2023年7月12日　14:06

晶体管 光刻（硅 半导体）
光通过透镜聚焦到很小区域

1byte=8bit
1kb=2^10bit = 1024byte =1000b
1TB=1000GB
1GB=十亿字节=1000MB=10^6KB

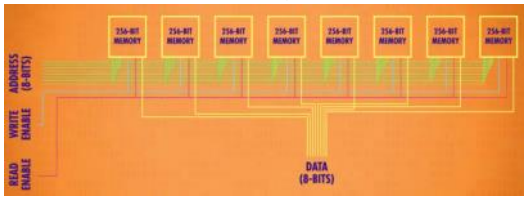
32 位与 64 位电脑的区别
32 位的最大数为 43 亿左右 32 位能表示的数字： 0——2的32次方-1， 一共2的32次方个数
64 位的最大数为 9.2*10^18

表示正负 1 是负， 0 是正（补码）
其余 31 位/63 位： 表示实数
浮点数=有效位数*指数
32 位数字中： 第 1 位表示正负， 第 2-9 位存指数。 剩下 23 位存有效位数
eg.625.9=0.6259（有效位数） *10^3（指数）

ASCⅡ
UNICODE 所有语言

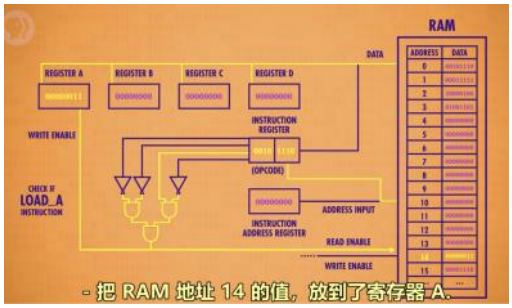
CPU
ALU 有 2 个单元， 1 个算术单元和 1 个逻辑单元 超前进位加法器

RAM（随机存取存储器） 寄存器 排成16*16门锁矩阵+74138 256位寄存器



256byte（8bit）
再抽象 矩阵嵌套

处理器 取指令fetch，解码decode，执行execute 看p7
含有一个寄存器在执行后暂存结果



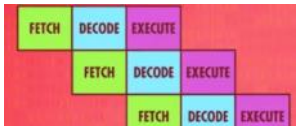
时钟 超频 加快时钟频率 降频 现代电脑自动调整

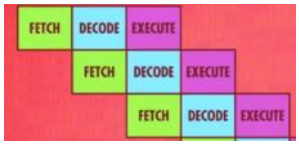
halt指令区分指令和数据
指令长度变化

早期通过加快晶体管速度， 来提升 CPU 速度。 但很快该方法到达了极限
后来给 处理器 设计了专门除法电路+游戏， 视频解码

为了不让 CPU 空等数据， 在 CPU 内部设置了一小块内存， 称为缓存， 让 RAM 可以一次传输一批数据到 CPU 中 cache hit
脏位： 缓存中与 RAM 不一致的数据 以便之后同步

流水线

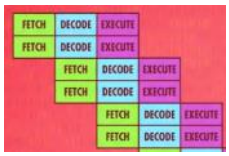




搞清数据的依赖关系 动态排序，乱序运行

一般遇到jump要停下来等待 高端cpu预测分支 提前猜并执行 猜错了就要清空流水线
猜的正确率90%

一次性处理多条指令 多个ALU



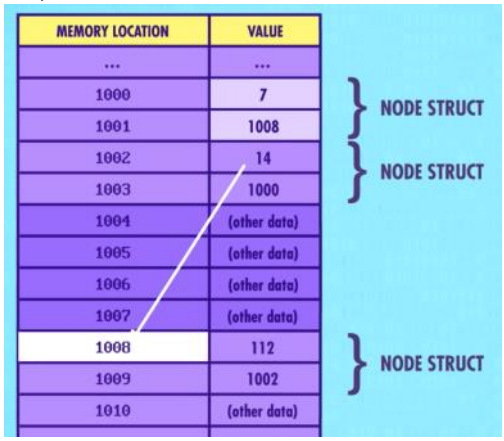
同时运行多个指令流（多核 CPU）
多核处理器： 一个 CPU 芯片中， 有多个独立处理单元。 但因为它们整合紧密， 可以共享一些资源

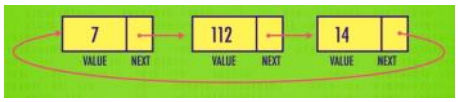
服务器 2-4个CPU
超级计算机 神威太湖之光 40960个CPU每个256核 无锡

高级编程语言→编译器→汇编码/机器码
编译器适配于不同硬件

排序
Dijkstra（图搜索 最短路径）

链表link list 存节点node （指针），指向下一个数，以null结尾
包括queueFIFO stackLIFO





前后双向指针 tree 最高：根；最低：叶 子；父节点
最多两个子节点 二叉树
树 根到叶是单向的 不能有循环

有循环叫图

图灵机：只要有足够的规则， 状态和纸带， 图灵机可以解决一切计算问题。 和图灵机一样完备， 叫做图灵完备

目前所有电脑都是图灵机

停机问题证明图灵机不能解决所有问题。

API
API控制哪些函数和数据让外部访问， 哪些仅供内部。

测试可以统称“质量保证测试”（QA）， 作用是找bug

beta版软件， 即是软件接近完成， 但没有完全被测试

过， 公司有时会向公众发布beta版， 以帮助发现问题。
alpha是beta前的版本， 一般很粗糙， 只在内部测试

VLSI软件 超大规模 自动设计芯片

进一步小型化会碰到的问题
1、 由于光的波长限制， 精度已到极限。
2、 量子隧穿效应： 当晶体管非常小、 电极之间可能只距离几个原子， 电子会跳过间隙， 会产生
漏电问题

OS

批处理
一个程序运行后会自动运行下一个程序。
多任务处理
操作系统能使多个程序在单个CPU上同时进行的能力， 叫做“多任务处理”

为了使所写程序和不同类型的电脑兼容， 我们需要操作系统充当软件和硬件之间的媒介， 更具体地说， 操作系统提供程序编程接口(API)来抽象硬件,叫“设备驱动程序”。 程序员可以用标准化机制， 和输入输出硬件（I/O） 交互

虚拟内存
多程序处理带来了一个程序所占用内存可能不连续的问题， 导致程序员难以追踪一个程序， 为了解决这个问题操作系统会把内存地址虚拟化， 这叫“虚拟内存"

动态内存分配
虚拟内存的机制使程序的内存大小可以灵活增减， 叫做“动态内存分配”， 对程序来说， 内存看上去是连续的

内存保护
给每个程序分配单独的内存， 那当这个程序出现混乱时， 它不会影响到其他程序的内存， 同时也能有效地防止恶意程序篡改其他程序， 这叫做内存保护

多用户分时操作系统（Multics）
用来处理多用户同时使用一台计算机的情况， 即每个用户只能用一小部分处理器， 内存等

Unix
把操作系统分成两个部分， 一个是操作系统的核心部分， 如内存管理， 多任务和输入/输出处理， 这叫做“内核”， 第二部分是一堆有用的工具， 比如程序和运行库

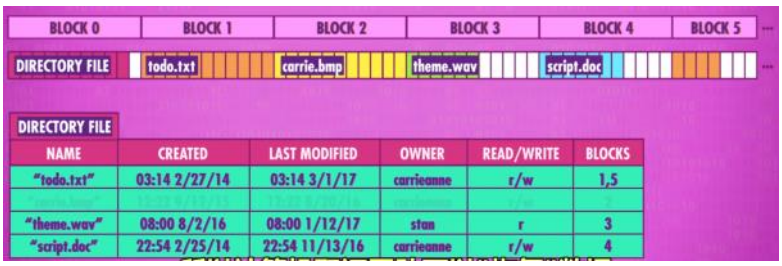
内存 memory
外存 storage

内存层次结构
在计算机中， 高速昂贵和低速便宜的内存混合使用以取得一个平衡
光盘
原理： 光盘表面有很多小坑， 造成光的不同反射， 光学传感器会捕获到， 并解码为 1 和 0
固定硬盘（SSD）
里面是集成电路

元数据metadata 描述数据的属性
存在文件头 header

DIRECTORY FILE

NAME	CREATED	LAST MODIFIED	OWNER	READ/WRITE	BEGIN	LENGTH
"todo.txt"	03:14 2/27/14	03:14 3/1/17	carrieanne	r/w	10	8
"carrie.bmp"	12:22 9/12/15	12:22 8/20/16	carrieanne	r/w	18	13
"theme.wav"	08:00 8/2/16	08:00 1/12/17	stan	r	31	6
"script.doc"	22:54 2/25/14	22:54 11/13/16	carrieanne	r/w	37	8

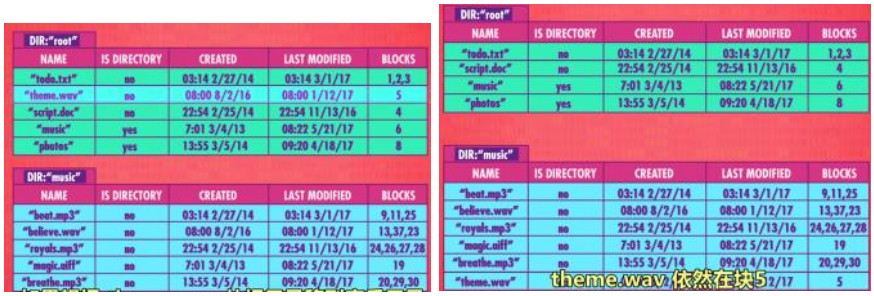


分块，方便改动

超出block1界限 开block5

Block2种内容删除 仅在dir种删除记录，但是block2内容在被覆盖前仍然存在，因此警察可以恢复数据

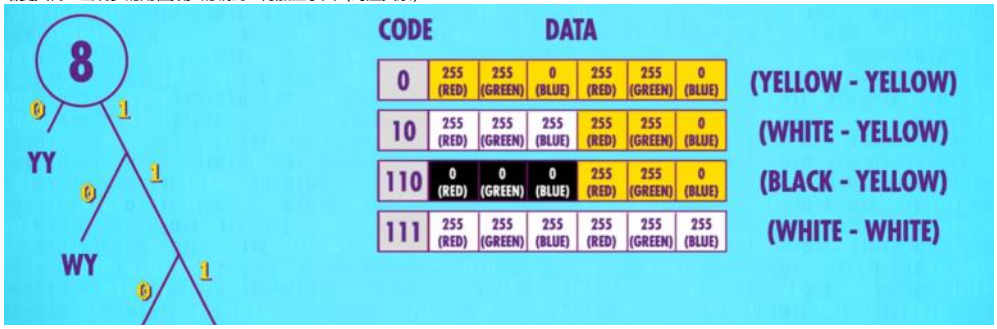
计算机自动碎片整理

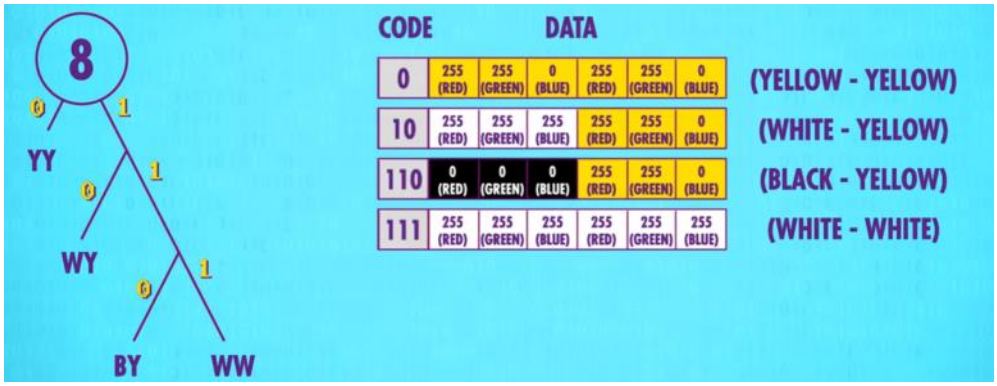


移动文件仅仅移动dir中的位置信息，不移动文件存储的block

无损压缩

哈曼夫树：出现多的用位数少的编码　再加上字典（对应关系）





感知编码
删掉人类无法感知的数据的有损压缩方法， 听不到或看不到
如通话卡顿

时间冗余（视频）
视频，人讲话背景不变，背景一直不动；人动手，进阶的视频压缩模式会找到帧与帧的相似性，然后打补丁。很多视频编码格式， 只存变化的部分。 因而有时候视频乱码

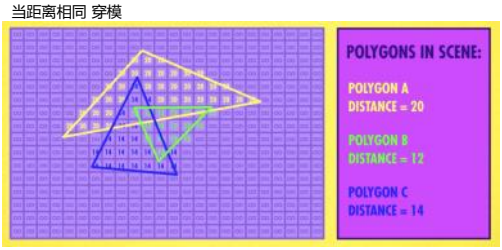
液晶显示器LCD
光栅扫描

显卡 有VRAM 01->显示图案

解释器和编译器类似， 区别是解释器运行时转换， 而编译器提前转换
兼容外接硬件的机器都叫IBM 兼容
微软的dos操作系统
Apple-II，苹果选封闭架构， 只有苹果在非"IBM 兼容"下保持了足够市场份额

出现鼠标 GUI
GUI是"事件驱动编程"， 而不像传统代码一样自上而下
创立了桌面， 窗口，剪切 复制 黏贴 windows
Macintosh 成功
Windows 95 提供新的图形界面， 并有Mac没有的新功能， 如多任务和受保护内存
史蒂夫·乔布斯被赶出自己的公司，因为无人为苹果开发软件
微软做失败的 Microsoft Bob——类似于房子的设计

线框渲染 算法 负责把3D坐标"拍平"显示到2D屏幕上
算法 负责减少3D图复杂度减少计算量减少掉帧
3D 用三角形构成立体图
扫描线渲染算法 用三角形填充图形
计算机图形学
抗锯齿 三角形边缘像素颜色浅一点
遮挡 画家算法 排序算法从远到近渲染
深度缓冲算法 初始为无限大
当距离相同 穿模



面剔除 Back Face Culling
由于游戏角色的头部或地面， 只能看到朝外的一面， 所以为了节省处理时间，会忽略多边形背面

表面法线 平面着色
基本的照明算法， 缺点是使多边形边界明显， 看上去不光滑
后续改进算法

纹理映射算法

GPU在显卡上，周围有专用的RAM

局域网LAN 以太网

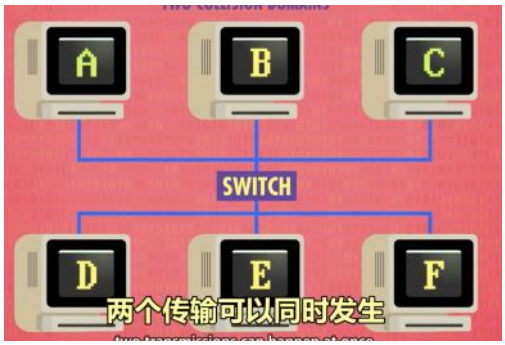
MAC地址 每个计算机唯一
用于确认局域网和WiFi传输的对象
wifi连接家里所有设备 形成局域网

载波侦听多路访问CSMA
多台电脑共享一个传输媒介， 共享媒介又称载体， 如WiFi的载
体是空气， 以太网的载体是电线。 载体传输数据的速度叫带宽

指数退避 Exponential Backoff
当多台计算机同时想要传输数据时， 就会发生冲突， 当计算机检测
到冲突 就会在重传之前等待一小段时间， 这一段时间包括固定时间+
随机时间， 再次堵塞时固定时间将会指数级增加

冲突域
载体和其中的设备总称为"冲突域"， 为了避免冲突， 可以用交换器
电路直接连接 银行或军队

交换机连接两个小网络 互联网是这样连接的



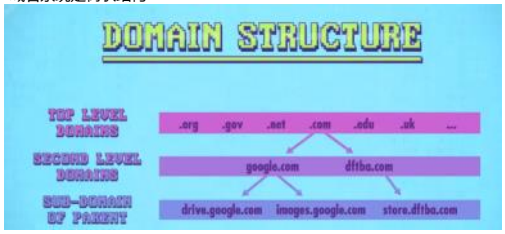
报文交换 路由
报文的具体格式简称IP， 每一个电脑都会有一个IP地址
好处， 可以用不同路由， 通信更可靠也更能容错。
坏处， 当报文比较大的时候， 会堵塞线路。 解决方法是 将大报文分
成很多小块， 叫"数据包"， 这叫"分组交换"。 路由器会平衡与其他
路由器之间的负载 以确保传输可以快速可靠， 这叫"阻塞控制"消息
沿着路由跳转的次数 叫"跳数"(hop count)， 看到哪条线路的跳数很
高， 说明出了故障， 这叫跳数限制

局域网再连到广域网（WAN）， 广域网的路由器一般属于你的互联网服
务提供商（ISP）， 再连更大的WAN， 往复几次， 最后连到互联网主
干，由大型高带宽路由群组成，再走到youtube服务器

IP最底层协议 负责把数据包送到正确的计算机
UDP负责把数据包传送到正确的程序， 有端口号（哪个程序）， 校验和（数据是否损坏）
检验 数据求和 低16位存储在header 发过去后检验一遍
UDP不确定数据是否到达，损坏直接丢弃 适合视频会议等 简单快速

所有数据必须到达 TCP协议
发送的文件到达后排序 删掉重复数据包
要求接收方确认无误后发送确认码（ACK）没收到ACK再发一次，并行传送， 确认码的成功率和
来回时间可以用来推测网络的拥堵程度， TCP可以根据这个调整传输率。 由于这个特点， TCP对
时间要求高的程序不适用

网址 ip：port
DNS服务器 - 域名系统 把域名和IP地址一一对应 供应商提供
域名系统是树状结构



OSI - 开放式系统互联通信参考模型
线路电信号 无线网无线信号 物理层

数据链路层 控制物理层 包括MAC地址 碰撞检测 指数退避等

网络层 报文交换 路由

传输层 UDP TCP

会话层

万维网 指web网页 基于互联网的一种程序
URL 网页的唯一网址
TCP协议访问.com
HTTP协议 在.com基础上访问网页 .com/courses
浏览器把网页渲染出来
HTML写网页的语言

```
<h1>Ode to Worf</h1>
Worf, son of Mogh, was born on the
<a href=http://www.kli.org> Klingon </a> homeworld
in 2340. Following his graduation from
Starfleet Academy in 2361, he became the
first Klingon warrior to serve in the
Federation.

<h2>Why is he so awesome?</h2>
<ol>
<li>Super Strong</li>
<li>Very Loyal</li>
<li>Good Friend</li>
<li>Amazing with <a href=http://en.wikipedia.org/
wiki/Bat'leth> Bat'leth</a></li>
</ol>
```

层叠样式表CSS javascript

搜索引擎 Google
改进排序方法， 按照 链接指向的多少来排序网站的优劣

网络中立性
平等地对待各个网站的数据包

计算机安全

保密性： 只有有权限的人， 才能读取计算机系统和数据
破解密码
完整性： 只有有权限的人， 才能使用和修改系统和数据
向服务器发送大量假请求，使其变慢或挂掉
可用性： 有权限的人， 可以随时访问计算机系统和数据

威胁模型分析

能力如何， 目标可能是什么， 可能用什么手段， 攻击手段又叫"攻击矢量"
"威胁模型分析"让你能为特定情境做准备， 不被可能的攻击手段数量所淹没。
很多安全问题可以总结成两个：你是谁？ 你能访问什么？

身份验证（Authentication）的三种方式：

What you know, 你知道什么 用户名和密码
What you have, 你有什么 指纹 虹膜扫描 问题：无法reset 别人拿到你的指纹你一辈子麻烦了
What you are, 你是什么

访问控制 Access Control

Bell LaPadula model 不能向上读取， 不能向下写入
机密权限可以读公开内容，不能写公开内容，防止机密泄露到公开文件

Malware 恶意软件

安全内核
安全内核应该有一组尽可能少的操作系统软件， 和尽量少的代码，因为往往是执行时出错

独立安全检查和质量验证
最有效的验证手段，开源代码让安全专家来检查，安全大会DEF CON

隔离 Isolation，沙盒 Sandbox
当程序被攻破后，如何限制损害， 控制损害的最大程度， 并且不让它危害到计算机上其他东西， 这叫"隔离"。

要实现隔离， 我们可以"沙盒"程序， 给每个程序独有的内存块， 其他程序不能动。
一台计算机可以运行多个虚拟机， 如果一个程序出错， 最糟糕的情况是它自己崩溃， 或者搞坏它处于的虚拟机。

黑客中的白帽子:好人 帮助修复漏洞
黑帽子：坏人

欺骗别人获得信息， 或让人安装易于攻击的系统
钓鱼邮件
假托电话、
木马 邮件附件里的照片或者发票实际是恶意软件 偷数据
或者勒索软件，加密你的文件，交赎金才解密

暴力尝试所有密码 密码错误等待时间解决这一问题
NAND镜像 需要实物接线复制内存，等待时间时覆盖内存，解决等待时间，iPhone5c可用

远程攻击 利用互联网

漏洞利用获取权限 如内存中缓冲区溢出（输入很长的用户名和密码）
黑客找到is_admin在哪并且设置为true
边界检查 Bounds Checking和金丝雀
——防止缓冲区溢出的手段， 金丝雀， 留出一些不用的空间， 当空间变少时， 说明有攻击者乱来。

代码注入 Code Injection 攻击使用数据库的网站
输入用户名和密码，发送到服务器sql数据库检查是否正确

分号结束语句，drop table users 删除所有现有账户名密码

同理添加一个新管理员账户，或让数据库泄露数据

因此拒绝输入分号括号等，或者删除特殊字符再执行

零日 漏洞 Zero Day Vulnerability
当软件制造者不知道软件有新漏洞被发现了， 这个漏洞被称为“零日 漏洞”

计算机蠕虫 Worms 僵尸网络
如果有足够多的电脑有漏洞， 让恶意程序可以在电脑间互相传播， 这种恶意程序叫做蠕虫
用于发送大量垃圾邮件，用别人电脑算力和电费挖比特币
发起ddos(发送大量垃圾信息) 攻击服务器

密码学

1977年"数据加密标准"（DES）
56位 之后算力发展容易遍历
2001年"高级加密标准"（AES）
128位192位256位 至今所有计算机都遍历不了

迪菲-赫尔曼密钥交换 对称加密
模幂计算 公共信息B和M
张三选一个X 计算B的X次方÷M的余数发给李四 李四拿到之后再算Y次方
李四选一个Y 计算B的Y次方÷M的余数发给张三 张三拿到之后再算X次方
两人未交换密钥的情况下都得到了B的XY次方÷M的余数
这些数字很大，以此为密钥使用AES加密

非对称加密
有公钥的只能加密不能解密 私钥才可以解密
最有名的非对称加密算法是RSA

也可以私钥加密公钥解密 证明数据来自权威加密者 如服务器

访问网站时都用到私钥加密公钥解密 随后双方对称加密

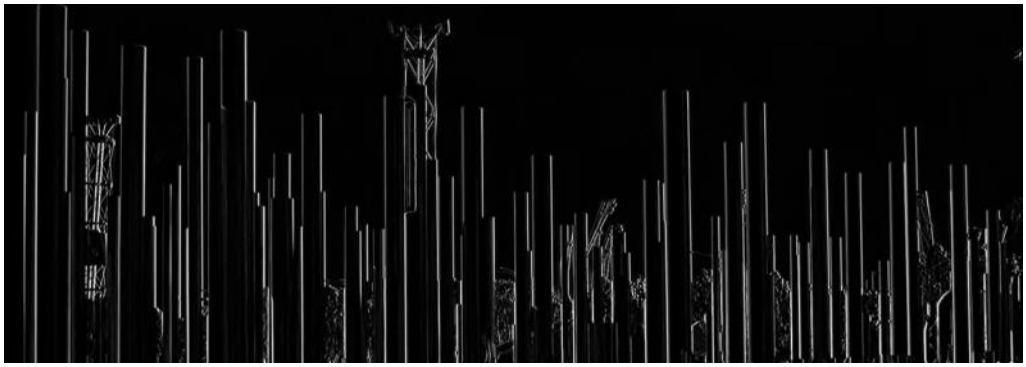
基于统计学

决策树
支持向量机 Support Vector Machines
本质上是用任意线段来切分决策空间， 不一定是直线

基于神经网络

强化学习 Reinforcement Learning
学习什么管用， 什么不管用， 自己发现成功的策略， 这叫强化学习。

垂直边缘敏感的卷积核，遇到边界算出来值大，其他小



水平边缘敏感的卷积核

二者统称prewitt算子

核/过滤器 kernel or filter

其他 锐化 模糊算子

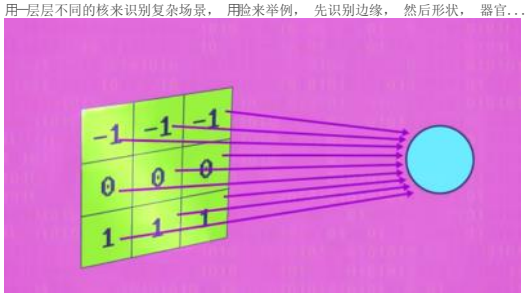
找线段，找对比色包裹的算子

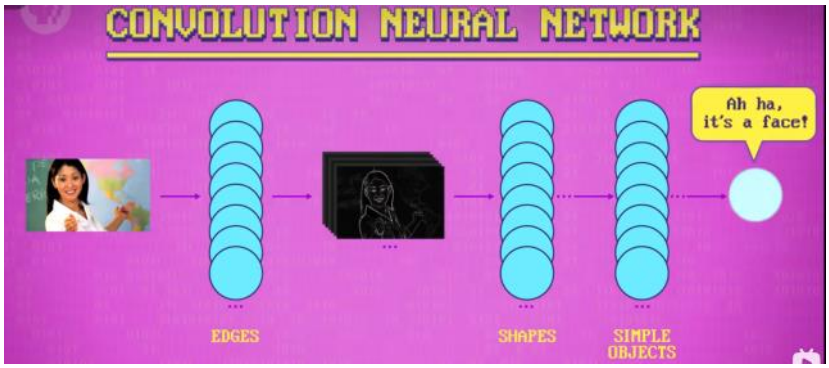
识别鼻子 眼睛的算子 组合成人脸检测算法

卷积神经网络CNN

算子值随着学习变化

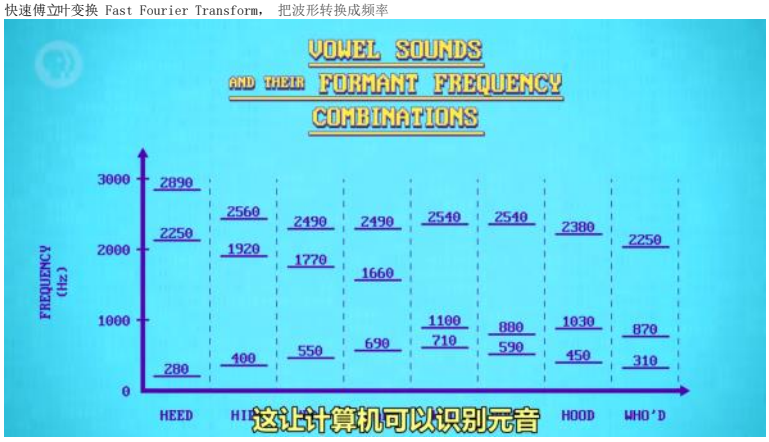
用一层层不同的核来识别复杂场景， 用脸来举例， 先识别边缘， 然后形状， 器官... 直至某一层把所有特征堆积在一起， 识别出脸之后， 可以进一步用其他算法定位面部标志， 如眼睛和眉毛具体位置， 从而判断心情等信息

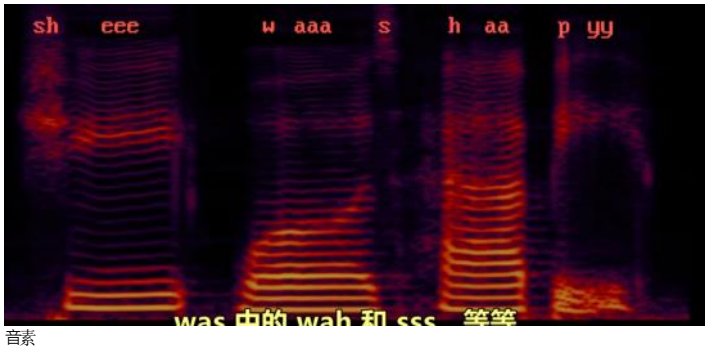




通过词性 Parts of speech和短语结构规则 Phrase structure rules构建分析树 Parse tree， 并结合语言模型 Language Model 来实现语音识别 Speech recognition

快速傅立叶变换 Fast Fourier Transform， 把波形转换成频率





音素

英语有44种音素 音素识别

语音合成

计算机发出声音

机器人控制的回路

负反馈回路

比例-积分-导数控制器PID 控制器通过控制三个值， 比例值——实际值和理想值差多少， 积分值——一段时间误差的总和， 前两者用来修正错误； 导数值（微分值） ——期望值和实际值之间的变化率， 用来避免未来的错误，这也叫预期控制， 来控制进程。

有智力有武器的机器人

机器人三定律

——让机器人不要伤害人类

计算机心理学

易用度vs专业度 二者兼得

颜色强度排序和颜色排序

人类擅长给颜色强度排序， 所以颜色强度很适合现实连续值； 而人类不擅长给颜色排序， 所以如果数据没有顺序，用不同颜色就很合适， 如分类数据。

直观功能 认出大于回想

Affect 计算机情商

Facebook的研究，post内容影响人积极or消极

CMC

计算机介质通信(Computer-mediated communication,CMC)是指使用计算机进行交流的过程，介质包括了文字、图像、语音、短信等形式

增强凝视：网课矫正老师面部，使得似乎看着网课学生而不是教室学生，增强参与感（视频会议同理）

HRI， 人类和机器人或计算机交互

教育科技

判断规则和选择算法， 组合在一起成为域模型

贝叶斯知识追踪

把学生的知识掌握当成隐藏变量， 根据学生答题的正确度， 更新学生掌握程度的估算值。
贝叶斯知识追踪有一组方程， 会用这四个概率， 更新学生模型， 评估其掌握程度。
学生已经学会的概率瞎猜的概率
失误的概率
做题过程中学会的概率

自 适应性程序

一种算法， 选择适合学生的问题， 让学生学。

教育数据挖掘

看学生答题时停顿的时间， 观察学生停顿和加速视频的时间段， 看论坛互动， 来评估学生的程度。

虚拟教学助手的研究

奇点 Singularity

——智能科技的失控性发展

非重复性思维工作