

浙江大学

本科实验报告

相机标定和鸟瞰图转换

课程名称： 计算机视觉

姓名： 夏豪诚

学院： 计算机科学与技术学院

专业： 信息安全

学号： 3170102492

指导老师： 宋明黎

2019 年 12 月 22 日

浙江大学实验报告

专业： 信息安全
姓名： 夏豪诚
学号： 3170102492
日期： 2019 年 12 月 22 日
地点： 无

课程名称： 计算机视觉 指导老师： 宋明黎 成绩：
实验名称： 相机标定和鸟瞰图转换 实验类型： 综合实验 同组学生姓名： 无

一、 实验环境

表 1: 测试环境

| item | detail |
|-----------|--|
| CPU | Intel® Core™ i7-6700HQ CPU 2.60GHz |
| RAM | 16.0GB DDR4 2133MHz |
| hard disk | SSD 256GB |
| OS | Windows 10 Pro 64-bit |
| IDE | Microsoft Visual Studio Community 2019 |
| OpenCV | 3.4.5 |

二、 实验目的和要求

1. 实验目的

实现相机参数的标定，并利用相机标定得到的参数作为输入实现鸟瞰图的转换与生成。

2. 基本要求

将《Learning OpenCV》书中上的三个例程，合并为一个程序。

- (1) Camera calibration 章节中的 Example 11-1, 11-2(对应于《Learning OpenCV3》中的 example_18-01.cpp 和 example_18-01_from_disk.cpp);
- (2) Bird' s-eye view transformation 章节中的 Example 12-1(对应于《Learning OpenCV3》中的 example_19-01.cpp)

实现的功能效果为：可以从摄像头中采集棋盘的图像，也可以从指定目录下读取图片进行相机标定。在相机标定完成后，标定参数存储在 intrinsics.xml 文件中。根据标定参数实现鸟瞰图转换，生成鸟瞰图。

三、实验内容和步骤

1. 实验内容

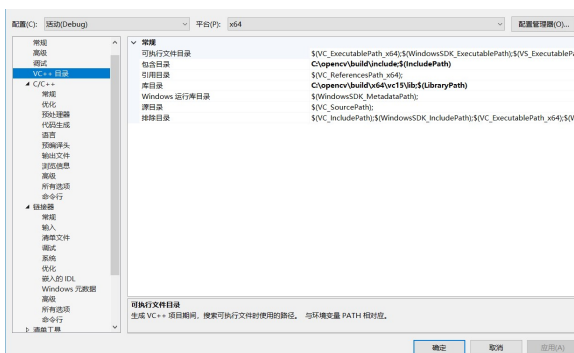
- (1) 在 TransBEV.cpp 中调用三个例程文件 example_18-01.cpp, example_18-01_from_disk.cpp 和 example_19-01.cpp。

针对这个部分的具体代码实现将在实验步骤中进行详细说明。

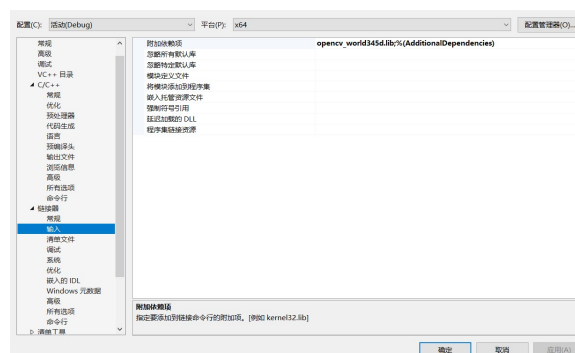
2. 实验步骤

3.2.0 实验环境配置

环境配置与之前实验相比略有不同，相同部分为在 Windows 10 操作系统的环境下，我们可以在官网下载 exe 后缀的可执行文件，运行后可以将编译好的对应版本 opencv 解压到制定的目录下，在本次实验中我选择的版本为 3.4.5，解压完成后为了可以在 VS 中调用 opencv 库函数我们需要修改 VC++ 目录下的包含目录和库目录添加 opencv 的解压位置，而后在链接器-> 输入-> 附加依赖项中添加 lib 文件。



(a) 配置图 1.



(b) 配置图 2.

图 1: VS 配置界面

不同的是，由于在 example_18-01_from_disk.cpp 的源码中有如下定义 `#include <dirent.h>`，而在 MSVC 的头文件中并不包括此文件，直接运行的时候会出现报错的情况，所以我们可以从 GitHub 上获得 [dirent.h](https://github.com/opencv/opencv/blob/master/include/opencv2/dirent.h)，并在 visual studio 的 MSVC 的 include 文件夹中添加此头文件。

| 此电脑 > 系统 (C:) > Program Files (x86) > Microsoft Visual Studio > 2019 > Community > VC > Tools > MSVC > 14.21.27702 > include | | | | |
|--|-----------------|--------------|-------|--|
| 名称 | 修改日期 | 类型 | 大小 | |
| delayimp.cpp | 2019/6/26 17:22 | C++ 源文件 | 5 KB | |
| delayimp.h | 2019/6/26 17:22 | C Header 源文件 | 5 KB | |
| deque | 2019/6/26 17:22 | 文件 | 62 KB | |
| dirent.h | 2018/5/9 1:53 | C Header 源文件 | 27 KB | |
| dloadsup.h | 2019/6/26 17:22 | C Header 源文件 | 23 KB | |
| dvec.h | 2019/6/26 17:22 | C Header 源文件 | 70 KB | |

图 2: 添加头文件

3.2.1 调用三个例程文件

我的实现方式是新建一个源文件 TransBEV.cpp，在其中调用三个例程文件的 main 函数。最终的工程结构如下：



图 3: 工程结构

头文件 TransBEV.h 中的内容如下：

```
#pragma once
#ifndef TRANSBEV_H
#define TRANSBEV_H

int ex18_1(int argc, char** argv);
int ex18_1_disk(int argc, char** argv);
int ex19_1(int argc, char** argv);

#endif
```

在不同例程文件中分别添加如下宏定义,定义main为ex18_1,ex18_1_disk和 ex19_1。而后在 TransBEV.cpp 中调用。TransBEV.cpp 的内容如下：

```
#include "TransBEV.h"
#include <iostream>

int main(int argc, char** argv)
{
    // according to the number of para to judge the picture from Camera or Disk
    int disk = -1; // default -1 means no format match
    if (argc == 8) disk = 0; // 0 means from Camera
    else if (argc == 9) disk = 1; // 1 means from disk
    switch (disk)
    {
    case 0:
        ex18_1(argc, argv);
        break;
    case 1:
        ex18_1_disk(argc, argv);
        break;
    default:
        std::cout << "please enter the correct format parameters\n"
```

```
<< "eg: TransBEV.exe 12 12 28 100 0.5 D:/calibration/ intrinsics.xml
    D:/birdseye/IMG_0215L.jpg \n"
<< "from disk: TransBEV.exe <1board_width> <2board_height> <3number_of_boards> \n"
<< "<4ms_delay_frameee_capture> <5image_scaling_factor> <6path_to_calibration_images>
    \n"
<< "<7path/camera_calib_filename> <8path/chessboard_image> \n"
<< "from camera: TransBEV.exe <1board_width> <2board_height> <3number_of_boards> \n"
<< "<4if_video,_delay_between_frameee_capture> <5image_scaling_factor> \n"
<< "<6path/camera_calib_filename> <7path/chessboard_image>"
<< std::endl;
return -1;
}
ex19_1(argc, argv);
return 0;
}
```

判断是从摄像头还是磁盘中获得图像,选择调用 example_18-01.cpp 或 example_18-01_from_disk.cpp 完成相机标定,在利用相机标定参数进行鸟瞰图转换和生成。

四、 主要仪器设备

计算机, Visual Studio 2019

五、 实验结果

1. 编译运行

在 VS2019 中编译成功后,在控制台运行可执行程序 TransBEV.exe,并指定对应参数。此时运行的命令仿照例程中的参数示例,为:

```
TransBEV.exe 12 12 28 100 0.5 D:/calibration/ intrinsics.xml D:/birdseye/IMG_0220L.jpg
```

控制台输出如下:

```

C:\Users\hc\Source\Repos\TransBEV\x64\Debug>TransBEV.exe 12 12 28 100 0.5 D:/calibration/ intrinsics.xml D:/birdseye/IMG_0220L.jpg
Reading in directory D:/calibration/
... Done. Number of files = 23
Collected 1total boards. This one from chessboard image #0, D:/calibration/IMG_0191.jpg
Collected 2total boards. This one from chessboard image #1, D:/calibration/IMG_0192.jpg
Collected 3total boards. This one from chessboard image #2, D:/calibration/IMG_0193.jpg
Collected 4total boards. This one from chessboard image #3, D:/calibration/IMG_0194.jpg
Collected 5total boards. This one from chessboard image #5, D:/calibration/IMG_0196.jpg
Collected 6total boards. This one from chessboard image #6, D:/calibration/IMG_0197.jpg
Collected 7total boards. This one from chessboard image #7, D:/calibration/IMG_0198.jpg
Collected 8total boards. This one from chessboard image #8, D:/calibration/IMG_0199.jpg
Collected 9total boards. This one from chessboard image #9, D:/calibration/IMG_0200.jpg
Collected 10total boards. This one from chessboard image #10, D:/calibration/IMG_0201.jpg
Collected 11total boards. This one from chessboard image #11, D:/calibration/IMG_0202.jpg
Collected 12total boards. This one from chessboard image #12, D:/calibration/IMG_0203.jpg
Collected 13total boards. This one from chessboard image #13, D:/calibration/IMG_0204.jpg
Collected 14total boards. This one from chessboard image #14, D:/calibration/IMG_0205.jpg
Collected 15total boards. This one from chessboard image #15, D:/calibration/IMG_0206.jpg
Collected 16total boards. This one from chessboard image #16, D:/calibration/IMG_0207.jpg
Collected 17total boards. This one from chessboard image #17, D:/calibration/IMG_0208.jpg
Collected 18total boards. This one from chessboard image #18, D:/calibration/IMG_0209.jpg
Collected 19total boards. This one from chessboard image #19, D:/calibration/IMG_0210.jpg
Collected 20total boards. This one from chessboard image #20, D:/calibration/IMG_0211.jpg
Collected 21total boards. This one from chessboard image #21, D:/calibration/IMG_0212.jpg
Collected 22total boards. This one from chessboard image #22, D:/calibration/IMG_0213.jpg

*** CALIBRATING THE CAMERA...

*** DONE!

Reprojection error is 0.788185
Storing Intrinsics.xml and Distortions.xml files

Image width: 1600
Image height: 1200
Intrinsic matrix:[1744.376097688622, 0, 800;
0, 1747.864518956945, 600;
0, 0, 1]
distortion coefficients: [0.09168668059630308, -1.183312595693614, 0, 0, 4.192775177933611]

*****
PRESS A KEY TO SEE THE NEXT IMAGE, ESQ TO QUIT
*****

Press 'd' for lower birdseye view, and 'u' for higher (it adjusts the apparent 'Z' height), Esc to exit

```

图 4: TransBEV.exe 运行的控制台输出

2. 文件变化

我们首先查看目录下的文件，看到 TransBEV.exe 运行的结果，相机标定参数文件 **intrinsics.xml**。

```

C:\Users\hc\Source\Repos\TransBEV\x64\Debug 的目录
2019/12/22 21:36 <DIR>
2019/12/22 21:36 <DIR>
2019/12/22 21:00 1,018,341 example_18-01.obj
2019/12/22 21:00 1,475,894 example_18-01_from_disk.obj
2019/12/22 21:00 768,785 example_19-01.obj
2019/12/22 18:22 563 intrinsics.xml
2019/12/21 23:31 76,998,072 opencv_world345.dll
2018/12/21 23:25 120,488,752 opencv_world345d.dll
2019/12/22 21:01 357,376 TransBEV.exe
2019/12/22 21:01 1,810,556 TransBEV.ilc
2019/12/22 21:01 99 TransBEV.log
2019/12/22 21:01 60,426 TransBEV.obj
2019/12/22 21:01 1,593,344 TransBEV.pdb
2019/12/22 21:01 <DIR> TransBEV.tlog
2019/12/22 21:01 838,656 vc142.idb
2019/12/22 21:01 2,134,016 vc142.pdb
13 个文件 207,511,880 字节
3 个目录 67,972,014,080 可用字节

```

图 5: 文件变化

intrinsics.xml的内容如下：

```

<?xml version="1.0"?>
<opencv_storage>
<image_width>1600</image_width>
<image_height>1200</image_height>
<camera_matrix type_id="opencv-matrix">
  <rows>3</rows>
  <cols>3</cols>
  <dt>d</dt>

```

```
<data>
  1.7443760976886224e+03 0. 800. 0. 1.7478645189569454e+03 600. 0. 0.
  1.</data></camera_matrix>
<distortion_coefficients type_id="opencv-matrix">
  <rows>1</rows>
  <cols>5</cols>
  <dt>d</dt>
  <data>
    9.1686680596303083e-02 -1.1833125956936137e+00 0. 0.
    4.1927751779336111e+00</data></distortion_coefficients>
</opencv_storage>
```

3. 运行结果

部分图片标定结果。

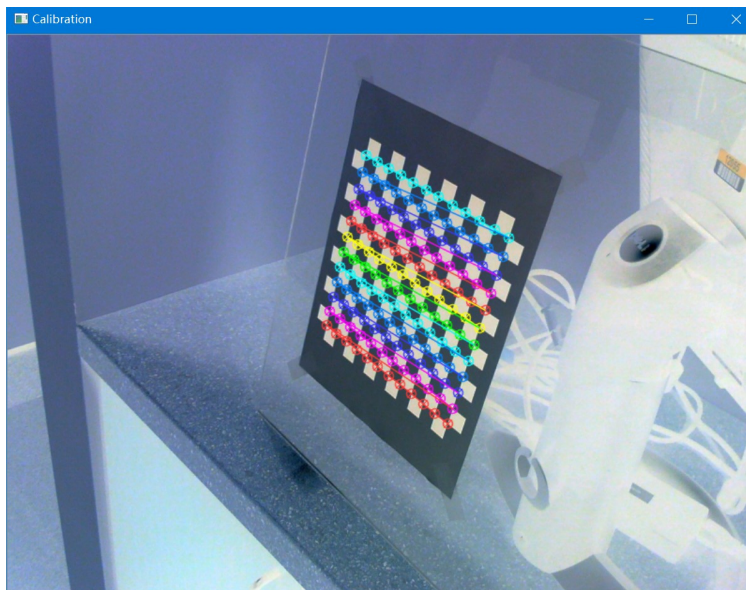


图 6: 图片标定结果

部分去畸变结果。

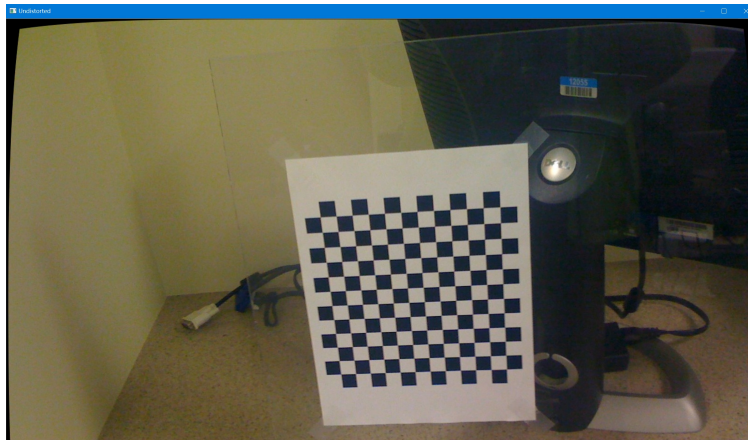


图 7: 去畸变结果

生成鸟瞰图结果。

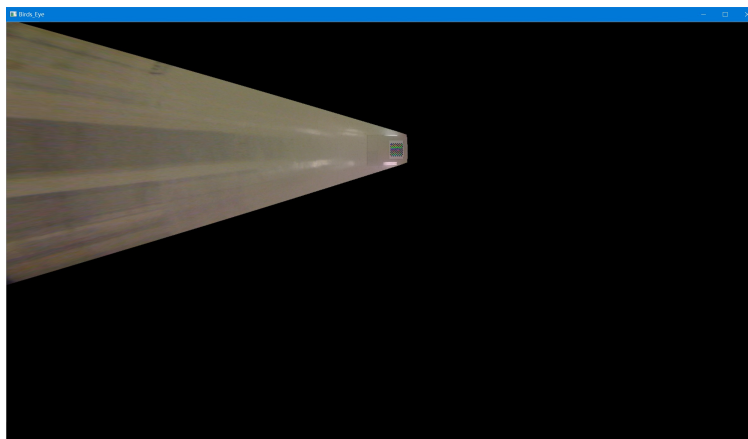


图 8: 鸟瞰图结果

六、 实验结果分析

在本次实验中总体而言工作量很小，因为只是直接使用例程，但是去理解代码中实现相机标定和鸟瞰图转换的过程仍然花费了一些时间，这次实验与课堂知识有着很好的衔接也更让我感受到 cv 能做到的从图像中提取信息，并利用信息这些进行处理强大能力，获益匪浅，希望能在之后的课程中学到更多的知识。