进展报告8.11

1. **无人机方面**
2. **目前的目标**：

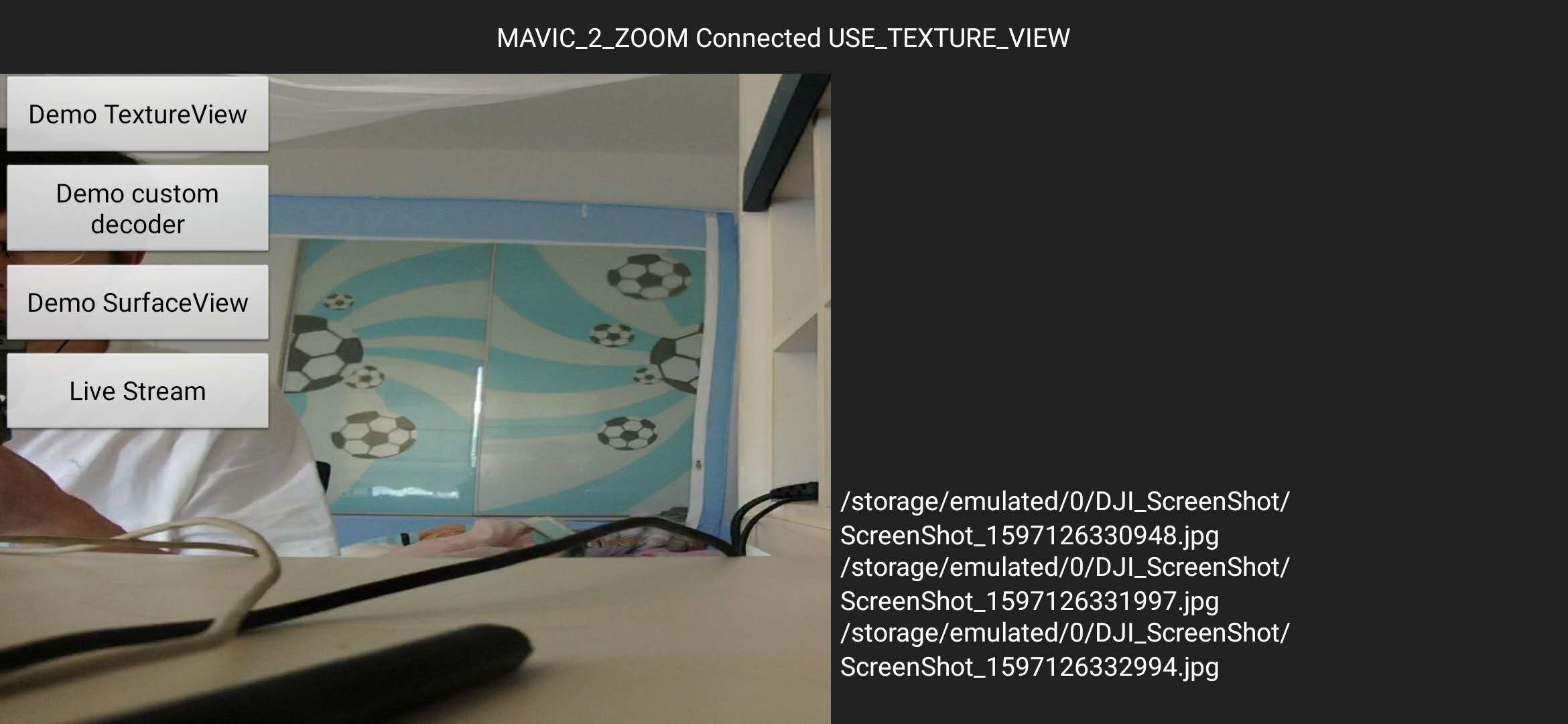
读取无人机摄像头的视频流，并使用计算机视觉技术对视频流进行处理。

1. **进展：**

可以读取无人机摄像头视频流，并对得到的图像帧进行简单处理。

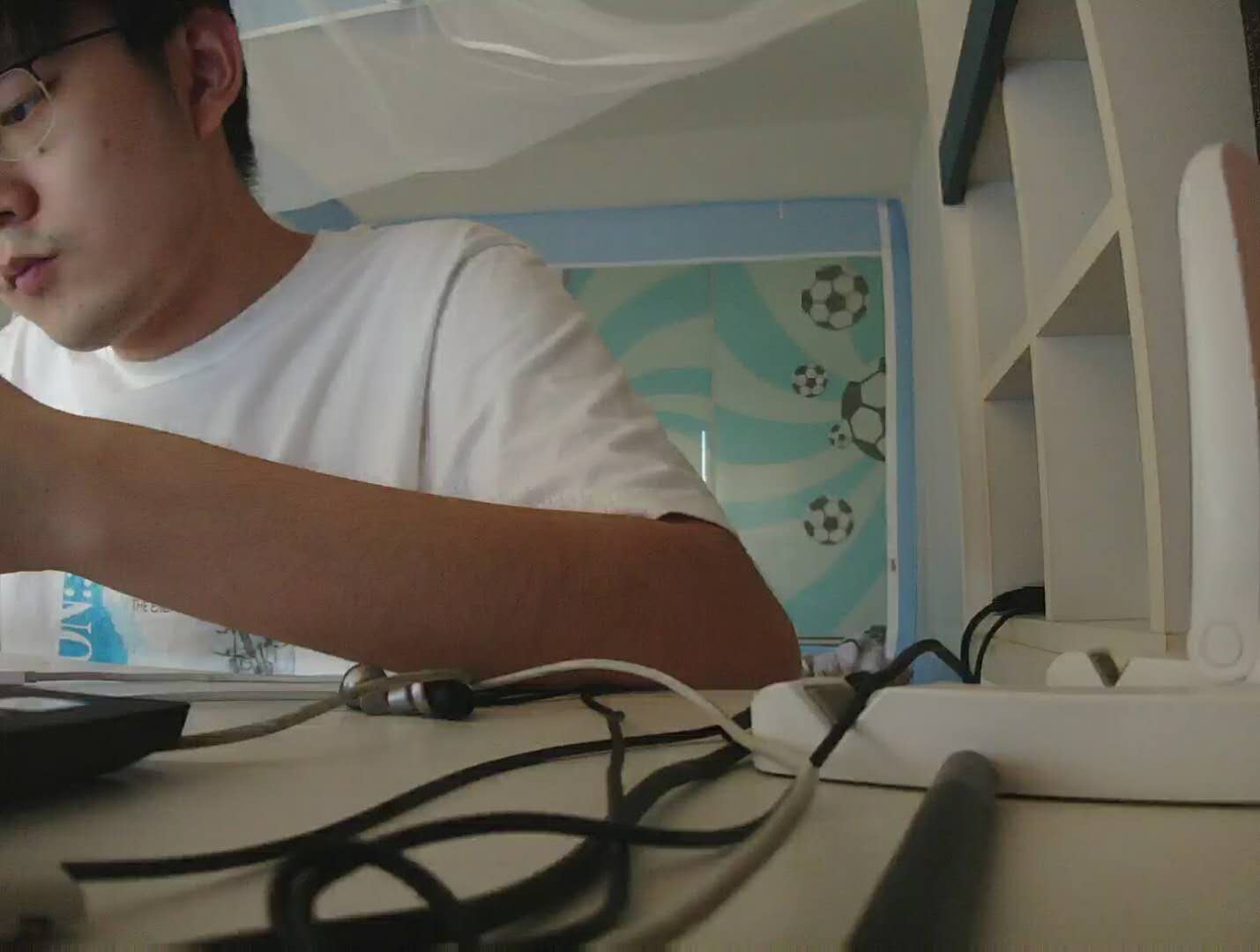
1. **详细介绍：**

如图是一个demo应用的界面：



显示了无人机摄像头的实时画面，左边的按钮可以对当前视频流取一个图像帧进行预览，最下面的按钮是screen shot，点击按钮开始对当前视频流每30帧进行截屏，截图保存到手机目录。

如图是正常情况下得到的截图画面：



其原理是通过实现一个视频解码器，监控无人机摄像头信息，并将得到的原始信息编码解码为yuv格式的图像帧流进行保存，然后在screenShot函数中将yuv格式的图像帧转为jpg图像保存到手机本地。

我尝试了直接对yuv图像进行简单的处理，由于yuv格式存储图像的方式是连续存储，在保存截图的时候直接将内存中的yuv图像前面某一段像素值赋值为1，得到如下的截图：



于是现在已经知道了读取并解码无人机摄像头得到的图像帧的存储数据结构，可以成功地对其进行简单的处理。

通过查阅资料，对于yuv格式的图像进行计算机视觉项目，如目标识别，人脸探测等，通常是引入openCV库，将其转为mat格式的二维数组进行处理，并可以直接调用openCV库中集成的人脸探测等函数，我进行了openCV的安装，但还没有完成调用openCV库进行处理的demo应用，网络上有一些例子可以使用openCV调用手机摄像头进行处理，我需要考虑利用这些例子的方法结合到我们的无人机图像处理代码中。

1. **下一步计划：**

引入openCV库，对摄像头视频流的图像帧使用计算机视觉等方法进行处理。

通过openCV库，实现一个简单的人脸探测demo，即对摄像头得到的实时画面，使用openCV库中自带的人脸探测函数，如CascadeClassifier分类器等，探测人脸部分，在实时视频中用红框实时显示，并截取人脸进行截图保存。

1. **Sample model checker**
2. **现在的目标**：

使用cpaChecker或llvm得到程序的CFG，并能够打印其节点与状态迁移边等。

1. **进展：**

阅读了cpaChecker与llvm源码中关于CFG构造的部分，学习如何自己写代码得到程序CFG。

1. **详细介绍**：

cpaChecker可以将程序的CFA（程序控制流自动机）保存为可视化的.dot文件，于是重点阅读了cpachecker/cfa/CFA.java等代码文件，发现在cpachecker/cfa/model/CFAEdge.java与cpachecker/cfa/model/CFANode.java文件中定义了CFA的边与节点，而cpachecker/cfa/CParser.java文件中定义了从C语言创建CFA的函数，但是我目前还不知道怎么样通过这些函数与结构自己实现代码读取C程序得到CFA。

llvm项目是一系列分模块、可重新用的编译工具链，它提供了一种代码的中间表示（IR），并且以库（Library）的方式提供了一系列接口可以对IR进行操作，llvm首先将程序翻译为IR，然后我们可以用llvm pass对程序IR进行遍历和修改。我现在需要重点关注的是如何编写llvm pass进行程序分析。

llvm提供了命令opt-dot-cfg可以将程序源码IR的CFG保存为可视化的.dot文件，我目前希望拿到CFG的一些数据进行分析和计算，于是需要自己编写pass，利用llvm的function类提供的接口，对程序基本块进行操作，现在正在阅读llvm/lib/Analysis下的CFG.cpp和CFGPrinter.cpp这两个文件，学习如何得到程序CFG。

1. **本周计划**：

学会如何编写llvm pass获取打印程序CFG。