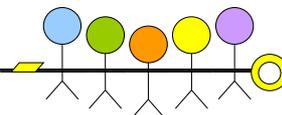


# 深度学习在计算机视觉中的应用

# 写在前面的话：受众群体



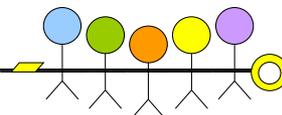
## □ 针对

- 初学者
- 对计算机视觉感兴趣者
- 从事AI行业
- 迷茫者

## □ 不针对

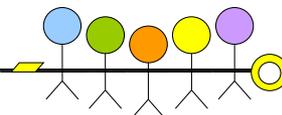
- 计算机视觉专业从业者
- 计算机视觉硕博学生
- 不针对大神们。。。

# 写在前面的话：相关知识储备

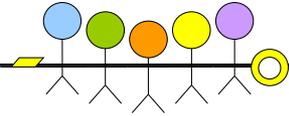


- 统计初步
- 线性代数
- 高等数学
- 数学优化
- Linux基础
- Python
- GPU CUDA
- 阅读英文文献的能力



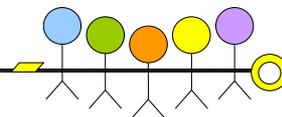


- 1. 计算机视觉
- 2. 深度学习
- 3. 卷积神经网络
- 4. YOLO初探
- 5. 深度神经网络训练流程
- 6. 评估指标
- 7. 硬件配置推荐
- 8. 总结及深度学习资源



# 1. 计算机视觉 computer vision

# 机器视觉vs计算机视觉



火龙果讲堂

uml.org.cn



# 一些计算机视觉任务

图片分类

Classification



CAT

图片分类+定位

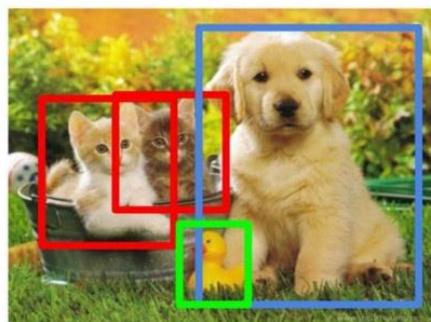
Classification  
+ Localization



CAT

物体检测

Object Detection



CAT, DOG, DUCK

语义分割

Instance  
Segmentation

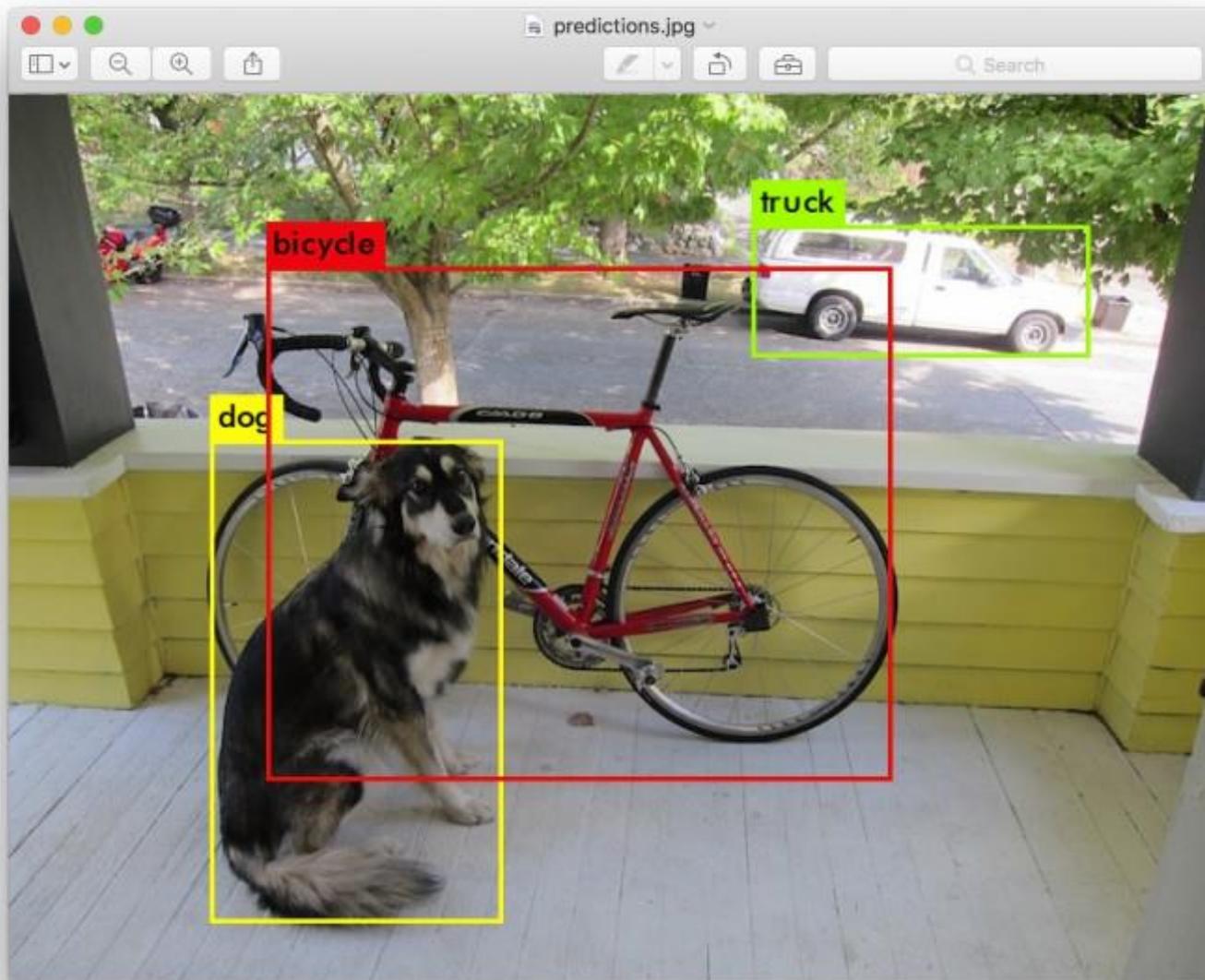
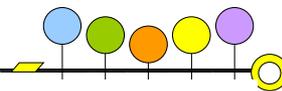


CAT, DOG, DUCK

Single object  
单一物体

Multiple objects  
多个物体

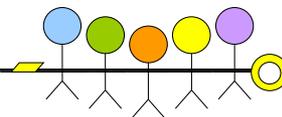
# 物体检测 object detection



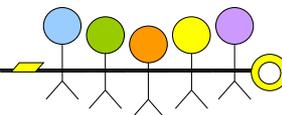
# 物体检测回顾 Recap Object Detection



- (1). Traditional objection detection:  
Cascade + Harr / SVM + HOG / DPM etc.
- (2). Proposal + deep learning classification:  
to extract proposal region and use deep learning to  
classify proposal region: RCNN / Fast-RCNN /  
Faster-RCNN / SPP-net / R-FCN (two stages)
- (3). **Deep learning + regression(end-to-end):**  
YOLO / SSD / DenseBox, RNN-RRC detection;  
combination of DPM and Deformable CNN (one stage)



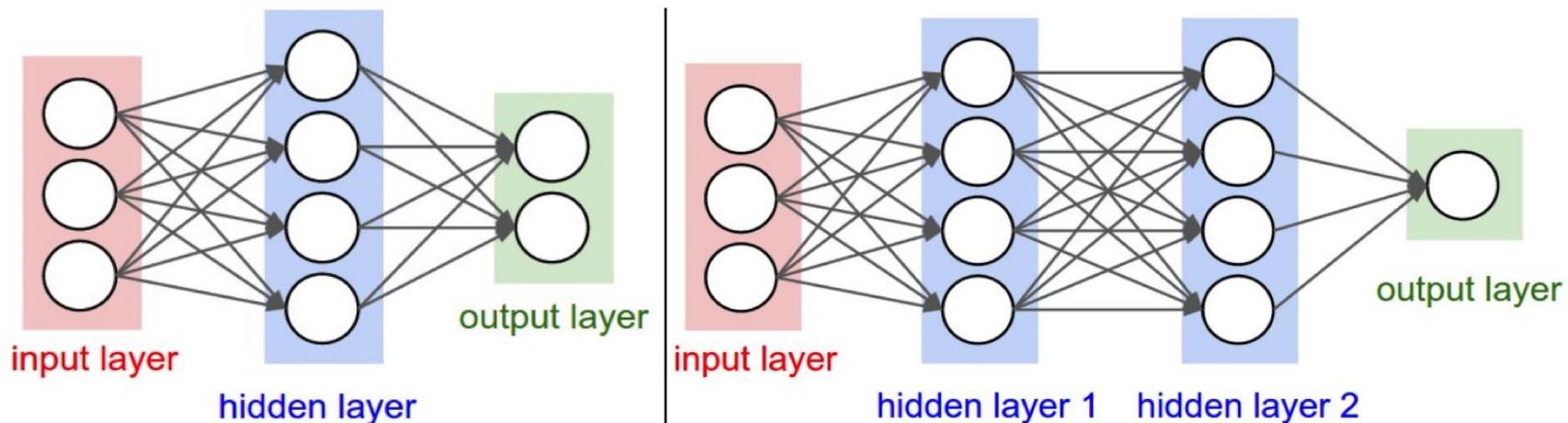
## 2. 深度学习 Deep Learning



- 语音识别， Speech recognition
- 机器翻译， Machine translation
- 自动驾驶， Self-driving
- 计算机视觉， Computer vision
- 自然语言处理， Natural language processing
- 对话系统， Dialogue system
- 生物信息学， Bioinformatics etc.

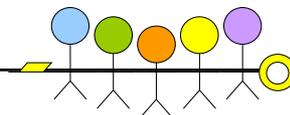
- 该方案可以让计算机从经验中学习，并根据层次化的概念体系来理解世界，而每个概念则通过与某些相对简单的概念之间的关系来定义。让计算机从经验获取知识，可以避免由人类来给计算机形式化地指定它需要的所有知识。层次化的概念让计算机构建较简单的概念来学习复杂概念。如果绘制出这些概念如何建立在彼此之上的图，我们将得到一张“深”（层次很多）的图。基于这个原因，我们称这种方法为 **AI 深度学习 (deep learning)**
- 定义来自于深度学习圣经
- 作者：**Ian Goodfellow, Yoshua Bengio, Aaron Courville**

# 深度学习图解-有深就有浅



Stanford university cs 231n

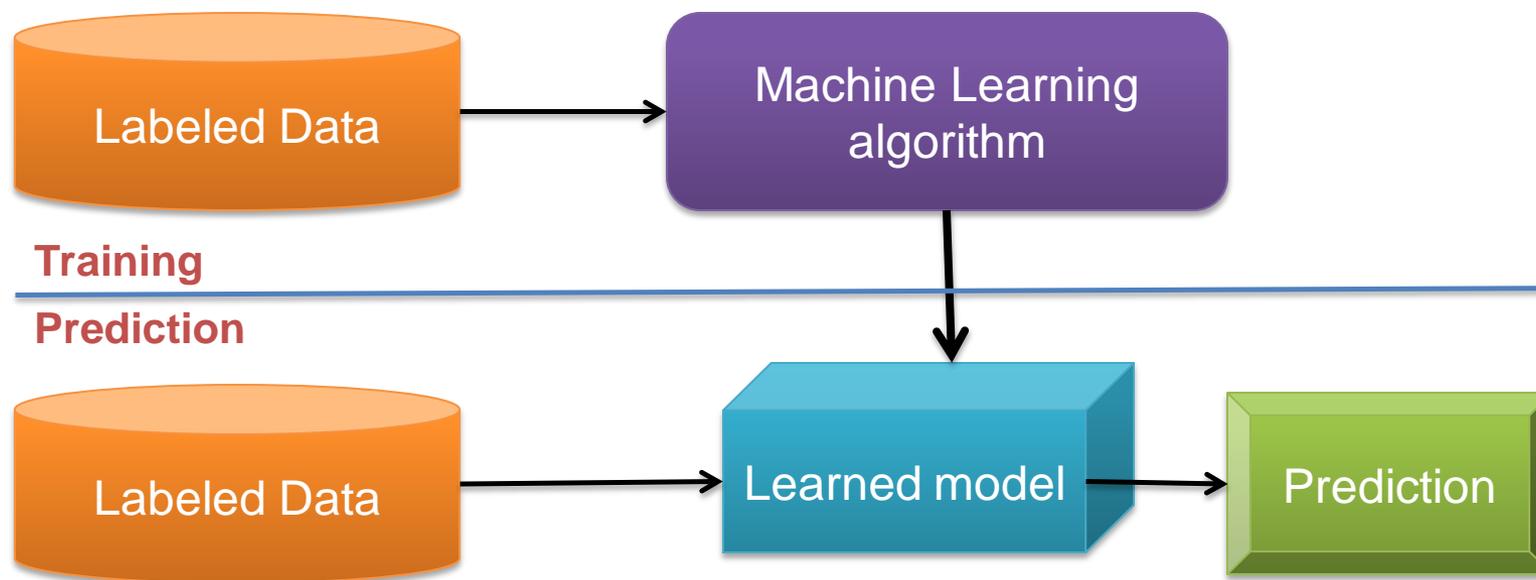
# 机器学习基础 Machine Learning Basics



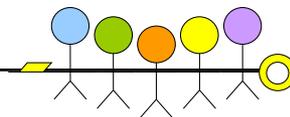
火龙果讲堂

uml.org.cn

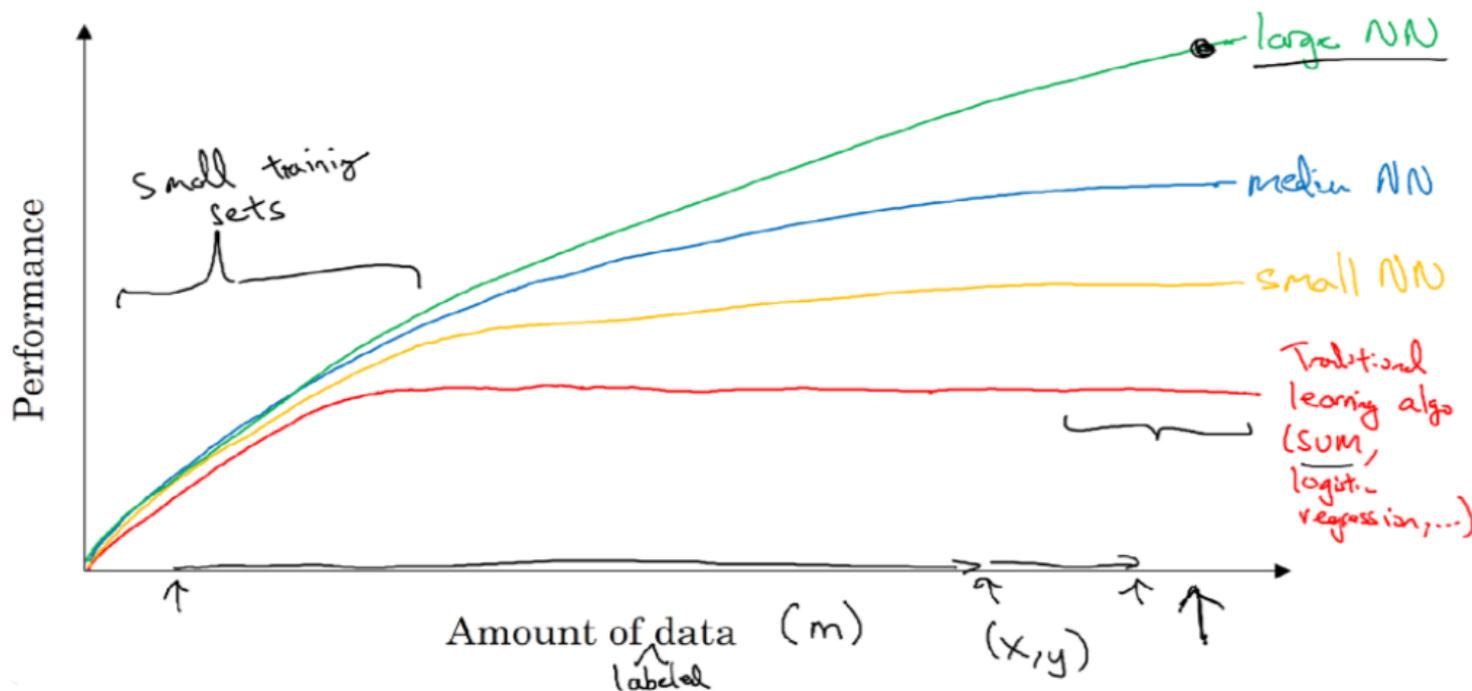
Machine learning is a field of computer science that gives computers the ability to **learn without being explicitly programmed**

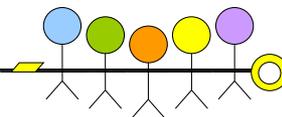


Methods that can learn from and make predictions on data.  
This slide from PhD Smini Lourentzou



## Scale drives deep learning progress

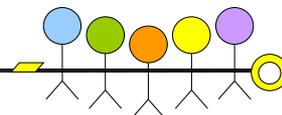




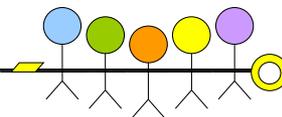
- Yoshua Bengio
- Geoffrey Hinton
- Yann Lecun
- Andrew Ng
- Ian Goodfellow
- Alex Krizhevsky etc...

注释: 排名不分先后

# 深度学习关键术语（部分）

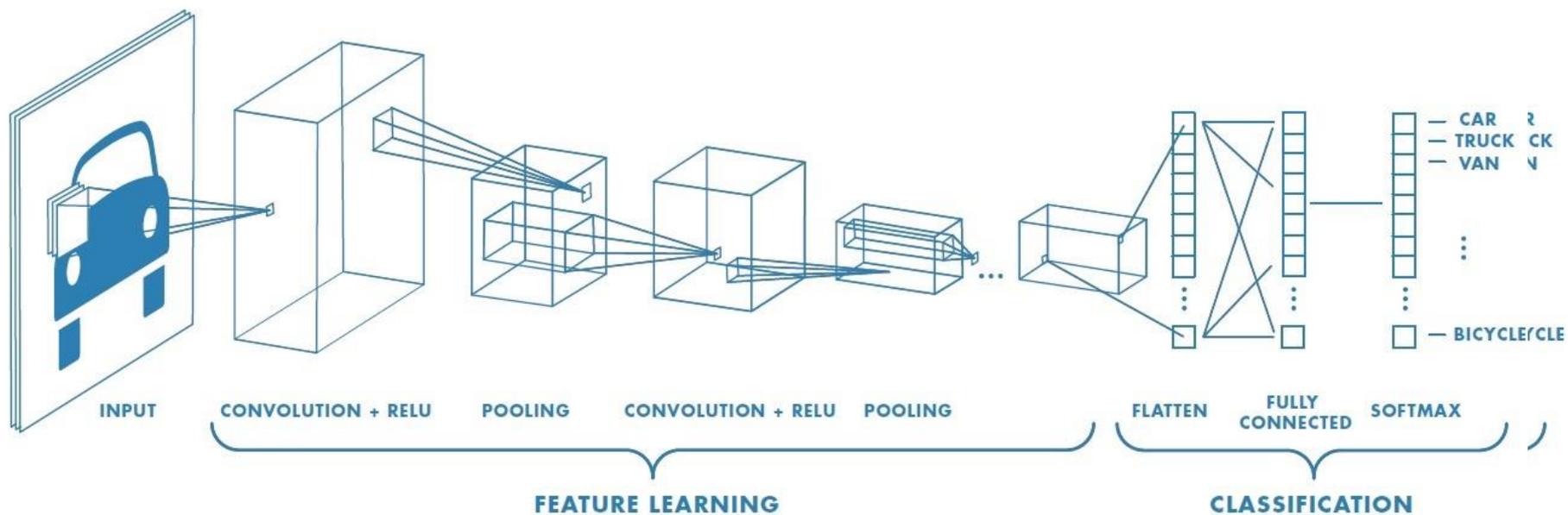


- 人工神经网络，artificial neural network
- 后向传播，back-propagation
- 全连接层，fully Connected Layer
- **卷积层**，convolutional Layer
- 过拟合，overfitting
- **激活函数**，activation function
- 权重，weights
- 损失函数，cost function
- **池化**，pooling
- 随机梯度下降，stochastic gradient descent
- **步长**，stride
- 填充，padding



### 3. 卷积神经网络

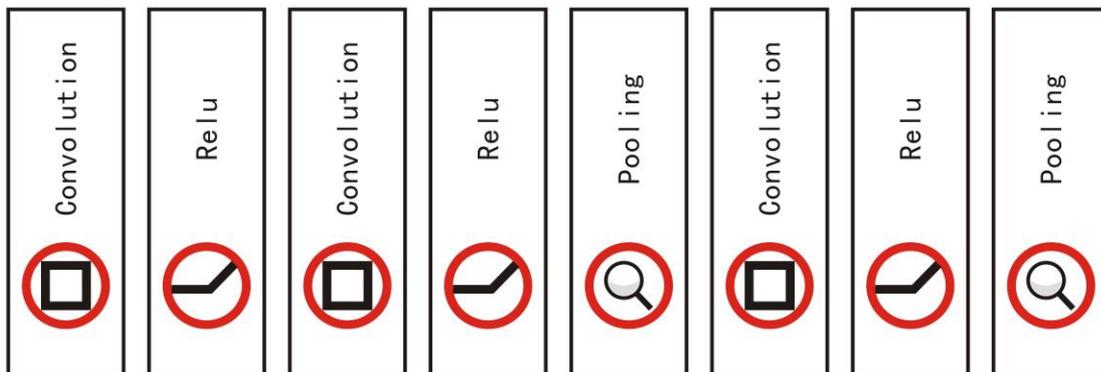
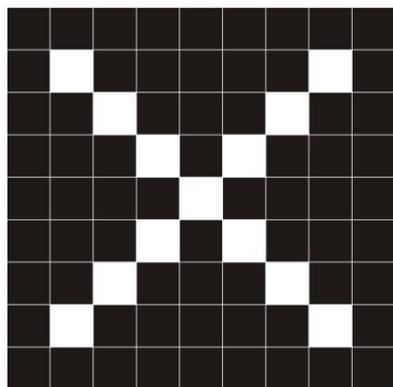
# 卷积网络结构



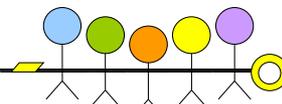
图片来自于网络

# 卷积神经网络结构平面图

## 卷积 激活 池化



# 卷积运算1



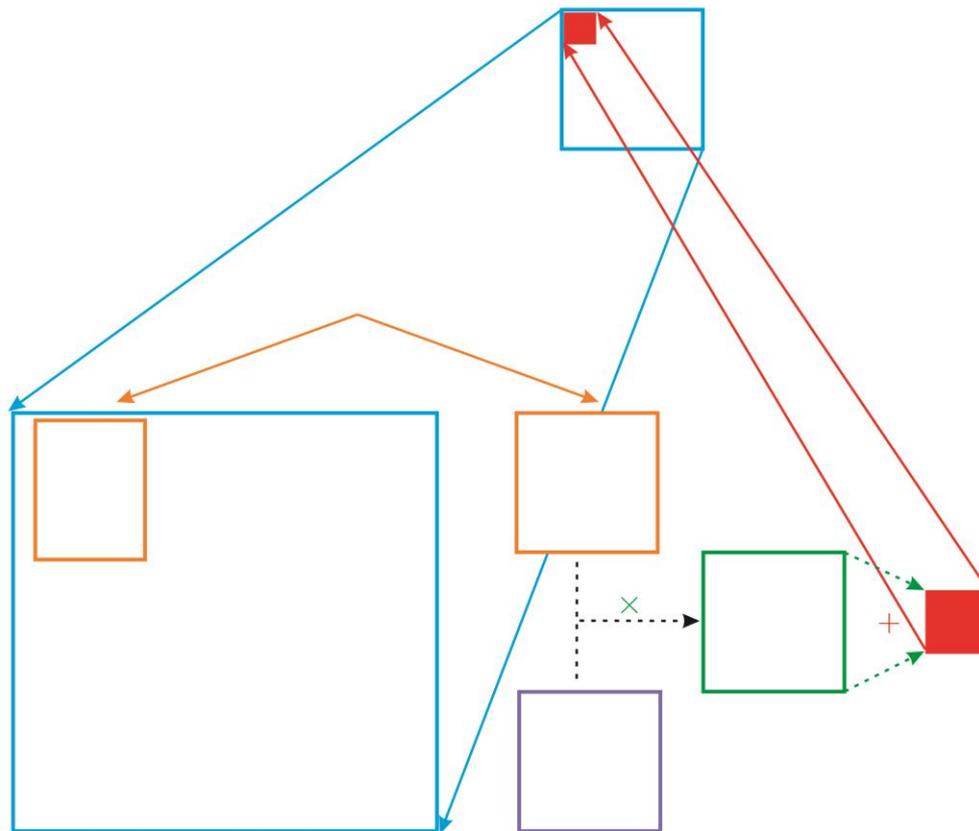
1 <sub>x1</sub>	1 <sub>x0</sub>	1 <sub>x1</sub>	0	0
0 <sub>x0</sub>	1 <sub>x1</sub>	1 <sub>x0</sub>	1	0
0 <sub>x1</sub>	0 <sub>x0</sub>	1 <sub>x1</sub>	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

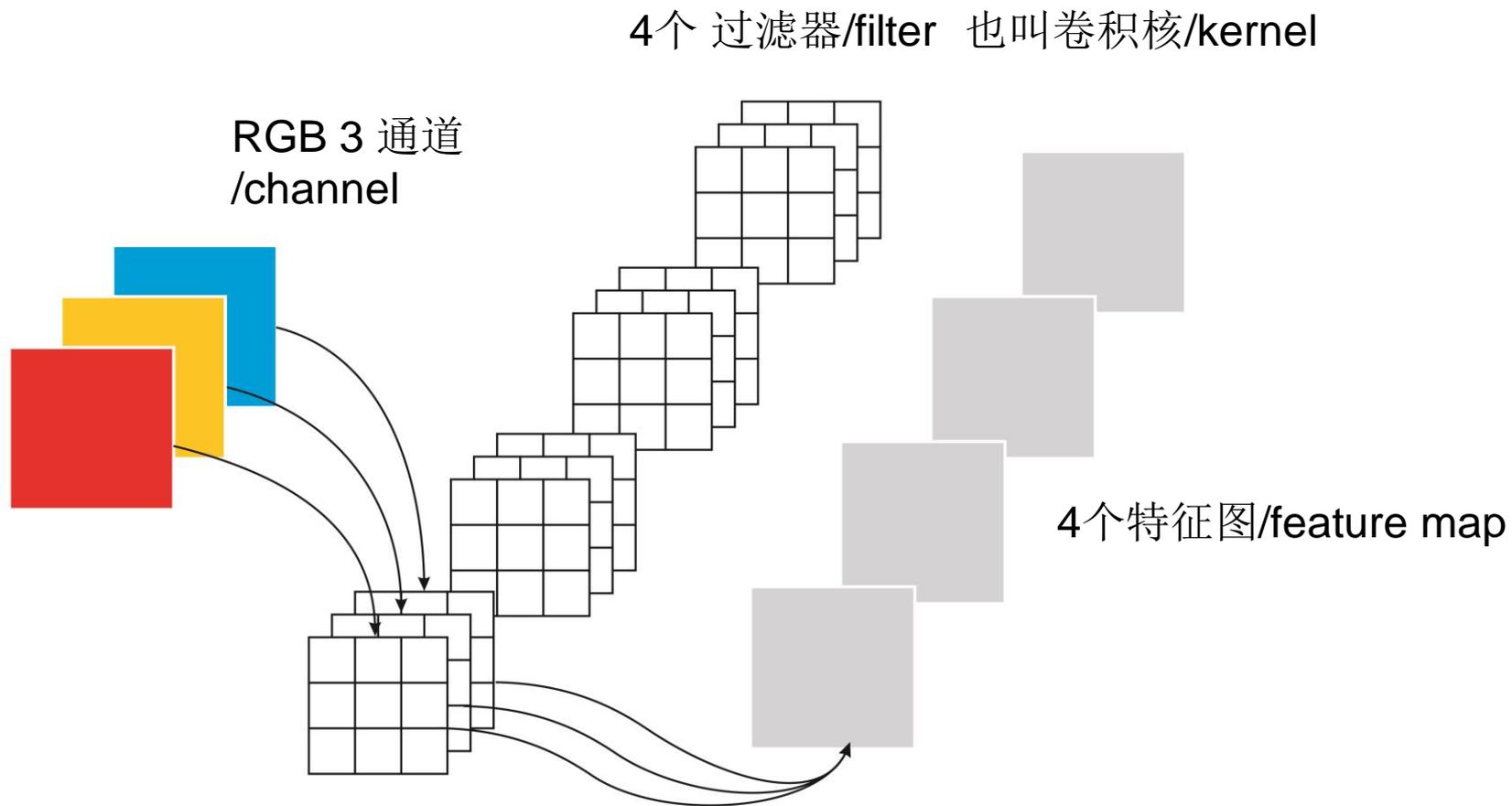
Convolved  
Feature

# 卷积运算2



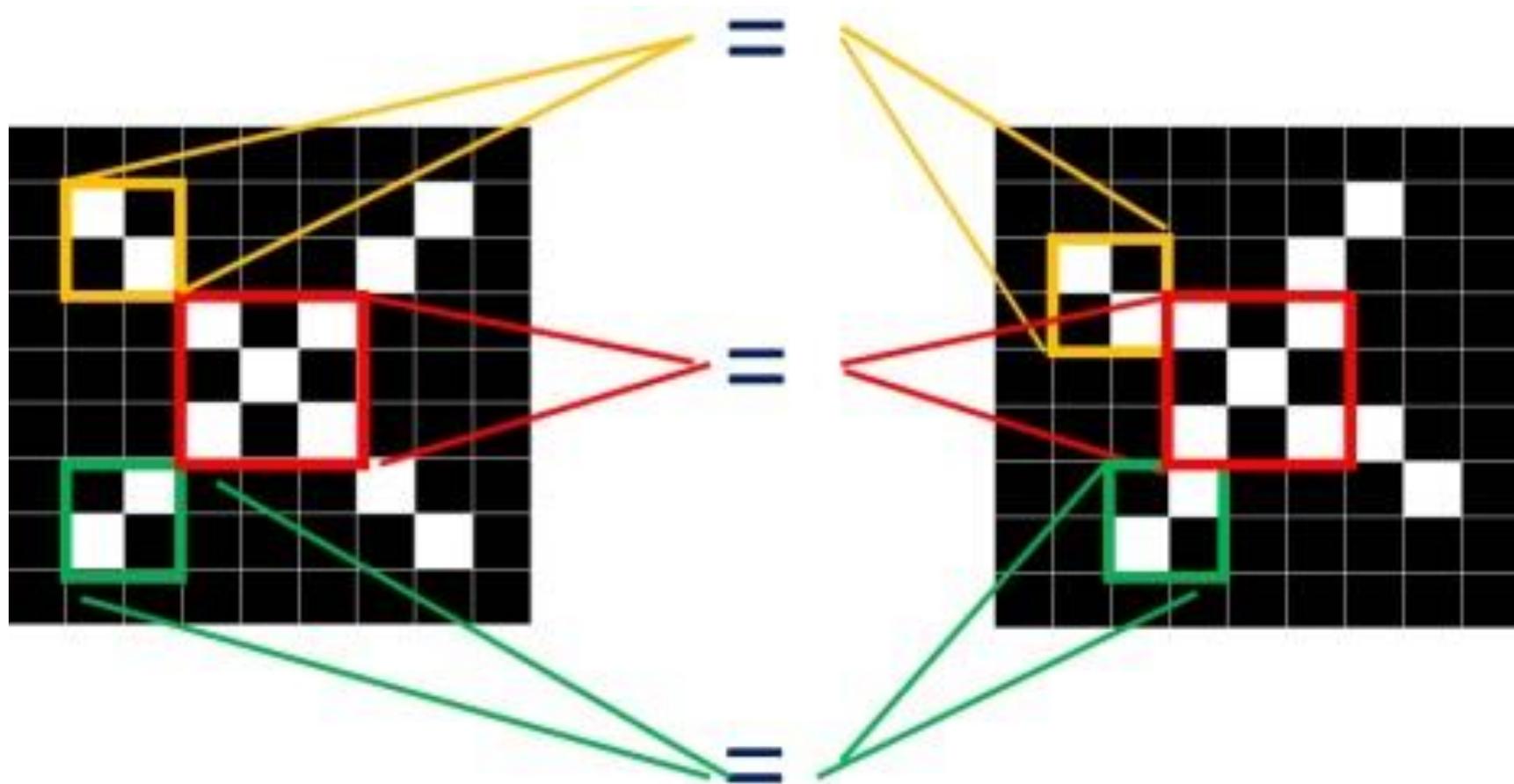
Convolutional Neural Networks (CNNs) explained  
卷积层解释 图片来源于网络

# 卷积运算3



卷积层/Convolutional layer，层上的每一个九宫格的数字叫做权重/weight  
训练卷积神经网络就是要为了得到训练数据集的神经网络权重

# 区域特征表达



Pictures are extracted from <https://zhuannlan.zhihu.com/p/27908027>

# 卷积可视化



This image is from Deeplizard

# 激活层 activation layer

常用激活函数有：

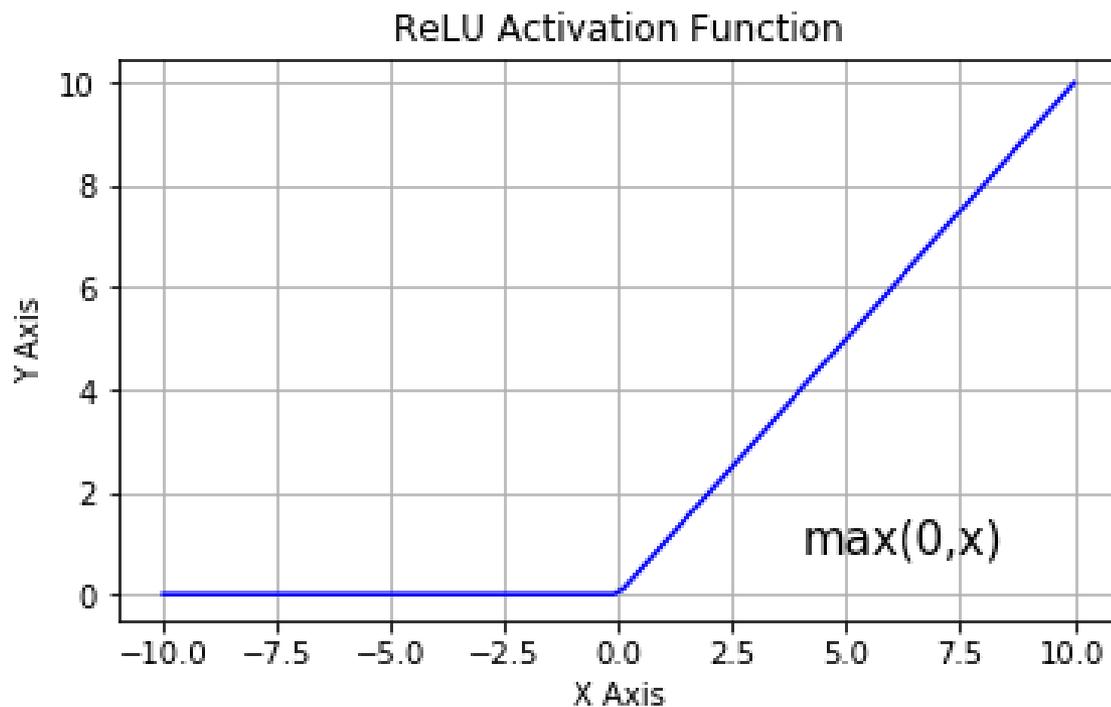
sigmoid

Tanh

ReLU, Leaky-ReLU,

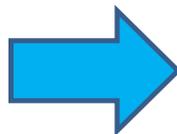
P-ReLU, R-ReLU

Maxout



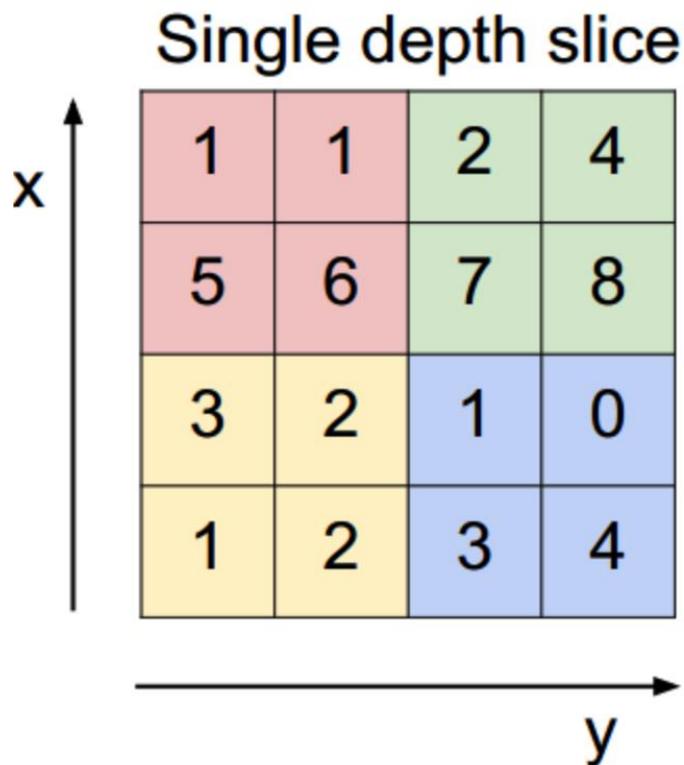
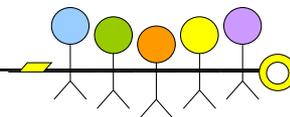
# 激活层运算结果

0.1	0.3	<b>-0.75</b>
1	5	12
7	<b>-9</b>	8
<b>-0.01</b>	3	<b>-0.63</b>



0.1	0.3	<b>0</b>
1	5	12
7	<b>0</b>	8
<b>0</b>	3	<b>0</b>

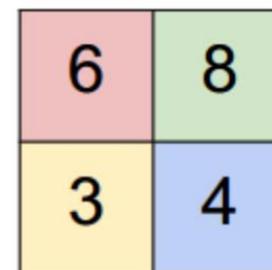
# 池化层 pooling layer



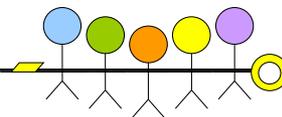
max pool with 2x2 filters  
and stride 2



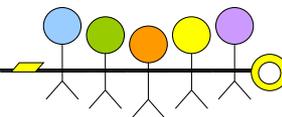
Maxpool



# 卷积神经网络框架（部分）

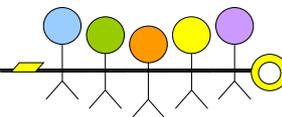


- Lenet (最早1986)
- Alexnet(2012)
- Googlenet(2014)
- VGG(2014)
- Deep Residual Learning(2015)
- R-CNN
- FAST R-CNN
- FASTER R-CNN
- Mask R-CNN
- SSD: Single Shot MultiBox Detector
- YOLO: You Only Look Once (2015-2018)



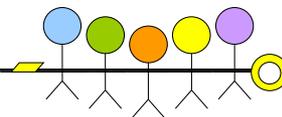
## 4. YOLO初探

# YOLO 一种目标检测框架



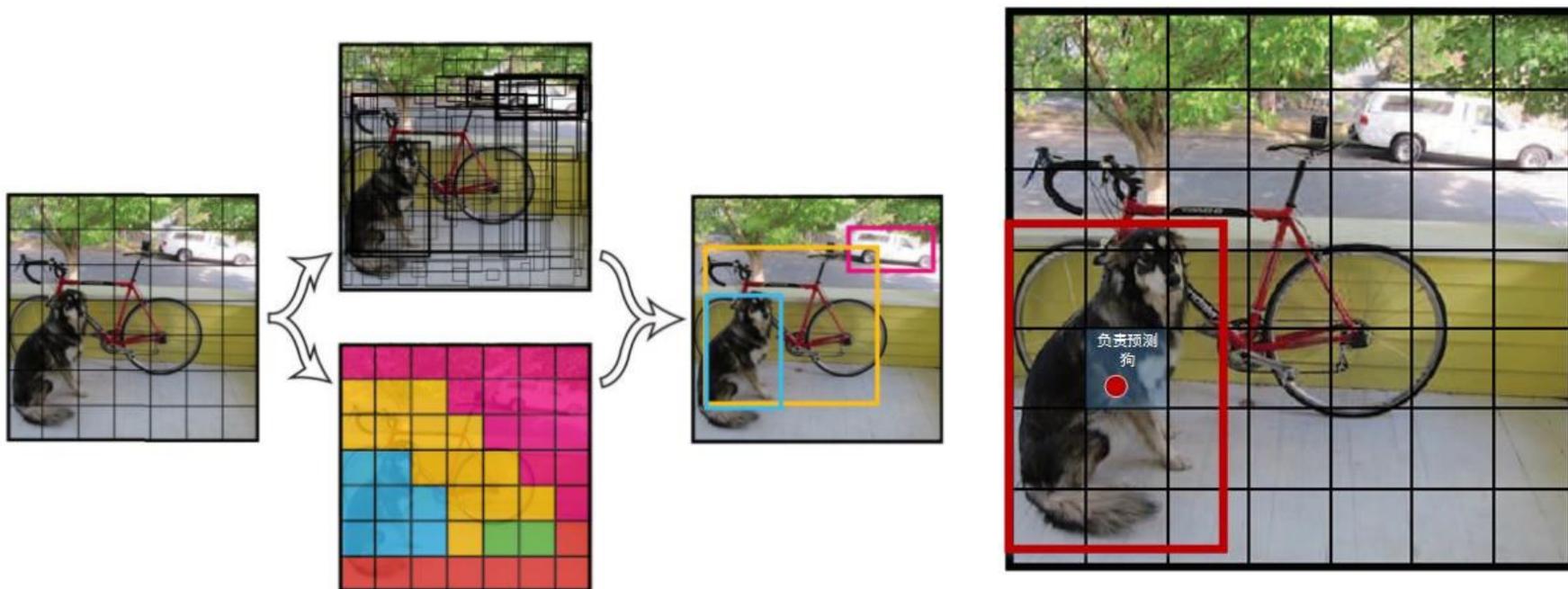
- YOLO: You Only Look Once
- 作者: Joseph Redmon, Ali Farhadi
- 官网: <http://pjreddie.com/darknet/yolo/>
- 论文地址yolo: <http://arxiv.org/abs/1506.02640>
- 论文地址yolo9000: <https://arxiv.org/abs/1612.08242>
- 论文地址yoloV3:  
<https://pjreddie.com/media/files/papers/YOLOv3.pdf>
- 代码地址: <https://github.com/pjreddie/darknet>

# YOLO术语解释（部分）

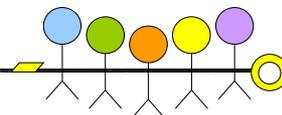


- 真实框框 ground truth
- 预测框框 bounding box
- 建议框 anchor box
- 网络架构文件 .cfg
- 网络权重文件 .weights
- 数据路径文件 .data
- 类别文件 .names

# YOLO v2-v3 核心思想

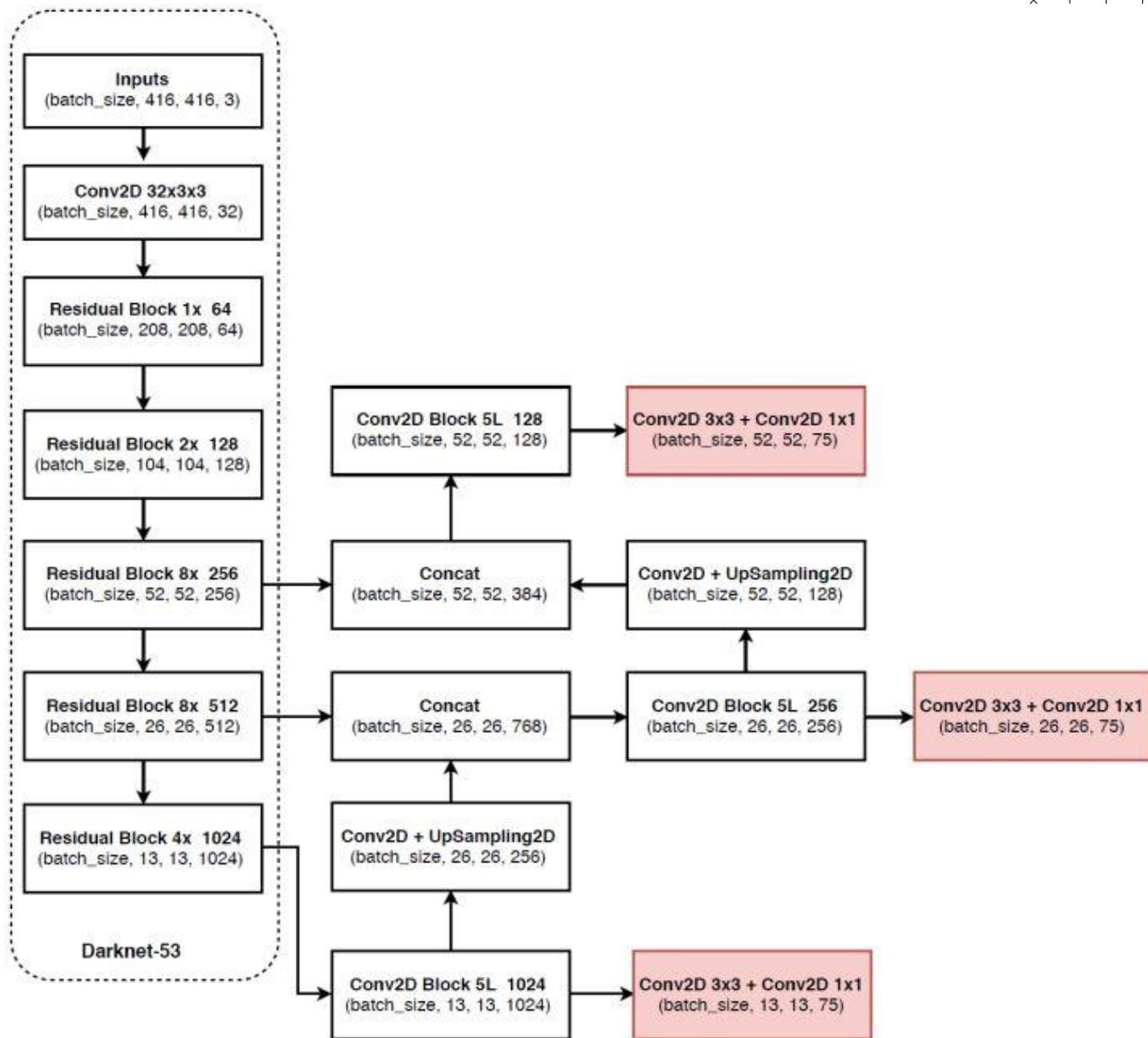
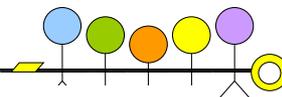


# YOLO v2

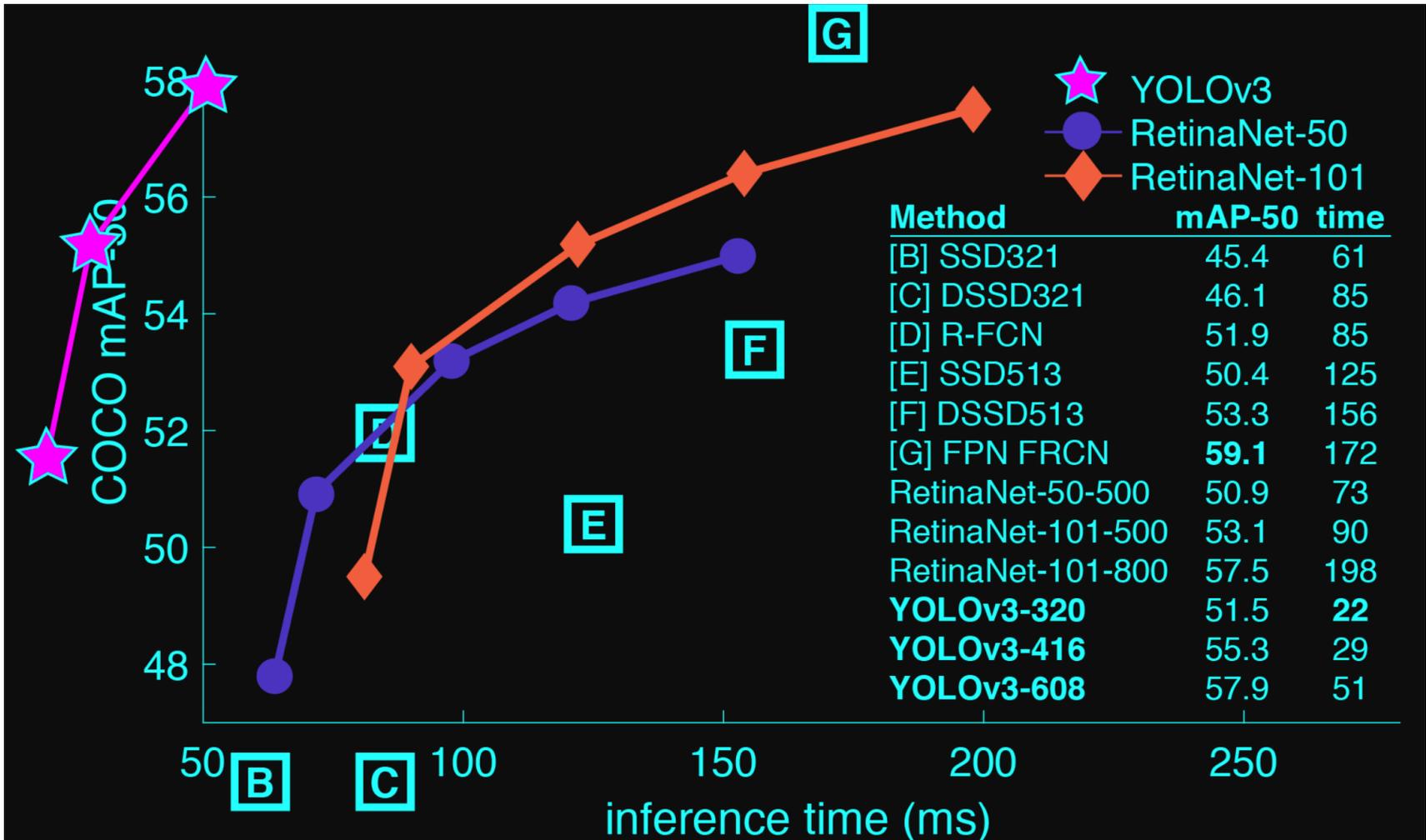


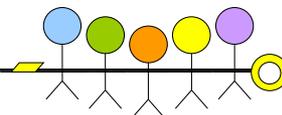
Type	Filters	Size/Stride	Output
Convolutional	32	$3 \times 3$	$224 \times 224$
Maxpool		$2 \times 2/2$	$112 \times 112$
Convolutional	64	$3 \times 3$	$112 \times 112$
Maxpool		$2 \times 2/2$	$56 \times 56$
Convolutional	128	$3 \times 3$	$56 \times 56$
Convolutional	64	$1 \times 1$	$56 \times 56$
Convolutional	128	$3 \times 3$	$56 \times 56$
Maxpool		$2 \times 2/2$	$28 \times 28$
Convolutional	256	$3 \times 3$	$28 \times 28$
Convolutional	128	$1 \times 1$	$28 \times 28$
Convolutional	256	$3 \times 3$	$28 \times 28$
Maxpool		$2 \times 2/2$	$14 \times 14$
Convolutional	512	$3 \times 3$	$14 \times 14$
Convolutional	256	$1 \times 1$	$14 \times 14$
Convolutional	512	$3 \times 3$	$14 \times 14$
Convolutional	256	$1 \times 1$	$14 \times 14$
Convolutional	512	$3 \times 3$	$14 \times 14$
Maxpool		$2 \times 2/2$	$7 \times 7$
Convolutional	1024	$3 \times 3$	$7 \times 7$
Convolutional	512	$1 \times 1$	$7 \times 7$
Convolutional	1024	$3 \times 3$	$7 \times 7$
Convolutional	512	$1 \times 1$	$7 \times 7$
Convolutional	1024	$3 \times 3$	$7 \times 7$
Convolutional	1000	$1 \times 1$	$7 \times 7$
Avgpool		Global	1000
Softmax			

# YOLO v3



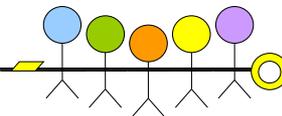
# YOLO v3速度测评-快！



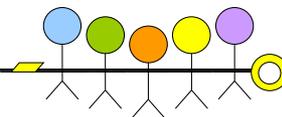


## 5. 深度神经网络训练流程 pipeline

# 训练流程 pipeline

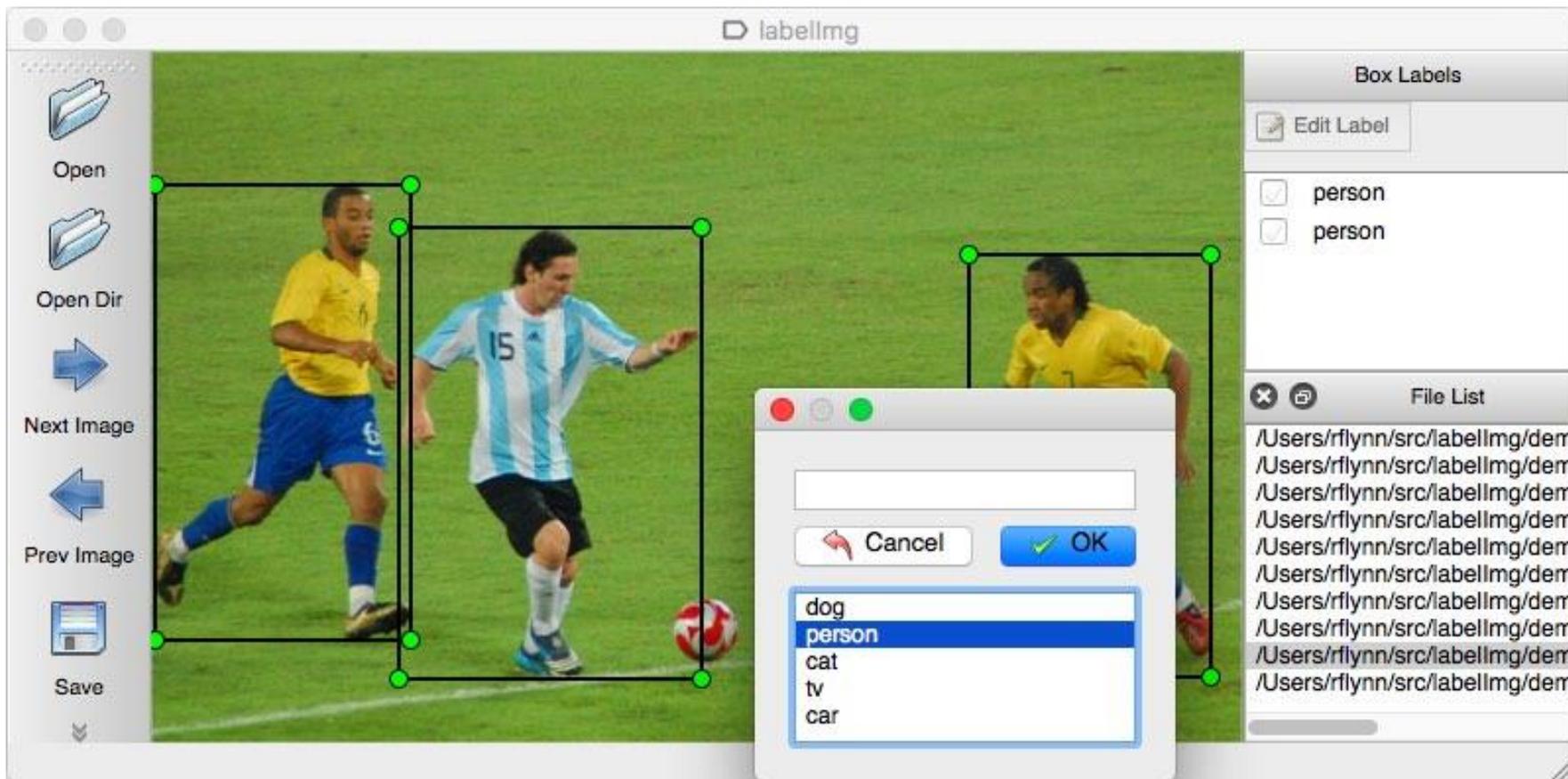


- 数据采集 collecting data
- 数据标注 annotated data
- 环境建立 software
- 训练 training

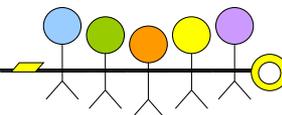


- ImageNet
- MNIST
- CIFAR-10 and CIFAR-100 dataset
- Microsoft COCO dataset
- Caltech 101 – Wikipedia
- PASCAL Visual Objects Classes(VOC)
- Places by MIT
- NIH

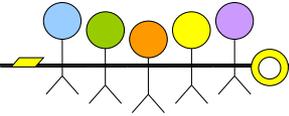
# 图片标注工具



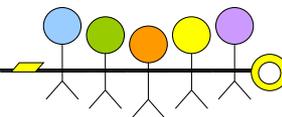
<https://github.com/tzutalin/labellmg>



- Linux操作系统
- CUDA+cuDNN
- Python
- YOLO(Darknet19, Darknet53)
- 其他深度学习框架：Tensorflow, Pytorch, Caffe等



## 6.评估指标 evaluation metrics

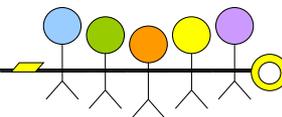


- 精确率，precision
- 召回率，recall
- 准确度，accuracy
- 识别准确率，mAP
- 检测效果，IOU

mAP (Mean Average Precision) 是把每个类别的AP都单独拿出来, 然后计算所有类别AP的平均值, 代表着对检测到的目标平均精度的一个综合度量:

$$MAP = \frac{\sum_{q=1}^Q AveP(q)}{Q}$$

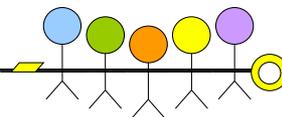
# 检测效果 IOU



检测效果 IOU  
intersection over union

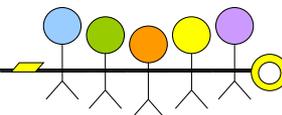
=



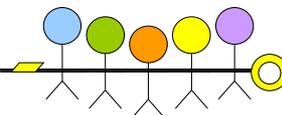


## 6. 硬件配置推荐

# 建议最低配置(个人使用)

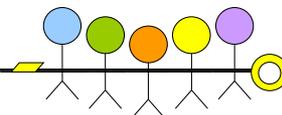


- 4 GTX 1080Ti
- Memory 64g
- SSD 128G, HDD 2TB , 7200RPM
- I7 酷睿第7代处理器。目前有第八代
- 电源，风扇，主板去装机城报价。。。。
- 咨询专业的GPU供应商定制
- 根据经济状况配机器



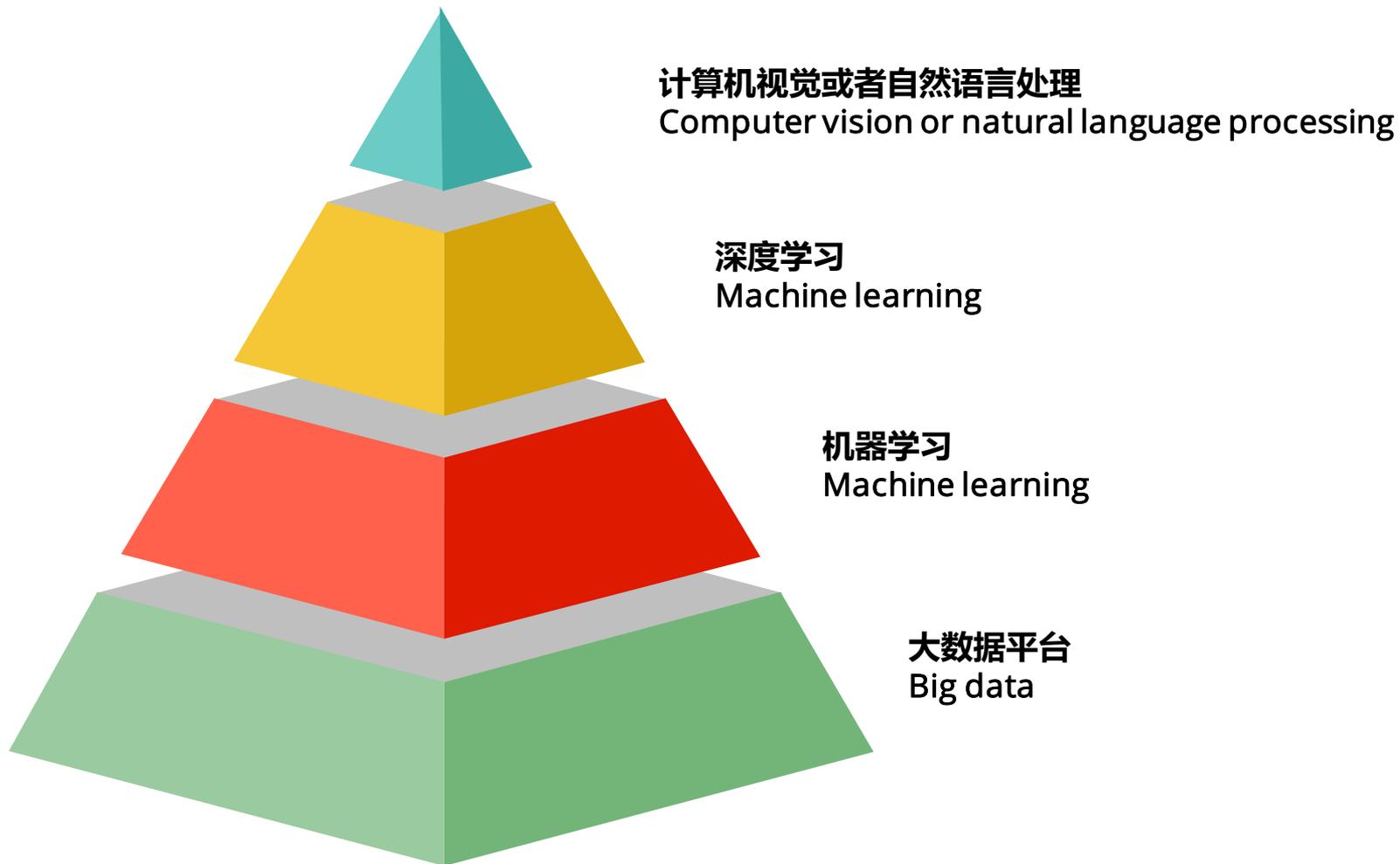
## 8.总结及深度学习资源

# 总结

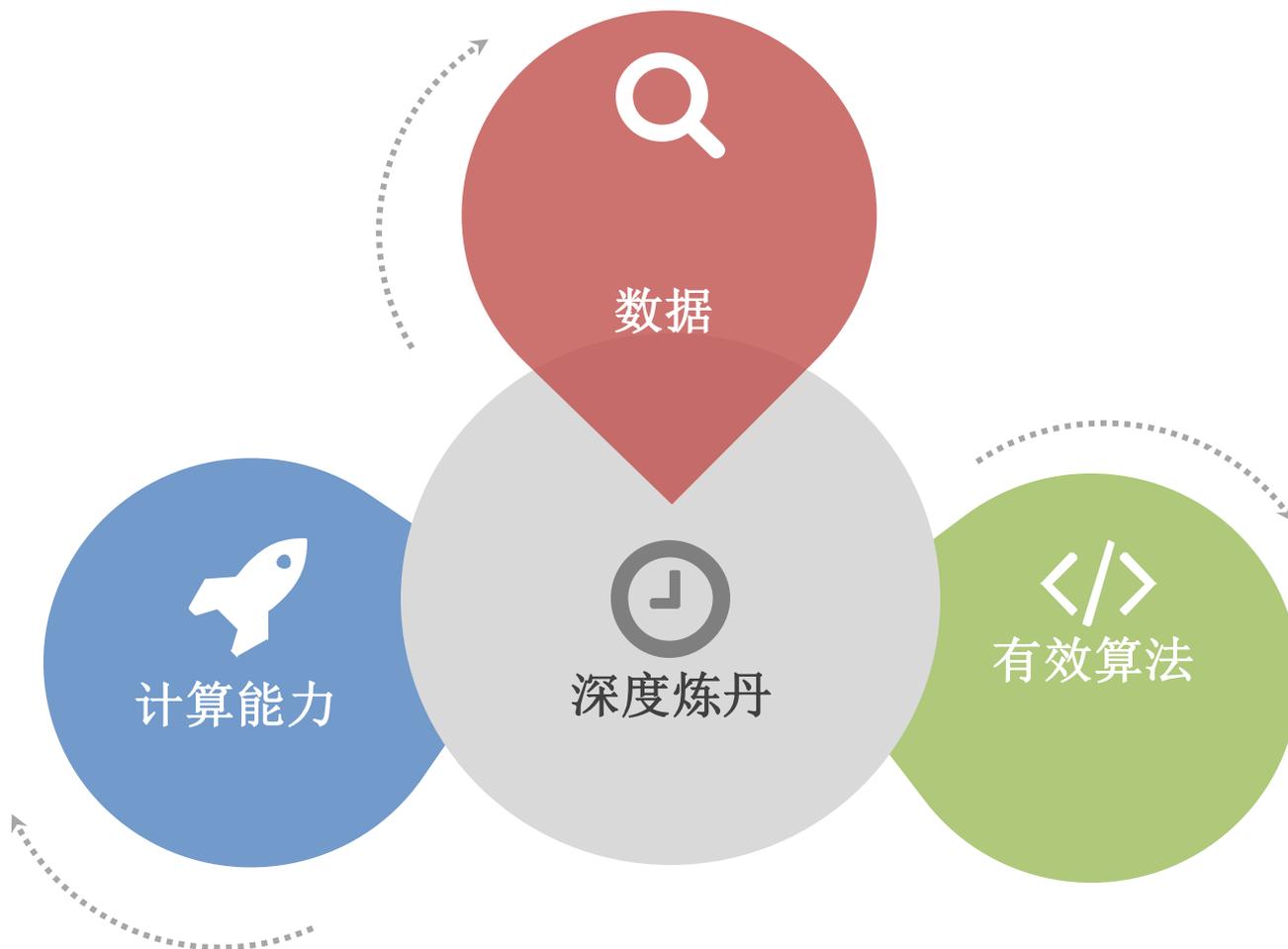
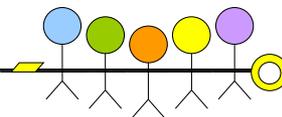


1. 机器视觉 vs 计算机视觉
2. 深度学习
3. 卷积神经网络初探
4. 一种计算机视觉框架YOLO
5. 深度学习硬件推荐配置

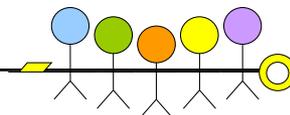
# 给大家建立层次感1



# 给大家建立层次感2



# 部分深度学习框架



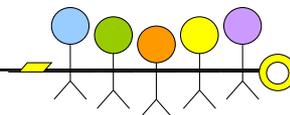
火龙果讲堂

uml.org.cn

来源于互联网

Name	Language	Link	Note
Microsoft Cognitive Toolkit	python	<a href="https://www.microsoft.com/en-us/cognitive-toolkit/">https://www.microsoft.com/en-us/cognitive-toolkit/</a>	A python deep learning library by Microsoft Previous known CNTK
Pylearn2	Python	<a href="http://deeplearning.net/software/pylearn2/">http://deeplearning.net/software/pylearn2/</a>	A machine learning library built on Theano
Theano	Python	<a href="http://deeplearning.net/software/theano/">http://deeplearning.net/software/theano/</a>	A python deep learning library
Caffe	C++	<a href="http://caffe.berkeleyvision.org/">http://caffe.berkeleyvision.org/</a>	A deep learning framework by Berkeley
Torch	Lua	<a href="http://torch.ch/">http://torch.ch/</a>	An open source machine learning framework
Overfeat	Lua	<a href="http://cilvr.nyu.edu/doku.php?id=code:start">http://cilvr.nyu.edu/doku.php?id=code:start</a>	A convolutional network image processor
Deeplearning4j	Java	<a href="http://deeplearning4j.org/">http://deeplearning4j.org/</a>	A commercial grade deep learning library
Word2vec	C	<a href="https://code.google.com/p/word2vec/">https://code.google.com/p/word2vec/</a>	Word embedding framework
GloVe	C	<a href="http://nlp.stanford.edu/projects/glove/">http://nlp.stanford.edu/projects/glove/</a>	Word embedding framework
Doc2vec	C	<a href="https://radimrehurek.com/gensim/models/doc2vec.html">https://radimrehurek.com/gensim/models/doc2vec.html</a>	Language model for paragraphs and documents
StanfordNLP	Java	<a href="http://nlp.stanford.edu/">http://nlp.stanford.edu/</a>	A deep learning-based NLP package
TensorFlow	Python	<a href="http://www.tensorflow.org">http://www.tensorflow.org</a>	A deep learning based python library

# 部分深度学习资源

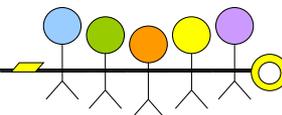


- <http://web.stanford.edu/class/cs224n>
- <https://www.coursera.org/specializations/deep-learning>
- <https://chrisalbon.com/#Deep-Learning>
- <http://www.asimovinstitute.org/neural-network-zoo>
- <http://cs231n.github.io/optimization-2>
- <https://medium.com/@ramrajchandradevan/the-evolution-of-gradient-descend-optimization-algorithm-4106a6702d39>
- <https://arimo.com/data-science/2016/bayesian-optimization-hyperparameter-tuning>
- <http://www.wildml.com/2015/12/implementing-a-cnn-for-text-classification-in-tensorflow>
- <http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp>
- <https://medium.com/technologymadeeasy/the-best-explanation-of-convolutional-neural-networks-on-the-internet-fbb8b1ad5df8>
- <http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/>
- <http://www.wildml.com/2015/10/recurrent-neural-network-tutorial-part-4-implementing-a-grulstm-rnn-with-python-and-theano/>
- <http://colah.github.io/posts/2015-08-Understanding-LSTMs>
- <https://github.com/hyperopt/hyperopt>
- <https://github.com/tensorflow/nmt>
- <https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/>
- 来源于互联网

# 学习资源很多，选择适合自己难度的



- fast.ai
- deeplearning.ai
- 莫烦公开课
- 李宏毅公开课
- Kaggle案例



感谢您的宝贵时间