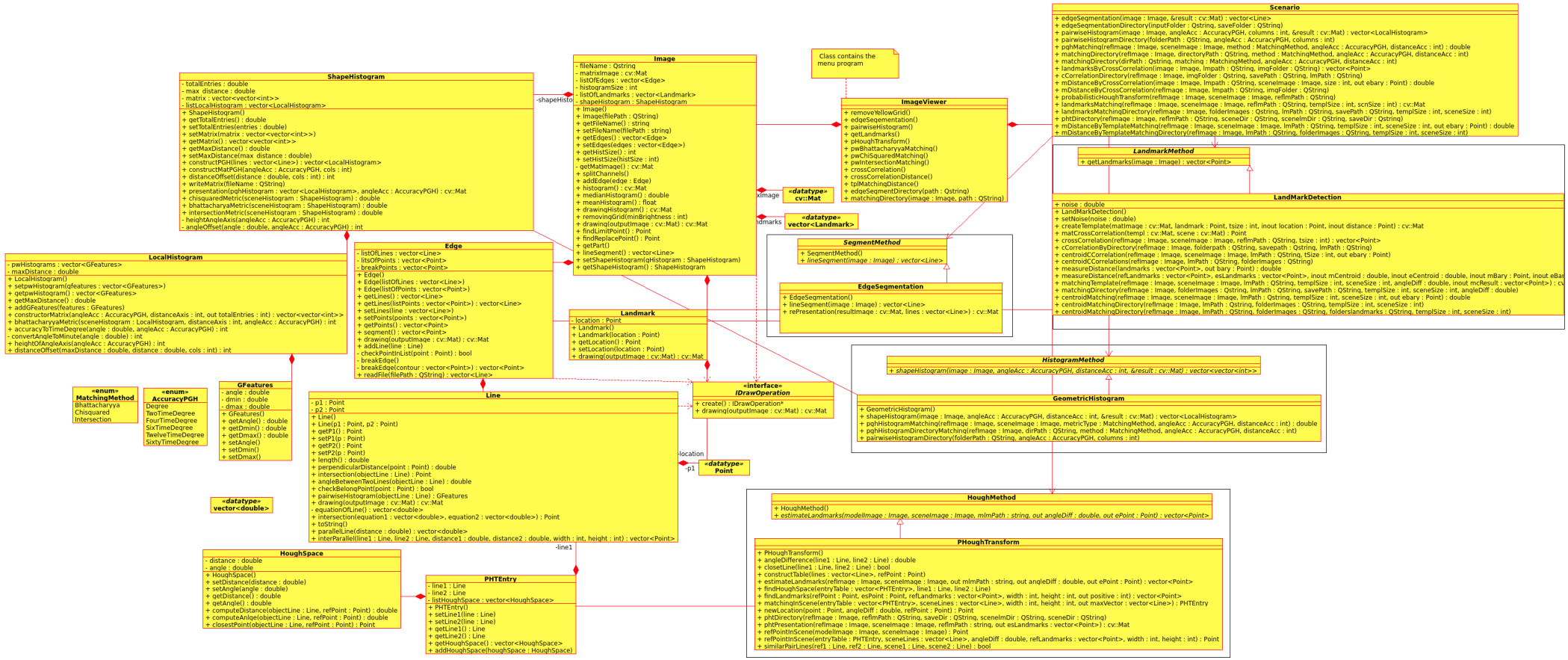
Scenario
+ edgeSegmentation(image : Image, &result : cv::Mat) : vector<Line>
+ edgeSegmentationDirectory(inputFolder : Qstring, saveFolder : QString)
+ pairwiseHistogram(image : Image, angleAcc : AccuracyPGH, columns : int, &result : cv::Mat) : vector<LocalHistogram>
+ pairwiseHistogramDirectory(folderPath : QString, angleAcc : AccuracyPGH, columns : int)
+ pghMatching(refImage : Image, sceneImage : Image, method : MatchingMethod, angleAcc : AccuracyPGH, distanceAcc : int) : double
+ matchingDirectory(refImage : Image, directoryPath : QString, method : MatchingMethod, angleAcc : AccuracyPGH, distanceAcc : int)
+ matchingDirectory(dirPath : Qstring, matching : MatchingMethod, angleAcc : AccuracyPGH, distanceAcc : int)
+ landmarksByCrossCorrelation(image : Image, lmpath : QString, imgFolder : QString) : vector<Point>
+ cCorrelationDirectory(refImage : Image, imgFolder : Qstring, savePath : QString, lmPath : QString)
+ mDistanceByCrossCorrelation(image : Image, lmpath : QString, sceneImage : Image, size : int, out ebary : Point) : double
+ mDistanceByCrossCorrelation(refImage : Image, lmpath : QString, imgFolder : QString)
+ probabilisticHoughTransform(refImage : Image, sceneImage : Image, refImPath : QstringQ)
+ landmarksMatching(refImage : Image, sceneImage : Image, refImPath : QString, templSize : int, scnSize : int) : cv::Mat
+ landmarksMatchingDirectory(refImage : Image, folderImages : Qstring, lmPath : QString, savePath : QString, templSize : int, sceneSize : int)
+ phtDirectory(refImage : Image, refImPath : QString, sceneDir : QString, sceneImDir : QString, saveDir : Qstring)
+ mDistanceByTemplateMatching(refImage : Image, sceneImage : Image, lmPath : QString, templSize : int, sceneSize : int, out ebary : Point) : double
+ mDistanceByTemplateMatchingDirectory(refImage : Image, lmPath : QString, folderImages : QString, templSize : int, sceneSize : int)
LandmarkMethod
+ getLandmarks(image : Image) : vector<Point>
LandMarkDetection
+ noise : double
+ LandMarkDetection()
+ setNoise(noise : double)
+ createTemplate(matImage : cv::Mat, landmark : Point, tsize : int, inout location : Point, inout distance : Point) : cv::Mat
+ matCrossCorrelation(templ : cv::Mat, scene : cv::Mat) : Point
+ crossCorrelation(refImage : Image, sceneImage : Image, refImPath : QString, tsize : int) : vector<Point>
+ cCorrelationByDirectory(refImage : Image, folderpath : QString, savepath : Qstring, lmPath : QString)
+ centroidCCorrelation(refImage : Image, sceneImage : Image, lmPath : QString, tSize : int, out ebary : Point)
+ centroidCCorrelations(refImage : Image, lmPath : QString, folderImages : QString)
+ measureDistance(landmarks : vector<Point>, out bary : Point) : double
+ measureDistance(refLandmarks : vector<Point>, esLandmarks : vector<Point>, inout mCentroid : double, inout eCentroid : double, inout mBary : Point, inout eBar
+ matchingTemplate(refImage : Image, sceneImage : Image, lmPath : QString, templSize : int, sceneSize : int, angleDiff : double, inout mcResult : vector<Point>) : cv
+ matchingDirectory(refImage : Image, folderImages : Qstring, lmPath : QString, savePath : QString, templSize : int, sceneSize : int, angleDiff : double)
+ centroidMatching(refImage : Image, sceneImage : Image, lmPath : Qstring, templSize : int, sceneSize : int, out ebary : Point) : double
+ centroidMatchingDirectory(refImage : Image, lmPath : Qstring, folderImages : QString, templSize : int, sceneSize : int)
+ centroidMatchingDirectory(refImage : Image, lmPath : QString, folderImages : QString, folderslandmarks : QString, templSize : int, sceneSize : int)
Class contains the menu program
ImageViewer
+ removeYellowGrid()
+ edgeSeqementation()
+ pairwiseHistogram()
+ getLandmarks()
+ pHoughTransform()
+ pwBhattacharyyaMatching()
+ pwChiSquaredMatching()
+ pwIntersectionMatching()
+ crossCorrelation()
+ crossCorrelationDistance()
+ tplMatchingDistance()
+ edgeSegmentDirectory(path : Qstring)
+ matchingDirectory(image : Image, path : QString)
Image
- fileName : Qstring
- matrixImage : cv::Mat
- listOfEdges : vector<Edge>
- histogramSize : int
- listOfLandmarks : vector<Landmark>
- shapeHistogram : ShapeHistogram
+ Image()
+ Image(filePath : QString)
+ getFileName() : string
+ setFileName(filePath : string)
+ getEdges() : vector<Edge>
+ setEdges(edges : vector<Edge>)
+ getHistSize() : int
+ setHistSize(histSize : int)
- getMatImage() : cv::Mat
+ splitChannels()
+ addEdge(edge : Edge)
+ histogram() : cv::Mat
+ medianHistogram() : double
+ meanHistogram() : float
+ drawingHistogram() : cv::Mat
+ removingGrid(minBrightness : int)
+ drawing(outputImage : cv::Mat) : cv::Mat
+ findLimitPoint() : Point
+ findReplacePoint() : Point
+ getPart()
+ lineSegment() : vector<Line>
+ setShapeHistogram(qHistogram : ShapeHistogram)
+ getShapeHistogram() : ShapeHistogram
«datatype»
cv::Mat
«datatype»
vector<Landmark>
ShapeHistogram
- totalEntries : double
- max_distance : double
- matrix : vector<vector<int>>
- listLocalHistogram : vector<LocalHistogram>
+ ShapeHistogram()
+ getTotalEntries() : double
+ setTotalEntries(entries : double)
+ setMatrix(matrix : vector<vector<int>>)
+ getMatrix() : vector<vector<int>>
+ getMaxDistance() : double
+ setMaxDistance(max_distance : double)
+ constructPGH(lines : vector<Line>) : vector<LocalHistogram>
+ constructMatPGH(angleAcc : AccuracyPGH, cols : int)
+ distanceOffset(distance : double, cols : int) : int
+ writeMatrix(fileName : QString)
+ presentation(pghHistogram : vector<LocalHistogram>, angleAcc : AccuracyPGH) : cv::Mat
+ chisquaredMetric(sceneHistogram : ShapeHistogram) : double
+ bhattacharyaMetric(sceneHistogram : ShapeHistogram) : double
+ intersectionMetric(sceneHistogram : ShapeHistogram) : double
- heightAngleAxis(angleAcc : AccuracyPGH) : int
- angleOffset(angle : double, angleAcc : AccuracyPGH) : int
-shapeHisto
LocalHistogram
- pwHistograms : vector<GFeatures>
- maxDistance : double
+ LocalHistogram()
+ setpwHistogram(gfeatures : vector<GFeatures>)
+ getpwHistogram() : vector<GFeatures>
+ getMaxDistance() : double
+ addGFeatures(features : GFeatures)
+ constructorMatrix(angleAcc : AccuracyPGH, distanceAxis : int, out totalEntries : int) : vector<vector<int>>
+ bhattacharyaMetric(sceneHistogram : LocalHistogram, distanceAxis : int, angleAcc : AccuracyPGH) : int
+ accuracyToTimeDegree(angle : double, angleAcc : AccuracyPGH) : int
- convertAngleToMinute(angle : double) : int
+ heightOfAngleAxis(angleAcc : AccuracyPGH) : int
+ distanceOffset(maxDistance : double, distance : double, cols : int) : int
Edge
- listOfLines : vector<Line>
- litsOfPoints : vector<Point>
- breakPoints : vector<Point>
+ Edge()
+ Edge(listOfLines : vector<Line>)
+ Edge(listOfPoints : vector<Point>)
+ getLines() : vector<Line>
+ getLines(listPoints : vector<Point>) : vector<Line>
+ setLines(line : vector<Line>)
+ setPoints(points : vector<Point>)
+ getPoints() : vector<Point>
+ segment() : vector<Point>
+ drawing(outputImage : cv::Mat) : cv::Mat
+ addLine(line : Line)
- checkPointInList(point : Point) : bool
- breakEdge()
- breakEdge(contour : vector<Point>) : vector<Point>
+ readFile(filePath : QString) : vector<Line>
Landmark
- location : Point
+ Landmark()
+ Landmark(location : Point)
+ getLocation() : Point
+ setLocation(location : Point)
+ drawing(outputImage : cv::Mat) : cv::Mat
SegmentMethod
+ SegmentMethod()
+ lineSegment(image : Image) : vector<Line>
EdgeSegmentation
+ EdgeSegmentation()
+ lineSegment(image : Image) : vector<Line>
+ rePresentation(resultImage : cv::Mat, lines : vector<Line>) : cv::Mat
«enum»
MatchingMethod
Bhattacharyya
Chisquared
Intersection
«enum»
AccuracyPGH
Degree
TwoTimeDegree
FourTimeDegree
SixTimeDegree
TwelveTimeDegree
SixtyTimeDegree
GFeatures
- angle : double
- dmin : double
- dmax : double
+ GFeatures()
+ getAngle() : double
+ getDmin() : double
+ getDmax() : double
+ setAngle()
+ setDmin()
+ setDmax()
«datatype»
vector<double>
Line
- p1 : Point
- p2 : Point
+ Line()
+ Line(p1 : Point, p2 : Point)
+ getP1() : Point
+ setP1(p : Point)
+ getP2() : Point
+ setP2(p : Point)
+ length() : double
+ perpendicularDistance(point : Point) : double
+ intersection(objectLine : Line) : Point
+ angleBetweenTwoLines(objectLine : Line) : double
+ checkBelongPoint(point : Point) : bool
+ pairwiseHistogram(objectLine : Line) : GFeatures
+ drawing(outputImage : cv::Mat) : cv::Mat
- equationOfLine() : vector<double>
+ intersection(equation1 : vector<double>, equation2 : vector<double>) : Point
+ toString()
+ parallelLine(distance : double) : vector<double>
+ interParallel(line1 : Line, line2 : Line, distance1 : double, distance2 : double, width : int, height : int) : vector<Point>
«interface»
IDrawOperation
+ create() : IDrawOperation*
+ drawing(outputImage : cv::Mat) : cv::Mat
-location
-p1
«datatype»
Point
-line1
HistogramMethod
+ shapeHistogram(image : Image, angleAcc : AccuracyPGH, distanceAcc : int, &result : cv::Mat) : vector<vector<int>>
GeometricHistogram
+ GeometricHistogram()
+ shapeHistogram(image : Image, angleAcc : AccuracyPGH, distanceAcc : int, &result : cv::Mat) : vector<LocalHistogram>
+ pghHistogramMatching(refImage : Image, sceneImage : Image, metricType : MatchingMethod, angleAcc : AccuracyPGH, distanceAcc : int) : double
+ pghHistogramDirectoryMatching(refImage : Image, dirPath : QString, method : MatchingMethod, angleAcc : AccuracyPGH, distanceAcc : int)
+ pairwiseHistogramDirectory(folderPath : QString, angleAcc : AccuracyPGH, columns : int)
HoughMethod
+ HoughMethod()
+ estimateLandmarks(modelImage : Image, sceneImage : Image, mlmPath : string, out angleDiff : double, out ePoint : Point) : vector<Point>
PHoughTransform
+ PHoughTransform()
+ angleDifference(line1 : Line, line2 : Line) : double
+ closetLine(line1 : Line, line2 : Line) : bool
+ constructTable(lines : vector<Line>, refPoint : Point)
+ estimateLandmarks(refImage : Image, sceneImage : Image, out mlmPath : string, out angleDiff : double, out ePoint : Point) : vector<Point>
+ findHoughSpace(entryTable : vector<PHTEntry>, line1 : Line, line2 : Line)
+ findLandmarks(refPoint : Point, esPoint : Point, refLandmarks : vector<Point>, width : int, height : int, out positive : int) : vector<Point>
+ matchingInScene(entryTable : vector<PHTEntry>, sceneLines : vector<Line>, width : int, height : int, out maxVector : vector<Line>) : PHTEntry
+ newLocation(point : Point, angleDiff : double, refPoint : Point) : Point
+ phtDirectory(refImage : Image, refImPath : QString, saveDir : QString, sceneImDir : QString, sceneDir : QString)
+ phtPresentation(refImage : Image, sceneImage : Image, refImPath : string, out esLandmarks : vector<Point>) : cv::Mat
+ refPointInScene(modelImage : Image, sceneImage : Image) : Point
+ refPointInScene(entryTable : PHTEntry, sceneLines : vector<Line>, angleDiff : double, refLandmarks : vector<Point>, width : int, height : int) : Point
+ similarPairLines(ref1 : Line, ref2 : Line, scene1 : Line, scene2 : Line) : bool
HoughSpace
- distance : double
- angle : double
+ HoughSpace()
+ setDistance(distance : double)
+ setAngle(angle : double)
+ getDistance() : double
+ getAngle() : double
+ computeDistance(objectLine : Line, refPoint : Point) : double
+ computeAnlge(objectLine : Line, refPoint : Point) : double
+ closestPoint(objectLine : Line, refPoint : Point) : Point
PHTEntry
- line1 : Line
- line2 : Line
- listHoughSpace : vector<HoughSpace>
+ PHTEntry()
+ setLine1(line : Line)
+ setLine2(line : Line)
+ getLine1() : Line
+ getLine2() : Line
+ getHoughSpace() : vector<HoughSpace>
+ addHoughSpace(houghSpace : HoughSpace)