《医学信息技术》课程

三、深度学习基础

黄恩待 副研究员

智能医学与生物医学工程研究院

2024-12-17

CONTENTS

01

基本概念

一、基本概念

机器学习：

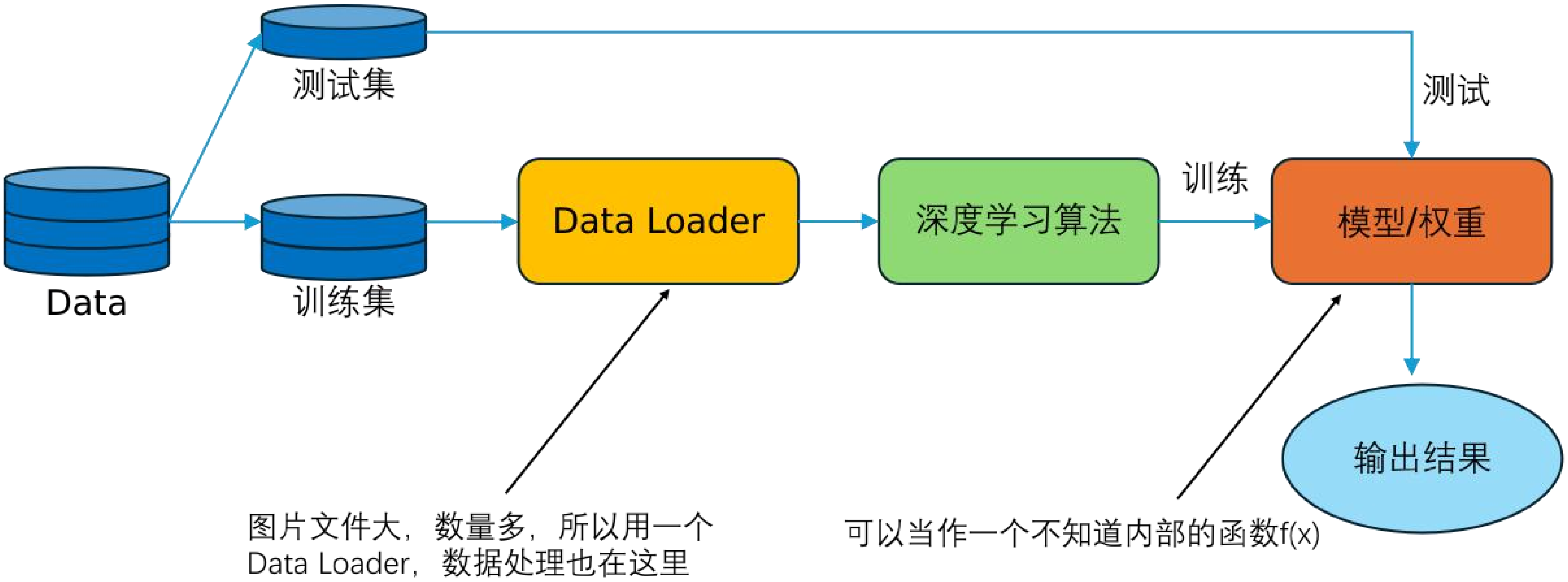
- 研究如何模拟和实现人类学习行为

深度学习：

- 以神经网络为代表的，层数很深的机器学习
- 实际中“机器学习“和“深度学习”不同

一、基本概念

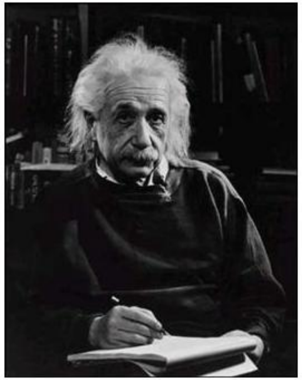
深度学习主要流程:



一、基本概念

计算机里的图像

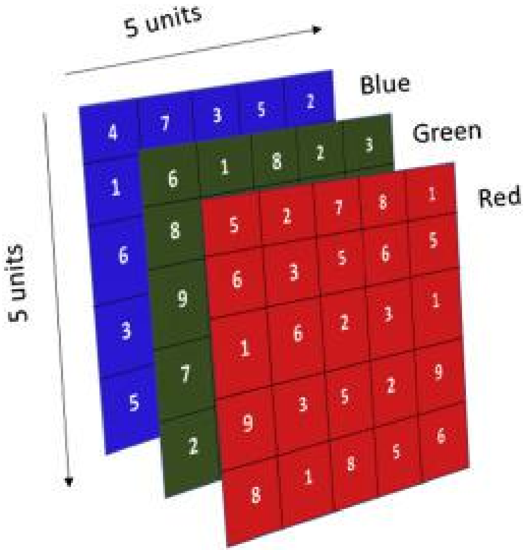
- JPG图像：红、绿、蓝三层矩阵，每个点都是0-255的值
- PNG图像：RGBA四层，A层代表透明程度（0全透明）



What we see

0	3	2	5	4	7	6	9	8
3	0	1	2	3	4	5	6	7
2	1	0	3	2	5	4	7	6
5	2	3	0	1	2	3	4	5
4	3	2	1	0	3	2	5	4
7	4	5	2	3	0	1	2	3
6	5	4	3	2	1	0	3	2
9	6	7	4	5	2	3	0	1
8	7	6	5	4	3	2	1	0

What a computer sees

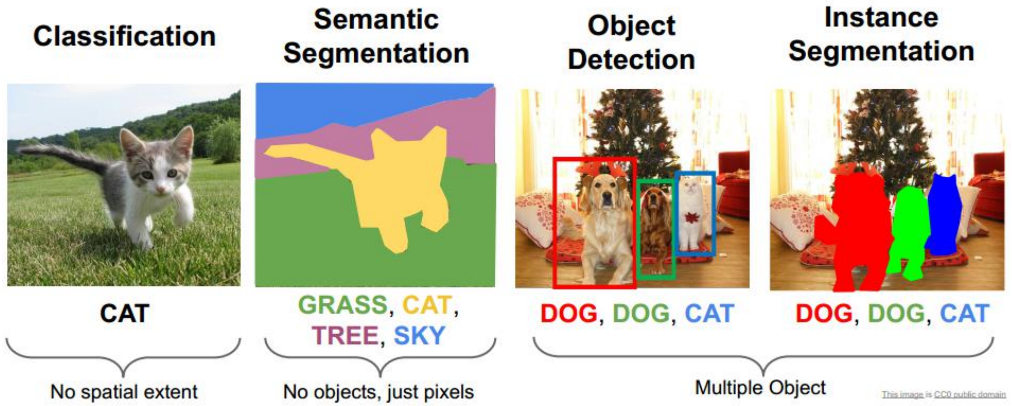


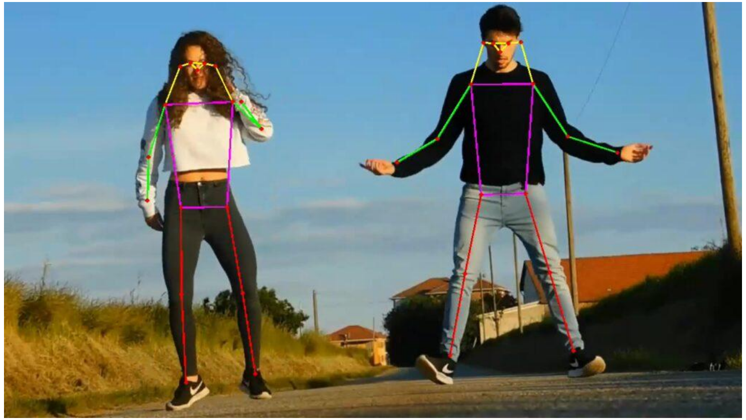
5X5 image matrix

一、基本概念

图像基本任务

- 图像分类（Image classification，AlexNet, VGG）
- 目标检测（Object detection，定位+分类，YOLO，Faster RCNN）
- 语义分割（Semantic segmentation，像素级分类，FCN, Unet）
- 实例分割（Instance segmentation，定位+像素级分类, Mask RCNN, Yolact++）
- 关键点检测 (Keypoint detection)

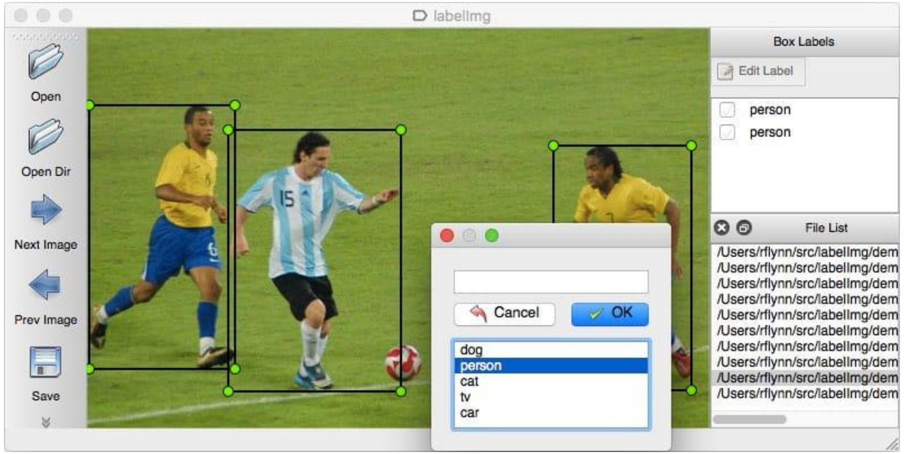


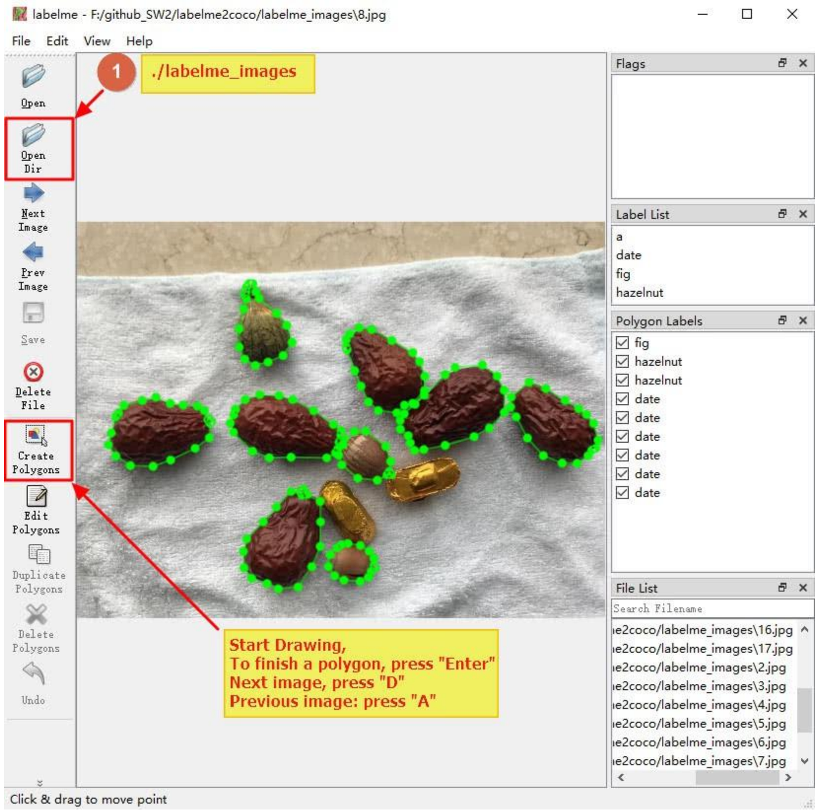


一、基本概念

数据标注

- Labelme: 用于实例分割的标注
- LabelImg: 用于目标检测的标注





一、基本概念

Data Loader (Pytorch)

归一化：

- 直接/255
- 标准化

数据预处理

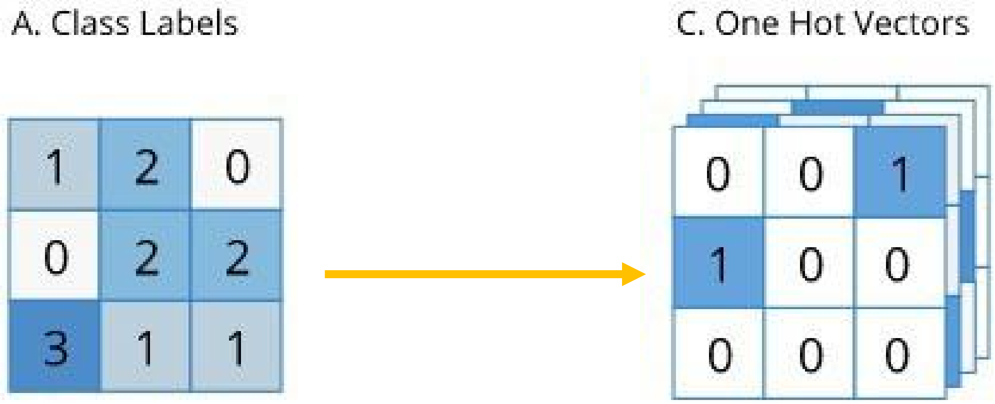
- One-hot编码

Human-Readable

Pet
Cat
Dog
Turtle
Fish
Cat

Machine-Readable

Cat	Dog	Turtle	Fish
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1
1	0	0	0

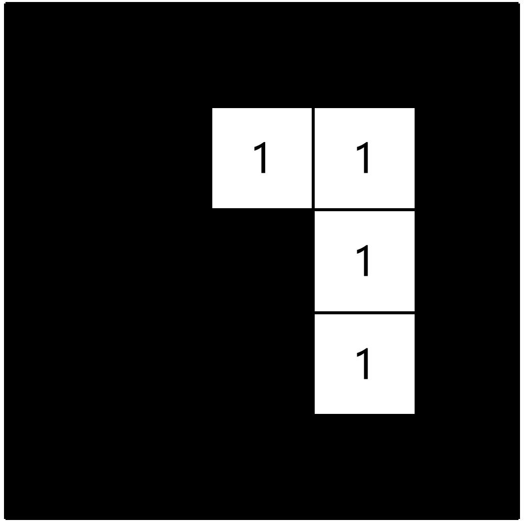


02

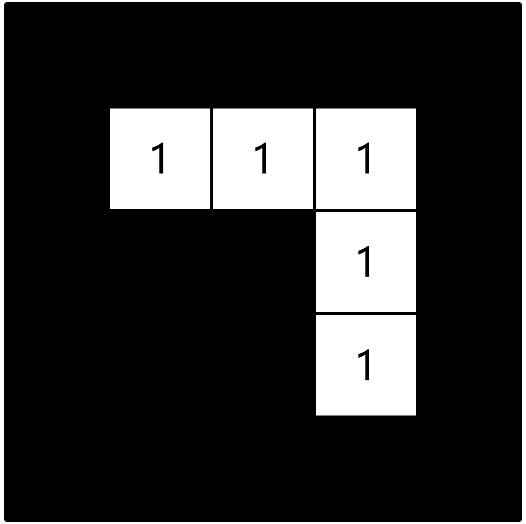
卷积神经网络CNN

二、卷积神经网络CNN

卷积运算



=?

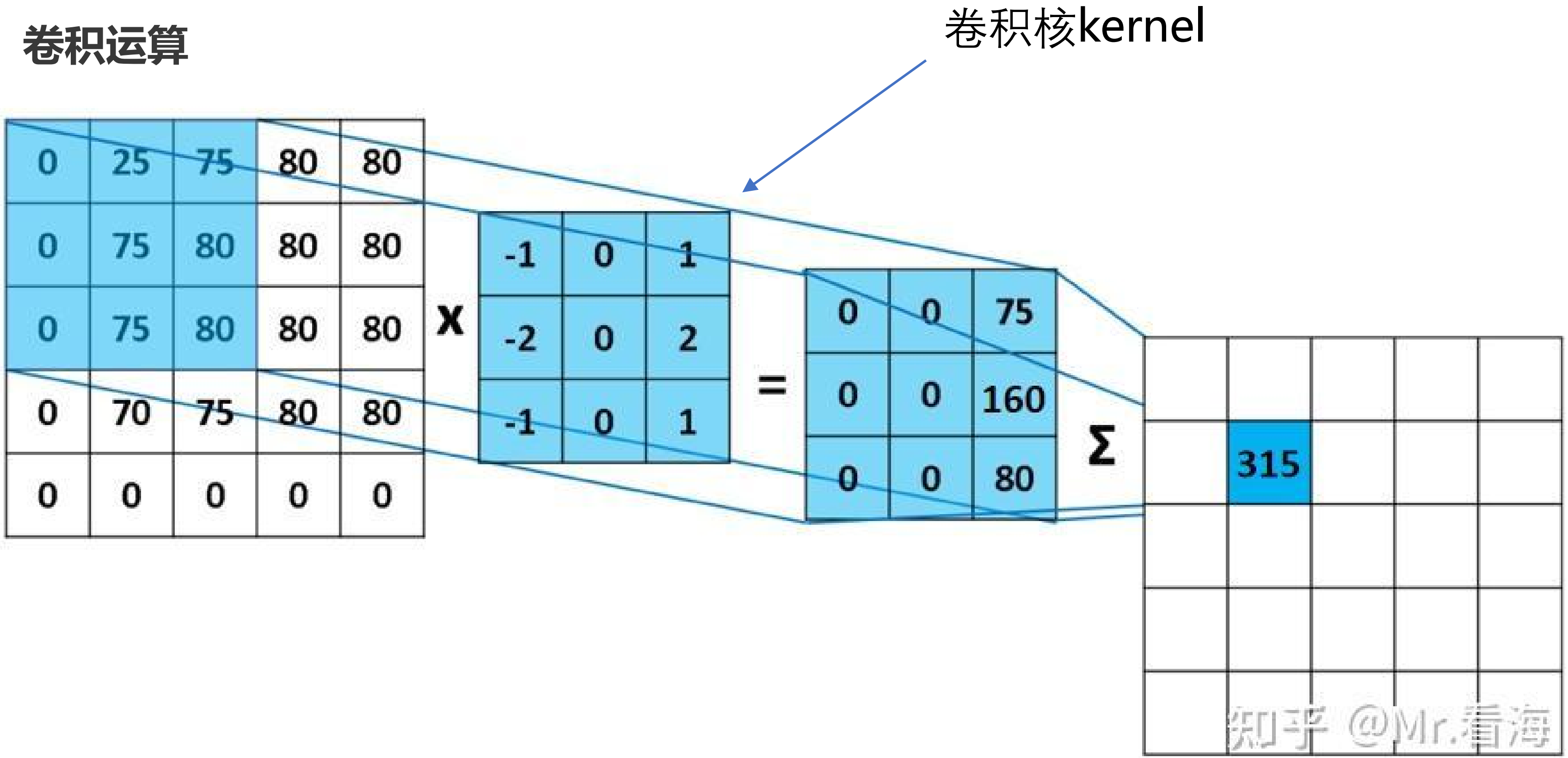


一个个数据判断?

有没有什么指标判断越相似，值越高，反之越低?

二、卷积神经网络CNN

卷积运算



二、卷积神经网络CNN

卷积运算

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \ast \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} = \sum \begin{bmatrix} 0*0 & 1*1 & 0*1 \\ 0*0 & 1*0 & 0*1 \\ 0*0 & 1*0 & 0*1 \end{bmatrix} = 1 \text{（不像）}$$

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \ast \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} = \sum \begin{bmatrix} 1*0 & 1*1 & 1*1 \\ 0*0 & 0*0 & 1*1 \\ 0*0 & 0*0 & 1*1 \end{bmatrix} = 4 \text{（很像）}$$

二、卷积神经网络CNN

卷积运算

$$\begin{bmatrix}1&1&1\\0&0&1\\0&0&1\end{bmatrix} * \begin{bmatrix}0&1&1\\0&0&1\\0&0&1\end{bmatrix} = \sum \begin{bmatrix}1*0&1*1&1*1\\0*0&0*0&1*1\\0*0&0*0&1*1\end{bmatrix} = 4\ \text{（很像）}$$

$$\begin{bmatrix}0&1&1\\0&0&1\\0&0&1\end{bmatrix} * \begin{bmatrix}0&1&1\\0&0&1\\0&0&1\end{bmatrix} = \sum \begin{bmatrix}0*0&1*1&1*1\\0*0&0*0&1*1\\0*0&0*0&1*1\end{bmatrix} = 4\ \text{（很像）}$$

二、卷积神经网络CNN

卷积运算

$$\begin{array}{|c|c|c|}\hline 0 & 1 & 1 \\ \hline 0 & 1 & 1 \\ \hline 0 & 0 & 1 \\ \hline \end{array} \ast \begin{array}{|c|c|c|}\hline 0 & 1 & 1 \\ \hline 0 & -5 & 1 \\ \hline 0 & 0 & 1 \\ \hline \end{array} = \sum \begin{array}{|c|c|c|}\hline 1*0 & 1*1 & 1*1 \\ \hline 0*0 & 1*-5 & 1*1 \\ \hline 0*0 & 0*0 & 1*1 \\ \hline \end{array} = -1 \text{（很不像）}$$

$$\begin{array}{|c|c|c|}\hline 0 & 1 & 1 \\ \hline 0 & 0 & 1 \\ \hline 0 & 0 & 1 \\ \hline \end{array} \ast \begin{array}{|c|c|c|}\hline 0 & 1 & 1 \\ \hline 0 & -5 & 1 \\ \hline 0 & 0 & 1 \\ \hline \end{array} = \sum \begin{array}{|c|c|c|}\hline 0*0 & 1*1 & 1*1 \\ \hline 0*0 & 0*-5 & 1*1 \\ \hline 0*0 & 0*0 & 1*1 \\ \hline \end{array} = 4 \text{（很像）}$$

二、卷积神经网络CNN

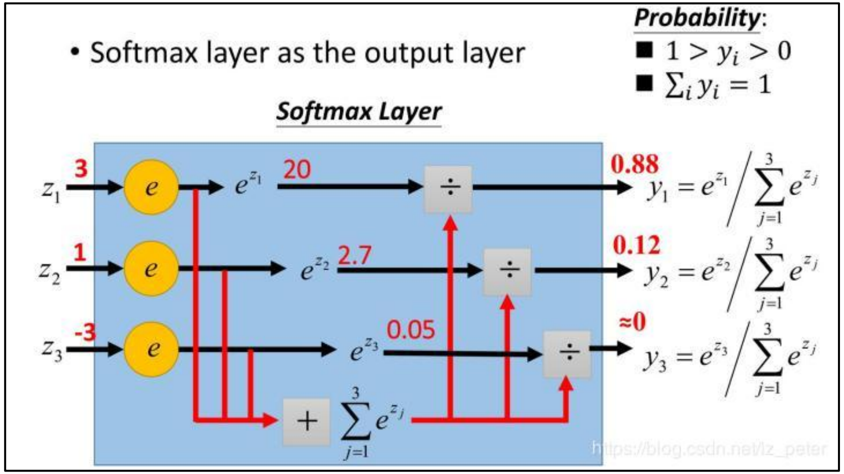
卷积运算

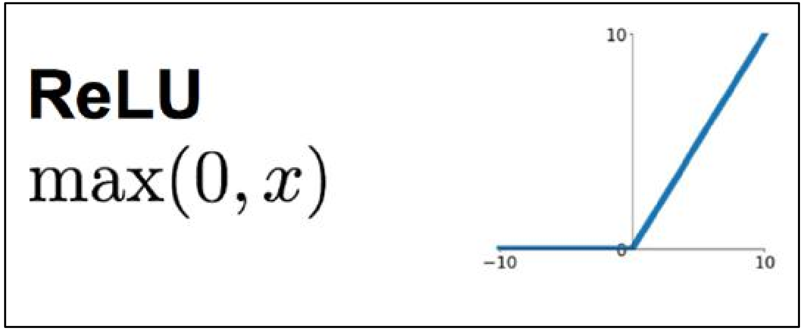
https://bbs.huaweicloud.com/blogs/303478

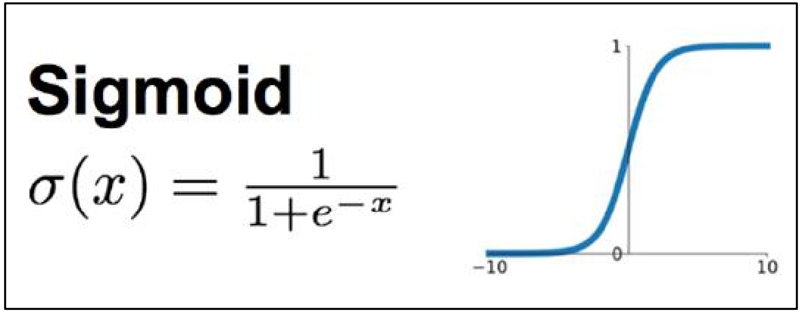
二、卷积神经网络CNN

激活函数

- ReLU最常见，通常卷积之后都要
- Sigmoid（全连接最后一层二分类）
- Softmax（全连接最后一层多分类）



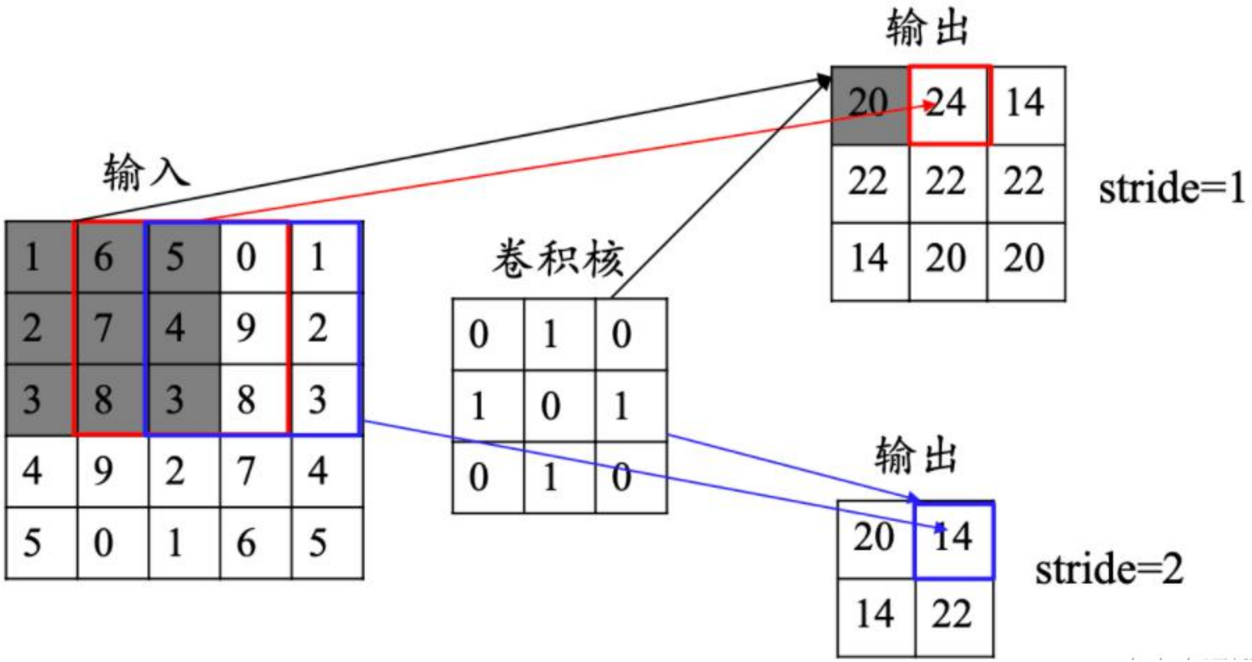




二、卷积神经网络CNN

步长stride

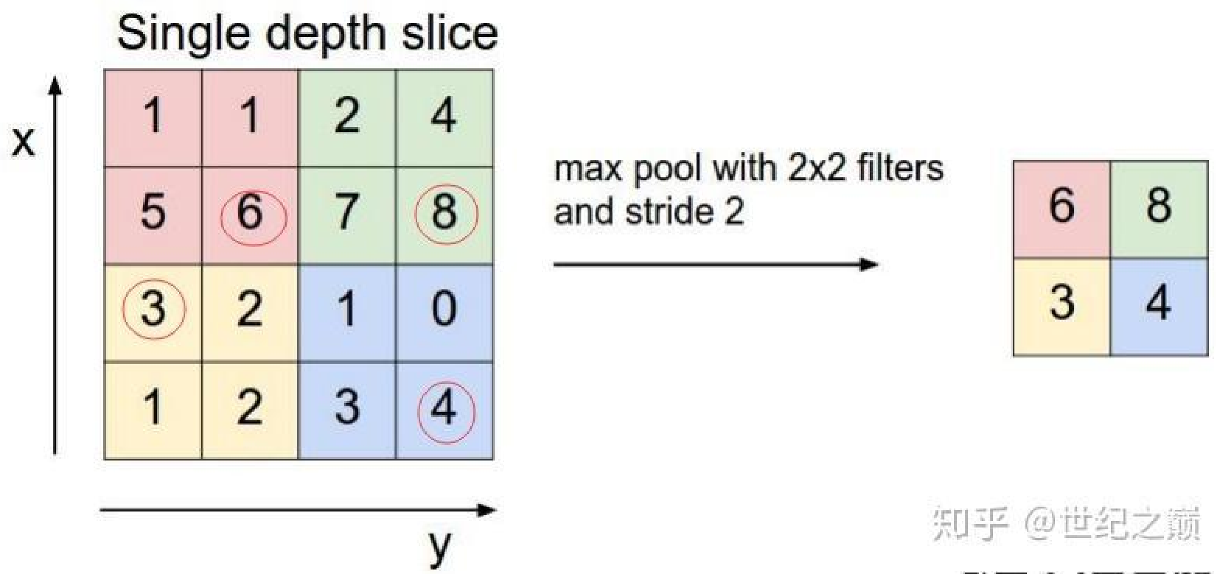
- 要确保图片大小不变，可以补0



二、卷积神经网络CNN

池化Pooling

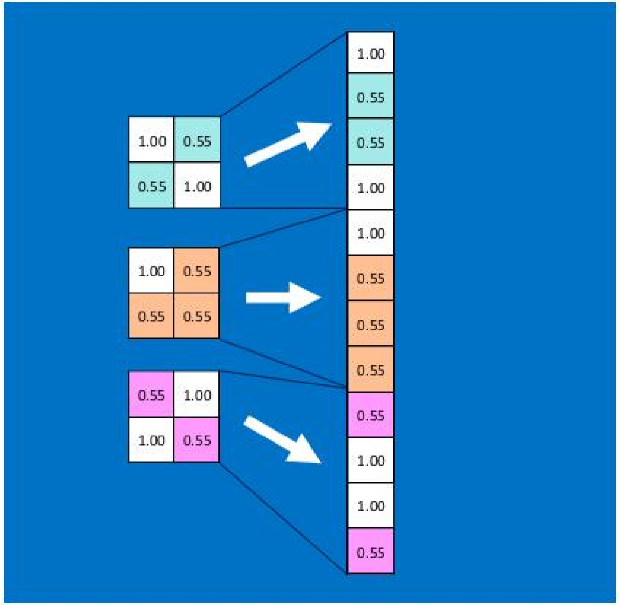
- 过滤低信号，保留高信号
- 让图片不同位置能够“交流”

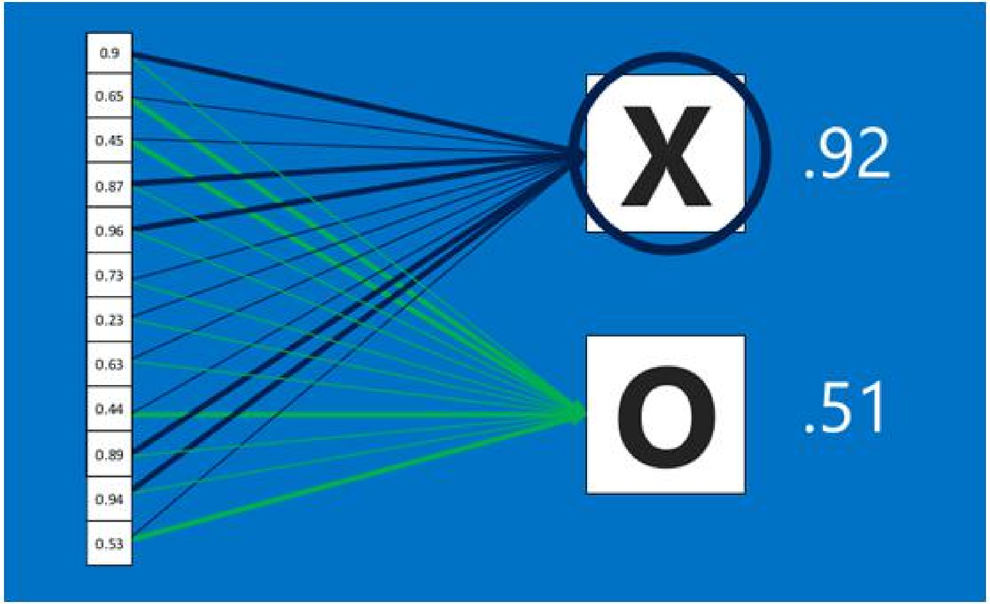


二、卷积神经网络CNN

输出方式（如分类）

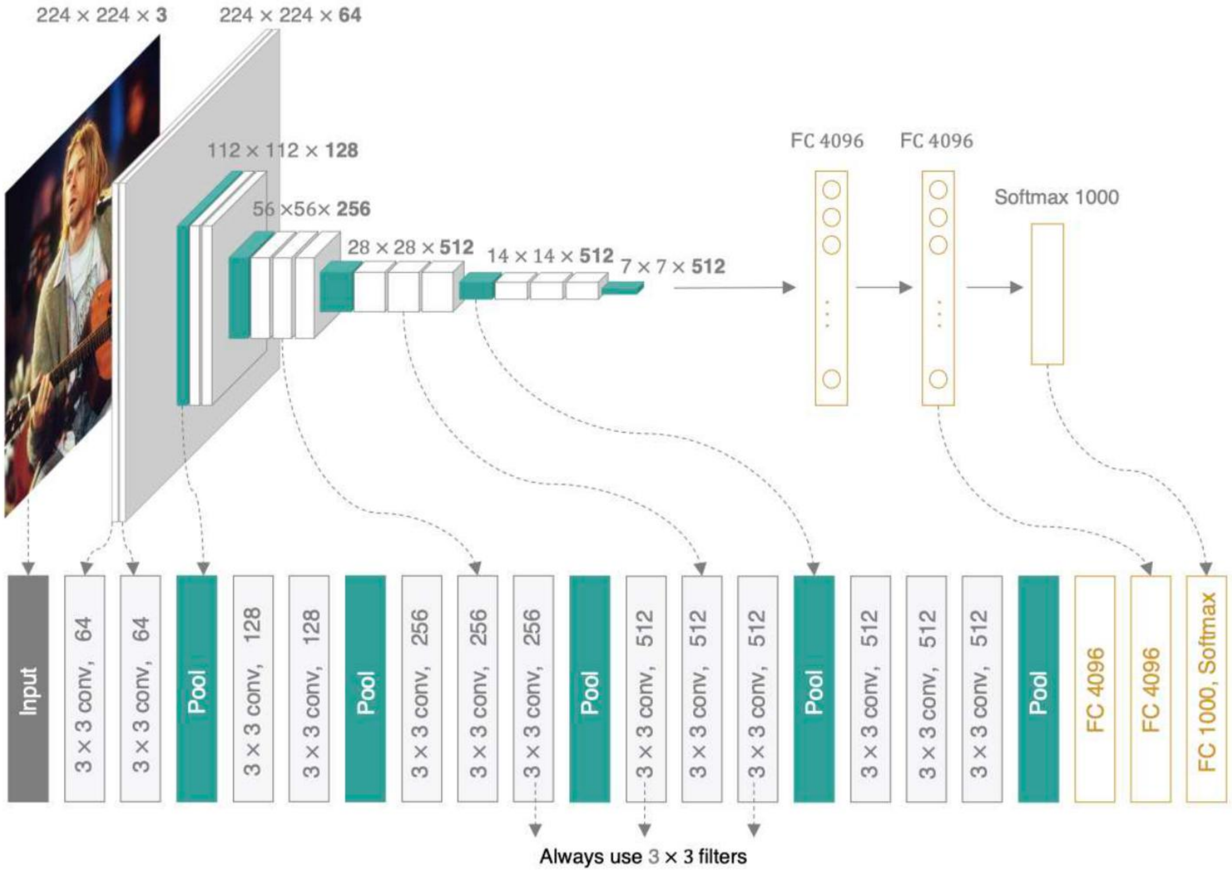
- 拉平
- 普通神经网络分类





二、卷积神经网络CNN

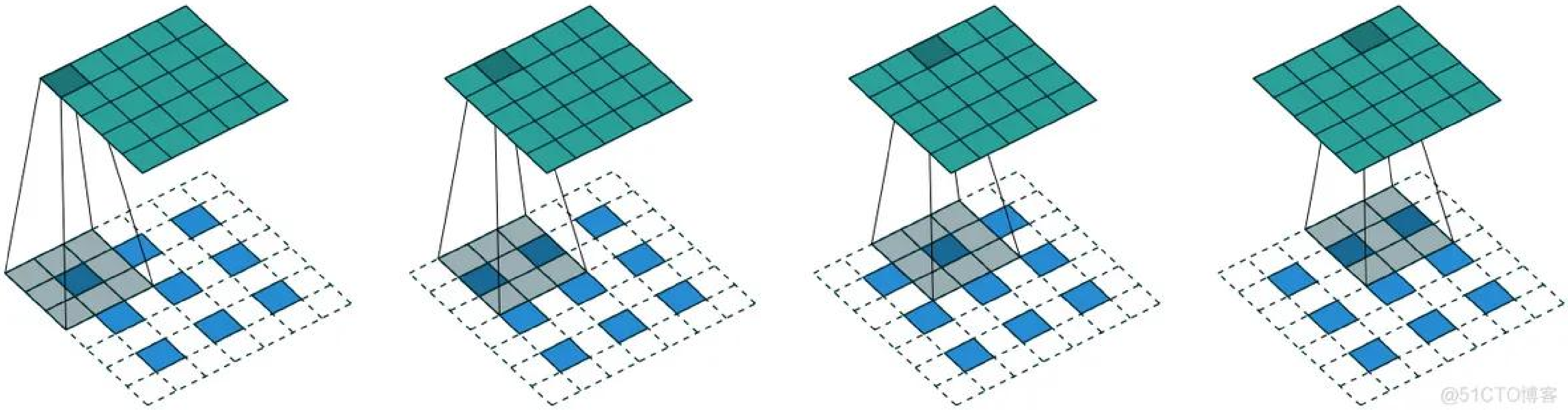
VGG16图像分类



二、卷积神经网络CNN

输出方式（如语义分割）

- 反卷积



二、卷积神经网络CNN

Unet（语义分割）

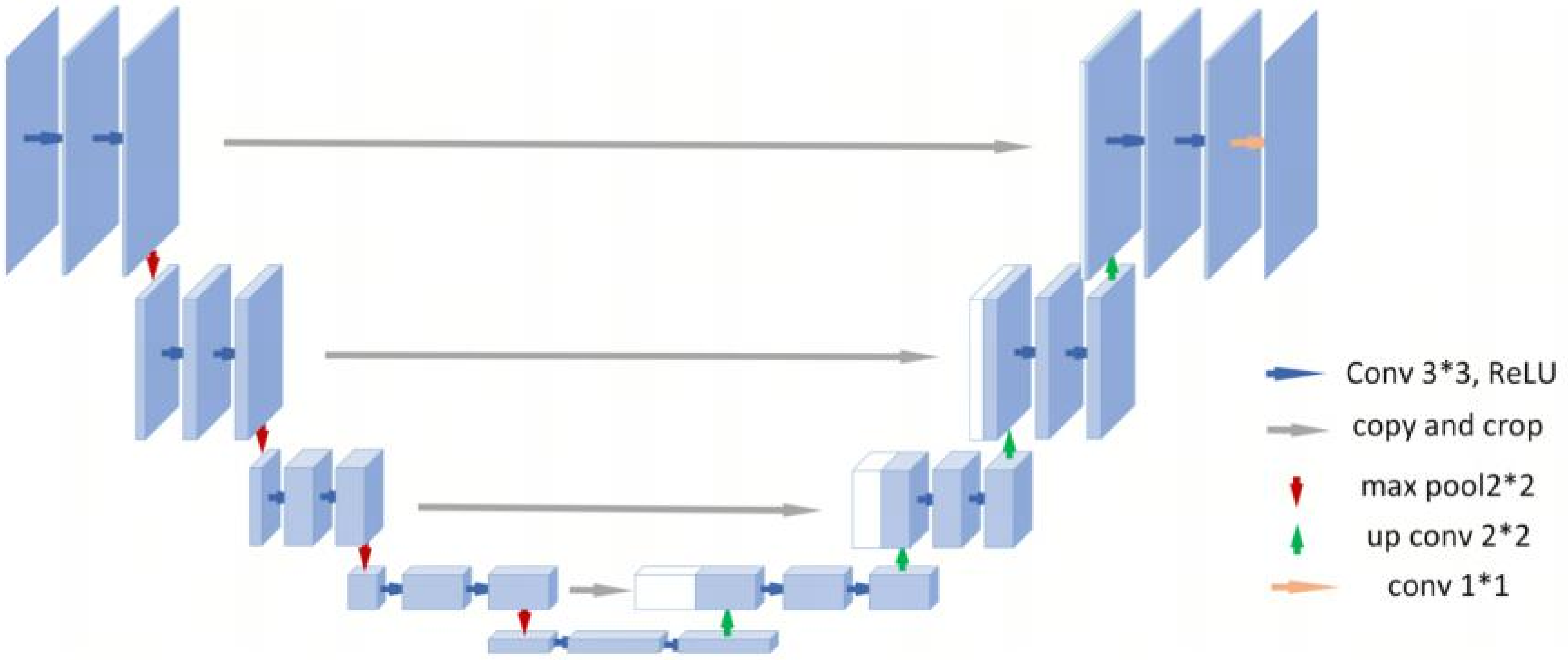


Fig. 1 Structure of U-Net

03

评价指标及常见算法

三、评价指标及常见算法

分类

- 准确率、Recall等

语义分割

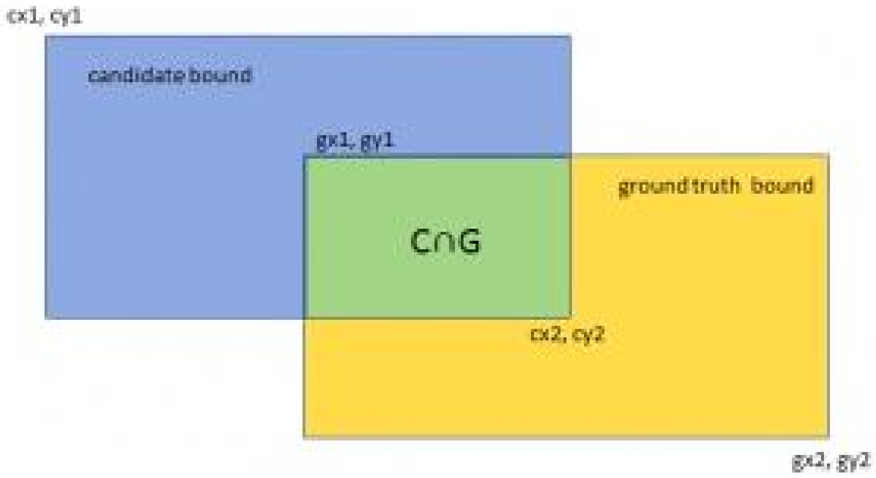
- mIoU

目标检测和实例分割

- mAP

关键点检测

- OKS（Object Keypoint Similarity）



IOU：

$$IoU = \frac{area(C) \cap area(G)}{area(C) \cup area(G)}$$

三、评价指标及常见算法

目标检测

- Faster RCNN: https://github.com/facebookresearch/detectron2
- YOLOv8: https://docs.ultralytics.com/zh

语义分割

- Unet: https://github.com/milesial/Pytorch-UNet/tree/master

实例分割

- Yolact++: https://github.com/dbolya/yolact
- Mask RCNN: https://github.com/facebookresearch/detectron2

三、评价指标及常见算法

- 不需要尝试自己写代码，会跑模型即可
- 重要的是给出类似的数据集
- 标注是最辛苦的
- 优先考虑Unet，简单又效果好

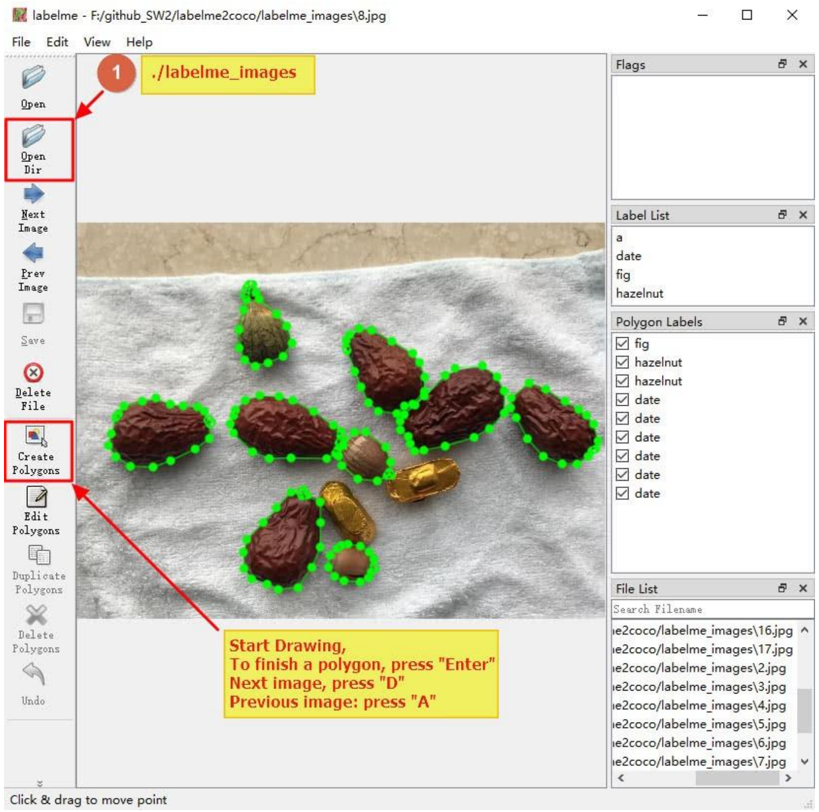
04

数据标注

四、数据标注

labelme

- 多边形（最常用）
- 点（如果做关键点检测）
- 折线（图中的场景）



四、数据标注

labelme

- 多边形（最常用）
- 点（如果做关键点检测）
- 折线（图中的场景）
- 标注文件是json文件（文本文件）


```
{
  "version": "5.4.1",
  "flags": {},
  "shapes": [
    {
      "label": "line",
      "points": [
        [
          5.128593040847198,
          84.35552193645991
        ],
        [
          9.818456883509883,
          68.92435703479576
        ],
        [
          0.0,
          57.22884221767376
        ]
      ],
      "group_id": null,
      "description": "",
      "shape_type": "linestrip",
      "flags": {},
      "mask": null
    },
    {
      "label": "line",
      "points": [
        [
          9.818456883509883,
          68.47049924357034
        ],
        [
          34.78063540090771,
          64.23449319213313
```

THE END

THANKS

黄恩待

智能医学与生物医学工程研究院

huangendai@nbu.edu.cn