压缩比（compression ratio）

1 1

$$compression\ ratio = \frac{B_0}{B_1}$$

B_0 – number of bits before compression

B_1 – number of bits after compression

average number of bits 平均比特数

字母S的信息源的熵η

$$\eta = H(S) = \sum_{i=1}^{n} p_i \log_2 \frac{1}{p_i}$$

1 1

$$= -\sum_{i=1}^{n} p_i \log_2 p_i$$

霍夫曼编码标准步骤

1

1. 按照概率从大到小的顺序，从上到下排列，si 概率
2. 取最小的两项，合并，加到下一列中，参加排序
3. 标出箭头
4. 上（1）下（0）
5. 画出树状图，左0，右1
6. Z字形扫描，从概率最大的开始安排
7. 写出哈夫曼01表示

JPEG 编码器

JPEG Encoder

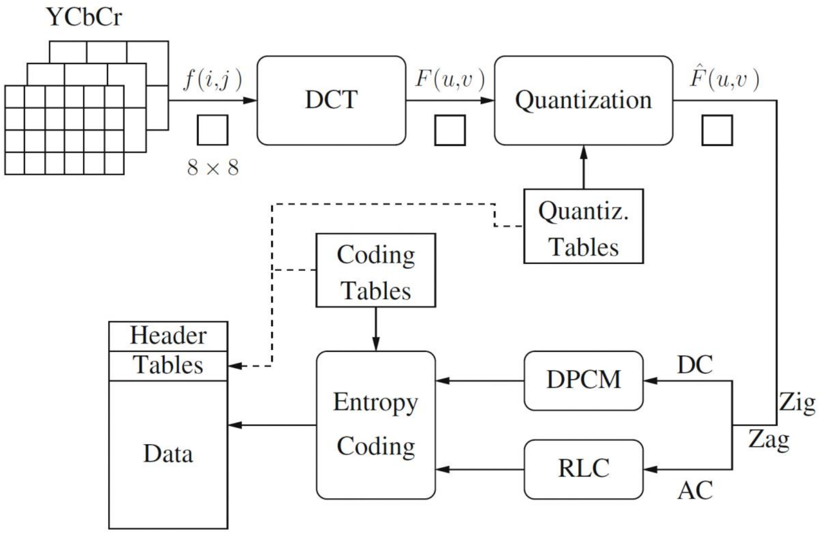
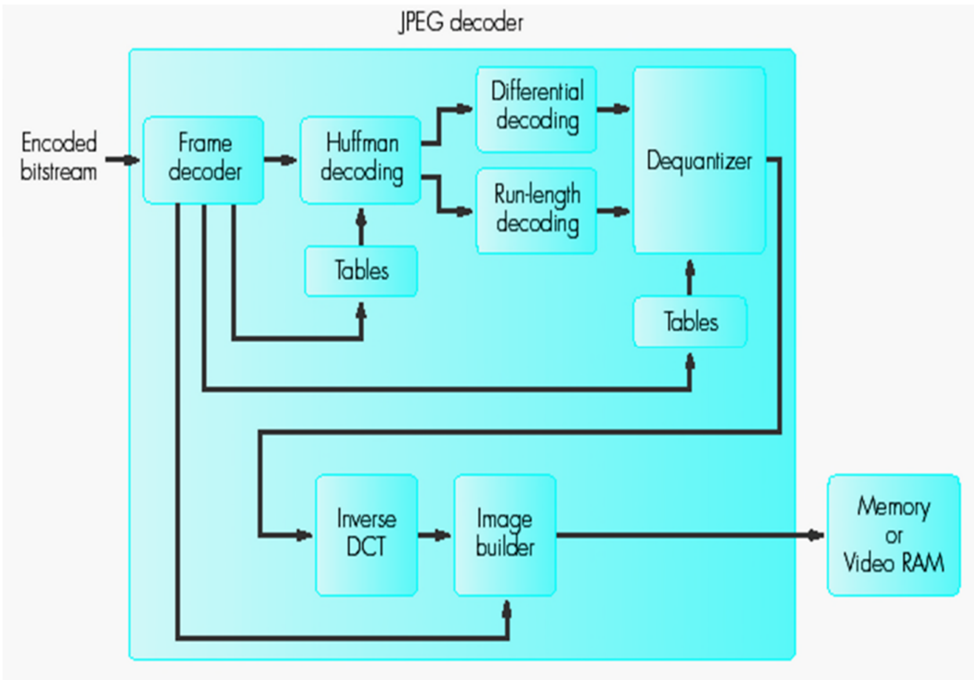


Fig. 9.1: Block diagram for JPEG encoder.

JPEG 解码器



传统DCT算法

$$S_{uv} = \alpha(u)\alpha(v)\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} s_{ij} \cos\frac{(2i+1)u\pi}{2N} \cos\frac{(2j+1)v\pi}{2N} \qquad u, v = 0, \dots, N-1$$

$$\alpha(k) = \begin{cases} \sqrt{\frac{1}{N}} & \text{for } k = 0 \\ \sqrt{\frac{2}{N}} & \text{for } k = 1, 2, \dots, N-1 \end{cases}$$

T矩阵背下来，用于偷鸡

$$T(i, j) = \begin{cases} \frac{1}{\sqrt{N}}, & \text{if } i = 0 \\ \sqrt{\frac{2}{N}} \cos\frac{(2j+1)i\pi}{2N}, & \text{if } i > 0 \end{cases},$$

2D DCT of A = $\mathbf{TAT}^T$

$$\mathbf{T} = \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ \frac{1}{\sqrt{2}}\cos\frac{\pi}{8} & \frac{1}{\sqrt{2}}\cos\frac{3\pi}{8} & \frac{1}{\sqrt{2}}\cos\frac{5\pi}{8} & \frac{1}{\sqrt{2}}\cos\frac{7\pi}{8} \\ \frac{1}{\sqrt{2}}\cos\frac{\pi}{4} & \frac{1}{\sqrt{2}}\cos\frac{3\pi}{4} & \frac{1}{\sqrt{2}}\cos\frac{5\pi}{4} & \frac{1}{\sqrt{2}}\cos\frac{7\pi}{4} \\ \frac{1}{\sqrt{2}}\cos\frac{3\pi}{8} & \frac{1}{\sqrt{2}}\cos\frac{9\pi}{8} & \frac{1}{\sqrt{2}}\cos\frac{15\pi}{8} & \frac{1}{\sqrt{2}}\cos\frac{21\pi}{8} \end{bmatrix}$$

$$= \begin{bmatrix} 0.5000 & 0.5000 & 0.5000 & 0.5000 \\ 0.6533 & 0.2706 & -0.2706 & -0.6533 \\ 0.5000 & -0.5000 & -0.5000 & 0.5000 \\ 0.2706 & -0.6533 & 0.6533 & -0.2706 \end{bmatrix}$$

输出维度

输出维度 = ((N - F + 2P)/S) + 1

N: 输入图像边长
F: 卷积核(filter)大小
P: 填充(padding)大小
S: 步长(stride)

线性分类器

$$f(x) = Wx + b$$

平方损失

$$L(x,y) = \sum_i (y_i - f(x_i))^2$$

均方误差

$$MSE = \frac{1}{N}\sum_i (y_i - f(x_i))^2$$

平均绝对误差

$$MAE = \frac{1}{N}\sum_i |y_i - f(x_i)|$$

softmax归一化

$$p_j = \frac{e^{z_j}}{\sum_k e^{z_k}}, \text{ where } z_j = f(x_j)$$

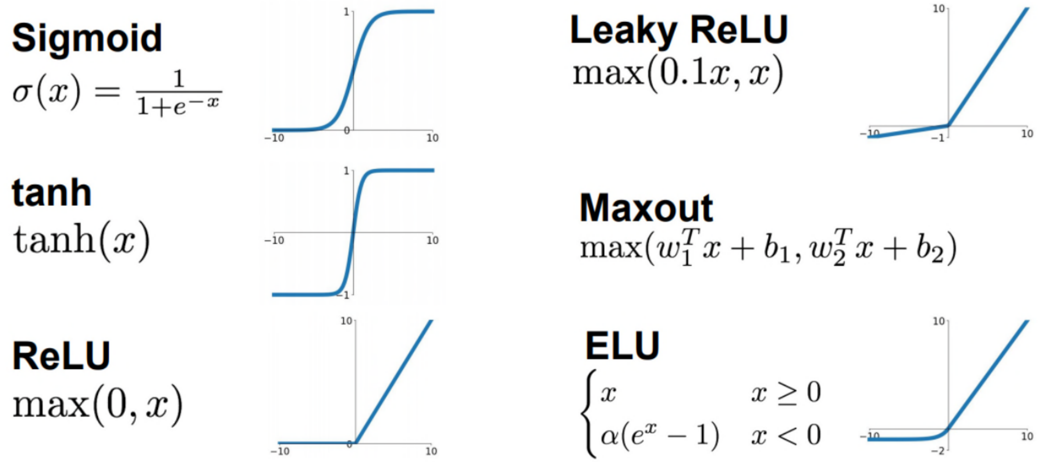
交叉熵损失

$$L = -\sum_j y_j \log_e p_j$$

激活函数

sigmoid要背诵

ReLU也要背诵



CNN 梯度下降公式

W 是我们想要优化或学习的参数

$$L(W) = \frac{1}{N}\sum_{i=1}^{N} L_i(x_i, y_i, W)$$

微分

梯度

$$\nabla_W L(W) = \frac{1}{N}\sum_{i=1}^{N} \nabla_W L_i(x_i, y_i, W)$$

参数下一个迭代参数

$$w_{t+1} = w_t - \alpha \nabla L(w_t)$$

拥有了当前的参数集　　学习率 Alpha

softmax Loss

z：Normalized probabilities

p：Predicted output Scores

$$p_j = \frac{e^{z_j}}{\sum_k e^{z_k}}$$

激活函数

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

Vanilla RNN

$$h_t = f_W(h_{t-1}, x_t)$$

$$\downarrow$$

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

$$y_t = W_{hy}h_t$$

LSTM必背公式

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}$$

门控多四项，sig三tanh一行。

遗忘乘旧忆，输入加新常。

输出乘激活，隐含藏心房。

长期记忆

$$c_t = \underset{\text{遗忘}}{f} \overset{\text{逐个元素乘法}}{\odot} \underset{\text{之前长期记忆}}{c_{t-1}} + \underset{\text{输入}}{i} \odot \underset{\text{门}}{g}$$

短期记忆

$$h_t = \underset{\text{输出门}}{o} \odot \tanh(\underset{\text{长期记忆}}{c_t})$$

ifog理解　　1　　1

i: Input gate, whether to write to cell
f: Forget gate, Whether to erase cell
o: Output gate, How much to reveal cell
g: Gate gate (?), How much to write to cell

i越大，写入长期记忆越多

f越大，忘掉的就越少，保留就越多

o越大，显示的就越多

g越大，写入就越多

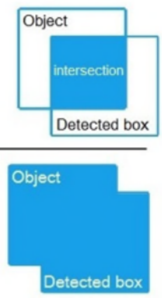
性能指标

平均精度 mean Average Precision (mAP) (AP)

$$\mathrm{mAP} = \frac{1}{N}\sum_{i=1}^{N}\mathrm{AP}_i$$

交并比 Intersection over Union (IoU)

$$\mathrm{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$



真阳性 True Positive (TP)

假阳性 False Positive (FP)

假阴性 False Negatives (FN)

只要检测出来了，不管IoU，都是positive，大于IoU的才是True，小于IoU阈值的是False

没检测出来，就是Negatives，第二个框都没有，IoU就是0,所以是False

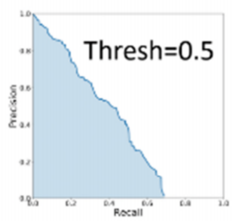
召回：真实存在，检测出来多少

召回 $$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} = \frac{\mathrm{TP}}{\#\ \text{ground truths}}$$

精度：所有检测出来的，真实的多少

精度 $$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} = \frac{\mathrm{TP}}{\#\ \text{predictions}}$$

AP50 or AP0.5



精度召回曲线下的面积

AP0.50: 0.55: 0.95

$$mAP_{\mathrm{COCO}} = \frac{mAP_{0.50} + mAP_{0.55} + \ldots + mAP_{0.95}}{10}$$