**iDT源码说明**

1. **文件简介：**

Video.cpp：测试OpenCV的函数。

OpticalFlow.h：定义了Farneback光流算法的相关函数。

Initialize.h：定义了初始化相关函数以及命令行参数解析函数。

Descriptors.h：定义了描述子、轨迹计算以及相机运动相关函数。

DenseTrackStab.h：定义了一些参数并初始化（如起始帧、结束帧、尺度变化数目及程度），定义了一些结构体（如视频信息、描述子信息等），定义了轨迹类、人体框类、帧类。

DenseTrackStab.cpp：主函数。

1. **功能描述：**

通过对视频数据进行密集采样得到轨迹信息，提取相应的特征描述子，输出最后的特征数据。

1. **输入说明：**

直接调用DenseTrackStab +视频文件路径 +相应参数

参数说明：-h 信息帮助

-S+参数 从第几帧开始计算（默认为0）

-E+参数 从第几帧结束计算（默认为最后一帧）

-L+参数 轨迹的长度（即多少帧后重新采样，默认为15帧）

-W+参数 采样间隔（默认为5像素）

-N+参数 计算描述子时周围取样的范围（默认为32像素）

-s+参数 在轨迹上构建时空体时空间上采样的正方形的大小（默认为2）

-t+参数 在轨迹上构建时空体时时间上采样的数目（默认为3）。

-A+参数 scale num（默认为8）。

-H+参数 人体框数据。

实际工程操作建议将从命令行输入获得文件路径做相应修改，即建立一个目标文件路径的txt文件，读文件流操作，或者使用vi脚本。

1. **处理过程说明：**

读视频的某一帧，然后根据该帧和上一帧计算投影变换矩阵，对当前帧进行变换后和上一帧计算光流信息，跟踪轨迹获得特征，循环直到计算完所有帧的特征信息。

1. 输出说明：

输出为直接将特征print到屏幕上，实际操作时建议使用文件流输出到txt文件或直接使用linux下的‘>’将输出保存到文件。

1. **操作环境说明：**

Linux系统选取最新版本即可，ffmpeg选取0.11.1，opencv选取2.4.2（不能用更高的opencv3，3较2有很大改动）。配置环境时要先安装ffmpeg，再安装opencv（由于安装opencv时要与ffmpeg联合编译，因此次序不能变）。

**a、ffmpeg安装**

安装ffmpeg时先安装x264解码器。解压源码后。

Sudo ./configure --enable-static --enable-shared

Sudo make

Sudo make install

Sudo ldconfig

然后安装必要的支持

make automake g++ bzip2 python unzip patch subversion ruby build-essential git-core checkinstall yasm texi2html libfaac-dev libmp3lame-dev libopencore-amrnb-dev libopencore-amrwb-dev libsdl1.2-dev libtheora-dev libvdpau-dev libvorbis-dev libvpx-dev libx11-dev libxfixes-dev libxvidcore-dev zlib1g-dev（可以一个一个sudo apt-get install检查有没有安装）

最后进行编译.

Sudo /configure --prefix=/usr/local/ffmpeg --enable-gpl --enable-nonfree --enable-version3 --enable-shared --enable-libopencore-amrnb --enable-libopencore-amrwb --enable-libfaac --enable-libmp3lame --enable-libx264 --enable-libxvid --enable-libvpx（这个不能一个一个来）

Sudo make

Sudo make install

Sudo Idconfig

**b、opencv2.4.2安装**

首先安装相应的支持（pkg-config和cmake）

apt-get install pkg-config

export PKG\_CONFIG\_PATH=$PKG\_CONFIG\_PATH:/usr/local/lib/pkgconfig

apt-get install cmake

然后安装opencv，进入opencv2.4.2文件夹。

Sudo mkdir release

cd release

sudo cmake -D CMAKE\_BUILD\_TYPE=RELEASE -D CMAKE\_INSTALL\_PREFIX=/usr/local -D BUILD\_PYTHON\_SUPPORT=ON ..

最后设置环境变量，用vim打开/etc/ld.so.conf.d/opencv.conf，结尾添加/usr/local/lib，然后ldconfig更新配置。

**c、iDT运行**

解压源码，直接make，就可以调用release里面的DenseTrackStab了，具体使用参数见输入说明。