

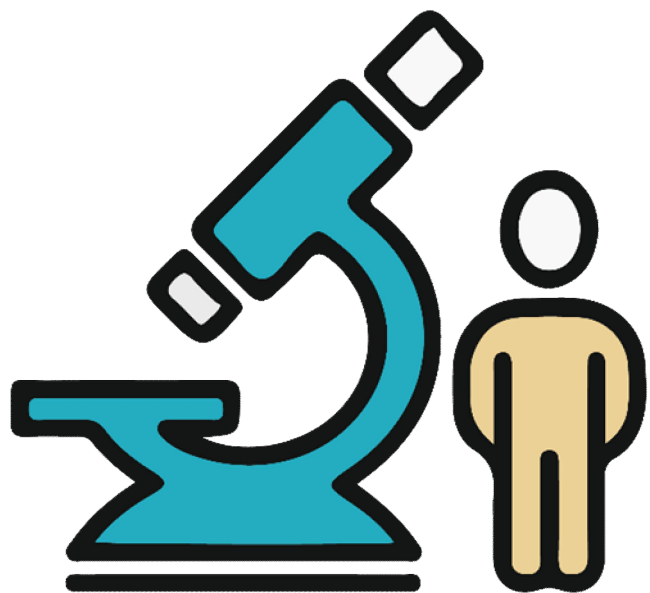
察言观色的狗狗

认识颜色识别/颜色识别体验

让孩子体验黑科技

陆吾智能





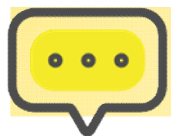
察言观色的狗狗

02.

PART 2



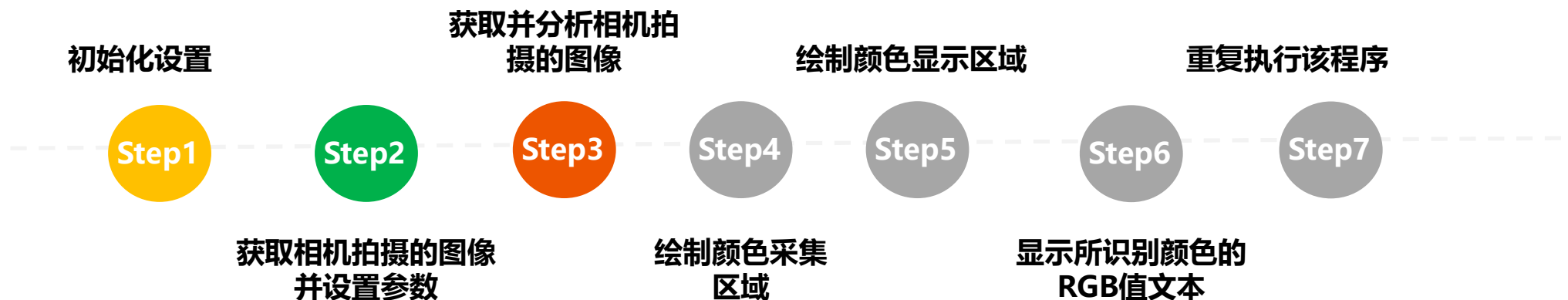
如何让狗狗能区分颜色呢？



步骤3：获取并分析相机拍摄的图像

任务要求：

- 1.创建相机变量；
- 2.创建RGB变量；
- 3.取用颜色分析编程积木，并将颜色采集的区域设置为屏幕正中央。





步骤3：获取并分析相机拍摄的图像

所需积木指令：

在积木指令区点击以下指令，并依次放在积木编程区：

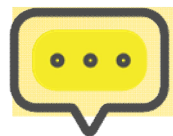
1. 变量：【将变量设定为...】
2. 扩展模块 | 摄像头：【获取相机捕捉的图像】
3. 人工智能模块 | 图形处理：【分析区域颜色】
4. 数学运算：【加减乘除】
5. 数学运算：【取商数】

赋值 camera 为 0

获取相机捕捉的图像

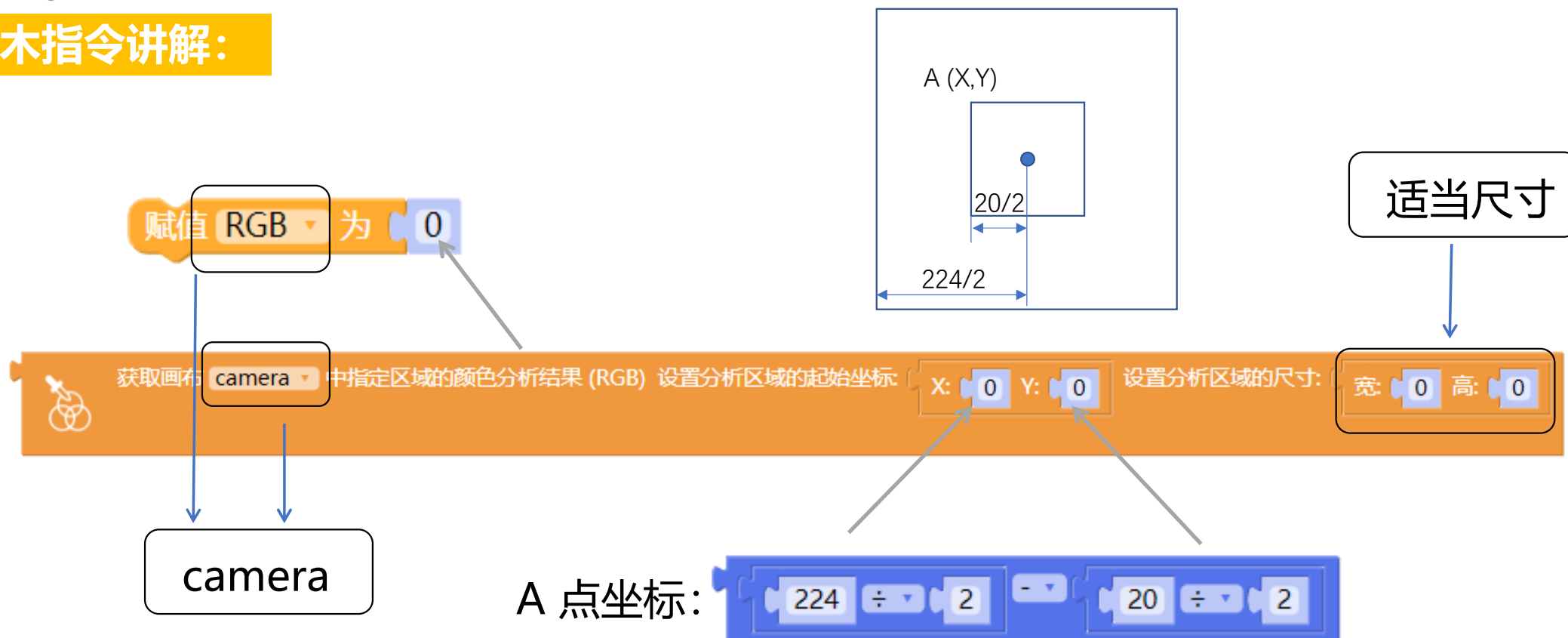
获取画布 camera 中指定区域的颜色分析结果 (RGB)
设置分析区域的起始坐标:
X: 0 Y: 0
设置分析区域的尺寸:
宽: 0 高: 0

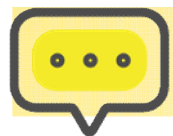
1 + 1



步骤3：获取并分析相机拍摄的图像

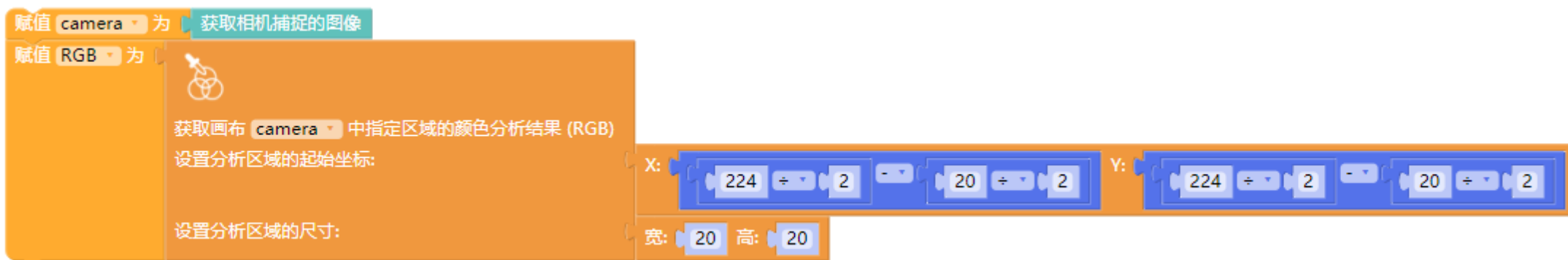
积木指令讲解：

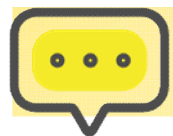




步骤3：获取并分析相机拍摄的图像

参考程序：





步骤4：绘制颜色采集区域

任务要求：

1. 设置画布坐标；
2. 在屏幕中央绘制矩形，标出采集颜色的区域。





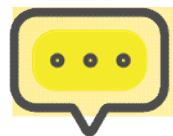
步骤4：绘制颜色采集区域

所需积木指令：

在积木指令区点击以下指令，并依次放在积木编程区：

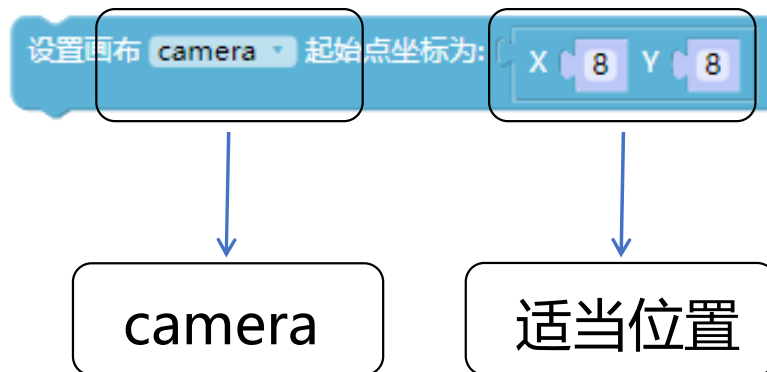
1. 扩展模块 | 屏幕：【设置画布起始点】
2. 扩展模块 | 屏幕：【绘制矩形】
3. 数学运算：【加减乘除】
4. 数学运算：【取商数】

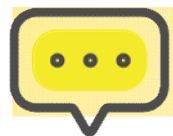




步骤4：绘制颜色采集区域

积木指令讲解：






步骤4：绘制颜色采集区域

参考程序：

设置画布 camera 起始点坐标为: X 8 Y 8

 在画布 camera 上绘制 空心 矩形

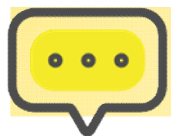
设置起始点坐标:

X: $224 \div 2 - 20 \div 2$ Y: $224 \div 2 - 20 \div 2$

尺寸: 宽 20 高 20

颜色: 红 200 绿 200 蓝 200

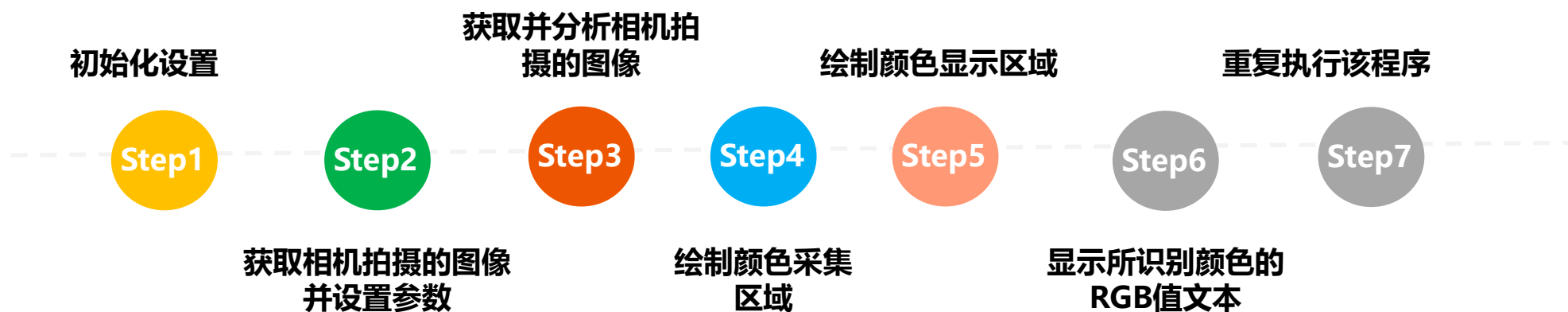
厚度: 1

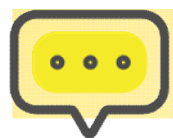


步骤5：绘制颜色显示区域

任务要求：

- 1.在屏幕适当的位置绘制所识别颜色矩形，用以显示所识别的颜色。





步骤5：绘制颜色显示区域

所需积木指令：

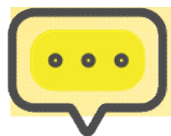
在积木指令区点击以下指令，并依次放在积木编程区：

1. 扩展模块 | 屏幕：【绘制矩形】
2. 扩展模块 | 屏幕：【RGB颜色值设定】
3. 元组：【初始化元组】





什么是元组



编程知识加油站

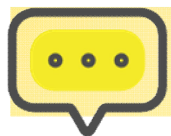
□ 什么是元组 (Tuple) ?

元组本质上是一种有序的集合，它由不同元素组成，每个元素可以储存不同类型的数据，如字符串、数字或元组本身。

表示方法：元组使用括号“ () ”来界定，元组中各元素之间用逗号隔开。例如：

`a=(1, 2, 3)`

`b=(1,)` 当只有一个元素的元组时，需要在这个元素的后面添加英文逗号。

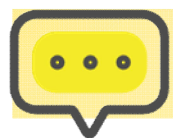


编程知识加油站

▣ 元组的特点

特点：

- 1.元组是有序的集合；
- 2.元组可以存储任意数据类型；
- 3.元组属于不可变序列，不能修改。

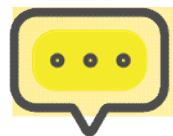


步骤5：绘制颜色显示区域

参考程序：

The image shows a Scratch code block for drawing a rectangle. The block is blue and has a small square icon in the top-left corner. It contains the following text and fields:

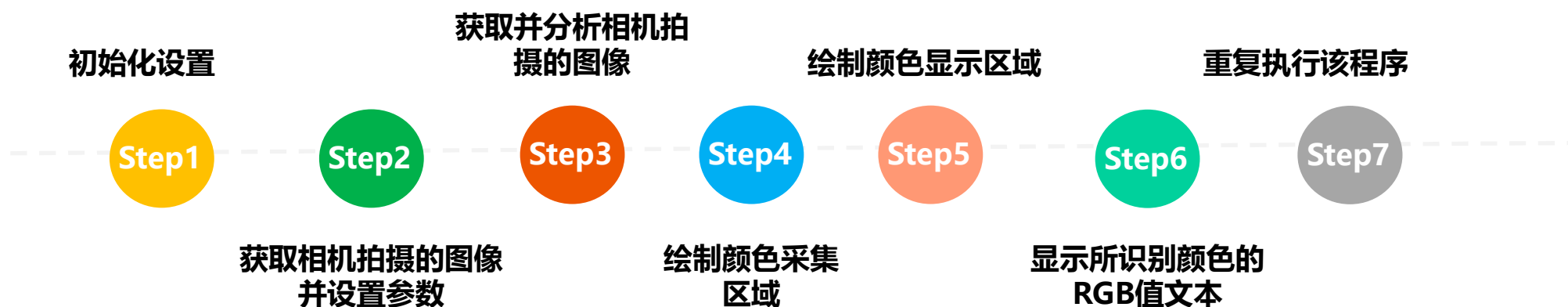
- 在画布 **camera** 上绘制 **实心** 矩形
- 设置起始点坐标:
 - X: **180** Y: **10**
- 尺寸:
 - 宽 **30** 高 **30**
- 颜色:
 - 红 元组 **RGB** 第 **0** 项 绿 元组 **RGB** 第 **1** 项 蓝 元组 **RGB** 第 **2** 项
- 厚度: **1**

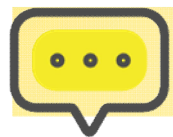


步骤6：显示所识别颜色的RGB值文本

任务要求：

1.在屏幕适当的位置显示RGB值文本。





步骤6：显示所识别颜色的RGB值文本

所需积木指令：

在积木指令区点击以下指令，并依次放在积木编程区：

1. 扩展模块 | 屏幕：【绘制文本】
2. 文字：【转为文本】
3. 变量：【新建RGB变量】

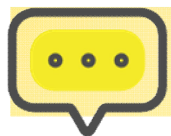




步骤6：显示所识别颜色的RGB值文本

参考程序：





步骤7：重复执行程序

任务要求：

- 1.加入“循环”积木块，让相机持续捕捉并分析图像；
- 2.加入“显示画布”积木块，让所绘制的框、文字显示于画布上。





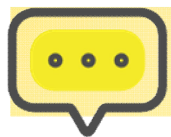
步骤7：重复执行该程序

所需积木指令：

在积木指令区点击以下指令，并依次放在积木编程区：

1. 循环：【一直重复执行】
2. 扩展模块 | 屏幕：【显示画布】





步骤7：重复执行该程序

完整参考程序：

LCD 屏幕初始化

相机初始化 设置捕捉画面的尺寸为: QVGA (320*240) 设置捕捉画面颜色模式为: 彩色

将相机捕捉到的图像尺寸调整至
宽: 224 高: 224 (单位: px)

设置摄像头图像的 对比度 为 0 (范围: -3至3)

一直重复

执行 赋值 camera 为 获取相机捕捉到的图像

赋值 RGB 为 获取画布 camera 中指定区域的颜色分析结果 (RGB) 设置分析区域的起始坐标: X: 224 ÷ 2 - 20 ÷ 2 Y: 224 ÷ 2 - 20 ÷ 2 设置分析区域的尺寸: 宽: 20 高: 20

设置画布 camera 起始点坐标为: X: 8 Y: 8

在画布 camera 上绘制 空心 矩形 设置起始点坐标: X: 224 ÷ 2 - 20 ÷ 2 Y: 224 ÷ 2 - 20 ÷ 2 尺寸: 宽: 20 高: 20 颜色: 红: 200 绿: 200 蓝: 200 厚度: 1

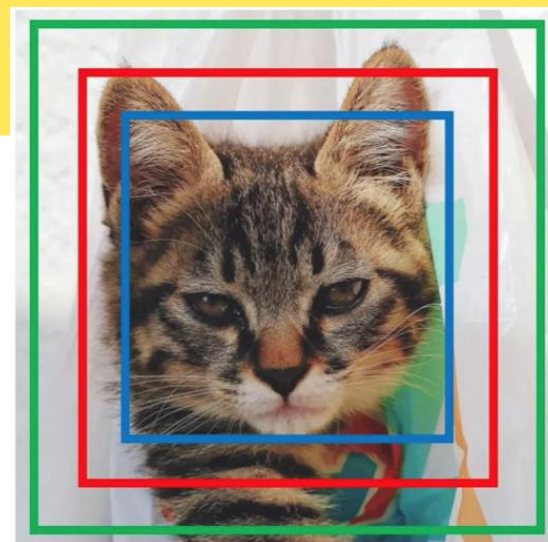
在画布 camera 上绘制 实心 矩形 设置起始点坐标: X: 180 Y: 10 尺寸: 宽: 30 高: 30 颜色: 红: 元组 RGB 第 0 项 绿: 元组 RGB 第 1 项 蓝: 元组 RGB 第 2 项 厚度: 1

Aa 在画布 camera 上绘制文本 设置起始点坐标: X: 10 Y: 10 内容: 建立字符串使用 RGB 颜色: 颜色 文本大小: 1 是否等宽字体

显示画布 camera




为什么需要将输入的图像裁剪为
 224×224 大小？





今天先学到这吧!

认识颜色识别/颜色识别体验



陆吾智能

