

《机器视觉》实验指导书

V 1.0

实验一 光源及图像采集、系统标定实验



哈尔滨工业大学（深圳）
实验与创新实践教育中心

目录

1.1 实验目的	2
1.2 实验注意事项	2
1.3 实验内容	2
1.4 实验设备	2
1.5 实验步骤	6
1.5.1 图像采集	6
1.5.2 相机标定	7
1.5.3 编程实现图像采集	13
1.6 实验任务要求	18

1.1 实验目的

- 1) 掌握连接工业数字相机，熟悉使用实验平台软件采集图片；
- 2) 了解各种光源，以及工业相机常用参数，熟悉使用软件设置参数；
- 3) 掌握 Halcon 联合 VS 通过编程实现打开相机并采集图片、显示图片；

1.2 实验注意事项

- 1) 上机前做好充分准备，包括学习工业数字相机相关基础知识，了解相机常用参数的含义等。
- 2) 上机时要遵守实验室的规章制度，爱护实验设备。要熟悉与实验相关系统软件的使用方法。
- 3) 实验过程中关键器件轻拿轻放，禁止未经允许随意拆装。
- 4) 相机和光源接线时，必须仔细核对是否正确（否则极易损坏器件）。
- 5) 实验结束，必须确保实验平台所有器件回归原位并摆放整齐，关闭电源并整理实验台，经老师检查后方可离开。

1.3 实验内容

- (1) 实现相机连接和图像采集、相机标定，理解在实际项目中相机的采图过程。
- (2) VS 调用 Halcon 库，C++编程实现相机连接和图像采集、显示。

1.4 实验设备

实验设备包括机器视觉运动平台，德国 basler aca1300-60gm 黑白相机，低角度环形光源。



图 1-1 机器视觉运动平台

打开机器视觉实验运动平台下方电气柜窗口，总空气开关如图 1-2 的位置，向上推动，电脑进入开机状态。待电脑开机之后，将遥控手柄的急停开关向右旋转，开关抬起，并按下绿色按钮，给设备上电，如图 1-3 所示。

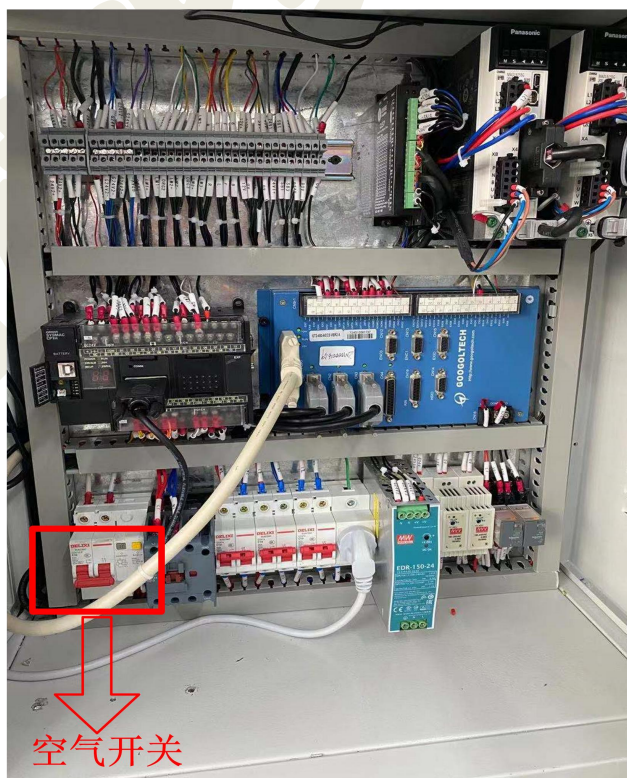


图 1-2 电控柜

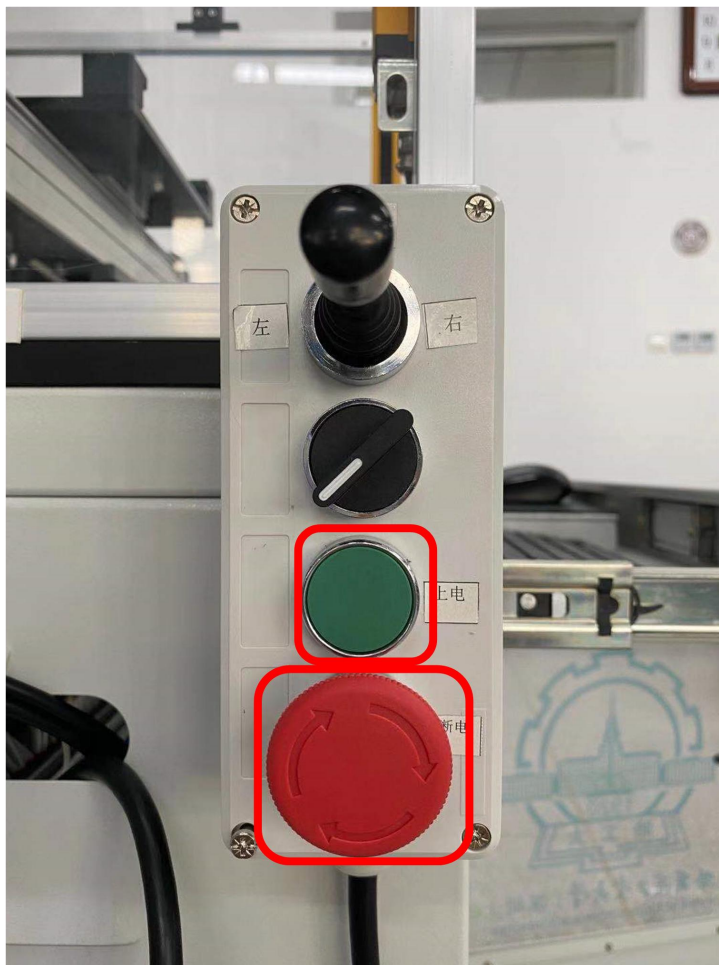


图 1-3 遥控手柄

图 1-4 所示为实验中陆续用到的各类光源，“1”为背光源，“2”为同轴光源，“3”为低角度环形光源，“4”为直射环形光源。德国 basler aca1300-60gm 黑白相机如图 1-5 所示。



图 1-4 各类光源示意图



图 1-5 basler aca1300-60gm 黑白相机

1.5 实验步骤

1.5.1 安装相机

- ① 给机器视觉实验平台上电，打开工控机，参见图 1-2、1-3。
- ② 转动运动平台 Z 轴上方的黑色旋钮，将相机卡在槽位，并旋紧旋钮固定相机，如图 1-6。

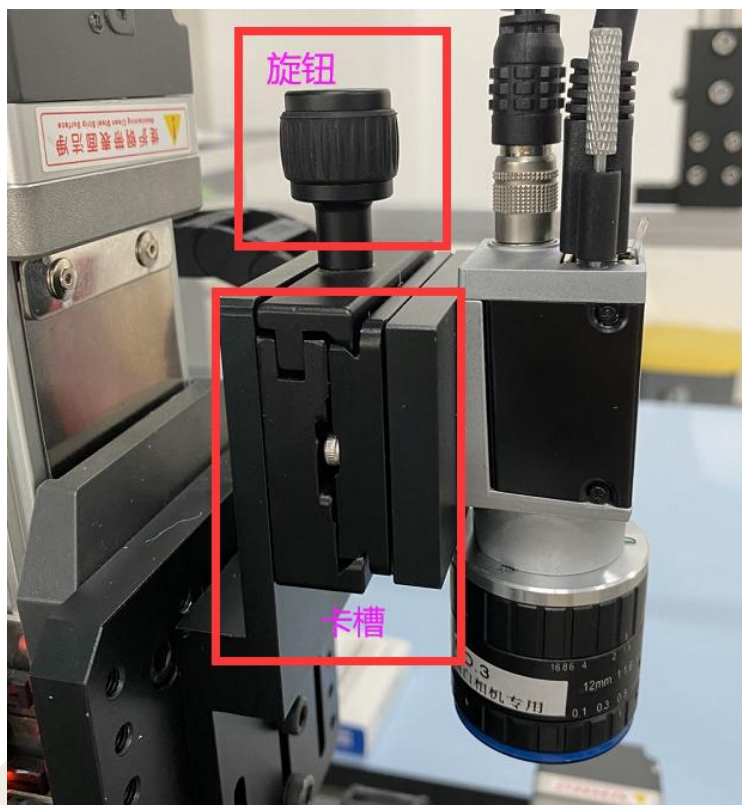


图 1-6 安装相机示意图

- ③ 连接相机网线、电源线如图 1-7，注意电源线的插头与插孔对准，切勿蛮力插拔。



图 1-7 连接相机示意图

④ 利用合适螺钉将光源固定在平台 Z 轴合适的位置上，参考下图 1-8，并将光源电源线接至平台 Z 轴后侧电源接口，“CH”按钮为通道选择，右侧两个按钮可以调节当前通道的亮度大小。



图 1-8 固定相机示意图

⑤ 若需要控制平台移动，应首先使能 X、Y 轴。方法是：启动桌面的“KImage 2D”软件，进入软件后，双击选择“线扫描拍照定位”项目。在弹出的提示是否保存配置，选择“否”。进入项目，点击“X 使能”和“Y 使能”，此时摇动摇杆即可控制平台进行 X、Y 方向移动，注意行程不要超出范围。

1.5.2 相机标定

相机参数需要根据实验相机的型号，查询该相机的像元尺寸、镜头的焦距等信息。在设置标定板参数时，若使用的是标准的标定板，则根据具体的标定板信息进行设置。

本次实验的标定板参数：

圆点列数：7

圆点行数：7

圆点间距：10 mm

圆点直径：5 mm

板厚度：3.75 mm

相机参数:水平/垂直像素尺寸: 5.3 X 5.3 μm

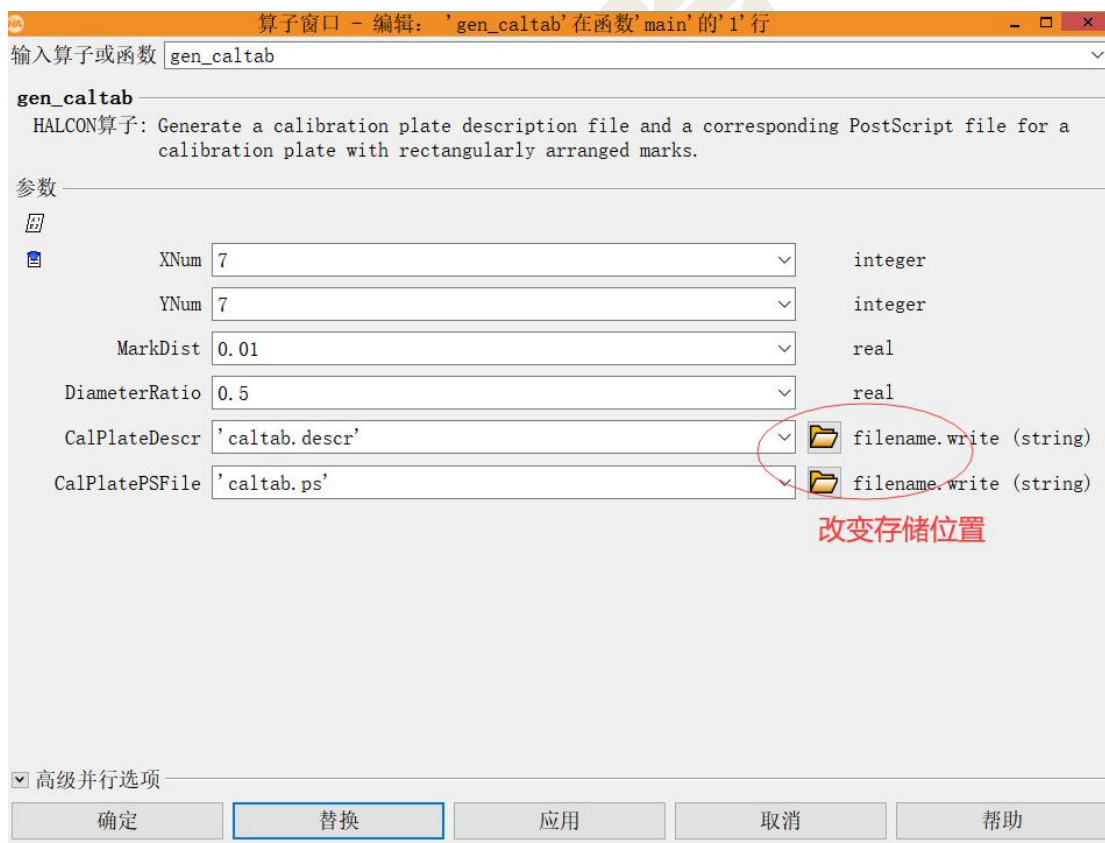
焦距: 12 mm

① 生成标定板

打开 Halcon, 输入算子 `gen_caltab`, 回车, 打开如下图 1-9 所示窗口, 即将生成一个 `.descr` 和一个 `.ps` 的描述文件。

本实验标定板点阵式 7*7, 每个圆点之间的间距是 0.01 米, 圆点直径与圆点间距离的比值为 0.5。生成的两个文件可以通过后面的文件夹符号 (红色圈出) 来改变存储位置, 建议新建文件夹, 勿使用软件默认存储位置。

点击“确定”, 运行后, 对应文件夹内会出现标定板文件。

图 1-9 算子 `gen_caltab`

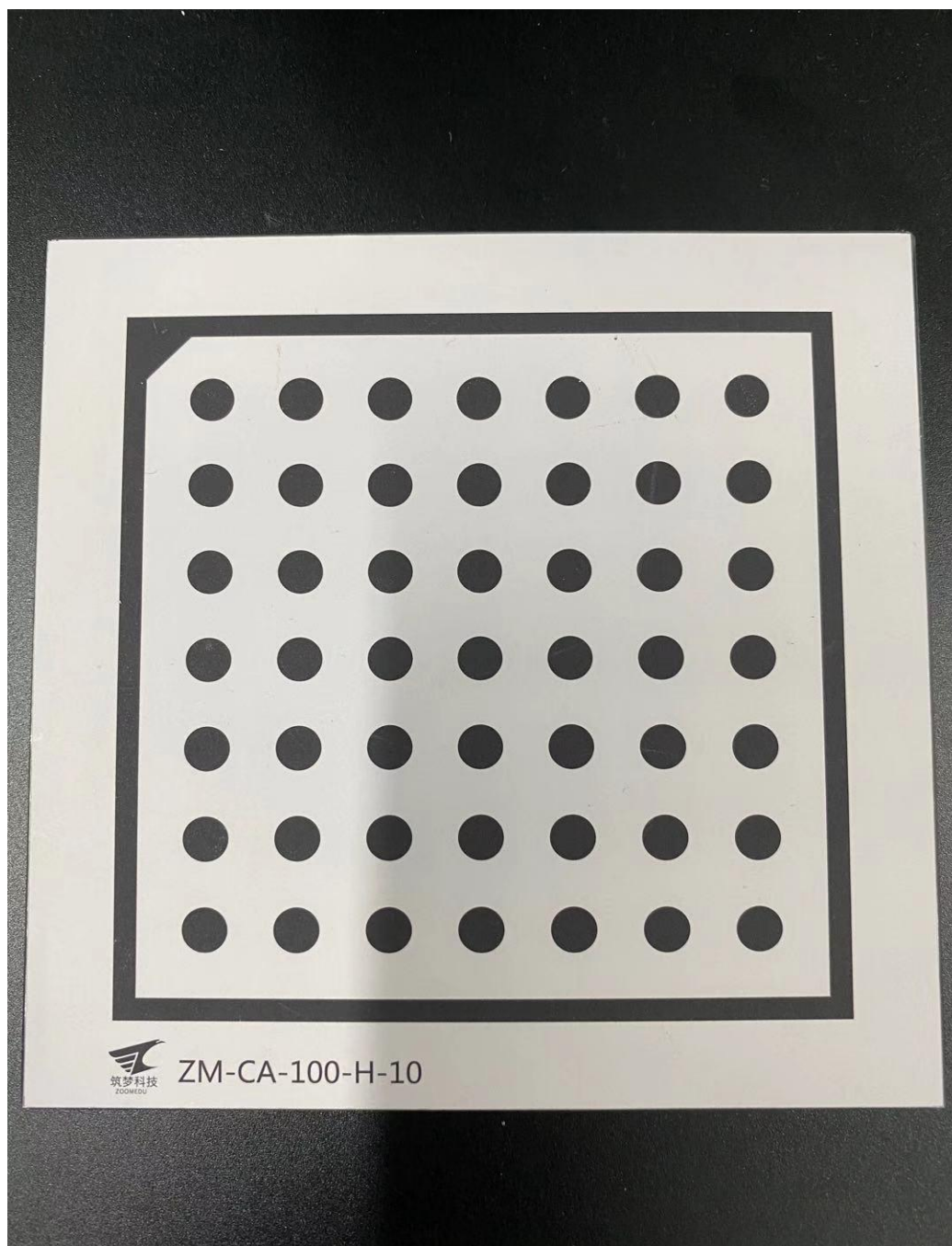


图 1-10 标定板 7X7

② 通过标定助手实现标定

打开标定助手，选择标定板的描述文件（刚才生成的.descr 的文件，不要用默认文件）、厚度和单个像元的宽高以及焦距，如下图。

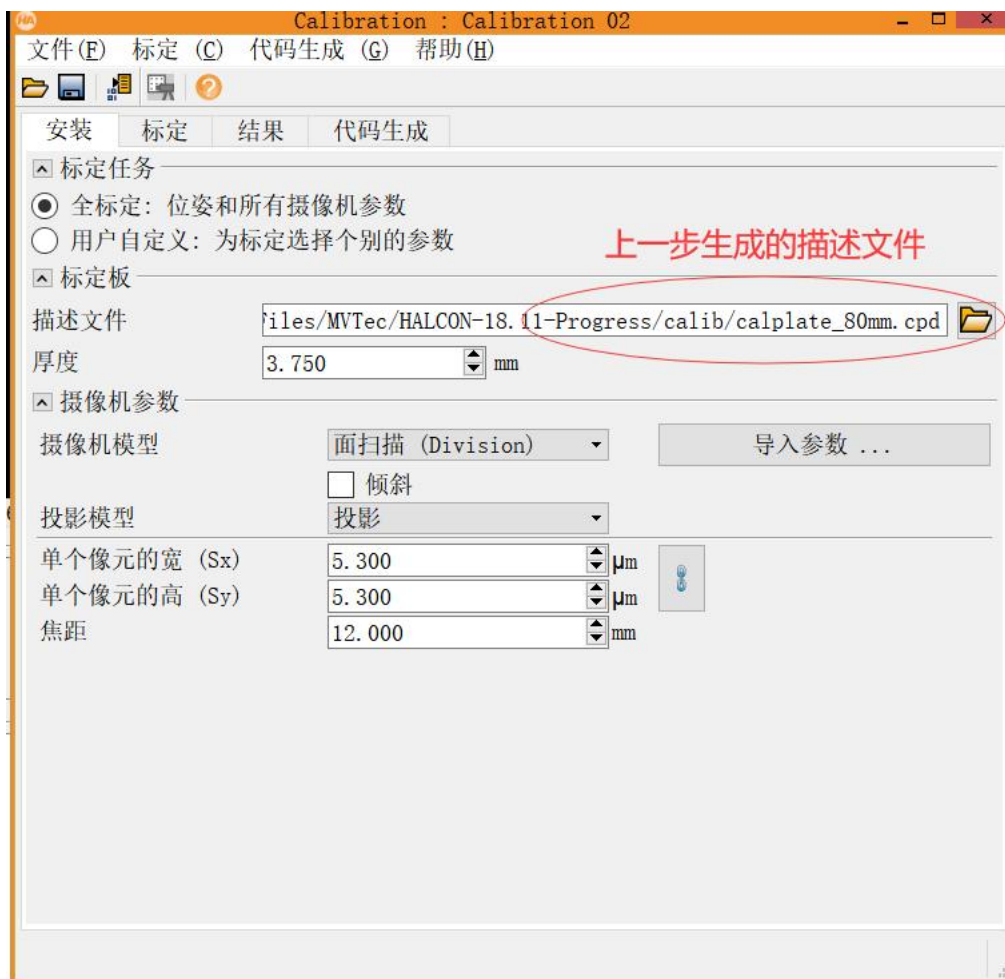



图 1-11 标定助手“安装”选项卡

③ 加载图像

可以实时采集，也可以采集好一起导入。一般推荐选择采集好后，一起导入的方式，即：**离线标定**。**注意**：相机为独占方式运行，同一时刻只允许一个线程占用相机，并且必须为“软触发”模式。若使用 Halcon 采集助手采集图片出现提示“超时”，软件卡死采图失败，则需更改相机当前模式。方法是：关闭 Halcon，保证相机不被其他软件占用，双击桌面“basler 驱动”软件，点击右上角图标“”连接相机，在“Acquisition Controls”的 Trigger Mode 由“ON”改为“OFF”，如下图。

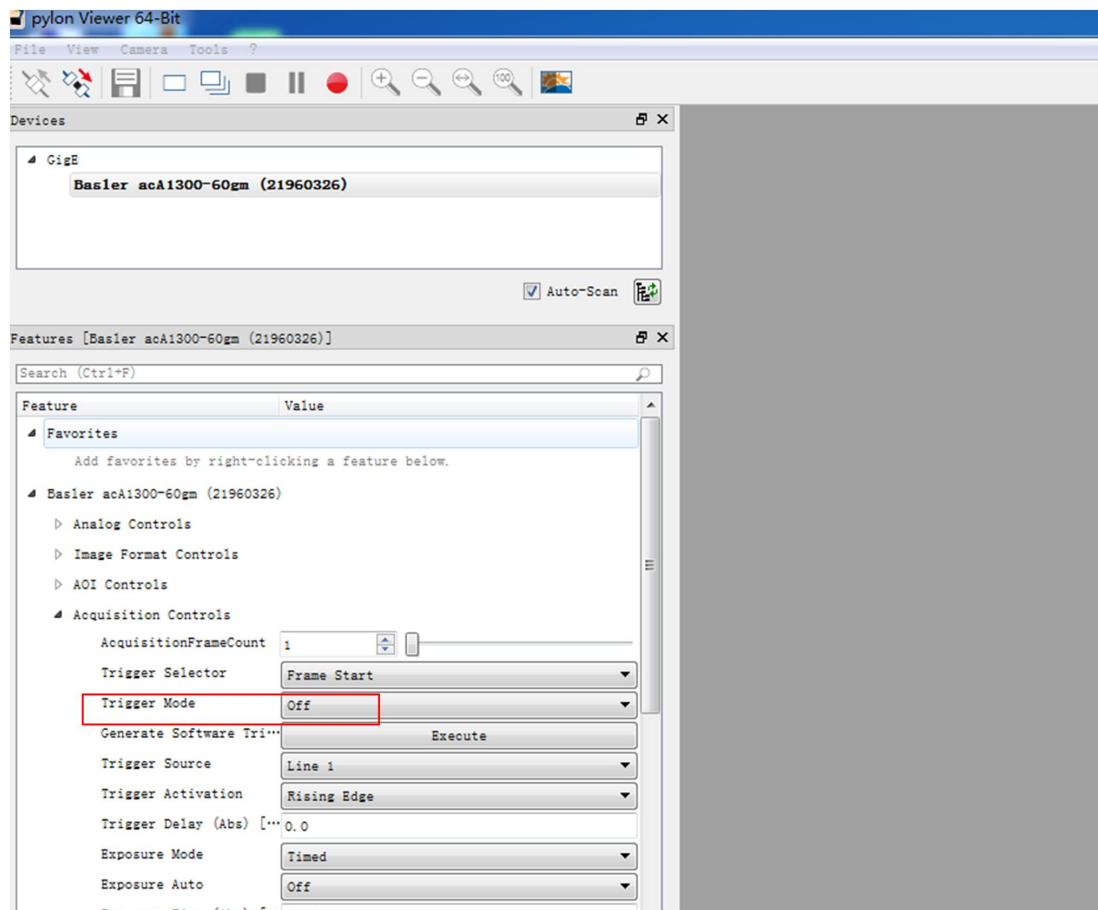


图 1-12 相机卡死处理方法

采集各个角度和位置的图像（大于 10 张），将载入的图像选择一幅为参考位姿，点击“设为参考位姿”，在“品质问题”中，显示对标定图像质量的评价，若精度要求不是非常高的场合，提示“检测出品质问题”可以酌情接受；若提示“标定点提取失败”则图像不可用，如果提示“确定”则表明图像质量没问题。选择状态为“确定”的一幅图像作为参考图像。单击右侧“标定”按钮，即可自行标定。如下图 1-13。

注意：实验过程中如需移动工作台，应首先使能 X、Y 轴。方法是：启动桌面的“KImage 2D”软件，进入软件后，双击选择“线扫描拍照定位”项目。弹出提示是否保存配置，选择“否”。进入项目，点击“X 使能”和“Y 使能”，此时摇动摇杆即可控制平台进行 X、Y 方向移动，注意行程不要超出范围。

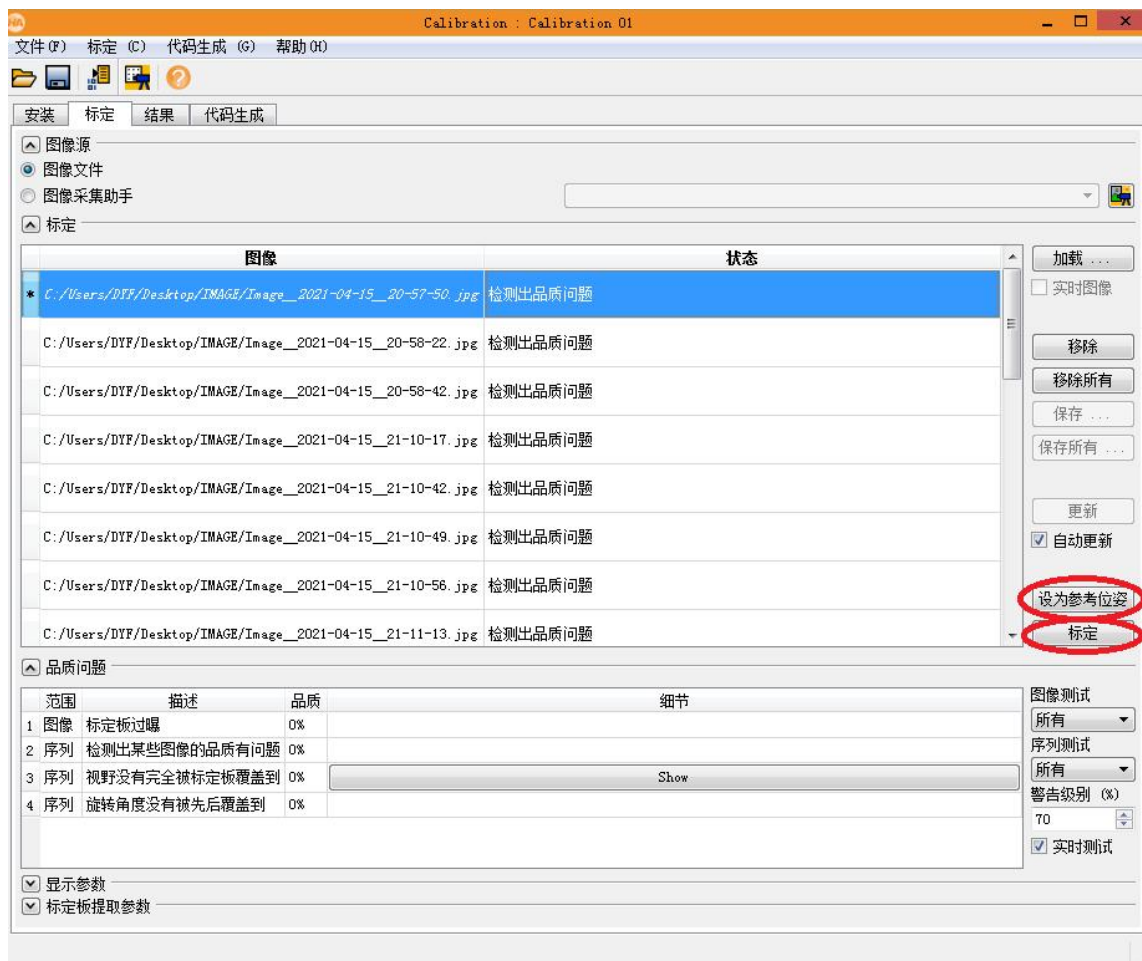


图 1-13 标定助手“标定”选项卡

④ 标定结果

标定完成后，“结果”选项卡显示标定的参数结果，状态显示“标定成功”，得到相机内参和外参。如图 1-14。插入代码即可将代码插入程序中。

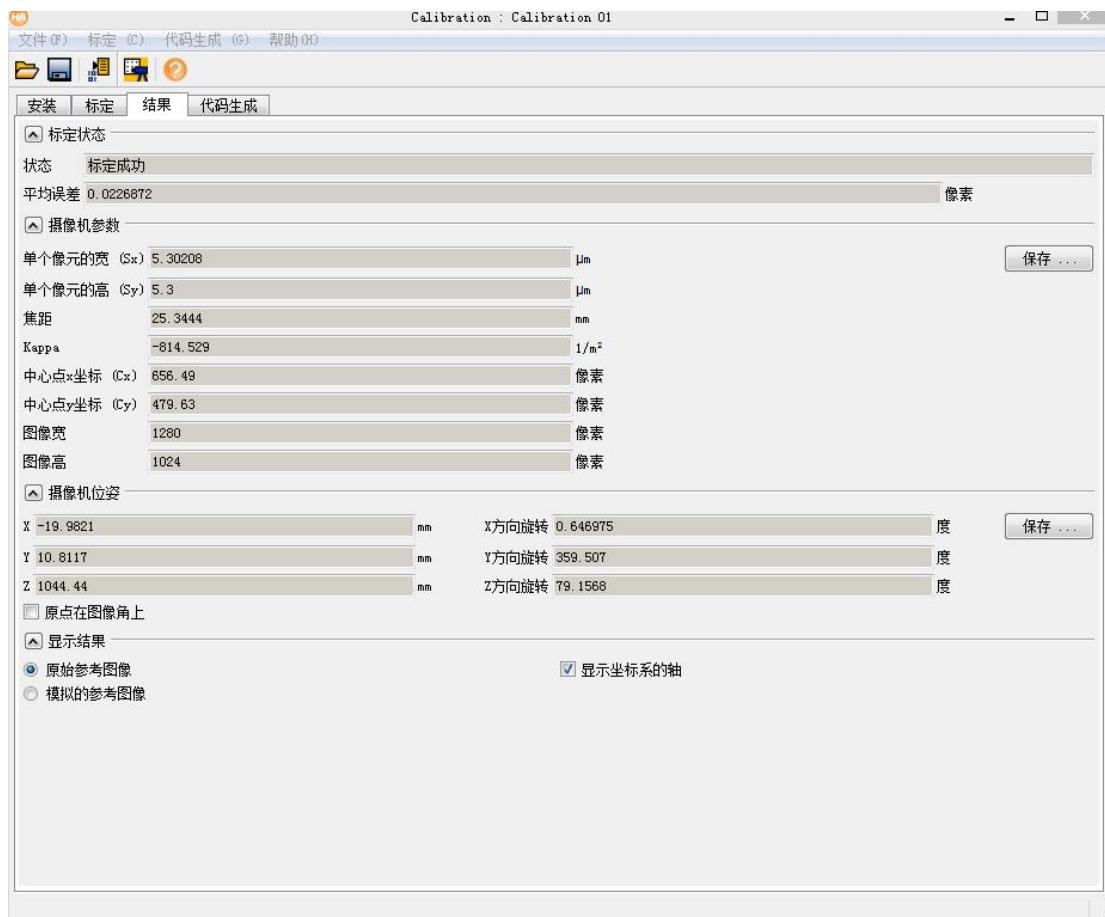


图 1-14 “结果”选项卡

1.5.3 编程实现图像采集

① 在开始之前需要安装好 Halcon 12（及以上版本）和 VS2010（及以上版本），实验室电脑安装 VS 版本为 2010，所以这里以 VS2010 为例。运行 VS2010，新建项目，并选择控制台应用程序（**注意是 64 位工程**）。

② 将 Halcon 相关路径导入到工程，在解决方案中右击，选择属性——配置属性——C++——常规——附加包含目录，包含 halcon 相关头文件。如图 1-15 所示。

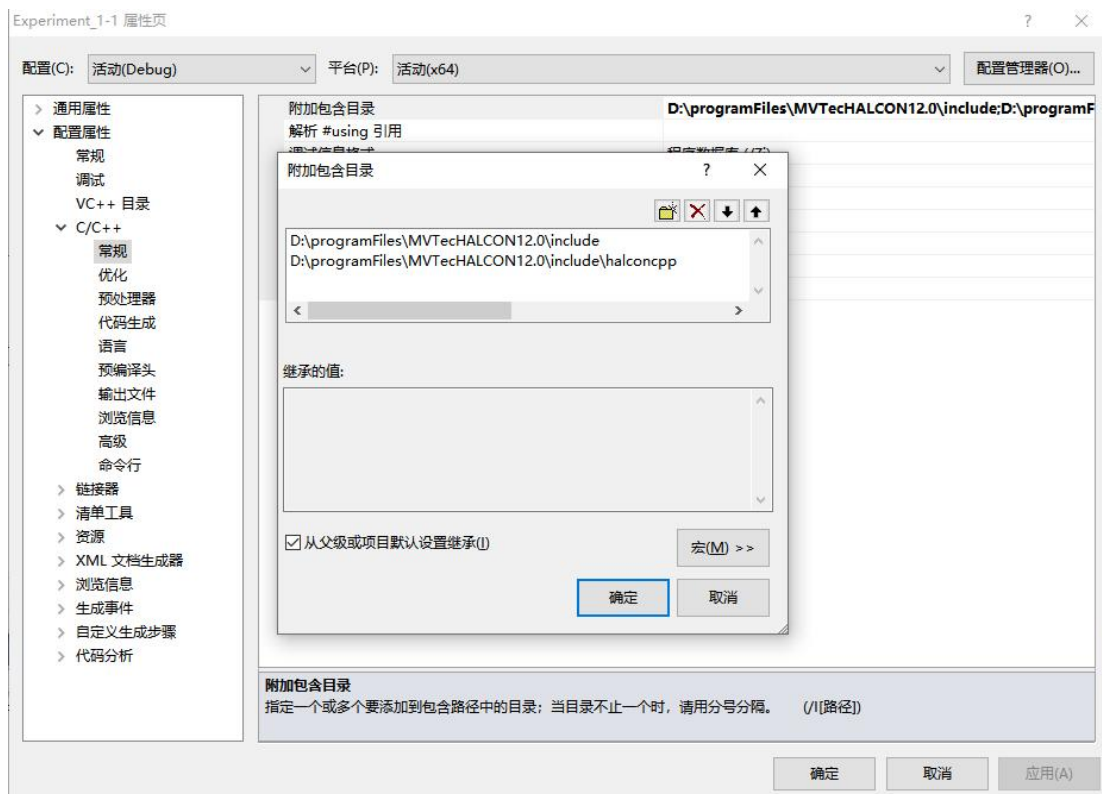


图 1-15 包含 halcon 相关头文件

③ 在解决方案中右击，选择属性——配置属性——链接器——常规——附加库目录，包含 halcon 相关库文件（一定是 64 位）。

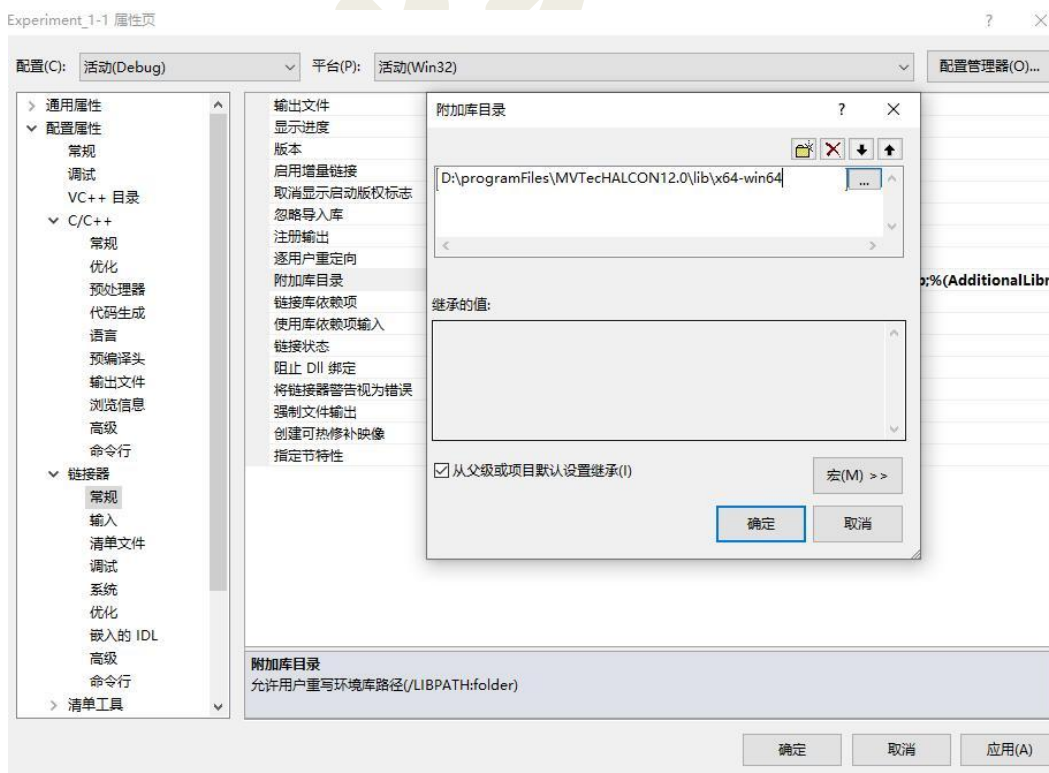


图 1-16 包含 halcon 相关库文件

④ 在解决方案中右击，选择属性——配置属性——链接器——输入——附加依赖项，填写依赖库。

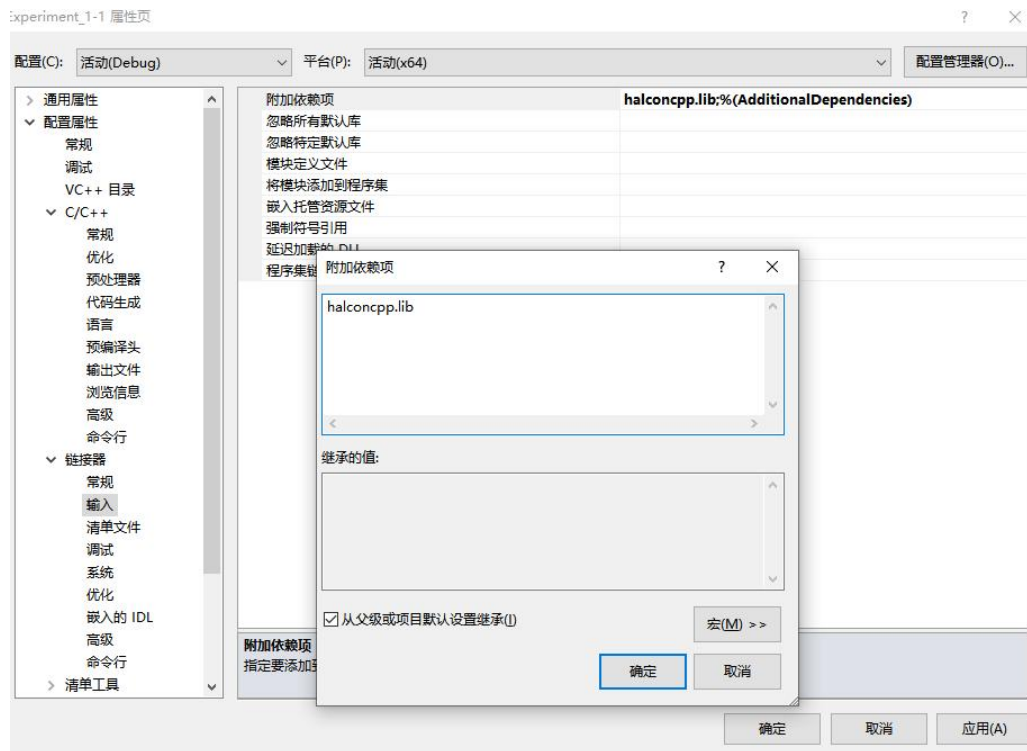


图 1-17 填写依赖库

⑤ 运行 Halcon ， 点击工具栏的“助手”——“打开新的 Image Acquisition”。



图 1-18 Image Acquisition

⑥ 点击“自动检测接口”，Halcon 将自动搜寻目前电脑连接的相机，切换到“连接”标签页。



图 1-19 “连接”标签页

⑦ 点击“连接”——“采集”或“实时”即完成单张/实时采图。

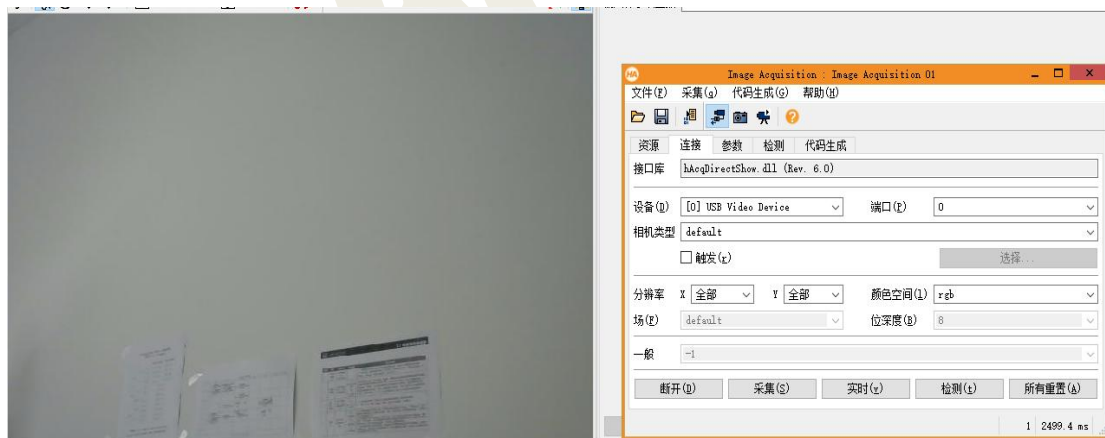
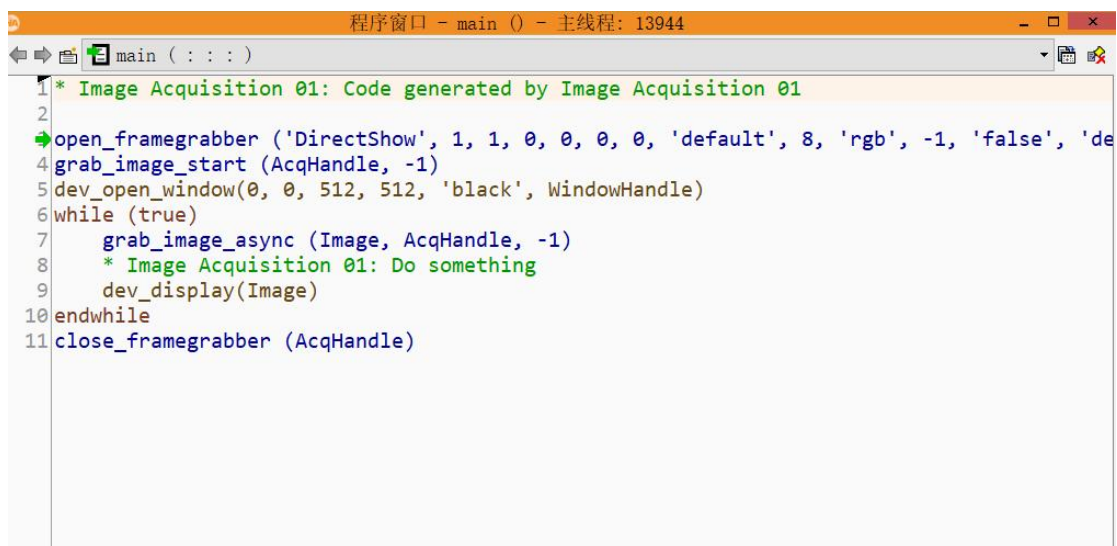


图 1-20 采集图像

⑧ 点击“代码生成”——“插入代码”，在程序编辑器可看到上述操作的 Halcon 代码。这里需要注意的是，一定要在生成的代码基础上，手动添加“dev_open_window”算子，否则导出的 C++ 程序在运行时，不会自动创建窗口，无法显示图片框。



```

程序窗口 - main () - 主线程: 13944
main ( : : : )
1 * Image Acquisition 01: Code generated by Image Acquisition 01
2
3 open_framegrabber ('DirectShow', 1, 1, 0, 0, 0, 0, 'default', 8, 'rgb', -1, 'false', 'de
4 grab_image_start (AcqHandle, -1)
5 dev_open_window(0, 0, 512, 512, 'black', WindowHandle)
6 while (true)
7     grab_image_async (Image, AcqHandle, -1)
8     * Image Acquisition 01: Do something
9     dev_display(Image)
10 endwhile
11 close_framegrabber (AcqHandle)

```

图 1-21 Halcon 代码

⑨ 选择工具栏“文件”——“导出”，如下图 1-22 所示，选择“C++”，即可生成 C++代码。



图 1-22 生成 C++代码

⑩ 用 VS2010 打开生成的 cpp 文件。阅读导出的 C++代码，可以看出连接

相机并采图的操作逻辑，重点关注“**action()**”函数体部分。红色框出的部分为打开相机的功能函数，当使用 halcon 打开相机时，该函数的各个参数赋值会自动生成，使用不同的接口以及相机，匹配的参数也会不同。

```

void action()
{
    // Local iconic variables
    HObject ho_Image;

    // Local control variables
    HTuple hv_AcqHandle;

    //Image Acquisition 01: Code generated by Image Acquisition 01
    OpenFramegrabber("DirectShow", 1, 1, 0, 0, 0, 0, "default", 8, "rgb", -1, "false",
        "default", "[0] USB Video Device", 0, -1, &hv_AcqHandle);
    while (0 != 1)
    {
        GrabImage(&ho_Image, hv_AcqHandle);
        //Image Acquisition 01: Do something
    }
    CloseFramegrabber(hv_AcqHandle);
}

```

图 1-23 函数参数赋值

主要代码如上图 1-23 所示。可以看到“OpenFramegrabber”就是直接使用 Halcon 导出的函数。若当前连接的相机不同，使用接口不同，该函数的参数也会不同，参数初始值由 halcon 自动生成。

⑪ 新建 VS 控制台应用程序，注意切换到 64 位，将 Halcon 导出的 C++ 代码移植到 VS 编辑区，注意头文件“include "HalconCpp.h"”，以及命名空间“using namespace HalconCpp;”两段代码的添加。配置相关 Halcon 链接库、头文件的路径，详见步骤①~④。最后编译、运行 VS 程序，弹窗显示采集到的图片。

1.6 实验任务要求

- (1) 通过 Halcon 实现相机标定；
- (2) Halcon 导出代码在 VS 平台运行，实现采集图像。