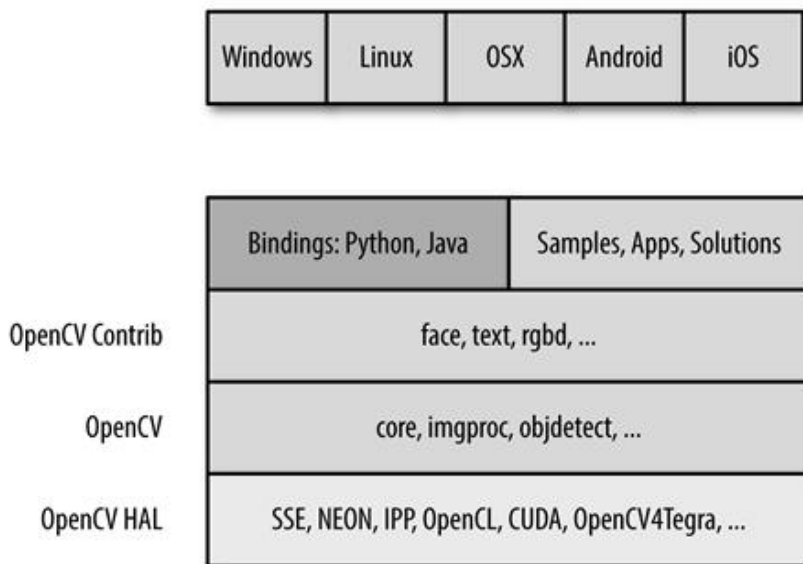


第 3 课 OpenCV 包含模块及组成结构

1.OpenCV 的组成结构

OpenCV 是由很多模块组成的，这些模块可以分成很多层：

- 1) 最底层是基于硬件加速层（HAL）的各种硬件优化。
- 2) 再上一层是 `opencv_contrib` 模块所包含的 OpenCV 由其他开发人员所贡献的代码，其包含大多数高层级的函数功能。这就是 OpenCV 的核心。
- 3) 接下来是语言绑定和示例应用程序。
- 4) 处于最上层的是 OpenCV 和操作系统的交互。



2.OpenCV 的具体模块

- 1) Core: 该模块包含 OpenCV 库的基础结构以及基本操作。
- 2) Improc: 图像处理模块包含基本的图像转换，包括滤波以及类似的卷积操作。
- 3) Highgui: 在 OpenCV 3.0 中，分割为 `imcodecs`、`videoio` 以及 `highgui` 三部分，这个模块包含可以用来显示图像或者简单的输入的用户交互函数。这可以看作是一个非常轻量级的 Windows UI 工具包。

- 4) **Video**: 该模块包含读取和写视频流的函数。
 - 5) **Calib3d**: 这个模块包括校准单个、双目以及多个相机的算法实现。
 - 6) **Feature2d**: 这个模块包含用于检测、描述以及匹配特征点的算法。
 - 7) **Objdetect**: 这个模块包含检测特定目标, 比如人脸或者行人的算法。也可以训练检测器并用来检测其他物体。
 - 8) **ML**: 机器学习模块本身是一个非常完备的模块, 包含大量的机器学习算法实现并且这些算法都能和 **OpenCV** 的数据类型自然交互。
 - 9) **Flann**: **Flann** 的意思是“快速最邻近库”。这个库包含一些你也许不会直接使用的方法, 但是其他模块中的函数会调用它在数据集中进行最邻近搜索。
 - 10) **GPU**: 在 **OpenCV** 中被分割为多个 **cuda*** 模块。**GPU** 模块主要是函数在 **CUDA** **GPU** 上的优化实现, 此外, 还有一些仅用于 **GPU** 的功能。其中一些函数能够返回很好的结果, 但是需要足够好的计算资源, 如果硬件没有 **GPU**, 则不会有什么提升。
 - 11) **Photo**: 这是一个相当新的模块, 包含计算摄影学的一些函数工具。
 - 12) **Stitching**: 本模块是一个精巧的图像拼接流程实现。这是库中的新功能, 但是, 就像 **Photo** 模块一样, 这个领域未来预计有很大的增长。
 - 13) **Nonfree**: 在 **OpenCV 3.0** 中, 被移到 **opencv_contrib/xfeatures2d**。**OpenCV** 包含一些受到专利保护的或者受到使用限制的(比如 **SIFT** 算法)算法。这些算法被隔离到它们自己的模块中, 以表明你需要做一些特殊的工作, 才可以在商业产品中使用它们。
 - 14) **Contrib**: 在 **OpenCV 3.0** 中, 融合进了 **opencv_contrib**。这个模块包含一些新的、还没有被集成进 **OpenCV** 库的东西。
 - 15) **Legacy**: 在 **OpenCV 3.0** 中, 被取消。这个模块包含一些老的尚未被完全取消的东西。
 - 16) **Ocl**: 在 **OpenCV 3.0** 中, 被取消, 取而代之的是 **T-API**。这是一个较新的模块, 可以认为它和 **GPU** 模块相似, 它实现了开放并行编程的 **Khronos OpenCL** 标准。
- 虽然现在模块的特性比 **GPU** 模块少很多, 但 **ocl** 模块的目标是提供可以运行在任何 **GPU** 或者是其他可以搭载 **Khronos** 的并行设备。这与 **GPU** 模块形成了鲜明的对比, 后者使用 **Nvidia CUDA** 工具包进行开发, 因此只能在 **Nvidia GPU** 设备上工作。