

DeepFish 智能鱼类检测系统 API 文档

Base URL: <http://fish.yutiantian.xyz>

SECRET_KEY: 1234@abcd

nc: 13

names: ['AngelFish', 'BlueTang', 'ButterflyFish', 'ClownFish', 'GoldFish', 'Gourami', 'MorishIdol', 'PlatyFish', 'RibbonedSweetlips', 'ThreeStripedDamselfish', 'YellowCichlid', 'YellowTang', 'ZebraFish']

1. 上传图像

上传一张图像用于后续预测。

请求

- URL:** /upload
- Method:** POST
- Headers:**
 - Authorization: Bearer <SECRET_KEY> (仅非 GET 请求需要)
- Body:** multipart/form-data
 - file: 图像文件 (支持 .png, .jpg, .jpeg, .gif)

响应

- 成功 (201):**

```
{  
    "id": 123,  
    "md5_filename": "a1b2c3d4e5f67890.jpg",  
    "message": "Image uploaded successfully"  
}
```

- 错误 (400/500):**

```
{ "error": "描述信息" }
```

2. 获取图像

根据图像 ID 获取原始图像或预测结果图像。

请求

- **URL:** /image/<image_id>
- **Method:** GET
- **Headers:** 无 (GET 请求无需认证)

响应

- 返回图像文件 (如 JPEG/PNG)
- 若图像不存在, 返回 404

3. 执行目标检测预测

对已上传的图像执行鱼类检测，支持三种模型。

请求

- **URL:** /predict
- **Method:** POST
- **Headers:**
 - Authorization: Bearer <SECRET_KEY>
 - Content-Type: application/json
- **Body:**

```
{  
    "image_id": 123,  
    "predict_type": "yolo_public_model"  
    // 可选值: "yolo_public_model",  
    // "faster_rcnn_public_model", "retinanet_public_model"  
}
```

响应

- 成功 (200) (若已存在缓存结果) :

```
{
  "original_image_id": 123,
  "result_image_id": 456,
  "prediction_id": 789,
  "predict_type": "yolo_public_model",
  "detections": [
    {
      "bbox": [
        175.21,
        32.36,
        310.26,
        574.92
      ],
      "class_id": 0,
      "class_name": "AngelFish",
      "confidence": 0.7532
    },
    {
      "bbox": [
        315.22,
        14.06,
        619.99,
        570.67
      ],
      "class_id": 0,
      "class_name": "AngelFish",
      "confidence": 0.7191
    }
  ],
  "message": "Prediction result retrieved from database"
}
```

- **成功 (200) (新预测完成) :**

```
{
  "original_image_id": 123,
  "result_image_id": 456,
  "prediction_id": 789,
  "predict_type": "faster_rcnn_public_model",
  "detections": [...],
  "message": "Faster R-CNN prediction completed successfully"
}
```

- **错误 (400/404/500):**

```
{ "error": "描述信息" }
```

说明：系统会自动缓存预测结果。对同一 `image_id` 和 `predict_type` 的重复请求将直接返回数据库中的结果。

4. 获取所有预测记录列表

获取系统中所有已完成的预测任务列表，支持多条件筛选与分页。

请求

- **URL:** `/predictions`
- **Method:** GET
- **Headers:** 无 (GET 请求无需认证)
- **Query 参数 (均为可选) :**
 - `id` : 预测记录 ID
 - `image_id` : 原始图像 ID
 - `result_image_id` : 结果图像 ID
 - `predict_type` : 预测模型类型 (如 `yolo_public_model` 等)
 - `page` : 页码 (默认 1)
 - `per_page` : 每页数量 (默认 20, 最大 100)

响应

- **成功 (200):**

```
{  
    "page": 1,  
    "per_page": 20,  
    "total": 123,  
    "total_pages": 7,  
    "predictions": [  
        {  
            "id": 789,  
            "image_id": 123,  
            "result_image_id": 456,  
            "predict_type": "yolo_public_model",  
            "result_json": { ... },  
            "created_at": "2025-11-05T08:30:00Z"  
        }  
        // ...更多记录  
    ]  
}
```

- **错误 (500):**

```
{ "error": "Database error: 错误描述" }
```

说明:

- 支持按 ID、image_id、result_image_id、predict_type 多条件筛选。
- 支持分页，per_page 最大为 100。
- result_json 字段为预测结果的详细内容（对象或数组）。

错误码说明

状态码	含义
400	请求参数缺失或格式错误
401	缺少或无效的 Authorization 头
404	图像 ID 不存在或文件丢失
500	服务器内部错误（如模型加载失败、数据库异常等）

注意:

- 所有非 GET 请求必须携带有效的 Authorization: Bearer <token> 头，其中 <token> 由服务端 .env 中的 SECRET_KEY 定义。
- 图像和预测结果均以 MD5 哈希命名，确保唯一性和去重。