



哈爾濱工業大學(深圳)
HARBIN INSTITUTE OF TECHNOLOGY, SHENZHEN

机器视觉课程设计-课题1

答辩人：方尧 罗赫铭

课程导师：吴晓军

专 业：自动化

班 级：19本自动化1班

2022年06月17日

目录

CONTENTS

01

课题要求

02

NCC模板匹配

03

四层->八层

04

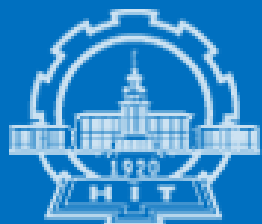
四层<->八层
结果对比

05

总结



哈爾濱工業大學(深圳)
HARBIN INSTITUTE OF TECHNOLOGY, SHENZHEN



哈爾濱工業大學(深圳)
Harbin Institute of Technology, Shenzhen

Part.01

课题要求



已经有一个 4 层金字塔的 NCC 模板匹配算法

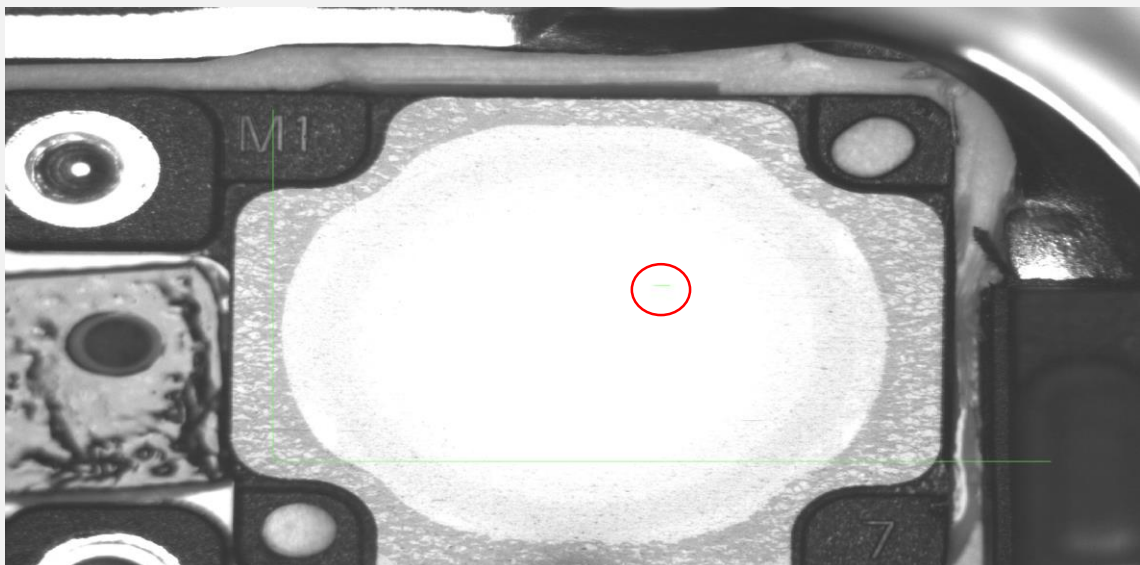
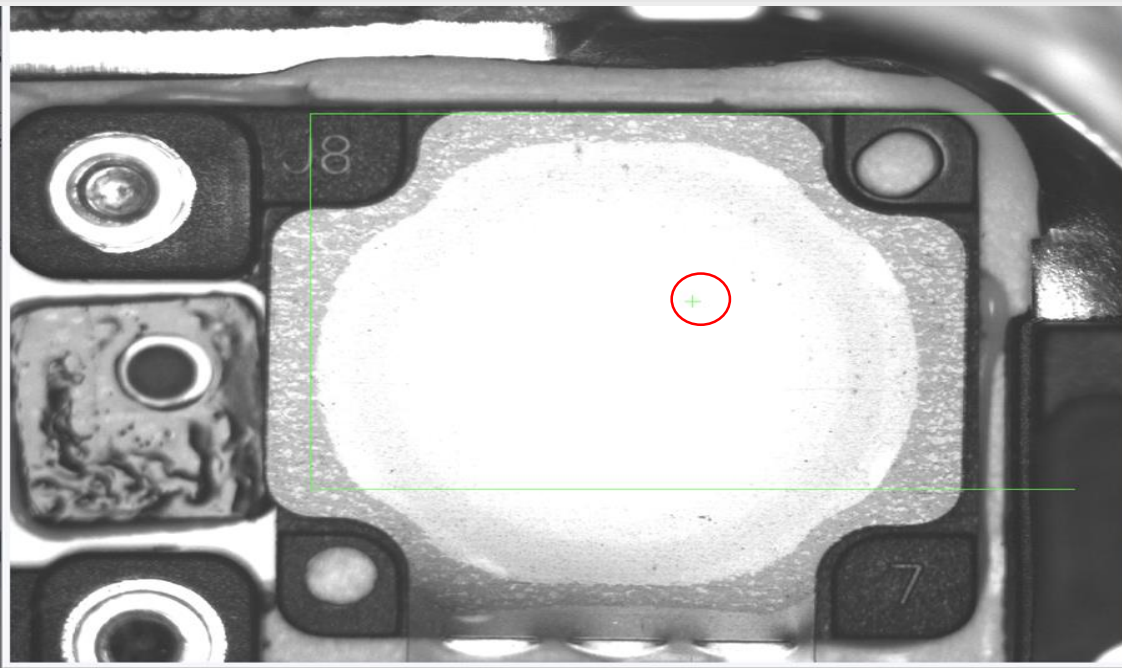
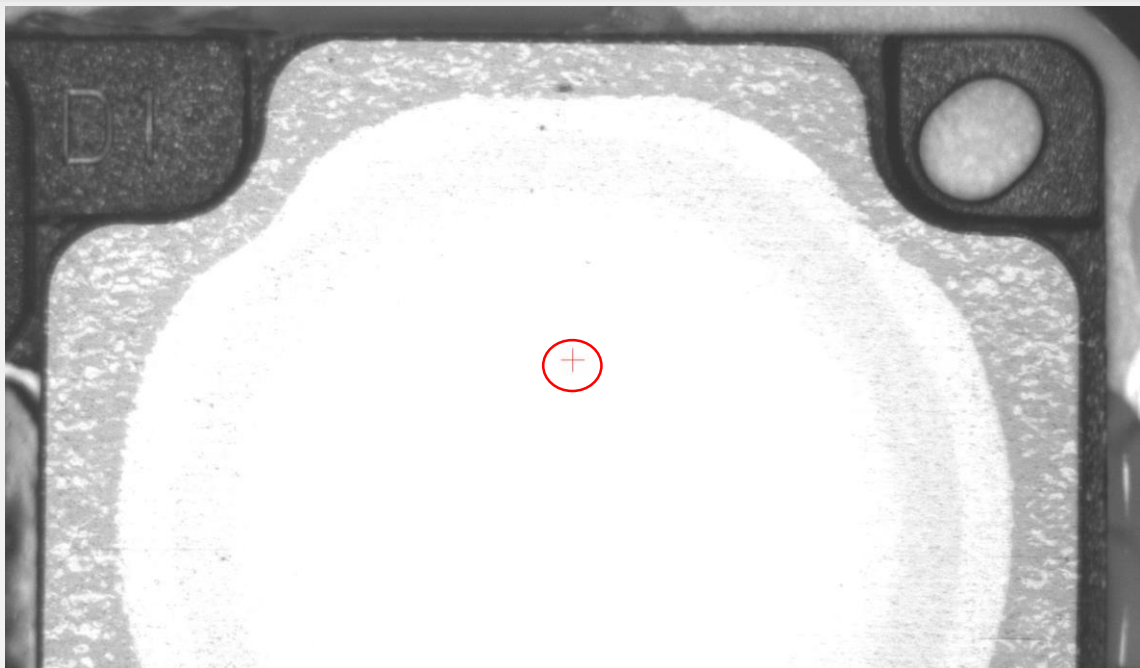
提供 C++VS2015 或以下 VS 工程Opencv1.0 的源码。

课题要求：将算法实现到 8 层金字塔的 NCC 模板匹配

目的：加速



01 课题要求——四层NCC效果



Template Matching

系统初始化

打开图像

打开模板

控制部分

设置参数

图像匹配

旋转部分

设置最大旋转角度: 0

设置旋转角度: 0

旋转匹配

退出系统

输出

匹配时间: 3495.0887 ms 最大匹配值: 10.249 是否报警: 0

File

2022年6月16日15时13分19秒 相机返回图像

2022年6月16日15时13分22秒 打开模板

2022年6月16日15时13分35秒 模板匹配完毕, 匹配中心位置是: x=1585.858688, y=877.817655

2022年6月16日15时16分20秒 打开模板

2022年6月16日15时17分1秒 模板匹配完毕, 匹配中心位置是: x=1585.858688, y=877.817655

2022年6月16日15时18分46秒 相机返回图像

2022年6月16日15时18分50秒 模板匹配完毕, 匹配中心位置是: x=1497.407155, y=875.168779



哈爾濱工業大學(深圳)
Harbin Institute of Technology, Shenzhen

Part.02

NCC模板匹配



$$NCC(r, c) = \frac{1}{n} \sum_{(u,v) \in T} \frac{t_{(u,v)} f(r+u, c+v) - m_f(r, c)}{\sqrt{s_t^2} \sqrt{s_f^2(r, c)}}$$

t-模板像素， f-目标图像

-1 <= NCC <= 1, NCC值越接近1， 表示匹配程度越高

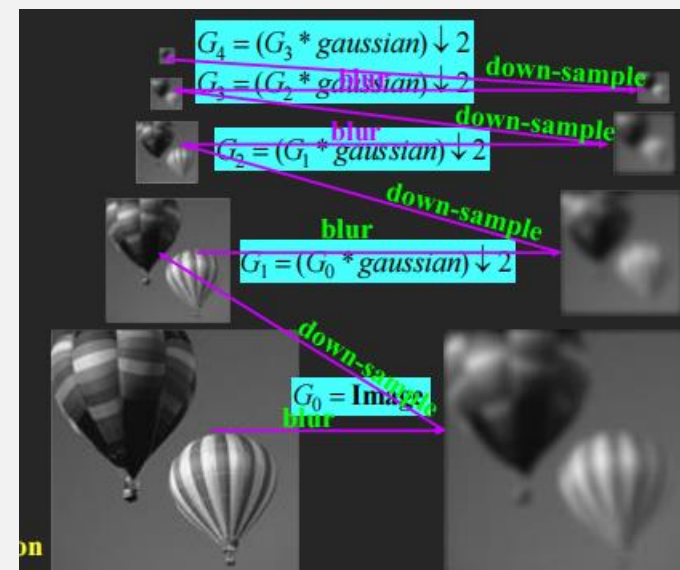
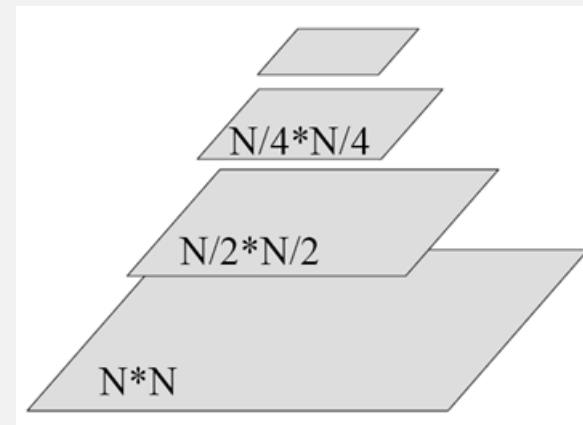
NCC点数取决于像素大小， 即模板大小和目标图像大小， R*C*U*V

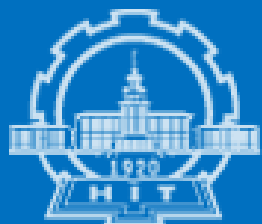
速度太慢！



解决方法：采用多层图像金字塔

1. 获取目标图像和模板图像，将目标图像和模板图像划分图像金字塔。
2. 对各层图像金字塔获取模板像素并计算灰度之和、均值和协方差。
3. 根据模板大小，以模板左上角为基准点，在目标图像上从左至右、从上至下移动模板，计算每次移动后对应模板区域的NCC值。
4. 遍历完成后通过排序和阈值处理得到目标位置。
5. 将结果映射到下一级金字塔重复步骤。





哈爾濱工業大學(深圳)
Harbin Institute of Technology, Shenzhen

Part.03

金字塔： 四层 -> 八层



更改1：模板/图像 扩增

```
pMatImage2=NULL;
pMatImage4=NULL;
pMatImage8=NULL;
pMatImage16=NULL;
pMatImage32 = NULL;
pMatImage64 = NULL;
pMatImage128 = NULL;
pMatImage256 = NULL;
pMatTemplate=NULL;
pMatTemplate2=NULL;
pMatTemplate4=NULL;
pMatTemplate8=NULL;
pMatTemplate16=NULL;
pMatTemplate32 = NULL;
pMatTemplate64 = NULL;
pMatTemplate128 = NULL;
pMatTemplate256 = NULL;
pMatFlag4=NULL;
pMatFlag8=NULL;

m_pTemplate=NULL;
//for(int i=0;i<9;i++)
//  m_fNccArray[i]=0.0;
```

更改2：内存分配和释放

```
CDBImageProcess::~~CDBImageProcess()
{
    //Release matrix memory
    cvReleaseImage(&m_pTemplate);
    cvReleaseMat(&pMatImage2);
    cvReleaseMat(&pMatImage4);
    cvReleaseMat(&pMatImage8);
    cvReleaseMat(&pMatImage16);
    cvReleaseMat(&pMatImage32);
    cvReleaseMat(&pMatImage64);
    cvReleaseMat(&pMatImage128);
    cvReleaseMat(&pMatImage256);
    cvReleaseMat(&pMatTemplate);
    cvReleaseMat(&pMatTemplate2);
    cvReleaseMat(&pMatTemplate4);
    cvReleaseMat(&pMatTemplate8);
    cvReleaseMat(&pMatTemplate16);
    cvReleaseMat(&pMatTemplate32);
    cvReleaseMat(&pMatTemplate64);
    cvReleaseMat(&pMatTemplate128);
    cvReleaseMat(&pMatTemplate256);
    cvReleaseMat(&pMatFlag4);
    cvReleaseMat(&pMatFlag8);
}
```



更改3：图像金字塔预处理

```
PyramidDown(pMatTemplate, pMatTemplate2, NULL); //1/2
PyramidDown(pMatTemplate2, pMatTemplate4, NULL); //1/4
PyramidDown(pMatTemplate4, pMatTemplate8, NULL); //1/8
PyramidDown(pMatTemplate8, pMatTemplate16, NULL); //1/16
PyramidDown(pMatTemplate16, pMatTemplate32, NULL); //1/16
PyramidDown(pMatTemplate32, pMatTemplate64, NULL); //1/16
PyramidDown(pMatTemplate64, pMatTemplate128, NULL); //1/16
PyramidDown(pMatTemplate128, pMatTemplate256, NULL); //1/16
```

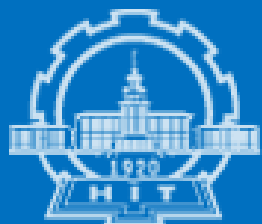
```
PyramidDown(pMatImage2, pMatImage4, NULL);
PyramidDown(pMatImage4, pMatImage8, NULL);
PyramidDown(pMatImage8, pMatImage16, NULL);
PyramidDown(pMatImage16, pMatImage32, NULL);
PyramidDown(pMatImage32, pMatImage64, NULL);
PyramidDown(pMatImage64, pMatImage128, NULL);
//PyramidDown(pMatImage128, pMatImage256, NULL);
cout << "1/2原图像大小: " << pMatImage2->width << " " << pMatImage2->height << endl;
cout << "1/4原图像大小: " << pMatImage4->width << " " << pMatImage4->height << endl;
cout << "1/8原图像大小: " << pMatImage8->width << " " << pMatImage8->height << endl;
cout << "1/16原图像大小: " << pMatImage16->width << " " << pMatImage16->height << endl;
cout << "1/32原图像大小: " << pMatImage32->width << " " << pMatImage32->height << endl;
cout << "1/64原图像大小: " << pMatImage64->width << " " << pMatImage64->height << endl;
cout << "1/128原图像大小: " << pMatImage128->width << " " << pMatImage128->height << endl;
//cout << "1/256原图像: " << pMatImage256->width << " " << pMatImage256->height << endl;
timer.EndTime();
cout << "预处理用时: " << timer.GetTime() << "ms" << endl;
```



更改4：图像金字塔处理

更改5：调试输出

```
PyramidDown(pMatImage01, pMatImage128, NULL);  
//PyramidDown(pMatImage128, pMatImage256, NULL);  
cout << "1/2原图像大小: " << pMatImage2->width << " " << pMatImage2->height << endl;  
cout << "1/4原图像大小: " << pMatImage4->width << " " << pMatImage4->height << endl;  
cout << "1/8原图像大小: " << pMatImage8->width << " " << pMatImage8->height << endl;  
cout << "1/16原图像大小: " << pMatImage16->width << " " << pMatImage16->height << endl;  
cout << "1/32原图像大小: " << pMatImage32->width << " " << pMatImage32->height << endl;  
cout << "1/64原图像大小: " << pMatImage64->width << " " << pMatImage64->height << endl;  
cout << "1/128原图像大小: " << pMatImage128->width << " " << pMatImage128->height << endl;  
//cout << "1/256原图像: " << pMatImage256->width << " " << pMatImage256->height << endl;  
timer.EndTime();  
cout<<"预处理用时: "<<timer.GetTime()<<"ms"<<endl;
```



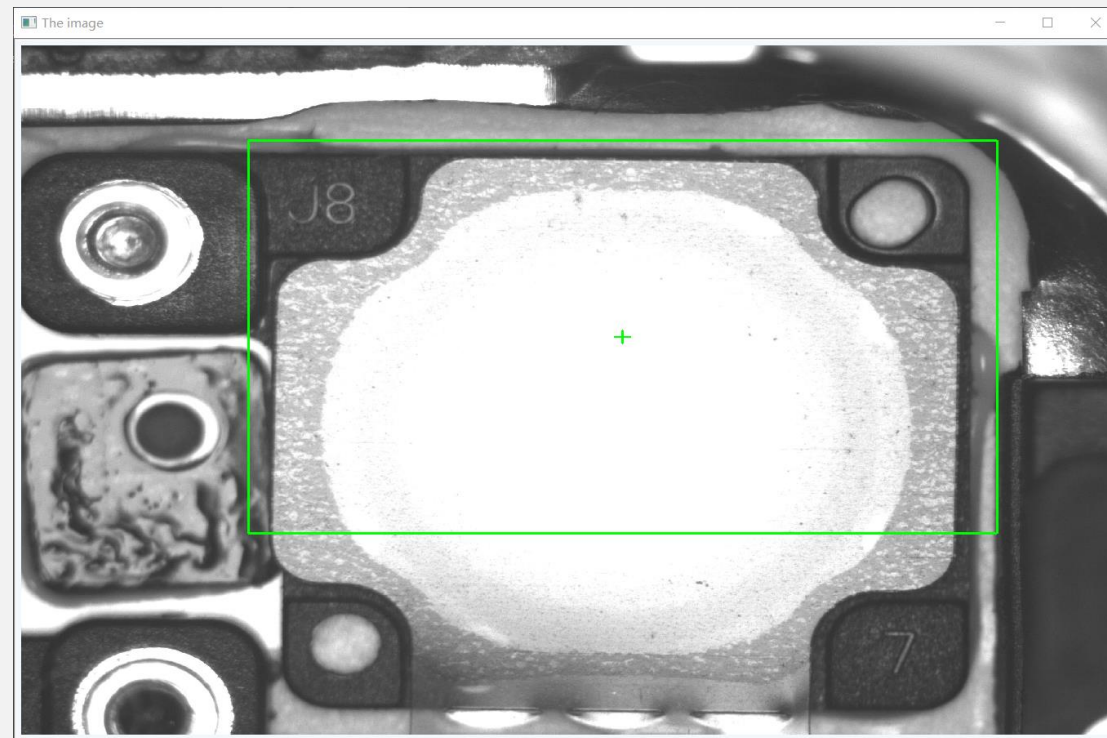
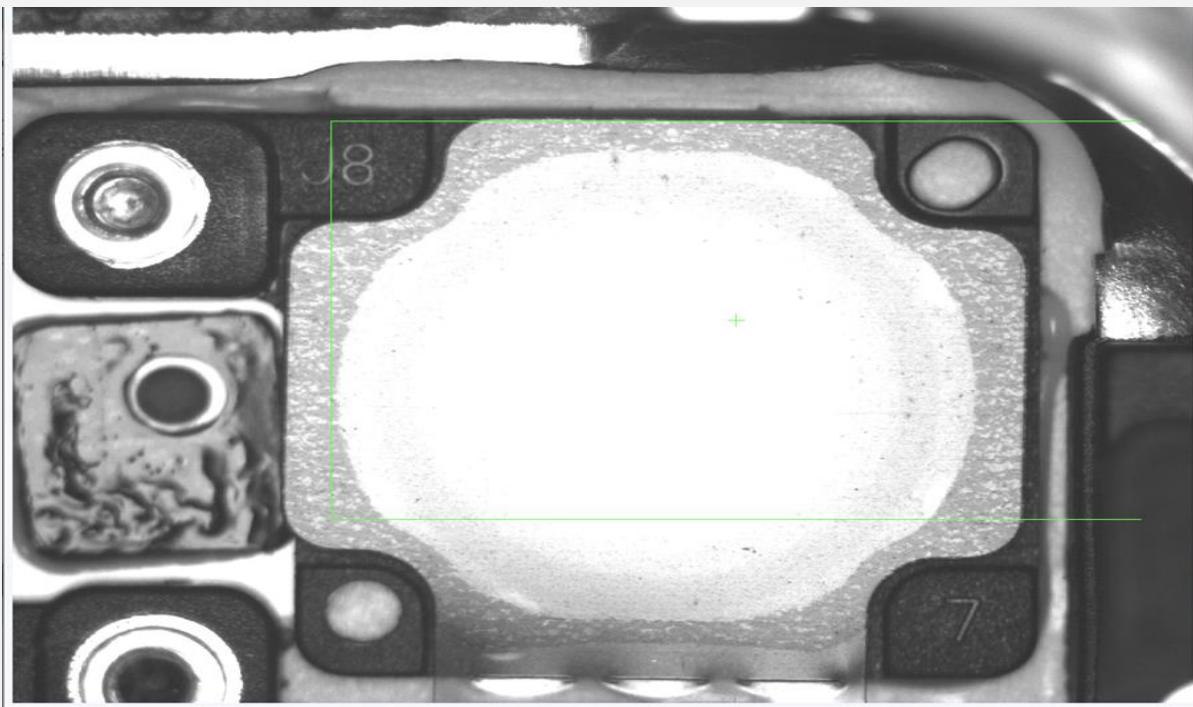
哈爾濱工業大學(深圳)
Harbin Institute of Technology, Shenzhen

Part.04

**金字塔： 四层 < - > 八层
结果对比**



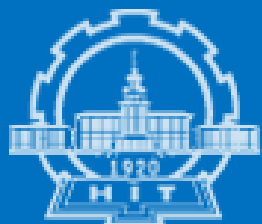
04 四层<->八层结果对比





04 四层<->八层结果对比

	4 层金字塔				8 层金字塔			
	匹配时 间/ms	匹配度	中心点		匹配时 间/ms	匹配度	中心点	
			X 坐标	Y 坐标			X 坐标	Y 坐标
1	1533.1	0.9365	1420.9	816.7	197.2	0.9433	1423.6	819.5
2	1685.9	0.9290	1416.0	853.6	210.2	0.9947	1417.5	853.5
3	1686.0	0.8912	1410.7	834.7	209.8	0.8923	1413.1	836.7
4	1609.9	0.9226	1442.0	848.8	209.7	0.9332	1442.9	849.5
5	1593.0	0.8920	1403.2	856.4	147.4	0.9382	1404.7	858.9
6	1729.6	0.9028	1431.9	860.0	208.7	0.9325	1432.0	861.1
7	1542.5	0.9013	1419.1	847.3	147.2	0.9210	1420.1	848.6
8	1710.5	0.8759	1451.2	877.5	209.5	0.8857	1452.0	879.9
9	1647.8	0.8994	1405.8	834.3	209.1	0.9022	1405.0	835.5
10	1597.1	0.9145	1418.3	831.1	209.8	0.9237	1419.8	832.3
平均	1633.5	0.907			195.9	0.927		



哈爾濱工業大學(深圳)
Harbin Institute of Technology, Shenzhen

Part.05

总结



感谢老师给了机会让我们有机会将课程所学与实际项目相结合，本次课设，既加深了我们对NCC图像金字塔的理解，也让我们锻炼了编程能力

谢谢观看！