

课程关键知识点总结

Lecture 1: Introduction

1. 计算机视觉概述：数据驱动的视觉任务解决方法，核心问题（如图像分类、检测、分割等）。
2. 课程概述：深度学习在计算机视觉中的应用框架，重点模块（CNN、RNN、Transformer 等）。

Lecture 2: Image Classification with Linear Classifiers

数据驱动方法：从训练数据中学习模型参数，而非手工设计特征。

K - 最近邻 (KNN) 算法：基于样本相似度的分类方法，距离度量（如 L2 距离）与超参数调优。

线性分类器：

1. 代数视角：权重矩阵与输入的线性变换 ($f(\mathbf{x}_i, \mathbf{W}) = \mathbf{W}\mathbf{x}_i + \mathbf{b}$)。
2. 几何视角：权重向量定义分类超平面，样本到平面的距离决定类别。

Softmax 损失函数：将线性输出转换为概率分布，公式为 $L_i = -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right)$ 。

MLE 与 MAP 估计：最大似然估计 (MLE) 与最大后验概率估计 (MAP) 的原理及应用场景。

维度灾难：高维空间中数据稀疏性对分类性能的影响，如 KNN 在高维下的局限性。

Lecture 3: Regularization and Optimization

正则化方法：

1. 防止过拟合的核心技术：Dropout（随机失活神经元）、早停法（监控验证集提前终止训练）、数据增强（旋转、缩放等扩充样本）。

梯度下降算法：

2. 批量梯度下降 (BGD)：基于全量数据计算梯度，收敛稳定但计算量大。
3. 随机梯度下降 (SGD)：基于单样本更新，适合大数据场景但收敛波动大。
4. 小批量梯度下降 (Mini-Batch SGD)：平衡计算效率与收敛稳定性。

优化器与学习率调度：

5. 动量 (Momentum)、AdaGrad、Adam：自适应调整学习率的策略。
6. 学习率衰减：过大导致不收敛，过小导致训练停滞，常见衰减方式（如指数衰减、余弦衰减）。

K-means 聚类：无监督学习中的聚类算法，基于距离划分数据簇，初始化与迭代优化流程。

Lecture 4: Neural Networks and Backpropagation

神经元结构与前向传播：

1. 多层感知机 (MLP) 的基本单元，输入输出线性变换与激活函数的组合。
2. 线性激活函数的局限性：无法拟合非线性关系。

反向传播算法：

3. 基于链式法则的梯度计算，权重更新公式 ($W \leftarrow W - \alpha \nabla W$)。
4. 计算图视角：通过反向传播高效求解各层梯度。

激活函数：

5. ReLU (修正线性单元)：解决梯度消失问题，公式为 $f(x) = \max(0, x)$ 。
6. Sigmoid：历史常用激活函数，但存在梯度饱和问题。

权重初始化：

7. 全零初始化导致神经元同质化，常用方法（如 Xavier 初始化、Kaiming 初始化）。

Lecture 5: Image Classification with CNNs

卷积层计算：

1. 输入尺寸、卷积核大小、步长 (Stride)、填充 (Padding) 对输出尺寸的影响公式：

$$\text{输出尺寸} = \left\lfloor \frac{W - K + 2P}{S} \right\rfloor + 1。$$

2. 卷积操作的物理意义：提取局部特征（如边缘、纹理）。

池化层：

3. 最大池化：保留局部区域最大值，实现平移不变性。
4. 平均池化：降低计算量，对噪声更鲁棒。

激活层：如 ReLU，为网络引入非线性能力。

数据增强：预处理阶段通过旋转、缩放、翻转等操作扩充训练数据，提升模型泛化能力。

Lecture 6: CNN Architectures

Batch Normalization（批归一化）：

1. 固定各层输入分布，加速训练并缓解梯度消失问题。

经典架构：

2. VGGNet：通过小卷积核（ 3×3 ）堆叠减少参数数量，加深网络层次。
3. ResNet：残差块设计（ $y = x + F(x)$ ），缓解深层网络梯度消失问题。
4. AlexNet、GoogLeNet：早期 CNN 里程碑模型的核心创新（如 ReLU、Inception 模块）。

模型深度与性能的关系：过深网络可能导致优化困难（如梯度消失），需结合残差连接等技术改进。

Lecture 7: Recurrent Neural Networks (RNN)

RNN 结构：

1. 处理序列依赖问题（如文本、视频），隐藏状态传递历史信息。

LSTM 与 GRU：

2. LSTM 门控机制：遗忘门、输入门、输出门，解决长序列梯度消失问题。
3. GRU：LSTM 的简化版本，合并遗忘门与输入门为更新门。

应用场景：

4. 语言建模、图像字幕生成、序列到序列任务（如机器翻译）、视频对象追踪。

Lecture 8: Attention and Transformers

Self-Attention 机制：

1. 通过计算输入序列中各元素的关联权重（Query-Key-Value），捕捉长距离依赖关系。
2. 公式：
$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Transformer 架构：

3. 核心组件：多头注意力（Multi-Head Attention）、前馈神经网络（FFN）、位置编码。

4. 优势：并行计算能力强，解决 RNN 长序列处理效率低的问题。

视觉应用扩展：

5. ViT (Vision Transformer)：将图像分块后直接应用 Transformer，打破传统 CNN 架构限制。

Lecture 9: Object Detection, Image Segmentation

目标检测：

1. 单阶段检测器（如 YOLO、SSD）：端到端直接预测边界框与类别。
2. 两阶段检测器（如 R-CNN、Fast R-CNN、Faster R-CNN）：先生成候选区域再分类。

图像分割：

3. 语义分割（像素级分类）、实例分割（区分同一类别的不同实例）、全景分割（兼顾语义与实例）。

特征可视化与理解：

4. 可视化卷积层激活值，解释模型决策依据；对抗样本（刻意设计的输入使模型误判）。

Lecture 10: Video Understanding

1. 视频分类：识别视频中的动作或事件，结合时间维度特征。
2. 3D CNNs：通过 3D 卷积核同时提取空间与时间特征（如 C3D 模型）。
3. 两流网络：分离空间流（处理单帧图像）与时间流（处理光流信息），提升动作识别性能。
4. 多模态视频理解：融合视觉、音频等多模态信息进行联合建模。

Lecture 13: Generative Models 1

1. 变分自编码器 (VAE)：

5. 基于概率图模型，通过编码器将数据映射到隐空间，解码器从隐空间重构数据，引入 KL 散度约束隐变量分布。

1. 生成对抗网络 (GAN)：

6. 生成器与判别器的对抗训练框架，生成器学习拟合真实数据分布，判别器区分真实与生成样本。

1. 自回归模型：

7. 基于序列依赖关系，通过历史输出预测下一时刻值（如 PixelRNN、PixelCNN），用于图像生成。