**人工智能课程报告**

畅雅雯 2015202004

我的课程报告将会从以下两个方面展开，首先，介绍在我们小组的AI Car制作中，我所担任的角色和负责的任务，并结合实际制作AI Car过程中遇到的问题和解决方法，谈谈自己的心得。其次，针对老师上课时重点讲到的卷积神经网络（Convolutional Neural Networks，简称CNN），我想讲讲自己对CNN里面一些概念的理解以及总结CNN与其他方法相比，有哪些优势。

**首先，先来讲讲关于我们小组的**AI Car**的制作吧。**

在我们小组AI Car制作的前一阶段（即前两次展示）中，其实我们没有特别明晰的分工，组装小车一般都是大家在一起相互讨论摸索着做的，其中也遇到了很多问题。当Arduino板还没有连接入小车时时，我们纯靠电池给马达供电，想看看小车的运动状态，发现小车的运动十分不稳定，处于一种随机的时走时停的状态，此外，小车也无法顺利的走直线，一直在打转。后来经过不断的调整才发现，小车不能持续走的原因是由于电池电量不够，我们换了充满电的电池之后，小车的运行状态就相对稳定了。而小车无法直走，则是由于小车两个轮子的马达驱动力不一致，因为我们小车的材料都是多买了一份作为后备的，所以我们首先是决定换掉驱动力较弱马达，可是这些马达都做得很不良心，我们没能找到驱动力相差不多的两个马达，最终还是决定后期通过修改导入Arduino板的程序来促使两边轮子的转速相同。

在小车能稳定动起来之后，下一步工作就是把Arduino板接入小车了。因为对Arduino板的各个接线口功能不是很熟悉，所以这个过程其实不是很顺利。当我们把避障和控制小车运动的程序导入Arduino板之后，发现小车完全就不动了。我们再三确定导入的程序没问题之后，就开始在小车的接线上找问题，在调整了接线确保接线是对了的之后，小车还是不能运动。后来我们把Arduino板上的输入端口从2改到3，又修改了相应程序之后，发现小车就能按照指示行走和避障了，原来可能是我们Arduino板上的端口2有问题。

在前两次的展示中，主要是由水治禹同学和任可欣同学负责写要上传到github上的文档以及第一次上台展示，我和姚菁同学负责准备PPT和第二次的上台展。

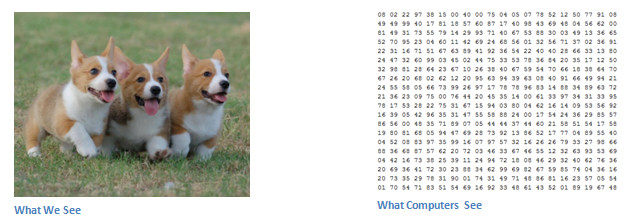
在AI Car制作的后一阶段（即第三次展示）中，我们有了明确的分工，我和任可欣同学负责完成小车循迹，并实现在循迹过程中同时避障的功能，水治禹同学和姚菁同学则负责完成图像识别功能。起初，在将小车循迹&避障功能的程序导入Arduino板里，将相应的传感器连接完毕，并且用胶布铺好“轨道”之后，我们发现小车并不按照我们铺好的轨道行走，而传感器的感应灯也不怎么亮。但是，我们把手放在底盘下面的感应器上，小车就能感应到，并顺着我们的手进行移动。于是我们猜测是小车底盘太高，传感器不够灵敏，导致其感应不到地上的“轨道”。于是我们找来螺丝刀，开始调整控制传感器灵敏度的小螺丝。因为一点点的转动，都会让传感器的灵敏度产生很大的变化，我们很担心会调过头，因此很小心，每次调整的幅度都很小，力求找到那个临界点。终于，皇天不负有心人，小车的传感器终于能够感受到我们铺设的轨道了，但是紧接着新的问题又出现了，我们的小车虽然能循迹，可是很不稳定，一会能按着轨迹走，一会又脱离掌控。正在我们苦于找不到小车不能稳定循迹的原因时，偶然发现，每次小车一运动到宿舍楼道两边铺设的黑色地板砖时，就变得十分稳定，而我们使用的是红色塑胶胶带铺设的轨道，我们就怀疑可能是轨道颜色的问题，传感器对不同颜色的灵敏度不同。于是，抱着试一试的态度，我们又买了黑色塑胶胶布，重新铺设了轨道，发现小车的循迹真的就稳定多了，这让我们很是惊喜。

小车的制作也已接近尾声，通过这次AI Car的制作，不但增强了自己的动手能力，还让我对Arduino编程也有了一定的了解，感觉Arduino编程其实不是很难。有机会真的可以去买一些材料，尝试老师说的，自己组装做一个小型的扫地机器人，也算是学以致用啦。

**下面我将谈一谈自己对于卷积神经网络（Convolutional Neural Networks，简称CNN）的理解，这里只着重谈一些想法上的东西，不涉及到算法层面。**

20世纪60年代，Hubel和Wiesel在研究猫脑皮层中用于局部敏感和方向选择的神经元时发现其独特的网络结构可以有效地降低反馈神经网络的复杂性，继而提出了卷积神经网络（Convolutional Neural Networks，简称CNN）。现在，CNN已经成为众多科学领域的研究热点之一，特别是在模式分类领域，由于该网络避免了对图像的复杂前期预处理，可以直接输入原始图像，因而得到了更为广泛的应用。

在讲图像识别之前，我们先要了解计算机视角。

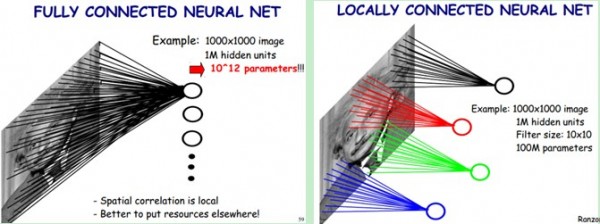


上图中的左边就是以人类视角看到的一张画有三只小狗的图片，但计算机看到的就是一连串的像素值，即右图所示。

在图像处理中，往往把图像表示为像素的向量，比如一个500×1000的图像，可以表示为一个1,000,000的向量。在一般的神经网络中，如果隐含层数目与输入层一样，即也是1,000,000时，那么输入层到隐含层的参数数据为1000000×1000000=10^12，虽然计算机的计算能力强大，但参数太多势必会影响效率，而且参数太多，做出来的模型泛化程度低，识别能力反而降低，就会出现过拟合(over-fitting）问题。而CNN则提供了两种有效的方法来减少参数数量，局部感知野和权值共享。

* **局部感知野**

局部效应在现实生活中普遍存在，在计算机的I/O设计上就充分考虑到了空间上的局部效应，而图像的空间联系也是局部的像素联系较为紧密，而距离较远的像素相关性则较弱。因而，每个神经元其实没有必要对全局图像进行感知，只需要对局部进行感知，然后在更高层将局部的信息综合起来就得到了全局的信息。在BP神经网络中，每一层的神经元节点是一个线性一维排列结构，层与层各神经元节点之间是全连接的。卷积神经网络中，层与层之间的神经元节点不再是全连接形式，利用层间局部空间相关性将相邻每一层的神经元节点只与和它相近的上层神经元节点连接，即局部连接。这样大大降低了神经网络架构的参数规模。如下图所示：左图为全连接，右图为局部连接。

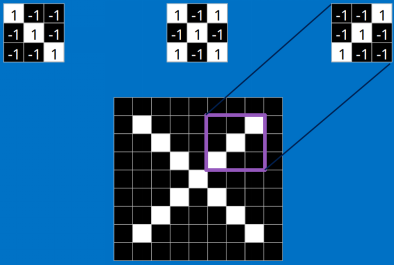
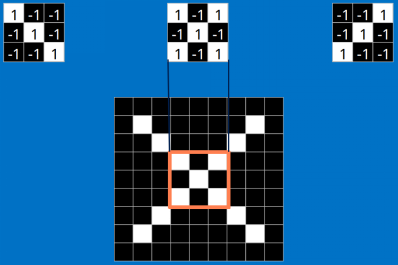
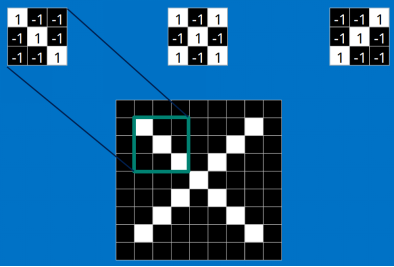


* **权值共享**

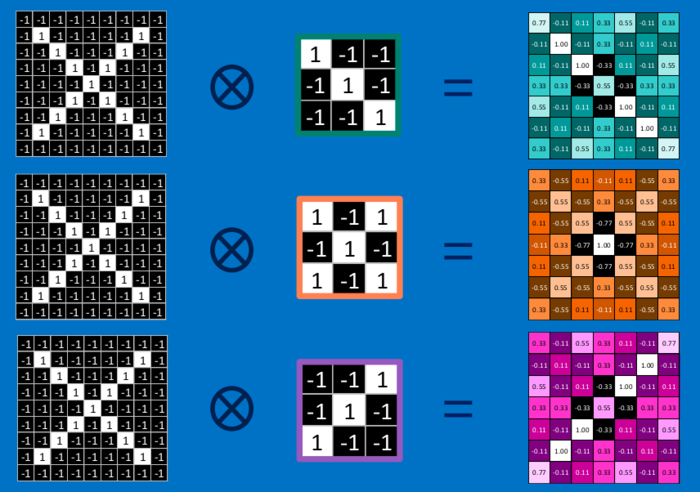
在卷积神经网络中，卷积层的每一个卷积滤波器重复的作用于整个感受野中，对输入图像进行卷积，卷积结果构成了输入图像的特征图，提取出图像的局部特征。每一个卷积滤波器共享相同的参数，包括相同的权重矩阵和偏置项。共享权重的好处是在对图像进行特征提取时不用考虑局部特征的位置。而且权重共享提供了一种有效的方式，使要学习的卷积神经网络模型参数数量大大降低。

**为了便于理解以上两个概念，我在网上找到了一个很好的例子，来解释说明：**

下图中红色圆圈标注的三个3×3像素矩阵即为对9×9原始图进行局部感知之后获得的三个**卷积滤波器**。



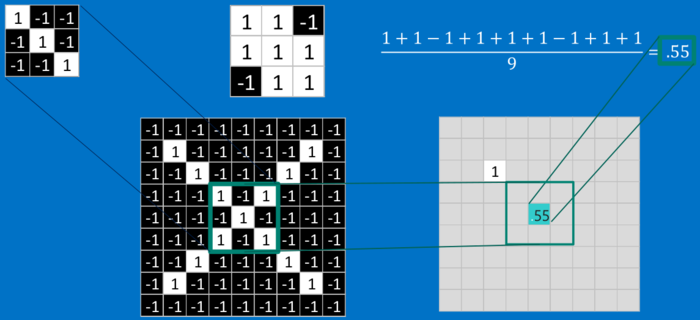
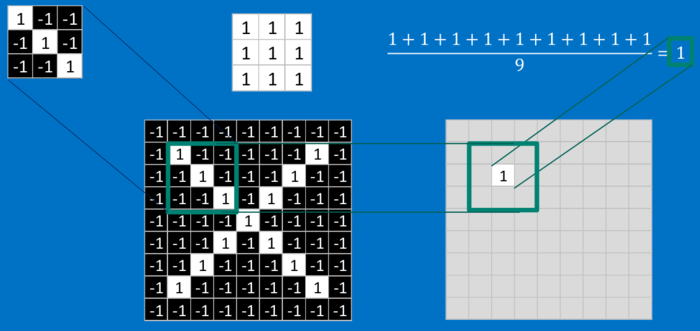
下面对每一个卷积滤波器和原图都进行一次卷积操作，得到右边7×7的矩阵



**原始图**

**特征图**

注：所谓的**卷积操作**就是将两个小块内对应位置的像素值进行乘法运算，然后将整个小块内乘法运算的结果累加起来，最后再除以小块内像素点总个数即可。见以下示意图：



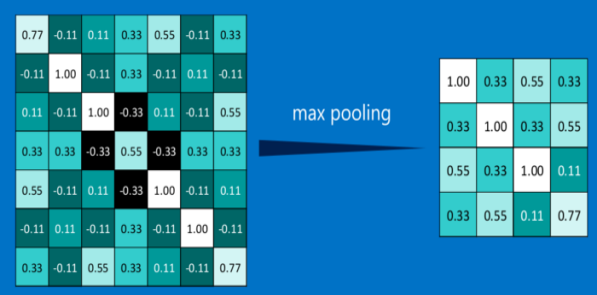
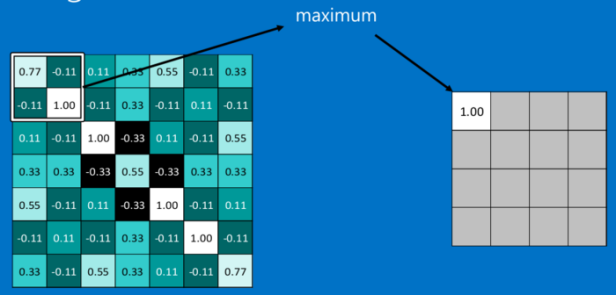
通过上述两步我们已经有效减少了参数数量。对于9×9 像素的原始图像，如果不用CNN方法，将会得到9×9×9×9=2401个参数，而如果按CNN的思路，我们通过学习得到了3个定义在3×3输入上的特征（即卷积过滤器），每一个特征和图像卷积都会得到一个 (9 − 3 + 1) × (9 − 3 + 1) = 49 维的卷积特征，由于有 3个特征，所以原始图像就会得到一个 49 × 3 = 147 个参数。

但是在我们的上述举例中，原始图像很小，只有9×9的像素，而且由于图像简单，所以我们学习得到的特征也少，只有3个。试想我们如果用100×100的原始图像，且该图像较为复杂，我们学习得到了200个特征，则即便通过上述计算方法，我们仍将会得到超过160万的参数，学习一个拥有这么多参数输入的分类器其实是很困难的，而且很容易出现过拟合 (over-fitting)。

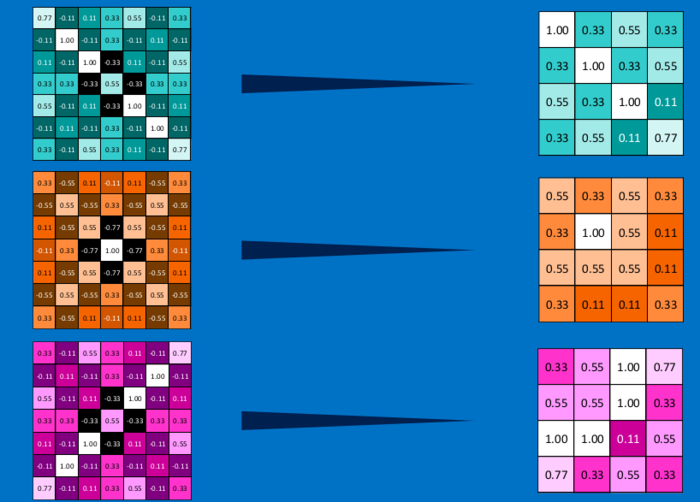
为了再一次减少参数，CNN提供了另一个有效的工具——“**池化(Pooling)**”。池化就是将输入图像进行缩小，减少像素信息，只保留重要信息。通常情况下，池化都是2\*2大小，比如对于max-pooling来说，就是取输入图像中2\*2大小的块中的最大值，作为结果的像素值，相当于将原始图像缩小了4倍。(注：同理，对于average-pooling来说，就是取2\*2大小块的平均值作为结果的像素值。)从上述描述中不难看出，其实，池化也是利用了像素的局部原理，对不同位置的特征进行聚合统计。

为了便于理解池化的含义，我们依旧使用上述举例（即：将9×9的原始图像经过卷积后得到的7×7的特征图）。

下面的例子中展示的是max-pooling：



然后对所有的特征图均执行max-pooling操作，得到如下结果：



因为最大池化（max-pooling）保留了每一个小块内的最大值，所以它相当于保留了这一块最佳的匹配结果（因为值越接近1表示匹配越好）。这也就意味着它不会具体关注窗口内到底是哪一个地方匹配了，而只关注是不是有某个地方匹配上了。如此，CNN能够发现图像中是否具有某种特征，而不用在意到底在哪里具有这种特征。此外，它减小了来自上层隐藏层的计算复杂度。这些池化单元也具有平移不变性，即使图像有小的位移，提取到的特征依然会保持不变。由于增强了对位移的鲁棒性，最大池采样方法是一个高效的降低数据维度的采样方法。

最后，我们来总结一下CNN的优点。

CNN是一种深度的监督学习下的机器学习模型，具有一些传统技术所没有的优点：具有极强的适应性，良好的容错能力、并行处理能力和自学习能力，可处理环境信息复杂，背景知识不清楚，推理规则不明确情况下的问题。善于挖掘数据局部特征，提取全局训练特征和分类，允许样品有较大的缺损、畸变，运行速度快，自适应性能好，具有较高的分辨率。通过结构重组和共享权值将特征抽取功能融合进多层感知器，省略识别前复杂的图像特征抽取过程，在模式识别各个领域都取得了很好的成果。

注：上述例子参考自网站：http://www.jianshu.com/p/fe428f0b32c1