1，dlib

编译dlib

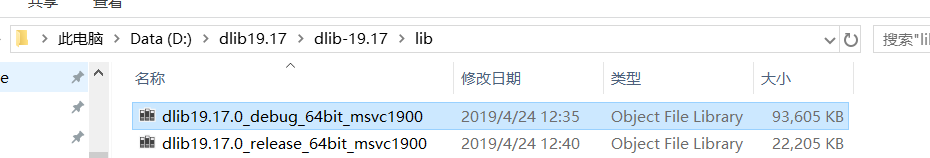
1，新建build

2，cmake .. -G"Visual Studio 14 2015 Win64" -T host=x64（在build文件夹生成vs工程）

3，cmake --build . --config Debug（编译,同时可编译release版本）

cmake --build . --config Release

4，新建一个lib文件夹，将build好的dubug与release的lib文件分别拷贝过来



在build/x64可以看到两个版本

其中模型文件在<http://dlib.net/files/>下载

2.创建工程

打开vs2015

C++ MFC

基于对话框

3.vs工程文件夹操作

选debug x64

在资源文件/XXX.rc2/Dialog/DIALOG文件

工具箱 pin下

先拉一个picture control 其中在杂项 ID 最重要

4.配置cuda

头文件配置：在工程属性里面vc++ 包含cuda的include目录

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.0\include

C++预处理器：DLIB\_USE\_CUDA DLIB\_HAVE\_SSE2

库文件配置：

连接器---常规—附加库目录

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.0\lib\x64

连接器---输入---附加依赖项 cuda.lib

5.opencv配置

头文件配置： c++---常规-----包含opencv的include目录

库文件配置：

连接器---常规—附加库目录

D:\opencv3.2\build\x64\vc14\lib（vc14就是vs2015）

连接器---输入---附加依赖项 opencv\_world341d.lib

6.dlib配置

头文件配置： c++---常规----- D:\dlib19.17\dlib-19.17

C++预处理器：

DLIB\_JPEG\_SUPPORT

DLIB\_PNG\_SUPPORT

库文件配置：

连接器---常规—附加库目录

D:\dlib19.17\dlib-19.17\lib

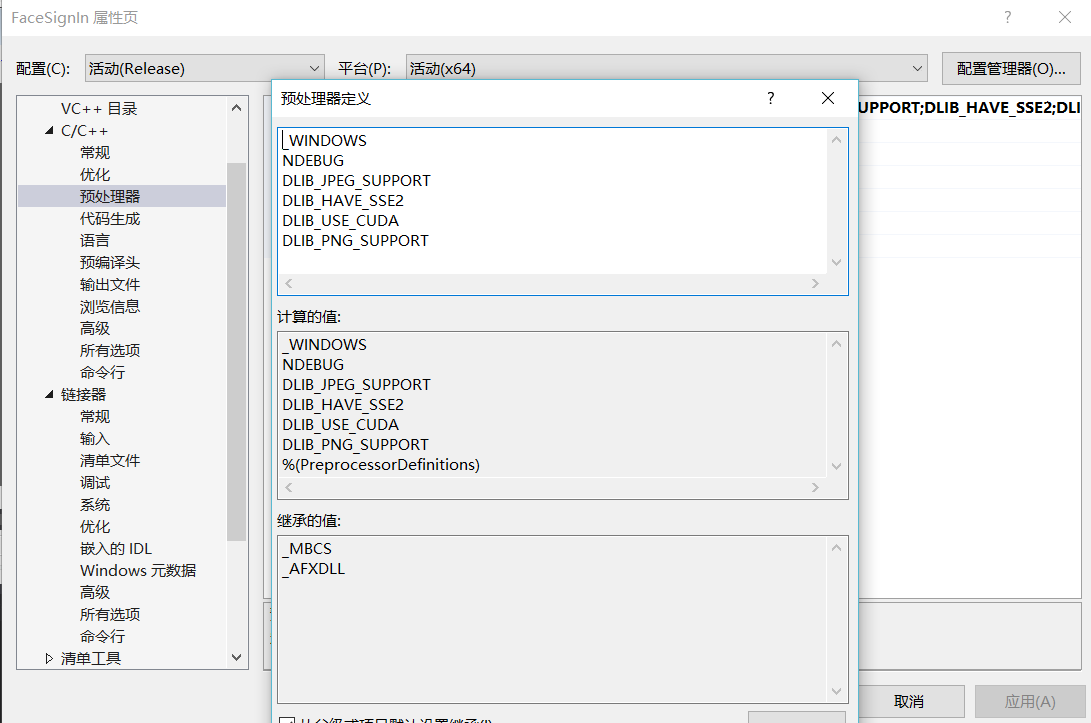
连接器---输入---附加依赖项

dlib19.17.0\_debug\_64bit\_msvc1900

dlib19.17.0\_release\_64bit\_msvc1900

这里面注意，属性配置的时候注意是debug还是release

全部：



\_WINDOWS

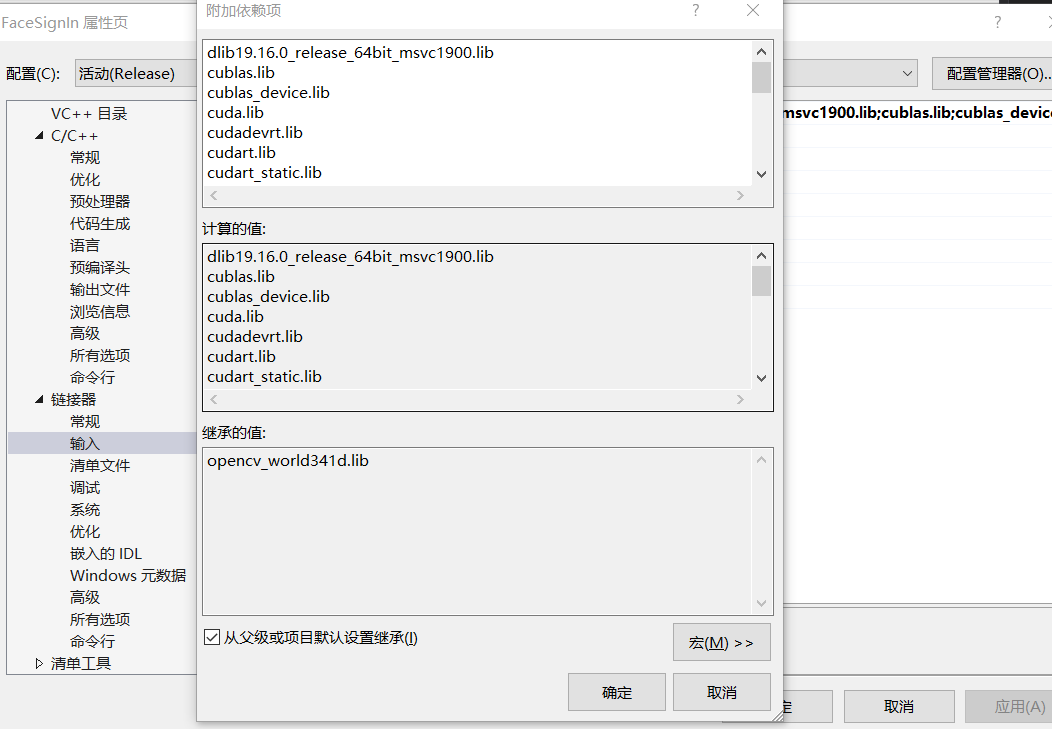
NDEBUG

DLIB\_JPEG\_SUPPORT

DLIB\_HAVE\_SSE2

DLIB\_USE\_CUDA

DLIB\_PNG\_SUPPORT



dlib19.16.0\_release\_64bit\_msvc1900.lib

cublas.lib

cublas\_device.lib

cuda.lib

cudadevrt.lib

cudart.lib

cudart\_static.lib

cudnn.lib

cufft.lib

cufftw.lib

curand.lib

cusolver.lib

cusparse.lib

nppc.lib

nppial.lib

nppicc.lib

nppicom.lib

nppidei.lib

nppif.lib

nppig.lib

nppim.lib

nppist.lib

opencv\_world341.lib