

## Manual de configuración y compilación de un proyecto con OpenCV4 en Windows

<http://umh1782.edu.umh.es>

- Requisitos:**
- Windows 7,8,10 - (64bits)
  - Microsoft Visual Studio C++ 2017 Community Edition
  - Librería OpenCV instalada y configurada

[Tutorial Instalación](#)

Más información en <http://opencv.org>

### 1) Crear el proyecto:

- Ejecutaremos la aplicación **Microsoft Visual Studio C++ 2017** y en el menú **File/Fichero** seleccionaremos **New Project/Nuevo Proyecto** (Figura 1)
- **Nota:** si hemos seleccionado el idioma en Visual Studio los menús y opciones se nos mostrarán en español.

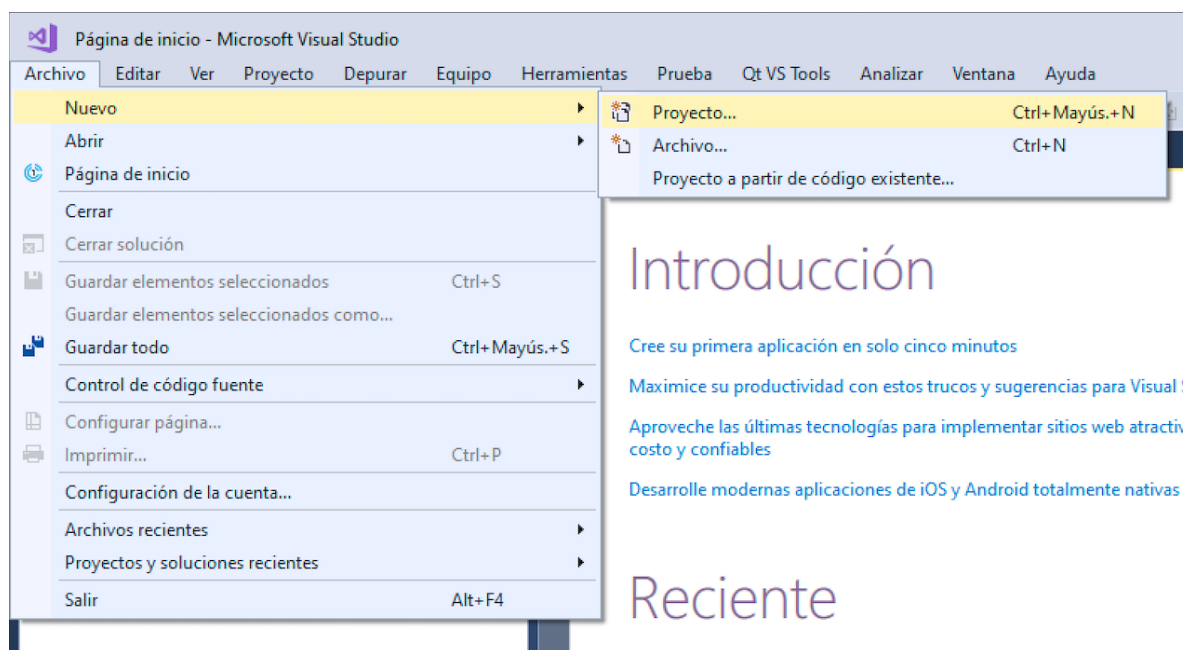


Figura 1 Crear run nuevo proyecto

- Nos aparecerá una ventana donde seleccionaremos el tipo de proyecto seleccionaremos: **'Empty Project/Proyecto Vacío'**. (Figura 2)
- Le asignaremos un nombre indicando el directorio donde deseamos que se cree el proyecto y **desactivaremos** la casilla **'Crear directorio para la solución'**. *No te preocupes, ya que aun así se creará el proyecto dentro de una carpeta con el mismo nombre del proyecto.*

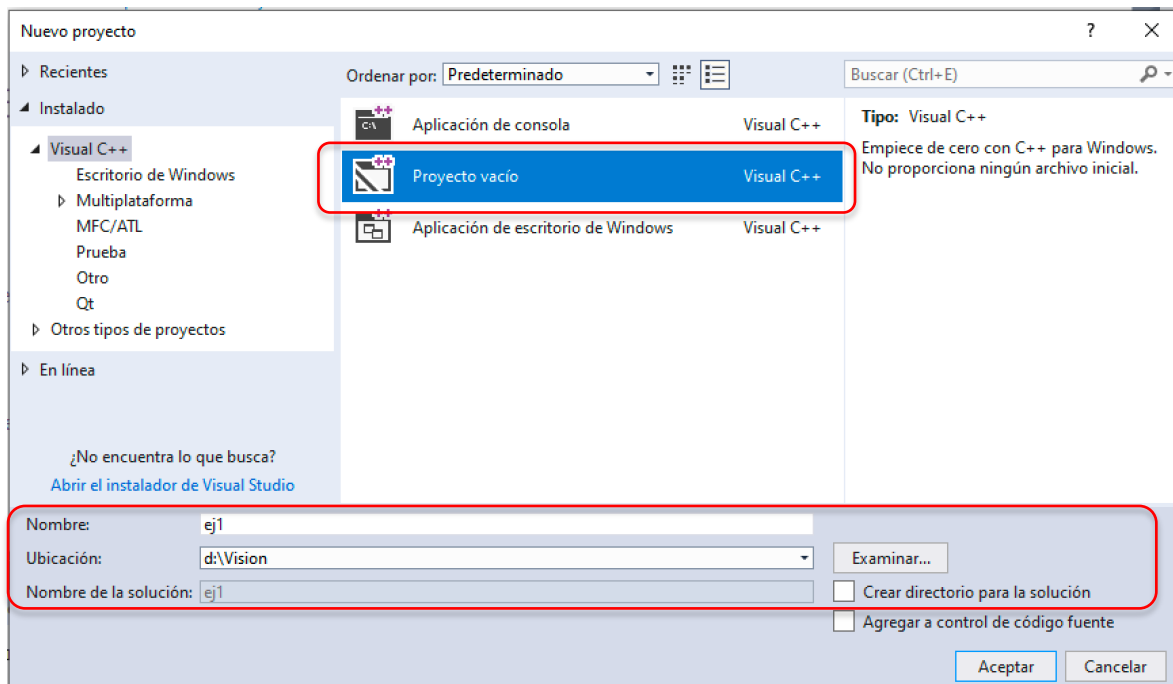


Figura 2 Selección tipo de proyecto y ubicación

- **Nota:** en la carpeta seleccionada no debe existir un proyecto previo con el mismo nombre.
- Pulsando el botón '**Aceptar**' ya tendremos creado nuestro proyecto.

Deberíamos tener la ventana de **Administración de Propiedades** del proyecto visible (Figura 3). Si estuviera otra ventana visible la seleccionaremos en la pestaña del pie de la ventana o desde el menú '**Ver**'

**Nota:** Si no estuviera disponible esta ventana, la activaríamos con la opción del menú: '**Herramientas** > **Personalizar**' Pestaña **Comandos**' - seleccionar '**Barra de menús**' > **Ver**, Pulsar botón '**Reestablecer todo**'

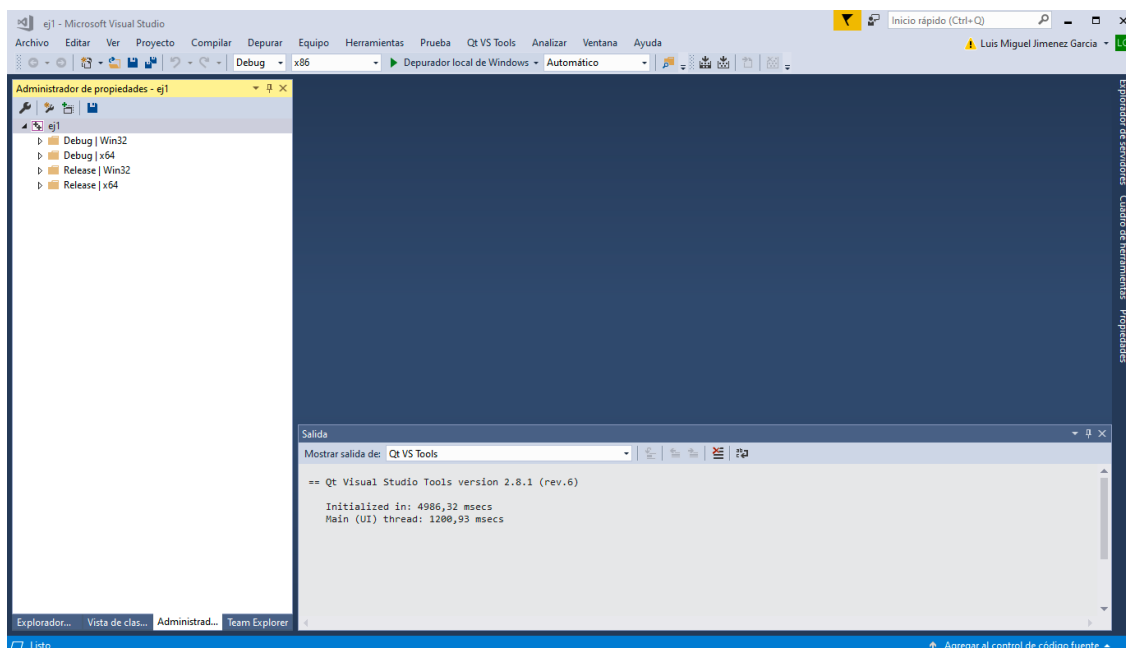


Figura 3 Ventana principal del proyecto

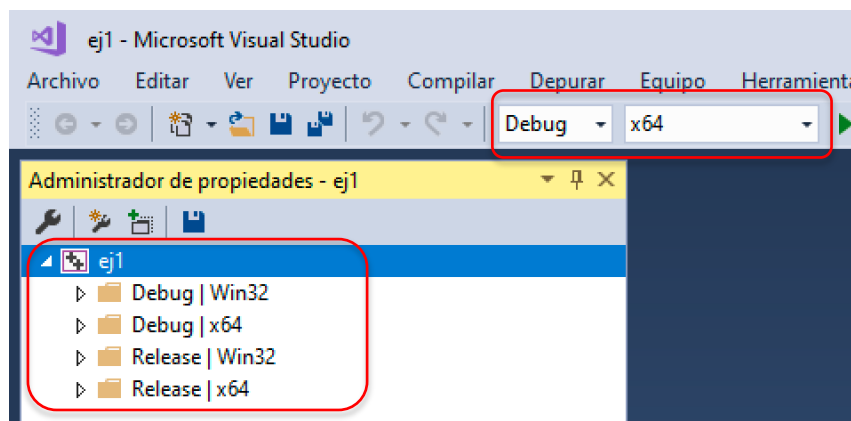


Figura 4. Configuraciones compilación

En la Figura 4 podemos observar que ha creado 4 configuraciones distintas (*Win32/x64 - Debug/Release*). Como compilaremos con la librería OpenCV4 en arquitectura 64bits, borraremos las dos configuraciones etiquetadas *Win32*. También seleccionaremos como configuración de trabajo '**Debug/x64**' (desplegable superior Figura 4)

Haciendo click con el botón derecho del ratón sobre ellas, nos muestra un menú desplegable en el que seleccionaremos la opción '**Quitar**' para las configuraciones *Win32* (Figura 5)

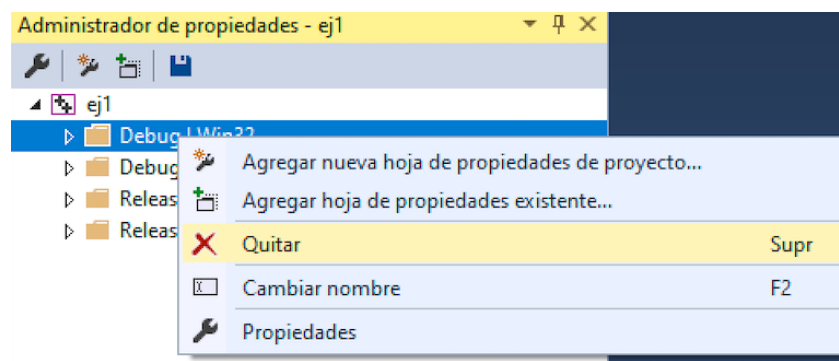


Figura 5. Eliminar configuración

A partir de aquí deberemos configurar cada una de las opciones de compilación de forma específica para cada proyecto, tal como se verá en los siguientes apartados.

### **3) Configurar el proyecto (1ª vez):**

Vamos a describir un método para configurar un proyecto que utilice la librería OpenCV, que es reutilizable en varios proyectos o programas. Por tanto distinguiremos dos casos:

- 1) **Primer proyecto**: crearemos unos ficheros especiales con toda la configuración
- 2) **Otros proyectos**: reutilizaremos los ficheros anteriores, por lo que no será necesario repetir los pasos previos

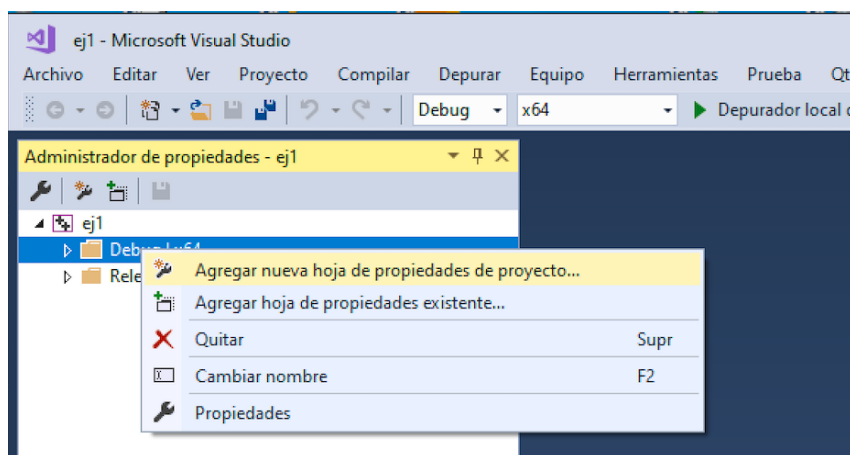
En la ventana del gestor de configuración (**'Property Manager'**) nos mostrará la página creada con dos apartados (***Debug x64/Release x64***). En Visual C++ tenemos dos modos de compilación, cada uno de ellos con una configuración independiente:

- **Modo Debug**: el compilador añade código extra que nos ayudara a depurar los errores de nuestro programa. Este es el modo que utilizaremos habitualmente
- **Modo Release**: cuando el programa está terminado, la compilación se optimiza eliminando el código de depuración, obteniendo así un ejecutable más pequeño y más rápido.

Vamos a describir la configuración de los dos modos:

#### **a) Configuración modo Debug:**

- Pulsaremos el botón derecho del ratón sobre nodo **'Debug x64'**, desplegando el menú contextual en el que seleccionaremos ***'Agregar nueva hoja de propiedades al proyecto'*** (figura 6)



*Figura 6. Crear Hoja de Propiedades*

- Nos abrirá una nueva ventana donde asignaremos el nombre del fichero de configuración (***opencv455\_vc15\_x64\_debug***) (Figura 7) Pulsando el botón **'Agregar'** creará el fichero de configuración en la carpeta de nuestro proyecto (***opencv455\_vc15\_x64\_debug.props***).

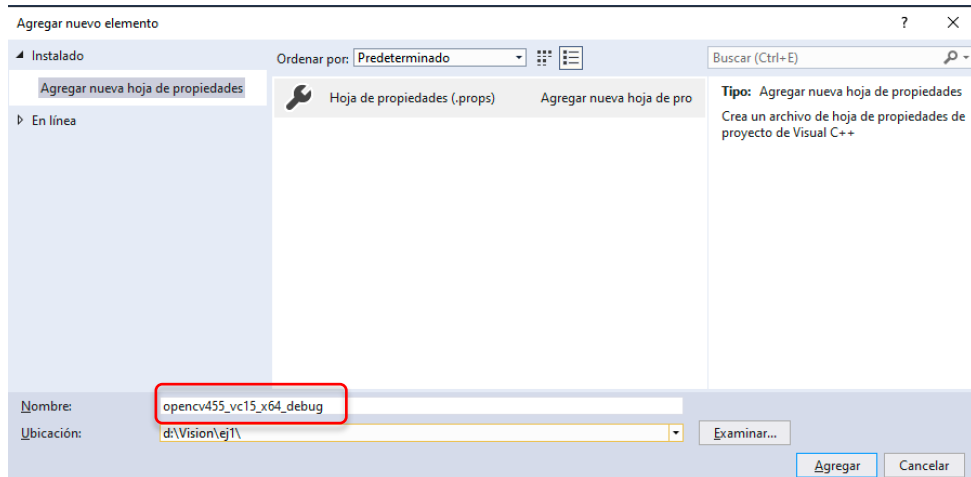


Figura 7. Nueva Hoja de Propiedades

- Pulsaremos el botón derecho del ratón sobre la etiqueta '**opencv455\_vc15\_x64\_debug**' de la hoja de propiedades del modo '**Debug**' que hemos creado. En el menú contextual seleccionaremos la opción '**Propiedades**'. Esto abrirá la ventana donde realizaremos la configuración del proyecto (Figura 8).

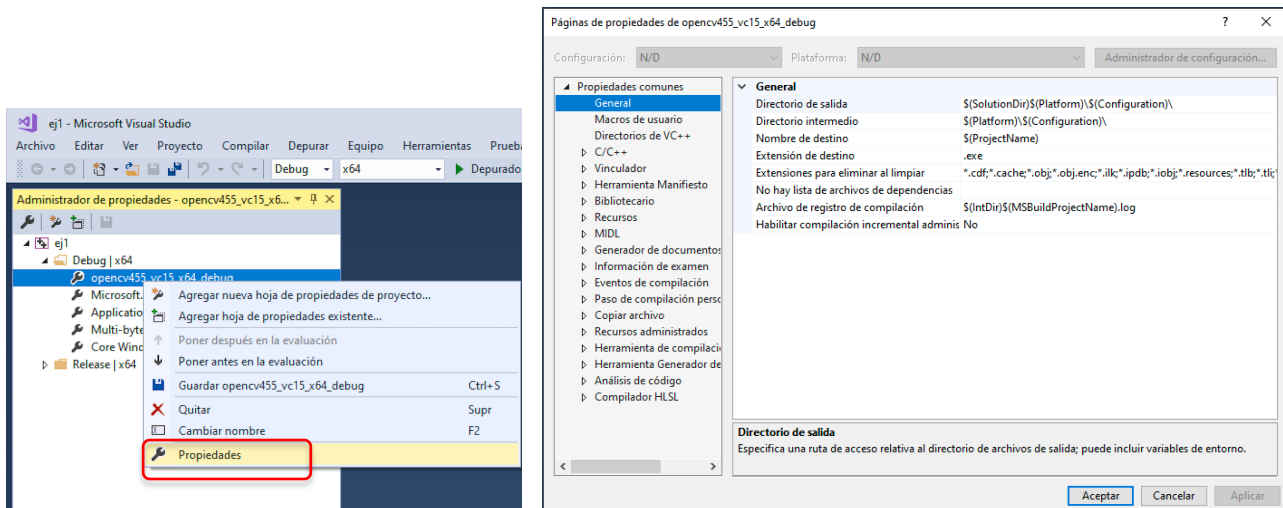


Figura 8. Configurar Hoja de Propiedades

A continuación modificaremos tres apartados del fichero de configuración:

- Directorios VC++ -> **Directorios de archivos de inclusión / Include Directories**
- Directorios VC++ -> **Directorios de archivos de bibliotecas / Library Directories**
- Vinculador/Linker -> Entrada/Input -> **Dependencias Adicionales/Additional Dependencies**

Vamos por partes:

### a.1) Directorio de archivos de inclusión:

- Editaremos la opción **Directorios VC++->Directorios de archivos de inclusión**, pulsando sobre la flecha azul del lateral derecho nos muestra un menú con la opción **<Editar>** que pulsaremos y abrirá una ventana de edición. Añadiremos el directorio: **c:\opencv\build\include** (Figura 9)

Opcionalmente, si tenemos instaladas librerías adicionales, añadiremos la carpeta: **c:\opencv\include**

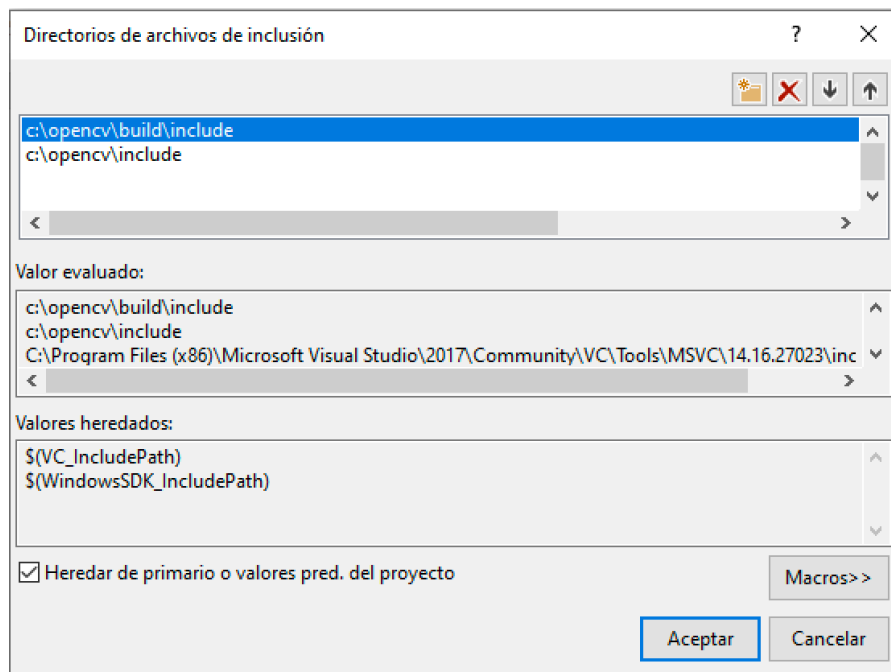
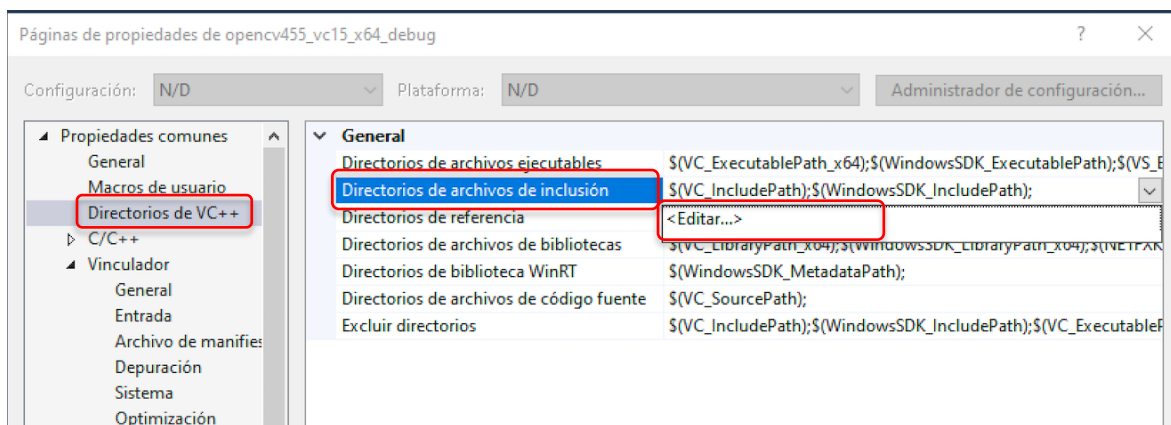


Figura 9. Configura directorio de cabeceras

**NOTA:** si queremos usar la librería compilada con los módulos **contrib** debemos modificar los directorios:

- **c:\opencv-contrib\build\include**
- **c:\opencv-contrib\include**

### a.3) Directorio de librerías:

- Editaremos la opción **Directorios VC++ -> Directorios de archivos de bibliotecas**, pulsando sobre la flecha azul del lateral derecho nos muestra un menú con la opción **<Editar>** que pulsaremos y abrirá una ventana de edición.
- Añadiremos el directorio específico de nuestra arquitectura y compilador (Figura 10):
  - Windows con arquitectura de 64 bits: **c:\opencv\build\x64\vc15\lib**
  - Opcionalmente añadiremos la carpeta: **c:\opencv\lib**

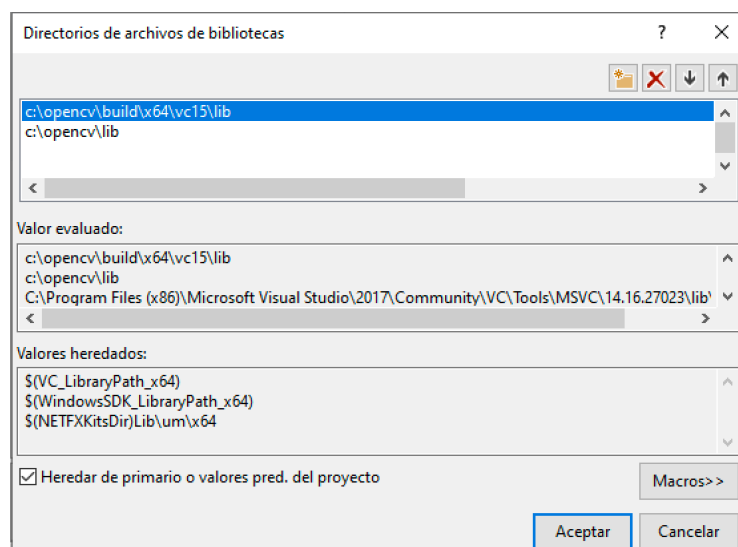
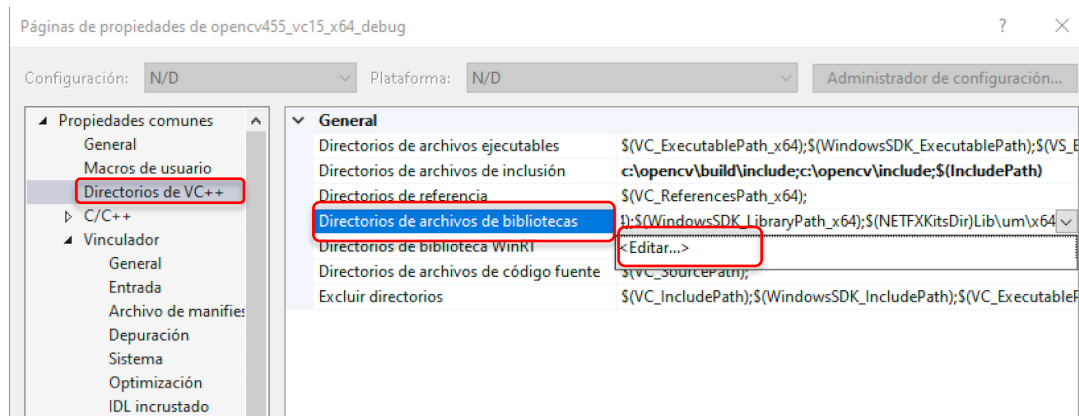


Figura 10. Configuración Directorios Bibliotecas

**NOTA:** si queremos usar la librería compilada con los módulos **contrib** debemos modificar los directorios:

- **c:\opencv-contrib\build\x64\vc15\lib**
- **c:\opencv-contrib\lib**

### a.3) Librerías:

- Editaremos la opción **Vinculador -> Entrada -> Dependencias Adicionales**, pulsando sobre la flecha azul del lateral derecho nos muestra un menú con la opción **<Editar>** que pulsaremos y abrirá una ventana de edición. Añadiremos el listado de librerías de la **opencv** (Figura 11)

opencv\_world455d.lib

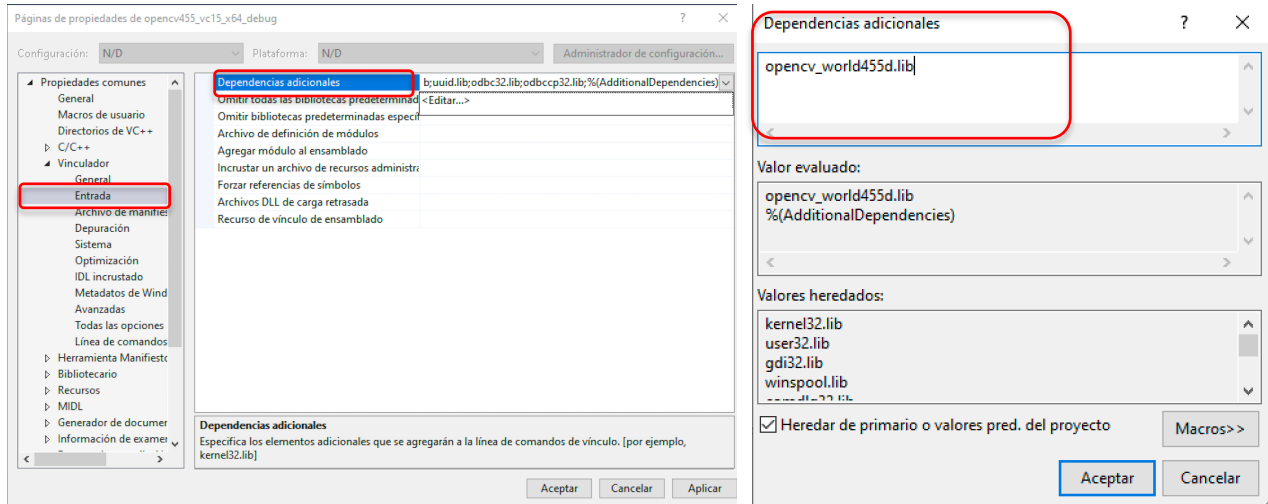


Figura 11. configuración de librerías

### b) Configuración modo Release:

- Repetiremos los mismos pasos del apartado anterior pero sobre la etiqueta **'Release'**. El fichero de propiedades lo llamaremos **'opencv455\_vc15\_x64\_release'**. La configuración de los directorios será la misma que el apartado anterior solo modificaremos la selección de librerías, en lugar de las versiones de depuración de las librerías de la **OpenCV**, deberemos incluir las librerías sin información de depuración:

opencv\_world455.lib

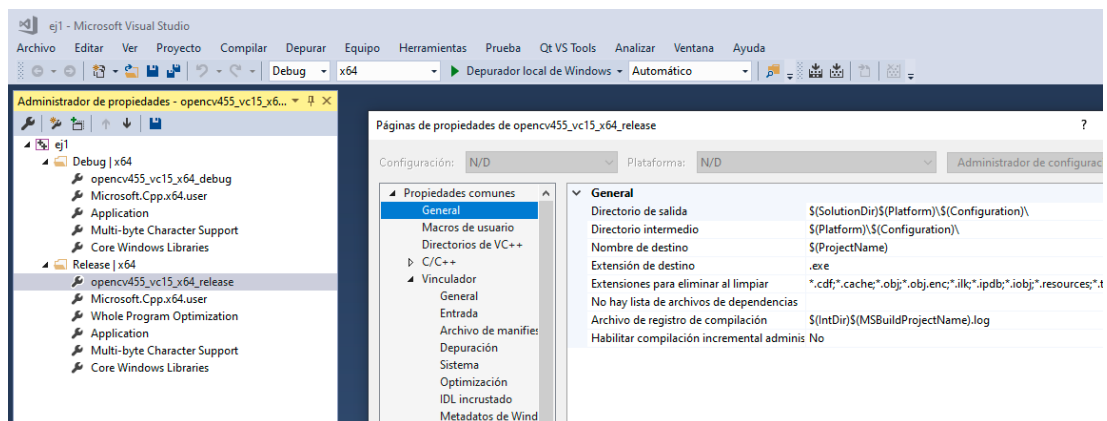


Figura 12. Fichero propiedades modo Release



Al terminar este apartado tendremos en la carpeta de nuestro proyecto dos ficheros con toda la configuración que guardaremos para uso futuro:

***opencv455\_vc15-x64\_debug.props***  
***opencv455\_vc15-x64\_release.props***

### **Configuración opcional**

En el tutorial de instalación de la librería OpenCV vimos como configurar la variable de entorno (**PATH**) con la ubicación de las **DLLs** de la librería. Si estamos trabajando con diferentes versiones de la librería o diferentes compiladores, podemos realizar esta configuración de forma específica para nuestro proyecto.

- Pulsaremos el botón derecho del ratón sobre nodo principal con el nombre del proyecto '**ej1**, (**Pestaña Explorador de Soluciones**)' desplegando el menú contextual en el que seleccionaremos '**Propiedades**' (Figura 13).
- **Nota:** debemos pulsar sobre el nombre del proyecto, no de la solución que está justo encima.

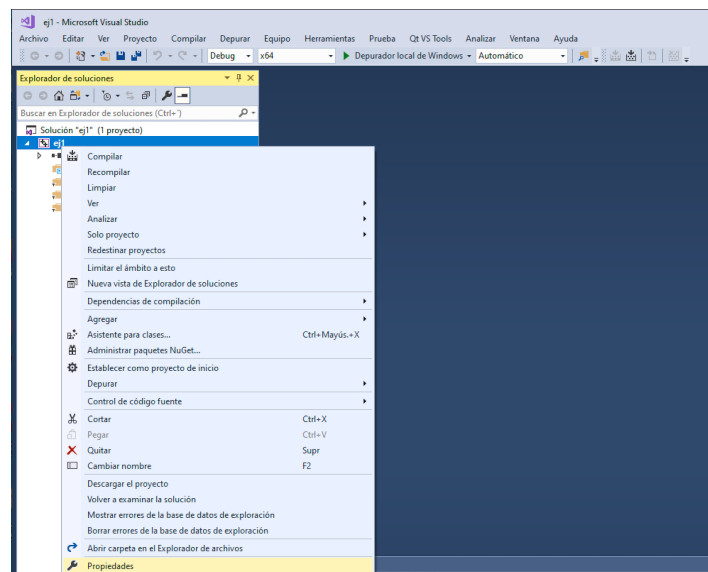


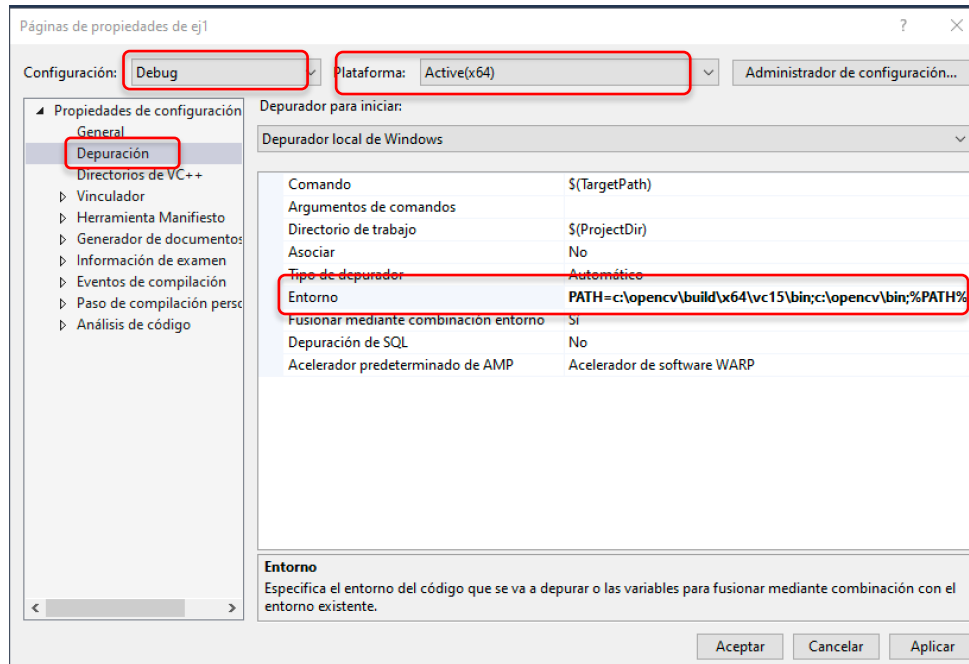
Figura 13. Propiedades del Proyecto (PATH)

- Elegiremos la **Configuración /Plataforma (Debug/x64)** que deseamos configurar. Marcaremos la opción '**Depuración**' y editaremos la opción '**Entorno**'.

Para el compilador VC15 (2017) y arquitectura **x64** pondríamos:  
***PATH=c:\opencv\build\x64\vc15\bin;c:\opencv\bin;%PATH%***

**NOTA:** si queremos usar la librería compilada con los módulos **contrib** debemos modificar los directorios:

***PATH=c:\opencv-contrib\build\x64\vc15\bin;c:\opencv-contrib\bin;%PATH%***

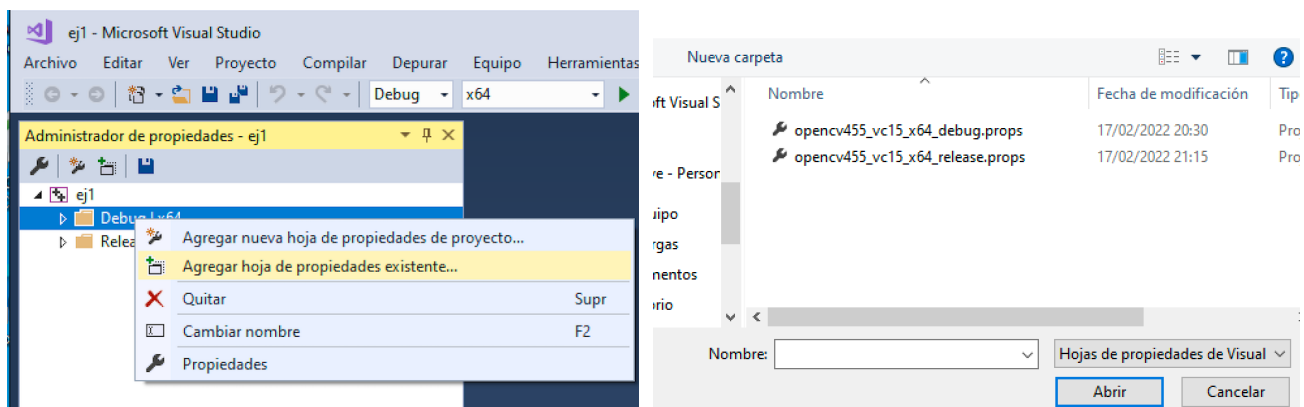


Otra variable que podemos configurar es **OPENCV\_LOG\_LEVEL=OFF** si queremos desactivar la información de cada llamada en modo Debug.

#### **4) Configurar el proyecto a partir de una página de propiedades previa:**

En este caso reutilizaremos los ficheros de configuración creados en el proyecto anterior.

- Copiaremos los ficheros '**opencv455\_vc15-x64\_debug.props**' y '**opencv455\_vc15-x64\_release.props**', en la carpeta de nuestro nuevo proyecto.
- En la ventana '**Administrador de Propiedades**' de nuestro proyecto, pulsando con el botón derecho del ratón sobre las pestañas '**Debug**' y '**Release**' añadiremos las hojas de propiedades correspondientes con la opción del menú '**Agregar Hoja de Propiedades Existente**'



- Realizaremos, si lo deseamos, la configuración opcional de las variables de entorno para depuración.

## 5) Añadir el código fuente de nuestro programa:

- Descargaremos el fichero de ejemplo '**ej1.cpp**' del blog de la asignatura. Lo copiaremos en la carpeta creada para nuestro proyecto.
- Activaremos la vista del '**Explorador de Soluciones**'. Menú '**View-> Explorador de Soluciones**'
- Sobre el nodo de la aplicación (**ej1**) desplegaremos las subcarpetas y pulsaremos el botón derecho del ratón sobre la carpeta '**Archivos de origen/Source Files**'. Seleccionaremos la opción '**Agregar/Elemento Existente**' y cargaremos el fichero de código **ej1.cpp**.

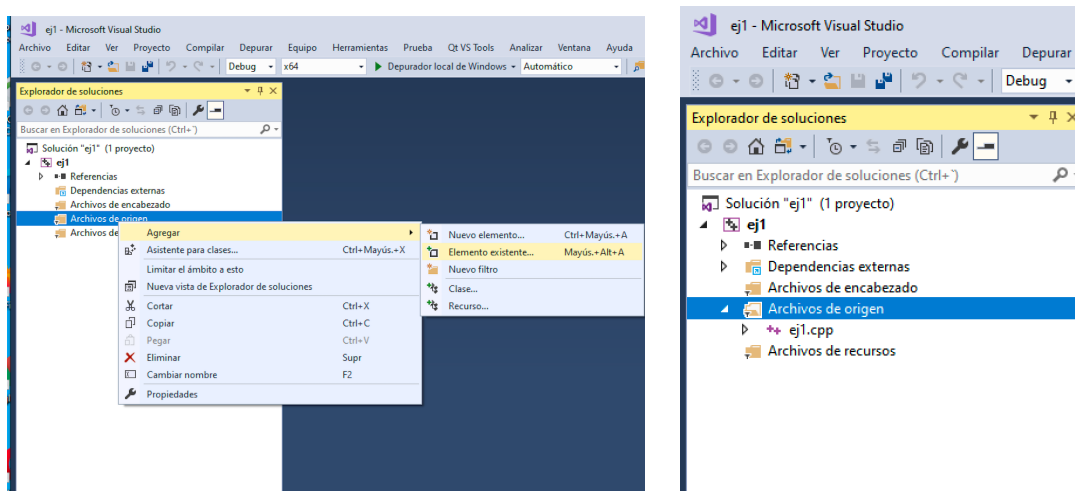


Figura 14. Añadir fichero de código fuente al proyecto

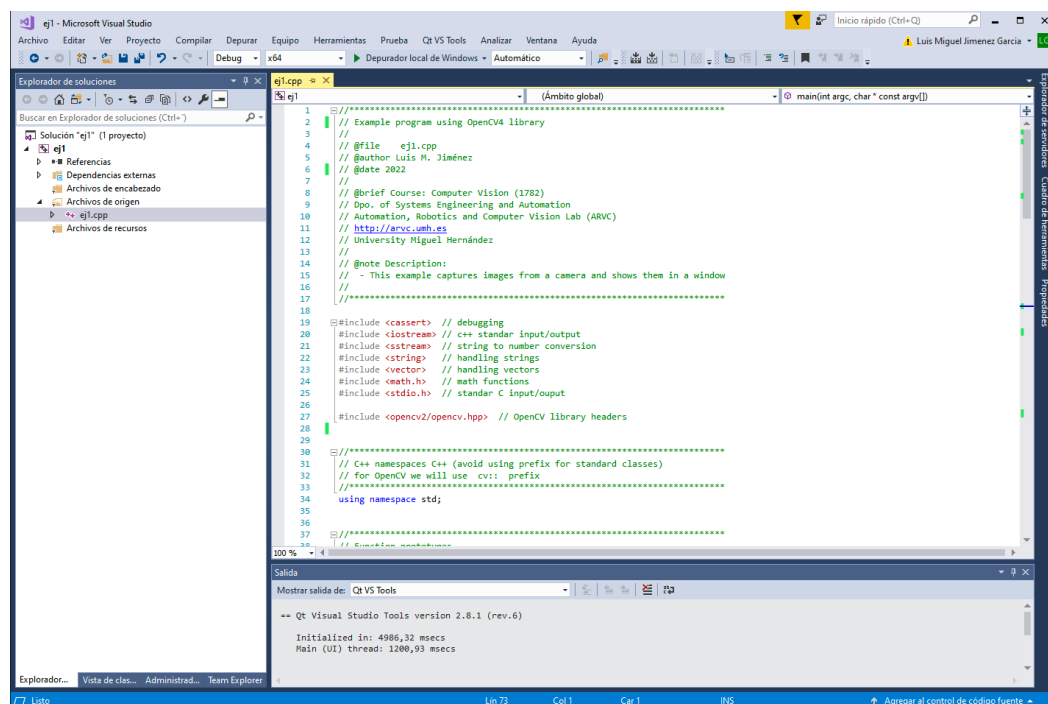


Figura 15. Editor código

- A continuación compilaremos el programa (Menú '**Compilar/Compilar Solución**' o **F7** Figura 16) y lo ejecutaremos (Menú '**Depurar/Iniciar Depuración**' o **F5** Figura 17). Disponemos también de botones de acceso rápido a estas opciones (figura 18)

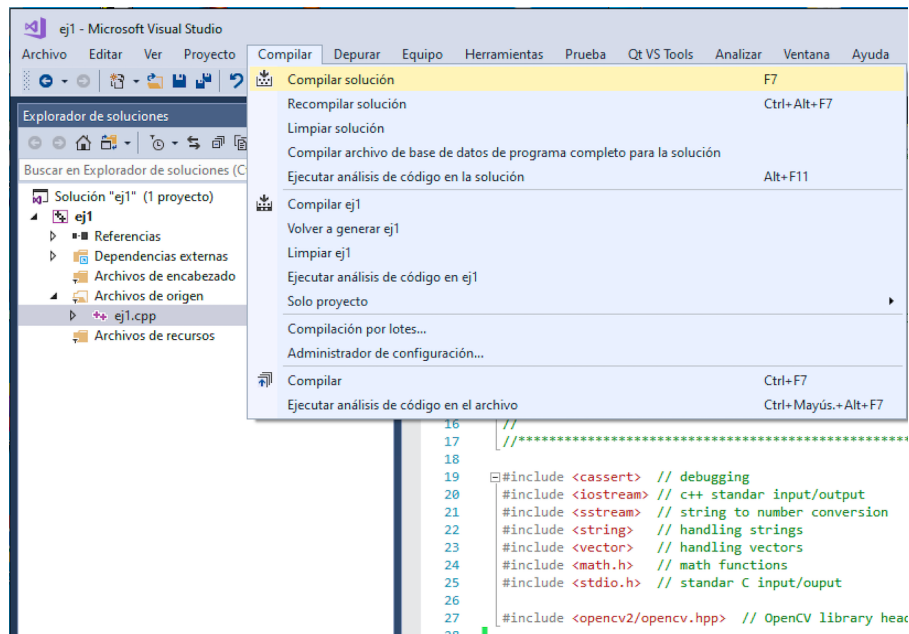


Figura 16. Compilar el proyecto

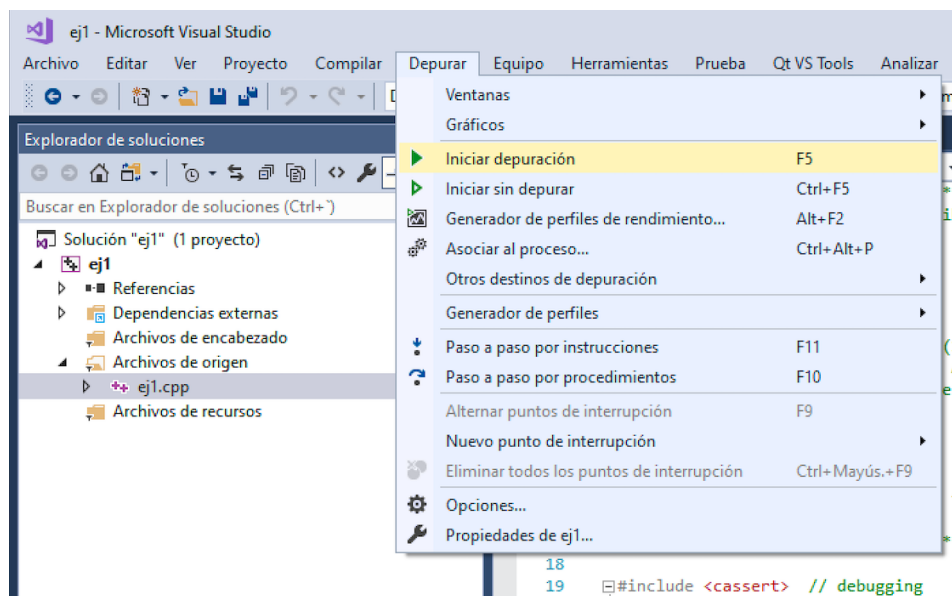


Figura 17. Ejecutar/Depurar el proyecto

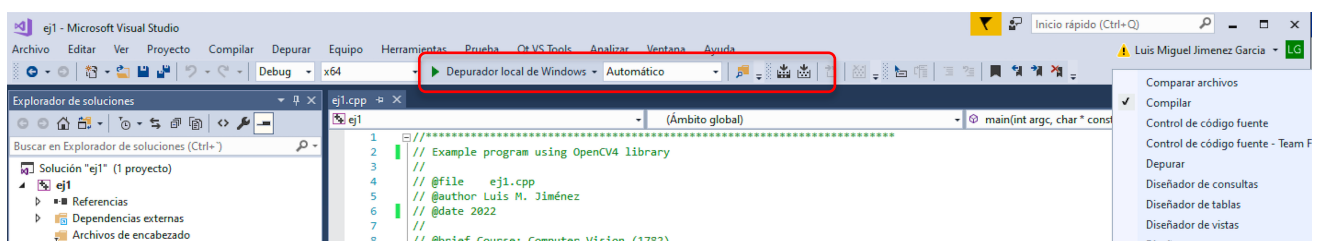


Figura 18. Botones de acceso rápido: Compilar/Depurar