

API 接口文档

1. 牛身检测（轮廓点）

接口描述

上传一张包含牛身的图片，进行牛身分割检测，返回牛身的轮廓点、前胸下点等关键信息。适用于需要获取牛只体型轮廓的应用场景。

请求

- **URL:** /view_body_new_service
- **方法:** POST
- **内容类型:** multipart/form-data

请求参数

参数名	类型	必填	说明
file	file	是	待检测的图片文件。支持常见图片格式（JPEG、PNG等）。文件大小不得超过 10 MB （可配置）。

响应

成功响应 (HTTP 200)

```
json
{
  "success": true,
  "message": "检测成功，找到 N 个轮廓，共 M 个点",
  "contour_points": [
    [x1, y1],
    [x2, y2],
    ...
  ],
```

```
    "front_down_point": [x, y],
    "image_width": 1920,
    "image_height": 1080
}
```

字段	类型	说明
success	boolean	固定为 true
message	string	结果描述信息
contour_points	array	所有轮廓点的二维坐标数组，每个元素为 [x, y]（整数）
front_down_point	array	牛身前胸下点的坐标 [x, y]（整数），若未检测到则返回空数组
image_width	integer	原始图片宽度（像素）
image_height	integer	原始图片高度（像素）

未检测到牛身 (HTTP 200)

```
json
{
    "success": false,
    "message": "未检测到牛身，请确认图片角度和清晰度",
    "contour_points": [],
    "image_width": 1920,
    "image_height": 1080
}
```

文件过大 (HTTP 413)

```
json
{
```

```
"detail": "图片过大 12.3MB, 超过限制 10MB, 请在 App 设置中降低质量/宽度"
}
```

图片解析失败 (HTTP 400)

```
json
{
  "detail": "无法解析图片文件（格式不支持或文件损坏）"
}
```

服务器内部错误 (HTTP 500)

```
json
{
  "success": false,
  "message": "服务器错误: {具体错误信息}",
  "contour_points": [],
  "image_width": 0,
  "image_height": 0
}
```

2. 牛身检测 (COCO 格式)

接口描述

上传一张包含牛身的图片，进行牛身分割检测，并将结果转换为 **COCO 数据集格式**（包含 images、annotations、categories）。适用于需要将标注结果导入 COCO 标注工具或用于训练的数据准备。

请求

- **URL:** /coco_view_body_service
- **方法:** POST
- **内容类型:** multipart/form-data

请求参数

参数名	类型	必填	说明
file	file	是	待检测的图片文件。支持常见图片格式。文件大小不得超过 10 MB （可配置）。

响应

成功响应 (HTTP 200)

json

```
{
  "success": true,
  "message": "检测成功，找到 N 个轮廓",
  "images": [
    {
      "file_name": "example.jpg",
      "height": 1080,
      "width": 1920,
      "id": 1
    }
  ],
  "annotations": [
    {
      "segmentation": [
        [x1, y1, x2, y2, ...]
      ],
      "area": 12345.67,
      "iscrowd": 0,
```

```
    "image_id": 1,
    "bbox": [x, y, width, height],
    "category_id": 1,
    "id": 1
  }
],
"categories": [
  {
    "id": 1,
    "name": "cow_body"
  }
]
}
```

字段	类型	说明
success	boolean	固定为 true
message	string	结果描述信息
images	array	COCO images 数组，包含图片基本信息
annotations	array	COCO annotations 数组，每个轮廓生成一个标注项
categories	array	COCO categories 数组，固定类别 cow_body (id=1)

注解字段说明：

- segmentation：轮廓点的一维数组 [x1, y1, x2, y2, ...]，坐标值为浮点数。
- area：轮廓面积（浮点数）。
- bbox：边界框 [左上角 x, 左上角 y, 宽度, 高度]（浮点数）。

- iscrowd: 固定为 0 (单个目标)。
- image_id: 关联的图片 ID (当前固定为 1)。
- category_id: 固定为 1。

未检测到牛身 (HTTP 200)

json

```
{  
  "success": false,  
  "message": "未检测到牛身",  
  "images": [],  
  "annotations": [],  
  "categories": [  
    {"id": 1, "name": "cow_body"}  
  ]  
}
```

文件过大 (HTTP 413)

json

```
{  
  "detail": "图片过大 12.3MB, 超过限制 10MB"  
}
```

图片解析失败 (HTTP 400)

json

```
{  
  "detail": "无法解析图片文件"  
}
```

服务器内部错误 (HTTP 500)

json

```
{
```

```
"success": false,

"message": "服务器错误: {具体错误信息}",

"images": [],

"annotations": [],

"categories": [

  {"id": 1, "name": "cow_body"}

]
```

通用说明

文件大小限制

当前配置的最大上传大小为 **10 MB** (MAX_UPLOAD_MB)。若图片超过此大小，接口会返回 HTTP 413 错误，并提示在 App 中降低质量或宽度。

支持的图片格式

底层使用 OpenCV 解码，支持常见的图片格式（JPEG、PNG、BMP、WebP 等），具体取决于 OpenCV 的编译配置。

性能指标

接口在处理日志中会输出各阶段耗时（读取、解码、检测、总计），方便性能监控。

健康检查

- **URL:** /health
- **方法:** GET
- **响应示例:**

json

```
{

  "status": "ok",

  "model_loaded": true,
```

```
"max_upload_mb": 10  
}
```