
Synexens Python3 使用说明

日期	对应 SDK 版本	描述	作者
20221214	v1.0.0.0	初始版本	gjl
20240112	v4.1.1.0	格式修改	ysy
20240229	v4.1.3.0	内容更新	gjl

目录

1. 概述.....	3
2. 环境依赖.....	3
3. 代码结构描述.....	3
4. 使用步骤.....	4
5. 其他平台.....	4
免责声明.....	5

1. 概述

支持设备: cs20 单频 cs20 双频 cs30 单频 cs30 双频

支持系统: windows, Ubuntu

2. 环境依赖

Python3.9 及以上

依赖模块: ctypes, numpy, opencv

3. 代码结构描述

一共包含有 3 个 python 代码文件

3.1、SynexensPythonSDK.py

该文件使用 ctypes 来加载外部动态库, 并对 c++ 动态库的接口进行了二次封装。

3.2、SYPythonDataDefine.py

该文件包含了 SDK 中所需的枚举及结构体

3.3、SynexensTest.py

该文件 SDK demo, 包含了基本的功能, 具体使用方法可以借鉴 c++ sdk demo。

opencv 的键值在 windows 及 ubuntu 中有所不同, 可根据系统自行修改, 对应的键

值在 SynexensTest 中注释说明。

4. 使用步骤

将 PythonSDk 下载到本地后，将其解压。内部包含一个 requirements.txt 文件。可以通过 `python install -r requirements.txt` 来安装所需的依赖包。

通过 `python SynexensTest.py` 来执行。

5. 其他平台

在 Linux 下，还需要将 so 文件夹放于终端的环境变量中。方法如下

添加当前用户当前终端的环境变量-临时

```
export LD_LIBRARY_PATH=/home/{用户名}/... #.so file path
```

添加当前用户的环境变量修改 ~/.bashrc 文件，在其末尾，添加环境变量

```
vim ~/.bashrc
```

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/home/{用户名}/...
```

使其生效：

```
source ~/.bashrc
```

然后在终端键入 `python SynexensPythonSDK.py` 来运行。

注意：本程序仅为 SDK 在 Python 开发语言下的调用 Demo

免责声明

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利,它们可能由更新之信息所替代。确保应用符合技术规范,是您自身应负的责任。本公司对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保,包括但不限于针对其使用情况、质量、性能、适销或特定用途的适用性的声明或担保。本公司对因这些信息及使用这些信息而引起的后果不承担任何责任。未经本公司书面批准,不得将该产品用作生命维持系统中的关键组件。