

Tutorial básico para uso de IgsTK bajo Windows

Este tutorial ha sido generado gracias al trabajo de María Luisa Pinto Salamanca, de la Universidad Nacional de Colombia

El paquete IGSTK Image Guided Surgery Toolkit es una librería software de código abierto que provee elementos básicos para desarrollar aplicaciones de procesamiento e interacción con imágenes medicas ofreciendo la posibilidad de integrar sistemas de navegación quirúrgica y realizar aplicaciones directamente en un quirófano.

Las posibilidades de uso intraoperatorio en cirugías guiadas por imágenes, hacen que la arquitectura de IGSTK sea robusta, diseñada para evitar al máximo errores de software que pongan en riesgo al paciente. Se ofrecen funcionalidad de alto nivel como:

- Visualización de imágenes medicas de CTs y MRIs en formato DICOM, así como videos de fluoroscopias.
- Integración de sistemas de navegación (tracking hardware) de DNI (<http://www.ndigital.com/products.php> con funciones para PolarisTracker y AuroraTracker) y Claron Technology Inc. (<http://www.clarontech.com/measurement.php> con funciones para el MicronTracker).
- Capacidad de integración con sistemas GUI.
- Soporta comunicación serial.
- Opciones de visualización (axial, sagital, coronal y 3D) y multiescalado para las imágenes.

Esta escrita en C++ y requiere de la plataforma cruzada CMake para compilar proyectos multiplataforma. Fue diseñada para funcionar en varios sistemas operativos o compiladores, incluidos con versiones superiores a:

- Microsoft Windows, Visual C++ 7.0 (no se incluye la versión 6.0 por los problemas de soporte para C++)
- Linux gcc 3.4
- Linux gcc 4.1
- MacOSX 4.0

Versión de trabajo e Instalación:

La última versión disponible (hasta la fecha actual) es IGSTK-4.2. Para la ejecución de cualquier versión se requiere instalar previamente las librerías de código abierto:

- Insight Segmentation and Registration Toolkit (ITK).
- Visualization Toolkit (VTK)
- GUI toolkits (puede ser FLTK, MFC o QT)
- CMake

IGSTK se compila correctamente con versiones específicas de estas librerías, lo cual exige verificar previamente las recomendaciones de compatibilidad.

Para IGSTK Release 2.0, se recomienda en [1], descargar e instalar previamente:

- ITK. Release 3.0.1
- VTK. Release 5.0
- FLTK 1.1
- CMake. Release 2.4

Para IGSTK Release 4.0, se recomienda en [2], descargar e instalar previamente:

- ITK. Release 3.10.1
- VTK. Release 5.2
- Qt. Release 4.3
- CMake. Release 2.6

La versión que se guía en este tutorial es IGSTK 4.2.

Instalación IGSTK con S.O. Windows XP:

Verifique los requisitos instalación de IGSTK v. 4.2. La generación de GUIs se hará con Qt. Release 4.0 o superior y FLTK 1.1 para la compilación de los ejemplos). Las otras librerías que se asumen instaladas previamente son: **vtk-5.2.1** y **InsightToolkit-3.10.1**. En los guiones anteriores se indico como instalar otras versiones de vtk e itk, el procedimiento es similar.

Descargue la librería [IGSTK-4.2.zip](http://www.igstk.org/IGSTK/resources/software.html) desde <http://www.igstk.org/IGSTK/resources/software.html>. Descomprimir en un directorio para los archivos de código fuente (C:\IGSTK\IGSTK4.2_source) y crear el directorio para los archivos binarios (C:\IGSTK\IGSTK4.2_bin).

Ejecutar CMake 2.6 (deprecated), indicar la ubicación de C:\.._source y C:\.._bin. Marcar show advanced values. Seleccionar el programa de compilación (para este caso MSVC 8.0 2005). SI tiene mas de una versión instalada para ITK se generara un mensaje de error (CMakeList.39), que se soluciona indicando manualmente los directorios de ejecución (bin) de cada librería en la pestaña con puntos suspensivos donde se denota que no se encontró librería, tenga cuidado con las versiones que selecciona. El procedimiento puede repetirse para VTK. En CMake, pulse Configure, visualizara algunos mensajes de restricción.

Una vez detectados los directorios de ITK y VTK, las opciones pendientes de CMake se mostraran en franjas rojas. Ubique y defina las variables BUILD_TESTING=OFF y IGSTK_BUILD_EXAMPLES=OFF (si lo desea puede compilar de una vez los ejemplos, recomendado para probar directamente con FLTK).

Si va a trabajar aplicación GUI con Qt:

Habilite la opción de GUI con Qt con la variable IGSTK_USE_Qt=ON, si es necesario indique también el directorio donde construyó (bin) a Qt. Si solamente hay una versión instalada, esta se tomara por defecto.

Si va a trabajar aplicación GUI con FLTK:

Esta librería se puede descargar desde

<http://public.kitware.com/IGSTKWIKI/index.php/Image:Fltk-1.1-12-16-05.zip>. El proceso de instalación también se realiza a través de CMake, la única restricción es que sea ejecutada antes de la instalación de IGSTK, habilitando la opción `IGSTK_USE_FLTK=ON`.

Puede seleccionar las opciones de FLTK y QT simultáneamente, previa instalación de cada librería. Presione nuevamente Configure, haga las especificaciones de versiones y directorios que requiera adicionar, hasta que las franjas rojas desaparezcan y pulse OK.

Diríjase al directorio de construcción y ejecute la solución del proyecto (Visual Studio Solution IGSTK.sln para MSVC2005, pero eso depende del compilador utilizado). Compile el proyecto según las versiones que tuvo en cuenta para ITK y VTK, todos en release y/o debug. Dependiendo de su sistema, esto puede tomar varios minutos.

Para verificar la instalación, copie los archivos HelloWorld.cxx y CMakeLists.txt del directorio `\..\IGSTK-4-2_source\Examples\Installation`. Si es necesario, agregue en el encabezado del archivo CMakeLists.txt:

```
cmake_minimum_required(VERSION 2.6)
```

Compile el proyecto generado HelloWorld.sln. Busque por consola el archivo y ejecútelo. Deberá ver el mensaje:

"IGSTK Hello...Iam using ITK too!" (figura 1a.)

Si las versiones de ITK o VTK no fueron compatibles se generaran errores en la aplicación de IGSTK, aunque el resultado de compilación de su proyecto asociado no lo haya registrado.

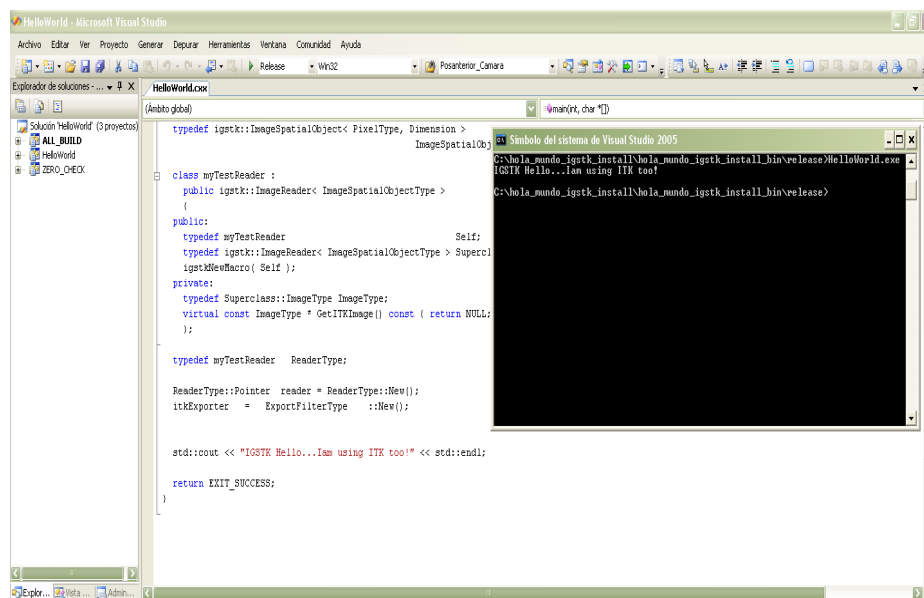


Figura 1. Prueba de instalación IGSTK. Compilación correcta.

Hola mundo IGSTK:

La mayoría de los ejemplos con GUI incluidos en el directorio de IGSTK, están compilados con FLTK. Si se incluye esta librería, una vez compilado el proyecto IGSTK, dirijase al directorio \..\IGSTK-4-2_bin\bin\debug. En la figura 3 se presenta la ejecución de las aplicaciones igstkVersionTest.exe y NeedleBiopsy.exe con la que se pueden analizar las versiones de las librerías instaladas, las funciones de lectura de imágenes DICOM de IGSTK y la aplicación de GUI con FLTK.

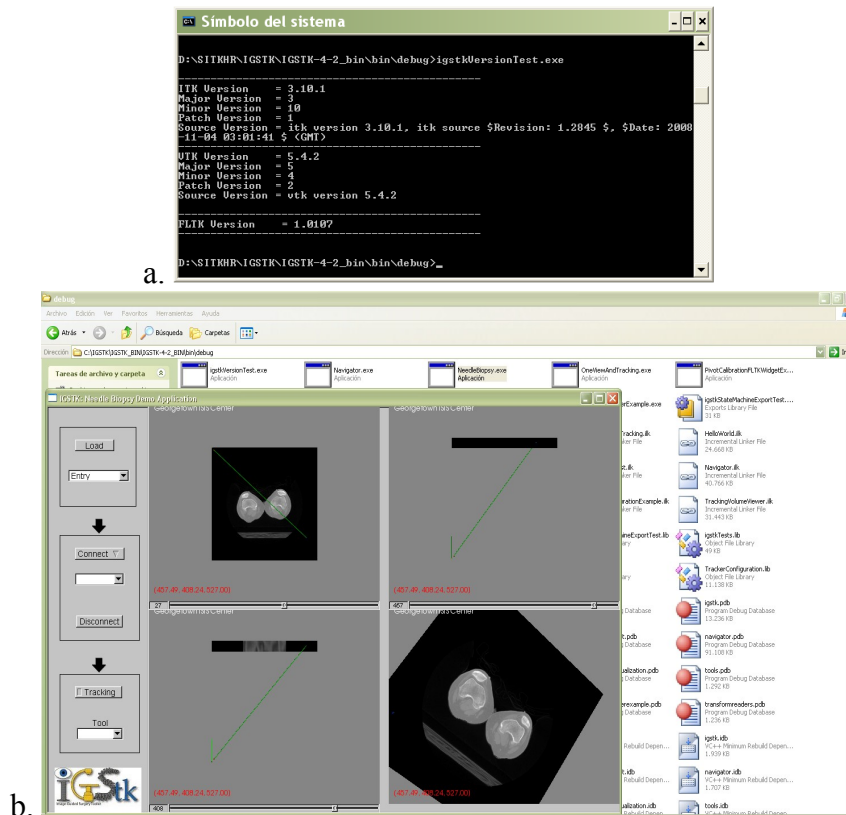


Figura 2. Ejecución Ejemplos IGSTK, GUI con FLTK. a) igstkVersionTest.exe. b) NeedleBiopsy.exe.

Ejemplo de aplicación.

Copie en un directorio los archivos que se presentan para un primer ejemplo de IGSTK con QT. Estos archivos corresponden al directorio \Sandbox-4-2\Examples\OneViewAndTrackingUsingQtWidget que también se puede descargar en <http://www.igstk.org/IGSTK/resources/software.html>, pero se han modificado las directivas de CMakeLists.txt para que se pueda compilar como un proyecto independiente con la opción de adicionar aplicaciones de vtk e itk. Puede comparar los archivos de cada caso para verificar que se adicionaron las líneas:

```
CMakeLists.txt:
cmake_minimum_required(VERSION 2.6)
PROJECT(OneViewAndTrackingUsingQTWidget)
#
# FIND IGSTK
#
FIND_PACKAGE(IGSTK)
IF(NOT IGSTK_DIR)
  MESSAGE(FATAL_ERROR "Indique donde se encuentra el directorio de IGSTK IGSTK_DIR")
ENDIF(NOT IGSTK_DIR)
INCLUDE(${IGSTK_USE_FILE})
#
# FIND ITK
#
FIND_PACKAGE(ITK)
IF (USE_ITK_FILE)
  INCLUDE (${USE_ITK_FILE})
ELSE (USE_ITK_FILE)
  MESSAGE( FATAL_ERROR "Esta aplicacion requiere ITK. No se encuentran sus componentes. Por favor
verifique su configuracion")
ENDIF (USE_ITK_FILE)
#
# FIND VTK
#
FIND_PACKAGE(VTK)
IF (USE_VTK_FILE)
  INCLUDE (${USE_VTK_FILE})
ELSE (USE_VTK_FILE)
  MESSAGE( FATAL_ERROR "Esta aplicacion requiere VTK. No se encuentran sus componentes. Por favor
verifique su configuracion")
ENDIF (USE_VTK_FILE)
#
# FIND Qt
#
FIND_PACKAGE(Qt4 REQUIRED)
IF(QT_USE_FILE)
  INCLUDE(${QT_USE_FILE})
ELSE(QT_USE_FILE)
  SET(QT_LIBRARIES ${QT_QT_LIBRARY})
ENDIF(QT_USE_FILE)

INCLUDE_DIRECTORIES (${QT_INCLUDE_DIR})

INCLUDE_DIRECTORIES(
  ${IGSTK_SOURCE_DIR}/Utilities/SceneGraphVisualization
  ${IGSTK_BINARY_DIR}/Utilities/SceneGraphVisualization
)

SET(OneViewAndTrackingUsingQTWidget_GUI_SRCS
  OneViewAndTrackingUsingQTWidgetGUI.ui
)

INCLUDE_DIRECTORIES(
  ${OneViewAndTrackingUsingQTWidget_SOURCE_DIR}
```

```

    ${OneViewAndTrackingUsingQTWidget_BINARY_DIR}
)

QT4_WRAP_UI(OneViewAndTrackingUsingQTWidget_QT_UI_SRCS
  {OneViewAndTrackingUsingQTWidget_GUI_SRCS})

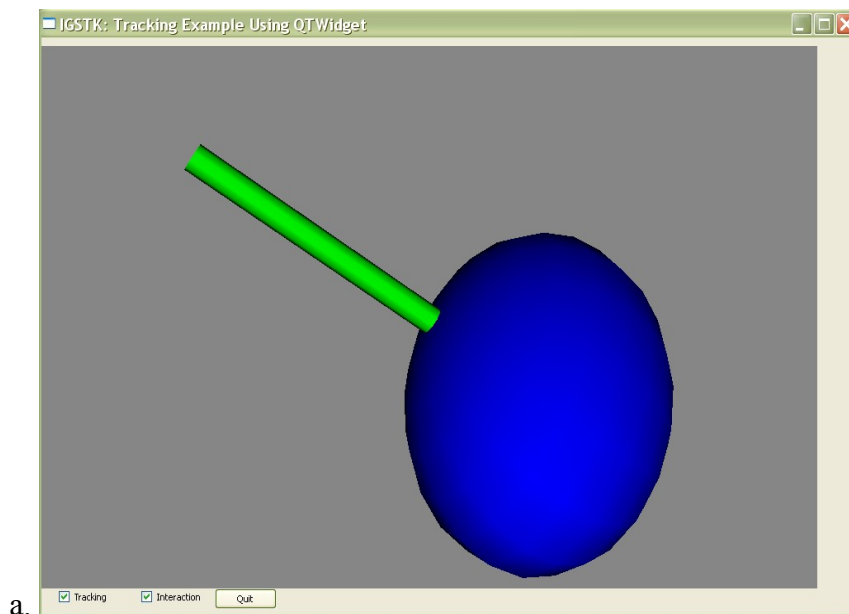
SET(OneViewAndTrackingUsingQTWidget_SRCS
  OneViewAndTrackingUsingQTWidget.cxx
  OneViewAndTrackingUsingQTWidgetGUI.cxx
)
QT4_AUTOMOC(${OneViewAndTrackingUsingQTWidget_SRCS})

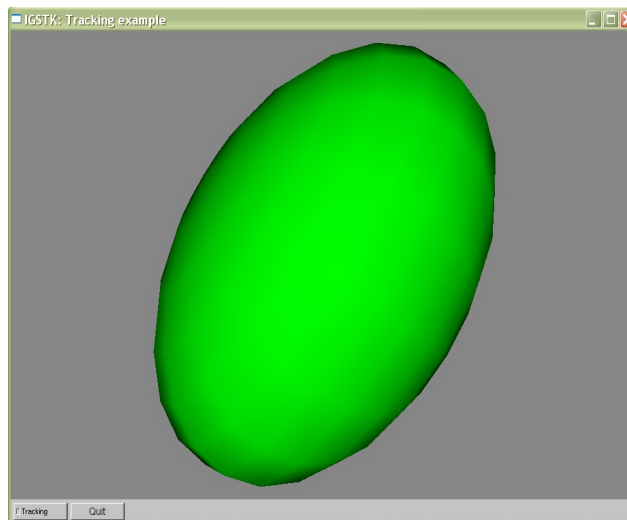
ADD_EXECUTABLE(OneViewAndTrackingUsingQTWidget
  {OneViewAndTrackingUsingQTWidget_QT_UI_SRCS} ${OneViewAndTrackingUsingQTWidget_SRCS})

TARGET_LINK_LIBRARIES(OneViewAndTrackingUsingQTWidget
  IGSTK
  ${QT_LIBRARIES})

```

Ejecute CMake indicando el directorio de construcción del primer ejemplo. Compile la solución del proyecto, teniendo en cuenta que se haga en modo Release o Debug tal como se compilaban las librerías IGSTK, ITK y VTK. En la figura 3 se muestran las ejecuciones del primer ejemplo con Qt y FLTK. Verifique las funciones de conexión de slots de Qt dadas en el archivo OneViewAndTrackingUsingQTWidgetGUI.cxx. Interactúe con las opciones de tipo, tamaño y ubicación de las primitivas de la escena, así como opciones de interacción del ratón (típica de qvtk o tracking) en OneViewAndTrackingUsingQTWidget.cxx.





b.

Figura 3. Ejemplo aplicación IGSTK. a) OneViewAndTrackingUsingQTWidget. b) OneViewAndTrackingNewUsingFLTKWidget.

REFERENCIAS

- [1] Cleary K.: The Book *For release 2.0*. Insight Software Consortium.
- [2] Cleary K., Cheng P., et. al. IGSTK: The Book *For release 4.2*. Insight Software Consortium. Disponible en <http://www.igstk.org/IGSTK/help/documentation.html>.
- [3] http://public.kitware.com/IGSTKWIKI/index.php/Download_IGSTK.