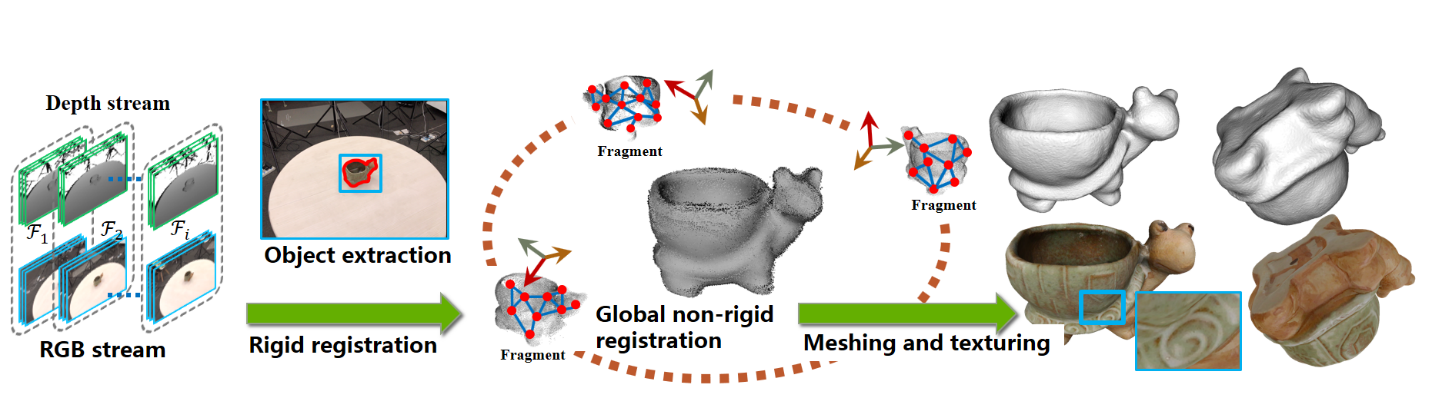
## 小物体重建算法说明文档

1. **系统简介**

本系统设计了一套完整的小物体重建算法，该算法利用彩色相机和深度相机等传感器，实现了对常见小物体的数字化三维重建。该算法能够捕捉和重建物体的三维形状和外观，以支持在数字化空间模拟三维的交互与感知。

本系统主要包括4个模块：1. 物体深度数据提取模块，2. 相机位姿估计和模型片段融合模块，3. 全局非刚体注册模块，4. 网格提取和纹理生成模块。其中前面三个模块是在线实时完成的，最后的网格提取和纹理生成模块是在点云模型生成之后离线后处理完成的。

图1 流程展示

1. **主要模块流程图**

输入彩色深度视频流

物体深度数据提取

相机位姿估计和模型片段

融合

全局非刚体注册

关键帧相机位姿更新

表面网格提取和纹理贴图

是

建立新的段

t<=T

否

否

全局非刚体注册

关键帧相机位姿更新

是

1. **各模块简介**

### 模型表达

本系统在实时在线处理阶段使用Surfel进行模型表达，Surfel为Surface Element的缩写，其将模型表达为一系列的圆形面片，每一个三维圆包含了颜色、位置、法向、半径、置信度信息。相机位姿的估计和模型片段的融合过程即为将深度图注册到当前的Surfel模型上，然后将深度图融合入当前的Surfel模型中。注册过程即将Surfel模型视为目标点云，深度图像视为源点云，利用简单的刚体ICP算法估计一个刚体变换矩阵。融合过程是判断每个深度点对应的三维点是否属于某一个圆形面片，如果是，则将该三维点融合入该圆形面片，更新圆形面片位置，置信度加一，如果该三维点不属于某一个圆形面片，则以该三维点作为中心生成一个新的圆形面片。在扫描融合过程中，随着时间推移，置信度低的圆形面片会被丢弃。

为了完成全局非刚性注册任务，我们修改了前述的Surfel模型表达，进一步增加了模型的自由度，将模型表达为一系列的具有重叠的Surfel模型片段。模型片段的数目随着扫模型片段在扫描过程中在线生成，全局非刚性注册的目标即通过全局优化算法在扫描过程中逐步减小模型片段的不一致性，提高扫描的鲁棒性。

### 输入模块

本系统输入为彩色深度图片数据流，彩色深度图片以某种方式保持同步。输入模块相关代码均在xSensors子目录下，本系统支持不同深度摄像头作为输入，包括华硕Xtion Pro、因特尔RealSense、微软Kinect1（2）、Structure Sensor。其中Structure Sensor是挂载在Ipad上的，彩色图像由Ipad摄像头直接获取，在Ipad端获取数据并进行压缩，利用无线WIFI实时传输到PC端，处理完成后将当前结果传回到Ipad端。不同的设备仅需要继承xCapture类并实现其数据获取方法即可。

### 物体深度数据提取

物体深度数据提取模块相关代码均在xObjTrack子目录下，主要功能由xObjectTrackor类实现。分为两个子模块：物体包围盒跟踪，平面检测及支撑平面删除。实现函数：

void xObjectTrackor::trackOpt1(…)

对于新的一帧数据，我们首先利用前一帧的物体包围盒进行深度非物体深度数据的剔除和平面提取，然后基于深度数据进行候选物体包围盒的提取，此时的目标物体可能包含多个候选物体包围盒，最后利用图像特征的匹配进行前一帧物体包围盒和当前帧候选包围盒的匹配，将匹配的多个候选包围盒进行合并，作为当前帧的物体包围盒。

### 相机位姿估计和模型片段融合

相机位姿估计和模型片段融合模块相关代码均在xSurfelFusion子目录下。主要功能由xSurfelFusion类实现。实现函数：

void xSurfelFusion::processFrame2(…)

其中相机位姿估计的实现函数为：

void xSurfelFusion::trackCamera(…)

模型片段融合的实现函数为：

void xSurfelFusion::dofusion(…)

对于物体深度数据提取之后的深度图片和彩色图片，利用RGBD-ICP算法进行相机位姿的计算，首先将当前模型投影到上一帧相机位姿获取模型深度图和彩色图，然后利用两帧深度图和彩色图计算相机位姿的变化量，将其乘到上一帧相机位姿上，即获得了当前帧相机位姿。

对于模型片段融合，Surfel的融合主要利用OpenGL的Feedback Buffer机制实现，主要原理如模型表达一节中所述。需要注意的是前面的投影是对于所有模型片段进行的，而这一步的模型融合仅仅针对当前模型片段。实现的时候每m帧在线融合作为一个模型片段。

### 全局非刚体注册

相机位姿估计和模型片段融合模块相关代码均在xDeformation子目录下。主要功能由xDeformation类实现。实现函数：

void xDeformation::deform(…)

在扫描过程中将所有帧按照每m帧一个段进行切分，段内根据深度数据、彩色数据进行相机位姿估计、局部融合，段间进行全局非刚体注册。在时刻t，当前模型由模型片段构成，用表示

模型片段包含了Surfel集合、Embedded Deformation（ED）图、关键帧彩色图像和关键帧相机位姿；其中，Surfel集合由帧物体深度融合而成。和分别为非刚性变换后的三维位置和法线向量。ED图中的节点代表物体表面局部的刚性变换，同步的节点具有不同的刚性变换。因此，整个物体可以进行非刚性变换。物体表面的每个顶点和ED图的5-7个节点相连接，其变换后的位置和法向，和，分别为经过这5-7个节点的刚性变换后的线性组合而得的三维位置和法线向量：

非刚性全局优化的优化对象是所有的ED图中的控制点局部刚体变换，优化目标是最小化能量函数：

其中，为各能量项的权重；上述优化目标是一个非线性最小二乘问题，使用高斯-牛顿迭代法进行求解，固定迭代步数为5。为了方便计算雅可比矩阵，将需要更新的法向量和关键帧相机位姿初始化为和，在每一轮迭代中假设和不变，在计算计算雅可比矩阵时不进行求导，然后在每一步迭代之后更新法向量和关键帧相机位姿。

### 表面网格提取和纹理贴图

在获取了Surfel表达的模型之后，我们使用经典的Poisson Surface Reconstruction算法将其三角化为网格，然后使用TexRecon算法将一系列彩色图片（每个模型片段选取一幅）合成一张模型的纹理贴图。

1. **结果展示**

本系统实现了小物体实时360度重建，并且可以支持高质量的纹理贴图。

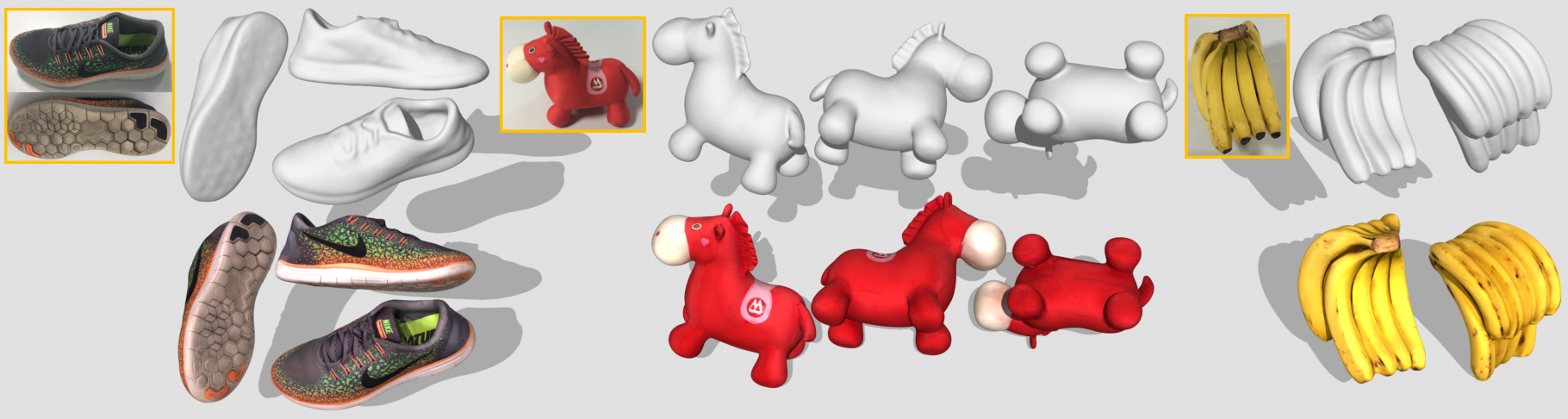


图2 360度三维重建结果展示

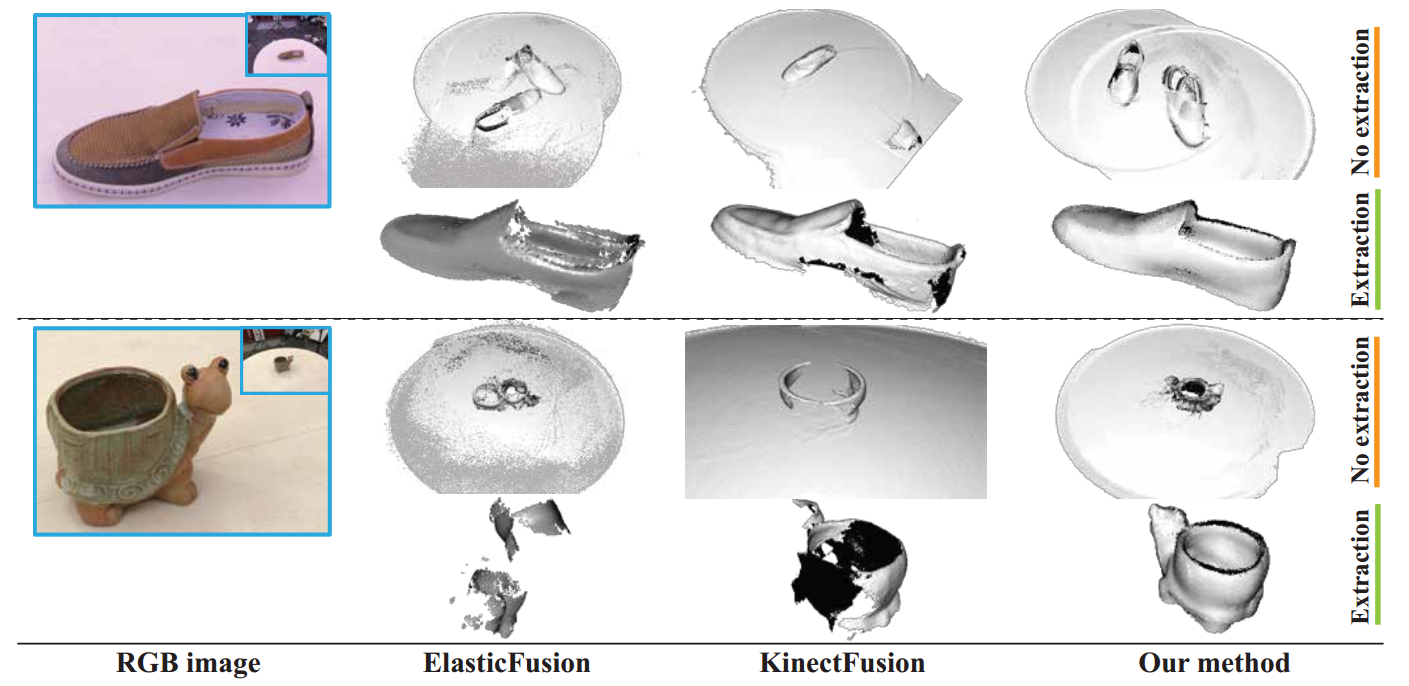


图3 非刚性全局注册三维重建与现有方法对比结果

下表是算法在所重建的物体上的统计表格，从左到右分别为深度数据帧数、物体大小、物体分割所用时间、刚性注册时间和全局非刚性注册算法（Global non-rigid registration）时间及生成网格和纹理的时间。可以看到的全局非刚性速度可以达到0.8ms每次迭代，可在实时重建中应用。

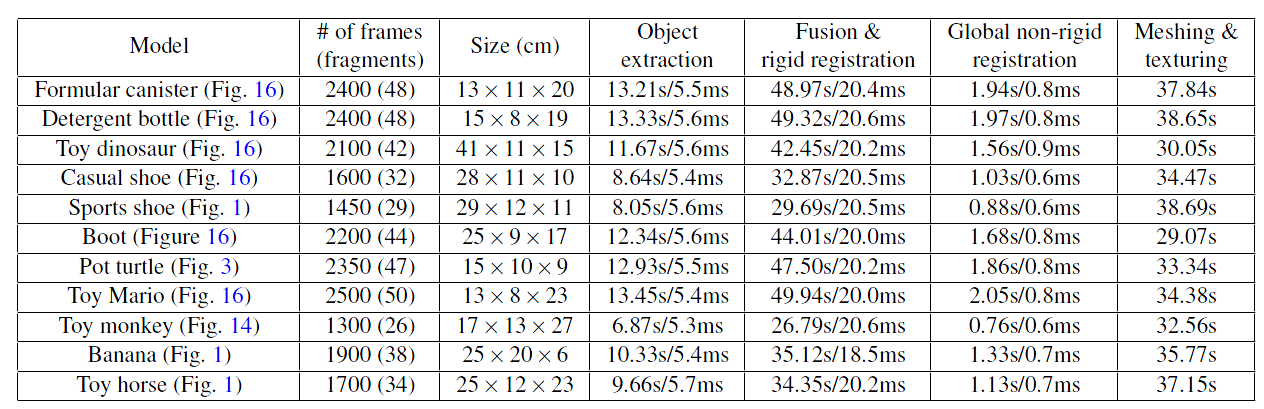
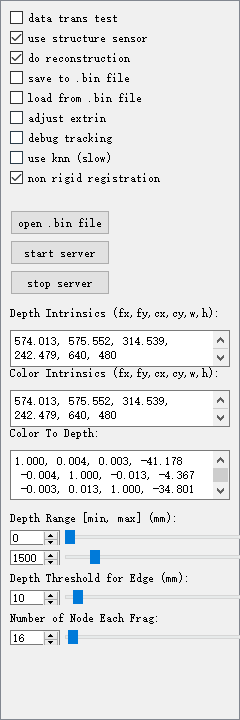


表1 不同模型实时重建对比

1. **程序使用说明**

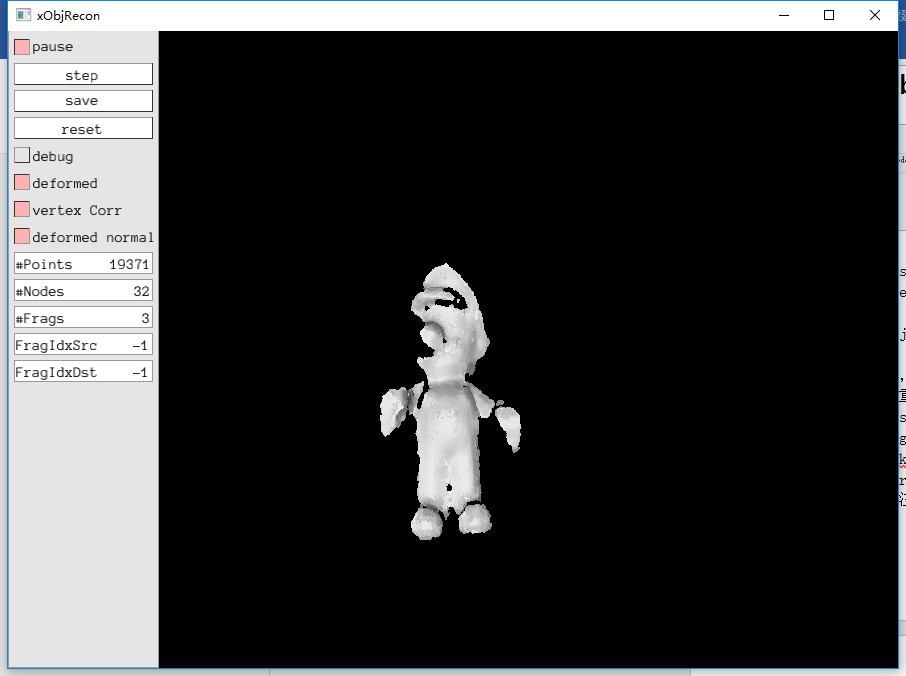
小物体重建程序有两部分组成：

1. PC端程序xObjRecon；
2. iPad端程序xObjReconCapture。
3. **PC端程序**
   1. **PC端界面：**

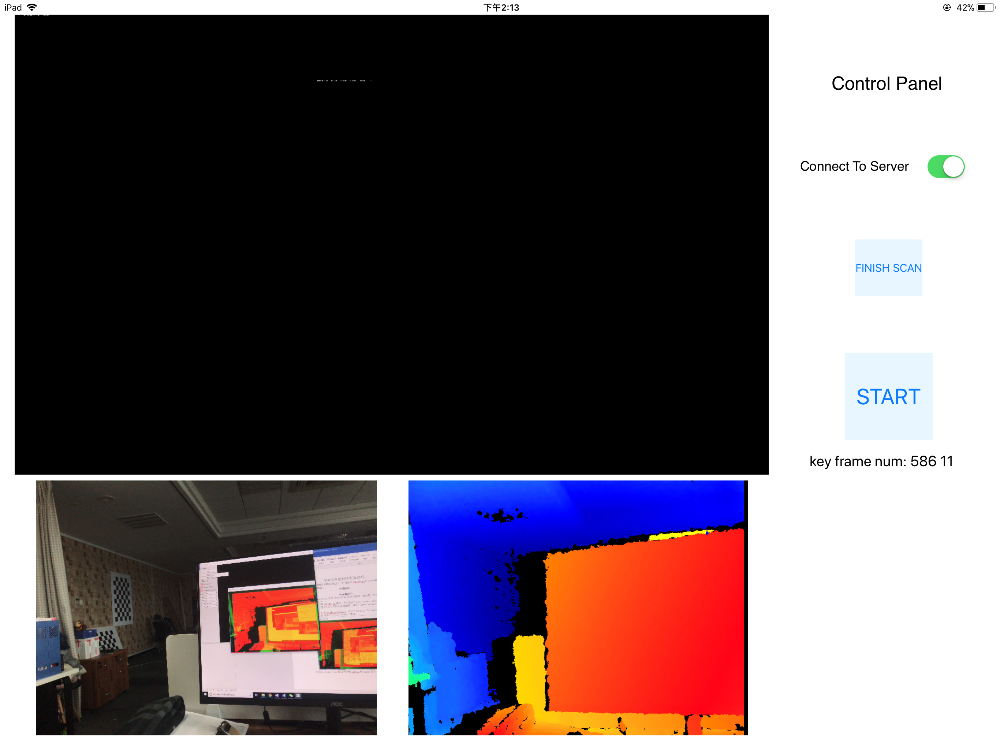


* 1. **配置说明**

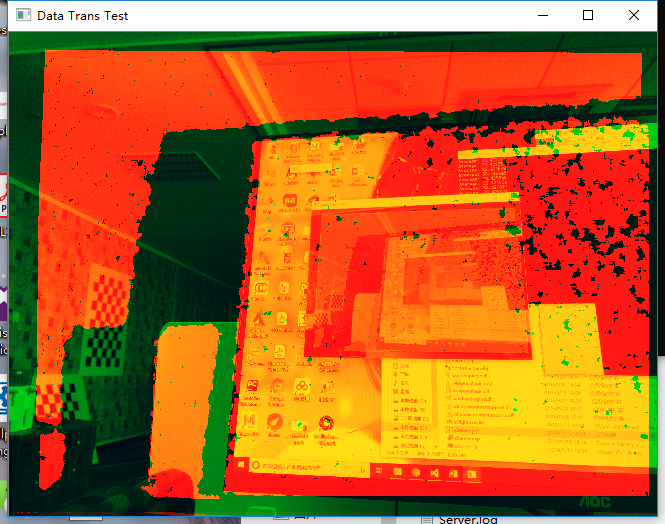
1. data trans test：用于数据传输检测测试
2. use structure sensor（默认选择）：表示使用structure sensor
3. do reconstruction(默认选择)：表示进行重建
4. save to .bin file：将拍摄的数据保存为.bin文件，保存目录为C:\xjm\snapshot\
5. load from .bin file：勾选该选项，点击“open .bin file”选择文件目录，再点击“start server”按钮后，将使用之前保存的.bin文件进行重建
6. adjust extrin：手动微调外参
7. debug tracking：对tracking步骤进行debug
8. use knn：使用knn寻找最近邻（默认通过投影找点对）
9. non rigid registration(默认选择)：应用非刚性注册，不选择时应用刚性注册
10. open .bin file：选择文件目录
11. start server：启动server端
12. stop server：关闭server端
13. Depth Intrinsics：在窗口设置深度内参（暂未实现）
14. Color Intrinsics：在窗口设置彩色内参（暂未实现）
15. Color To Depth: 在窗口设置color对齐到depth的转换矩阵（暂未实现）
16. Depth Range：拍摄的深度范围
17. Depth Threshold for Edge：深度边缘检测的阈值
18. Number of Node Each Frag：每个fragment的Node数
    1. **渲染界面**



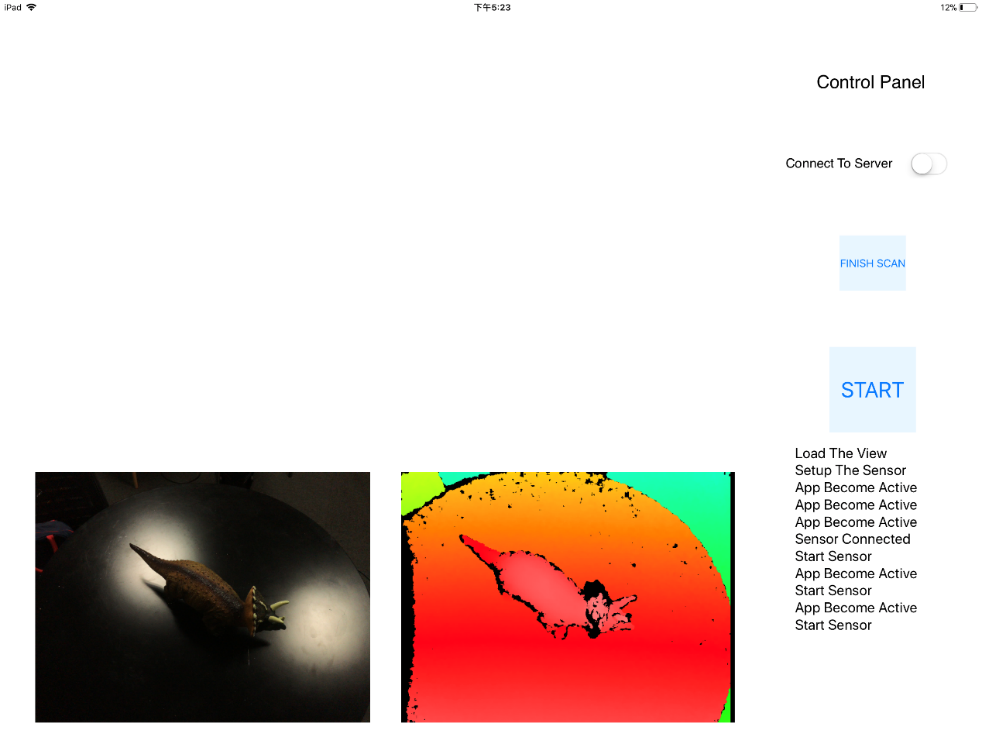
1. 界面中间为当前渲染的模型
2. pause：暂停
3. step：每次前进一帧
4. save：保存当前重建结果以及当前每个fragment的结果
5. reset：重置（暂未实现）
6. debug：进入debug模式
7. deformed：在debug模式下显示点云
8. vertex corr：在debug模式下画出对应点之间的连线
9. deformed normal: 在debug模式下画出每个点的normal
10. Points：总点数
11. Nodes：Node数量
12. Frags：fragment数量
13. FragIdxSrc：设置画点对的source fragment
14. FragIdxDst: 设置画点对的target fragment
15. **iPad端程序**
16. **操作步骤**
17. Structure sensor连接iPad。PC与iPad连接同一路由器，PC的ip地址设为192.168.31.8。
18. 在PC上运行xObjRecon程序，可先勾选“data trans test”，进行数据传输测试。点击“start server”按钮，启动server端。
19. 打开iPad应用xObjReconCapture，点击“Connect To Server”按钮连接到server端。PC端命令行点击“Start”按钮



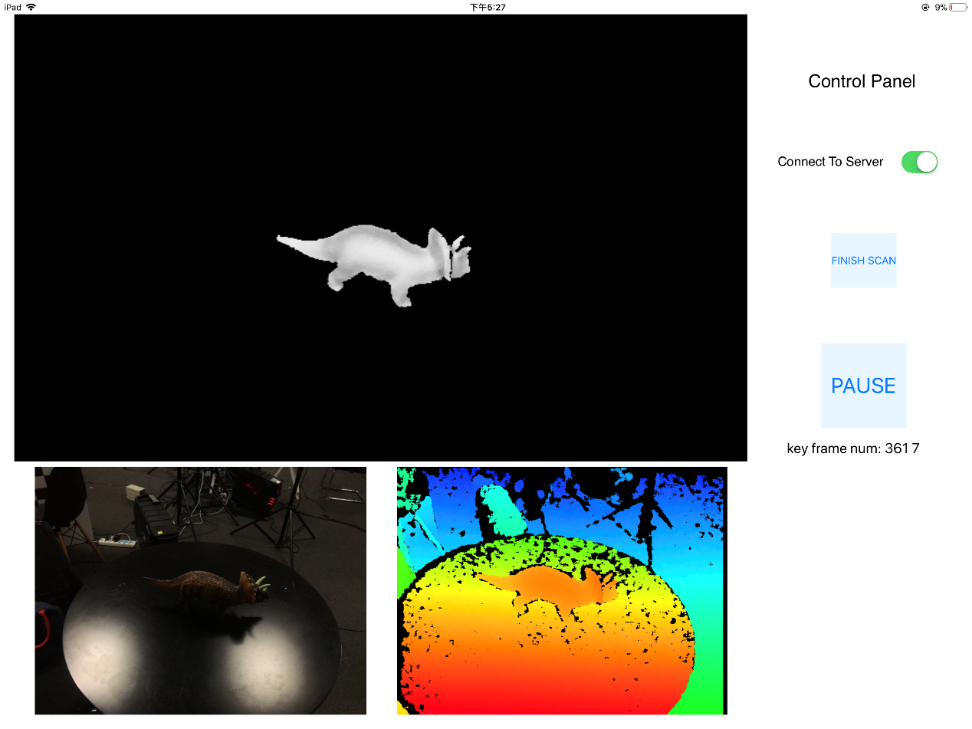
**界面左下角是iPad拍摄的彩色图，右边是Structure sensor拍摄的深度图。**



1. 在Data Trans Test窗口观察到color和depth对齐，并且窗口中的图片和iPad界面的彩色图、深度图相比没有明显时延，说明网络通畅，数据传输没有问题。
2. 在PC上重新运行xObjRecon程序。点击“start server”按钮，启动server端。

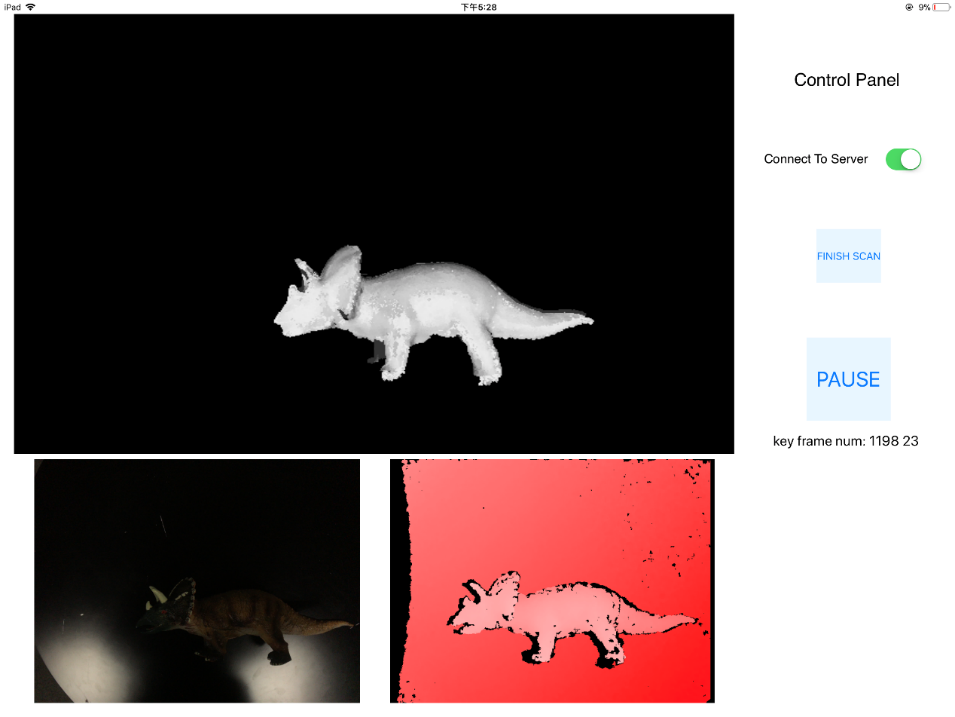


1. 重启iPad应用xObjReconCapture，点击“Connect To Server”按钮连接到server端。PC端命令行点击“Start”按钮开始拍摄。



**中间是渲染出的图像。**

1. 移动iPad，使拍摄对象处于彩色图中心，提取出拍摄对象。绕物体拍摄一周，点击“Pause”按钮暂停。



1. 转动物体，继续拍摄。注意：此时物体相对相机的位置要与之前暂停时刻的位置相近。
2. 如此反复，直至物体拍摄完全。点击“FINISH SCAN”,结束拍摄，将每个关键帧的彩色图传输到电脑中，并在电脑中保存重建模型和每个关键帧的相机位置。