

# SDK RapidScan Demo 说明文档

在进行扫描之前检查一下设备是否已经插好电源并正确连接好，注意 USB 接口应与电脑的 USB3.0 接口连接。以下为 SDK Rapid Scan Demo 的操作使用说明，整体流程如图 1 所示。

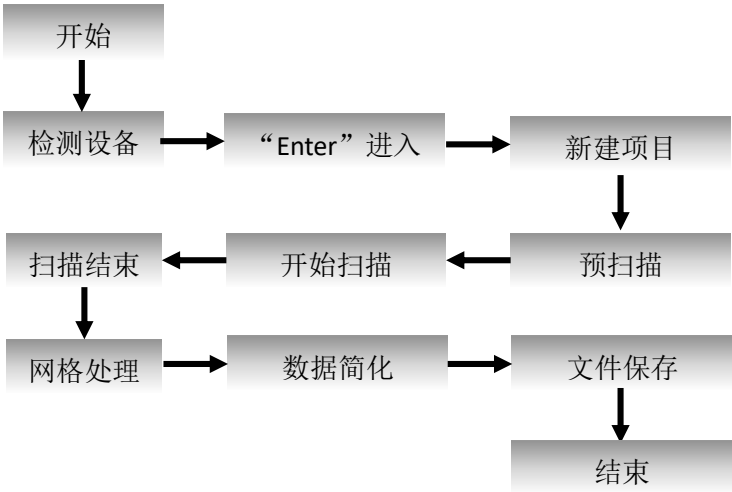


图 1 Rapid Scan 工作流程

## 1、开始

打开 SDK Rapid Scan，主界面如图 2 所示。

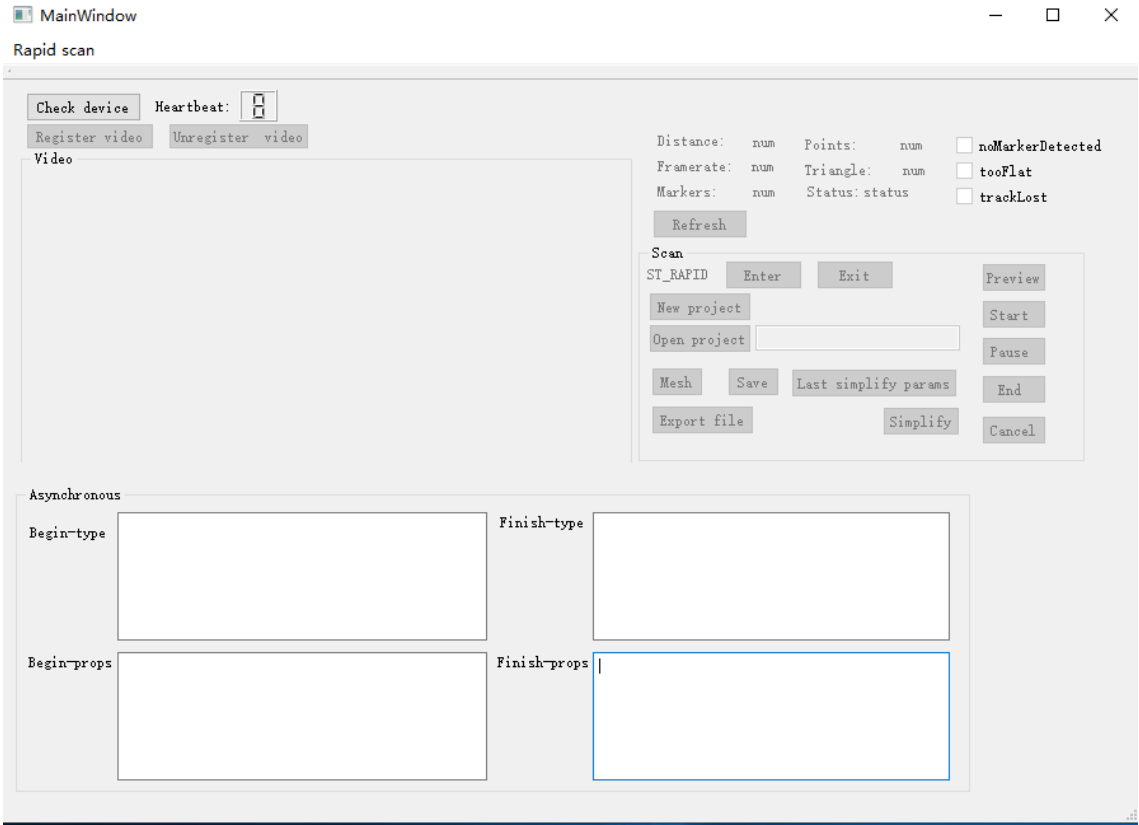


图 2 SDK Rapid Scan 主界面图

## 2、设备检测

点击如图 3 所示“check device”按钮进入设备检测状态，并出现一个进度条表示检测进度。

使用函数：mainwindow.cpp->on\_pushButton\_DeviceCheck\_clicked

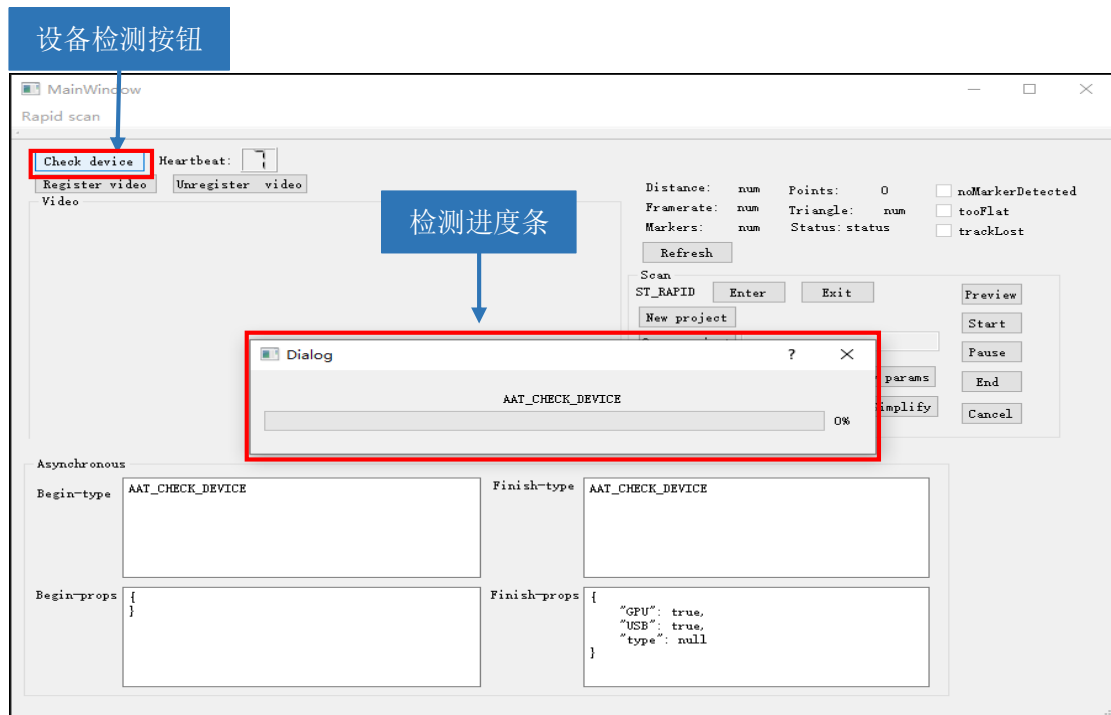


图 3 “check device”检测按钮界面示意图

3、点击“Enter”按钮时设备进入扫描状态，同时界面会出现相应的进度条，如图 4 所示。

使用函数：mainwindow.cpp-> on\_pushButton\_ScaneEnterScan\_clicked

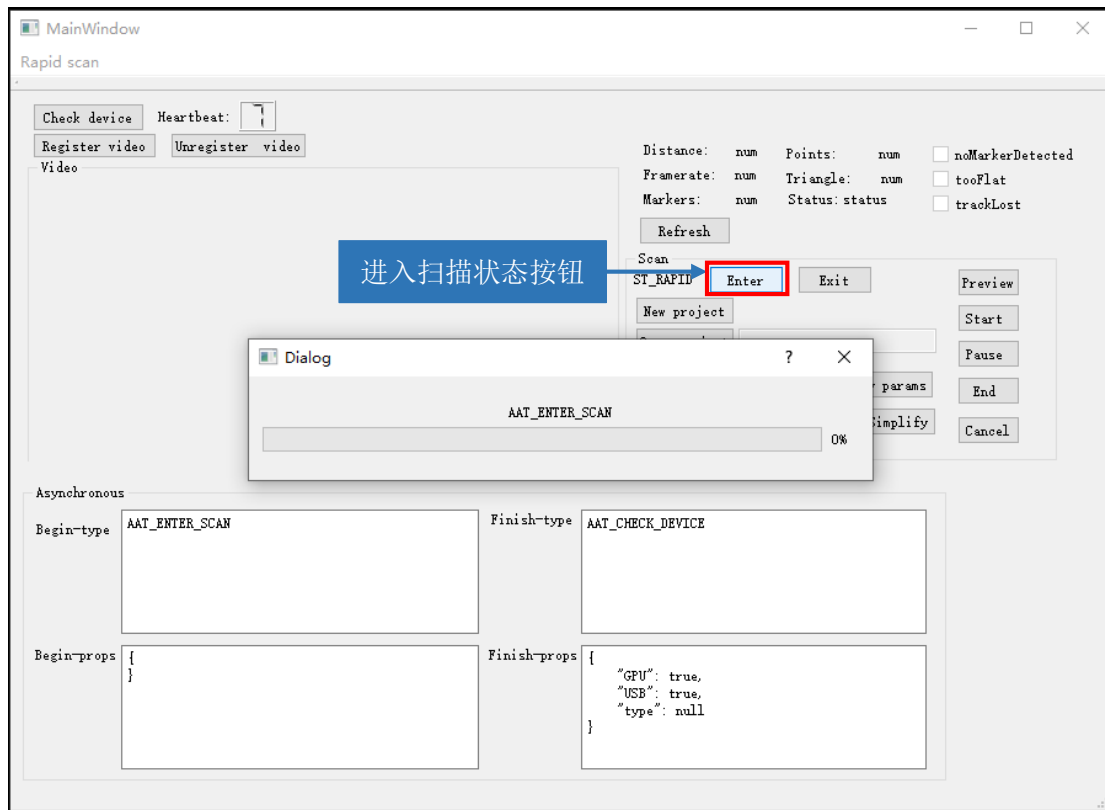
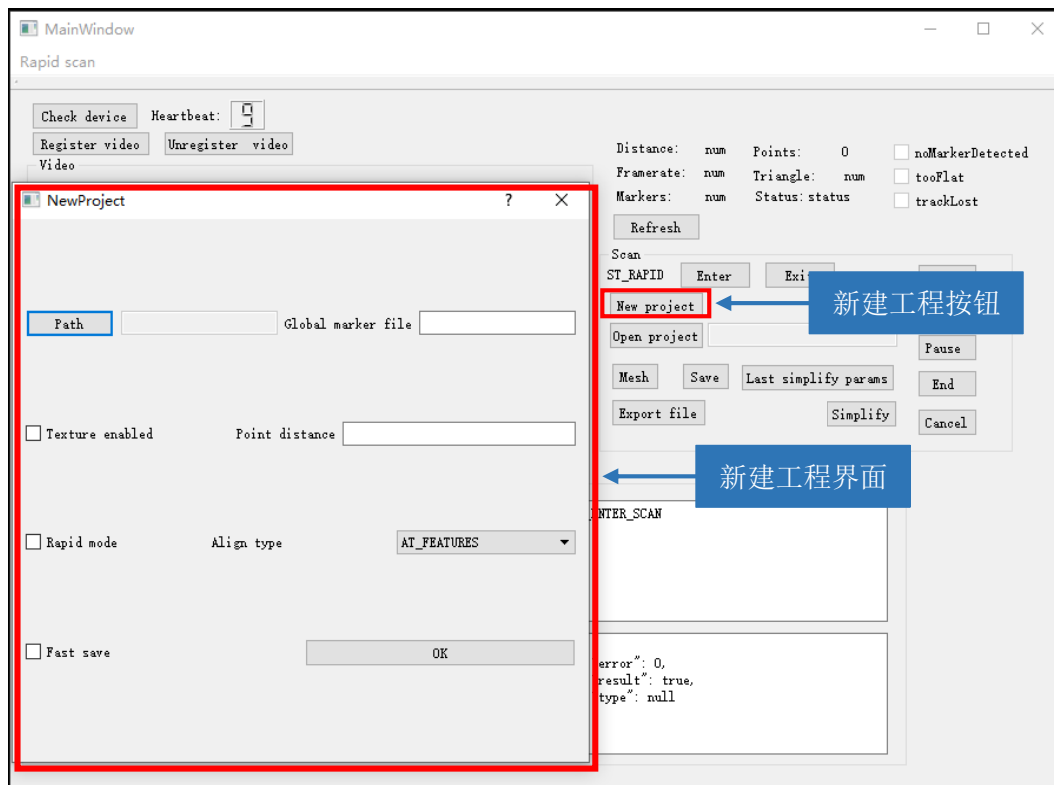


图 4 进入扫描状态界面示意图

#### 4、新建项目工程

点击“New project”按钮进入新建项目界面，如图 5 所示，新建项目的界面参数详情如表 1 所示。

使用函数：mainwindow.cpp-> on\_pushButton\_ScanNewProject\_clicked



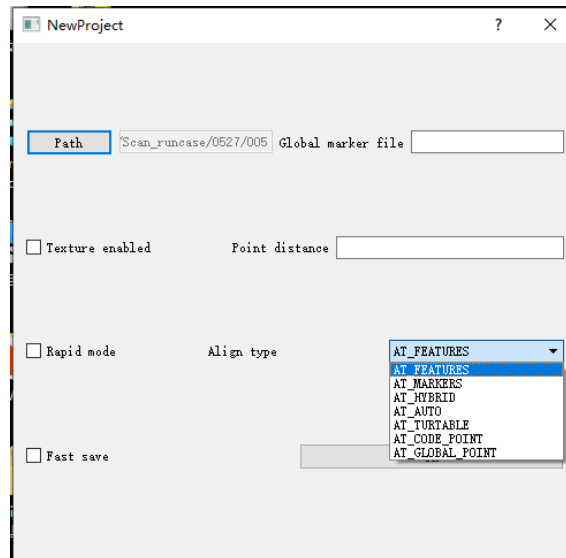


图 5 新建项目界面示意图

表 1 新建项目参数表

参数	参数含义	UI 类型	对应函数
Path	选择保存工程的路径	按钮+文本	commonui.cpp->onPathButtonClicked
Global marker file	全局框架点文件路径	按钮+文本	commonui.cpp->NewProject(QDialog *parent)
Texture enabled	是否使用纹理相机	复选框	commonui.cpp->NewProject(QDialog *parent)
Rapid mode	决定扫描帧率的高低	复选框	commonui.cpp->NewProject(QDialog *parent)
Fast save	是否保存每帧图像	复选框	commonui.cpp->NewProject(QDialog *parent)
Point distance	点距	文本	commonui.cpp->NewProject(QDialog *parent)
Align type	拼接类型：AT_FEATURES(特征拼接) AT_MARKERS(标志点拼接) AT_HYBRID(混合拼接) AT_AUTO(自动拼接)	下拉列表	commonui.cpp->NewProject(QDialog *parent)
OK	确认参数无误，进入新项目	按钮	commonui.cpp->onPushButtonClicked

此处我们选择新建项目的路径，设置 Point distance=1，其他均为默认，点击“OK”按钮。此时界面如图 6 所示，其右上角为扫描仪的参数，详情可见表 2。

当勾选 Rapid mode 和 Fast save 复选框时，扫描时点云的速度增加较快，帧率为不勾选 Rapid mode 复选框时的 2 倍，最终导出的文件也相对较大。

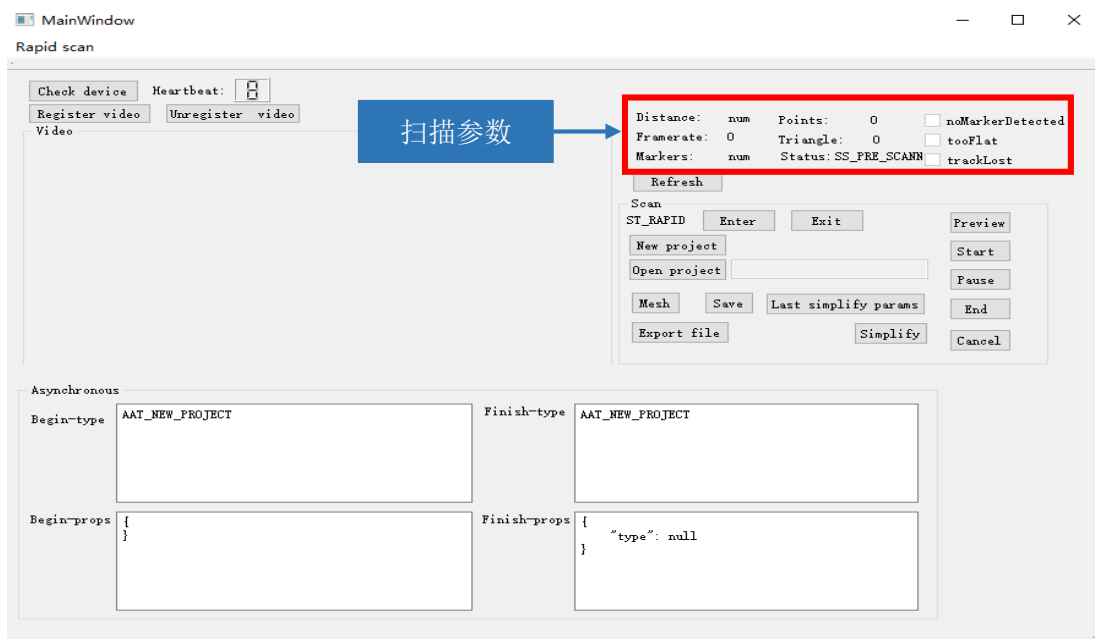


图 6 扫描仪参数

表 2 扫描仪参数表

参数	参数含义	UI 类型	对应函数
Distance	设备与扫描物体的距离	文本	mainwindow.cpp-> ScanDist
Framerate	帧率	文本	mainwindow.cpp-> ScanFramerate
Markers	扫描到标志点的数量	文本	mainwindow.cpp-> ScanMarkerCount
Points	扫描得到的点云数	文本	mainwindow.cpp->ScanPointCount
Triangle	模型三角网格数	文本	mainwindow.cpp-> ScanTriangleCount
Status	设备扫描状态	文本	mainwindow.cpp->ScanStatus
noMarkerDetected	没有扫描到标志点(不能手动勾选, 自动判断)	复选框	mainwindow.cpp-> ScanNoMarkerDetected
TooFlat	太平滑(不能手动勾选, 自动判断)	复选框	mainwindow.cpp-> ScanTooFlat
trackLost	跟踪丢失(不能手动勾选, 自动判断)	复选框	mainwindow.cpp-> ScanTrackLost

5、点击“Preview”按钮进入设备预扫模式,界面如图 7 所示,预扫界面的参数详情见表 3。

对应的函数: mainwindow.cpp-> on\_pushButton\_Pre\_clicked

界面对应 startscan.cpp

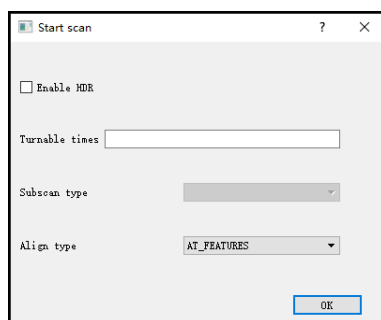
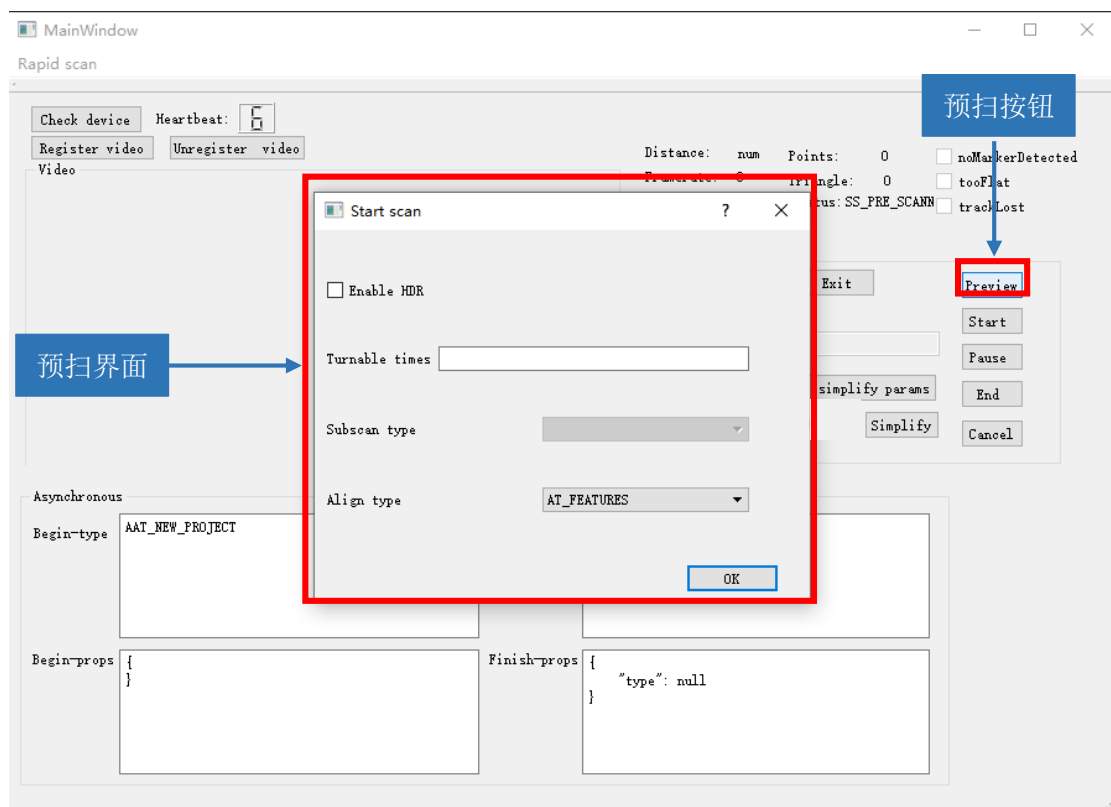


图 7 预扫界面示意图

表 3 预扫参数表

参数	参数含义	UI 类型	对应函数
Enable HDR	HDR 使能	复选框	startscan.cpp->on_pushButton_clicked
Turnable times	转台的转动的次数	文本	startscan.cpp->on_pushButton_clicked
Subscan type	扫描子类型（灰色，快扫没有选项）	下拉列表	startscan.cpp->on_pushButton_clicked
Align type	拼接类型： AT_FEATURES(特征拼接) AT_MARKERS(标志点拼接) AT_HYBRID(混合拼接) AT_AUTO(自动拼接)	下拉列表	startscan.cpp->on_pushButton_clicked
OK	确认参数无误，进入预扫	按钮	startscan.cpp->on_pushButton_clicked

此处我们参数都为默认，点击“OK”按钮进入预扫模式。点击“Register video”进行相机视频注册，若前面操作正确，此时在 Video 窗口下可以显示扫描仪的左右相机实时扫描的视图，如图 8 所示。

“Register video”按钮：相机视频注册 mainwindow.cpp->  
on\_pushButton\_RegisterProcessor\_clicked

“UnRegister video”按钮：相机视频不进行注册 mainwindow.cpp->  
on\_pushButton\_UnregisterProcessor\_clicked

“Heartbeat”文本：服务端的心跳（当值为 0 时代表服务端死掉，已经无法  
为客户端提供数据）mainwindow.cpp-> onHeartbeat

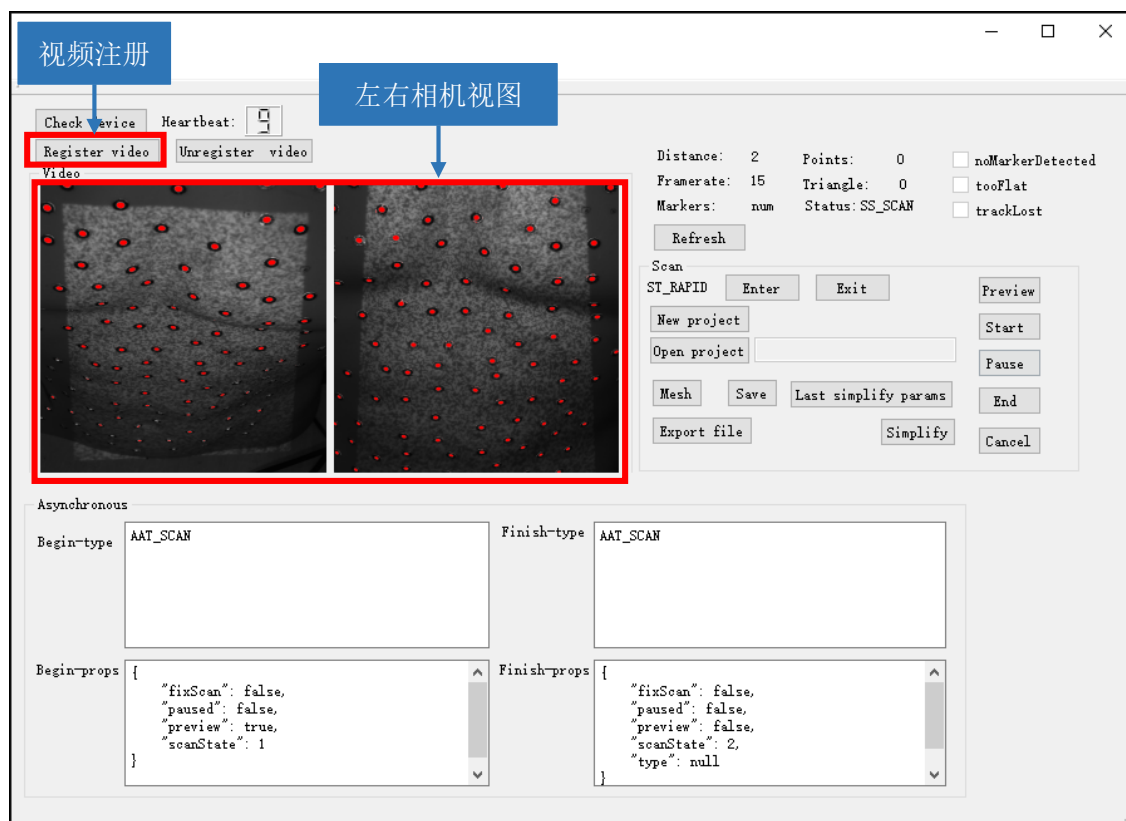


图 8 视频注册界面示意图

## 6、开始扫描

点击“Start”按钮对模型开始扫描，界面如图 9 所示，开始扫描界面的参数与预扫描中的参数一致，除“OK”按钮的含义为进入扫描外，其他参数详情可见表 3。此处我们均使用默认参数，点击“OK”按钮进入扫描。

对应的函数：mainwindow.cpp-> on\_pushButton\_start\_clicked

界面对应 startscan.cpp

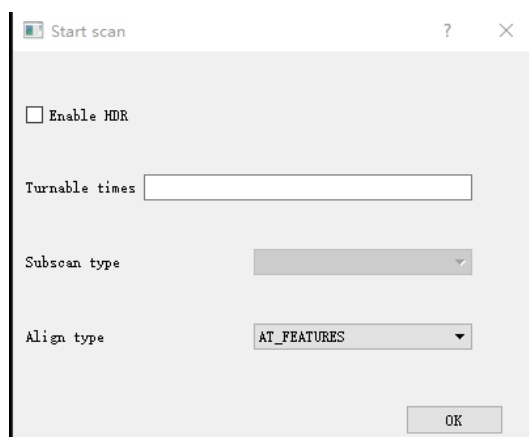
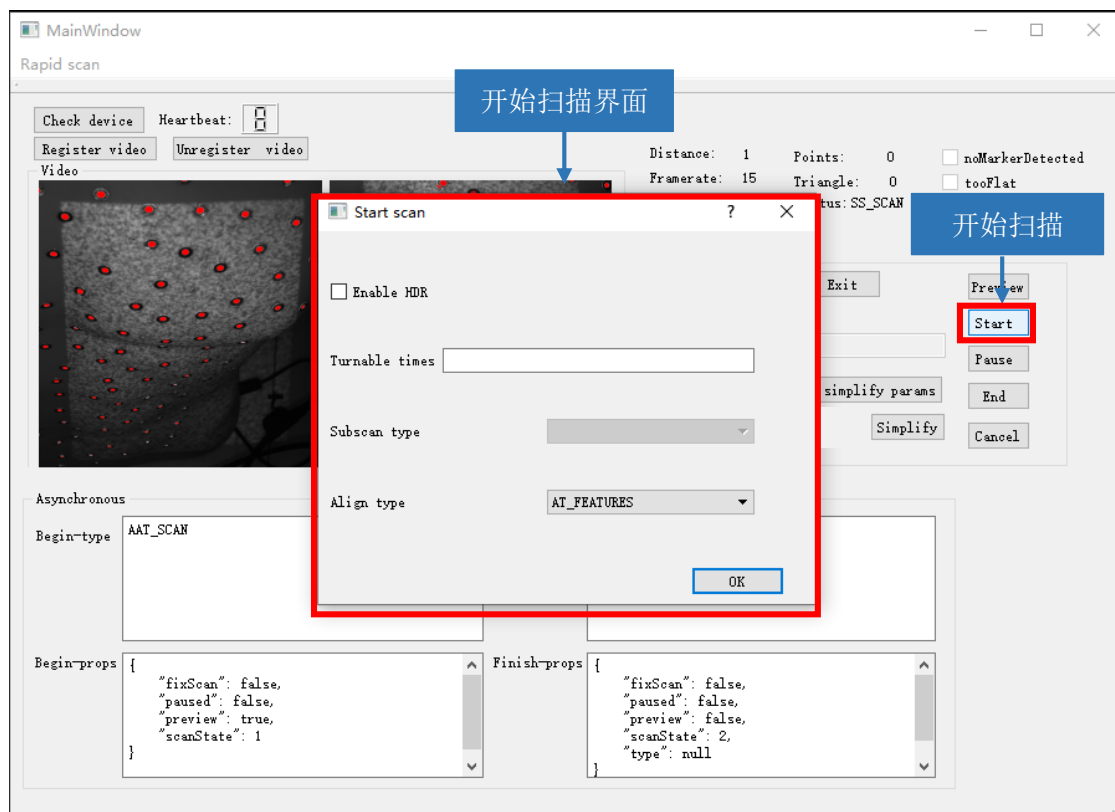


图 9 开始扫界面示意图

扫描过程中扫描参数如 Heartbeat 文本、Distance、Points、Framerate 等会实时的更新见图 10，点击 Refresh 按钮会对数据进行手动更新。

Refresh 按钮对应的函数：mainwindow.cpp->on\_pushButton\_Refresh\_clicked



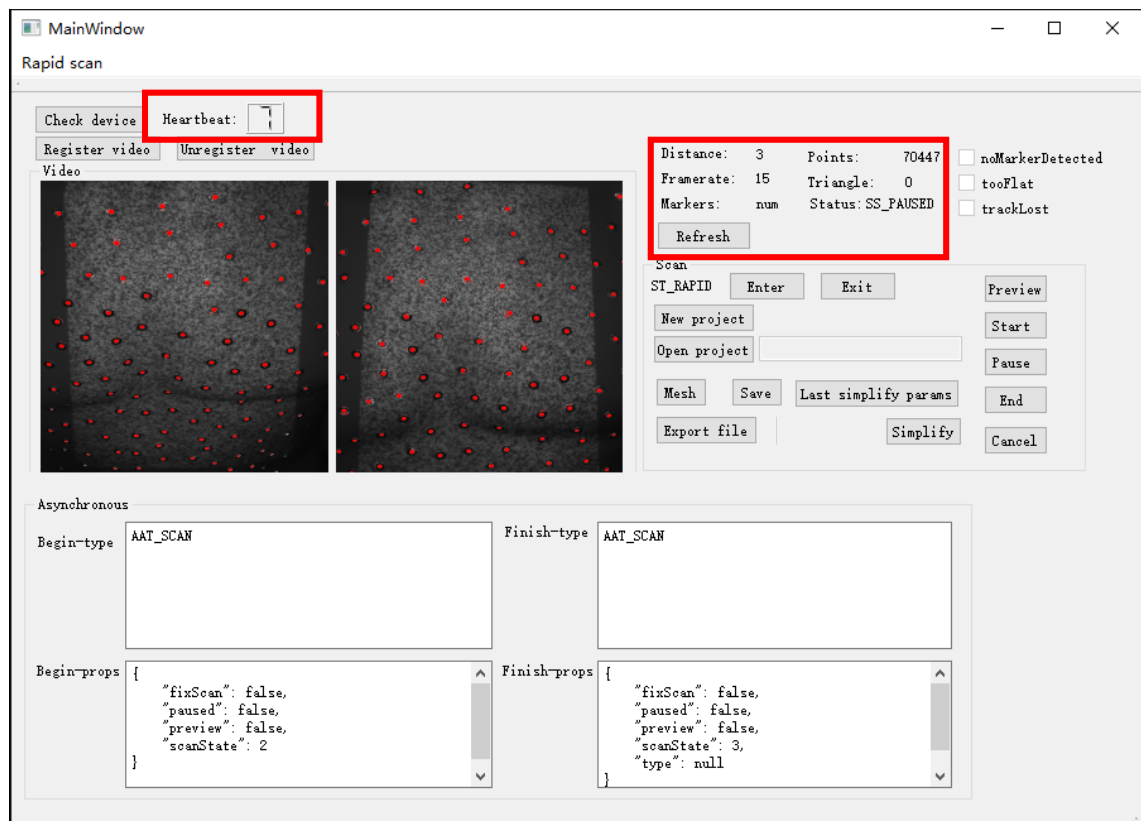


图 10 扫描过程数据实时更新示意图

## 7、扫描结束

当数据采集完成时点击“End”按钮，并设置参数结束扫描，如图 11 所示，参数详情见表 4。此处我们均为默认参数，点击“OK”按钮结束扫描，图 12 为结束扫描后数据处理的一个进度条示意图。

对应的函数：mainwindow.cpp-> on\_pushButton\_scanEndScan\_clicked

界面对应 moc\_endscan.cpp、endscan.cpp

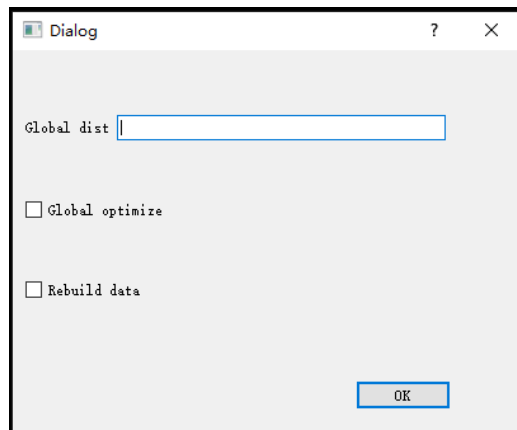
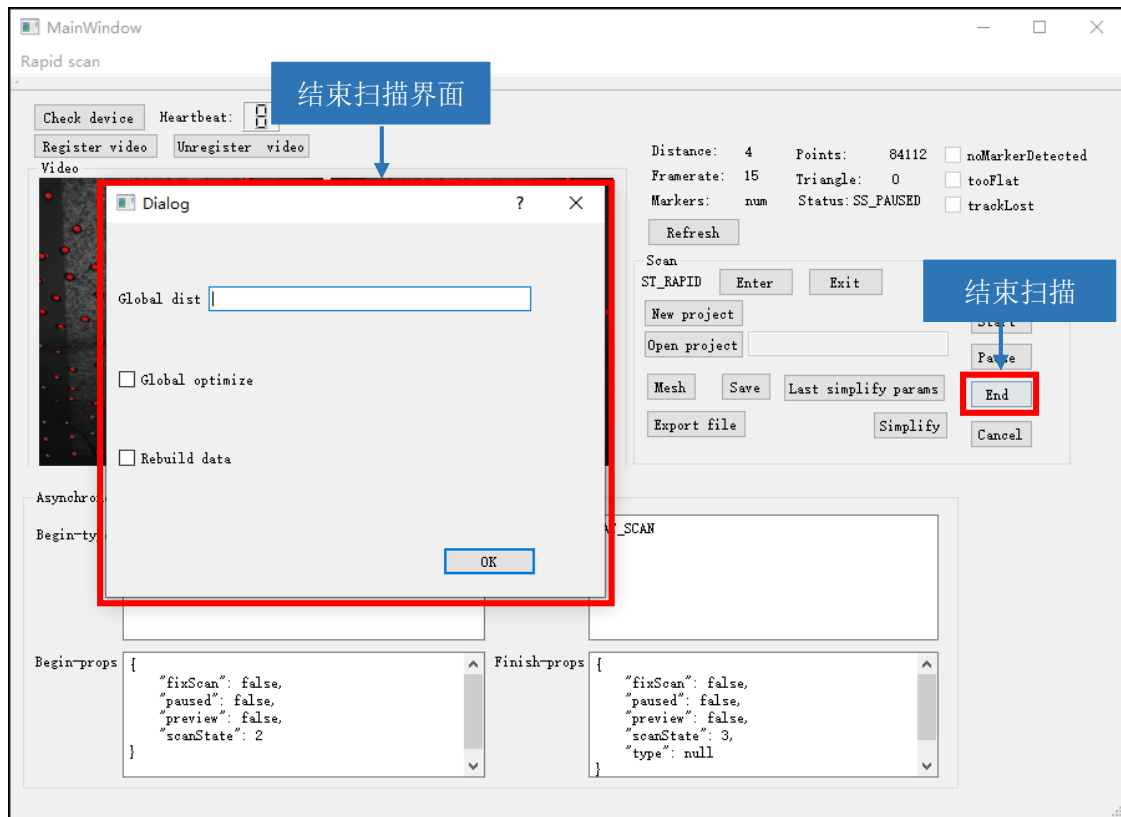


图 11 结束扫描界面示意图

表 4 结束界面参数表

参数	参数含义	UI 类型	对应函数
Point dist	点距	文本	endscan.cpp-> on_pushButton_endScan_clicked
Global optimize	全局优化	复选框	endscan.cpp-> on_pushButton_endScan_clicked
Rebuild data	重建数据	复选框	endscan.cpp-> on_pushButton_endScan_clicked
OK	确认参数无误，结束预扫	按钮	endscan.cpp-> on_pushButton_endScan_clicked

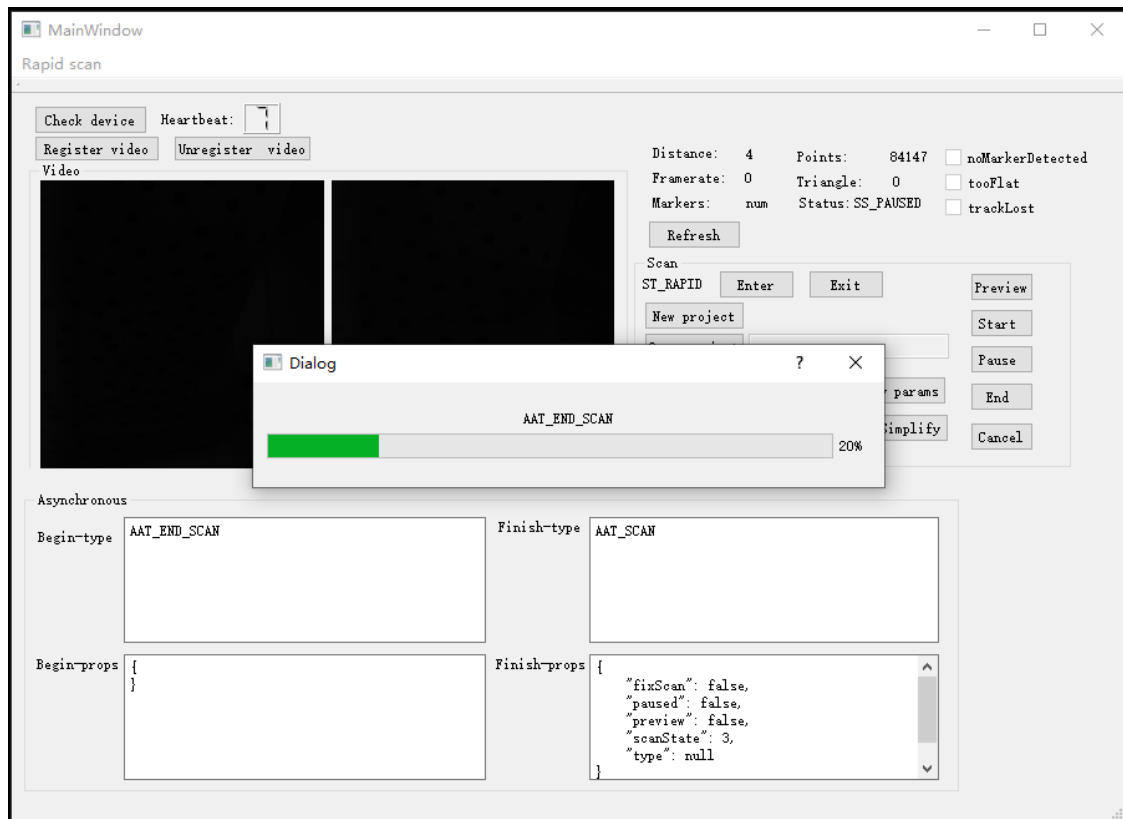


图 12 结束扫描数据处理进度示意图

## 8、网格处理

点击“Mesh”按钮，对点云进行网格化处理，示意图如图 13 所示，Mesh 处理界面的参数详情见表 5。我们选择默认参数进行网格处理。

“Mesh”按钮对应的函数：mainwindow.cpp->  
on\_pushButton\_scanMesh\_clicked  
界面对应： mesh.cpp

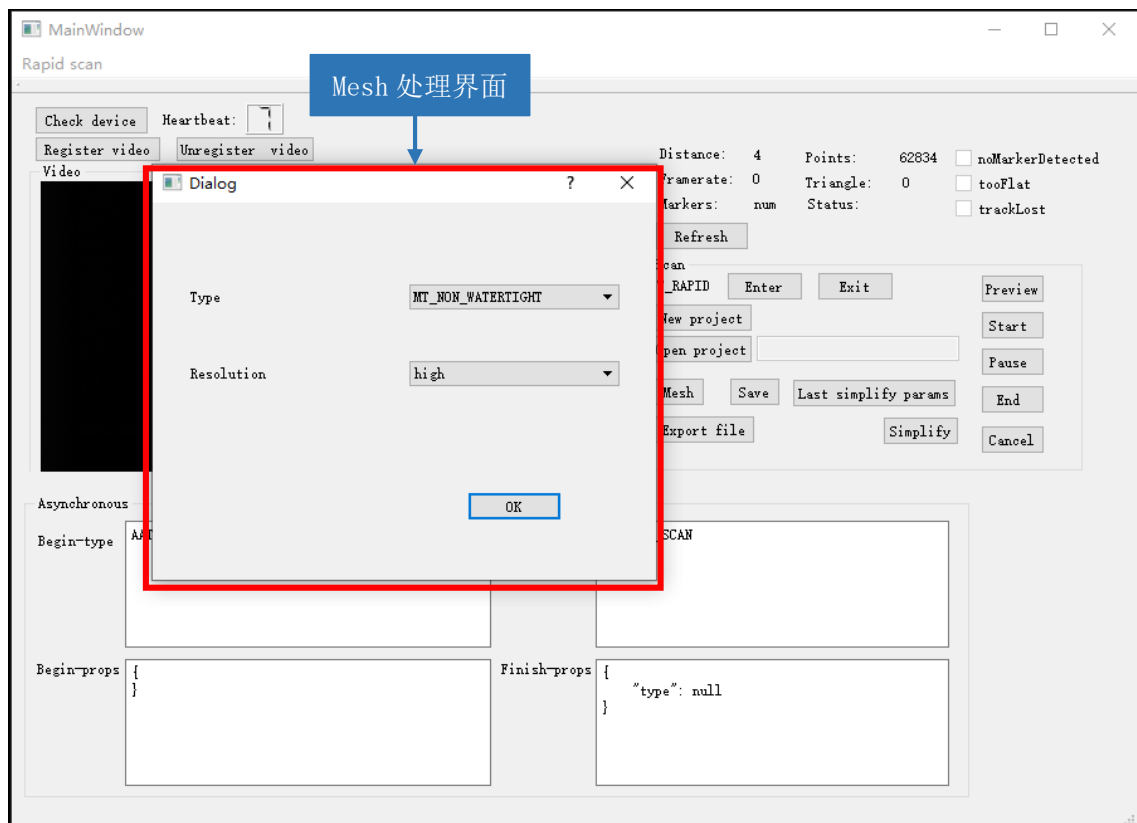
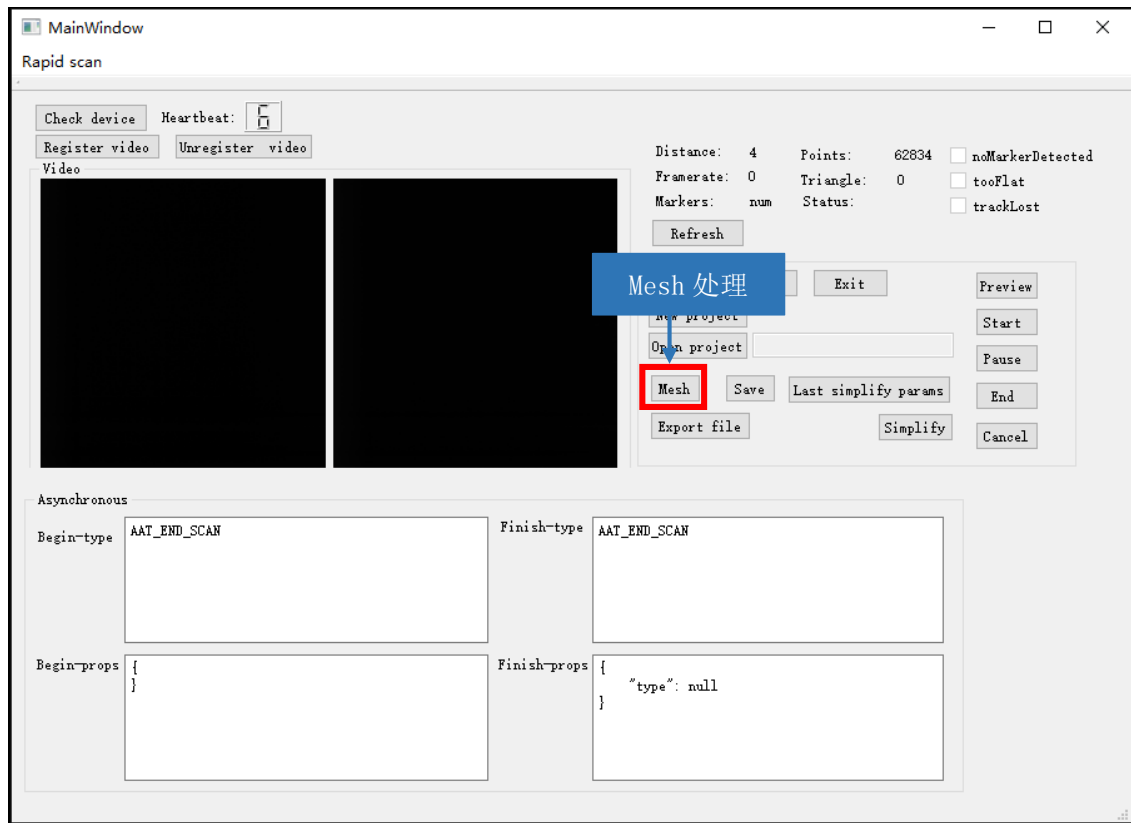


图 13 Mesh 处理界面示意图

表 5 Mesh 界面参数表

参数	参数含义	UI 类型	对应函数
Type	网格封装的类型: MT_NON_WATERTIGHT (非闭合) MT_WATERTIGHT (闭合)	下拉列表	mesh.cpp-> on_pushButton_mesh_clicked
Resolution	网格密度: high(高细节) medium(中细节) low(低细节)	下拉列表	mesh.cpp-> on_pushButton_mesh_clicked
OK	确认参数无误, 进行网格处理	按钮	mesh.cpp-> on_pushButton_mesh_clicked

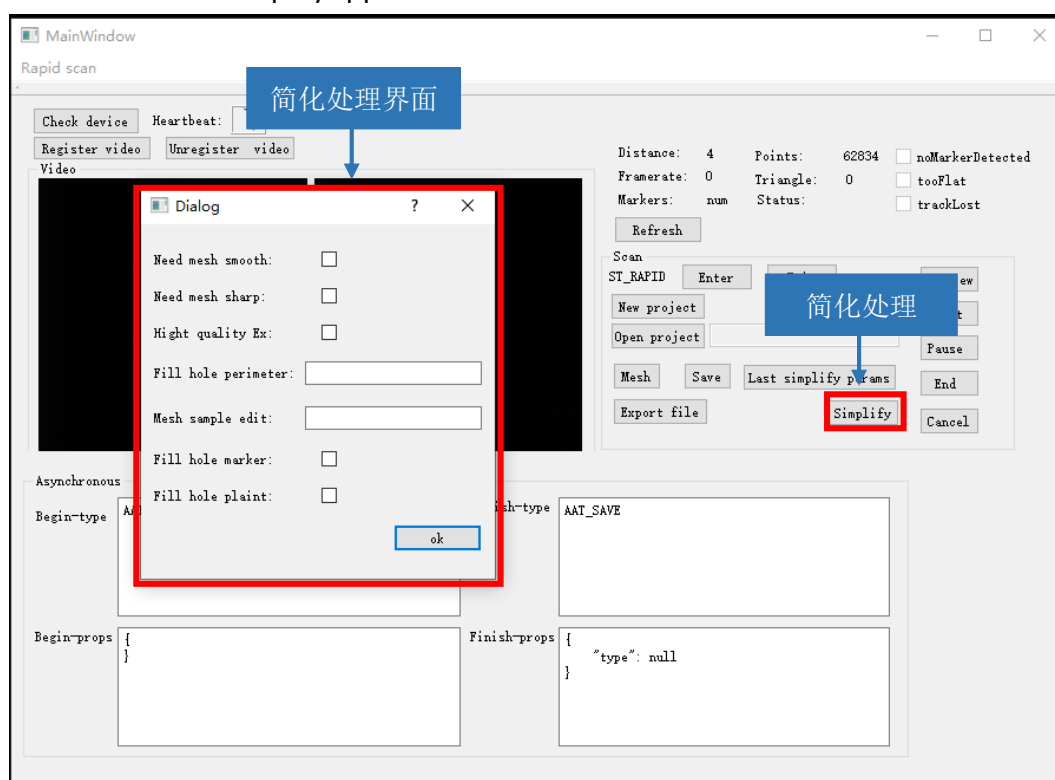
### 9、简化处理

点击“Simplify”按钮完成网格数据简化, 简化界面如图 14 所示, 简化参数详情见表 6。此处勾选“Need mesh smooth”和“Need mesh sharp”复选框进行网格简化处理。

“Simplify”按钮对应的函数:

mainwindow.cpp->on\_pushButton\_scanSimplify\_clicked

界面对应: simplify.cpp



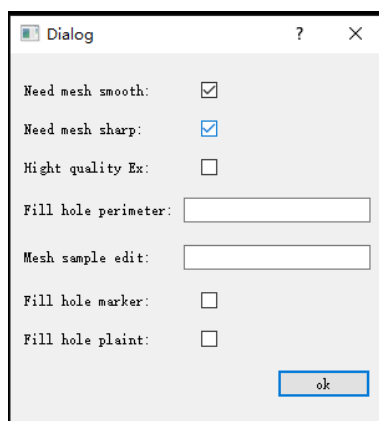


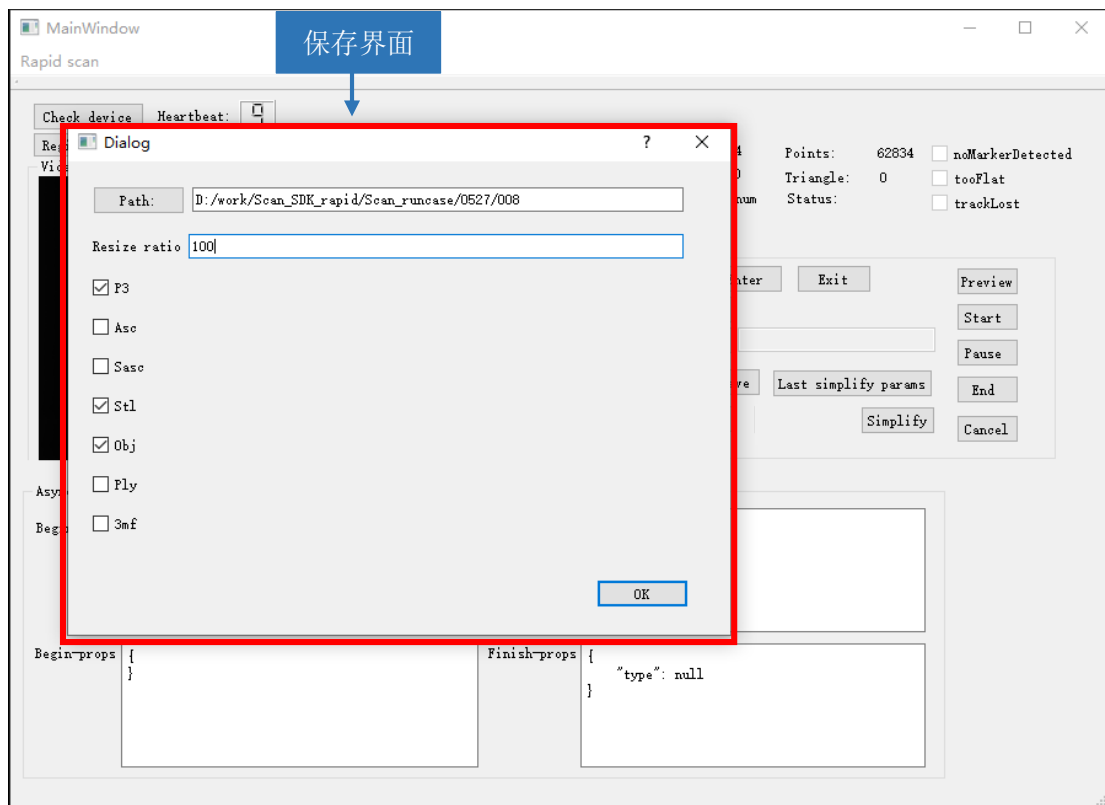
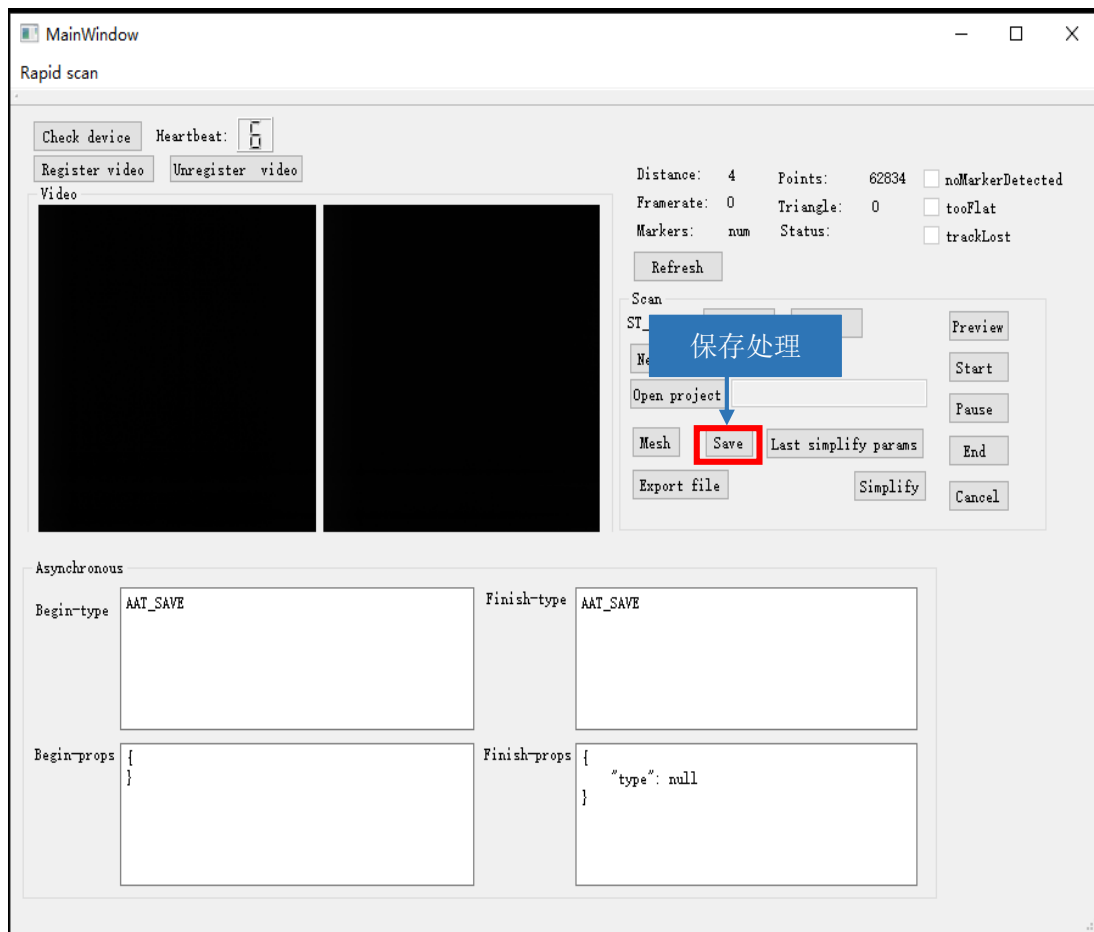
图 14 Simplify 处理界面示意图

表 6 Simplify 处理界面参数表

参数	参数含义	UI 类型	对应函数
Need mesh smooth	网格平滑	复选框	simplify.cpp->on_pushButton_clicked
Need mesh sharp	网格锐化	复选框	simplify.cpp->on_pushButton_clicked
Hight quality Ex	高质量	复选框	simplify.cpp->on_pushButton_clicked
Fill hole perimenter	补洞参数	文本	simplify.cpp->on_pushButton_clicked
Mesh sample edit	简化比例	文本	simplify.cpp->on_pushButton_clicked
Fill hole marker	标志点补洞	复选框	simplify.cpp->on_pushButton_clicked
Fill hole plaint	普通补洞	复选框	simplify.cpp->on_pushButton_clicked
OK	确认参数无误，进行简化处理	按钮	simplify.cpp->on_pushButton_clicked

10、点击“Save”按钮对模型数据进行保存处理，界面如图 15 所示，保存界面参数详情见表 7。此处我们设置好保存路径后设 Resize ratio 为 100，然后点击“OK”按钮进行保存。（保存比例的大小不影响文件的大小，会改变模型的大小。）

“Save”按钮对应的函数：mainwindow.cpp->on\_pushButton\_scanSave\_clicked  
保存界面对应：save.cpp



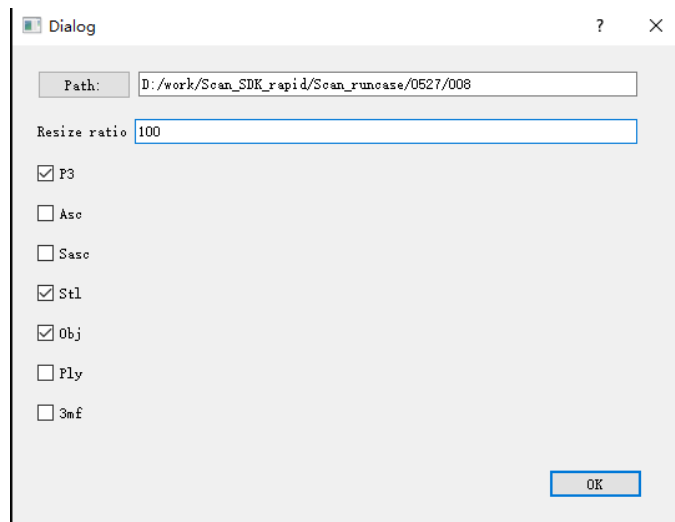


图 15 Save 保存界面示意图

表 7 Save 处理界面参数表

参数	参数含义	UI 类型	对应函数
Path	保存路径	按钮+ 文本	save.cpp->on_pushButton_Path_clicked
P3、Asc、 Sasc、Stl、 Obj、Ply、 3mf	文件保存格式	复选框	save.cpp->on_pushButton_clicked
OK	确认参数无误， 进行保存处理	按钮	save.cpp->on_pushButton_clicked

11、图 16 红色框中的参数对应的含义见表 8。



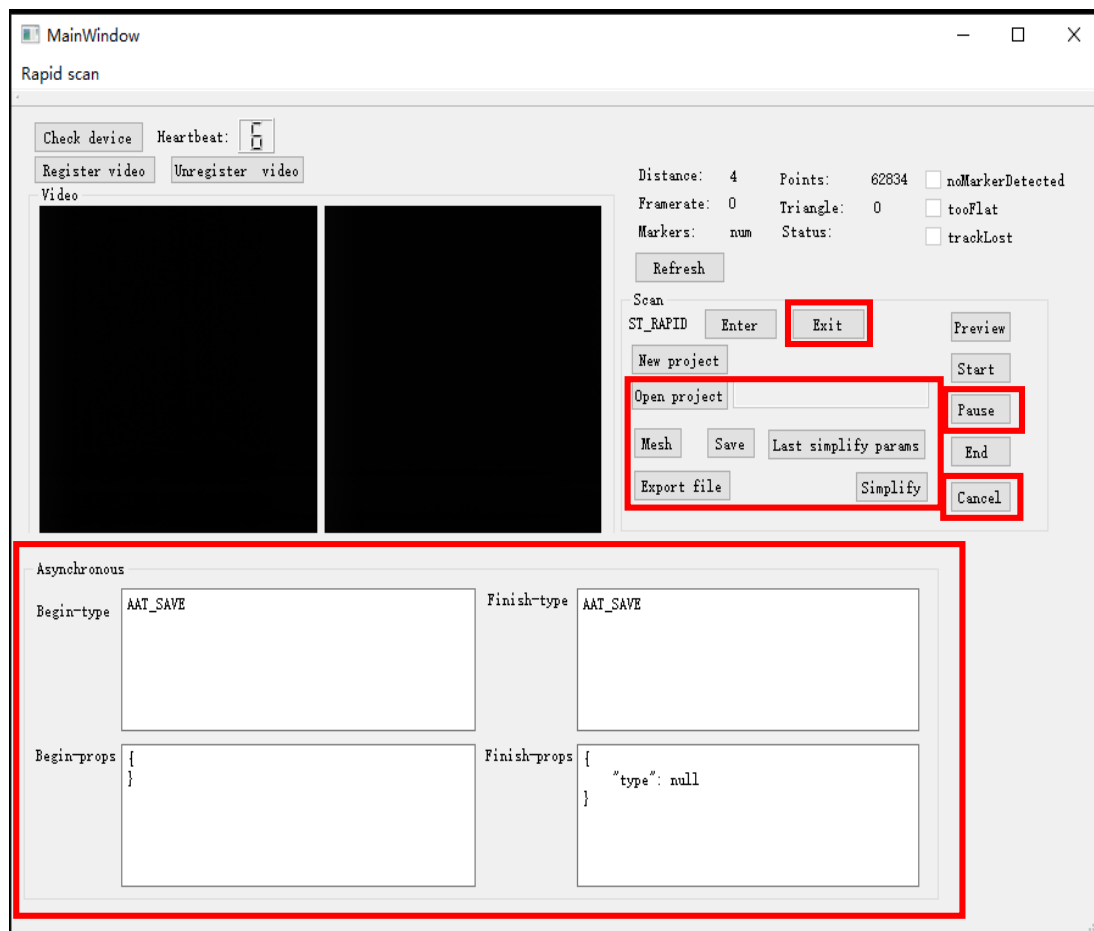


图 16 框选中的参数

表 8 参数列表

参数	参数含义	UI 类型	对应函数
Exit	退出该次扫描	按钮	mainwindow.cpp-> on_pushButton_ScanExitScan_clicked
Open project	打开项目 (.rap_prj)	按钮+ 文本	mainwindow.cpp-> on_pushButton_ScanOpenProject_clicked
Pause	暂停	按钮	mainwindow.cpp-> on_pushButton_Pause_clicked
Cancel	取消	按钮	mainwindow.cpp-> on_pushButton_scanCancelScan_clicked
Mesh	网格封装	按钮	mainwindow.cpp-> on_pushButton_scanMesh_clicked
Save	文件保存	按钮	mainwindow.cpp-> on_pushButton_scanSave_clicked
Last simplify params	获取最新的简化参数	按钮	mainwindow.cpp-> on_pushButton_ScanLastSimplifyParams_clicked
Export file	文件导出	按钮	mainwindow.cpp-> on_pushButton_ScanExportFile_clicked
Simplify	简化	按钮	mainwindow.cpp-> on_pushButton_scanSimplify_clicked
Begin-type	异步回调开始类型	文本	Mainwindow.cpp-> onPublishReceived
Begin-props	异步回调开始属性	文本	mainwindow.cpp-> onPublishReceived

Finish-type	异步回调完成类型	文本	mainwindow.cpp-> onPublishReceived
Finish-props	异步回调完成属性	文本	mainwindow.cpp-> onPublishReceived

12、完成扫描后在选择的路径下会生成 3 个后缀分别为 “.rap\_prj”、“.dat”、“.dat” 的工程文件，在保存模型路径下会生成与所选格式相对应的文件。

附：

1、点击“Concel”按钮时出现图 17 所示界面，其参数见表 9。

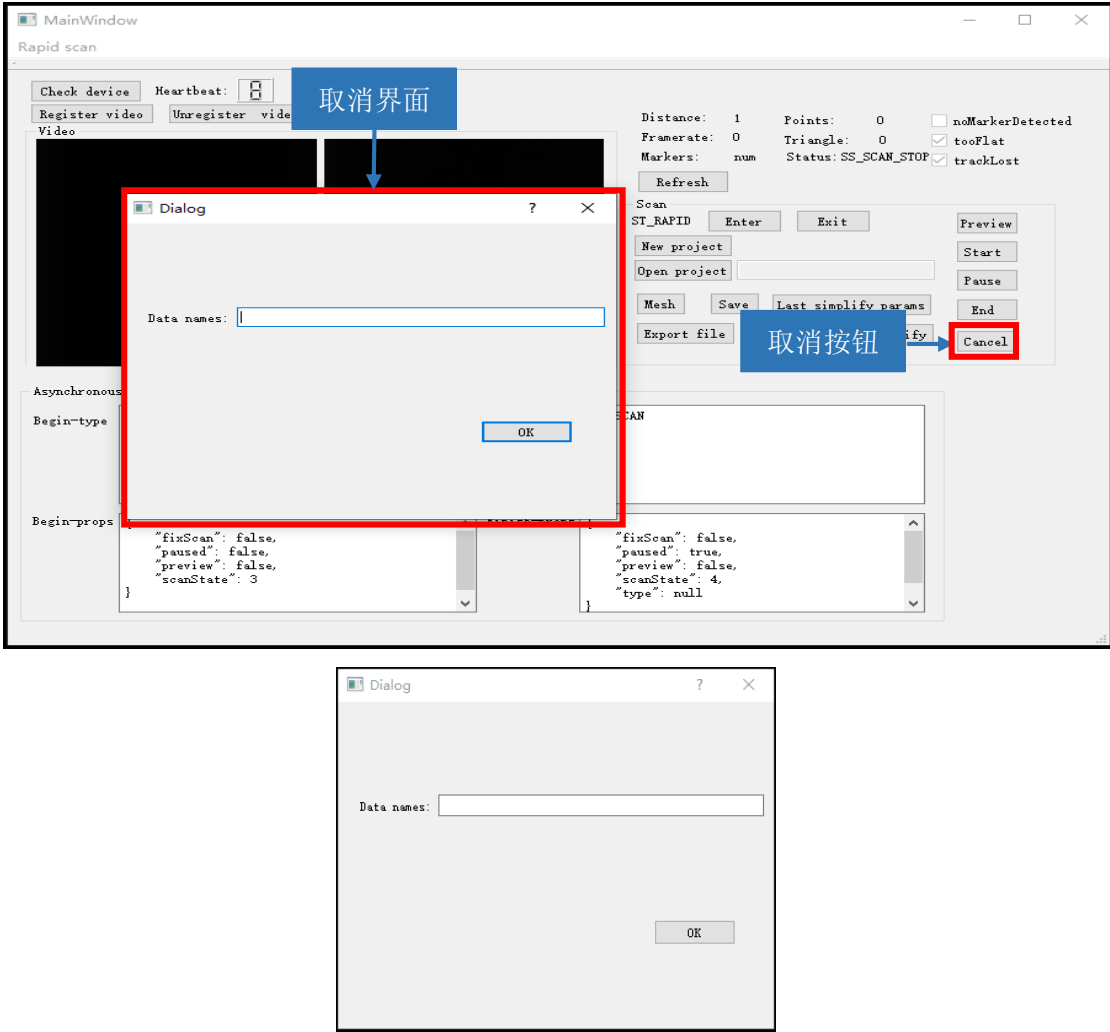


图 17 Cancel 处理界面示意图

表 9 Cancel 处理界面参数表

参数	参数含义	UI 类型	对应函数
Data names	文件名称	文本	cancel.cpp->on_pushButton_cancelScan_clicked
OK	确认参数无误，进行取消处理	按钮	cancel.cpp->on_pushButton_cancelScan_clicked

2、点击“Pause”按钮暂停扫描，出现图 18 所示界面，子界面与图 9 所示的子界面一致。当暂停后再继续扫描时可再次点击“Pause”按钮，若点击“End”按钮则表示扫描结束。

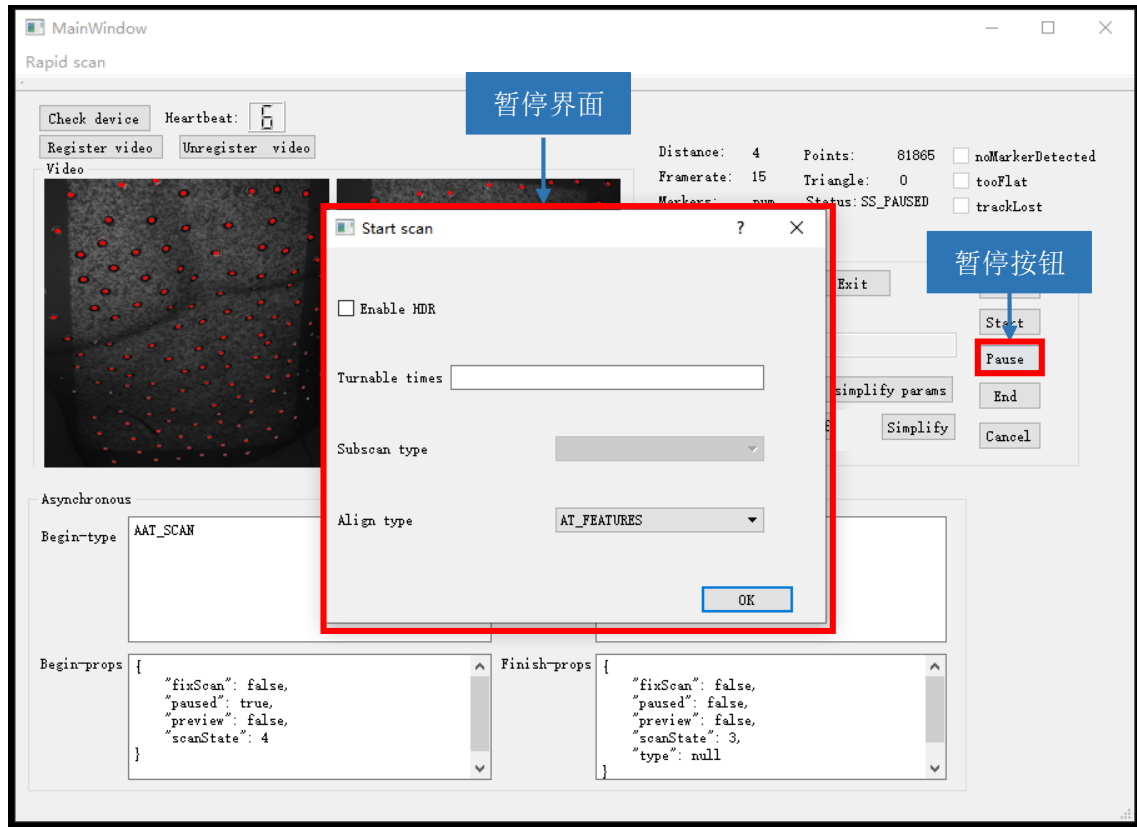


图 18 Pause 暂停界面示意图

3、点击“Exit”按钮可以退出正在进行的扫描项目，再次扫描可从说明的第 3 步开始进行新项目的扫描。