

Windows10 下 OpenCV 环境搭建

(Visual Studio 2017 + OpenCV3.4.1)

安装 OpenCV3.4.1

下载地址: https://sourceforge.net/projects/opencvlibrary/files/opencv-win/3.4.1/opencv-3.4.1-vc14_vc15.exe/download

1. 新建 D:\opencv 文件夹。
2. 用 Winrar 打开安装文件 opencv-3.4.1-vc14_vc15.exe, 导航到其中 opencv/build/ 目录, 解压 bin、include 和 x64 文件夹到上一步新建的文件夹里。完成后 D:\opencv 下有 bin、include 和 x64 三个目录。

bin	2018/5/10 13:36	文件夹
include	2018/5/10 13:36	文件夹
x64	2018/5/10 14:41	文件夹

3. 打开 x64 并删除 vc14 文件夹。
4. 以管理员身份运行命令提示符并执行以下两条命令。

```
set OPENCV_DIR =" D:\opencv\x64\vc15"  
set Path="%Path%;%OPENCV_DIR%\bin"
```

安装 Visual Studio 2017

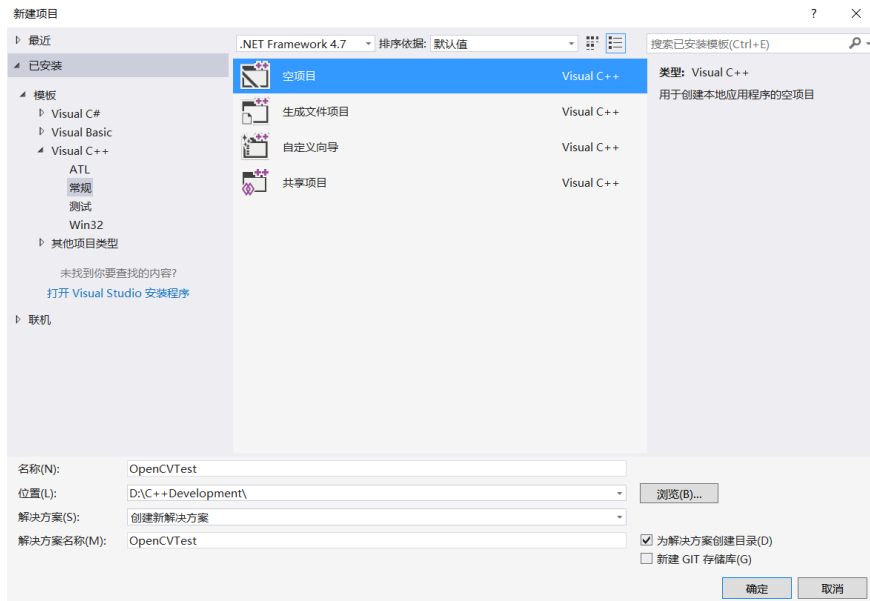
下载地址: <https://www.visualstudio.com/zh-hans/downloads/>

运行安装程序, 选择需要的 VS 版本; 在工作负载里选择使用 C++ 的桌面开发, 页面下方选择安装位置。然后执行安装。

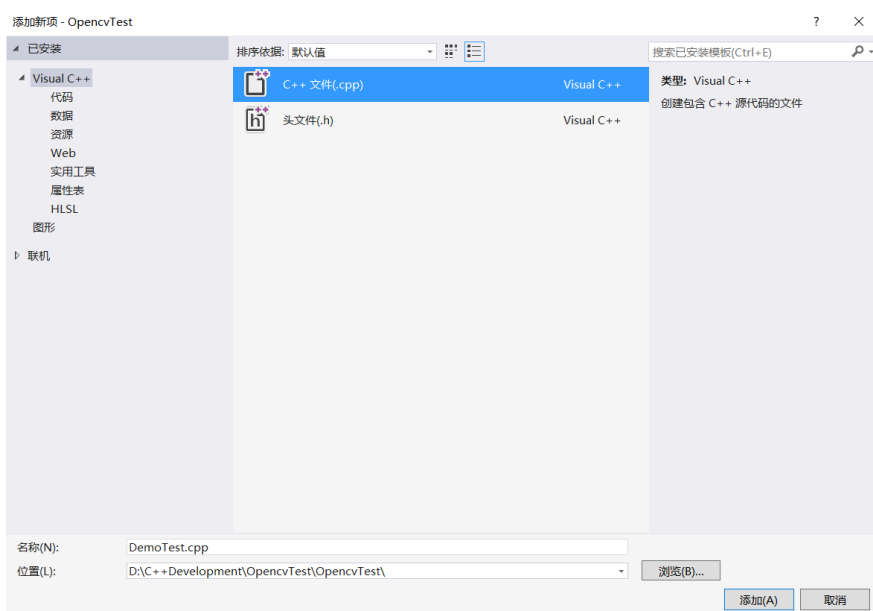
配置 Visual Studio 2017 项目

1. 打开 VS2017, 选择 C++ 作为常用工作负载和任意颜色主题, 如果提示登陆可以选择跳过登陆。
2. Ctrl+Shift+N 新建项目, 左侧导航栏里展开 Visual C++ 选中常规, 中间列表里选空项目。名称改为 OpencvTest, 位置填入工程目录, 选中为解决方案创建目录。

点击确定。



3. Ctrl+Shift+A 添加新建项，选中 C++ 文件，名称改为 DemoTest.cpp，点击添加。

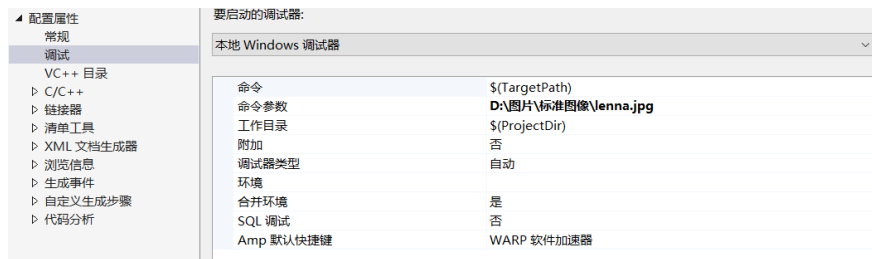


4. 解决方案资源管理器里选中项目，Alt+Enter 打开属性，在属性页配置选择 Debug 或者活动(Debug)，平台选择 x64 或者活动(x64)。

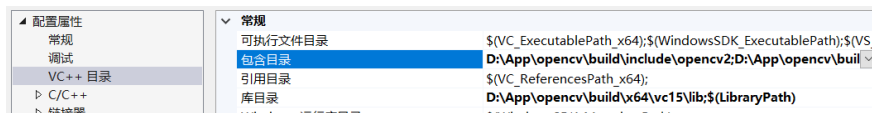


5. 左侧导航栏选调试，要启动的调试器选本地 Windows 调试器，命令参数里输入测

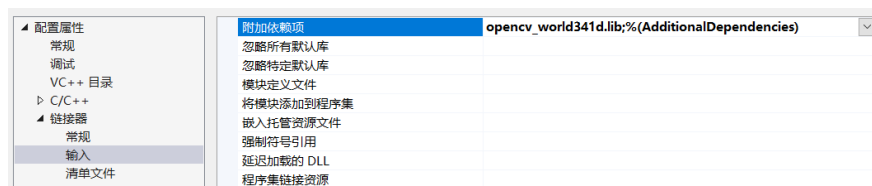
试用图片的路径。



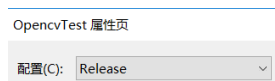
6. 左侧导航栏选 **VC++ 目录**，右侧**常规**下**包含目录**改为 `D:\opencv\include\opencv2;D:\opencv\include\opencv;D:\opencv\include;$(IncludePath)`，**库目录**改为 `D:\opencv\x64\vc15\lib;$(LibraryPath)`。



7. 左侧导航栏展开**链接器**选**输入**，右侧**附加依赖项**改为 `opencv_world341d.lib;%(AdditionalDependencies)`并点击窗口右下角**应用**。



8. 窗口上部的**配置**改为 **Release** 或**活动(Relase)**，重复 5、6 两个步骤。



9. 将第 7 步图所示的**附加依赖项**改为 `opencv_world341d.lib;%(AdditionalDependencies)` 并点击窗口右下角**确定**。

10. **解决方案资源管理器**里双击打开 **DemoTest.cpp** 文件。



11. 输入下面的代码。

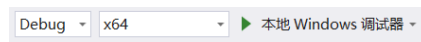
```
#include <opencv2/opencv.hpp>
using namespace cv;
int main(int argc, char** argv)
```

```

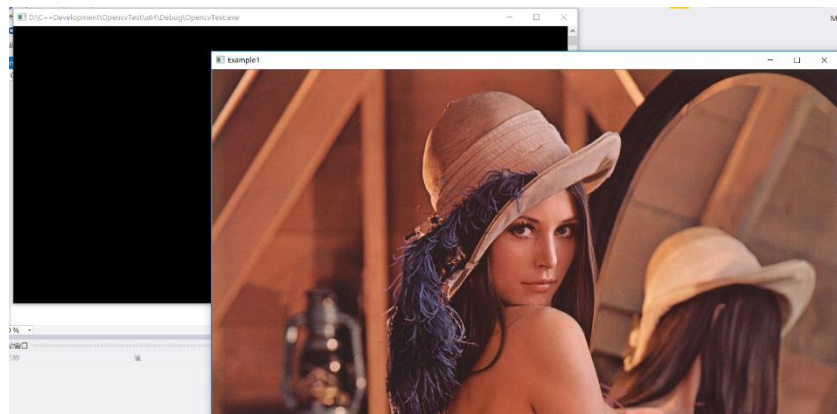
{
    Mat img = imread(argv[1], -1);
    if (img.empty()) return -1;
    namedWindow("Example1", WINDOW_AUTOSIZE);
    imshow("Example1", img);
    waitKey();
    destroyWindow("Example1");
    return 0;
}

```

12. 工具栏上选择 x64 平台并点击本地 Windows 调试器，如果弹出重新编译的对话框点击确定。



13. 自动编译运行后效果如图，成功显示第 5 步设置的图片。

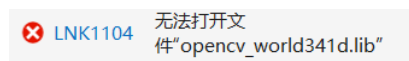


常见问题

1. 找不到 opencv_world341.dll 或者 opencv_world341d.dll。
 解决方案：检查安装 OpenCV3.4.1 第 1 步里环境变量是否设置正确，确认正确后重启 Visual Studio。Visual Studio 调试环境将在启动时重新加载环境变量。



2. 链接时无法打开文件 opencv_world341.lib 或者 opencv_world341d.lib。
 解决方案：检查配置 VS2017 项目的第 6 步，确保在当前活动配置（Debug 或这 Release）下正确设置了库目录。



3. 出现大量错误。

解决方案：a. 检查所有路径设置；b. 生成平台选择 x64，因为 OpenCV 安装包里的默认编译为 64 位版本。



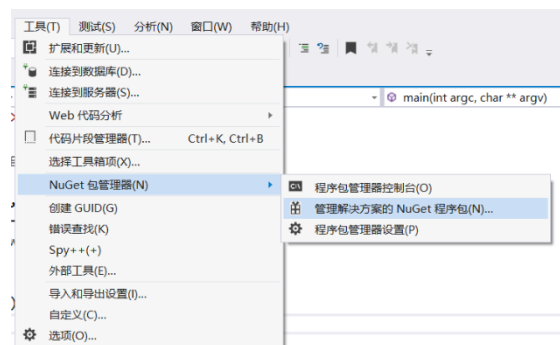
4. 成功编译但运行一闪而过。

解决方案：检查配置 VS2017 项目的第 5 步，正确设置测试图片的路径。

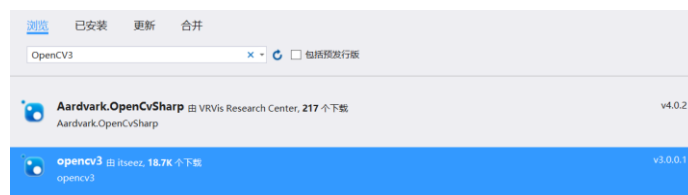
附录：用 Nuget 安装和配置 OpenCV

VS2017 集成了强大高效的第三方开源库管理组件 Nuget，可以用它快速安装和配置 OpenCV。按照配置 Visual Studio 项目 1~3 步新建项目，然后完成以下步骤。

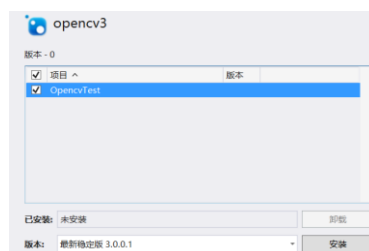
1. 工具->NuGet 包管理器->管理解决方案的 NuGet 程序包....。



2. 在 NuGet 管理界面选择浏览，输入 opencv3 进行搜索，点击选择需要的版本。



3. 右侧面板里勾选当前项目然后点击安装。



4. 开始使用吧。