# 1Panel + Docker 一键部署 Dify 及RTSP视频帧捕获API服务

## 第一部分：RTSP视频帧捕获API服务

这是一个基于FastAPI的服务，可以通过API端点按需从RTSP流（如海康威视摄像头）中捕获视频帧。

### 系统要求

* Python 3.7+
* 依赖项见requirements.txt文件

### 配置说明

在运行服务前，您需要在app.py中配置您的摄像头：

1. 打开app.py文件
2. 找到CAMERAS字典
3. 编辑该字典，添加您的摄像头名称（作为键）和对应的RTSP地址（作为值）。例如：
4. CAMERAS = {
5. "滇池": "rtsp://您的用户名:您的密码@摄像头IP1:554/stream1",
6. "keting": "rtsp://您的用户名:您的密码@摄像头IP2:554/stream1",
7. }

### 安装步骤

1. 安装依赖项：
2. pip install -r requirements.txt

### 运行服务

在终端中运行以下命令：

uvicorn app:app --host 0.0.0.0 --port 8000

### API接口说明

#### 按名称获取视频帧

* GET /get\_frame\_by\_name?name=<摄像头名称>
  + 从预配置的摄像头名称获取JPEG格式的视频帧
  + **示例：** http://localhost:8000/get\_frame\_by\_name?name=dzy

#### 按URL获取视频帧（旧版）

* GET /get\_frame?rtsp\_url=<RTSP地址>
  + 从指定的RTSP流URL获取JPEG格式的视频帧
  + **示例：** http://localhost:8000/get\_frame?rtsp\_url=rtsp://admin:password@摄像头IP:554/Streaming/Channels/101

## 第二部分：用 1Panel 在线安装 Docker（推荐）

适用于：Linux 服务器（Ubuntu 20/22/24、Debian 11/12/13 等） 目的：先通过 1Panel 快速装好 Docker，再用官方 docker-compose 一键拉起 Dify，全程 5-10 分钟。

1. 以 **root** 身份登录服务器：
2. sudo -i
3. 执行 1Panel 官方一键脚本：
4. bash -c "$(curl -sSL https://resource.fit2cloud.com/1panel/package/v2/quick\_start.sh)"
5. 按提示完成安装：
   * 默认会装最新稳定版 Docker & Docker-Compose；
   * 安装完成会打印类似：
   * 1Panel 访问地址：http://<服务器IP>:<端口>/<安全入口>
   * 用户名：admin
   * 密码：\*\*\*\*\*\*
6. 浏览器打开上述地址，登录 1Panel 后台 → **容器** → **Docker** 即可图形化管理。

如果 Docker 安装失败，可单独执行备用脚本：

bash <(curl -sSL <https://linuxmirrors.cn/docker.sh>)

### 使用 Dify 官方 Docker-Compose 部署

### Dify 官方 Docker-Compose 部署

* 1. **回到服务器终端，**克隆官方仓库
  2. **下一步可能需要魔法上网**
  3. git clone https://github.com/langgenius/dify.git
  4. cd dify/docker
  5. 复制并编辑环境变量：
  6. cp .env.example .env
  7. # 按需修改 .env，如端口、密钥、模型配置等
  8. vim .env
  9. 一键启动：
  10. docker compose up -d
  11. 查看容器状态：
  12. docker compose ps
  13. 所有容器状态为 healthy 即成功。
  14. 浏览器访问：
  15. http://<服务器IP>:80

首次打开会进入初始化向导，按提示创建管理员账号即可。

1. 配置Dify
2. 点击右上角配置模型供应商

屏幕的手机截图

AI 生成的内容可能不正确。 图形用户界面, 应用程序, 网站

AI 生成的内容可能不正确。

1. 选择自己的模型推荐gemini
2. 返回首页（工作室）点击导入 DSL 文件
3. 剩下的Dify应用中有说明