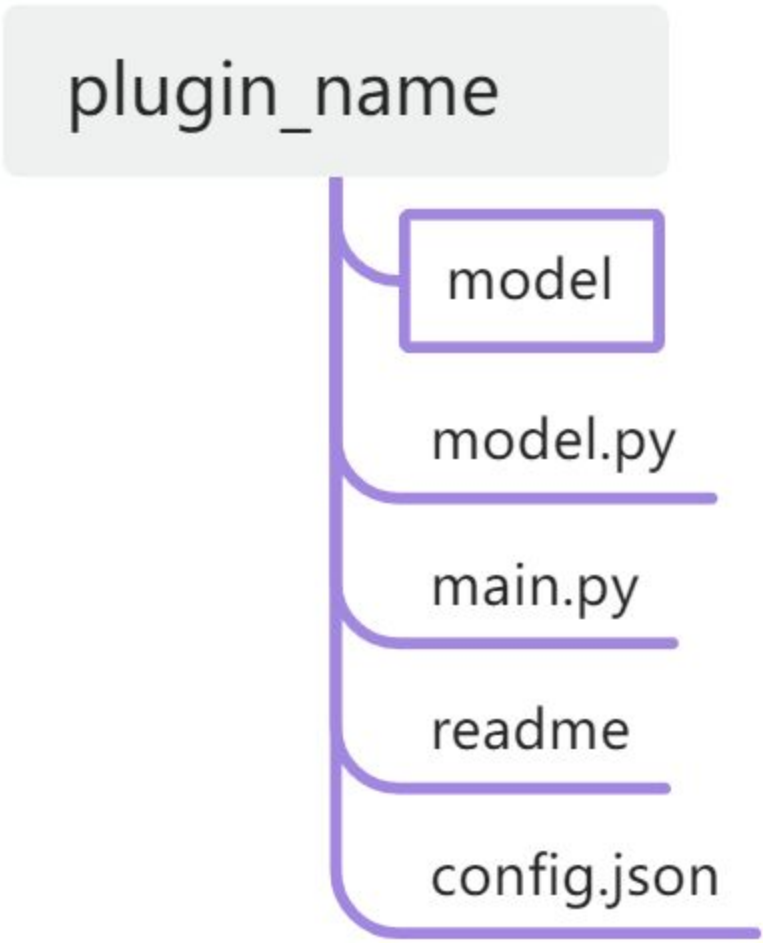
算法插件开发手册

1. 算法插件



- plugin_name：插件名称，不可重复
- model：文件夹，存放模型文件，例如yolo的模型yolov8n.pt
- model.py：编写模型加载代码，必需的文件，不可修改文件名称，必须包含方法def load()，load方法的返回值将被传入main.py的detect方法的model参数中，

  示例如下：

model.py Python

```python
from ultralytics import YOLO
from pathlib import Path


def load(algo_config: dict):
    """模型加载逻辑
            Args:
                algo_config: 算法config.json文件中的内容
    """
    file_path = Path(__file__).resolve()
    model = YOLO(file_path.parents[0] / 'model' / 'yolov8n.pt')
    return model

```

- main.py：编写算法代码，必需的文件，不可修改文件名称，必须包含方法def detect(model, frame: cv2.Mat, context:dict)

  示例如下：

main.py Python

```python
import cv2

def detect(model, frame: cv2.Mat, context:dict):
    """
    :param model:
            model.py的load方法的返回值
           frame:
            一个NumPy数组，它包含了从视频流中读取的当前帧的图像数据
           context:
            上下文内容，可能包括的字段：
                config：插件config.json的内容
                options：java端对此算法的配置信息
                threshold：java端为该算法配置的阈值，不同的算法会有不同的阈值配置方式，阈值的默认值可能在config.json中配置
                detect_area：java端为该算法配置的框选范围，坐标是都是百分比

    :returns:
        可以为空
    """
    results = model(frame)
    annotated_frame = results[0].plot()
    cv2.imwrite(f'detect.jpg', annotated_frame)
```

- config.json：写入算法插件的配置信息，采用json格式。

字段说明：

version字段：必填，写入算法插件的版本号，格式必须是x.y.z，每个部分都是非负整数，不包含其他内容。

author字段：选填，开发者。

description字段：选填，填入算法插件的描述信息，此信息会在上传的时候保存到java端。

threshold字段：选填，算法默认阈值，不同算法有不同的含义。

dev_id字段：选填，推理卡或TPU的id。

其他字段：可自行添加其他自定义字段。

示例如下：

config.json Markdown

```
{
  "version": "1.0.0",
  "author": "gc",
  "description": "This is a demo plugin for Sight.",
  "threshold": [
    0.25,
    0.7
  ],
  "dev_id": 0
}
```

- README.MD：编写算法插件的readme文档，算法描述，可不写

README.MD Markdown

```
这是一个markdown格式的说明文档
```

2. 算法插件打包