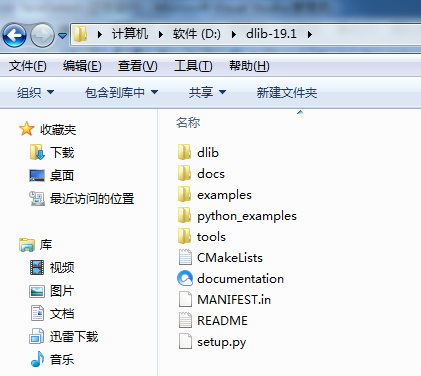
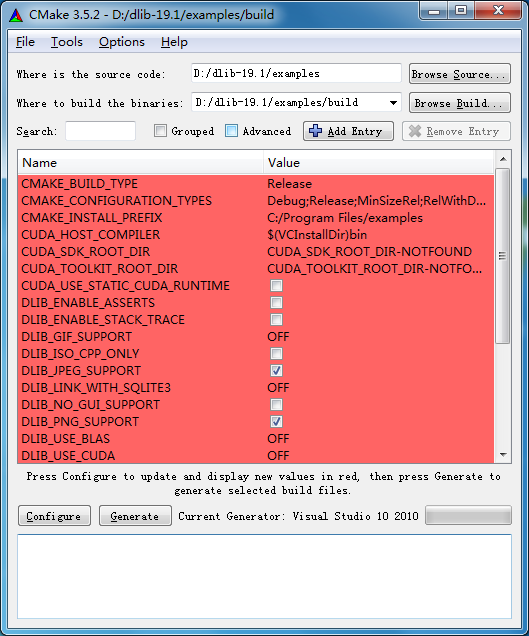
# DLIB的编译与使用

1. 下载dlib-19.1压缩包，解压到D盘，解压后目录如下：

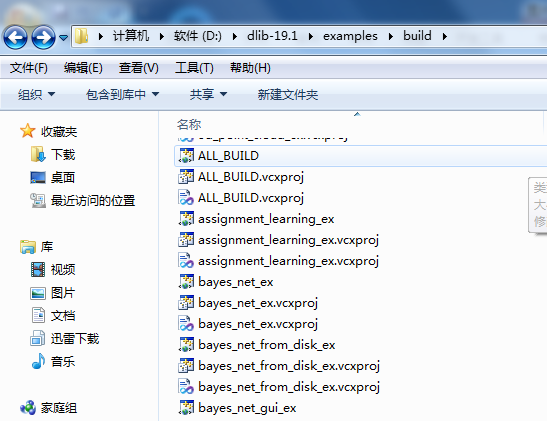


1. 使用Cmake进行编译

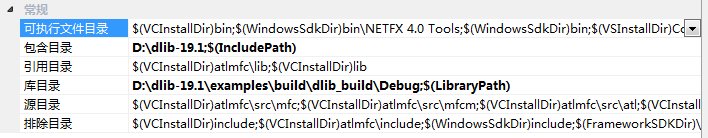
Where is the source code 中填写examples目录，where to build the bianries 填写 生成目录。Configure一下，如果显示OpenCV目录找不到，手动填写OpenCV的build目录，再Configure一下，然后Generate。



1. 编译后生成许多工程文件，随便使用Visual Studio打开一个工程文件，生成解决方案。



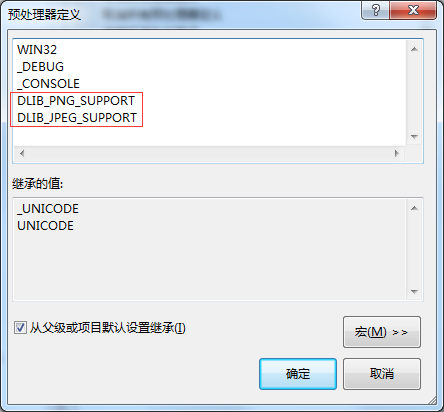
1. 新建一个工程，设置包含目录和库目录



添加附加依赖项：



在预处理器中添加DLIB\_PNG\_SUPPORT和DLIB\_JPEG\_SUPPORT：



1. 在工程中添加如下代码：

#include "stdafx.h"

#include <dlib/image\_processing/frontal\_face\_detector.h>

#include <dlib/gui\_widgets.h>

#include <dlib/image\_io.h>

#include <iostream>

using namespace dlib;

using namespace std;

int main(int argc, char\*\* argv)

{

frontal\_face\_detector detector = get\_frontal\_face\_detector();

image\_window win;

array2d<unsigned char> img;

load\_image(img, "E:\\TestImage\\family.jpg");

std::vector<rectangle> dets = detector(img);

cout << "Number of faces detected: " << dets.size() << endl;

win.clear\_overlay();

win.set\_image(img);

win.add\_overlay(dets, rgb\_pixel(255,0,0));

cin.get();

}

运行结果：

