



人工智能基础与进阶

简单几何形状识别

杨国琦

上海交通大学

目录 content

第一节 识别几何形状的算法

第二节 识别多边形图形

第三节 直线和圆的检测

第四节 直线和圆检测的应用

第五节 简单手势形状识别

上海交通大学人工
智能创新教育实验室

上海交通大学人工
智能创新教育实验室

上海交通大学人工
智能创新教育实验室

第一节 识别几何形状的算法

上海交通大学人工
智能创新教育实验室

上海交通大学人工
智能创新教育实验室

上海交通大学人工
智能创新教育实验室

识别几何形状的算法—问题提出



婴儿



儿童



少年



成年

黑白，线条简单认知到彩色与形状的辨识。

颜色和形状精细分辨、组合理解以及推理

纹理的细节组成、色彩的丰富、绚丽

表象背后的内涵、联想、推理

视觉大尺度

遵循逐次求精、量变到质变、演化提升

视觉小尺度

视觉多尺度
+心理内涵

视觉尺度+
心理感知推理

计算机如何完成这一“成长”过程



识别几何形状的算法—问题提出

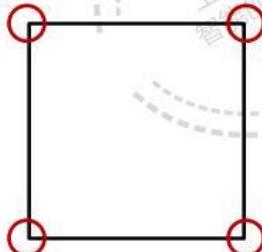
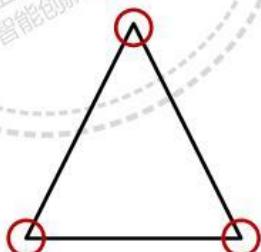
计算机如何完成这一“成长”过程

技术累积与应用选择的过程

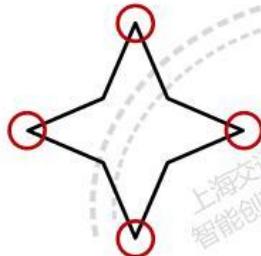


特征与分类器		应用场景	
分类器设计:		工件检测 包装检测 瑕疵检测	
FCM分类器 决策树 随机森林 神经网络 贝叶斯分类器 支撑向量机	1. 几何特征 (长宽高、面积、周长、) 3. 纹理特征 (灰度、彩色) 5. 梯度特征 7. 矩特征	2. 边缘特征 (边缘形状、边缘曲率) 4. 角点特征 6. 频域特征 8. 特征点与局部特征	字符检测 人脸识别 指纹识别 虹膜识别 电路板检测 交通标志识别
深度学习:	不确定性特征 无法表述特征 时变空变特征	场景识别 车辆识别与跟踪 图像检索 视频内容检索 异常行为识别 人的表情、年龄、性别 行人意图识别 行车意图识别	其他

识别几何形状的算法—角点检测



角点在几何上通常可以被定义为两条边的交点，比如，三角形有三个角点，矩形有四个角点



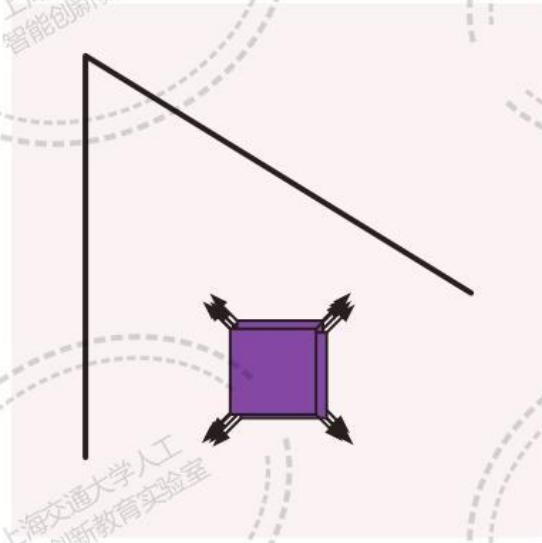
在计算机视觉中，角点是两个边缘的交点，或者说是在局部区域内具有两个主方向的特征点

识别几何形状的算法—Harris角点检测算法

上海交通大学人工
智能创新教育实验室

上海交通大学人工
智能创新教育实验室

上海交通大学人工
智能创新教育实验室



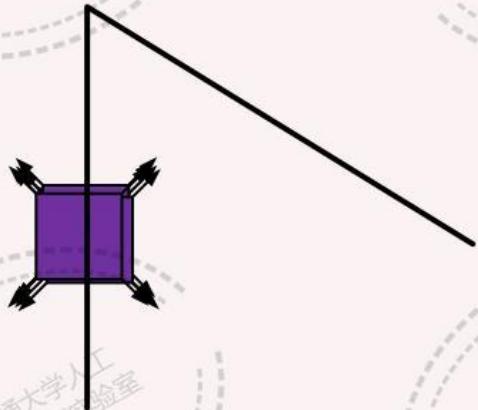
假设在图中有一个滑动的小窗
口，每次滑动都计算窗口内区
域的平均灰度值



如果在各个方向上移动这个小
窗口，窗口内区域的平均灰度
值都没有发生变化，那么就认
为在窗口内遇到了**背景**区域

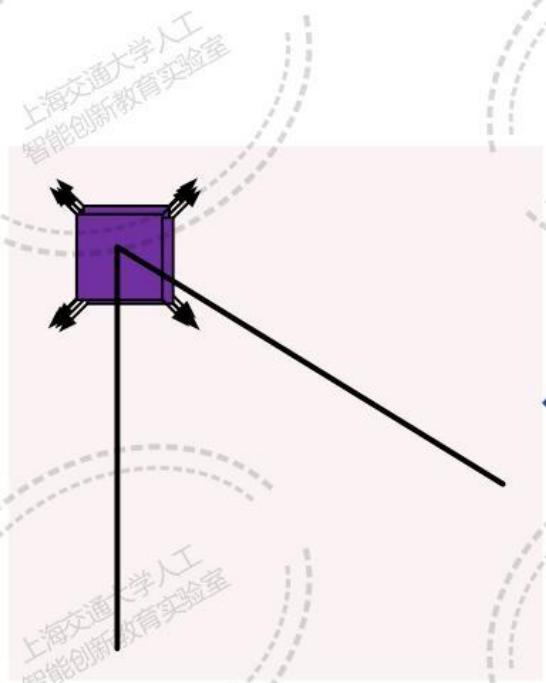
背景

识别几何形状的算法—Harris角点检测算法



如果在各个方向上移动这个小窗口，窗口内区域的平均灰度值在某一特定方向（边缘方向）上没有发生改变，那么就认为在窗口内遇到了**边缘**

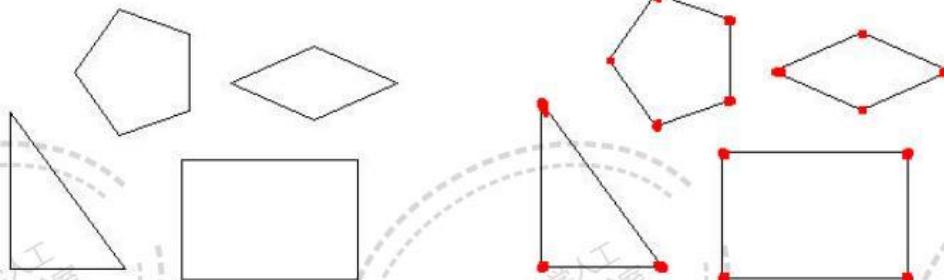
识别几何形状的算法—Harris角点检测算法



如果在各个方向上移动这个小窗口，
窗口内区域的平均灰度值在各个移
动的方向发生明显的改变，那么就
认为在窗口内遇到了**角点**

识别几何形状的算法—Harris算法检测结果

经过Harris角点检测算法，几何图形中的角点都被正确标记出来

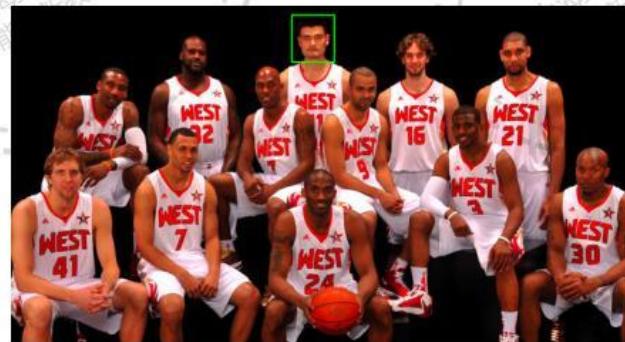


识别几何形状的算法—模板匹配



模板

查找和定位
→



图片

模板匹配（Template Matching）是一种经典的从一张较大图片中查找并定位一个模板图片的方法

识别几何形状的算法—模板匹配

模板匹配算法把模板图像在较大图像上进行滑动，并在每个位置比较模板和较大图像上相应区域的相似性，相似性最高的地方即所得的匹配结果。相似性通过逐像素计算平方差或者相关系数等参数得到。在模板匹配中，常用的相似性度量有以下几种：

- 误差绝对值：逐像素计算模板与原图的灰度差的绝对值，并求和。
- 平方差：逐像素计算模板与原图的灰度的平方差，并求和。
- 归一化平方差：在平方差的结果基础上，除以原图和模板的灰度均值，以进行归一化。
- 相关系数：逐像素计算模板与原图的灰度的乘积，并求和。
- 归一化相关系数：在相关系数的结果基础上，除以原图和模板的灰度均值，以进行归一化。

识别几何形状的算法—模板匹配例子

把模板图片从左到右滑动，从上到下滑动，并在每个位置计算模板图片和原图的误差绝对值的和

5	6
6	5

模板

6	7	4
7	5	6
4	6	5

原图

3	

结果图

5	6
6	5

模板

6	7	4
7	5	6
4	6	5

原图

3	6

结果图

5	6
6	5

模板

6	7	4
7	5	6
4	6	5

原图

3	6
6	

结果图

5	6
6	5

模板

6	7	4
7	5	6
4	6	5

原图

3	6
6	0

结果图

结果图中最小值为第二行第二列的0，也即模板图片在原图的第二行第二列处最为相似，原图该处即为需要检测的目标

第二节 识别多边形图形

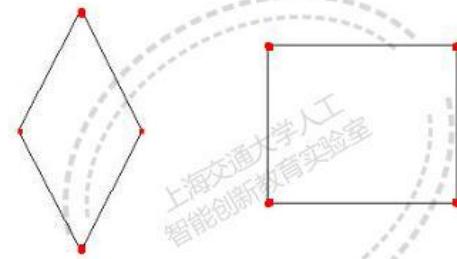
识别多边形图形

三角形有3个角点和三条边，根据角点个数和边即可实现三角形的识别



一般的四边形和矩形都可以检测到4个角

点



识别多边形图形—箭头

为了更好地体会图像处理的基本思想，我们把交通转向标识抽象到以下的四种箭头标志：前进、后退、向左、向右。

采取的识别方式是基于角点特征的识别方法、基于模板匹配的识别方法、基于HOG特征提取的SVM分类方法。



识别多边形图形—箭头

预处理工作——提取标志牌



识别多边形图形—箭头

预处理工作——提取标志牌

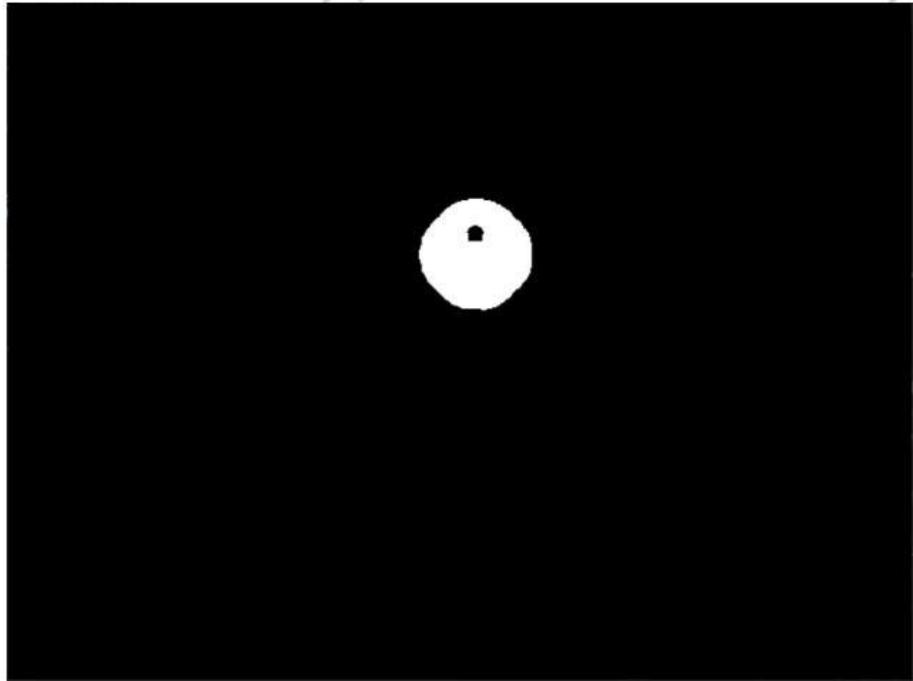
1. 转换颜色空间



识别多边形图形—箭头

预处理工作——提取标志牌

2. 在单个颜色空间阈值分割，通常选取H空间



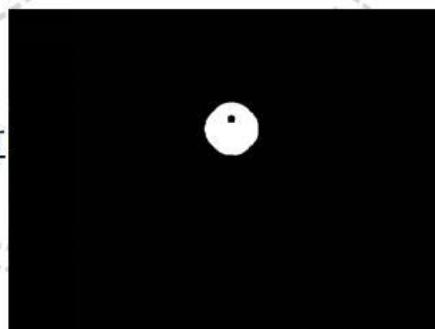
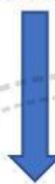
识别多边形图形—箭头

预处理工作——提取标志牌

3. 掩模运算



掩模运算



识别多边形图形—箭头识别过程 (1)

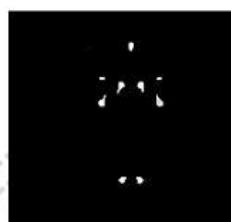
第一种方法：角点特征的识别方法

1. 检测角点

转为灰度图



Harris 角点检测



阈值筛选



检测出的9个角点

识别多边形图形—箭头识别过程 (2)

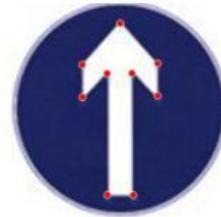
第一种方法：角点特征的识别方法

2. 角点精确定位（聚类）



$$(\bar{x}, \bar{y}) = \frac{1}{N} \sum_{i=1}^N (x_i, y_i)$$

计算每个簇的中心，
得到角点的精确坐标



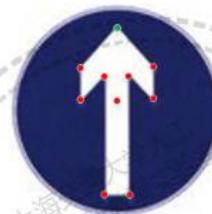
识别多边形图形—箭头识别过程（3）

第一种方法：角点特征的识别方法

3. 求出箭头的重心

取9个箭头角点的坐标的平均值即可求出重心

$$(x_g, y_g) = \frac{1}{9} \sum_{i=1}^9 (x_i, y_i)$$

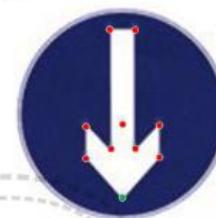
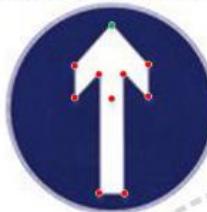


识别多边形图形—箭头识别过程（4）

第一种方法：角点特征的识别方法

4. 求出箭头的顶端

箭头朝上和朝下情形：顶点与重心的横坐标几乎相同，而纵坐标相差较远。利用这个特征，计算出每个角点与重心的横纵坐标差值，分别在横坐标和纵坐标用阈值进行排除即可筛选出顶点。



箭头朝左和朝右情形：重心的纵坐标与顶点的纵坐标几乎相同，而横坐标相距较远，计算出重心与各个角点的横纵坐标的差值，用阈值分别在横坐标和纵坐标进行排除，筛选出顶点。



识别多边形图形—箭头识别过程 (5)

第一种方法：角点特征的识别方法

5. 完成判别

判断重心到顶点的向量方向，该方向即为箭头所指向的方向



角点信息获得后，可以采用前面的决策树方法进行决策树设计，也可以用后面我们将学到的分类器方法，如：SVM等进行处理。在这里，我们将提取到的特征向量与标准向量进行对比，从而完成对箭头方向的判别。

识别多边形图形—箭头识别过程（6）

第一种方法：角点特征的识别方法

7. 分类判别

理想情况下，上、下、左、右四个方向箭头所对应的标准向量分别为

$$\mathbf{e}_1 = (0, 1), \mathbf{e}_2 = (0, -1), \mathbf{e}_3 = (-1, 0), \mathbf{e}_4 = (1, 0)。$$

以向上的箭头为例，令所提取到的特征向量为 α 。

利用向量内积，求得 α 和四个标准向量的夹角，最终分类即为夹角最小的那一类。

$$\eta_i = \arccos \frac{\alpha \cdot \mathbf{e}_i}{|\alpha| |\mathbf{e}_i|}$$

识别多边形图形—箭头识别过程 (6)

第一种方法：角点特征的识别方法

7. 分类判别

假设 $\alpha = (0.02, 0.98)$

计算其与四个标准向量的夹角：



$$\eta_1 = 1.63^\circ$$

$$\eta_2 = 178.36^\circ$$

$$\eta_3 = 91.17^\circ$$

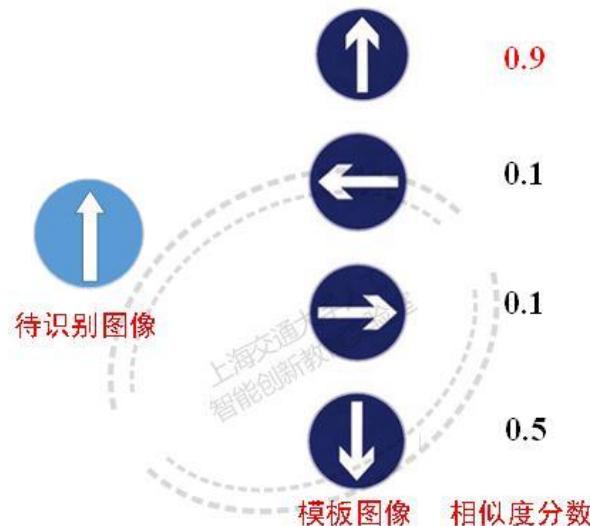
$$\eta_4 = 88.83^\circ$$

因此， α 应被分类为“上”。

识别多边形图形—箭头识别过程

第二种方法：模板匹配的识别方法

准备好4种不同朝向的箭头模板图像，与待识别箭头图案进行模板匹配，相似度分数最高的模板所指的方向即为待识别箭头图案的方向



识别多边形图形—箭头识别过程

第三种方法：HOG特征+SVM分类方法

对待识别图像提取其方向梯度直方图(Histogram of Oriented Gradient, HOG)特征。

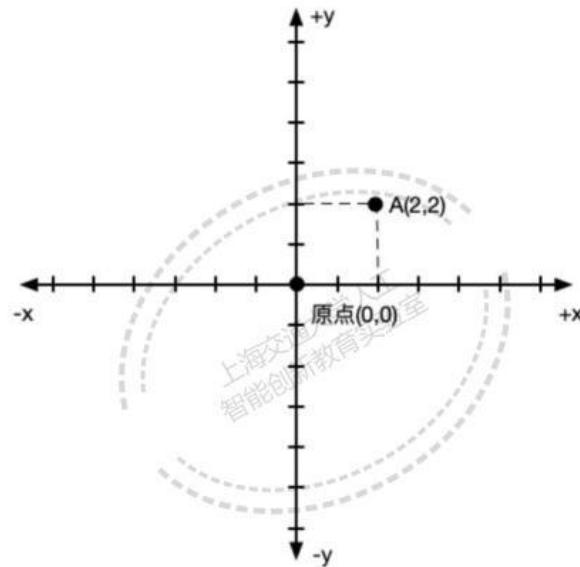


将获得的特征图使用训练好的SVM分类器进行分类。

第三节 直线和圆的检测

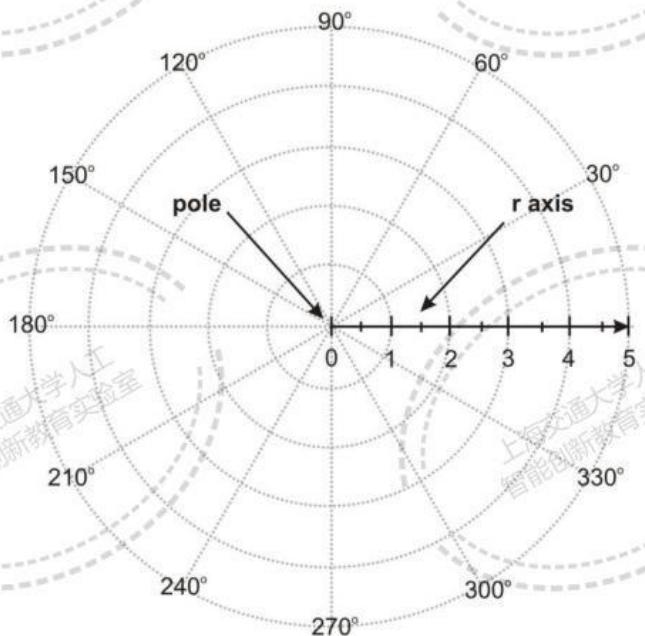
预备知识—坐标系

坐标系，是理科常用辅助方法，常见有直线坐标系，平面直角坐标系。为了说明质点的位置、运动的快慢、方向等，必须选取其坐标系。在参照系中，为确定空间一点的位置，按规定方法选取的有次序的一组数据，这就叫做“坐标”。



预备知识—坐标系

坐标系的种类很多，常用的坐标系有：笛卡尔坐标系、平面极坐标系、柱面坐标系（或称柱坐标系）和球面坐标系（或称球坐标系）等。

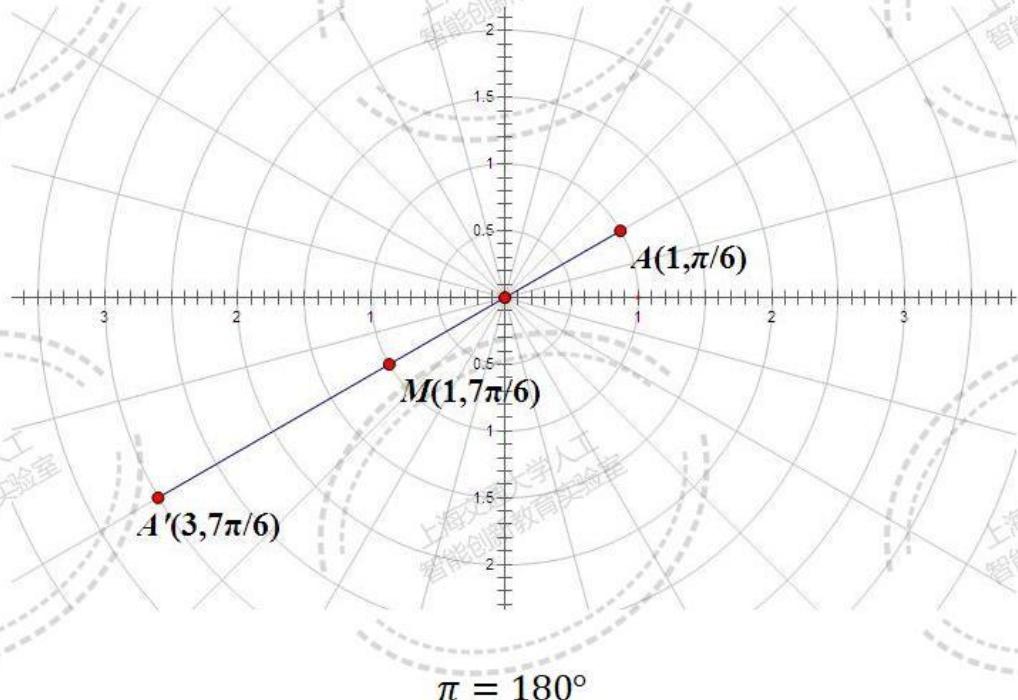


Pole: 极点

r axis: 极轴

预备知识—坐标系

极坐标系中，任意一点以 (ρ, θ) 表示。



预备知识—坐标系

极坐标系中，特殊直线的方程。

1. 过原点（极点）的直线：

$$\theta = \alpha \quad (\text{斜率恒定不变})$$

2. 平行于x轴的直线：

$$\rho \cdot \sin \theta = C \quad (\text{到x轴的距离不变})$$

3. 平行于y轴的直线：

$$\rho \cdot \cos \theta = C \quad (\text{到y轴的距离不变})$$

预备知识—坐标系

极坐标系中，一般直线的方程。

$$y = kx + b$$

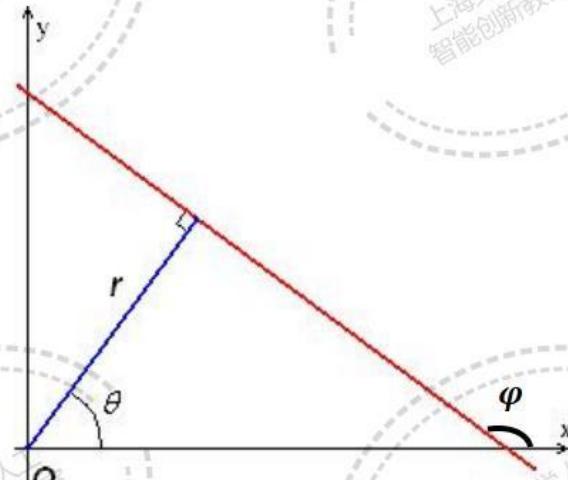
$$\begin{cases} x = \rho \cdot \cos \theta \\ y = \rho \cdot \sin \theta \end{cases}$$

$$\rho \cdot \sin \theta = k \cdot \rho \cdot \cos \theta + b$$

$$\rho \cdot (k \cdot \cos \theta + \sin \theta) + b = 0$$

$$\rho \cdot \sqrt{1 + k^2} \cdot \cos(\theta - \varphi) + b = 0$$

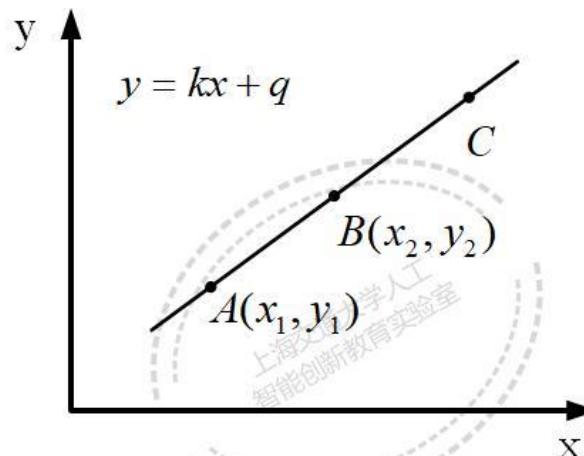
$$\tan \varphi = \frac{1}{k}, \sin \varphi = \frac{1}{\sqrt{1+k^2}}, \cos \varphi = \frac{k}{\sqrt{1+k^2}}$$



直线和圆的检测—直线检测

平面坐标系下直线的表示：

通过 $A(x_1, y_1)$ 和 $B(x_2, y_2)$ 两点的直线可以用 $y = kx + q$ 来表示，其中 k 为斜率， q 为截距。确定 k 和 q 两个参数即可确定一条直线。



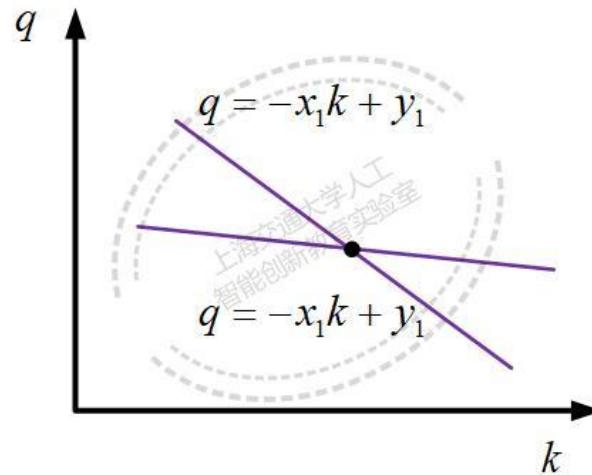
直线和圆的检测—直线检测

将 $A(x_1, y_1)$ 和 $B(x_2, y_2)$ 代入 $y = kx + b$ ，可得方程组

$$q = -x_1 k + y_1$$

$$q = -x_2 k + y_2$$

以 q 为横坐标， k 为纵坐标，上述两式为在 (q, k) 平面的两条直线，我们称 (q, k) 空间为霍夫空间



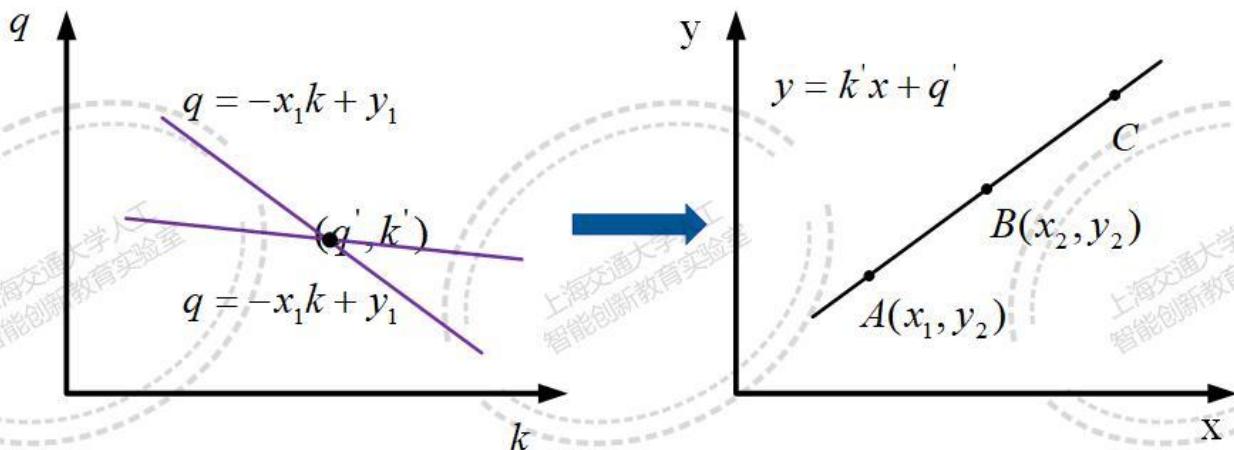
直线和圆的检测—直线检测

霍夫空间下直线的表示：

霍夫空间中两条直线的交点所得的 k' 和 q' 为原空间中通过A、B两点的直线的斜率和截距。

霍夫空间下点的表示：

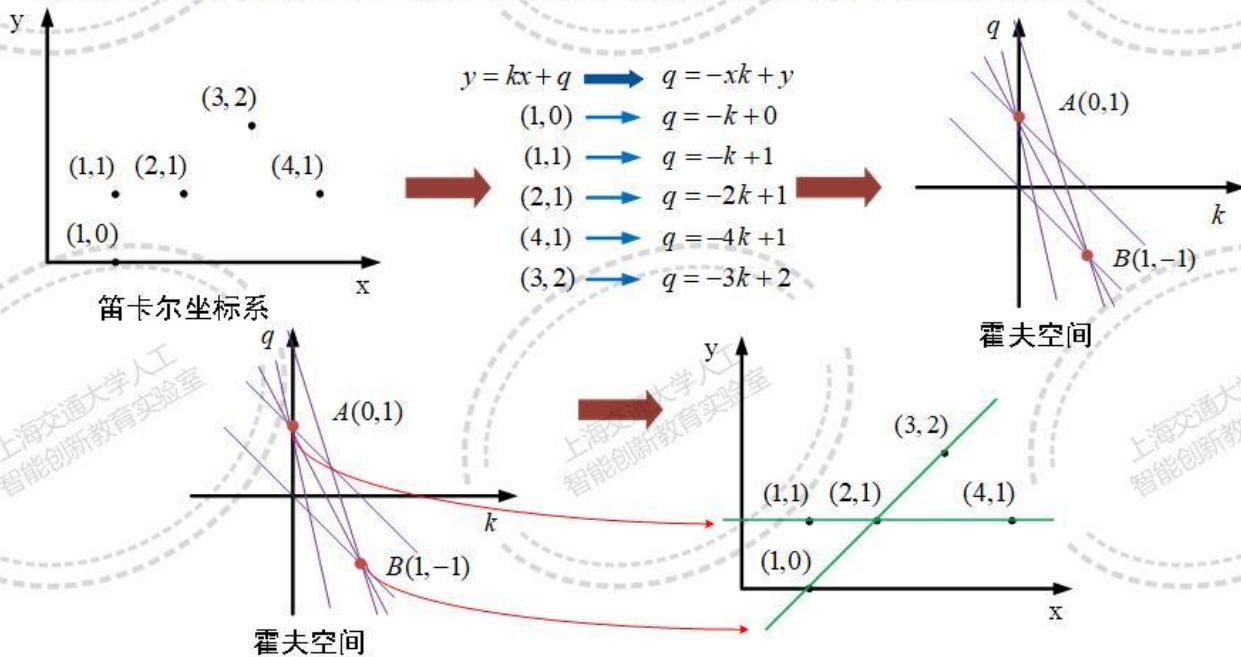
霍夫空间中两条直线对应着原空间中的A和B两点。



直线和圆的检测—直线检测

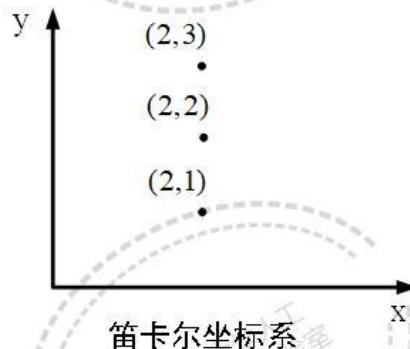
霍夫空间中存在多条直线的交点，交点的坐标代表原坐标系中同时经过多个点的一条直线的斜率和截距，由此确定在原坐标系中通过多个点的一条直线。

把直角坐标系中的点转换成霍夫空间中的直线，选取一个足够大的阈值 m ，如果检测到在霍夫空间中有不少于 m 条直线的交点，则认为在原空间中检测到了1条直线



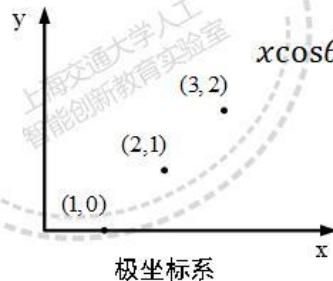
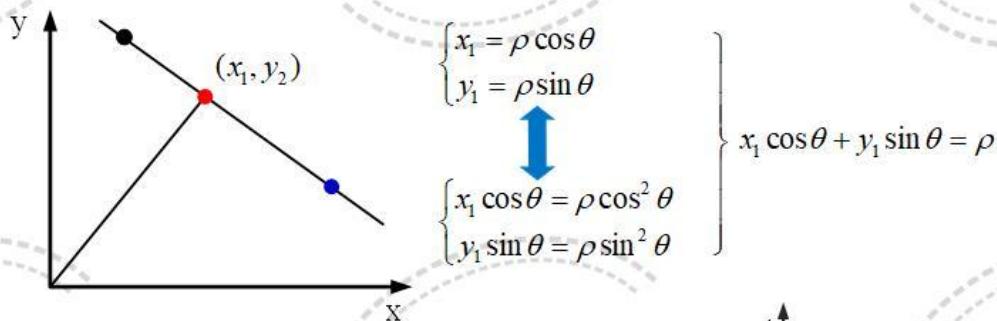
直线和圆的检测—直线检测

特殊情况： 经过这三点的直线的斜率 k 为无穷大，那么在霍夫空间中相应的交点也在无穷远处，实际上计算机不能处理无穷大的数。

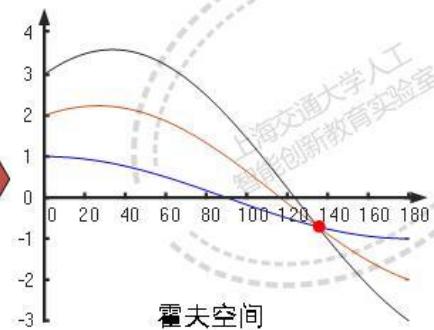


直线和圆的检测—直线检测

为了处理这种情况，将直角坐标系转换为极坐标系，然后再进行霍夫变换。在极坐标系下，一条直线由极径 ρ 和极角 θ 唯一确定。霍夫空间变为以 θ 为横坐标，以 ρ 为纵坐标。同样用在霍夫空间检测交点的方法来检测直线。



$$\begin{aligned} x \cos \theta + y \sin \theta &= \rho \rightarrow \rho = \cos \theta \cdot x + \sin \theta \cdot y \\ (1, 0) &\rightarrow \rho = \cos \theta \\ (2, 1) &\rightarrow \rho = 2 \cos \theta + \sin \theta \\ (3, 2) &\rightarrow \rho = 3 \cos \theta + 2 \sin \theta \end{aligned}$$



直线和圆的检测—直线检测

例子，检测如图所示的图像中的直线

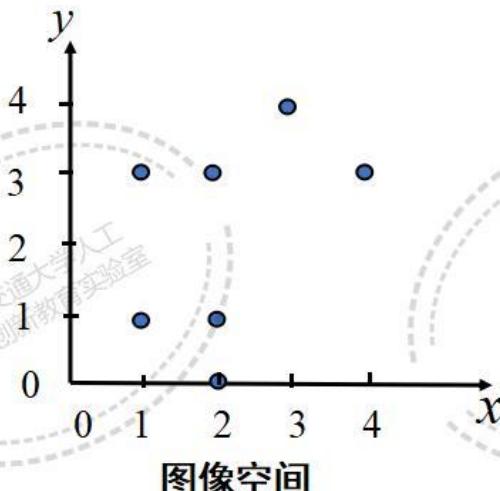
步骤：

1. 离散化 θ 。

$$\theta = -45^\circ, 0^\circ, 45^\circ, 90^\circ$$

2. 按点的坐标(x,y)和每个角度 θ 求r

$$r = x \cos \theta + y \sin \theta$$



(x,y)	-45°	0°	45°	90°
(2,0)	1.4	2	1.4	0
(1,1)	0	1	1.4	1
(2,1)	0.7	2	2.1	1
(1,3)	-1.4	1	2.8	3
(2,3)	-0.7	2	3.5	3
(4,3)	0.7	4	4.9	3
(3,4)	-0.7	3	4.9	4

直线和圆的检测—直线检测

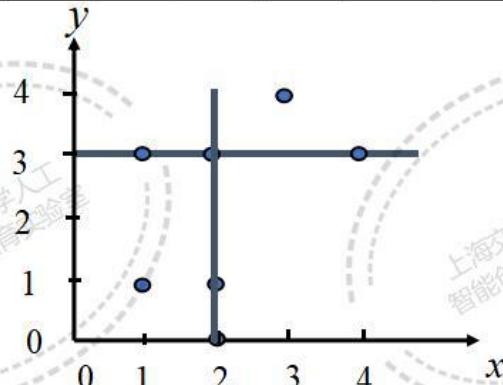
3. 统计 (r, θ) 出现的次数。

	-1.4	-0.7	0	0.7	1	1.4	2	2.1	2.8	3	3.5	4	4.9
-45°	1	2	1	2		1							
0°					2		3			1		1	
45°						2		1	1		1		2
90°			1		2					3		2	

最大次数3出现：

$$(r, \theta) = (2, 0^\circ)$$

$$\text{和 } (r, \theta) = (3, 90^\circ)$$



霍夫变换检测到的直线

直线和圆的检测—直线检测

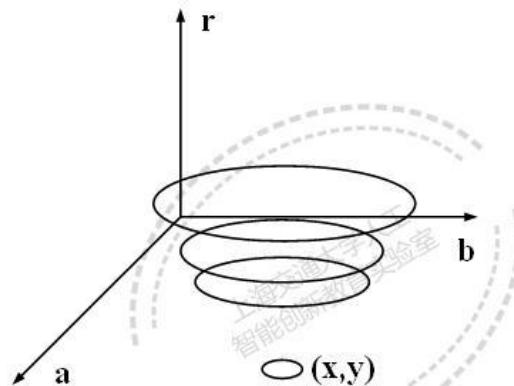
总 结

根据直角坐标系和极坐标系变换域之间的关系，总结霍夫变换具有如下主要性质：

- 直角坐标系中的一点对应于极坐标中的一条正弦曲线
- 变换域极坐标系中一点对应于直角坐标系中的一条直线
- 直角坐标系一条直线上的N个点对应于极坐标系中共点的N条曲线。

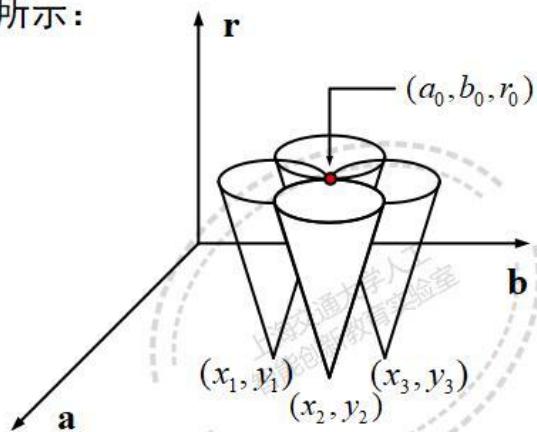
直线和圆的检测—圆的检测

检测圆的原理与检测直线类似，不同的是直线是在二维空间，因为 $y=kx+q$ ，只有 k 和 q 两个参数。而圆形的一般性方程表示为 $(x-a)^2 + (y-b)^2 = r^2$ 。有三个参数圆心坐标 a, b ，和半径 r 。使用 (a, b, r) 来确定一个圆心为 (a, b) 半径为 r 的圆。图像平面的一个圆对应着参数空间的一个点，图像平面中的一点 (x, y) 通过霍夫变换转换到 (a, b, r) 参数空间得到一个三维锥面。



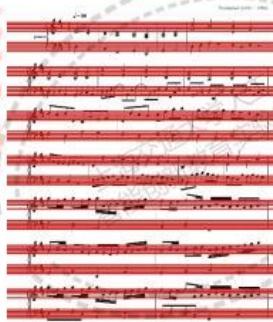
直线和圆的检测—圆的检测

过图像空间中任意一点的圆对应于参数空间中的一个三维锥面。因此，过图像空间上同一圆上的点，对应的参数空间中的三维锥面，在 r 高度必然相交于一点 (a, b, r) 。这样通过检测这一点可以得到圆的参数，可求出过多个点的相应的圆。图像平面的方程转化为参数平面上的示意图如下所示：



直线和圆的检测—示例

直线、圆和椭圆检测结果示意图如下所示：



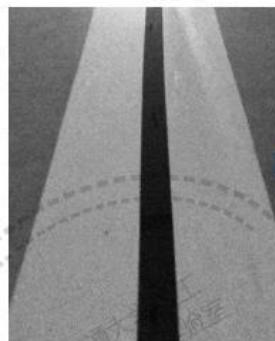
票面号码: 001014597204	开票日期: 2023-09-11
开票人: 深圳市增值税普通发票	发票代码: 044002000111
地税局: 上海市税务局	发票号码: 27938030
纳税人识别号: 1210000042500615X0	机打金额: 7451.08
地址、电话:	税额: -<<928>/~6*>-51-1226124521365
开户行及账号:	-*>+98<65<0*>0843-7519-3874
经办人(签章):	><65*>9209<6*>1>036408/<
金额栏:	7451.08<>>-<<7149<93*>18<>+8
税额栏:	*<<928>/~6*>-51-1226124521365
合计栏:	-*>+98<65<0*>0843-7519-3874
备注栏:	><65*>9209<6*>1>036408/<
开票人: 吴爱芳	开票日期: 2023-09-11 12:18:00
地税局: 上海市税务局	发票代码: 044002000111
纳税人识别号: 91440300591865259K	发票号码: 27938030
地址、电话:	机打金额: 7451.08
开户行及账号:	税额: -<<928>/~6*>-51-1226124521365
经办人(签章):	><65*>9209<6*>1>036408/<
金额栏:	7451.08<>>-<<7149<93*>18<>+8
税额栏:	*<<928>/~6*>-51-1226124521365
合计栏:	-*>+98<65<0*>0843-7519-3874
备注栏:	><65*>9209<6*>1>036408/<

第四节 直线与圆检测的应用

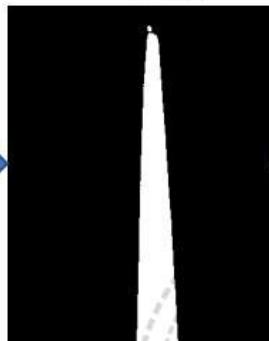
拓展应用—车道线检测

智能小车车道线检测（1）

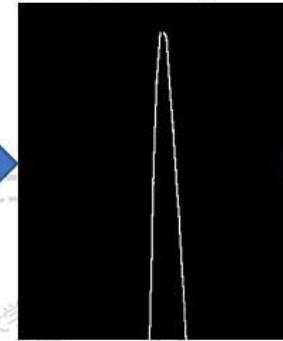
原图



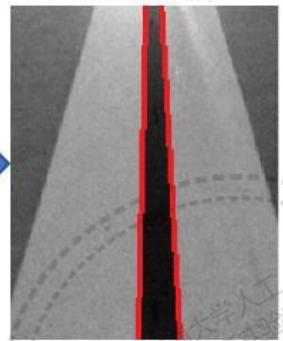
二值化



边缘检测



直线检测



拓展应用—车道线检测

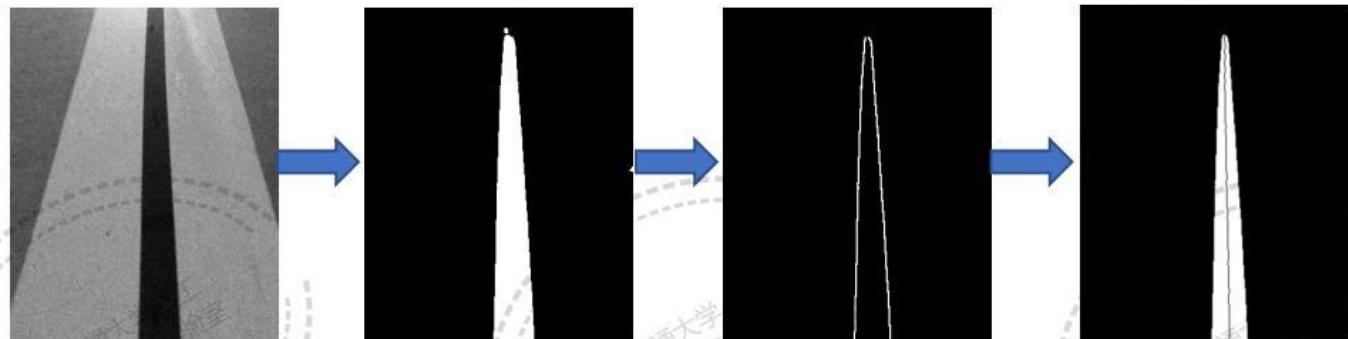
智能小车车道线检测（2）

原图

二值化

边缘检测

行扫描点



拓展应用—车道线检测

无人驾驶车道线检测 — 直道



驾驶场景图像



灰度图



边缘检测

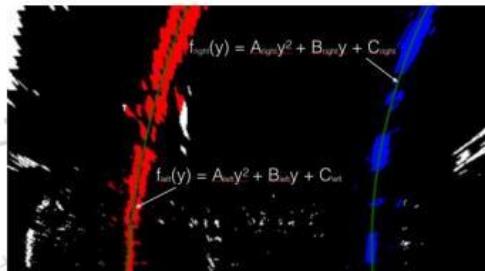


直线检测

拓展应用—车道线检测

无人驾驶车道线检测—弯道

遇到弯道时，直线检测无法得到可靠的结果，在进行边缘提取后，采用多项式拟合的方法进行车道线检测。



拓展应用—车道线检测

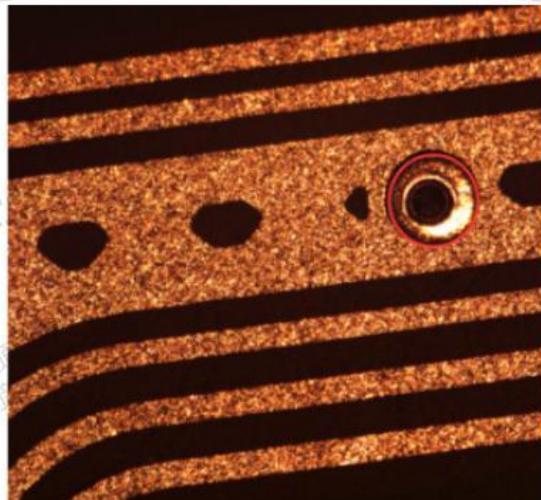
无人驾驶车道线检测—复杂情况

使用深度学习分割车道

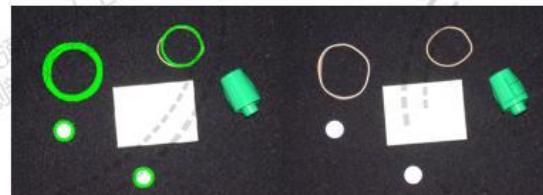
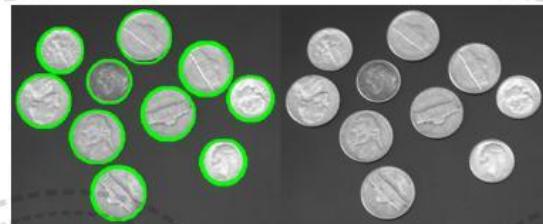


拓展应用—其他应用

利用霍夫圆检测，对电路板
加工中的圆孔进行检测



利用霍夫圆检测，对圆形物体
进行计数



拓展应用—其他应用

综合各种几何形状的检测方法，对餐盘进行检测，实现无人结账

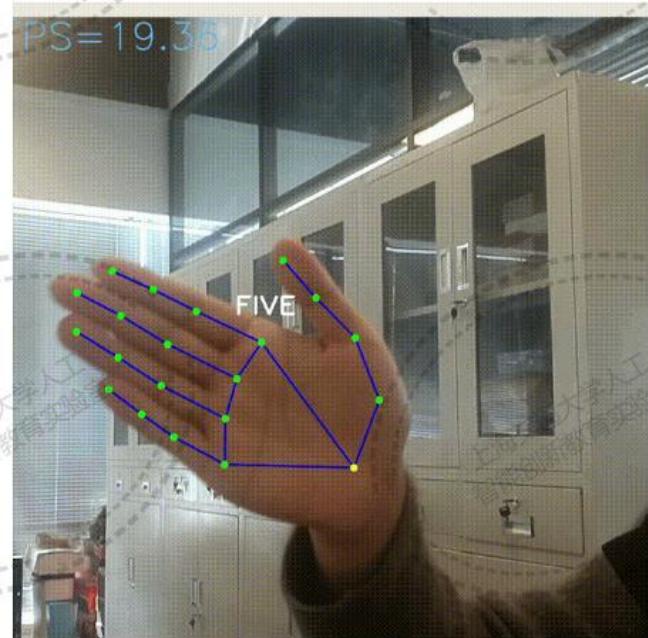


第五节 简单手势形状识别

简单手势形状识别

手势识别是计算机科学和语言技术中的一个主题，目的是通过数学算法来识别人类手势。手势可以源自任何身体运动或状态，但通常源自面部或手。

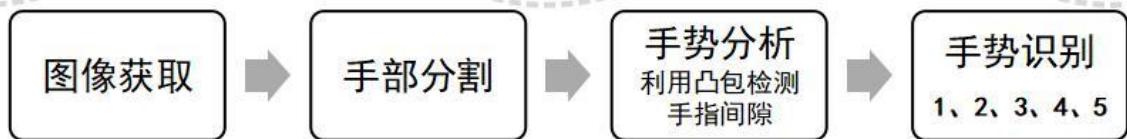
手势识别可以被视为计算机理解人体语言的方式，从而在机器和人之间搭建比原始文本用户界面或甚至GUI（图形用户界面）更丰富的桥梁。用户可以使用简单的手势来控制或与设备交互，而无需接触它们。



简单手势形状识别关键技术

关键技术：

手势无论是静态或动态，其识别顺序首先需进行图像的获取手的检测和分割手势的分析，然后进行静态或动态的手势识别。



主要步骤：

手势分割、手势分析、手势识别。

介绍一种基于颜色空间的简单手势识别算法。

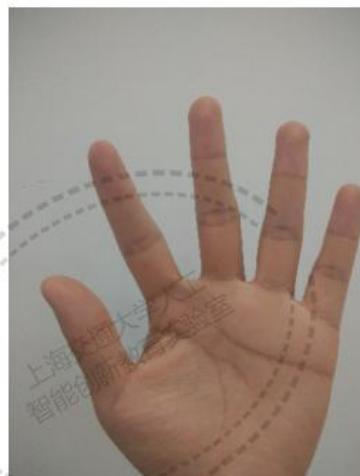


手部分割

主要内容：

转换颜色空间、通道提取、阈值分割。

肤色是手和其他背景最明显的区分特征，手的颜色范围较统一并且有聚类性，基于肤色的分割方法处理速度快，对旋转、局部遮挡、姿势变换具有不变性，因此利用不同的颜色空间来进行手势分割是现在最常用的方法。



手部分割

主要内容：

肤色提取：读取图像，将其转换到合适的颜色空间，选取某一通道进行阈值分割。肤色分割的内容都大致相同，包括：

1. 基于HSV颜色空间的阈值肤色识别；
2. 基于RGB颜色空间的阈值肤色识别；
3. 基于灰度空间的阈值肤色识别。

接下来介绍基于HSV和RGB颜色空间的肤色识别的阈值选择。

手部分割

基于HSV颜色空间的阈值肤色检测：

在HSV空间中，H为色相，S为饱和度，V为明度。一般通过色相值对肤色进行检测。

$$2 \leq H \leq 13$$

基于RGB颜色空间的阈值肤色检测：

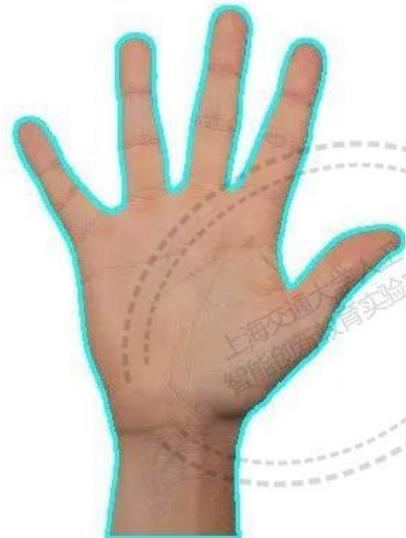
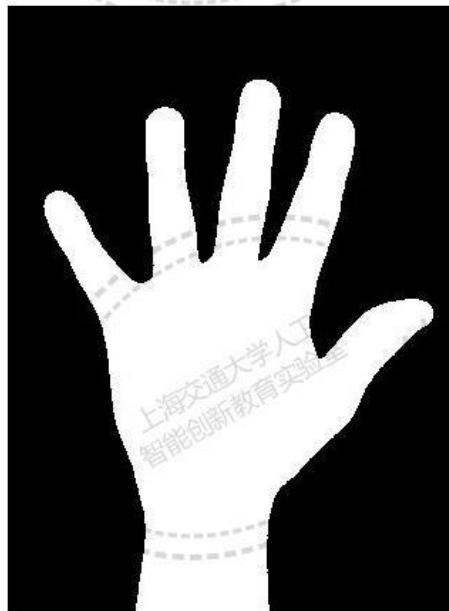
在RGB彩色模型中，每种颜色都以红(R)、绿(G)、蓝(B)光谱成分表示，用RGB颜色空间对物体的颜色进行描述，很符合人们的直观感受，但是肤色受亮度的影响较大，需要用复杂的约束才能将皮肤区域较好地检测出来。

$$\begin{cases} (R, G, B) > (95, 40, 20) \\ \max(R, G, B) - \min(R, G, B) > 15 \\ R - G > 15 \\ R > B \end{cases}$$

轮廓提取

轮廓提取：

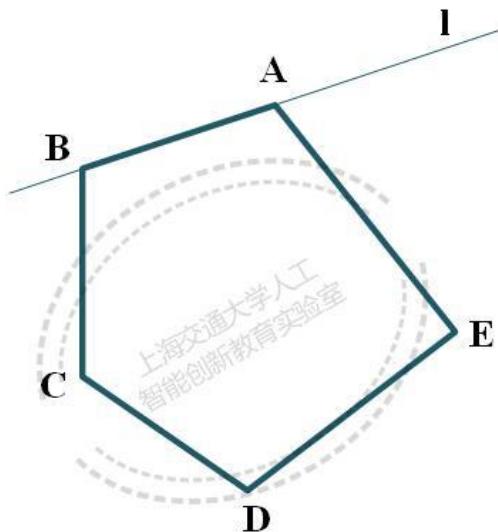
首先对分割好的手部图像提取外轮廓（Canny算子）。需要注意的是，提取到的轮廓还需要进行筛选处理以防出现多个轮廓的情况。



手指间隙检测

凸多边形：

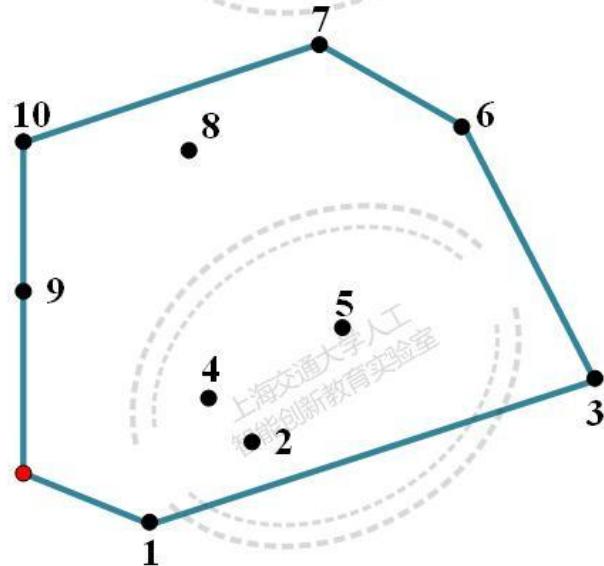
如果一个多边形的任意一边所在的直线，能够使该多边形的其他各边都在直线的一侧，那就认为这个图形是凸多边形。凸多边形的所有内角都小于 180° ，任意两个顶点的连线段都处于多边形的内部或边界上。



手指间隙检测

凸包：

在给定的二维平面上存在一个有限点的集合，能够完全的包围这个集合的凸多边形有无数个，**凸包就是其中面积最小的一个。**

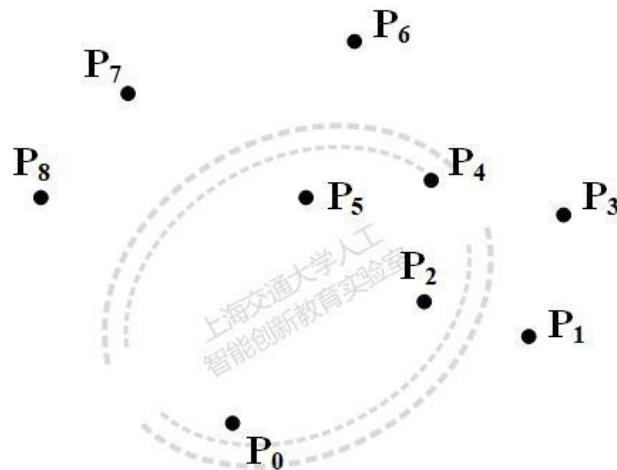


手指间隙检测

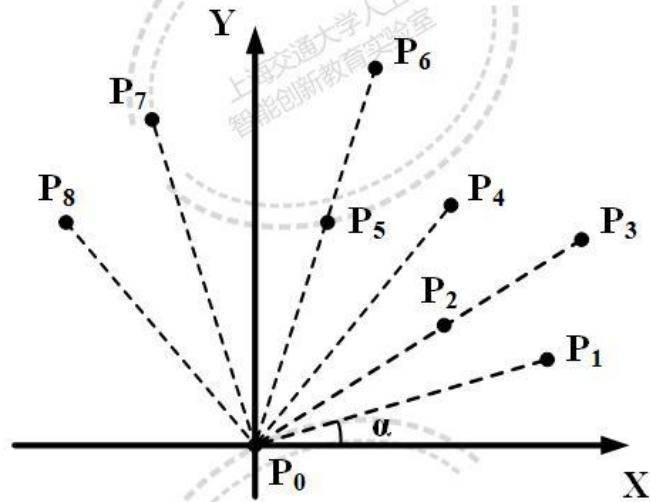
凸包检测——Graham扫描法

Graham扫描的思想是：先找到凸包上的一个点，然后从那个点开始按逆时针方向逐个找凸包上的点。主要算法步骤如下：

假设有如下几个点，寻找包围它们的凸包

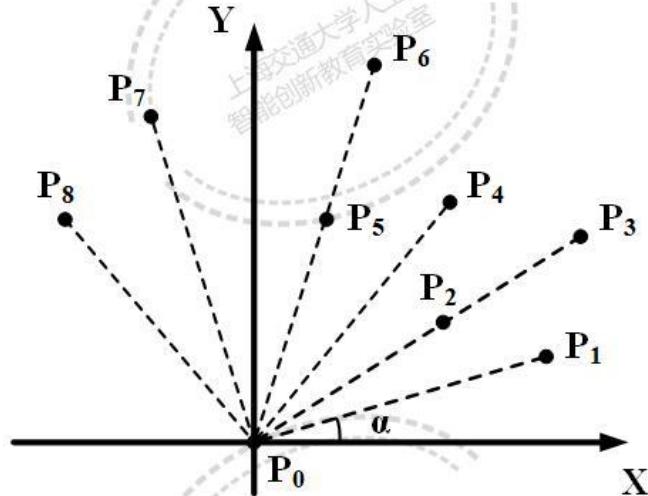


手指间隙检测



- ① 把所有点放在二维自然直角坐标系中（注意与图片像素坐标系区分），则纵坐标最小的点一定是凸包上的点，如图中的 P_0 。
- ② 以 P_0 作为坐标参考点，如上图所示。

手指间隙检测



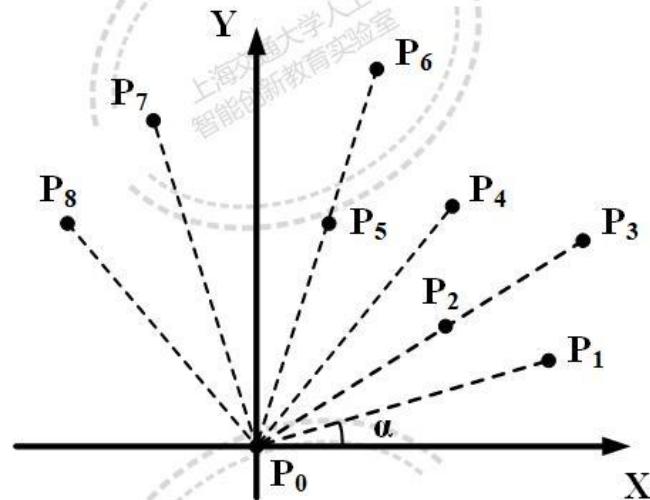
③ 计算各个点相对于 P_0 的幅角 α ，按从小到大的顺序对各个点排序。当 α 相同时，距离 P_0 比较近的排在前面。例如上图得到的结果为 $P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8$ 。由**定义**可以知道，结果中第一个点 P_1 和最后一个点 P_8 一定是凸包上的点。

手指间隙检测

栈顶→



当前点: P_2



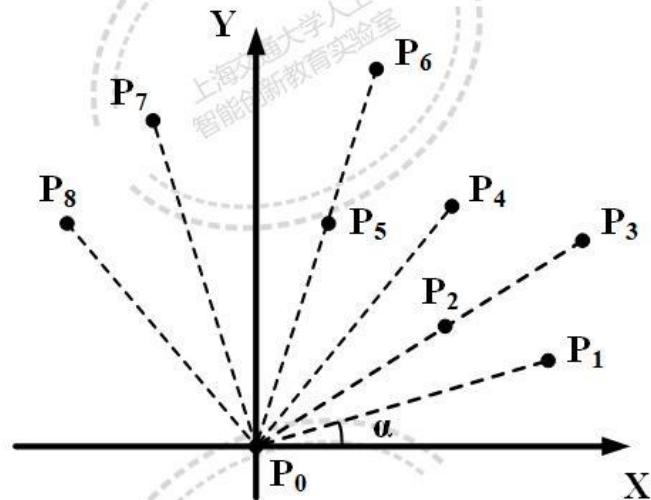
以上，已经知道了凸包上的第一个点 P_0 和第二个点 P_1 ，把它们放在栈里面。现在从步骤3求得的结果中，把 P_1 后面的点拿出来做当前点，即 P_2 。接下来开始找下一个点。

手指间隙检测

栈顶→

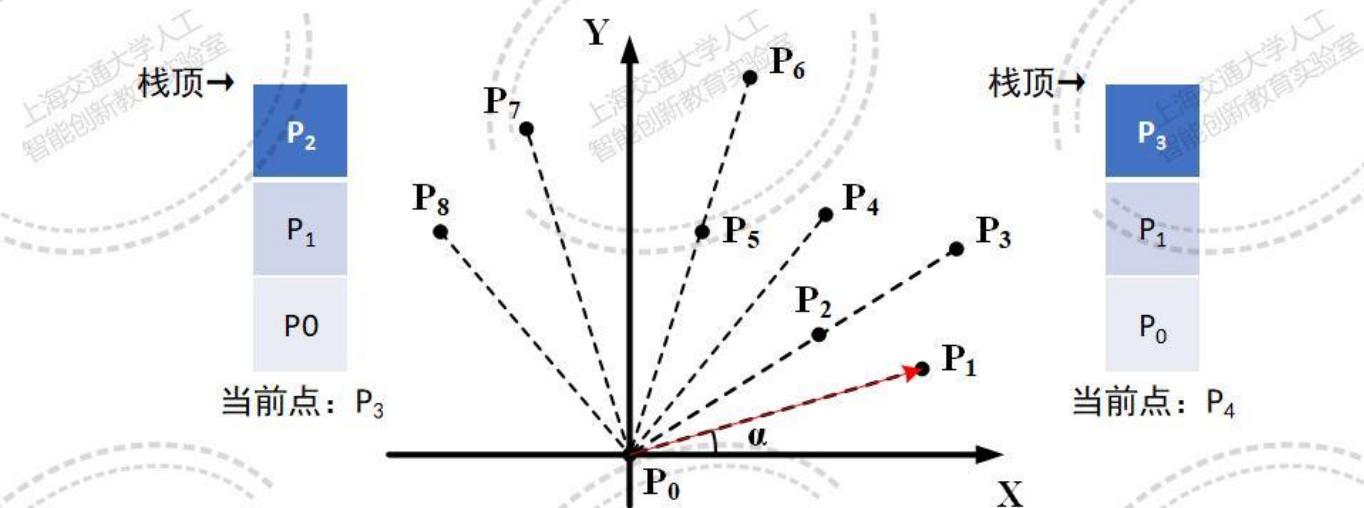


当前点: P₂



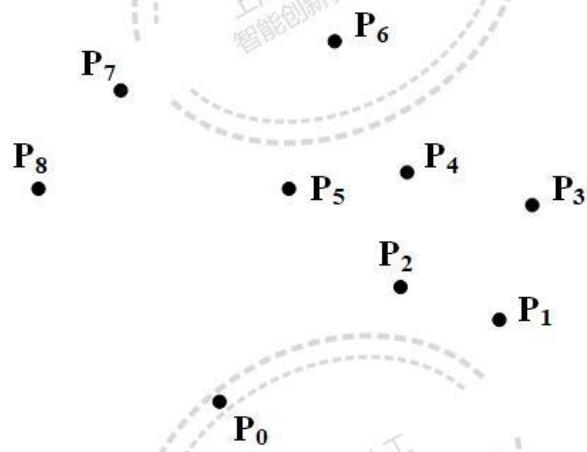
- ④ 根据栈顶的两个点作出有向直线L (P_{x-1} 指向 P_x)，记 $P_{x-1} \rightarrow P_x$ 。看当前点是在有向直线 L 的右边还是左边（怎么看：把有向直线作为竖直轴）。如果在直线的右边就执行步骤⑤；如果在直线
上，或者在直线的左边就执行步骤⑥。

手指间隙检测



- ⑤ 栈顶的那个元素不是凸包上的点，把栈顶元素出栈、当前点入栈。
把下一个点作为新的当前点，执行步骤④。
- ⑥ 当前点是凸包上的点，把它压入栈，执行步骤⑦。
- ⑦ 检查当前点是不是步骤③结果中最后一个元素。是最后一个元素
则结束；如果不是则把下一个点作为新的当前点，返回步骤④。

手指间隙检测

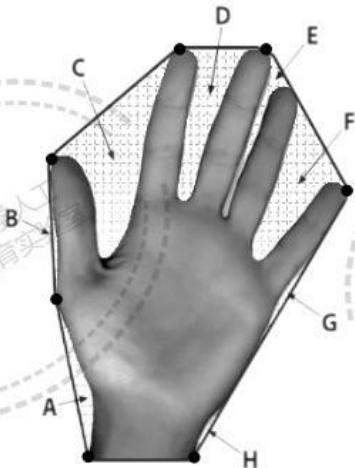


总结：右出栈，非右入栈。

手指间隙检测

凸缺陷检测：

如下图所示，黑色的轮廓线为凸包，而凸包与手掌轮廓之间的部分为凸（包）缺陷。每个凸缺陷区域有四个特征量：起始点，结束点，距离凸包最远点，最远点到凸包的距离。利用这四个特征量，我们可以进一步计算凸缺陷的角度特征。

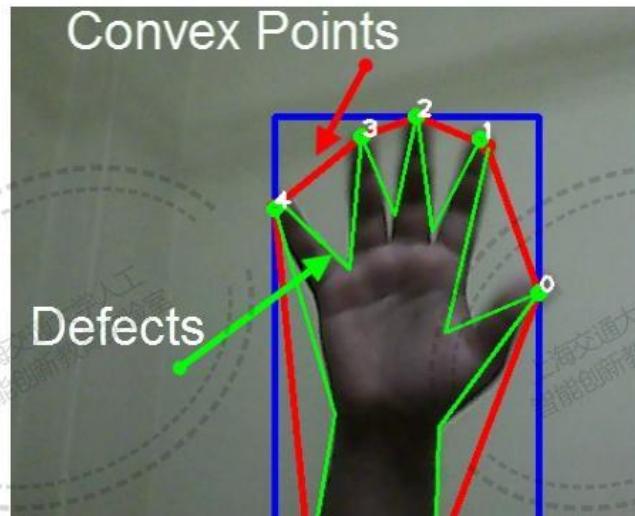
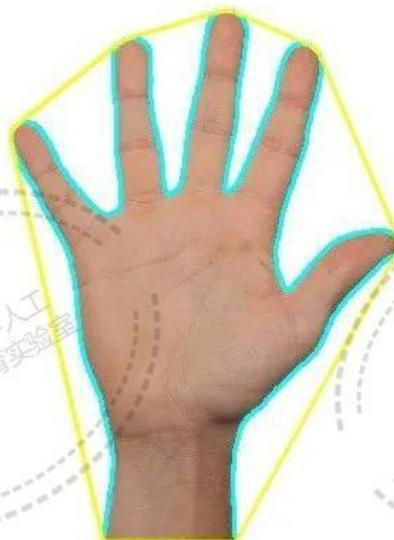


起始点，结束点：为凸包上的顶点；
距离凸包最远点：即边缘点到凸包
距离最大点；
将这三个点构成的三角形，运用余
弦定理，来判断凸缺陷。

手指间隙检测

凸包检测（左图）和凸缺陷检测（右图）：

手掌与凸包检测轮廓线的任何偏离的地方都可以视为凸度缺陷，这样可以检测出手指间隙中最低点。



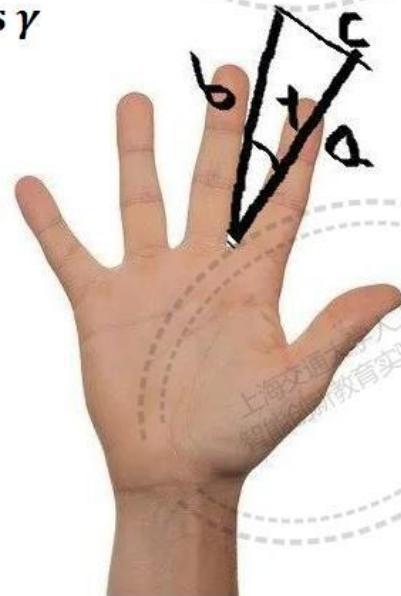
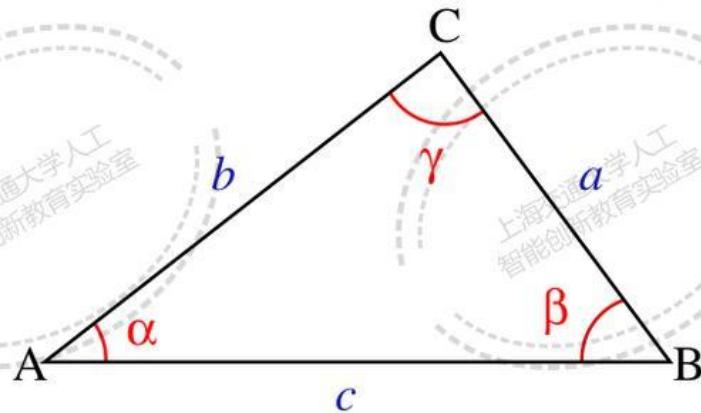
手指间隙检测

余弦定理：已知三角形的三边长，根据余弦定理计算手指间的夹角 γ ，

当 $\gamma < 90^\circ$ 时，认为成功检测到了手指间隙。

$$c^2 = a^2 + b^2 - 2 \cdot a \cdot b \cdot \cos \gamma$$

$$\gamma = \arccos \frac{a^2 + b^2 - c^2}{2 \cdot a \cdot b}$$



手势识别

2

5

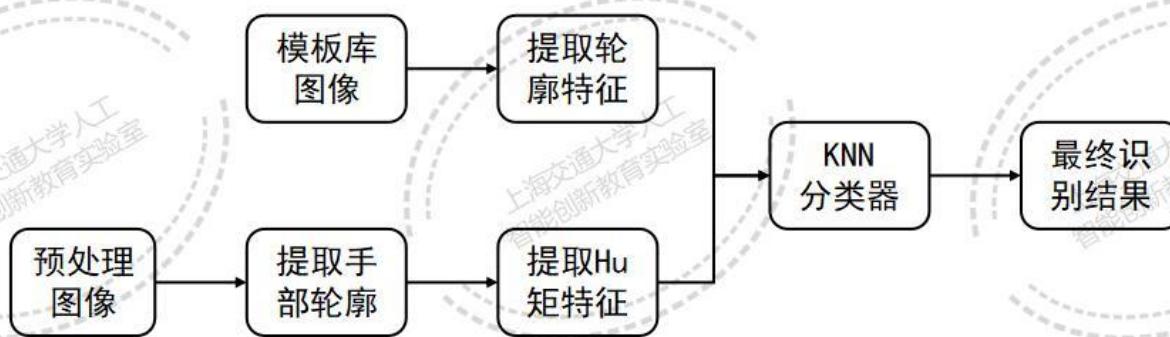


最后，根据检测到的手指间隙（图中红点）总数，判断手势所表达的数字。

复杂手势识别

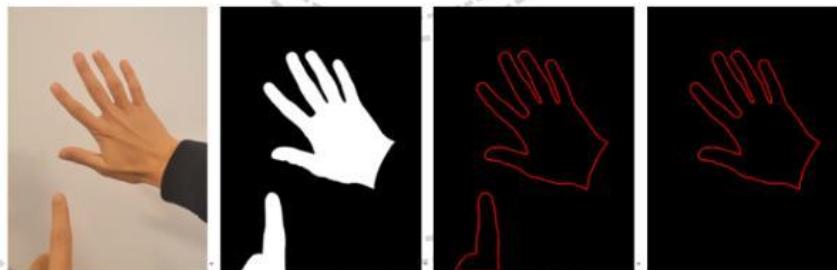
要对更加复杂的手势进行识别，则需要在获取手部轮廓后对其进行特征提取，然后利用设计好的分类器对特征进行分类。常用的手势特征有：Hu矩特征、边界傅里叶描述子、方向梯度直方图(HoG)、Haar-like等。

介绍一种简单的分类识别算法——该算法提取轮廓的Hu矩特征，使用KNN对该特征进行分类。



复杂手势识别

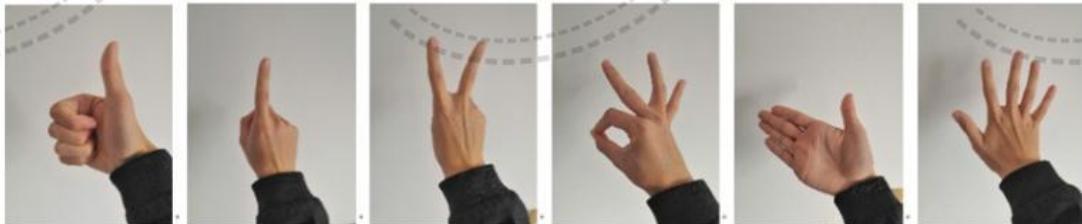
矩是统计学中的一个概念，用来反映随机变量的分布情况。矩特征具有旋转、平移和缩放不变性，因此又被称为不变矩。。



对上面四幅图中最后的一副（也就是提取到的手部轮廓）进行Hu矩特征提取，用提取到的特征向量进行识别分类。

复杂手势识别

矩是统计学中的一个概念，用来反映随机变量的分布情况。矩特征具有旋转、平移和缩放不变性，因此又被称为不变矩。Hu矩特征具有7个分量，如下所示：



手势类型	H[0]	H[1]	H[2]	H[3]	H[4]	H[5]	H[6]
Good	0. 0444	0. 0464	0. 1071	0. 0473	0. 3857	0. 0725	0. 7034
沉默	0. 0289	0. 0177	0. 0320	0. 0281	0. 5399	0. 0270	0. 6095
胜利	0. 0826	0. 0835	0. 0858	0. 0859	0. 2823	0. 1179	0. 7379
OK	0. 1318	0. 1320	0. 1319	0. 1337	0. 7541	0. 1485	0. 0762
握手	0. 0844	0. 0844	0. 1097	0. 0856	0. 2864	0. 0867	0. 7372
你好	0. 1372	0. 1373	0. 1377	0. 1421	0. 7509	0. 1563	0. 0403

简单手势识别——手势识别

KNN分类：

提取到待识别手势的Hu矩特征之后，利用KNN算法对其进行分类。

假设待识别手势和模板手势的Hu矩特征分别为

$$(H[0], H[1], H[2], H[3], H[4], H[5], H[6])$$

$$(H_m[0], H_m[1], H_m[2], H_m[3], H_m[4], H_m[5], H_m[6])$$

则KNN距离计算公式为

$$D = \sqrt{\sum_{i=0}^6 (H[i] - H_m[i])^2}$$



谢谢聆听

THANKS FOR YOUR ATTENTION