无人驾驶技术入门（十六）
初识深度学习之交通标志分类

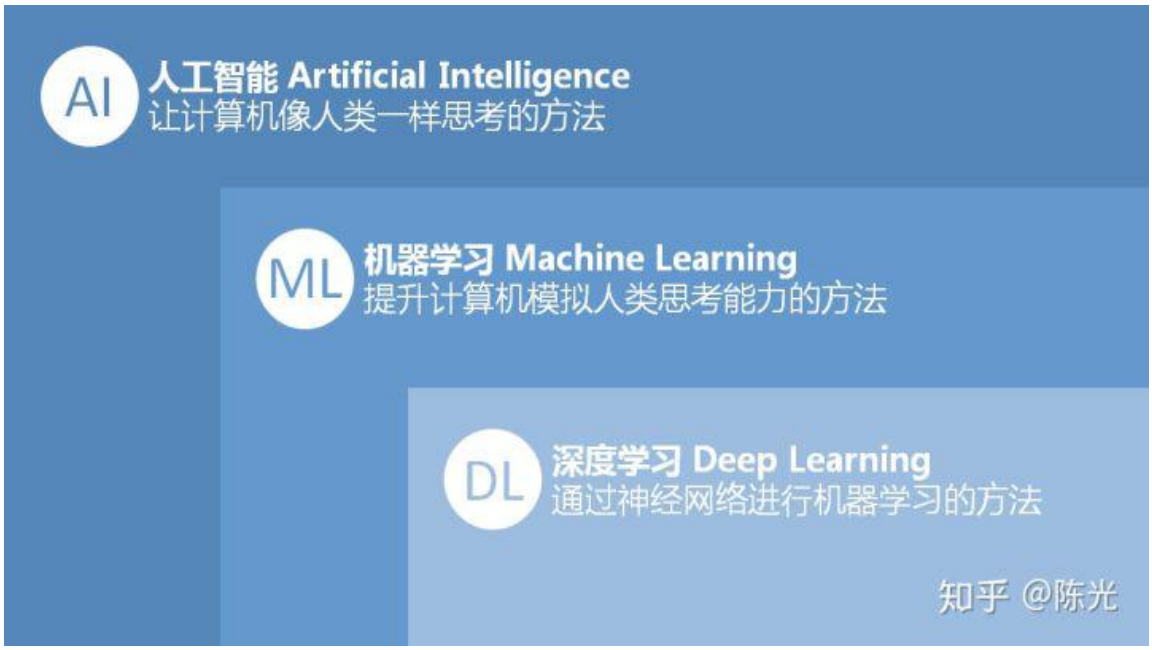
陈 光

在上两期的《无人驾驶技术入门》中，我以车道线检测为例，介绍了计算机视觉领域一些基本的算法。应用图像处理算法和调试算法阈值，就能实现车道线的检测和跟踪。

车道线检测、跟踪的项目，主要是通过设置 ROI（感兴趣区域）、调试算法阈值，通过人为设定规则的方式实现车道线检测。随着人工智能技术的发展，近几年在图像处理领域越来越多地采用深度学习的方式进行图像中物体的识别。使用深度学习的方法识别图像，不仅性能更为鲁棒，而且相比于设定规则的方式，识别率更高。

在本次分享中，我将以优达学城(Udacity)无人驾驶工程师学位中提供的交通标志牌分类项目为例，介绍深度学习相关的入门知识。

先通过一张简单的图，认识工智能、机器学习和深度学习的关系。由图可以看出深度学习是机器学习的一个分支，机器学习又是人工智能的一个分支。



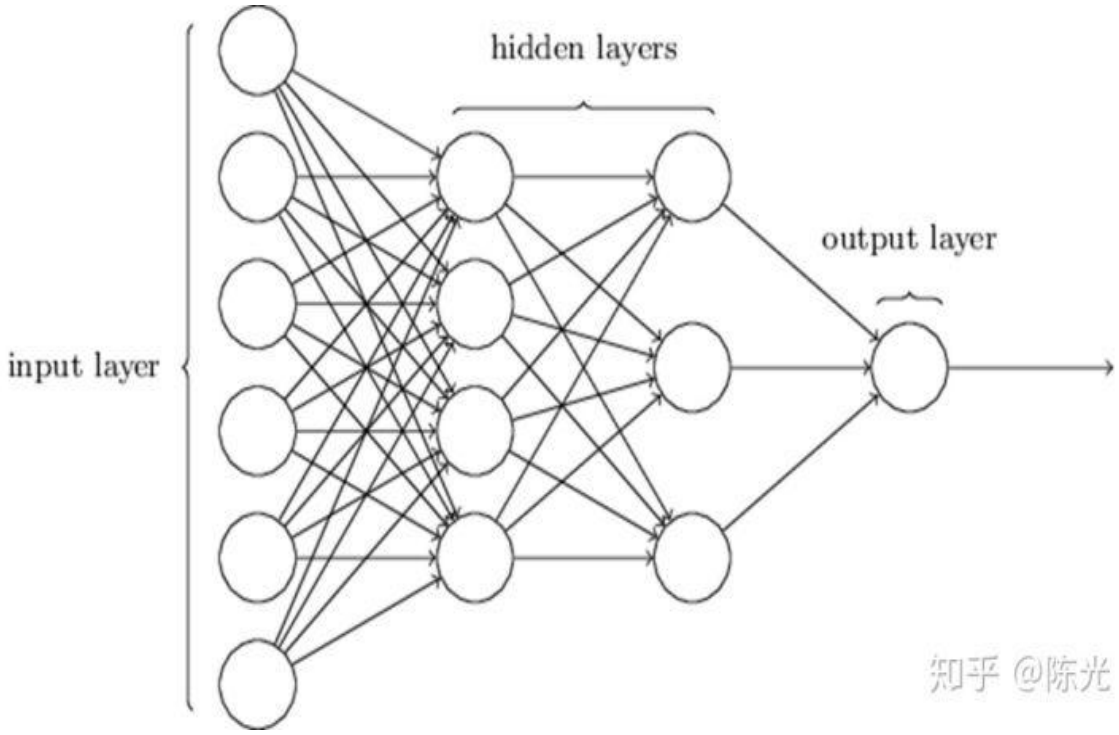
图片出处：http://www.primeton.com/read.php?id=2378

人工智能最早可以追溯到上个世纪五十年代，受制于当时的计算能力，人工智能技术并没有得到很好地发展；直到上世纪八十年代，计算机算力的大幅提高，人工智能才得以蓬勃发展，继而衍生出了机器学习技术，机器学习的出现，帮助人类解决了很多诸如垃圾邮件分类、房价估计等简单问题，也辅助解决图像识别等复杂问题，但准确度未能达到预期。直到深度学习（通过的神经网络进行机器学习）技术的出现以及并行计算技术的加持，使得图像识别等复杂问题的准确度得到了大幅提升，一举超越了人类识别的水平。越来越多的科学工作者、工程人员和资本投入到了深度学习领域。

人工智能主要是为了解决预测（回归）和分类两大问题。在生活中，预测的例子有很多，比如根据房屋面积等信息预测房屋的价格，或是根据前几年的销售额，预测今年的销售额等。分类的问题也有很多，比如判定股票的涨跌，图像中的物体（比如手写数字、字母）的识别等。

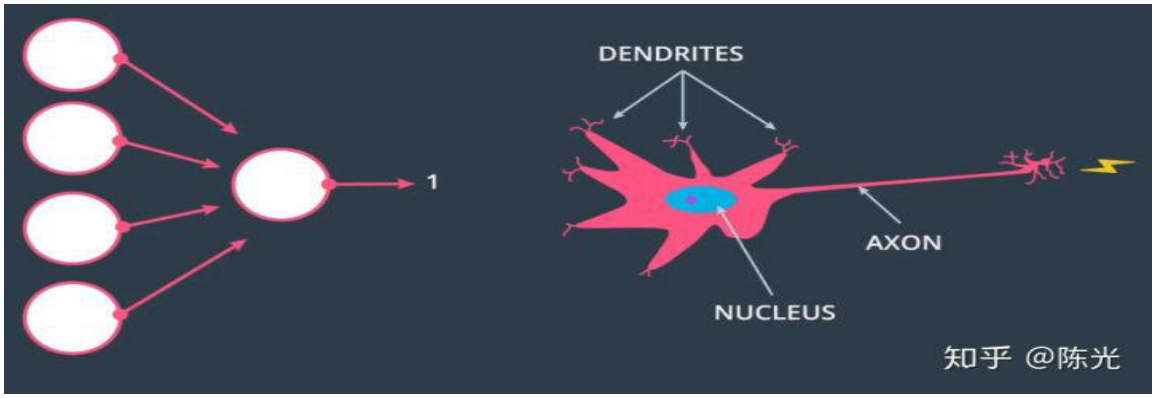
1 认识神经网络

提到神经网络时，我们总会看到如下由圆圈和线组成的网络，下面说一下这样绘制的原因。



图片出处：https://chatbotslife.com/how-neural-networks-work-ff4c7ad371f7

人类的神经元通过多个树突接收数据，经过处理后，将信号通过轴突发出，与上述结构十分相似，因此我们搭建的网络图也被称为神经网络。

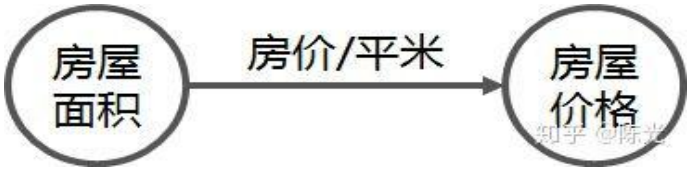


图片出处：优达学城(Udacity)无人驾驶工程师学位

通过一个房价计算的例子，解释一下这里的圆圈和线段。

在一个地区，决定一个房子最直接因素就是房子的面积，面积越大，房子的价格就越高。

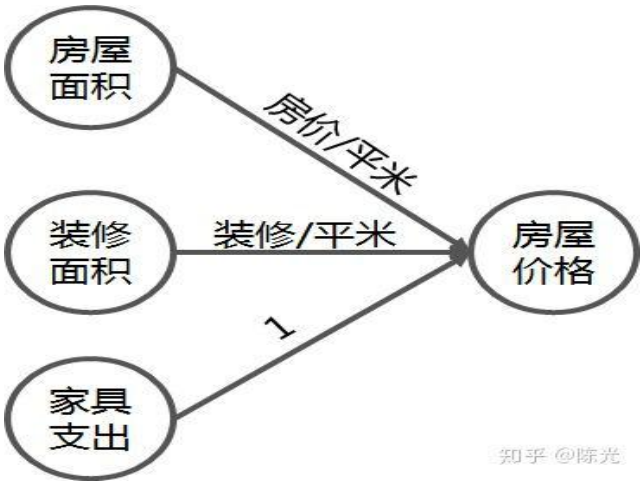
即房屋价格 = 房屋面积*每平方米房价。我们用两个圆圈和一条线段可将这个关系表示为：



这是房屋价格最简单的计算方法。

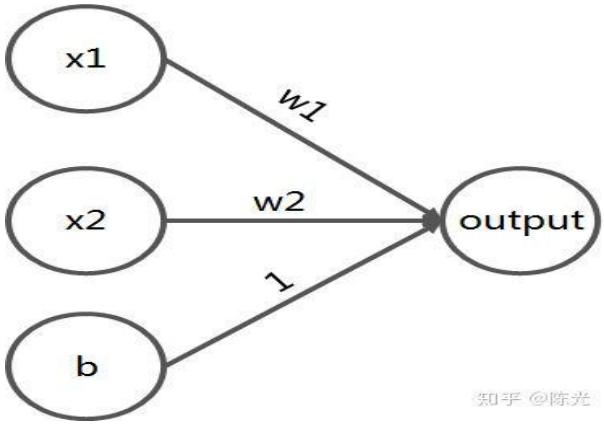
但是房屋价格还受到其他因素的影响，比如是是否装修、家具等。

引入装修和家具的支出，得房屋价格 = 房屋面积*每平方米房价 + 装修面积 * 每平方米装修 + 家具支出 * 1。最终的房屋价格组成的图应该如下所示：



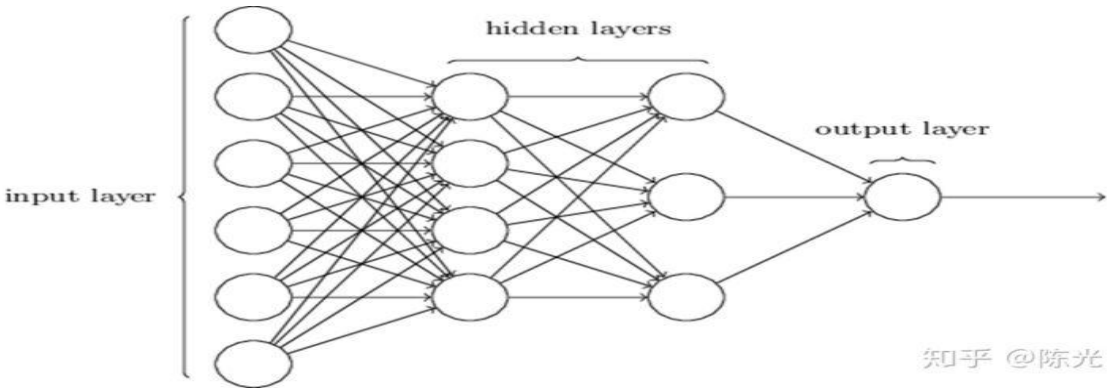
这就组成了一个预测房屋价格的基本网络。在这个网络中房屋面积、装修面积、家具支出是这个网络的输入，房价/平米、装修/平米为这个网络的参数，线段代表的是这个参数的乘法运算，房屋价格为这个网络的输出。

我们将上面的网络图做一个抽象表达，使其能够应用于除房价预测外的更多场景。如下所示：



对于这个简单的网络而言，x1、x2、b 被称作这个网络的输入，位于这一层的数据被称为输入层(Input Layer)；w1、w2 被称作这个网络的参数；线段为参数的运算规则，这里既可以是四则运算，也可是复杂的函数运算；output 为这个网络的输出，位于这一层数据被称为输出层(Output Layer)。

房价预测问题相对直观、简单，不需要太过复杂的网络即可实现。可一旦面对复杂的问题（如图像识别）时，无法通过简单的线性网络描述清楚，需要引入更多的参数和更为复杂的计算（比如 sigmoid、relu 等函数）。就出现了这种需要包含隐藏层(hidden layers)的网络。当网络越大时，整个网络所包含的参数就越多，网络也越复杂。网络越复杂，神经网络中的参数越难解释其作用，这就是深度神经网络被称为“黑盒”的原因。



图片出处：https://chatbotslife.com/how-neural-networks-work-ff4c7ad371f7

2 神经网络的参数

房价计算的神经网络搭建好后，我们就可以通过向网络中输入房屋面积、装修面积、家具支出等信息，得到房屋的价格了。当网络的参数（房价/平米、装修/平米）越准确时，使用该模型预测得到的输出（房屋价格）也将越准确。因此合理的参数设置，决定着一个神经网络的好坏。

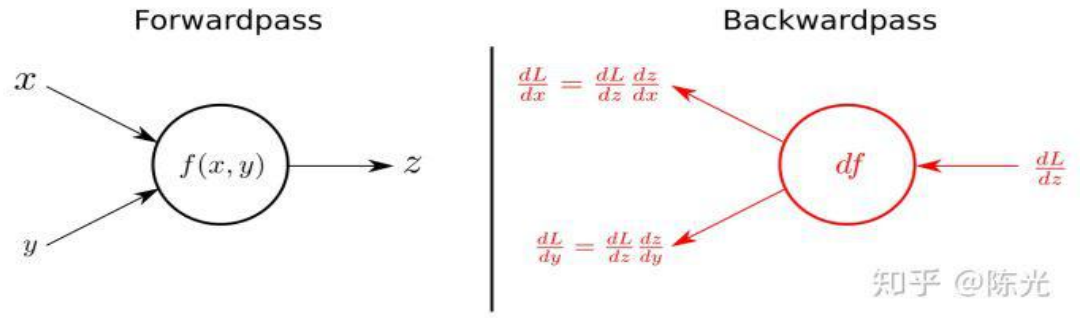
在深度学习技术普及前，神经网络的参数，是根据开发者的经验设置的。再通过真实的数据，带入验证，不断微调，使网络预测出的值尽可能接近真实值，进而得到越来越准确的参数。这种人为设置参数的行为在浅层的神经网络中尚可行得通，一旦网络参数达到千甚至上万级别时，这种方法变得不再可行。

为解决深层神经网络的参数调试方法，深度学习领域的专家提出了反向传播(Back propagation)理论。

数据由输入层传入，再经过隐藏层的一系列计算得到结果，并由输出层传出的这个过程被称为前向传播(Forward propagation)。反向传播的思路与前面提到的人为设置参数的方法类似，也是通过对比网络预测值与真实值之间的差异，进而微调网络。

不过反向传播的做法与人为设置参数有所不同，它需要计算预测值和真实的损失函数 L，损失函数可以理解为预测值和真实值之间的差值，差值越大，损失函数越大。

完成预测值与真值的损失函数计算后，通过求取前向传播参数的偏导的方法，将损失函数对参数的偏导传递到前一层网络，利用这个偏导与一个系数（学习率）的乘积更新网络中的参数。随后继续传播到更上一层的网络，直到网络中所有的参数都被更新。



图片出处：[Back Propagation in Convolutional Neural Networks]

每有一组数据，就可以利用反向传播的方法进行一次参数的更新，这就是深度学习网络会随着训练数据量的增大，变得越来越准确的原因。

反向传播的理论在优达学城(Udacity)无人驾驶工程师学位的深度学习基础课程中做了详细的介绍，也可以参考文章《一文弄懂神经网络中的反向传播法》，该文使用了一个简单的网络一步步阐述了反向传播的过程，浅显易懂。

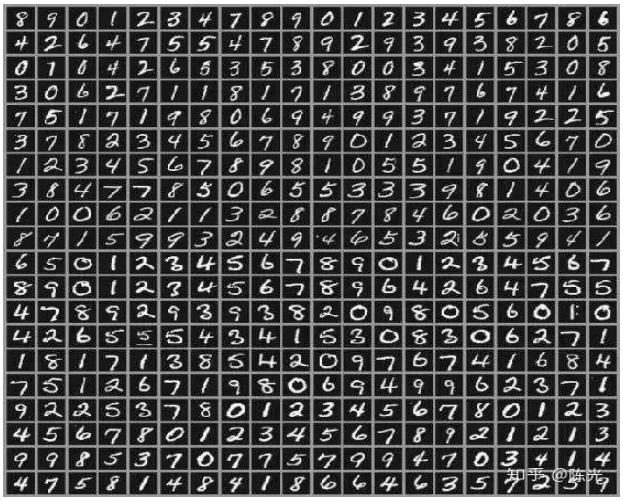
3 训练集、验证集、测试集

在前面的介绍中，我一直使用数据一词来表达神经网络的输入。实际上这些数据在神经网络的不同阶段有不同的称呼。他们分别是训练集(Training Set)、验证集(alidation Set)和测试集(Test Set)。

训练集和验证集是在神经网络模型的训练阶段使用的数据，而测试集是在神经网络模型完成训练后，用于评估模型时所使用的数据。做一个简单的比喻，训练集就是的学生的课本，学生需要根据课本来学习知识（训练模型）；验证集就是课后习题，学生通过课后习题来判断自己是否掌握了课本上的知识；测试集就是期末考试（评估模型），期末考试的题一般是课本和课后习题中没有，但是十分类似的题。

一个学生的成绩好不好，看下他期末考的好不好就知道了。一个神经网络模型好不好，看看它在测试集中的表现就知道了。

深度学习领域比较出名的数据集当属 MNIST 手写体数字数据集了，它包含了 60000 个训练样本和 10000 个测试样本。部分样本如下所示：



图片来源：http://yann.lecun.com/exdb/mnist/index.html

使用 Google 推出的深度学习框架 TensorFlow，能够直接获取 MNIST 手写体数字数据集，代码如下：

```
from tensorflow.examples.tutorials.mnist import input_data

mnist = input_data.read_data_sets('/home/my_mnist_data',one_hot=True)

train_features = mnist.train.images
test_features = mnist.test.images

train_labels = mnist.train.labels.astype(np.float32)
test_labels = mnist.test.labels.astype(np.float32)
```

代码中的 train_features 和 test_features 分别为训练集和测试集，即为手写字体数字的图片集合；train_labels 和 test_labels 分别是训练集和测试集的图像所对应的标签，即 0-9 的数字集合。

MNIST 数据集未提供验证集，工程上一般会从训练集中取出 15%~20%的数据作为验证集，余下的 80%~85%的数据作为训练集，用于完成训练过程。

4 使用 LeNet-5 做交通标志牌分类

了解以上内容后，就能大致理解神经网络的工作原理了。再补充一下 TensorFlow 的语法知识、看几个 TensorFlow 的例子，就可以自己动手搭建神经网络了。

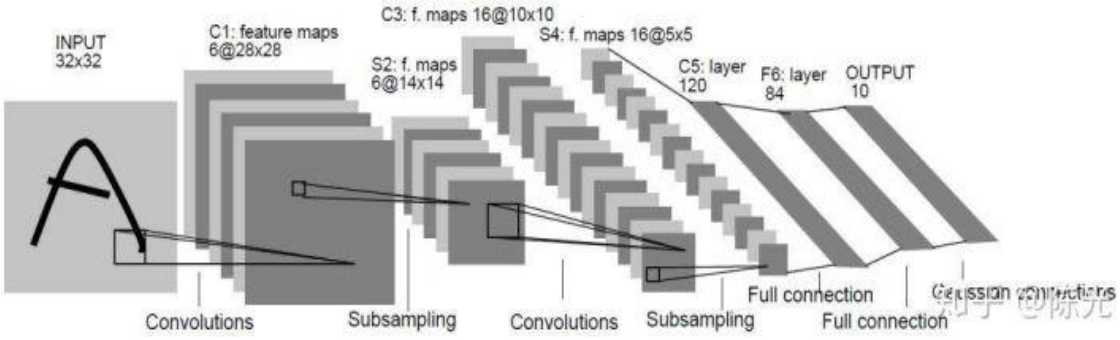
如果面对复杂的图像处理问题，需要使用卷积神经网络(CNN)。卷积神经网络是由卷积神经网络之父 Yann Lecun 在贝尔实验室工作期间，为解决手写数字识别而提出的。卷积是一个特殊的函数，其在神经网络中的定位与四则远算或某些特殊函数的地位没有区别。

下面我们通过导入交通标志牌的训练集，使用卷积神经网络之父 Yann Lecun 提出的 LeNet 模型，训练一个能识别交通标志的神经网络。

首先导入交通标志牌的数据。优达学(Udacity)无人驾驶工程师学位为我们提供了 34799 张图组成的数据集、4410 张图组成的验证集和 12630 张图组成的测试。这些数据集一共包含了 43 种不同的标志牌，比如限速、转向、停车标志牌。部分训练集的样本如下图所示：

由于 LeNet-5 默认需要输入尺寸为(32 x 32 x 1)的单通道的图像，因此我将训练集、验证集和测试集都进行灰度、缩放和归一化处理。处理后的部分样本如下：

LeNet-5 是一个不太复杂的卷积神经网络，下图显示了其结构。网络输入的是单通道的二维图像，先经过两次卷积层到池化层，再经过全连接层，最后使用 softmax 分类作为输出层。



图片出处：优达学城(Udacity)无人驾驶工程师学位

有关 LeNet-5 模型更详细介绍可以参看网络解析（一）：LeNet-5 详解。

在训练时会出现一个问题：训练集的准确率很高，但是验证集的准确率上不去。这表明模型训练时过拟合了，导致验证集只能达到 89%左右的识别率，而课程的要求是达到 93%以上。

为了解决模型过拟合导致的模型准确率低的问题，我做了两件事：

使用 imgaug 库做数据增广

使用 imgaug 库能够通过很简单的代码完成图像的翻转、平移、旋转、缩放、仿射变换、加噪声、修改颜色通道等功能。以实现数据库的增广，达到丰富训练集的目的。下图是 imgaug 库对同一张图片实现的数据增广的效果。

我对交通标志牌训练集添加随机噪声、修改对比度和横向翻转操作，完成了数据集增广。

在 LeNet-5 模型的全连接层后加入了 Dropout 函数

在 LeNet-5 网络中加入 Dropout 函数，能够让网络不会太依赖某些参数，因为这些参数随时可能被丢弃掉。在训练时，网络会被迫地学习一切的冗余表示，以确保至少将某些重要信息保存下来。当网络中的某个参数被丢弃时，还有其他参数能够完成相同的工作，这就是 Dropout 的功能。

在网络中加入 Dropout 函数的方式可以使得网络更加稳固，并能防止过拟合。

应用数据增广和 Dropout 函数后，重新训练即可使模型在测试集中的准确率超过 93%，达到要求。

5 结语

以上就是《深度学习入门之交通标志分类》的全部内容。文中的部分源码、图片和数据集来自优达学城(Udacity)无人驾驶工程师学位的第三个项目。

在这次分享中，我介绍了深度学习中所涉及的有关神经网络的理论知识。包括神经网络中的参数，反向传播原理，训练集、验证集和测试集的区别。在正文的最后介绍了如何利用 LeNet-5 网络实现交通标志牌的分类工作，当分类效果不理想时，分析原因并提供了解决方案。

在无人驾驶领域，深度学习除了用于识别图像中的物体外，还在激光点云分类障碍物、障碍物的轨迹预测、端到端的运动控制等领域得到了广泛应用。为现阶段无人驾驶技术的发展提供了巨大的帮助。掌握深度学习的理论知识和应用方法能够帮助我们解决无人驾驶领域很多棘手的问题。

好了\(^o^)/~，这篇分享就到这啦，我们下期见~

本文原载：知乎号“陈光”，作者授权转载。

临菲信息技术港

临菲信息技术港公众号

临菲学堂