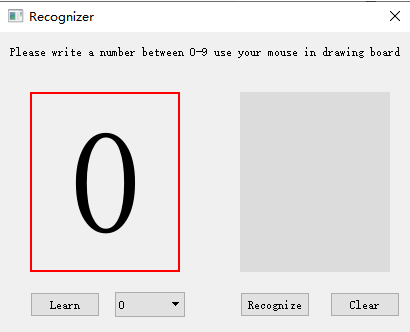
**一、数字识别流程**

1.界面表现

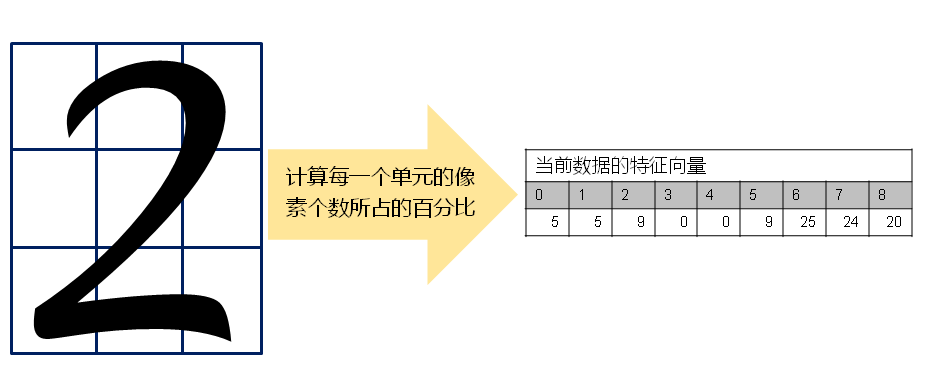


2.学习

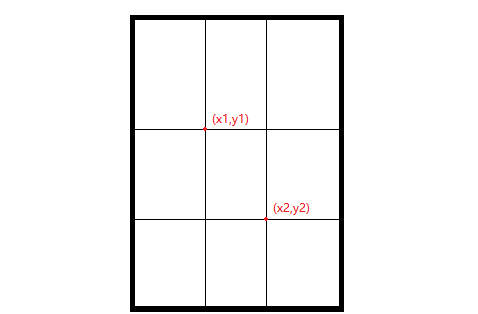
第一步：特征抽取

特征抽取算法：

将数字所在范围进行九宫格的分割，分割之后就可以计算每个九宫格里面有多少个像素，求出每个格中的像素数占整个数字像素的百分比，这样就可以得到一个九维的向量，它表示九维空间中的一个点，这样的一个点用模式识别的术语来说就是”特征向量“。

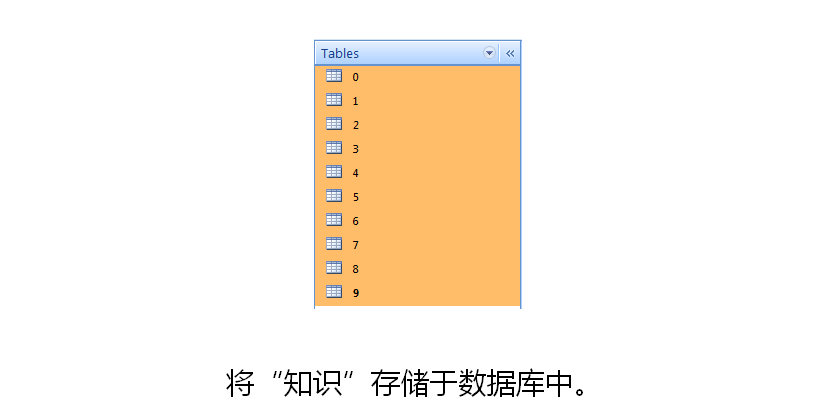


程序中的特征抽取函数是：RecognizerAlgorithm::featureExtraction，程序中的点（x1，y1）和（x2，y2）对应下图中的两个点，用于辅助计算绘制的点落在哪个区域



第二步：特征向量存储（知识学习）

用特征抽取算法就可以对每一个数字图像抽取出一个特征向量，将这些向量存入“知识库”，数据库中存放了“0-9”共十个数字的分类，一个分类下可以存放多个对应的特征向量。



3.识别

（1.用学习时的方法抽取未知的特征向量，抽取出的向量是一个九维的向量，从解析几何的角度看他是九维空间中的一个点。

（2.计算已知数字向量和未知数字特征向量的空间距离。求空间距离的公式：

对于二维空间中的两点A(A1,A2) ,B(B1,B2)，距离d的计算公式是这样的：

d\*d=(A1-B1)\*(A1-B1)+(A2-B2)\*(A2-B2)

那么对于九维空间中的两点A(A1,A2，A3...A9) , B(B1,B2,B3...B9)，距离公式是这样的：

d\*d=*(A1-B1)*\**(A1-B1)*+*(A2-B2)*\**(A2-B2)*+*(A3-B3)*\**(A3-B3)*+...+*(A9-B9)*\**(A9-B9)*

（3.选择空间距离最近的特征向量所对应的数字作为结果输出

（4.程序中对应的识别函数为：RecognizerAlgorithm::get