

# 第二次算法组培训

## ——opencv之图像处理

### 1.课前准备：

- 掌握Linux，CMake基本操作：CMakeLists编写，Linux终端make编译c++工程
- 下载并编译opencv，跑通example程序
- 可以了解一下opencv图像处理的基本内容

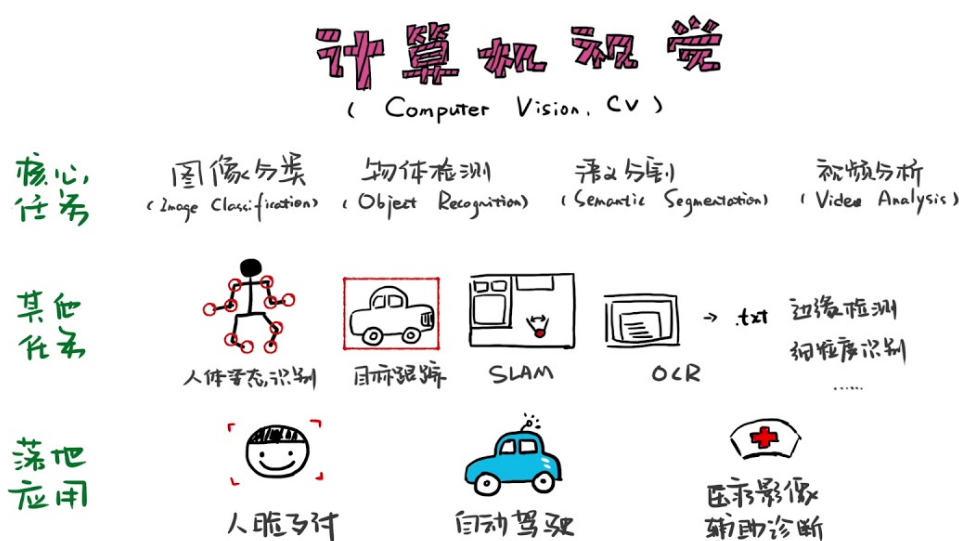
可供参考的文档和网址：OpenCV3编程入门.pdf

[【OpenCV入门教程之四】ROI区域图像叠加&初级图像混合 全剖析-CSDN博客](#)

b站上opencv-c++相关教程

### 2.什么是计算机视觉

计算机视觉是一门研究如何使机器“看”的科学，更进一步的说，就是指用摄影机和电脑代替人眼对目标进行识别、跟踪和测量等机器视觉，并进一步做图形处理，使电脑处理成为更适合人眼观察或传送给仪器检测的图像或其他形式的信息。



### 3.图像在计算机中的表示：

首先我们第一个要深深深深的刻在脑子里的概念就是：

图片是由像素点构成的

而像素中的点存储着数字，对应不同的颜色



根据每个像素的表示方式，图像一般分为三类：

- 二值图
- 灰度图
- 彩色图像（RGB、HSV、YUV）

## 4.关于opencv:

---

OpenCV（Open Source Computer Vision Library）是一个开源的计算机视觉库，它提供了很多函数，这些函数非常高效地实现了计算机视觉算法（最基本的滤波到高级的物体检测皆有涵盖）



OpenCV 使用 C/C++ 开发，同时也提供了 Python、Java、MATLAB 等其他语言的接口。

OpenCV 是跨平台的，可以在 Windows、Linux、Mac OS、Android、iOS 等操作系统上运行。

OpenCV 的应用领域非常广泛，包括图像拼接、图像降噪、产品质检、人机交互、人脸识别、动作识别、动作跟踪、无人驾驶等。

OpenCV 还提供了机器学习模块，你可以使用正态贝叶斯、K最近邻、支持向量机、决策树、随机森林、人工神经网络等机器学习算法。

OpenCV是一个开源的计算机视觉库，可以从 <http://opencv.org> 获取

■ 网上有很多opencv的安装详细教程，这里就不着重介绍了，要求在Linux-c++环境下配置opencv  
一个简单的样例：使用opencv读入图像，显示图像，显示彩色空间转灰度图后的图像，保存图像

```
#include <opencv2/opencv.hpp>
#include <iostream>
using namespace std;
using namespace cv;
int main(int, char *argv[])
{
    Mat in_image, out_image;
    // Usage: <cmd> <file_in> <file_out>
    //读取原始图像
    in_image = imread(filepath); //filepath-读入图像路径
    if (in_image.empty())
    {
        //检查是否读取图像
        cout << "Error! Input image cannot be read...\n";
        return -1;
    }
    //创建两个具有图像名称的窗口
    namedWindow("show1", WINDOW_AUTOSIZE);
    namedWindow("show2", WINDOW_AUTOSIZE);
    //在之前创建的窗口中显示图片
    imshow("show1", in_image);
    cvtColor(in_image, out_image, COLOR_BGR2GRAY); //BGR转灰度图
    imshow("show2", in_image);
    cout << "Press any key to exit...\n";
```

```
waitKey(); // wait for key press
//写入图像
imwrite(filepath2, in_image); //filepath2-写入图像路径
return 0;
}
```