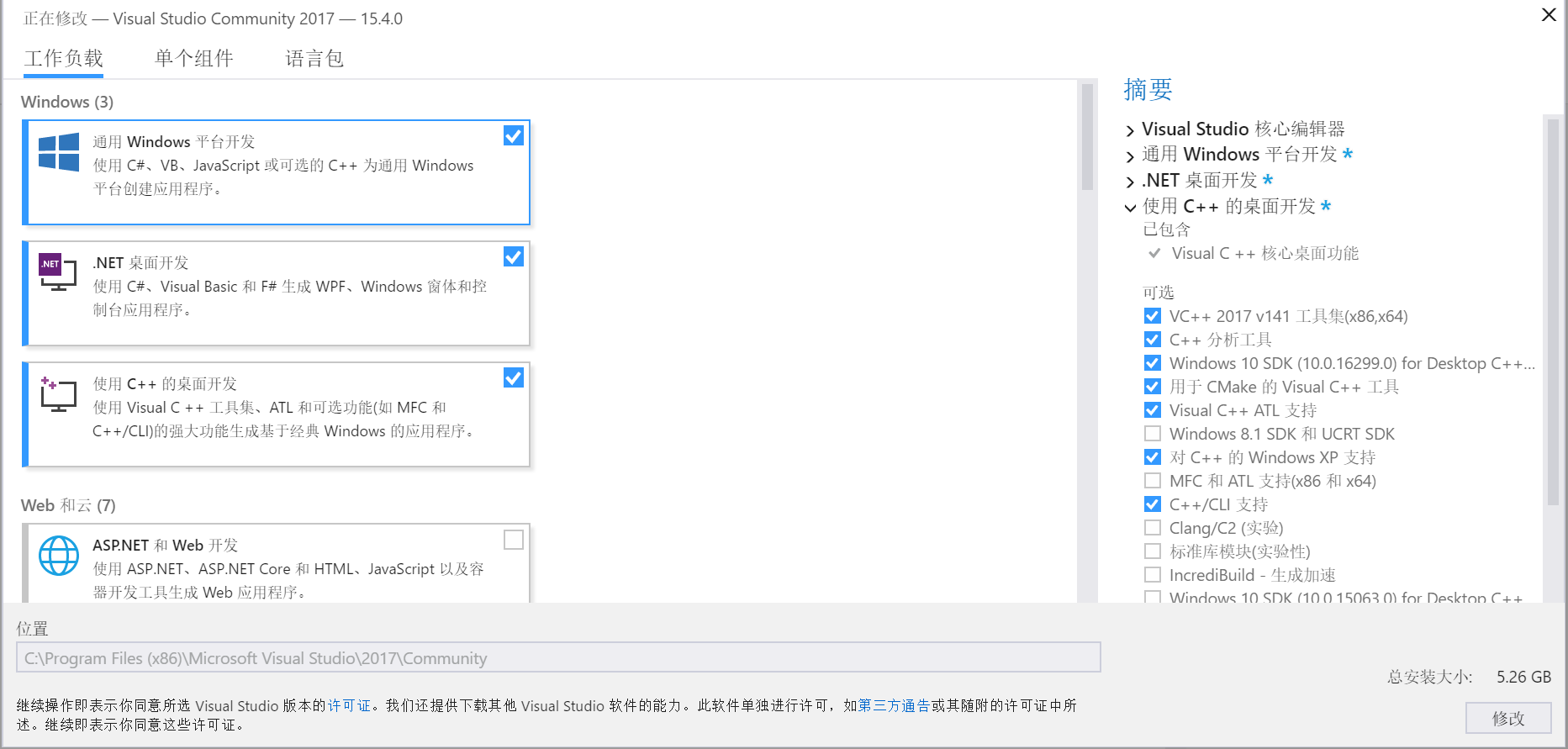
**WIN10下为VS2017配置OpenCV-3.2.0（64位编译环境）**

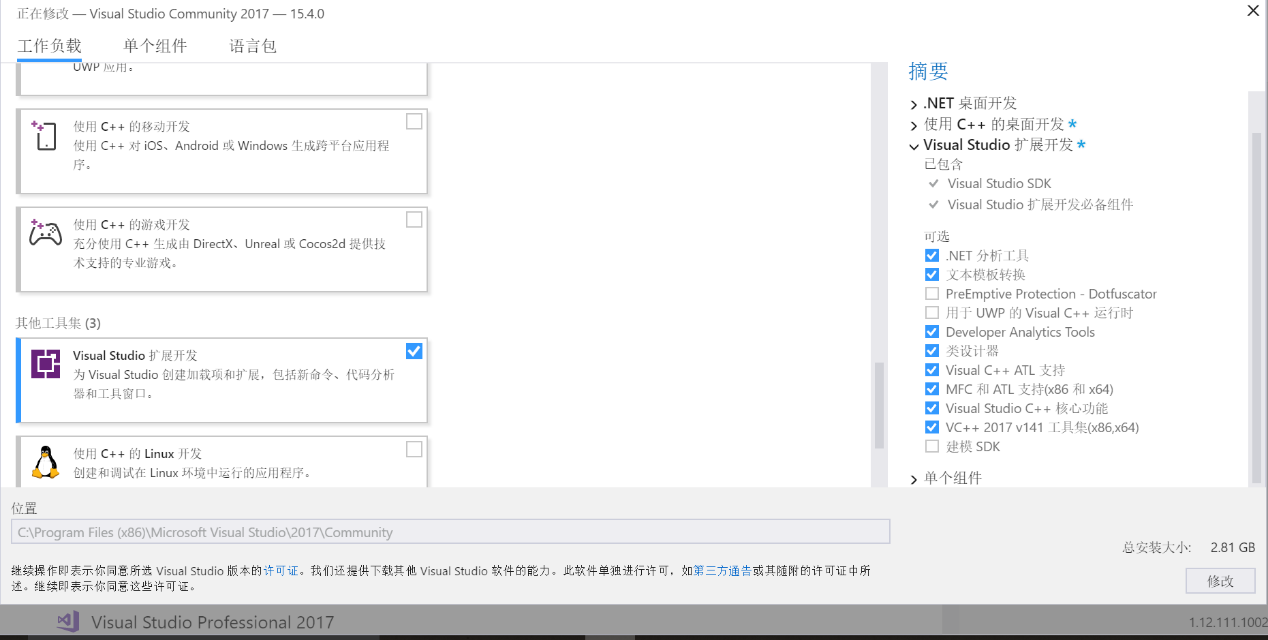
开发环境搭载：

1. VS2017版本
2. OpenCV-3.2.0版本
3. 为电脑的属性配置环境变量，以及在VS2017的项目属性中设置头文件、库文件、动态链接库等。
4. 安装VS2017

1、工作负载安装

进入Microsoft官网，下载VS2017进行安装。在安装过程中，出现界面要求选择安装开发平台与插件，可按照图1所示的工作负载选项进行勾选，选择Windows(3)下的“通用Windows平台开发”“.NET桌面开发”“使用C++的桌面开发”，选择“其他工具集”下的“Visual Studio扩展开发”同时选择右侧摘要里可选的选项“类设计器”，如图1.1所示。

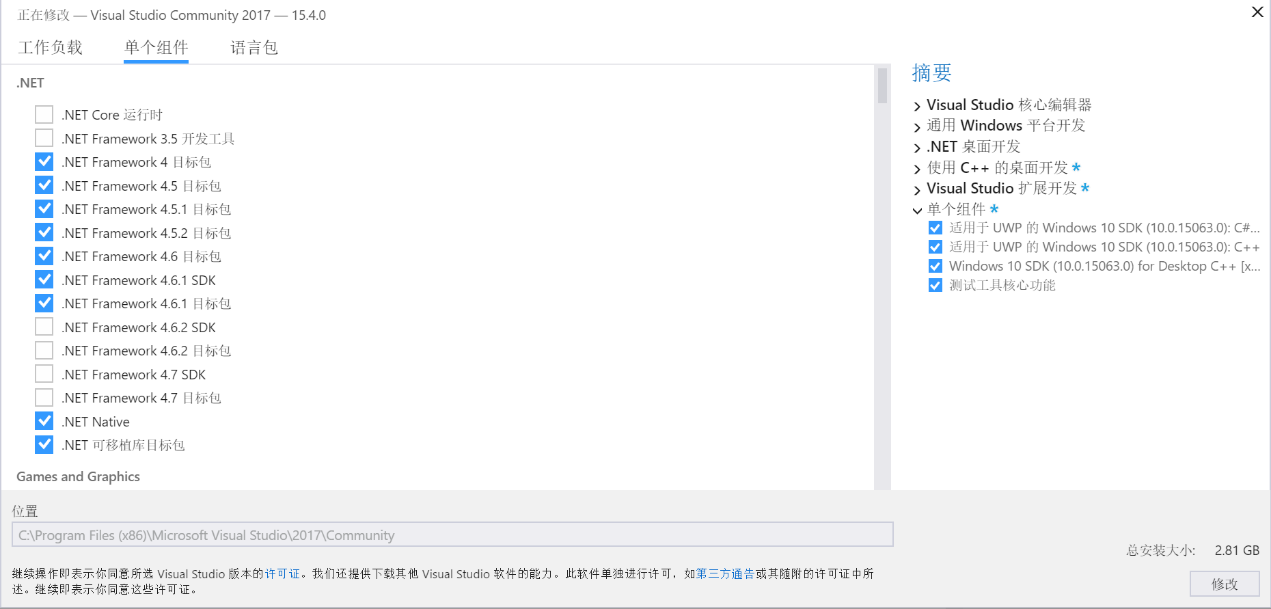




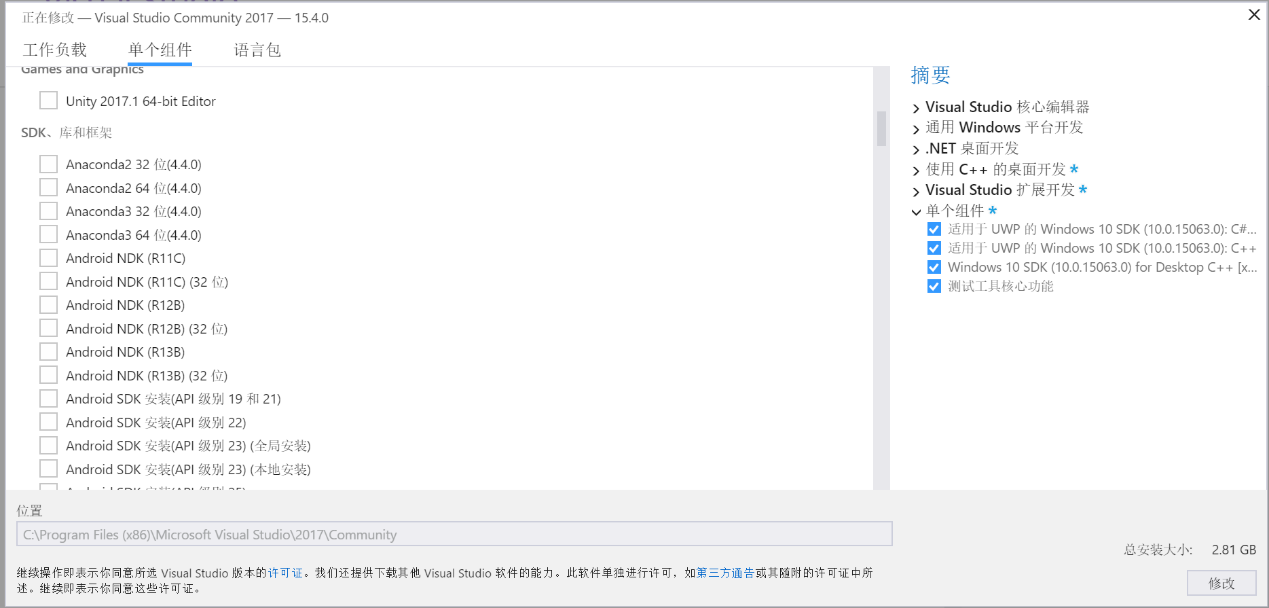
**图1.1 VS2017安装过程中选择开发平台**

2、单个组件安装

单个组件安装需要勾选的选项如图1.2~图1.11所示。



**图1.2 单个组件安装选项1**



**图1.3 单个组件安装选项2**



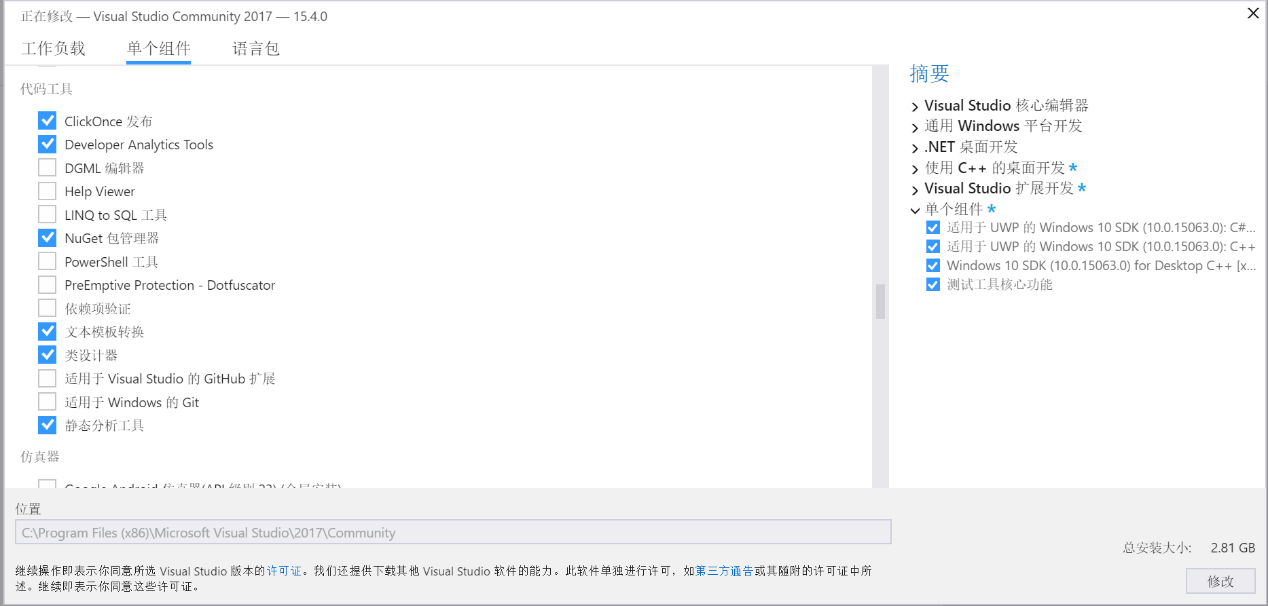
**图1.4 单个组件安装选项3**



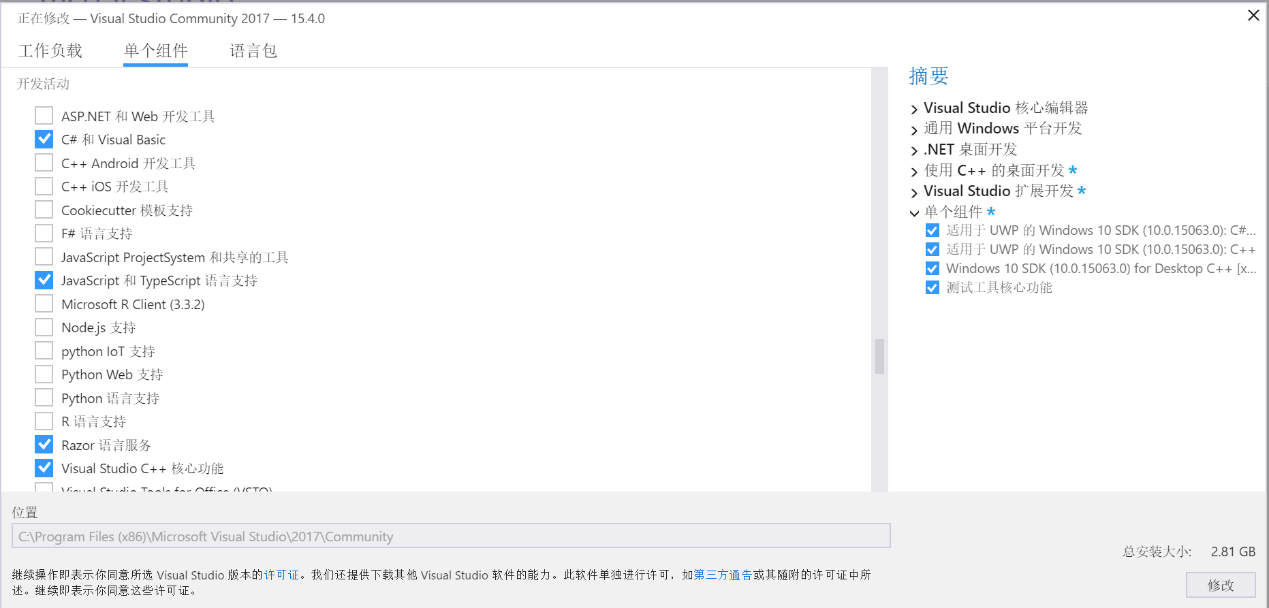
**图1.5 单个组件安装选项4**

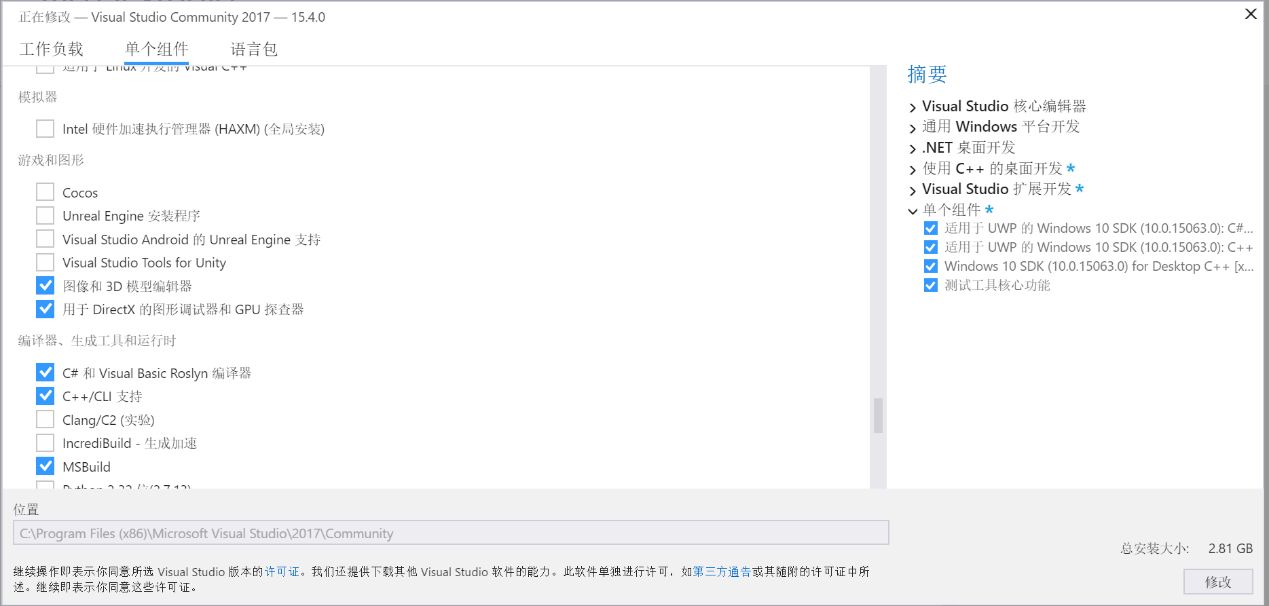


**图1.6 单个组件安装选项5**



**图1.7 单个组件安装选项6**

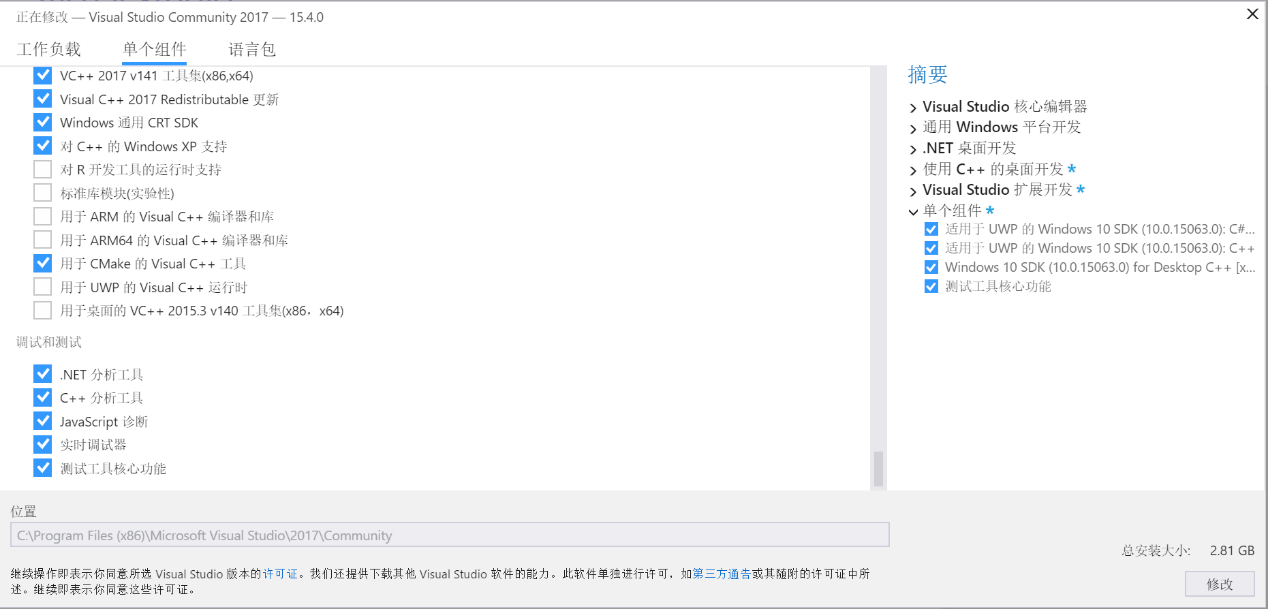
**图1.8 单个组件安装选项7**



**图1.9 单个组件安装选项8**



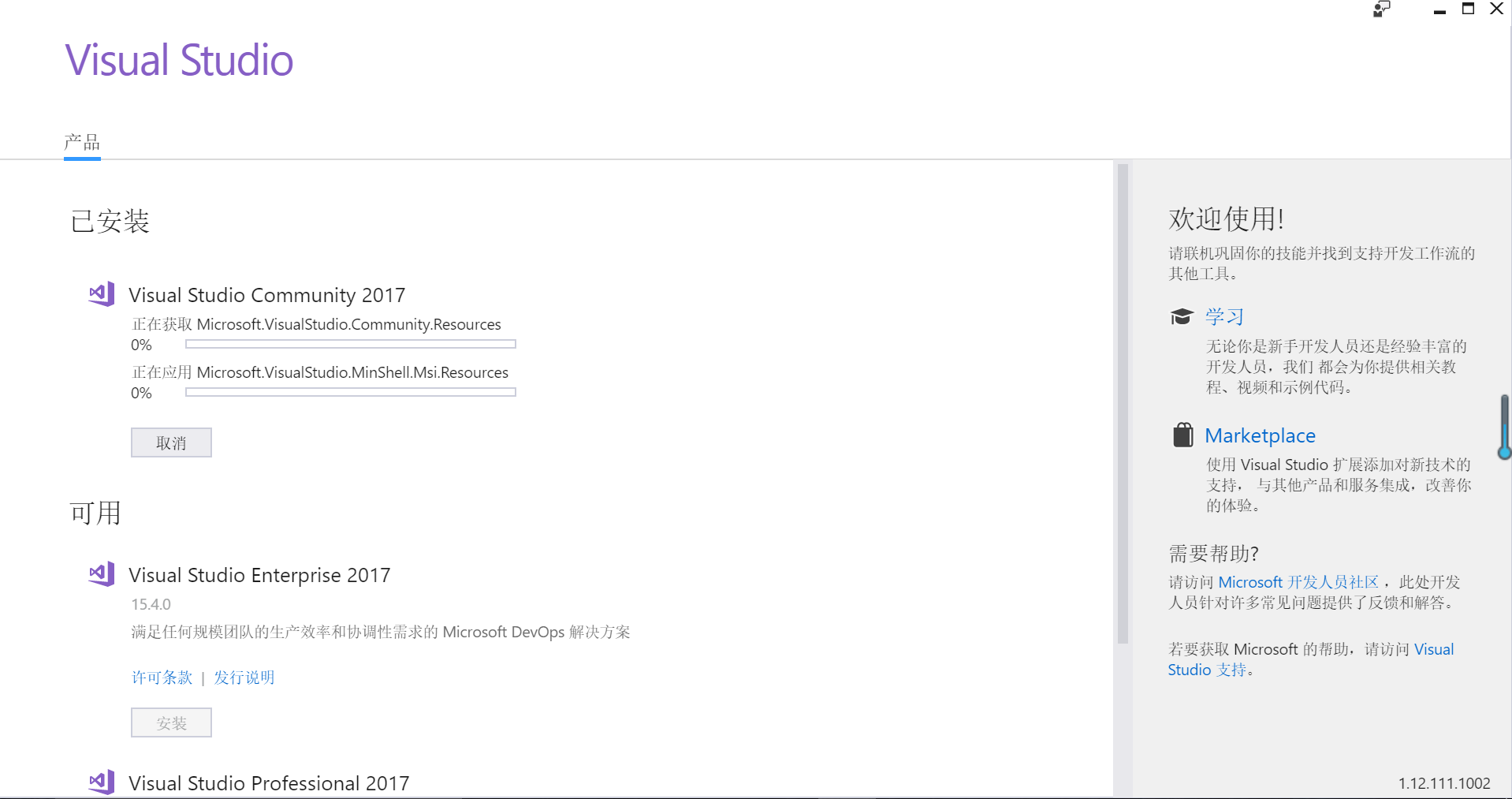
**图1.10 单个组件安装选项9**



**图1.11 单个组件安装10**

1. 安装Visual Studio进程

勾选好需要安装的平台和组件后，点击“确定”，开始安装，如图1.12所示。



**图1.12 VS2017安装进程**

二、安装OpenCV-3.2.0

从OpenCV官网下载OpenCV-3.2.0，双击下载好的.exe文件，如图2.1所示。弹出界面选择释放的路径，本机选择释放在F盘。安装好的OpenCV包含的文件应如图2.2所示，本机选择将OpenCV安装在F盘。以下的步骤均是以本机的安装路径为例讲述，如有不同，请记得更改路径。



**图2.1 OpenCV-3.2.0安装包**

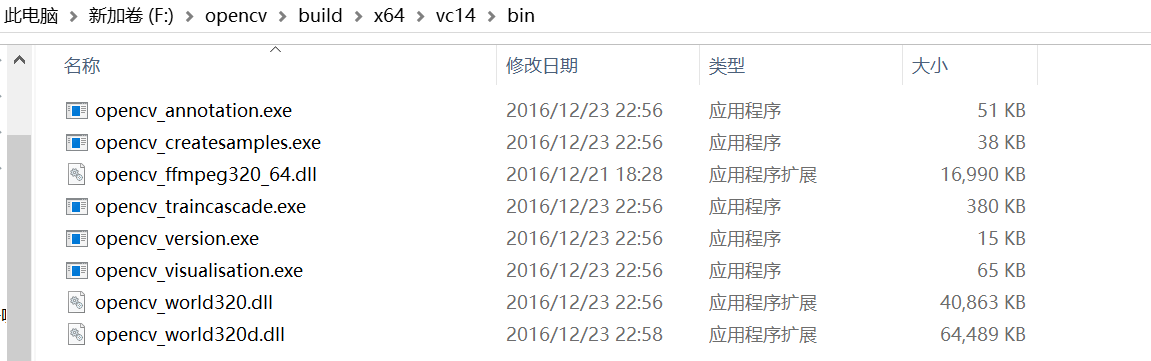


**图2.2 opencv安装完成**

三、为VS2017配置OpenCV3.2.0

1、添加环境变量

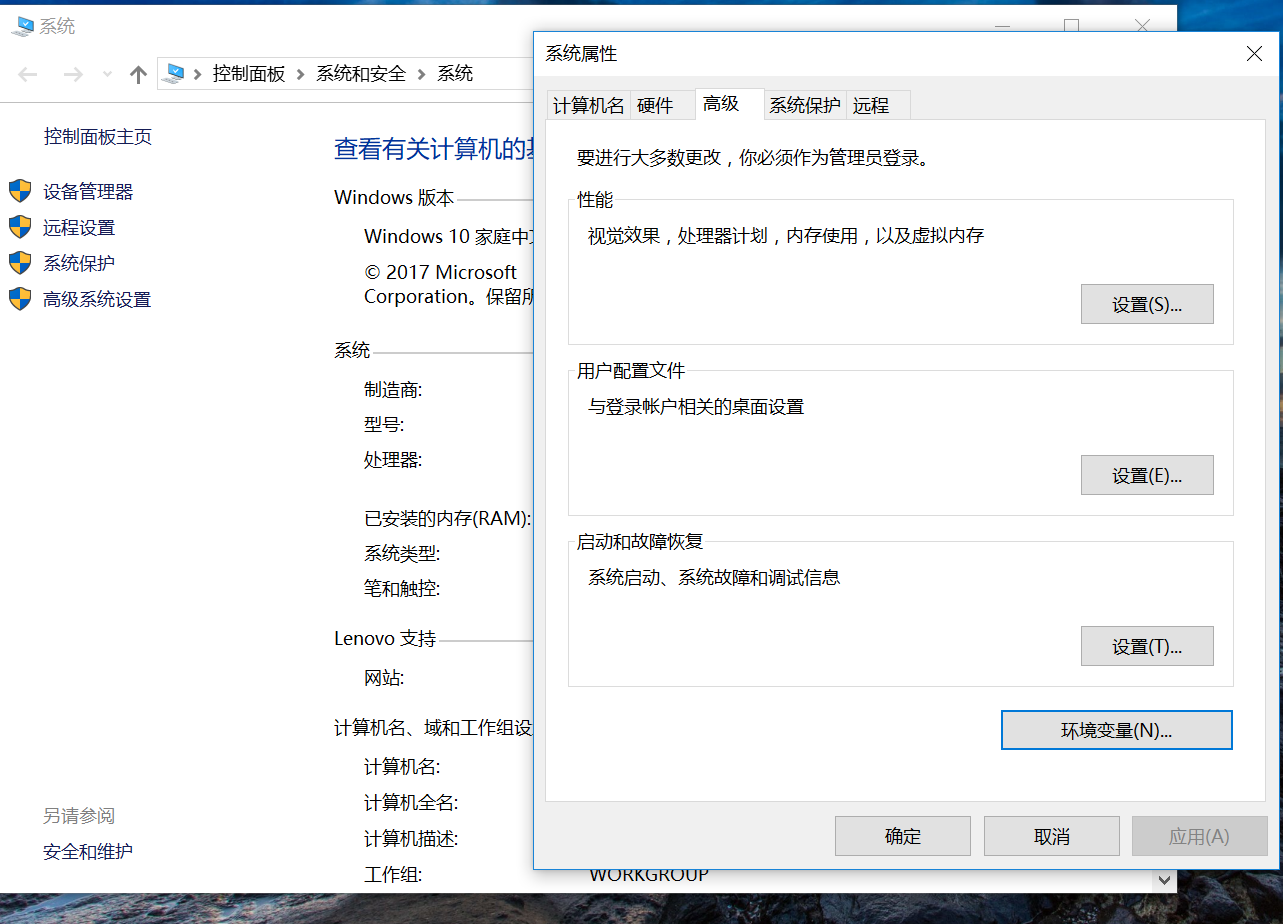
（1）复制opencv-build-x64-vc14-bin的路径F:\opencv\build\x64\vc14\bin（本机的OpenCV的安装位置为F盘），如图3.1所示.



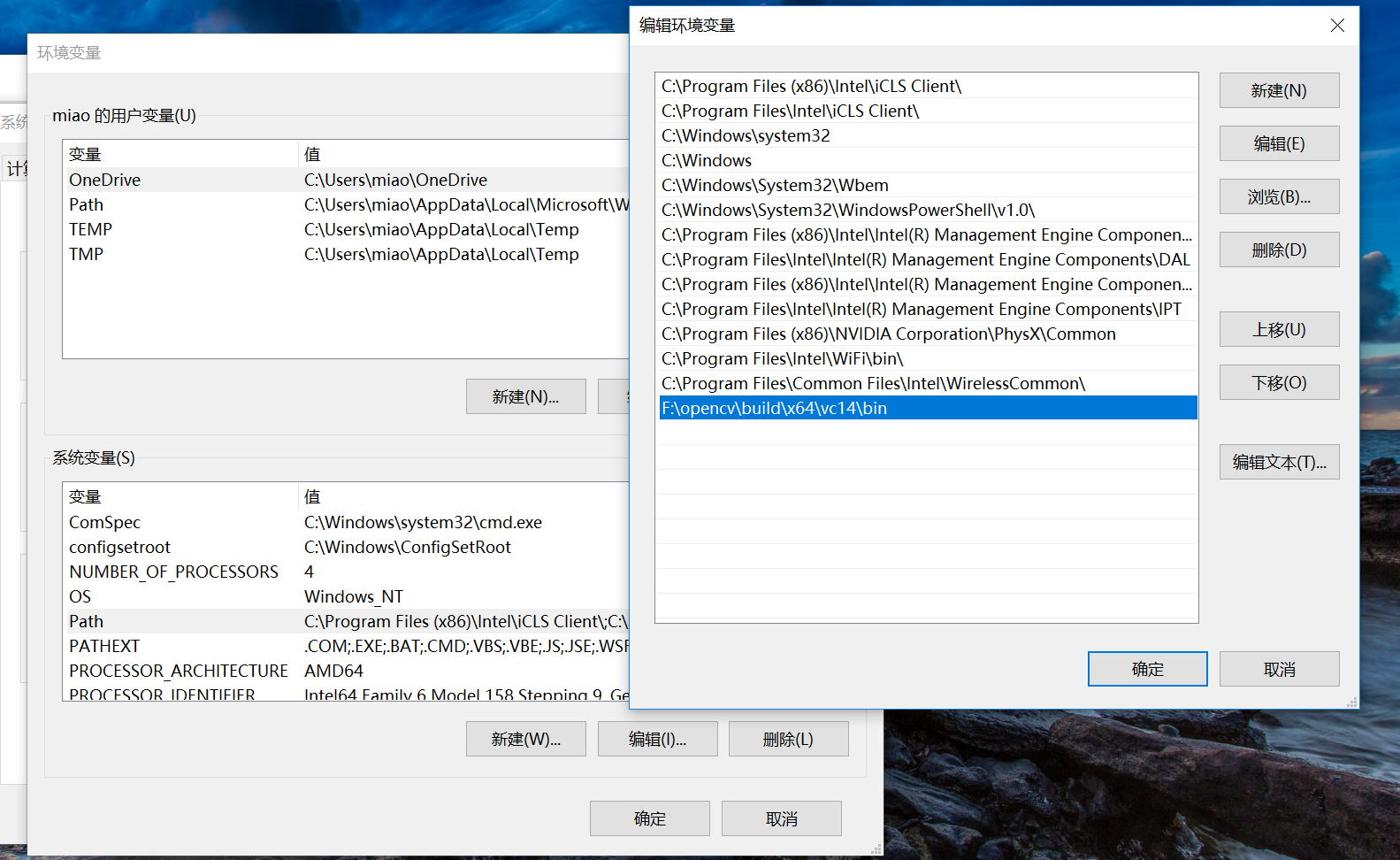
**图3.1复制路径**

（2）增加环境变量到Path

右键桌面上的此电脑，选择“属性”，进入此电脑的系统设置，点击“高级系统设置”，进入“系统属性”页面，如图3.2所示。点击右下角的“环境变量”按钮，选择“系统变量”中的“Path”，编辑环境变量，添加在第1步中复制的路径F:\opencv\build\x64\vc14\bin，如图3.3所示，最后确定。



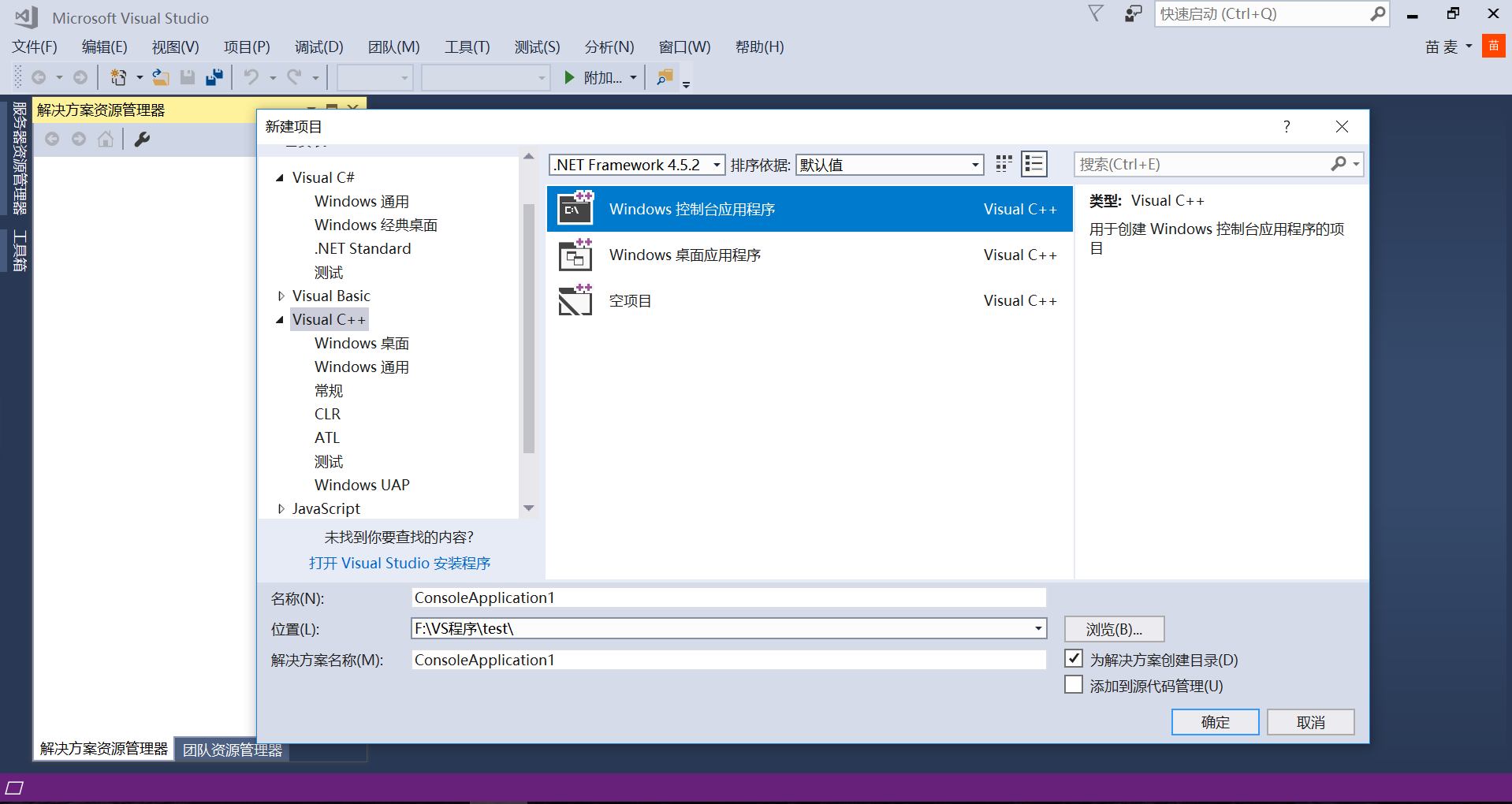
**图3.2 找到计算机的环境变量**



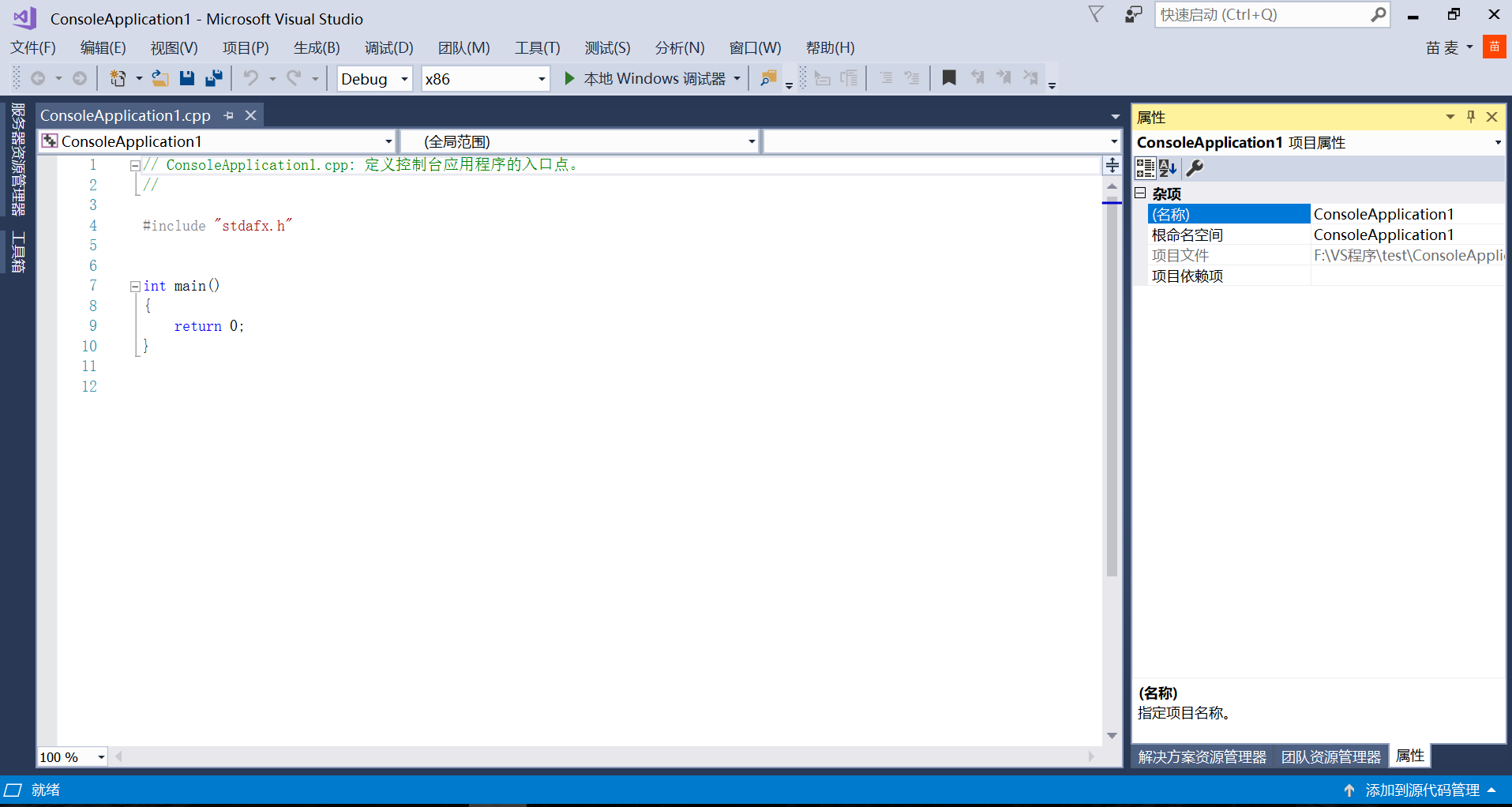
**图3.3 添加环境变量**

2、新建一个VS项目

打开VS2017，选择“文件-新建-项目”，选择Visual C++的Windows控制台应用程序，填写项目名称，选择该项目的保存路径，如图3.4所示，点击确定后，如图3.5所示。



**图3.4 新建一个C++项目**



**图3.5 新建项目的页面**

3、为64位的Debug添加包含目录、库目录和附加依赖项

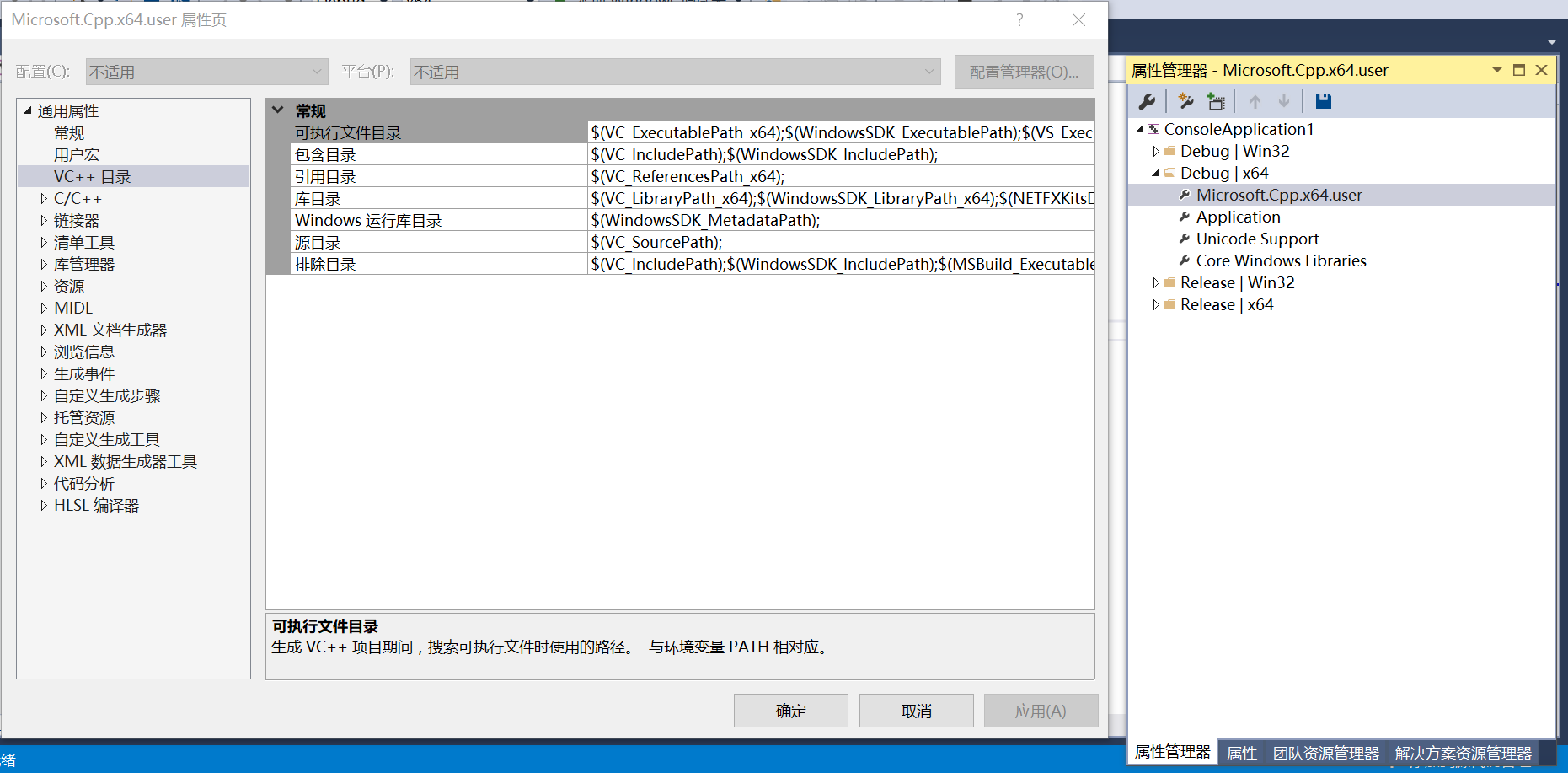
（1）找到项目Debug|64的属性管理器

如图3.6所示，打开项目的属性管理器。



**图3.6打开属性管理器**

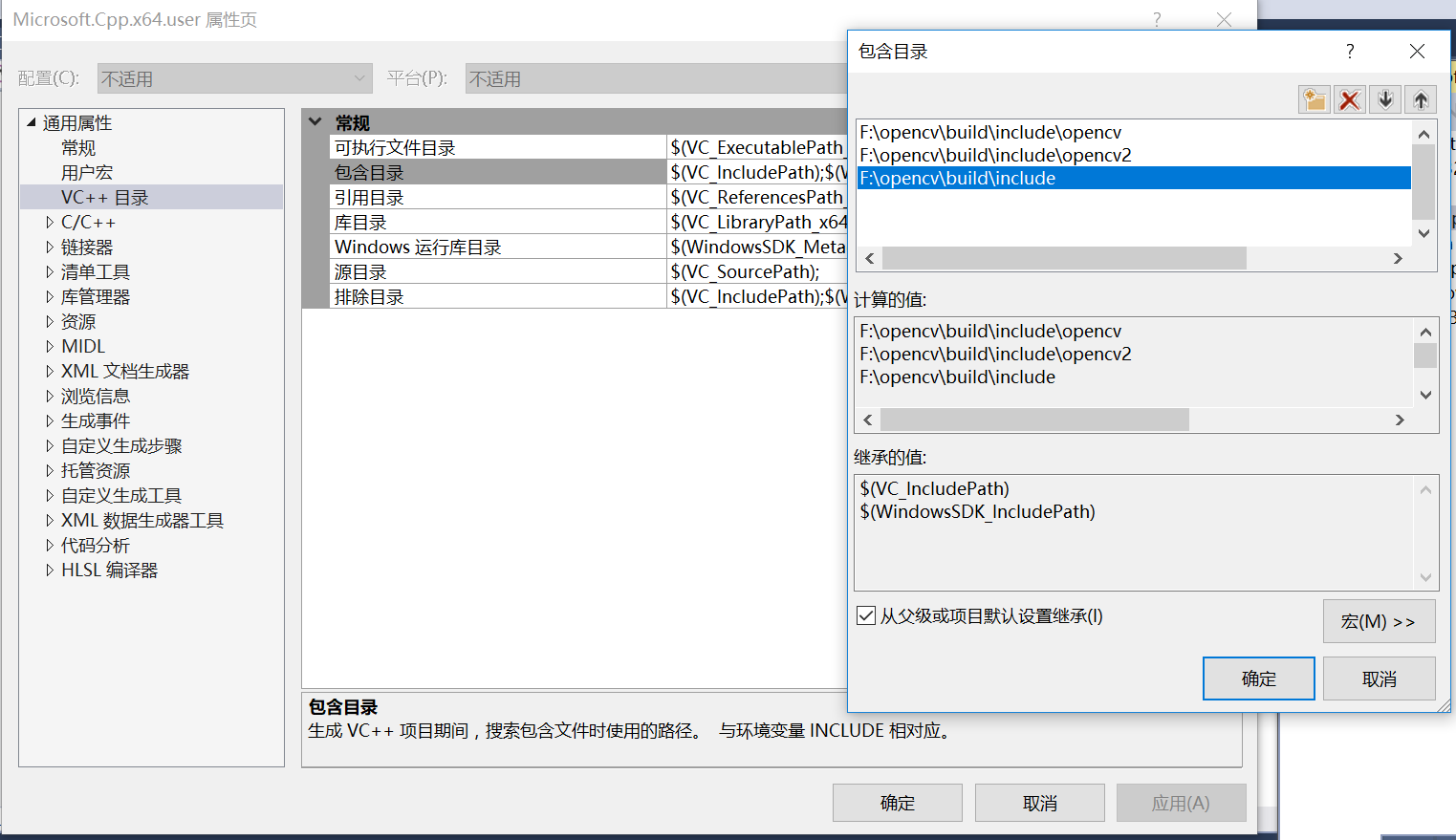
选择**Debug|64，双击Microsoft.Cpp.x64.user，打开项目的属性页，如图3.7所示。**



**图3.7 打开项目的属性页**

（2）添加包含目录

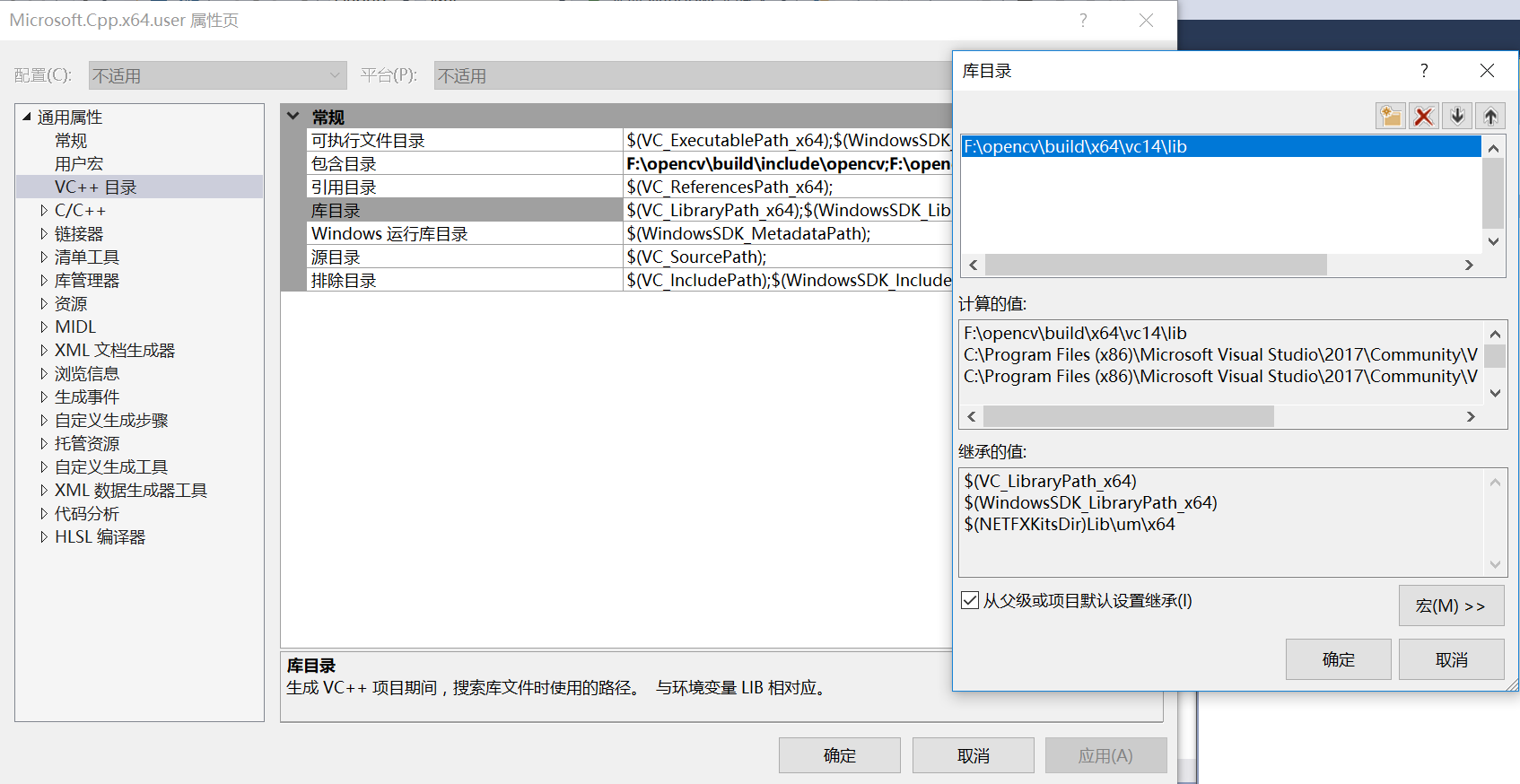
选择“VC++目录-包含目录”，编辑包含目录，增加3条路径，如图3.8所示。



**图3.8 添加包含目录**

（3）添加库目录

选择“VC++目录-库目录”，编辑库目录，增加1条路径，如图3.9所示。



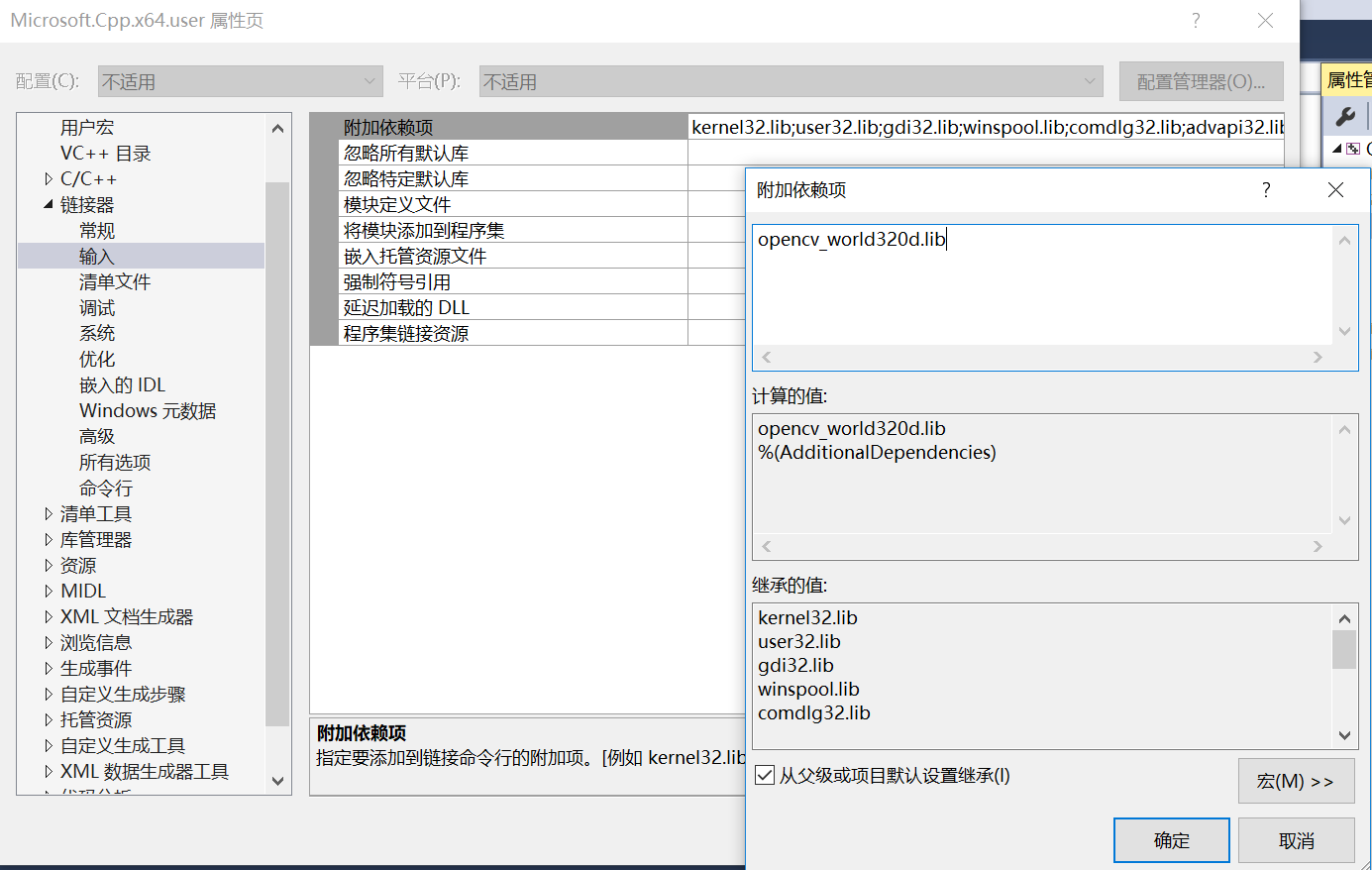
**图3.9 添加库目录**

（4）添加附加依赖项

如图3.10所示，找到OpenCV3库名称，其中，opencv\_world320.lib是Release对应得库文件，opencv\_world320d.lib是Debug对应的库文件。进入属性页的“链接器-输入-附加依赖项”，在编辑框中输入库文件的名字opencv\_world320d.lib，如图3.11所示。



**图3.10库文件名称**



**图3.11添加附加依赖库**

4、为64位的Release添加包含目录、库目录和附加依赖项

Release的配置过程与Debug几乎相同，在打开属性管理器后要选择Release|x64，如图3.12所示，右击属性，进行包含目录、库目录和附加依赖项的配置。其中，前两者的配置过程与Debug完全相同，附加依赖项的配置唯一的区别是添加的库名字是opencv\_world320.lib。



**图3.12 打开Release|x64的属性页**

至此，已经搭建好配置有OpenCV3的VS2017 64位编译环境。下附测试用例（test.jpg可以改变命名，该图像要与.cpp文件放在同一文件夹下）：

#include "stdafx.h"

#include <opencv2/opencv.hpp>

#include <iostream>

#include<opencv2/imgproc/imgproc.hpp>

using namespace cv;

using namespace std;

int main()

{

Mat girl = imread("test.jpg");

namedWindow("the first one");

imshow("the first one", girl);

waitKey(0);

return 0;

}

**补充：**添加opencv扩展库（64位）

1. 将文件夹”opencv\_contrib”扩展库添加至opencv的安装路径下。
2. 修改电脑的环境变量，添加路径如下：

opencv\_contrib\Release\_x64\x64\vc14\bin

opencv\_contrib\Debug\_x64\x64\vc14\bin

1. 修改包含目录。将opencv\opencv\_contrib\Debug\_x64\include\opencv目录下的内容拷贝至opencv\build\include\opencv；
2. 将opencv\opencv\_contrib\Debug\_x64\include\opencv2拷贝至F:\opencv\build\include\opencv2；将opencv\opencv\_contrib\Debug\_x64\x64\vc14\bin目录下的内容拷贝至F:\opencv\build\x64\vc14\bin；将opencv\opencv\_contrib\Debug\_x64\x64\vc14\lib目录下的内容拷贝至F:\opencv\build\x64\vc14\lib。若有同名文件，请跳过。
3. 添加附加依赖项。

对于Debug，添加附加依赖项内容为：

opencv\_aruco320d.lib

opencv\_bgsegm320d.lib

opencv\_bioinspired320d.lib

opencv\_calib3d320d.lib

opencv\_ccalib320d.lib

opencv\_core320d.lib

opencv\_datasets320d.lib

opencv\_dnn320d.lib

opencv\_dpm320d.lib

opencv\_face320d.lib

opencv\_features2d320d.lib

opencv\_flann320d.lib

opencv\_fuzzy320d.lib

opencv\_highgui320d.lib

opencv\_imgcodecs320d.lib

opencv\_imgproc320d.lib

opencv\_line\_descriptor320d.lib

opencv\_ml320d.lib

opencv\_objdetect320d.lib

opencv\_optflow320d.lib

opencv\_phase\_unwrapping320d.lib

opencv\_photo320d.lib

opencv\_plot320d.lib

opencv\_reg320d.lib

opencv\_rgbd320d.lib

opencv\_saliency320d.lib

opencv\_shape320d.lib

opencv\_stereo320d.lib

opencv\_stitching320d.lib

opencv\_structured\_light320d.lib

opencv\_superres320d.lib

opencv\_surface\_matching320d.lib

opencv\_text320d.lib

opencv\_tracking320d.lib

opencv\_video320d.lib

opencv\_videoio320d.lib

opencv\_videostab320d.lib

opencv\_xfeatures2d320d.lib

opencv\_ximgproc320d.lib

opencv\_xobjdetect320d.lib

opencv\_xphoto320d.lib

对于Release，添加的附加依赖项：

opencv\_aruco320.lib

opencv\_bgsegm320.lib

opencv\_bioinspired320.lib

opencv\_calib3d320.lib

opencv\_ccalib320.lib

opencv\_core320.lib

opencv\_datasets320.lib

opencv\_dnn320.lib

opencv\_dpm320.lib

opencv\_face320.lib

opencv\_features2d320.lib

opencv\_flann320.lib

opencv\_fuzzy320.lib

opencv\_highgui320.lib

opencv\_imgcodecs320.lib

opencv\_imgproc320.lib

opencv\_line\_descriptor320.lib

opencv\_ml320.lib

opencv\_objdetect320.lib

opencv\_optflow320.lib

opencv\_phase\_unwrapping320.lib

opencv\_photo320.lib

opencv\_plot320.lib

opencv\_reg320.lib

opencv\_rgbd320.lib

opencv\_saliency320.lib

opencv\_shape320.lib

opencv\_stereo320.lib

opencv\_stitching320.lib

opencv\_structured\_light320.lib

opencv\_superres320.lib

opencv\_surface\_matching320.lib

opencv\_text320.lib

opencv\_tracking320.lib

opencv\_video320.lib

opencv\_videoio320.lib

opencv\_videostab320.lib

opencv\_xfeatures2d320.lib

opencv\_ximgproc320.lib

opencv\_xobjdetect320.lib

opencv\_xphoto320.lib