1. FileManager类

成员函数：

void getFilenamesByPathAndExtension(string path, string extension)

* 读取path路径下的所有以extension为后缀的文件

string getPathFromUser()

* 弹出对话框让用户选择路径

void getFilenameWithExtensionFromUser(string extension)

* 弹出对话框让用户选择以extension为后缀的文件

void getFilenamesByList()

* 通过List文件读取filenames

1. MappingParameters

double scale, rotation, translationX, translationY;

* 分别表示缩放大小，旋转角度，X方向位移，Y方向位移

void caculateNewCoordinates(double x, double y, double &resX, double &resY);

* 根据参数计算新坐标

void getMappingMatrix(cv::Mat &mappingMatrix);

* 返回映射(2, 2)的矩阵

void getTranslationMatrix(cv::Mat &translationMatrix);

* 返回一个(n, 2)的矩阵，第一列是tx，第二列是ty

1. AlignShape类

成员函数：

MappingParameters findBestMapping(Mat Ax, Mat Ay, Mat Bx, Mat By, Mat W)

* 用附录A的方法求得scale, rotation, translationX, translationcoor
* 这里的A和B都是一个单独的shape，即只有一列。

void caculateNewCoordinatesForTrainingShapes(Mat X, Mat Y, MappingParameters P, Mat &newX, Mat &newY)

* 根据mapping参数P来计算新的坐标
* 每个点的新坐标都根据MappingParameters::caculateNewCoordinates()来计算
* 这里的X和Y有所有的shape，即有多列

void getMeanShape(Mat X, Mat Y)

* 计算meanShape
* 这里的X和Y存有所有的shape

void alignTrainingShapes(Mat trainingShapesX, Mat trainingShapesY, Mat& nShapesX, Mat& nShapesY)

* (1)对每一个shape[i]作findBestMapping(shape[1],shape[i])，得到mappingParametersOfAllShapes[]
* (2)调用converMappingParametersToNewCoordinates()计算新坐标。调用getMeanShape()求得meanShape
* (3)调用findBestMapping(Shape[1], meanShape)，之后再用caculateNewCoordinatesForTrainingShapes()转换meanShape
* (4)把每一个shape对齐到meanShape，返回到(2)直到收敛
* 判断收敛：比较两次求得的meanshape的距离
* 最后把每个训练数据的新坐标返回

1. PrincipleComponentAnalyze类

void main(Mat shapesX, Mat shapesY)

* 把同一个训练数据的X、Y合成一列向量
* 调用opencv的PCA

1. TrainingData类

成员：

trainingImages[i] 训练用的图像

trainingShapesX[i] 训练用shapes的X坐标，每一列为一个训练数据

trainingShapesY[i] 训练用shapes的Y坐标

W 权重对角矩阵

AlignedShapesX[i] 对齐后的shapes

成员函数：

void loadTrainingShapeByFilename(string filename)

* 读取地址为filename的单个shape
* shape信息用\_shapeExtension存储

void loadTrainingImageByFilename(string filename)

* 读取地址为filename的单个image
* image信息以图片格式存储

void loadTrainingShapesAndImagesByPath()

* 读取path下所有的shape和image，要求两者同名
* path从FileManager::getPath()取得

void loadTrainingShapesAndImagesByList()

* listname由FileManager::getFilenameWithExtension() 取得
* list文件以\_listExtension结尾

void getWk(int k)

* 计算每一个Wk的值
* Wk为第k个点到其他每一个点距离的方差之和的倒数
* 方差可以用meanStdDev取得

void generateMatrixW()

* W为对角阵
* Wk,k由getWk()取得

void captureStatisticsOfAlignedShapes()

* 调用PrincipleComponentAnalyze::main()

1. ActiveShapeModels

void loadImageByFilename()

* Filemanager::getFilenameWithExtension()

void getInitialShape()

* get the initial shape of the searched image

void calculateMovementForShape()

* return dX

MappingParameters computeMappingParameters()

* AlighShape::findBestMapping()

void deformShape(Mat shapeX, Mat shapeY, Mat dX, Mat dY)

void iterativeSearch()

* shape = getInitialShape()
* (1) calculateMovementForShape(shape, dX, dY)
* (2) computeMappingParameters(shape, dX, dY)
* (3) shape = convertMP2Coordinates
* (4) deformShape(shape, dX, dY)
* goto (1) until the process converges