Ch1: Introduction to computer vision

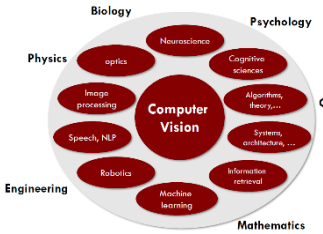
从人类视觉系统学习到的计算机视觉系统的理念

0. 什么是计算机视觉?计算机视觉是人工智能(AI)的一个领域,计算机和系统从数字图像,视频和其他视觉输入中获取有意义的信息,并采取相应行动或做出决策.(See, Understand, Response)

1. (KP1)人类视觉系统:人类的多目视觉:3D 重建中的多相机视角;看事物时具有多尺度的特点:SIFT,图像金字塔,捕捉图像多尺度特征;**人眼视觉注意力机制:**Atten 机制在 CV 中的广泛应用,SA:**人类对于形状的把握:**语义分割的方法,标记轮廓,具有极高的效率;利用先验知识;视觉错觉(illusion):人类视觉感知->计算机视觉方法;层次结构化->多尺度融合;仅显著性->显著性检测+注意力机制

2. 计算机视觉覆盖内容:图像增强:背景模糊\超分辨率重建\降噪\阴影去除;图像编辑:风格迁移\图像生成\图像修复;检测识别:目标检测\动作识别\语义分割;造型和动作捕捉:3D 结构建模\姿态识别\对象追踪;

3. (KP2)构建一个 CVS 相关的领域知识:构建一个自动驾驶系统需要涉及的知识



4. (KP3)计算机视觉的应用:人脸检测(数字相机)\面部识别\生物特征识别(指纹识别)\OCR\特效\3D 建模\VR\AR;Laptop:生物特征识别\OCR\3D 面容识别\;Smartphone:二维码扫描\计算摄影\情绪检测;Web:图像搜索\图像标题;Medical Imageing: CAT\MRI 重建\辅助诊断

5. (KP4)计算机视觉面临的挑战:视角变换,光照差异,遮挡,背景混淆,尺度差异,变形,类内差异(细粒度类别),局部模糊性.

6. 计算机视觉系统的工作流:视觉捕获(光学摄影机,激光扫描···)->数据预处理(二次采样,平滑降噪,对比度增强,调整尺度空间)->特征提取(从图像中提取多种复杂度的特征)->定制化的处理(根据任务本身定义:检测,分割···)->后处理

7. 出勤检测系统:face detection -> face alignment->feature extraction -> matching\recognition

Ch2:Filters

0. 图像种类:二值图(0 黑 1 白)\灰度图[黑 0,255 白]\彩色图(RGB 色彩空间,HVS 色彩空间[色调,饱和度,值即强度])

1. (KP1)卷积基础:计算对应元素之积之和(不反转);**性质**:移位不变性(Shift Variant),输出值取决于图像邻位的模式,而不是图像邻位的位置;满足交换律\结合律\分配律

高斯核性质:A. 从图像中去除高频成分;**B.** 和核自己进行卷积仍为另一个高斯核: a. 可用小σ卷积核重复卷积以得与大σ核卷积相同的结果; b. 和标准差为σ的核卷积两次结果与和$\sqrt{2}\sigma$核卷积一次结果相同; **C.**可分离的核: 可以拆成 2 个 1D 高斯,分步运算降低复杂度 **D.** σ越大模糊力度越大

核的可拆分性:图像$n \times n$,核$m \times m$,直接卷积复杂度$O(n^2m^2)$;拆分后$O(n^2m)$

2. 图像滤波: 对图像中的每一个像素根据其局部的邻居依照一个 function(定义了如何结合邻居的值)进行计算;目的:从图像中提取有用信息、或修改/增强图像属性

常见滤波器(3 × 3 为例):值为 1/9- 求平均值,模糊;仅中心元素为 1-不变;仅[2,3]处为 1-图像左移 1 个像素

(KP2)噪声种类: A.椒盐噪声:随机出现白色(255)像素和黑色(0)像素; B. 脉冲噪声(Impulse Noise)随机出现白像素;C. 高斯噪声:从高斯正态分布中产生的噪声

(KP3)用滤波应对噪声

特点	高斯滤波(线性)	中值滤波(非线性)
原理	高斯核(和为 1)	领域中值
适用	高斯噪声	椒盐噪声
优点	简单高效	保护边缘信息,用邻域内真实值进行填充
缺点	模糊边缘,丢失特征	非线性,速度慢(对邻域排序)

3.(KP4)锐化:原始图像-平滑图像得高频细节,原始图像+高频细节得锐化后图片;具体实现:仅中心元素为 2 的滤波器-值全为 1/9 滤波器

Ch3:边缘检测

0. 边缘检测: 检测图像中突变(不连续)之处.应用:识别物体\恢复几何和视角.

边缘是图像强度函数中发生迅速变化之处

(KP2)边缘滤波器:

A. Prewitt:

$$G_x=\begin{bmatrix}-1&0&1\\-1&0&1\\-1&0&1\end{bmatrix} \qquad G_y=\begin{bmatrix}1&1&1\\0&0&0\\-1&-1&-1\end{bmatrix}$$

B. Roberts

$$G_x=\begin{bmatrix}0&1\\-1&0\end{bmatrix} \qquad G_y=\begin{bmatrix}1&0\\0&-1\end{bmatrix}$$

C. Sobel: 相当于同时实现了差分和高斯滤波(拆分为[1,2,1]*[1,0,-1]). **缺点**:定位差,对梯度的方向有偏好(水平\竖直),可能忽略斜边缘,导致 False Negative.

$$G_x=\begin{bmatrix}-1&0&1\\-2&0&2\\-1&0&1\end{bmatrix} \qquad G_y=\begin{bmatrix}1&2&1\\0&0&0\\-1&-2&-1\end{bmatrix}$$

在滤波方式上,所有算子计算梯度的方式都是求平方根;但是 Roberts 计算 x 轴梯度没有那么方便;Roberts 算法速度比 Sobel 快,能应对频率更高的边缘,但是对于噪声更加敏感. 使用 Roberts 算子计算梯度方向来的角度要减去$\pi/4$;**在实现效果上,**Roberts 算子定位比较精确,但由于不包括平滑操作,所以对噪声比较敏感. Prewitt 算子和 Sobel 算子都是一阶的微分算子,前者是平均滤波,后者是加权平均滤波且检测的图像边缘可能大于 2 个像素.这两者对于灰度渐变低噪声图像检测效果较好,对于混合多复杂噪声的图像处理效果不理想.

$|G_x|, |G_y|$ of center pixel,判断边缘方向

边缘检测的评价指标:好的信噪比,即将非边缘点判定为边缘点概率要低;高的定位性能,即检测到的边缘点要尽可能在实际边缘中心;对单一边缘仅有唯一相应,即单个边缘产生多个响应概率要低,并且虚假响应边缘应该得到最大的抑制.

1. (KP1)图像梯度:方向$\theta=\tan^{-1}\left(\frac{\partial f}{\partial y}/\frac{\partial f}{\partial x}\right)$;边缘强度$\|\nabla f\|=\sqrt{\left(\frac{\partial f}{\partial x}\right)^2+\left(\frac{\partial f}{\partial y}\right)^2}$;图像梯度方向和边缘方向**垂直**

应对图像中的噪声:先平滑再检测边缘;先卷积平滑再对结果求导与先对图像求导再卷积平滑效果相同->DOG(Derivative of Gaussian Filter,和为 0),节省求导操作(大σ检测大尺度边缘,小σ检测细粒度边缘)

2. (KP3)Canny 边缘检测器:

步骤:A. 图像灰度化处理; **B.** 对图像进行高斯滤波降噪处理(平滑). 高斯核由两个参数决定(核的 size 和σ)高斯核上每一个位置的值由$g(x,y)=\frac{1}{2\pi\sigma^2}e^{-\frac{(x^2+y^2)}{2\sigma^2}}$计算, 当$\sigma$不变时,随着 size 增大,平滑范围变大,图像模糊程度增加;当 size 不变时,随着σ增大, 图像模糊程度也增加(原因:噪声与邻域像素非常不同,有限差分算子对这种不同很敏感);**C.** 用一阶偏导的有限差分算子计算每一个像素点梯度的幅值和方向, 这里用 Sobel 算子. [B.C 通常合为一步,直接使用高斯差分核 DoG 与图像滤波]

D. 非极大抑制(应对 thick egdes). 对梯度幅值进行非极大抑制,即寻找像素点梯度的局部最大值.沿像素点梯度方向比较前后像素梯度值.为此,需要将梯度角度θ设定为$\{-\pi,\pi]$, 并近似到 4 个方向(水平\垂直\两个对角线方向,0\45°\90°\135°)以$\pm i\pi/8(i=1,3,5,7)$分割,落在每个区域的梯度角给一个特定值代表四个方向之一;前后像素的值可以根据θ就近取八邻域的像素值,也可以是插值的结果.**E.** 用双阈值法检测和连接边缘(应对 discontinuous egdes). 梯度值大于高阈值的边缘像素点被认为是强边缘点,保留;小于高阈值大于低阈值的点标记为弱边缘点;小于低阈值的点则被抑制掉. 弱边缘点如果最后与强边缘点连通则被认为是强边缘点,不连通则归类为非边缘点.

参数影响: 对于σ,大的值可以筛选出明显的边缘;小的值可以筛选出更多的边缘(edges scale). 双阈值的值的选取会决定最终选择出来的边缘效果(edges kept)

优缺点:优点: 简单容易实现,可以处理缺失数据,泛化性高。对于缺失数据可以处理(理解是这种依赖局部特征的基本都不care 缺失值); **缺点:** 计算量大,速度慢: 需要进行多个步骤,包括高斯滤波、梯度计算、非极大值抑制、双阈值检测和边缘连接等; 需要手动设置两个阈值参数, 影响检测效果(阈值过高会导致漏检真实边缘, 阈值过低会导致误检非边缘。)

4. 基于深度学习的边缘检测方法:多尺度特征学习; 传统的像素差分算子+CNN

Ch4 Local Feature-Corners

0. 局部特征: Motivations: 全局特征有较大的局限性;描述和匹配局部特征可以提升对有遮挡物和同类别变种(Intra-category variations)的鲁棒性.

Main components: A.检测.找到一个与众不同的 keypoints(关键点)集合 **B. 描述.** 从每一个 keypoint 周围提取向量特征描述符. **C. 匹配:**决定两个视图下特征描述符的对应性(计算不同描述子之间的相关性).

[例]有两张雪山图片,如何结合他们? Step1:关键点检测+提取局部特征. Step2: 匹配特征. Step3:对齐图像

特性:紧凑,高效表示: 特征数远小于像素数量; 显著性: 每一个 keypoint 都是与众不同的; 局部性: 每个特征只占据一块相当小的区域,对遮挡和杂乱场景更鲁棒; 重复性:相同的特征可以在几张图像中都被找到,尽管存在几何和光照变换.

1. (KP1)角点: 在角点周围区域, 图像梯度有两个或多个主要方向.(在角点沿任何方向移动 window 都会引起强度剧烈变化). 形式化表达:window $w(x,y)$ for the shift $[\mu,v]$

$$E(\mu,v)=\sum_{(x,y)}w(x,y)[I(x+\mu,y+v)-I(x,y)]^2$$

2. (KP2)用二阶动量矩阵估计$E(\mu,v)$

$$\begin{aligned}E(u,v)&=\sum_{(x,y)}w(x,y)[I(x+u,y+v)-I(x,y)]^2\\&\approx\sum_{(x,y)}w(x,y)\left[I(x,y)+\frac{\partial I}{\partial x}(x,y)u+\frac{\partial I}{\partial y}(x,y)v-I(x,y)\right]^2\\&\approx\sum_{(x,y)}w(x,y)\left[\frac{\partial I}{\partial x}(x,y)u+\frac{\partial I}{\partial y}(x,y)v\right]^2\text{(消除重复项)}\\&=\sum_{x,y}w(x,y)\left(u^2f_x^2(x,y)+2uvf_x(x,y)f_y(x,y)+v^2f_y^2(x,y)\right)\\&=\sum_{x,y}w(x,y)(u^2I_x^2+2uvI_xI_y+v^2I_y^2)\text{(简化)}\\&=[u\ v]\left(\sum w(x,y)\begin{bmatrix}I_x^2&I_xI_y\\I_xI_y&I_y^2\end{bmatrix}\right)\begin{bmatrix}u\\v\end{bmatrix}\\&=\begin{bmatrix}u\\v\end{bmatrix}^T H\begin{bmatrix}u\\v\end{bmatrix}\end{aligned}$$

计算 H:以选中矩阵每个像素作为中心点,为每个像素计算梯度

Harris 矩阵 H 的特征值分析: 对平移旋转 co,对尺度 not co

两个特征值反应相互垂直的方向上的变化情况,分别代表变化最快和最慢的方向(值大快小慢);$SVD(H)=U\sum V\rightarrow(\lambda_1,\lambda_2),\lambda_1>\lambda_2.\ \lambda_1\approx\lambda_2\approx 0$,两个方向上变化都很小,兴趣点位于光滑区域(Flat region);$\lambda_1>0,\lambda_2\approx 0$, 一个方向变化快,一个方向变化慢, 兴趣点位于边缘区域(Edge);$\lambda_1,\lambda_2>0$,两个方向变化都很快,兴趣点位于角点区域

角点响应函数:用 Harris 角点准则代替矩阵分解(需计算矩阵 H 行列式和迹)

$$R=det(H)-\alpha\text{trace}\,(H)^2=\lambda_1\lambda_2-\alpha(\lambda_1+\lambda_2)^2,\alpha\in[0.04,0.06]$$

判定准则: R 较大负值-边缘;较大正值-角点;倾向于 0 平滑区

性质: 参数α:增大α值,将减小角点响应值 R,减少被检测角点的数量;减小α值,将增大角点响应值 R,增加被检测角点的数量

Harris 角点检测对亮度和对比度的变化不敏感.

Harris 角点检测角点位置在平移和旋转变换前后具有协变性,响应函数值具有不变性; 响应函数值在尺度变换前后不具有不变性, 角点位置也不具有协变性.小尺度下的角点被放大后可能被认为是图像边缘.

不变性和协变性:如果变换前后的响应函数值一致则为不变性;如果提取的特征经过变换(先提取后变换)和原图先变换再提取特征的点能对应的上则为协变性.

3. (KP3)Harris 角点检测步骤: A.计算每个像素的偏导数. **B.** 在高斯 window 范围里计算每个像素周围的第二动量矩阵H. **C.** 计算角点响应函数R. **D.** 确定阈值 R, 小于阈值的就不检测是否为图像中角点. **E.**求响应函数的局部极大值(非极大值抑制)

Ch5 Local features – Blob detection

1. 斑点检测思路: 构造一个 response-scale 函数,找到可以让这个函数取得局部最大值的 region size. 这个 region size 下"框选"的就是检测到的斑点

动机:在同一张图像的不同尺度的版本里独立检测对应区域

2. (KP1)尺度空间的斑点检测

LoG 斑点检测:$LoG=\sigma^2(G_{xx},G_{yy}),G_\sigma=\frac{1}{2\pi\sigma^2}e^{-\frac{(x^2+y^2)}{2\sigma^2}}$

步骤: A. 让图像和尺度归一化 LoG 算子

$\nabla^2_{\text{norm}}g=\sigma^2\left(\frac{\partial^2 g}{\partial x^2}+\frac{\partial^2 g}{\partial y^2}\right)$ 在几个尺度下进行卷积. **B.** 在尺度空间下找到平方拉普拉斯响应的局部(响应幅值要大于所有的 26 个邻域点的响应幅值)最大值(对应的尺度);尺度选择:当斑点半径$r=\sqrt{2}\sigma$时响应最大,这个半径为 LoG 算子的零平面半径(令 LoG=0).

对 LoG 进行尺度归一化的原因: 避免随着σ增大响应值趋于 0; 物理解释:卷积运算表达了随着一个函数在另一个函数上移动时的重叠量, 由于这里运算中卷积核大小没变,让重叠量达到最大对应的尺度就是斑点的大小.

(KP2)LoG 算子的近似: 对 LoG 的高效实现

$$\begin{aligned}DoG=G(x,y,k\sigma)-G(x,y,\sigma),G(x,y,k\sigma)-G(x,y,\sigma)&\approx(k-1)\sigma^2\nabla^2G\\&=(k-1)LoG(\sigma)\end{aligned}$$

性质:对放缩和旋转具有(响应值)不变性和(斑点位置)协变性

区分两种 DoG 算子:A. (Difference of Gaussians) $DoG=G(x,y,k\sigma)-G(x,y,\sigma)$ 是对 LoG 的估计和逼近,零点用于边缘检测,最大值用于斑点检测; **B.** (Derivative of Gaussian) $G_x=\frac{\partial G}{\partial x}=G*[-1\ 1],G_y=\frac{\partial G}{\partial y}=G*\begin{bmatrix}-1\\1\end{bmatrix}$用于边缘检测

3. (KP3)SIFT(Scale Invariant Feature Transform):关键点描述子

简介: 既包含特征检测又包含特征描述. SIFT 算子是把图像中检测到的特征点用一个 128 维的特征向量进行描述, 因此一幅图像经过 SIFT 算法后表示为一个 128 维的特征向量集.SIFT 算法的实质是在不同的尺度空间上查找关键点(特征点),并计算出关键点的方向.SIFT 查找到的关键点是一些十分突出,不会因光照\仿射变换和噪音等因素而变换的点及亮区的暗点等。SIFT 特征对旋转、尺度缩放、亮度变化等保持不变性, 对于光照、仿射和投影变换也有一定的不变性, 是非常稳定的局部特征.

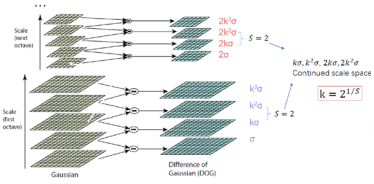
SIFT 流程:A.多尺度极值检测:搜索所有尺度上的图像位置,通过高斯微分函数来识别潜在的对于**尺度**和**旋转不变**的兴趣点.

B.特征点精确定位:在每个候选的位置上,通过一个拟合精细的模型来确定位置和尺度,特征点的选择依据它们的稳定程度.**C.** 分配方向: 基于图像局部的梯度方向,分配给每个特征点一个活多个方向,所有后面的对图像数据的操作都相对于关键点的方向\尺度\位置进行变换,从而提供对于这些变换的不变性. **D.** 特征点描述子: 在每个关键点周围的领域内,在选定的尺度上测量图像局部的梯度. 这些梯度被变换成一种表示,这种表示允许比较大的局部形状的变形和光照变化.

图像金字塔:图像金字塔就是利用不同的尺度的图像堆叠来构建的多个图像大小下的图像矩阵. 在同一 Octave 内,不同层图像的 size 是一样的,后一层图像的高斯平滑因子σ是前一层图像高斯平滑因子的k倍,图像的高斯平滑因子会从最开始一层的σ

每层乘以k最终变成 2σ,即$k^s=2$以实现高斯平滑因子在所有层图像(包括跨 Octave 时)的平滑变换;在不同 Octave 间,后一组的第一个图像是前一组的倒数第三个图像(正数第$s+1$个图像)在 size(H 和 W 都减半)上的二分之一下采样(通过隔点采样实现),即图像 size 是前一组的一半(H 和 W 都减半),分辨率是前一组的四分之一

$k=2^{\frac{1}{s}},s$是DoG金字塔**中间**层数



从输入的视角来看:A. 对输入下采样与翻倍 filter 大小效果相同; B.用小σ核重复与输入卷积核用大σ核卷积一次效果相同.

确定关键点的尺度与位置信息: 为了寻找 DoG 函数的极(大\小)值点, 每一个像素点要与它所有的相邻点比较(包括相邻同尺度的 8 个相邻点和上下相邻尺度对应的 9×2 个点共 26 个点比较,以确保在尺度空间和二维图像空间都检测到极值点)

确定关键点的方向: 计算特征点的梯度,构建梯度直方图.在关键点位置周围选取邻域,并在该区域计算关键点的梯度大小和方向.梯度直方图将 0~360 度的方向范围每 10 度划分成 n(36)个柱(bins),方向直方图的峰值则代表了该关键点处领域梯度的方向,以直方图中的最大值作为该关键点的主方向进行旋转,为了增强匹配鲁棒性,只保留峰值大于主方向峰值 80%的方向作为该关键点的辅方向.

关键点描述符: 围绕特征点,将特征点周围的 16*16 的 window 划分为 4*4 的 grid of cells,对于每个 4*4 的 grid 都计算出其在 8 个方向上的梯度幅度和,归一化.然后这个特征点就可以用 128(16 cells*8 orientations)维的向量表示了.

性质:SIFT 特征对旋转\尺度缩放\光照剧烈变化鲁棒,对于仿射变换有一定的不变性但是不多;速度快且高效,可以实时运算

Ch6: Dense features

0. 检测人类的困难: 光照\外观\姿势\视角变化, 遮挡\杂乱

1. HOG(Histogram of Oriented Gradients)

思想:局部物体的外观\形状-局部物体梯度的强度和方向.

(KP1)步骤:A.把图像窗口划分为若干个小的空间区域(a block, cells, 1 个 cell 有多个 pixel),cells 可以是矩形(R-HOG)或者圆形(C-HOG); B. 对每个 cell 中每个像素的梯度方向和幅值进行统计,将 180°(图像梯度具有对称性,所以是 180°而非 360°,梯度值非负)划分为 9 个区间,将每个像素根据 9 个区间的梯度方向投照梯度幅值进行带权重累加. C. 在一个 block 内拼接直方图(比如一个 block 内有 4 个 cell,也就有 4 个 9 维的特征向量, 这一步要做的就是把这 4 个向量拼接在一起) D.归一化(对上一步中拼接起来的 36 维向量进行归一化,即局部归一化) E. 训练分类器: 用 SVM 对前面提取的图像特征向量进行训练,寻找一个最优超平面作为决策函数.

[划分 cell 并计算 9 维特征->滑动窗口得到 block,拼接并归一化向量->将所有 block 的特征向量拼接得到图像的 HoG 特征]

(KP2)计算:**A.**关于 Blocks,Cells:一个 block 有 2*2cells,一个 cell 有 8*8 pixels->一个 block 有 16*16 pixels;邻近 block 有 50%重叠->stride=8pixels. 那么对于 64*128 的图像,有 7(128-16+8[stride] + 0[padding] /8[stride])*15=105 个 blocks,每个 block 有 4*9 维特征向量,所以一共有 105*36=3780 维特征向量.B.先来看蓝色圈圈出来的像素点,它的角度是 80,幅值是 2,所以它在第五个 bin 里面加了;再来看红色的圈圈出来的像素点,它的角度是 10,幅值是 4,因为角度 10 介于 0-20 度的中间(正好一半),所以把幅值一分为二地放到 0 和 20 两个 bin 里面去.(按比例分配, 160°-180°分在 0°组和 160°组中).

优势:可以记录局部形状信息;对局部几何以及光度的变换具有不变性.**劣势:**在有遮挡\旋转和尺度缩放(不确定主导方向和相应比例)\有噪声(梯度对噪声敏感)的时候较困难;特征维度太多

HOG V.S. SIFT: A. 用途:HOG 通常用于描述轮廓(密集),SIFT 通常用于关键点检测(稀疏). B. SIFT 的梯度直方图是用的特征点的主梯度; HOG 用的是图片原始的坐标轴. C. SIFT 有多尺度描述符,HOG 对梯度直方图进行了归一化.

Ch6 Texture Representation

0. (KP1)为什么需要使用纹理?纹理的应用: A.从纹理中塑形: 根据图像纹理估计表面方向或形状;(对材质\外观有很好的表征效果) B. 根据纹理线索进行分割/分类: 分析\表示纹理,将有相同纹理的图像区域分组. C. 根据示例生成新的纹理 patches/图像; 纹理的表征是尝试对局部结构的重复 patterns 进行总结,滤波器组有助于测量局部邻域中的各种冗余结构,特征空间可以是多维的

Ch7 Fitting

0. 动机:已经知道如何提取特征, 希望根据一个简单的模型对多个特征进行分组, 从而形成更高层次、更紧凑的特征表示法

1. 已知哪些点属于直线,找到"最佳"直线参数-最小二乘

$$E=\|Y-XB\|^2\text{where}Y=\begin{bmatrix}y_1\\\vdots\\y_n\end{bmatrix}X=\begin{bmatrix}x_1&1\\\vdots&\vdots\\x_n&1\end{bmatrix}B=\begin{bmatrix}m\\b\end{bmatrix}$$

$$\frac{dE}{dB}=-2X^TY+2X^TXB=0$$

$$B=(X^TX)^{-1}X^T-B\text{的最小二乘解}$$

问题:值不具有旋转不变性;无法拟合垂直直线(-用一般式解决);对噪声\异常值不鲁棒

$$E=\sum_{i=1}^{n}(ax_i+by_i-d)^2$$

$$\frac{\partial E}{\partial d}=\sum_{i=1}^{n}-2(ax_i+by_i-d)=0$$

$$d=\frac{a}{n}\sum_{i=1}^{n}x_i+\frac{b}{n}\sum_{i=1}^{n}y_i=a\bar{x}+b\bar{y}$$

$$E=\sum_{i=1}^{n}(a(x_i-\bar{x})+b(y_i-\bar{y}))^2=\left\|\begin{bmatrix}x_1-\bar{x}&y_1-\bar{y}\\\vdots&\vdots\\x_n-\bar{x}&y_n-\bar{y}\end{bmatrix}\begin{bmatrix}a\\b\end{bmatrix}\right\|^2=(UN)^T(UN)$$

$$\frac{dE}{dN}=2(U^TU)N=0$$

Solution to$(U^TU)N=0$,subject to $\|N\|^2=1(i.e.a^2+b^2=1)$:

$\rightarrow$ eigenvector of U^TU associated with the smallest eigenvalue

这个解不是严格意义上的$(U^TU)N=0$的解,而是在约束条件下使得$(UN)^T(UN)$最小的解

2. (KP1)异常值处理:更鲁棒的拟合-RANSAC(Random sample consensus):拟合圆, 最少需要采样 3 点才能唯一确定一个圆$(x-a)^2+(y-b)^2=r^2$;两个二维平面的刚性线性变换 3 组点

流程:A. 均匀随机的选择一个可以估计出目标模型的最小数据集(直线-2 点). **B.**使用这个数据集来拟合模型(最小二乘法,只有 2 点则直接连线) **C.**把所有剩余的其他点带入这个模型,计算出"内点"(比如到这条直线的距离小于距离阈值t)的数目,再剩下的其他点就被作为异常点拒绝掉.**D.**比较当前模型和之前推出最好模型的"内点"数目,记录最大"内点"数及对应的模型参数. E.重复 A-D 步,直到迭代轮次结束或者当前模型已经足够好("内点"数>期望拟合点数目的阈值d)或者最好的模型,最后使用所有内点进行重新拟合

流程优化:事先确定合适的迭代次数 N,为此,定义以下变量:初始点数s(可以拟合目标模型的最小点数), 经过 N 次迭代后可以找到正确点的概率(至少有一次采样没有异常点的概率)p,外点比率e. 在模型一次迭代使用s个点的情况下, 选取的点至少有 1 个是外点的概率是$1-(1-e)^s$,减去在 N 次迭代下模型每次都至少采样到了一个外点的概率就是采样 N 次后得到正确模型的概率$p=1-(1-(1-e)^s)^N$,化简得$N=log(1-p)/log(1-(1-e)^s)$

优缺点:优势:简单,泛化性强;可以泛化到许多不同的任务上去;在实际中表现较好;劣势:有许多参数需要调节. 在内点率较低的时候表现不好(迭代次数太多,或者完全失败无法拟合).在最小数目的样本上不是总能获得一个好的初始化模型; 不够精确,在完成迭代后需要再用最小二乘重新拟合

动态调整迭代次数 N:(外点率e常先验未知-选择最坏情况 0.5,如果更多内点被找到再调整)初始$N=\infty,sample_count=0$;当$N>sample_count$, 采样计算内点数量,如果内点率是目前最高, $e=1-$(number of inliners)/(total number of points)再次重新计算 N 并 $sample\_count+=1$,直到$N<sample_count$(注意:N 仅在模型内点数比之前多时才更新,一旦更新一定在变小; 每次更新的结果不一定比之前更好,即更新效果是随机的)

3. (KP2)存在许多条直线: 投票的方法-Hough transform

投票策略;motivations: 噪声特征(有许多)不会一致地投票给单个模型(真实特征会倾向于投票给正确模型,噪声特征投票对象相对分散);只要还有足够多的特征能够对好的模型达成一致,缺失数据就不会造成问题.

流程:A.把参数空间下所有的累加器(一个网格对应一个累加器 accumulator)初始化为 0; **B.**

```
For each feature point (x,y) in the image
  For θ = 0 to 180
    ρ = x cos θ + y sin θ
    H(θ, ρ) = H(θ, ρ) + 1
  end
end
```

C. 找到被投票数最多($H(\theta,\rho)$)的最大值对应的(θ,ρ),则在图像中检测到的直线为$x\cos\theta+y\sin\theta=\rho$

应对噪声:A.选择合适的格子\离散化程度:太松散(格子太大),很多不同的曲线会相交于同一个格子里,使这个格子票数很大;太细小(格子太小),原本相近的点会给不同的格子里投票.**B.**增加邻域格子(给累加器做平滑累加),具体做法是对网格进行投票的时候不仅对当前网格进行投票,投票值根据到中心网格的距离进行加权(原本只对一个网格投 1 票,改进为对周围网格同时投 0.3,0.2 票,Hard label->Soft label). **C.** 去除不相关的特征,只选择有较显著梯度幅值的边缘点.

改进版霍夫变换:

For each egde point (x,y)

θ=gradient orientation at (x,y)

$\rho=x\cos\theta+y\sin\theta,H(\theta,\rho)=H(\theta,\rho)+1$,end

优缺点:A.优势:在检测图像中的直线\圆\其他几何形状时很有效;对噪声和部分遮挡鲁棒;可以处理多个相交或者重叠的形状;对被检测的形状提供了参数估计;**B.劣势:** 计算代价很高;需要给累加器空间设置合适的阈值;对图像分辨率和网格离散化的粒度敏感;在处理不是目标形状的特征(这些图像某一部分具有目标特征)也可能会造成尖锐的波峰,比较难以找到合适的分格大小

Ch8: Segmentation

0. (KP1)任务性质和基本原理:将图像分离成连贯的对象,将外观相似的像素集合在一起以便更高效地进行进一步处理;困难:过分割和欠分割;---**方法**:Clustering

1. (KP2)K-Means:基于划分的聚类算法

流程: A. 选择 K 个随机点作为聚类中心(means).**B.迭代:**把每一个数据实例赋给最近(可以用多种距离度量)的 mean; 对每一批被赋到同一个 mean 的簇群中的点计算均值(mean)更新为新的 mean.**C.**当不再有点的赋值被更新时停止;

收敛性:$c^*,\delta^*=argmin_{c,\delta}\frac{1}{N}\sum_i^N\sum_t^K\delta_{ij}(c_i-x_j)^2$单调递减序列,下界为 0 一定收敛;获得的是局部最优值

优缺点:A.优点: 简单,容易计算,能够保证收敛到局部最优解,收敛速度快.**B.缺点**:对中心点初始位置的确定\异常值比较敏感,K-means 是非确定性的,K 值不好确定;对于非凸数据集(非球形簇群)比较难收敛,只有在均值可计算时才可用;

2. (KP3)Mean-Shift:基于密度的聚类算法

思路:聚类为在一个 mode 内的吸引盆地内的所有数据点.吸引盆地为所有轨迹都趋向于同一个模式的区域

流程: A. 选择并计算特征(颜色,梯度,纹理···)**B.** 在单个特征点初始化窗口.**C.** 为每个窗口执行均值漂移直到收敛. **D.**合并收敛到相同峰值或者模式的窗口.

优缺点:A.优点:泛化性好,独立应用的工具;与模型无关,对数据簇群的形状没有任何先验的假设;只有一个参数(窗口大小);可

以找到多种模式;对异常点鲁棒(异常点会形成单体簇群,可被轻松识别). **B.缺点:**输出依赖于窗口大小;计算代价高;对于高维数据表现不佳;

3.(KP4) Normalized cut:基于 graph 的聚类算法

思路:最小化不同子集之间的连接权重,最大化同一子集内部的连接权重。

做法: Min-cut 对于图像建立相似度矩阵,像素相似的相似度大,不相似的就相似度小.要把一个图分成两部分所需要的 cost 就是两个图连通边的权重的和.需要找到最小的切割代价进行切割. 但容易直接把孤立点进行一个切割.给切割的代价函数做一个归一化.assoc(A,V)表示的是和 A 中所有点有连接边的权重和,这样就倾向于选择点多的阵列了.所以这个方法也叫 Normalized cut,最终目标是找到使得 Ncut 最小的分割方案.通过对线段大小进行归一化处理，修正最小切割偏差:

$$Ncut(A,B)=\frac{cut(A,B)}{assoc(A,V)}+\frac{cut(A,B)}{assoc(B,V)}$$

$assoc(A,V)$ = sum of weights of all edges that touch A

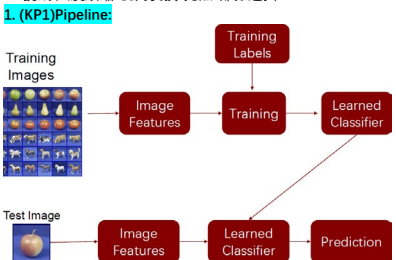
优缺点:A.优点:通用框架,可以适用多种不同特征以及相似度计算形式. **B:缺点:**较高的存储需求和时间复杂度.

4. 基于深度学习的解决方法:优势:端到端;到 per-pixel 的 Acc;鲁棒性高,泛化性好,一个模型对多个数据集;**困难:**类别不平衡(长尾分布),混淆类别,有挑战的场景(变体,杂乱,光照)

Ch9:Visual recognition

0. 挑战:尺度放缩\视角变换\光照\部分遮挡

1. (KP1)Pipeline:



2. (KP2)Bag of features:思路:量化特征空间生成离散视觉词集

步骤:A.拿来一堆图像,提取特征;**B.**学习视觉词库(常用聚类方法实现,一个通用特征列表,比如出现概率高的特征,做一个特征到图片类别的映射),**C.**给定一张新图片,提取特征并(利用视觉词库量化特征)构建直方图. **D.**通过视觉词汇频率表示图像,利用抽出的直方图分类

优缺点:A.优点:对图像内容的紧凑总结;对几何\变形\视角变换适应性强;实践效果好.**B.缺点:**基础模型忽略了几何形状,必须在后续验证或者通过特征编码;当对整张图像作为 bag 时前景和后景可能混合在一起;最优词库的构成(大小)不明确.

选择 vocabulary 大小:太小:视觉词汇不能代表所有 patches; **太大:**quantization artifacts, overfitting

Ch10: Detection

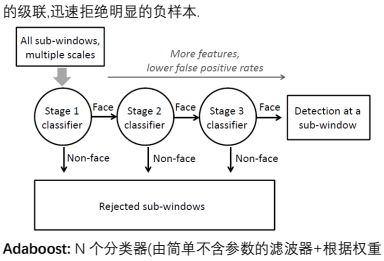
0. (KP1)Window-based object detection:训练 A.获取训练数据. B.定义特征. C.定义分类器. 给定一新图像:A.滑动窗口 B.由分类器打分.;

优缺点:A.优点:对不可变形的物体有较好识别能力,检测速度快;**B.缺点:**非刚性\可变形的物体无法通过假定固定的二维结构很好的捕获;或必须假定固定视角的表示方法才行;光照;目标位姿;遮挡;杂乱;类内外观;视角;需要大量训练集

1. Boosting: 一开始,所有训练样本权重相同.每一轮训练中,训练一个有最低加权训练误差的分类器,增加由当前弱分类器预测错误的样本在下一次的训练权重. 对所有的弱分类器做线性加权,加权比例为其预测 Acc.

2. (KP2)Viola-Jones face detector:

Main idea:A.利用感兴趣窗口内可高效计算的“矩形”特征来表示局部纹理 **B.** 选择可鉴别特征作为弱分类器.**C.**使用弱分类器的 boosted 组合作为最终分类器.**D.** 形成这样的分类器的级联,迅速拒绝明显的负样本.



Adaboost: N 个分类器(由简单不含参数的滤波器+根据权重选的最合适的阈值θ组成)->选择ε最小的分类器->重新给样本赋权(初始为均权)->N-1 个分类器->获得根据.

Boosting 细节:A.为每个滤波器选择最合适的阈值θ; **B.** 循环迭代(初始时各样本权重相同),目前为第 t 轮:a.对样本权值进行归一化.$w_{t,i}<-\frac{w_{t,i}}{\sum_j w_{t,i}}$ b.对每一个特征计算损失$\epsilon_j=\sum_i w_i\left|h_j(x_i)-y_i\right|, h_j(x)=1, if\ f_j(x)>\theta_j\ else\ h_j(x)=0$ c. 选择损失最小的分类器. d. 对样本重新赋权$w_{t+1,i}\leftarrow w_{t,i}\beta_t^{1-|h_t(x_i)-y_i|}, \beta=\frac{\epsilon_t}{1-\epsilon_t}$,减少正确预测样本的权重.**C** .最终投票来组合多个弱分类器,每个分类器权重与训练误差成反比

$$h(x)=\begin{cases}1 & \sum_{t=1}^{T}\alpha_t h_t(x)>\frac{1}{2}\sum_{t=1}^{T}\alpha_t,\beta_t=\frac{\epsilon_t}{1-\epsilon_t},\alpha_t=log\frac{1}{\beta_t}\\ 0 & ,otherwise\end{cases}$$

注意力级联: 一开始使用简单分类器拒绝较多明显的不含目标的窗口;只有通过前一级分类器检验的图像才会进入下一个分类器;如果任意一级分类器给出拒绝则样本会被拒绝.

级联训练:A.设置当前阶段检测正确率和误报率(false negative rage);**B.**添加弱分类器(特征)直至达到目标正确率;**C.**如果此时误报率不够低则再添加一个阶段;**D.**用当前阶段误报样本训练作为下一个训练阶段的负训练样本.

特点:训练慢检测快.积分图快速算特征.特征捕捉精准.快速拒绝非面部的窗口,提高召回率(预测的正确样本中确实是正确样本的比率)

总结:积分图是为了快速计算;boosting 是为了选择特征;注意力级联是为了快速拒绝非目标窗口

3. Pedestrian detection with HOG: (先有同样大小\尺寸的正负样本图片)把图片划分为小格子(cells),统计梯度方向,作统计然后作为特征.用提取的 HOG 特征在 SVM 训练一个分类器,一个行人模版.利用滑动窗口提取目标的 HOG 特征,找到响应的极大值.图像金字塔的方式来应对多尺度下的图像检测.

Ch11:Deep learning for CV-basic nets

0. (KP1)What can we contribute?针对特定任务设计.

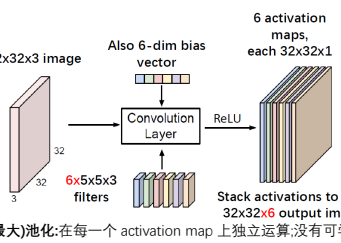
网络架构:Backbone+Modules;损失函数:目标函数,正则化方法;训练策略:可获得的数据形式与标签;监督学习;迁移学习;自监督学习

1. (KP2)图像分类方面的先进网络:-卷积神经网络

每一个 filter 对应一个 channel,产生一个 feature map.

卷积结果大小计算公式:$r=(n-k+2\times p+s)/s$,其中,m 是 image 边长,k是卷积核边长,p是 padding 长度(在一侧填充),s是步长 stride.

每个 filter 5*5*3+1(bias)=76params,共 76*6=456params



(最大)池化:在每一个 activation map 上独立运算;没有可学习的参数;让表示更小更易计算;引入了空间不变性;图片变小,速度变快.卷积是线性,最大池化是非线性

流程: Input image->Convolution(Learnable)->非线性激活->Spatial pooling(可重复堆叠多次)->Feature maps

What can CNN do for images:A.一些模式(一只鸟的鸟嘴)比整个图像小很多-不需要看见整张图像来发现一个模式,可以减少参数;**B.**同样的模式会出现在图像不同区域(没有必要重新学不同地方同一特征)-可以共享参数;**C.**对像素下采样不会改变对象语义->可以下采样让图像更小,使网络处理图像时参数更少.(性质 A.B.->卷积:减少参数,保留空间结构性质 **D.**->最大池化:减少参数,压缩特征);在最后加上全连接层作为分类层.

What can CNN do for CV:分类,分割, 目标检测,生成.

AlexNet:A. ReLU 激活函数,$ReLU(x)=max(x,0)$.来替代 Sigmoid 函数,减缓深层网络梯度消失问题 **B.**重叠池化(池化窗口步长s小于窗口大小k:$s<k$),增加了计算量,但也增加了特征提取效果.**C.**在全连接层使用 Dropout 防止过拟合.**D.**使用 GPU 加速训练(将模型划分到两个 GPU 上并行训练).**E.**使用裁剪\翻转等变换进行数据增强.**F.** LRN(Local Response Normalization)局部响应归一化,后面被证明作用不大.

VGG 使用“块”的网络,与 AlexNet 相比,使用小尺寸卷积核代替大尺寸卷积核(全为3 × 3),3 个3 × 3的卷积核相当于是7 × 7的感受野. **A.**就卷积核而言参数量变小 **B.**故而可以让网络更深;

改进:A.去掉了 LRN 层,由于作者发现深度网络中 LRN 的作用并不明显. **B.**使用小尺寸卷积核代替大尺寸卷积核 3 × 3(AlexNet 中有7 × 7的卷积核),**C.**池化层减小(2 × 2),步长为 2(AlexNet 中3 × 3,步长为 2). **D.**提供了可重复利用的“块级”结构,比 AlexNet 网络架构设计更规整,加深网络 .**E.**整体来看 VGG-16 比 AlexNet 所占内存更大(25x),参数量更大(2.3x)

缺点:参数量大,对存储空间和内存需求更高;计算量大,训练时间长,耗费更多计算资源;容易过拟合;

意义:证明了“深度”对 CNN 的重要性;确立了使用小卷积核的设计理念;是优秀的特征提取器,至今仍被广泛用作基础网络(图像分类,目标检测,图像分割,特征提取,迁移学习)

GoogLENet 含并行连结的网络.**A.**提出了 Inception 块,该结构将 CNN 中常用的卷积($1\times1conv, 3\times3conv, 5\times5conv, 3\times3max\ pooling(stride=1\ to\ retain\ the\ resolution)$)concatenate 到一起,拓展网络宽度,使网络对尺度变化更鲁棒.**B.** 1 × 1卷积进行降维,减少计算量. **C.**引入全局平均池化层代替全连接层

缺点:网络结构较为复杂,超参数较多,调优困难;

优点:参数量少(比 VGG 减少 12 倍);计算效率高;多尺度特征提取;性能优秀;内存使用效率高

ResNet 残差网络: **A.**提出了 Residual block，引入了跳跃连接(Skip connection)的恒等映射,$y=x+f(x)$,认为$f(x)=y-x$是对残差的拟合.通过这样的方式,原始信号可以跳过一部分网络层,直接在更深的网络层传递,解决的是深层网络的退化问题,使得训练更深的网络成为可能.**B.**使用了 batch normalization

特性:每个阶段特征图尺寸减半,通道数翻倍;使用全局平均池化层;广泛使用 bn;训练更容易收敛

ResNet Bottleblock(使用$1\times1conv$)的作用:保证空间布局信息的同时,减少模型的计算和参数量(变换 feature map 通道数),从而加速模型的训练和推理.

变种: ResNet-18/34: 使用基本残差块->ResNet-50/101/152:使用瓶颈残差块->预激活 ResNet:调整激活函数位置->Wide ResNet:增加通道数->ResNeXt:引入组卷积

如何改进模型: A.提高多样性:更深\更广\更多尺度\更多交互路径.**B.**简化优化:将梯度后移至浅层(ResNet ,DenseNet) **C.**提高互动的充分性和适应性:空间/通道注意力\组注意力

2. (KP3)图像分割方面的先进网络

挑战:A.将分类任务转换成分割任务-没有重用共享特征,不高效. **B.**使用全卷积神经网络,设计一个只有卷积层的网络同时对所有像素进行预测-在高分辨率图像上进行卷积太昂贵了;

技术:先下采样再上采样,上采样的实现-**A** .Unpooling:记录池化时的索引(最大元素的索引), 把之前降采样的结果进行升维(填充 0),得到的输出是稀疏的.**B.**Deconvolution(反卷积):不是卷积的逆运算,特殊的前向卷积操作,在输入特征图像素间插入 0,对扩展后的特征图进行常规卷积.可以恢复特征的空间信息.

挑战:需要细粒度细节

技术:A .Skip-connections 跳跃连接.以相同的分辨率融合编码器和解码器特征图.**B.** Dilated Convolution 空洞卷积:在相邻条目之间插入一行或一列 0.以扩大 filter 的范围

3. (KP4)目标检测方面的先进网络

目标:预测边界框\类别标签和置信度得分;确定每一次检测的真假

传统方法:SIFT\HOG+SVM

R-CNN Region proposals + CNN features: **A.**首先取一个预训练卷积神经网络，根据需要检测的目标类别数量,训练网络的最后一层. **B.**对输入图像进行选择性搜索得到每张图片的感兴趣区域(Region of Interest)，调整候选区域尺寸,使其符合 CNN 的输入尺寸要求.**C.**得到这些区域后,训练 SVM 来辨别目标物体和背景. 对每个类别,都要训练一个二元 SVM. **D.**训练一个线性回归模型,为每个辨识到的物体生成更精确的边界框.

第 B 步 Region of Interest 过程如下: 将一张图片作为输入,将图片分为多个区域, 基于颜色\结构\尺寸\形状,将相似的区域合并成更大的区域,生成最终的目标物体位置.

Fast R-CNN Region proposals + RoI pooling on top of a conv feature map: (R-CNN 步骤太多太慢,每个图片检测 40s~50s) Fast R-CNN 只需要通过一个 CNN 即可生成感兴趣区域. **A.**输入图片. **B.**输入到一个卷积层中，生成感兴趣区域. **C.**利用 RoI 池化层对这些区域进行重新调整,保证每个区域的尺寸相同,然后将其输入到全连接层中. **D.**在网络顶层用 Softmax 层输出类别,同样使用一个线性回归层,输出相对应的边界框.

评价:但是即使这样，Fast R-CNN 也有某些局限性。它同样用的是选择性搜索作为寻找感兴趣区域的,这一过程通常较慢,与 R-CNN 不同的是, Fast R-CNN 处理一张图片大约需要 2 秒,但在大型真实数据集上这种速度仍然不够理想.

Faster R-CNN RPN+RoI pooling: Faster R-CNN 是 Fast R-CNN 的优化版本,二者的主要的不同在于感兴趣区域的生成方法. Fast R-CNN 使用的是选择性搜索,而 Faster R-CNN 使用的是 Region Proposal 网络(RPN).RPN 将图像特征映射作为输入,生成一系列 objective proposals,每个都带有相应的分数.

步骤:A.输入图像到卷积网络中,生成该图像的特征映射 **B.**在特征映射上应用 Region Proposal Network,返回 object proposals 和相应分数.**C.** 应用 RoI 池化层,将所有 proposals 修正到同样尺寸. **D.**最后,将 proposals 传递到完全连接层,生成目标物体的边界框

RPN:将 CNN 中得来的特征映射输入到 Faster R-CNN 中,然后将其传递到 RPN 中. RPN 会在这些特征映射上使用一个滑动窗口,每个窗口会生成具有不同形状和,尺寸的 k 个 anchor box.Anchor boxes 是固定尺寸的边界框它们有不同的形状和大小.对每个 anchor,RPN 都会**预测两点:**

首先是 anchor 就是目标物体的概率(不考虑类别).第二个就是 anchor 经过调整能更合适目标物体的边界框回归量。现在我们有了不同形状\尺寸的边界框,将它们传递到 RoI 池化层中。经过 RPN 的处理,proposals 可能没有所述的类别.我们可以对每个 proposal 进行切割,让它们都含有目标物体.这就是 RoI 池化层的作用.它为每个 anchor 提取固定尺寸的特征映射.

速度比 Fast R-CNN 相比更快了(2ms),从流程上讲分离了 proposal genaration 和 region classifictaion 的过程.

YOLO:

思路:没有边界框 proposal.一次性为每个位置预测一个类和一个框在网格中的位置。

步骤: A.将图像分为7 × 7的 cells,每个 cell 训练一个检测器,这个检测器需要预测目标的类别概率分布\边界框\置信分数.**B.**经过层层卷积得到7 × 7 × 30维张量

4. (KP5)Transformer

Self-attention: 编码器自注意力: q,k,v 都来自于同一个输入向量,使用不同的权重矩阵;**解码器自注意力**: q,k,v 也仍是同一个序列,但是对应于未来解码器输出的值被 mask 掉(先计算然后 mask);

用途:捕获序列的上下文关系,不关心输入序列的位置和顺序

Cross-attention: q,k,v 来自不同序列. **编码器-解码器注意力**:q 来自于前一个解码层的输出,k\v 来自编码器的输出.

Multi-head attention:将注意力机制分成多个头(head)每个头独立学习不同的注意力模式,最后将所有头的输出合并.

Basic transformer model: encoder-decoder 架构.

encoder(MHSA+FeedForward):接收整个输入序列并输出相同长度的编码序列;**decoder**(MHSA+CA(encoder-decoder attention)+FeedForward):根据编码器输出和之前预测的 token 预测下一个 token

特点: 擅长处理长序列,每一次注意力计算都会看整个输出;可以处理无序数据集或者带有位置编码的有序数据集.具有高度可扩展性和高度可并行性;所有输入的对齐和注意力评分都可以并行处理;训练更快,模型更大,在视觉和语言任务中表现更佳.正在迅速取代 RNN,LSTM,甚至可能(?)取代卷积.

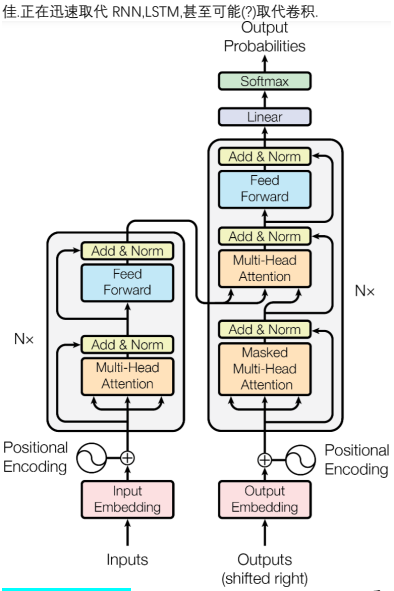


Image transformer: **A.** Detection Transformer(DETR):CNN 和 tfm 的混合,旨在标准的识别任务;**B.**使用 tfm 进行 Image captioning(从 pixels 到 language)**C.**Vision transformer(ViT):把 image 划分为 patches,将线性投影并展平的 patches 输入标准 tfm encoder;对于14 × 14的 patch 需要16 × 16个 patches 来表示224 × 224的图像;transformers 据称比 CNN\混合架构的计算效率更高.**D.**分层的 transformer-Swin:Window multi-head self-attention

与其他网络类型的比较: **A.**RNNs 适用于有序序列.**优势:**不受固定上下文大小的限制(原则上);经过一个 RNN 层后,h_T就能“看到”整个序列”; **劣势:**不可并行,需要按顺序计算隐藏状态;隐藏状态的表达能力有限. **B.**1D 卷积网络,适用于多维度的网格.**优势:**在训练时每一个输出都可以被并行计算.劣势:需要堆叠多层卷积以让输出看到整个序列.**C.** Transformers 适用于向量的集合. 优势:擅长处理长序列,在经过一个自注意力层后,每一个输出就可以看到所有的输入序列; 每一个输出在训练时都可以并行计算. **劣势:**内存密集型,注意力运算的成本是输入序列长度的二次方

Transformer 的应用:NLP(Transformer, BERT, Compressive Transformer),CV(DETR, ViT, Swin Transformer), Audio(Speech Transformer, Reformer TTS), Multimodal(VisualBERT, DALL E)

改进方式: A.模型效率:轻量级注意力(如稀疏注意力变体);和分而治之方法(如递归和分层).**B.**模型泛化能力.**C.**模型迁移应用.

CNN vs Transformer:A.CNN: CNN 是一种简化的 SA(SA 具有可学习的感受野);归纳偏置:Translation invariance 平移不变性(shared kernel across spatial postions);Locality 局部性(restricted window).**B.**Transformer:全局感受野层,适用于远距离依赖性建模;更少的结构先验假设(容易在小规模数据上过拟合).

5. (KP6)Self-supervised learning

Motivation:有标签数据很难获得,无标签数据比较容易获得.

Semi-supervised learning 把有标签数据和无标签数据混合在一起作为训练集以产生更好的模型.具体来说,从有标签数据中先训练得到一个模型然后用来对无标签数据预测获得 Pseudo -label,之后将这个数据从无标签数据集中移除加入有标签数据集并重复这个过程.

Self-supervised learning 是一种特殊的无监督学习(子集),它通过解决 pretext tasks 来产生足够好的特征以解决下游任务.pretext tasks 的标签是自动生成的,不需要人工标注.常常使用如分类\回归等监督学习的目标函数来学习通用的特征表示.

工作流:A. Pretext task:基于数据本身定义一个任务,无需人工标注,可以被认为是无监督学习任务,但是通过监督学习目标来学习,从 pretext task 中学习到足够好的特征提取器.**B.**下游任务:数据集是标注好的;在特征提取器基础上附加一个浅层网络,在少量目标任务的有标签数据集上对浅层网络进行训练.

Self-supervised learning V.S. unsupervised learning:A.自监督学习是无监督学习的一种(一个子集),都不需要人工标注数据.**B.**无监督学习是任何不需要标签的学习方式,目的是直接从数据中发现结构或者模式,如聚类和密度估计.**C.** 自监督学习的学习器会从数据中生成标签,然后用来解决有监督任务.

Self-supervised learning V.S. generative learning:A.相同点:二者的目标都是在没有人工标签注释的情况下从数据中学习.

B. 生成学习旨在为数据分布建模,例如生成逼真的图像. 如 GAN 和 VAE .C.自我监督学习旨在利用 pretext tasks 学习高级语义特征.如数据预测(图像上色和图像恢复),图像变化(背景预测\拼图解密\旋转角度预测)的预测,和是否相似的预测.

优点: 无需人工标注: 大幅减少对标注数据的依赖，特别适合大规模数据集； **通用特征表示：** 学到的特征具有迁移性,可以用于多个下游任务.**高效利用数据：** 充分挖掘未标注数据的信息,提高模型性能.

自监督学习的三种类型

A.数据预测 （Data Prediction）

原理：

模型根据输入数据 x，预测与其相关的输出 x'。

输出通常是对输入数据的一种形式重建（如数据补全或特征预测）。

典型任务：

图像重建（Image Reconstruction）：

例如输入部分被遮挡的图片，模型预测完整图像。

文本填充（Text Infilling）：

例如输入一个句子，模型预测被遮盖的单词。

目标：

学习数据的高层语义特征，以便在数据重建任务中提升表示能力。

B.变换预测（Transformation Prediction）

原理：

给定经过某种变换的输入数据 $T(x)$，模型需要预测该变换 T是什么。

这种方法通过设计特定的预训练任务，逼迫模型学习数据的结构化特性。

典型任务：

图像旋转预测（Rotation Prediction）：

输入一张旋转后的图片，模型预测旋转角度（如 0°、90°、180°、270°）。

颜色恢复（Colorization）：

输入一张灰度图像，模型预测其颜色信息。

数据增强分类（Augmentation Classification）：

输入经过随机数据增强后的数据，模型预测使用的增强类型。

目标：

通过对变换的预测任务，增强模型对数据的全局和局部特征的理解。

C.孪生网络方法（Siamese Methods）

原理：

输入两组数据（如 $T(x)$和 $T'(x)$，分别通过同一个网络，模型预测两者的相似性或差异性。

通常用于学习数据表示的相对关系，而不是绝对值。

典型任务：

对比学习（Contrastive Learning）：

输入两组经过不同增强的同一数据，模型预测它们是否属于同一个原始样本。

图像匹配（Image Matching）：

判断两张图片是否包含相同内容。

相似度学习（Similarity Learning）：

学习数据之间的距离表示（如欧几里得距离、余弦相似度）。

目标：

学习具有辨别能力的特征表示，适合下游任务（如分类、检索）。

6. (KP6)Transfer Learning

Traditional V.S. Transfer Learning:A.传统机器学习:一般情况下为一个任务学一个系统.B.迁移学习:从源模型迁移知识到目标任务.

Model Fine-tuning:使用范围:当两个任务数据集(源数据数量多,目标数据很少,One-shot learning 在目标域只有一些样本)都是有标签的时候;思路:通过源数据训练模型， 然后通过目标数据微调模型,但由于目标数据有限,需要避免过拟合;具体算法:**Conservative Training:**让前后两个训练模型差的不要那么大,防止 overfitting.**Layer Transfer:**只 train 部分的 layer,图片:copy 前面几层,train 后面几层.

Multitask Learning:使用范围:当两个任务都是有标签的时候.共用一些 layer,可以直接从输出层开始,也可以用中间几层.应用:过语言\语音识别.

Domain adaptation: **适用范围:**源数据有标签,目标任务没有标签.**简介:**用算法产生分类的原因可能是因为数据的 domain 不同。我们需要把数据放到同样的 domain 下面，或者说规避掉 domain 的影响。前面的层可以把 domain 分开，接到后面的层让它对 domain 进行分类。同时有一个网络附加在后面去分类 label. 当然我们的目标是让 label 的分类越来越准,然后让 domain 的分类越来越不准。要做到这样的目的只需要在反向传播的时候给 domain 那里的梯度它加个负号就行。具体算法: **A.基于差异的方法,**在特征空间中最小化 domain 距离,关注于设计合适的距离度量.**B. 基于对抗的方法,**最小化 domain 分类准确率, 最大化标签分类准确率,domain 分类器和 label 预测器的目标都是最大化对应的分类准确率,但是要欺骗 domain 分类器的同时满足 label 分类器.**C.基于重建的方法,**目标样本的数据重建是一项辅助任务,同时注重在两个领域之间创建共享表征， 并保持每个领域的各自特点

Self-taught learning:适用范围(源数据没有标签,目标数据有标签)

Self-taught clustering:适用范围(源数据和目标数据都没有标签)

Knowledge distillation:将大型深度神经网络中的知识提炼到小型网络中

Target Data		Source Data (not directly related to the task)	
		labelled	unlabeled
	labelled	Fine-tuning Multitask Learning	Self-taught learning
	unlabeled	Domain adaptation: -Discrepancy-based -Adversarial-based -Reconstruction-based	Self-taught Clustering

原理:让模型变小一点,这样就可以放到真实的应用了.模型的话就是教师学生模型.给定一个大的模型， 然后让小模型去学习大模型的各层的一个输出,从而达到对大模型的一个学习能力.

分类: 基于响应:对于相同数据输出应相近;基于特征:对于相同数据， 每层的输出应相近;前面这两种都用到了特定层的输出.基于关系:考虑不同层和不同样本之间的关系;跨模态蒸馏:在训练或测试过程中,可能无法获得某些模式的数据或标签.