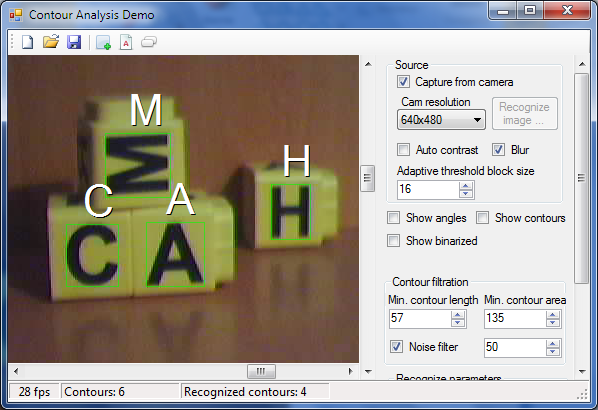
原理说明书（简化+中文版）

## 目标窗口



## CA的基本原理Bases of the Contour Analysis

相同起点时→NSP标准标量积（复数；实部-cosα；范数-形状一样）

不同起点时→ICF互相关函数（|τmax|范数-形状一样；arg(τmax)）旋转角）

ACF自相关函数：性质（1.“和”不因起点不同而改变；2.对称性0-k/2-k；3.⚪-平；突起（明显转角的地方）-有凹陷）

## 算法复杂性的来源

获取轮廓O(N2)

解决方法：初步过滤以减少轮廓数

复杂性O(N2k2t) 其中N-线性图像大小 k-轮廓长度 t模板数

## 轮廓描述符ACF和ICFContour Descriptor

2个ACF O(k)→0-k/2对称性 继续↘

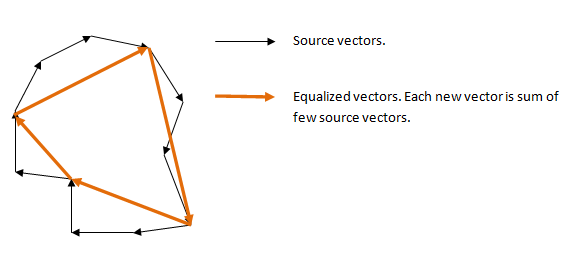
1个ICF O(k2)

ACF O(kt)

ACF结合小波卷积→继续↘

但ACF不精确 简单几何形状时🉑；复杂时经ACF筛选后，再由ICF精确确定

## 均衡化与轮廓平滑Equalization of Contours



## 局限性CA Limitation

无法处理较复杂情况：eg交叉、部分不可见（有断点）

优越性：简单+高速→允许实时识别

## 库ContourAnalysis Library：

**1. project** ContourAnalysis：实现基本功能-创建轮廓、轮廓的标量积（NSP）、均衡、评估ICF和ACF、比较和搜索模板

**（1）The class** Contour：包含轮廓的基本操作-标量积、缩放、均衡、标准化、频谱评估、评估ACF和ICF。

**（2）The class** Template ：用于创建模板的基。存储有-轮廓、ACF、初始轮廓（区域）的线性参数、轮廓的范数、名称（可用作识别符）。

**（3）Class**TemplateFinder ：实现对给定轮廓模板的快速搜索。操作结果为FoundTemplateDesc包括-初始轮廓、匹配的模板、与模板相关的相似率、旋转角度、轮廓比例。

**2. project**  ContourAnalysisProcessing包含-图像的初步处理方法、轮廓的选择、它们的过滤和识别、用于自动生成模板以识别打印符号的工具。用库library OpenCV (EmguCV .NET wrapper)对图像进行操作。

**（1）The class**ImageProcessor用于图像处理。它是存储模板的基础。它还包含用于搜索轮廓的设置。

方法ImageProcessor.ProcessImage()接收输入的图像。操作结果是列出发现的轮廓（ImageProcessor.samples）和已识别轮廓（ImageProcessor.foundTemplates）的列表。

The ImageProcessor工作原理如下：

1. 将图像转换为灰度：
2. 用AdaptiveThreshold进行二值化：
3. 提取轮廓：
4. 根据线性参数（长度、平方等）过滤轮廓：
5. 对轮廓进行均衡化、计算ACF和ACF描述符：
6. 匹配模板：

静态类**The static class** TemplateGenerator用于自动生成特定字体的数字模板。

**3. project** ContourAnalysisDemo除了两个库项目之外，还有一个演示示例demo-example显示了使用网络摄像机对库的操作。演示包含用于创建和编辑模板、识别调优的工具，并允许从网络摄像机生成轮廓识别，还允许创建增强现实。