> bool TrackContainer< Object >::createTrack (Object & object, int initT)

Here is the call graph for this function:



Here is the caller graph for this function:



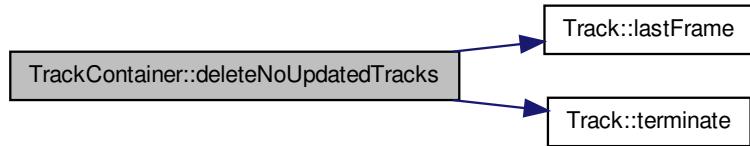
4.25.3.2 template<class Object> bool TrackContainer< Object >::createTrack (Object & object, int initT, int id)

Here is the call graph for this function:



4.25.3.3 template<class Object > void TrackContainer< Object >::deleteNoUpdatedTracks(int size)

Here is the call graph for this function:



Here is the caller graph for this function:



4.25.3.4 template<class Object > bool TrackContainer< Object >::is NewlyUpdated(int at)

Here is the caller graph for this function:

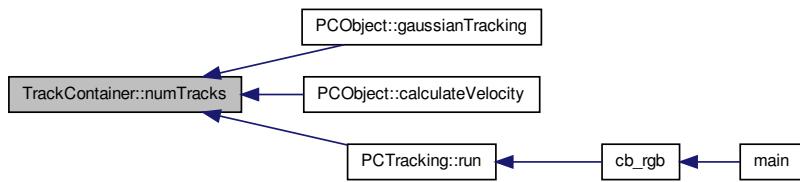


4.25.3.5 template<class Object> bool TrackContainer< Object >::merge (TrackContainer< Object > container)

4.25.3.6 template<class Object > int TrackContainer< Object >::newId()

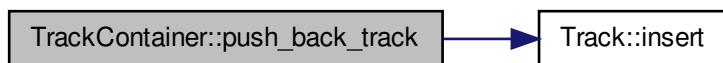
4.25.3.7 template<class Object> int TrackContainer< Object >::numTracks ()
[inline]

Here is the caller graph for this function:



4.25.3.8 template<class Object> bool TrackContainer< Object >::push_back_track (int *id*, Object & *object*, int *time*)

Here is the call graph for this function:



Here is the caller graph for this function:



4.25.3.9 `template<class Object> bool TrackContainer< Object >::updateObject (int id, int time, Object & object)`

Here is the call graph for this function:



4.25.4 Member Data Documentation

4.25.4.1 `template<class Object> int * TrackContainer< Object >::b [protected]`

4.25.4.2 `template<class Object> int TrackContainer< Object >::currentT`

4.25.4.3 `template<class Object> vector<int> TrackContainer< Object >::deletedTrackIDs`

4.25.4.4 `template<class Object> int * TrackContainer< Object >::g [protected]`

4.25.4.5 `template<class Object> int TrackContainer< Object >::maxFrame [protected]`

Reimplemented in [PCTrackContainer](#).

4.25.4.6 `template<class Object> int TrackContainer< Object >::maxID [protected]`

4.25.4.7 `template<class Object> int TrackContainer< Object >::oldCnt [protected]`

4.25.4.8 `template<class Object> int TrackContainer< Object >::oldCnt_gaussians [protected]`

4.25.4.9 `template<class Object> int* TrackContainer< Object >::r [protected]`

4.25.4.10 `template<class Object> VecTrackPtr TrackContainer< Object >::tracks`

The documentation for this class was generated from the following files:

- [/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/trackcontainer.h](#)
- [/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/trackcontainer.hpp](#)

4.26 TrackPoint Struct Reference

```
#include <pctracking.h>
```

Public Attributes

- int `pointID`
- int `trackID`

4.26.1 Member Data Documentation

4.26.1.1 int `TrackPoint::pointID`

4.26.1.2 int `TrackPoint::trackID`

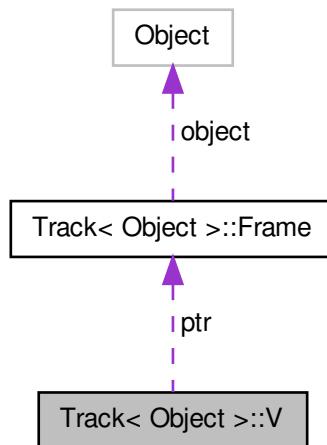
The documentation for this struct was generated from the following file:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking.h

4.27 Track< Object >::V Struct Reference

```
#include <track.h>
```

Collaboration diagram for Track< Object >::V:



Public Attributes

- int `id`
- bool `isIn`
- int `nFrame`
- int `edgeID`
- `Frame` ptr
- bool `isTrack`
- int `nTrack`

```
template<class Object> struct Track< Object >::V
```

4.27.1 Member Data Documentation

4.27.1.1 template<class Object > int Track< Object >::V::edgeID

4.27.1.2 template<class Object > int Track< Object >::V::id

4.27.1.3 template<class Object > bool Track< Object >::V::isIn

4.27.1.4 template<class Object > bool Track< Object >::V::isTrack

4.27.1.5 template<class Object > int Track< Object >::V::nFrame

4.27.1.6 template<class Object > int Track< Object >::V::nTrack

4.27.1.7 template<class Object > Frame Track< Object >::V::ptr

The documentation for this struct was generated from the following file:

- /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/[track.h](#)

4.28 Vcsh Class Reference

```
#include <csfeature.h>
```

Public Member Functions

- `Vcsh ()`
- `~Vcsh ()`
- void `addrbg` (`Vector3i` `rgb`)

Public Attributes

- `std::vector< float >` `hist`

4.28.1 Constructor & Destructor Documentation

4.28.1.1 `Vcsh::Vcsh() [inline]`

4.28.1.2 `Vcsh::~Vcsh() [inline]`

4.28.2 Member Function Documentation

4.28.2.1 `void Vcsh::addrbg(Vector3i rgb)`

4.28.3 Member Data Documentation

4.28.3.1 `std::vector<float> Vcsh::hist`

The documentation for this class was generated from the following file:

- `/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.h`

4.29 Weight Struct Reference

```
#include <pctracking.h>
```

Public Attributes

- `double k1`
- `double l2`
- `Type type`

4.29.1 Member Data Documentation

4.29.1.1 `double Weight::k1`

4.29.1.2 `double Weight::l2`

4.29.1.3 `Type Weight::type`

The documentation for this struct was generated from the following file:

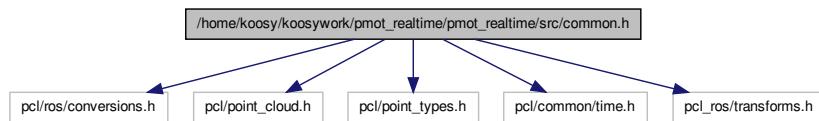
- `/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking.h`

Chapter 5

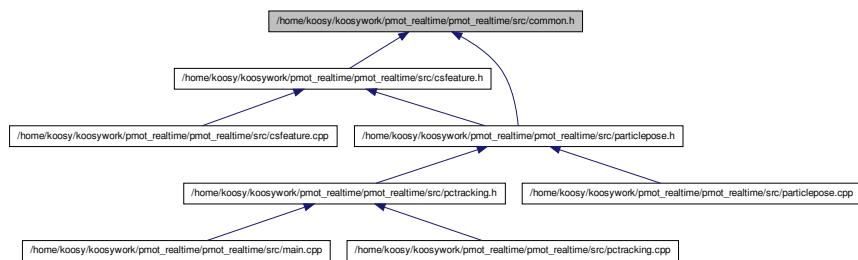
File Documentation

5.1 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/common.h File Reference

```
#include <pcl/ros/conversions.h>      #include <pcl/point_-  
cloud.h> #include <pcl/point_types.h> #include <pcl/common/time.-  
h> #include <pcl_ros/transforms.h> Include dependency graph for  
common.h:
```



This graph shows which files directly or indirectly include this file:



Typedefs

- `typedef pcl::PointXYZRGB PointT`
- `typedef pcl::PointCloud< PointT > Cloud`
- `typedef Cloud::Ptr CloudPtr`
- `typedef Cloud::ConstPtr CloudConstPtr`
- `typedef pcl::PointXYZRGBNormal PointNT`
- `typedef pcl::PointCloud< PointNT > CloudN`
- `typedef CloudN::Ptr CloudNPtr`

Functions

- `void rgbToHSV (int r, int g, int b, float &hr, float &sr, float &vr)`
- `float colordist (PointNT a, PointNT b)`

5.1.1 Typedef Documentation

5.1.1.1 `typedef pcl::PointCloud<PointT> Cloud`

5.1.1.2 `typedef Cloud::ConstPtr CloudConstPtr`

5.1.1.3 `typedef pcl::PointCloud<PointNT> CloudN`

5.1.1.4 `typedef CloudN::Ptr CloudNPtr`

5.1.1.5 `typedef Cloud::Ptr CloudPtr`

5.1.1.6 `typedef pcl::PointXYZRGBNormal PointNT`

5.1.1.7 `typedef pcl::PointXYZRGB PointT`

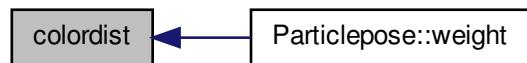
5.1.2 Function Documentation

5.1.2.1 `float colordist (PointNT a, PointNT b) [inline]`

Here is the call graph for this function:

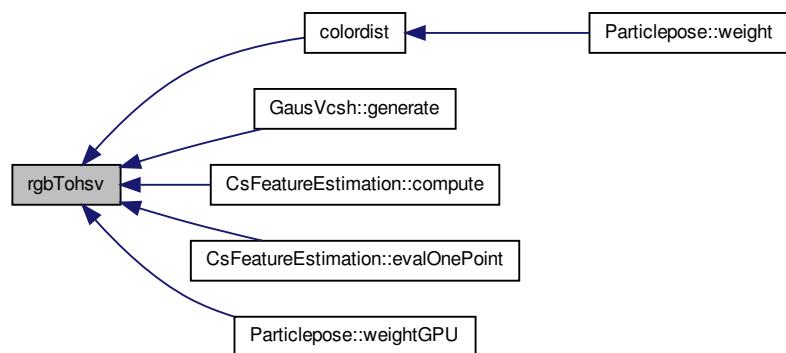


Here is the caller graph for this function:



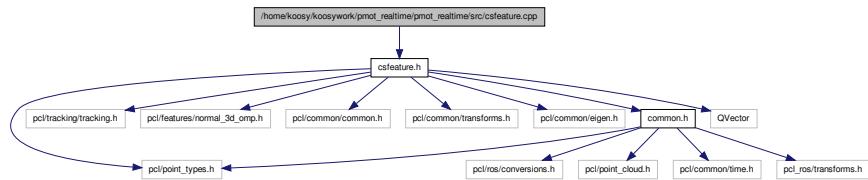
5.1.2.2 void rgbTohsuv (int r, int g, int b, float & hr, float & sr, float & vr) [inline]

Here is the caller graph for this function:



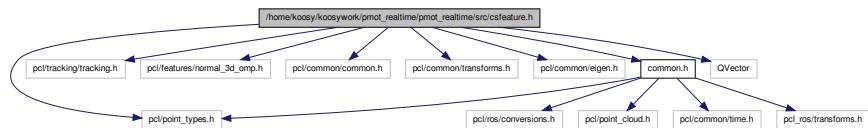
5.2 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.cpp File Reference

```
#include "csfeature.h" Include dependency graph for csfeature.cpp:
```

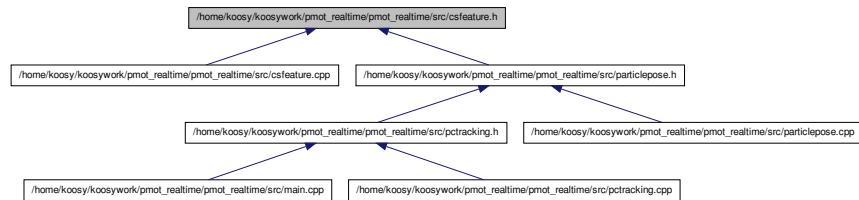


5.3 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csfeature.h File Reference

```
#include <pcl/point_types.h> #include <pcl/tracking/tracking.h>
#include <pcl/features/normal_3d_omp.h> #include <pcl/common/common.h>
#include <pcl/common/transforms.h> #include <pcl/common/eigen.h>
#include "common.h" #include <QVector> Include dependency graph for csfeature.h:
```



This graph shows which files directly or indirectly include this file:

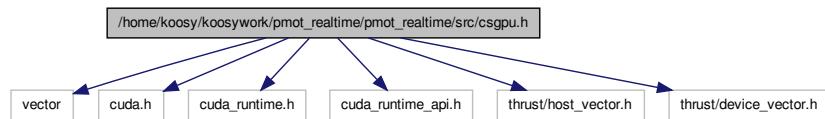


Classes

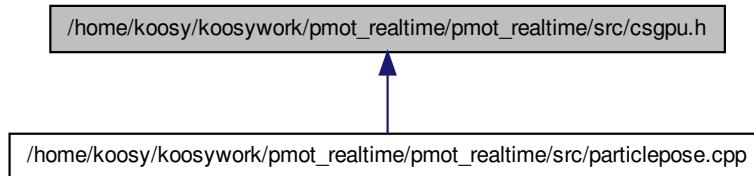
- class [Gaus](#)
- class [Vcsh](#)
- class [GausVcsh](#)
- class [CsFeature](#)
- class [CsFeatureEstimation](#)

5.4 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/csgpu.h File Reference

```
#include <vector>    #include <cuda.h>    #include <cuda_-
runtime.h> #include <cuda_runtime_api.h> #include <thrust/host-
_vector.h> #include <thrust/device_vector.h> Include depen-
dency graph for csgpu.h:
```



This graph shows which files directly or indirectly include this file:

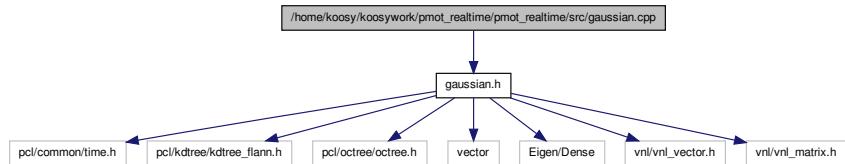


Classes

- class [CSGPU](#)

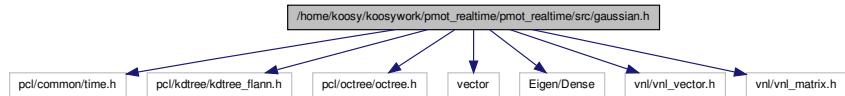
5.5 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussian.cpp File Reference

```
#include "gaussian.h" Include dependency graph for gaussian.cpp:
```

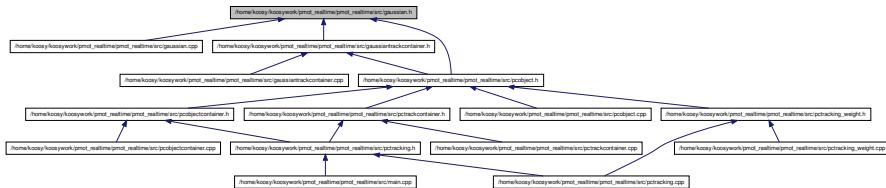


5.6 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussian.h File Reference

```
#include <pcl/common/time.h> #include <pcl/kdtree/kdtree_flann.h> #include <pcl/octree/octree.h> #include <vector> x  
#include <Eigen/Dense>      #include <vnl/vnl_vector.h> x  
#include <vnl/vnl_matrix.h> Include dependency graph for gaussian.h:
```



This graph shows which files directly or indirectly include this file:



Classes

- class Point

Typedefs

- typedef pcl::PointXYZRGB PointT
- typedef pcl::PointCloud< PointT > Cloud
- typedef Cloud::Ptr CloudPtr
- typedef Cloud::ConstPtr CloudConstPtr
- typedef vector< Point > Points

5.6.1 Typedef Documentation

5.6.1.1 typedef pcl::PointCloud<PointT> Cloud

5.6.1.2 typedef Cloud::ConstPtr CloudConstPtr

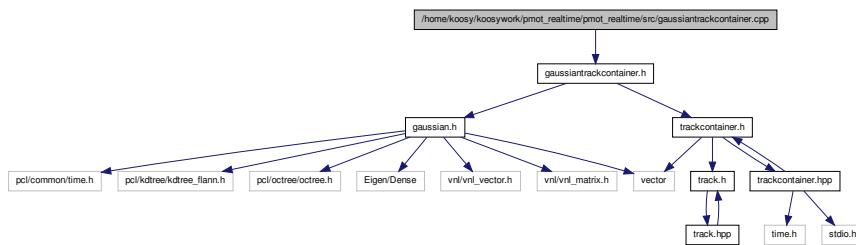
5.6.1.3 typedef Cloud::Ptr CloudPtr

5.6.1.4 typedef vector<Point> Points

5.6.1.5 typedef pcl::PointXYZRGB PointT

5.7 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussiantrackcontainer.cpp File Reference

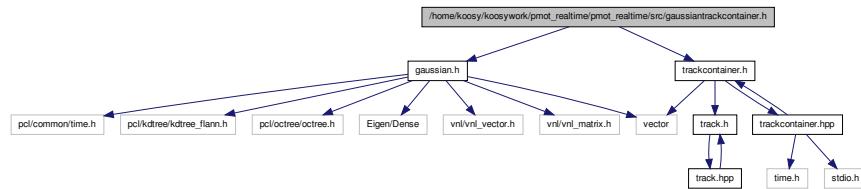
#include "gaussiantrackcontainer.h" Include dependency graph for gaussiantrackcontainer.cpp:



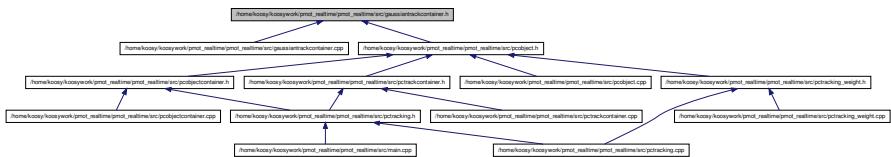
5.8 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/gaussiantrackcontainer.h File Reference

#include "gaussian.h" #include "trackcontainer.h" Include de-

pendency graph for gaussiantrackcontainer.h:



This graph shows which files directly or indirectly include this file:



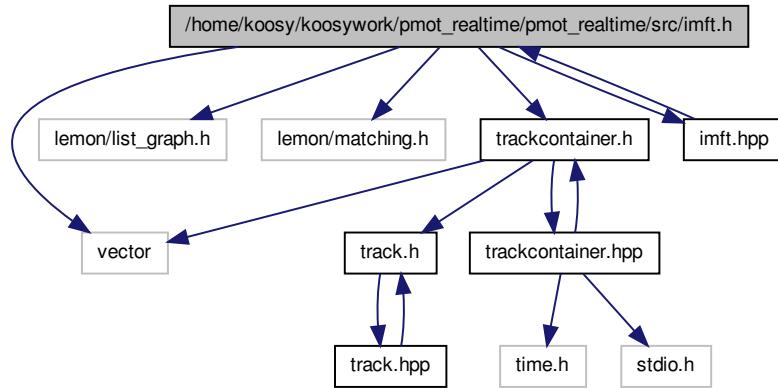
Classes

- class [GaussianTrackContainer](#)

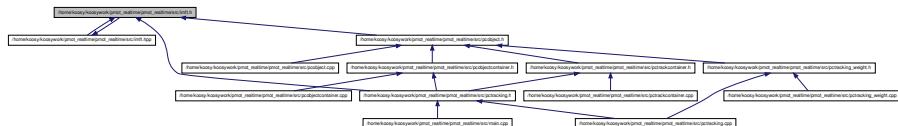
5.9 /home/koosy/koosywork/pmot_realtme/pmot_realtme/src/imft.h File Reference

```
#include <vector> #include <lemon/list_graph.h> #include
<lemon/matching.h> #include "trackcontainer.h" #include
```

"imft.hpp" Include dependency graph for imft.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [IMFT< Object >](#)

Defines

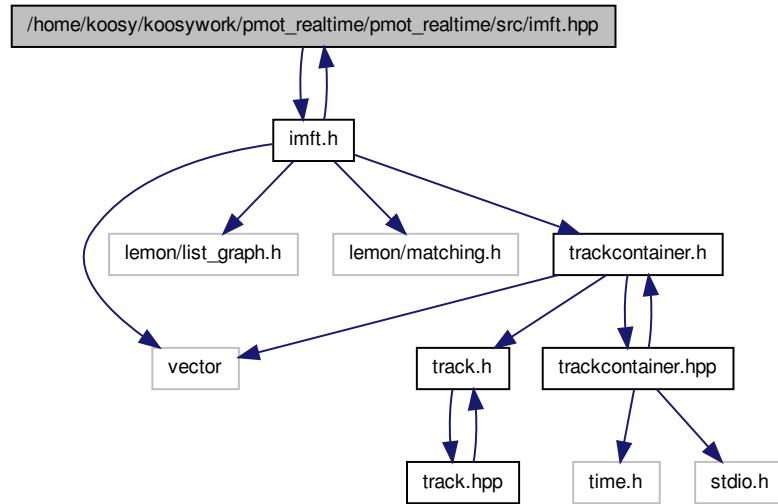
- #define [MAXWEIGHT](#) 100000.

5.9.1 Define Documentation

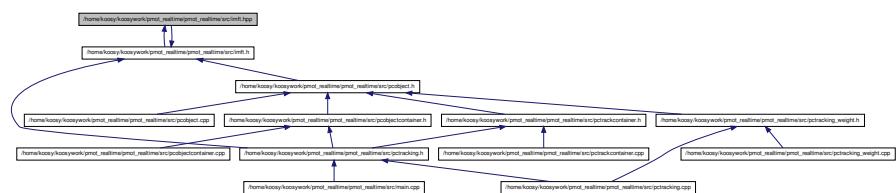
5.9.1.1 #define MAXWEIGHT 100000.

5.10 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/imft.hpp File Reference

#include "imft.h" Include dependency graph for imft.hpp:



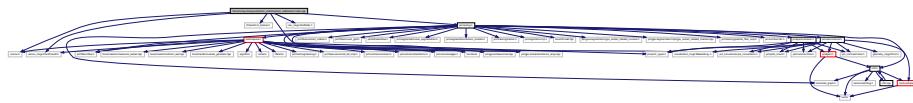
This graph shows which files directly or indirectly include this file:



5.11 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/main.cpp File Reference

```
#include <ros/ros.h> #include <sensor_msgs/PointCloud2.h>
#include <tf/transform_listener.h> #include <visualization_msgs/MarkerArray.h> #include <nav_msgs/GridCells.h> x
```

#include "pctracking.h" Include dependency graph for main.cpp:



Functions

- bool `readParameters ()`
- void `transform (string frame)`
- void `cuttingRange (double range)`
- void `downsampling (double scale)`
- void `planeExtraction (int nPlane)`
- void `workingspace ()`
- void `cb_rgb (const pcl::PCLPointCloud2ConstPtr &input)`
- int `main (int argc, char **argv)`

Variables

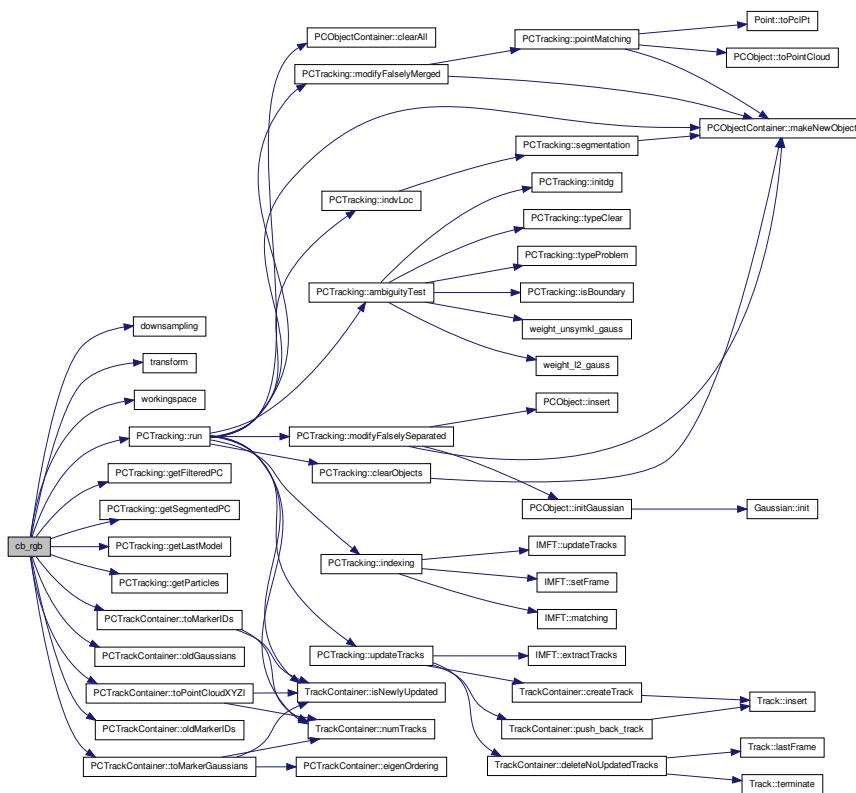
- string `param_frame_id`
- string `param_sub_topic`
- string `param_pub_points_trackID`
- string `param_pub_markers_trackID`
- string `param_pub_markers_gaussians`
- string `param_pub_markers_gmms`
- string `param_pub_markers_edges`
- string `param_pub_points_filtered`
- string `param_pub_points_segmented`
- string `param_pub_gmmreg_model`
- string `param_pub_gmmreg_scene`
- string `param_workspace_topic`
- int `param_3d6d`
- double `param_samplingRatio`
- double `param_simplifyRatio`
- double `param_segmentTolerance`
- double `param_filtering_range`
- double `param_workspace_x`
- double `param_workspace_y`
- double `param_workspace_z`
- double `param_workspace_width`
- double `param_workspace_height`
- double `param_workspace_zheight`
- int `param_bufernum`
- ros::Publisher `pub_filtered`

- ros::Publisher `pub_segmented`
- ros::Publisher `pub_track`
- ros::Publisher `pub_trackID`
- ros::Publisher `pub_model`
- ros::Publisher `pub_scene`
- ros::Publisher `pub_gaussians`
- ros::Publisher `pub_gmms`
- ros::Publisher `pub_edges`
- ros::Publisher `pub_workspace`
- ros::Publisher `pub_lastmodel`
- ros::Publisher `pub_particles`
- tf::TransformListener * `tf_listener`
- nav_msgs::GridCells `workspace`
- double `cont_sampling`
- double `cont_simplify`
- int `sumTotalPoint` = 0
- int `sumTruePoint` = 0
- int `sumErrorPoint` = 0
- double `computationT` = 0
- bool `isdebug` = 1
- PCTracking * `pctracking`
- CloudPtr `pCloud_input`
- CloudPtr `pCloud`
- double `lastT`
- double `nowT`

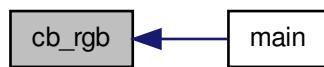
5.11.1 Function Documentation

5.11.1.1 void cb_rgb (const pcl::PCLPointCloud2ConstPtr & input)

Here is the call graph for this function:



Here is the caller graph for this function:



5.11.1.2 void cuttingRange (double range)

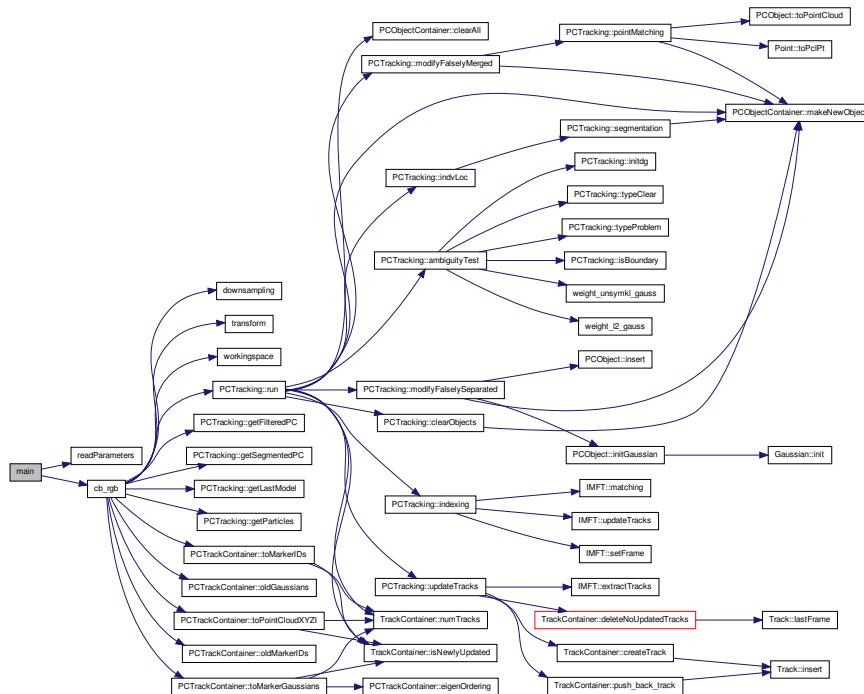
5.11.1.3 void downsampling (double scale)

Here is the caller graph for this function:



5.11.1.4 int main (int argc, char ** argv)

Here is the call graph for this function:



5.11.1.5 void planeExtraction (int nPlane)

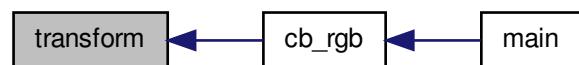
5.11.1.6 bool readParameters ()

Here is the caller graph for this function:



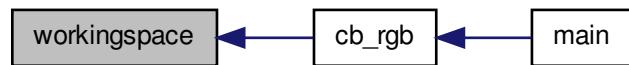
5.11.1.7 void transform (string frame)

Here is the caller graph for this function:



5.11.1.8 void workspace ()

Here is the caller graph for this function:



5.11.2 Variable Documentation

5.11.2.1 double computationT = 0
5.11.2.2 double cont_sampling
5.11.2.3 double cont_simplify
5.11.2.4 bool isdebug = 1
5.11.2.5 double lastT
5.11.2.6 double nowT
5.11.2.7 int param_3d6d
5.11.2.8 int param_buffernum
5.11.2.9 double param_filtering_range
5.11.2.10 string param_frame_id
5.11.2.11 string param_pub_gmmreg_model
5.11.2.12 string param_pub_gmmreg_scene
5.11.2.13 string param_pub_markers_edges
5.11.2.14 string param_pub_markers_gaussians
5.11.2.15 string param_pub_markers_gmms
5.11.2.16 string param_pub_markers_trackID
5.11.2.17 string param_pub_points_filtered
5.11.2.18 string param_pub_points_segmented
5.11.2.19 string param_pub_points_trackID
5.11.2.20 double param_samplingRatio
5.11.2.21 double param_segmentTolerance
5.11.2.22 double param_simplifyRatio
5.11.2.23 string param_sub_topic
5.11.2.24 double param_workspace_height

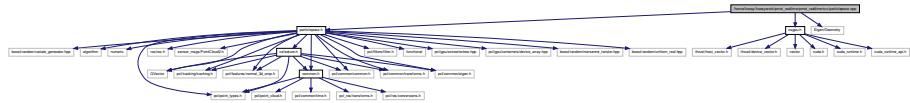
```
5.11.2.25 string param_workspace_topic  
5.11.2.26 double param_workspace_width  
5.11.2.27 double param_workspace_x  
5.11.2.28 double param_workspace_y  
5.11.2.29 double param_workspace_z  
5.11.2.30 double param_workspace_zheight  
5.11.2.31 CloudPtr pCloud  
5.11.2.32 CloudPtr pCloud_input  
5.11.2.33 PCTracking* pctracking  
5.11.2.34 ros::Publisher pub_edges  
5.11.2.35 ros::Publisher pub_filtered  
5.11.2.36 ros::Publisher pub_gaussians  
5.11.2.37 ros::Publisher pub_gmms  
5.11.2.38 ros::Publisher pub_lastmodel  
5.11.2.39 ros::Publisher pub_model  
5.11.2.40 ros::Publisher pub_particles  
5.11.2.41 ros::Publisher pub_scene  
5.11.2.42 ros::Publisher pub_segmented  
5.11.2.43 ros::Publisher pub_track  
5.11.2.44 ros::Publisher pub_trackID  
5.11.2.45 ros::Publisher pub_workspace  
5.11.2.46 int sumErrorPoint = 0  
5.11.2.47 int sumTotalPoint = 0  
5.11.2.48 int sumTruePoint = 0
```

5.11.2.49 `tf::TransformListener* tf_listener`

5.11.2.50 `nav_msgs::GridCells workspace`

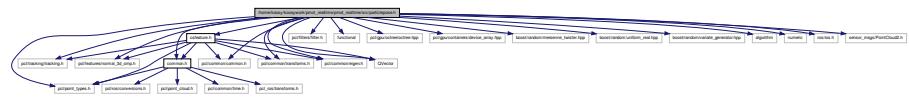
5.12 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/particlepose.cpp File Reference

```
#include "particlepose.h" #include "csgpu.h" #include <- Eigen/Geometry> Include dependency graph for particlepose.cpp:
```

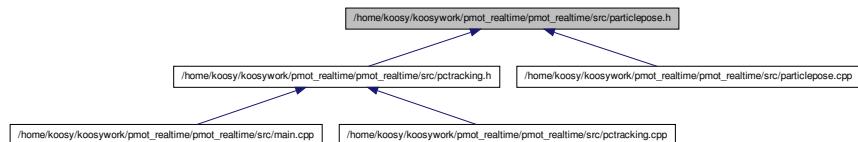


5.13 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/particlepose.h File Reference

```
#include <pcl/point_types.h> #include <pcl/tracking/tracking.h>
#include <pcl/features/normal_3d_omp.h> #include <pcl/common/common.h>
#include <pcl/common/transforms.h> #include <pcl/common/eigen.h>
#include <pcl/filters/filter.h> #include <functional> x
#include <pcl/gpu/octree/octree.hpp> #include <pcl/gpu/containers/device_array.hpp>
#include <boost/random/mersenne_twister.hpp> #include <boost/random/uniform_real.hpp> #include
<boost/random/variate_generator.hpp> #include <algorithm> x
#include <numeric> #include <QVector> #include <ros/ros.h>
#include <sensor_msgs/PointCloud2.h> #include "common.h" #include "csfeature.h" Include dependency graph for particlepose.h:
```



This graph shows which files directly or indirectly include this file:



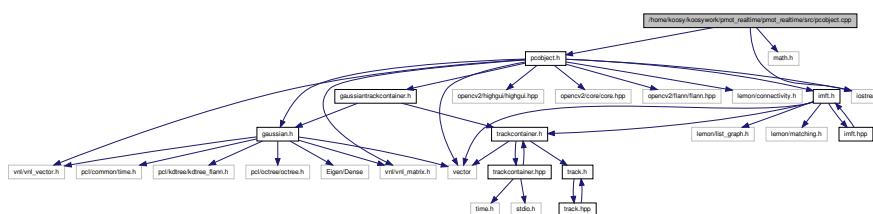
Classes

- class Particlepose

5.14 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pcobject.cpp

File Reference

```
#include "pcobject.h" #include <math.h> #include <iostream> x  
Include dependency graph for pcobject.cpp:
```



Defines

- #define pi 3.141592

Variables

- `bool isDebug = 1`

5.14.1 Define Documentation

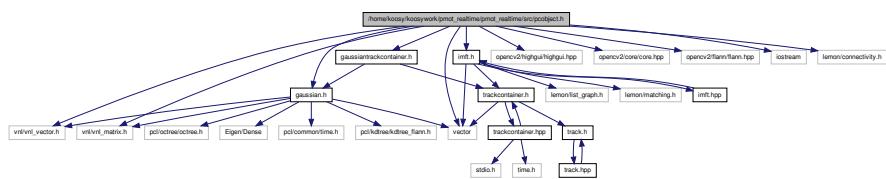
5.14.1.1 #define pi 3.141592

5.14.2 Variable Documentation

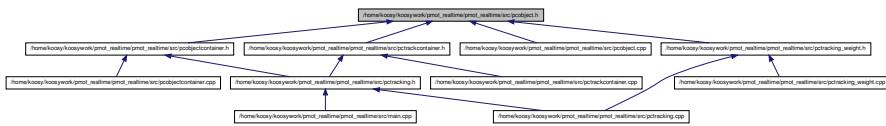
5.14.2.1 bool isDebug = 1

5.15 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pcobject.h File Reference

```
#include "vector" #include "opencv2/highgui/highgui.hpp"
#include "opencv2/core/core.hpp" #include "opencv2/flann/flann.hpp"
#include <vnl/vnl_vector.h> #include <vnl/vnl_matrix.h>
#include <iostream> #include <lemon/connectivity.h>
#include "gaussian.h" #include "imft.h" #include "gaussiantrackcontainer.h"
#include dependency graph for pcobject.h:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [PCObject](#)
- struct [PCObject::Edge](#)
- struct [PCObject::Edge_spatial](#)

Defines

- `#define MIN_WEIGHT_RATIO 5`

Enumerations

- enum [SIMPLE](#) { [SIMPLE_HCKL](#), [SIMPLE_HCL2](#), [SIMPLE_FA](#) }
- enum [STATE](#) { [NOGMM](#), [PRIORGMM](#), [POSTGMM](#) }

5.15.1 Define Documentation

5.15.1.1 #define MIN_WEIGHT_RATIO 5

5.15.2 Enumeration Type Documentation

5.15.2.1 enum SIMPLE

Enumerator:

SIMPLE_HCKL

SIMPLE_HCL2

SIMPLE_FA

5.15.2.2 enum STATE

Enumerator:

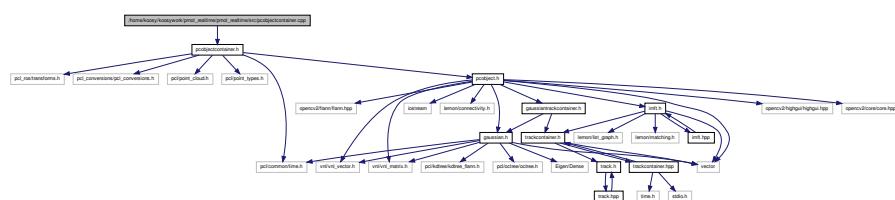
NOGMM

PRIORGMM

POSTGMM

5.16 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pcobjectcontainer.cpp
File Reference

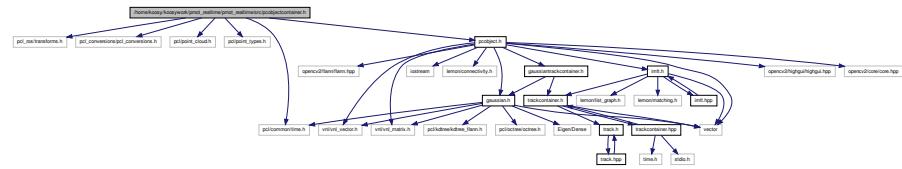
#include "pcobjectcontainer.h" Include dependency graph for pcobjectcontainer.-
cpp:



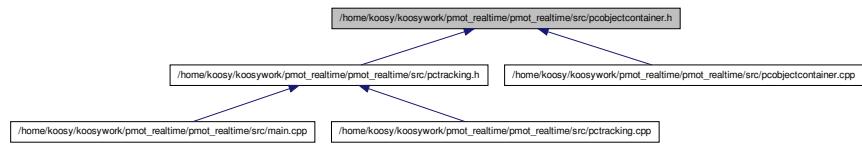
5.17 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pcobjectcontainer.h
File Reference

#include <pcl_ros/transforms.h> #include <pcl_conversions/pcl-
_conversions.h> #include <pcl/point_cloud.h> #include

```
<pcl/point_types.h> #include <pcl/common/time.h> #include  
"pcobject.h" Include dependency graph for pcobjectcontainer.h:
```



This graph shows which files directly or indirectly include this file:

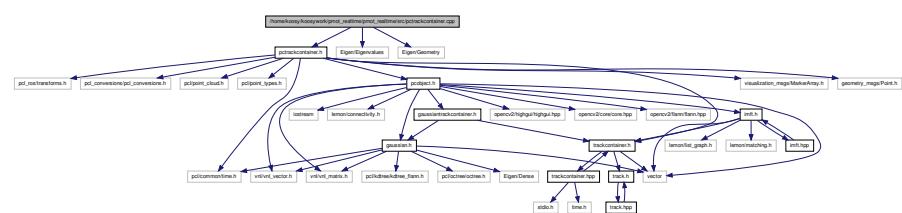


Classes

- class `PCObjectContainer`

5.18 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctrackcontainer.cpp

```
#include "pctrackcontainer.h" #include <Eigen/Eigenvalues>  
#include <Eigen/Geometry> Include dependency graph for pctrackcontainer.-  
cpp:
```



5.19

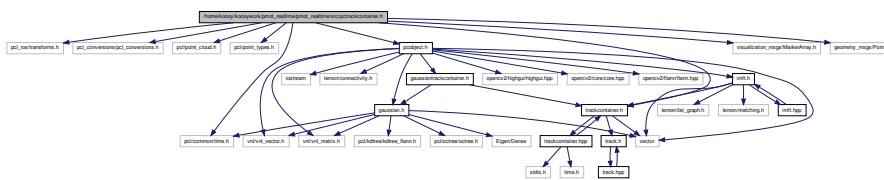
/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctrackcontainer.h

File Reference

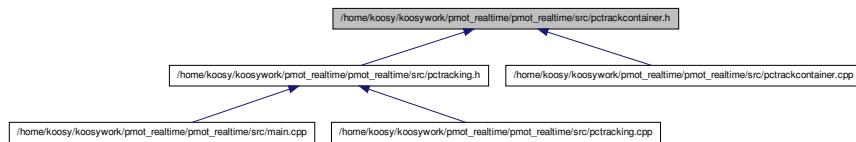
5.19 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctrackcontainer.h 109

File Reference

```
#include <pcl_ros/transforms.h> #include <pcl_conversions/pcl-
_conversions.h> #include <pcl/point_cloud.h> #include
<pcl/point_types.h> #include <pcl/common/time.h> #include
<pcobject.h> #include "trackcontainer.h" #include <visualization-
_msgs/MarkerArray.h> #include <geometry_msgs/Point.h> ×
Include dependency graph for pctrackcontainer.h:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [PCTrackContainer](#)

Defines

- `#define PI 3.141592`

5.19.1 Define Documentation

5.19.1.1 `#define PI 3.141592`

5.20 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking.cpp File Reference

```
#include "pctracking.h" #include "pctracking_weight.h" x
#include <iostream> #include <boost/thread.hpp> #include
<boost/date_time.hpp> Include dependency graph for pctracking.cpp:
```



Variables

- bool `isDebug_comp` = 1

5.20.1 Variable Documentation

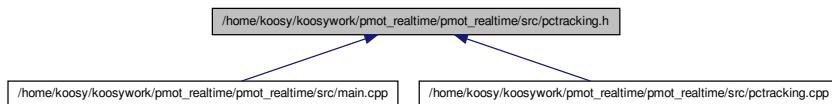
5.20.1.1 bool `isDebug_comp` = 1

5.21 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking.h File Reference

```
#include "pcobjectcontainer.h" #include "pctrackcontainer.-
h" #include "imft.h" #include "particlepose.h" #include
<pcl/filters/extract_indices.h> #include <pcl/filters/voxel-
_grid.h> #include <pcl/kdtree/kdtree.h> #include <pcl/segmentation/sac-
_segmentation.h> #include <pcl/segmentation/extract_-
clusters.h> #include <pcl/tracking/tracker.h> #include
<pcl/filters/filter.h> #include <pcl/registration/icp.-
h> #include <boost/thread.hpp> #include <pcl/gpu/octree/octree.-
hpp> #include <pcl/gpu/containers/device_array.hpp>x
#include <pcl/gpu/segmentation/gpu_extract_labeled_clusters.-
h> #include <pcl/gpu/segmentation/impl/gpu_extract_labeled-
_clusters.hpp> #include <pcl/tracking/particle_filter_-
omp.h> #include <pcl/surface/mls.h> #include <lemon/list-
_graph.h> Include dependency graph for pctracking.h:
```



This graph shows which files directly or indirectly include this file:



Classes

- struct [Scene](#)
- struct [TrackPoint](#)
- struct [Frame](#)
- struct [ObjectNode](#)
- struct [Weight](#)
- struct [FalseObjects](#)
- struct [BoundingBox](#)
- class [PCTracking](#)

Defines

- #define [SQR\(X\)](#) ((X)*(X))
- #define [pi](#) 3.141592
- #define [L2CUT](#) -1
- #define [FPS_CALC\(_WHAT_\)](#)

Typedefs

- typedef vector<[PCObject](#) * > [VecObjectPtr](#)
- typedef vector<[PCObject](#) > [VecObject](#)

Enumerations

- enum [Type](#) { [UNDEFINED](#), [PROBLEM](#), [CLEAR](#), [PARENT](#), [CHILD](#), [DEFINED](#) }
- enum [Time](#) { [PREVIOUS](#), [CURRENT](#) }

5.21.1 Define Documentation

5.21.1.1 #define [FPS_CALC\(_WHAT_ \)](#)

Value:

```

do \
{ \
    static unsigned count = 0; \
    static double last = pcl::getTime (); \
    double now = pcl::getTime (); \
    ++count; \
    if (now - last >= 1.0) \
{ \
    ROS_INFO("FPS of %s: %f", __PRETTY_FUNCTION__, double(count)/double(now - last)); \
    count = 0; \
    last = now; \
} \
}while(false)

```

5.21.1.2 #define L2CUT -1

5.21.1.3 #define pi 3.141592

5.21.1.4 #define SQR(X) ((X)*(X))

5.21.2 Typedef Documentation

5.21.2.1 typedef vector<PCObject> VecObject

5.21.2.2 typedef vector<PCObject*> VecObjectPtr

5.21.3 Enumeration Type Documentation

5.21.3.1 enum Time

Enumerator:

PREVIOUS

CURRENT

5.21.3.2 enum Type

Enumerator:

UNDEFINED

PROBLEM

CLEAR

PARENT

CHILD

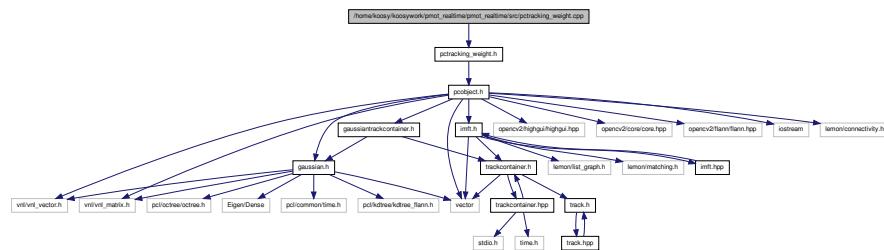
DEFINED

5.22 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking_weight.cpp File

Reference

5.22 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking_weight.cpp File Reference ¹¹³

```
#include "pctracking_weight.h" Include dependency graph for pctracking-weight.cpp:
```



Functions

- double `weight_l2_rev (PCObject &o1, PCObject &o2)`
- double `weight_loc (PCObject &o1, PCObject &o2)`
- double `weight_unsymkl_gauss (PCObject &o1, PCObject &o2)`
- double `weight_symkl_gauss (PCObject &o1, PCObject &o2)`
- double `weight_l2_gauss (PCObject &o1, PCObject &o2)`
- double `weight_gaussian_predictive_rev (Gaussian &g1, Gaussian &g2)`

5.22.1 Function Documentation

5.22.1.1 double `weight_gaussian_predictive_rev (Gaussian & g1, Gaussian & g2)`

5.22.1.2 double `weight_l2_gauss (PCObject & o1, PCObject & o2)`

Here is the caller graph for this function:



5.22.1.3 double `weight_l2_rev (PCObject & o1, PCObject & o2)`

5.22.1.4 double `weight_loc (PCObject & o1, PCObject & o2)`

5.22.1.5 double weight_symkl_gauss (PCObject & o1, PCObject & o2)

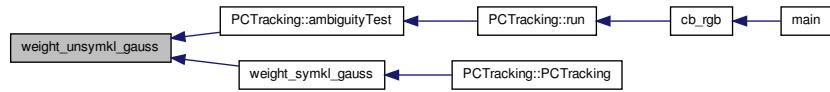
Here is the call graph for this function:



Here is the caller graph for this function:

**5.22.1.6 double weight_unsymkl_gauss (PCObject & o1, PCObject & o2)**

Here is the caller graph for this function:



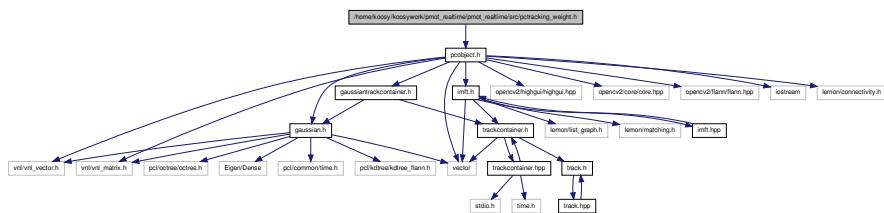
5.23

/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking_weight.h

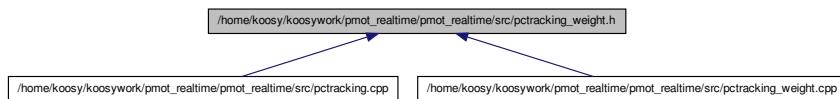
File Reference

5.23 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/pctracking_weight.h File Reference 115

#include "pcobject.h" Include dependency graph for pctracking_weight.h:



This graph shows which files directly or indirectly include this file:



Defines

- #define SQR(X) ((X)*(X))
- #define pi 3.141592

Functions

- double weight_l2_rev (PCObject &o1, PCObject &o2)
- double weight_loc (PCObject &o1, PCObject &o2)
- double weight_symkl_gauss (PCObject &o1, PCObject &o2)
- double weight_unsymkl_gauss (PCObject &o1, PCObject &o2)
- double weight_l2_gauss (PCObject &o1, PCObject &o2)
- double weight_gaussian_predictive_rev (Gaussian &g1, Gaussian &g2)

5.23.1 Define Documentation

5.23.1.1 #define pi 3.141592

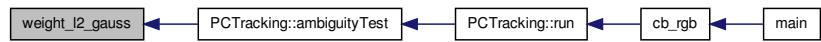
5.23.1.2 #define SQR(X) ((X)*(X))

5.23.2 Function Documentation

5.23.2.1 double weight_gaussian_predictive_rev (Gaussian & $g1$, Gaussian & $g2$)

5.23.2.2 double weight_l2_gauss (PCObject & $o1$, PCObject & $o2$)

Here is the caller graph for this function:



5.23.2.3 double weight_l2_rev (PCObject & $o1$, PCObject & $o2$)

5.23.2.4 double weight_loc (PCObject & $o1$, PCObject & $o2$)

5.23.2.5 double weight_symkl_gauss (PCObject & $o1$, PCObject & $o2$)

Here is the call graph for this function:

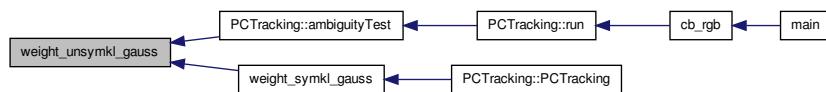


Here is the caller graph for this function:



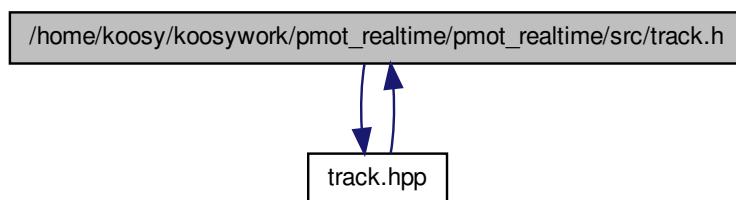
5.23.2.6 double weight_unsymkl_gauss (PCObject & o1, PCObject & o2)

Here is the caller graph for this function:

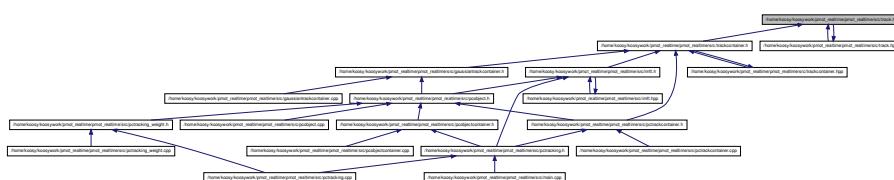


5.24 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/track.h File Reference

#include "track.hpp" Include dependency graph for track.h:



This graph shows which files directly or indirectly include this file:



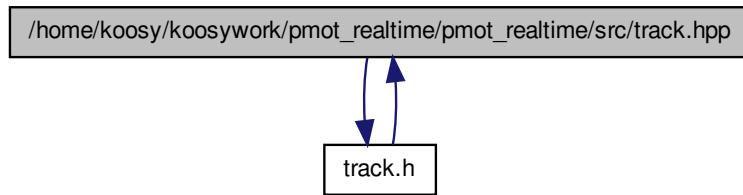
Classes

- class [Track< Object >](#)
- struct [Track< Object >::Frame](#)

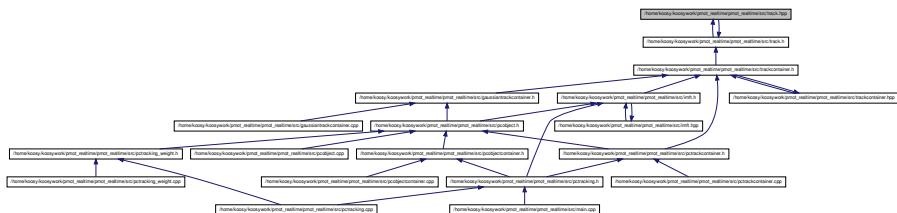
- struct `Track< Object >::Node`
- struct `Track< Object >::V`

5.25 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/track.hpp File Reference

#include "track.h" Include dependency graph for track.hpp:



This graph shows which files directly or indirectly include this file:



5.26 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/trackcontainer.h File Reference

#include "track.hpp" #include <vector> #include "trackcontainer.-

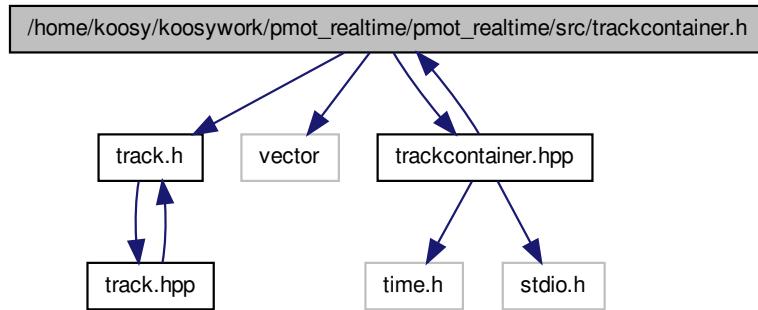
5.27

/home/koosy/koosywork/pmot_realtime/pmot_realtime/src/trackcontainer.hpp

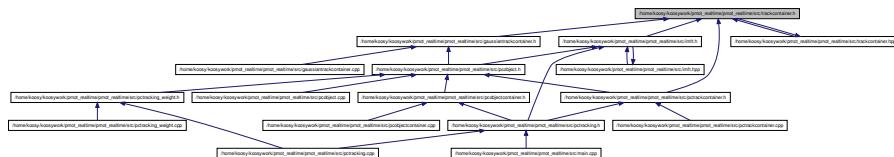
File Reference

119

hpp" Include dependency graph for trackcontainer.h:



This graph shows which files directly or indirectly include this file:



Classes

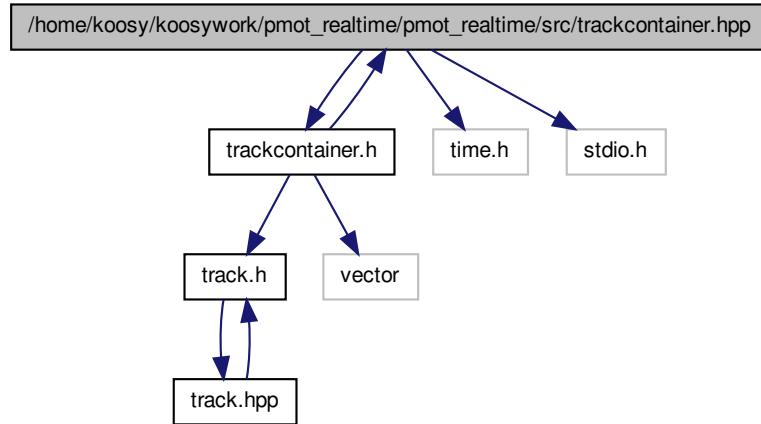
- class [TrackContainer< Object >](#)

5.27 /home/koosy/koosywork/pmot_realtime/pmot_realtime/src/trackcontainer.hpp

File Reference

```
#include "trackcontainer.h" #include <time.h> #include
```

<stdio.h> Include dependency graph for trackcontainer.hpp:



This graph shows which files directly or indirectly include this file:

