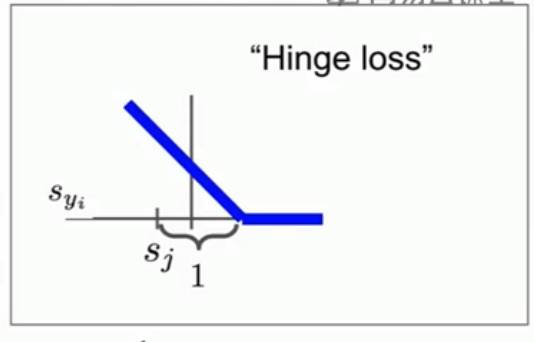
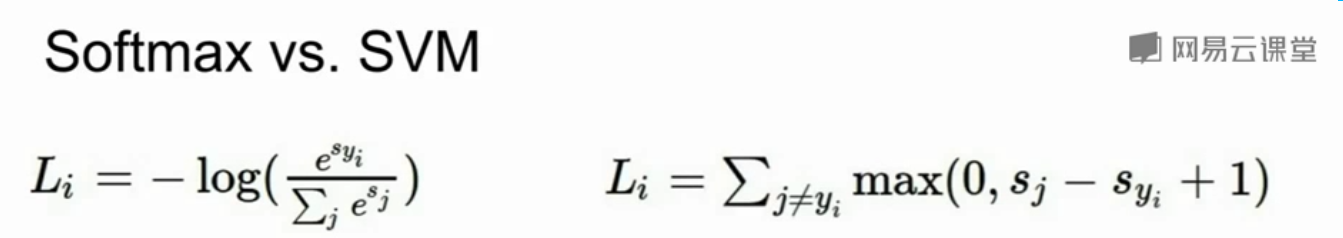
# 多分类SVM分类

loss值计算方式

* 当正确分类的分数比其他任意错误分类的分数高于一个阈值，则损失为零否则损失为错误分类的分数加一减去正确分类分数
* 将上述对于每个错误分类计算出的损失相加既可以得到整体的损失
* 计算整个训练集的loss求平均
* 该loss为合页损失函数



1. 这里的1被称为safety margin安全边界
2. scale是什么？
3. That this free parameter of one kind of washes out and is canceled with this scale like the overall setting of the scale in W
4. 对于这个loss在开始迭代的时候应该为C-1，因为初始的W值比较平均，导致计算出的score值较平均，loss为1，求和之后是C-1（C是所有的类别）
5. multinomial logistic regression（多项逻辑斯蒂回归损失函数），将score指数化使结果为正数，之后归一化



1. SVM的loss计算方法与softmax的区别，SVM的loss计算中当正确分类分数超过错误分类一定数值的时候loss为零，即不再向更好的方向优化。Softmax loss不会为零而是不断向更好的方向优化
2. 正则化，包括添加l1正则化（更加倾向于稀疏的矩阵），l2正则化（更加倾向于将整体的多项式的最高次数降低）

# 优化算法

为了寻找最好的w计算得到最小的loss，需要采取优化的算法，然而对于大量参数来说，求解最小值较为复杂，因此对于如何得到一个较好的w有一些策略对应算法

策略1：随机生成w计算loss

策略2：跟随斜率

1. 在一维函数f(x)中，求解斜率就是对f(x)求导
2. 在多位函数中，求导数得到的是梯度，梯度是由编导数组成的向量，有着和x（参数）相同的形状，负梯度方向指向了函数下降最快的方向
3. 函数某点任意方向的斜率等于这一点上梯度和该点单位方向向量的点积

**计算梯度的方法：**

1. 有限差分法(the method of finite differences)，根据梯度的定义计算梯度，在W上加一个小的偏移量后计算loss值的变化，除以偏移量可近似得到梯度向量
2. 使用数值分析的方法计算梯度

**随机梯度下降**

不计算整个训练集的误差和梯度值，在每一次迭代中，选取一小部分训练样本组成minibatch（通常为2的n次幂）使用minibatch估算误差综合

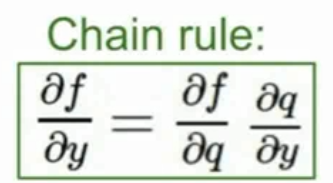
**图像的特征**

颜色直方图，方向梯度梯度图（将图像按八个像素区为八份，在八个像素区的每一部分计算像素值的主要边缘方向）、词袋bag of word（用特征向量表示一句话方法之一是计算不同词在这段话中出现的次数）

# 反向传播

计算任意复杂函数的解析梯度方法。计算图是指用来表示任意函数的图，图中的节点表示我们要执行的每一步的计算。

反向传播是链式法则的递归调用，从计算图的最后面向前计算所有的梯度



每个计算节点有一个本地梯度计算公式，将该公式与后向传播的梯度相乘可以得到该计算节点对应参数相对于整个f的梯度

**计算图节点详解**

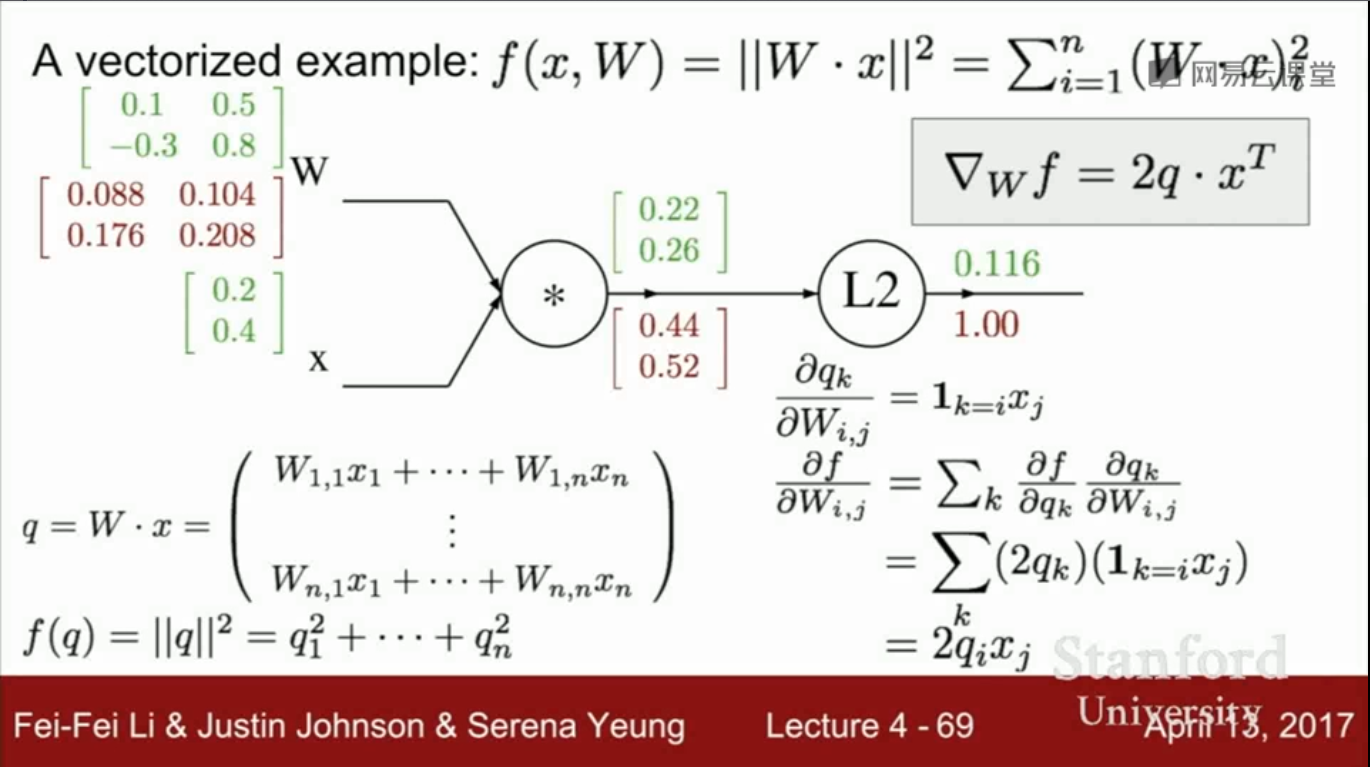
加法门是一个梯度分布器，将上游的梯度分发给下游的各个节点，max门是将上游的梯度只传递给下游其中一个节点，乘法门是一个梯度转换器，根据另一个分支的值对其进行缩放。

当一个节点的上游有多个运算节点时，梯度会在在这个节点累加

**变量是高维向量的情况**

这时梯度变为了雅克比矩阵，该雅克比矩阵中每一行都是输出对于输入求偏导的结果。但是由于第一个输入的参数只与第一输出有关，因此雅克比矩阵只是一个对角矩阵

例子



# 神经网络

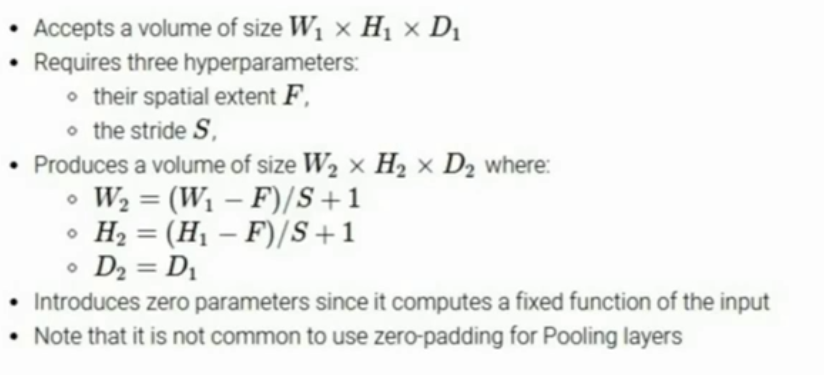
神经网络是由简单函数构成的一组函数，用层次化的方法堆叠起来，来形成一个复杂的非线性函数。

权重矩阵的每一行相当于一个模板

# 视觉之外的卷积神经网络

感受野，卷积核大小可以称作输入数据的感受野，卷积核数量是深度，池化的意义是让结果更小且更容易控制。

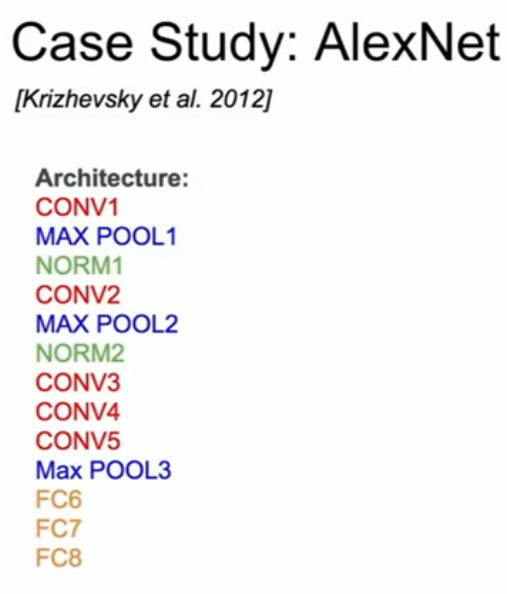
池化常用最大池化法。在池化中，通常设计步长使得卷积核无重叠，并且在池化中一般不padding，输出大小的计算方法



池化层一般为2\*2卷积核加步长2或者3\*3卷积核步长为3

# CNN架构

**1. AlexNet**



AlexNet的大小：

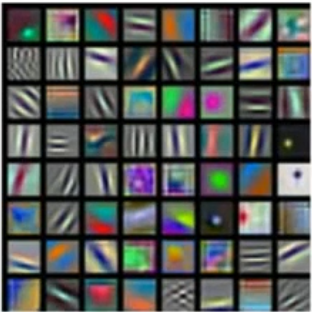
输入227\*227\*3images、

第一层96个11\*11，步长4=》55\*55\*96（大小11\*11\*3\*96）

第二层3\*3卷积核，步长为2

# 特征可视化、倒置、对抗样本

为了直观的看到卷积层的作用，在课程中将AlexNeta的第一层卷积(64\*11\*11\*3)，将三个通道作为rgb的数值（所有权重值被限制在0-255之间），画出64个11\*11大小的图片，如下图所示

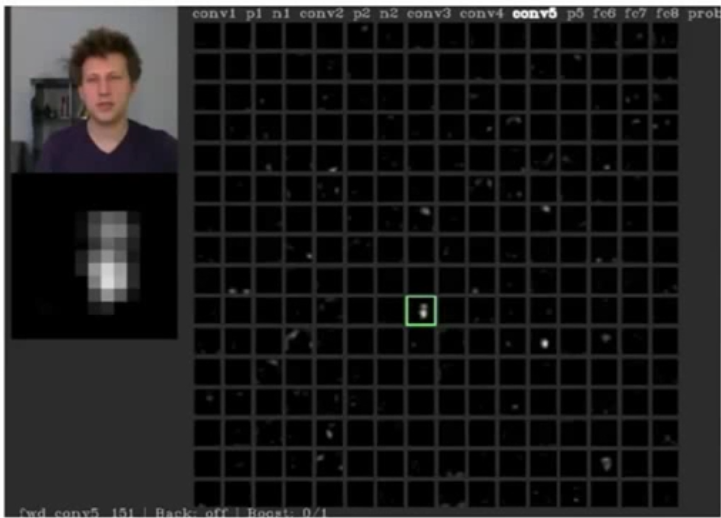


可以看到其中大部分在寻找的是有向边（不同方向的明亮线条），还有完全相反颜色组成的卷积核。

对最后一层的可视化，采用多个图片组成的batch经过训练过的n维向量（课程中的AlexNet为4096维），对这些n维向量进行knn分类或者降维（PCA、t-SNE t-分布领域嵌入），对4096维特征进行降维，降维到2维，将2维点对应原图片放到2维坐标系上



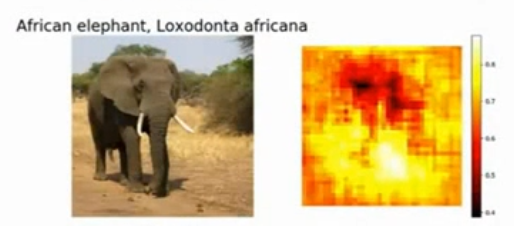
对中间层的权重图的可视化不可解释，但对中间层的激活映射图在一些情况下是可解释的，课程中举了AlexNet的第五层卷积后的特征图（128\*13\*13），可视化为128个13\*13的矩阵，如下图所示



可视化中间层的另一个措施是，记录大量图片某层卷积网络输出的一个通道。找到这个通道中被最大激活的点，找出该点对应于图片的位置，可以看到卷积网络在寻找什么。

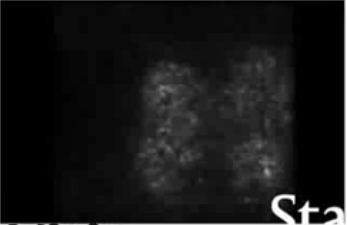


还有一种方法是将图片的一部分遮住，将这个遮住的部分在图像上滑动，求出遮住后的图像与原图像之间的分数差距，并绘制热力图，可以得到图片的哪一部分对分类决策起到非常重要的作用



还有一种方法是计算每个像素点对分类分数的梯度，并取绝对值和rgb通道上的最大值得到灰度图称为显著图



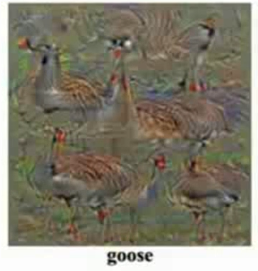


在进行语义分割的时候也会计算显著图，使用显著图在没有任何标签的情况下直接进行语义分割。

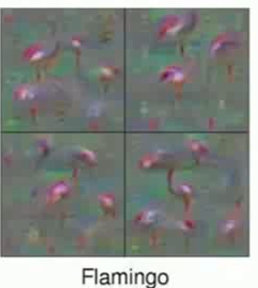
考察图像像素对网络内部神经元分值的影响，计算中间值对图像的梯度，从而得到像素对特定神经元的影响

引导式反向传播，在反向传播中加relu，只传递正影响。

考察什么样的输入会激活神经元，使用梯度上升的方法，神经网络权重保持不变，从初始的随机值组成的图片中，改变图像的像素，使分数最大化，使用正则化值防止过拟合。从而生成网络想要生成的图片



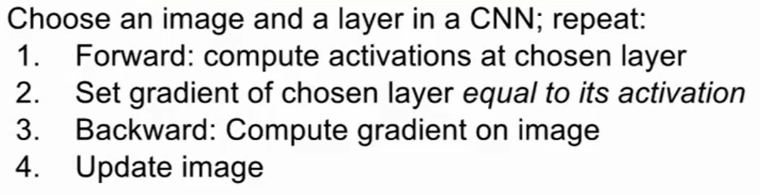
如果定期在优化时加入高斯噪声作为正则化，并将一些小梯度置为0，高斯模糊处理后，图像变得平滑



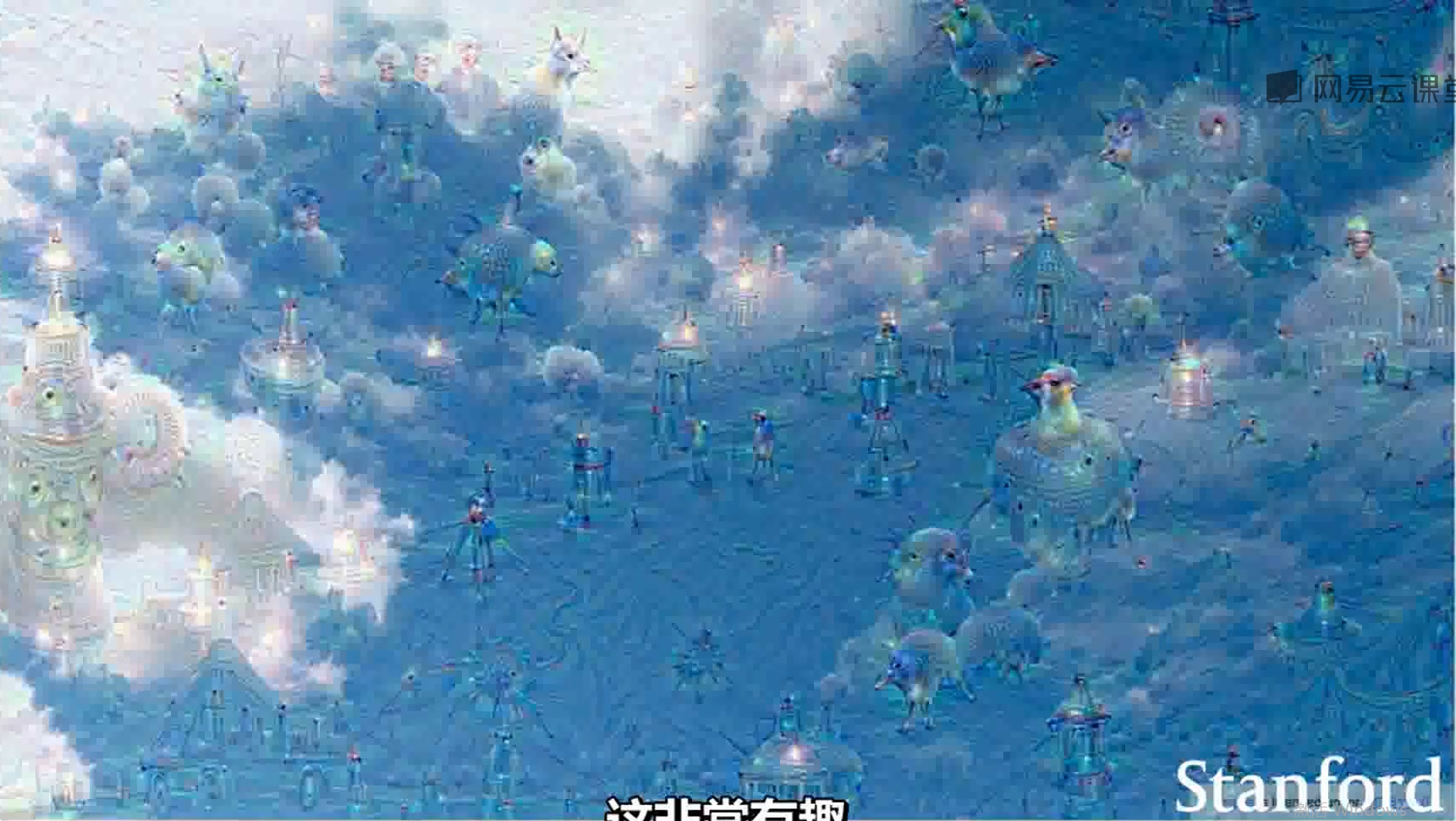
利用上述原理，若输入一张正常物体的照片，而让网络去合成其他物体，从而希望这张图片可以更像其他物体，但事实上并不会有所改变。

# ·DeepDream

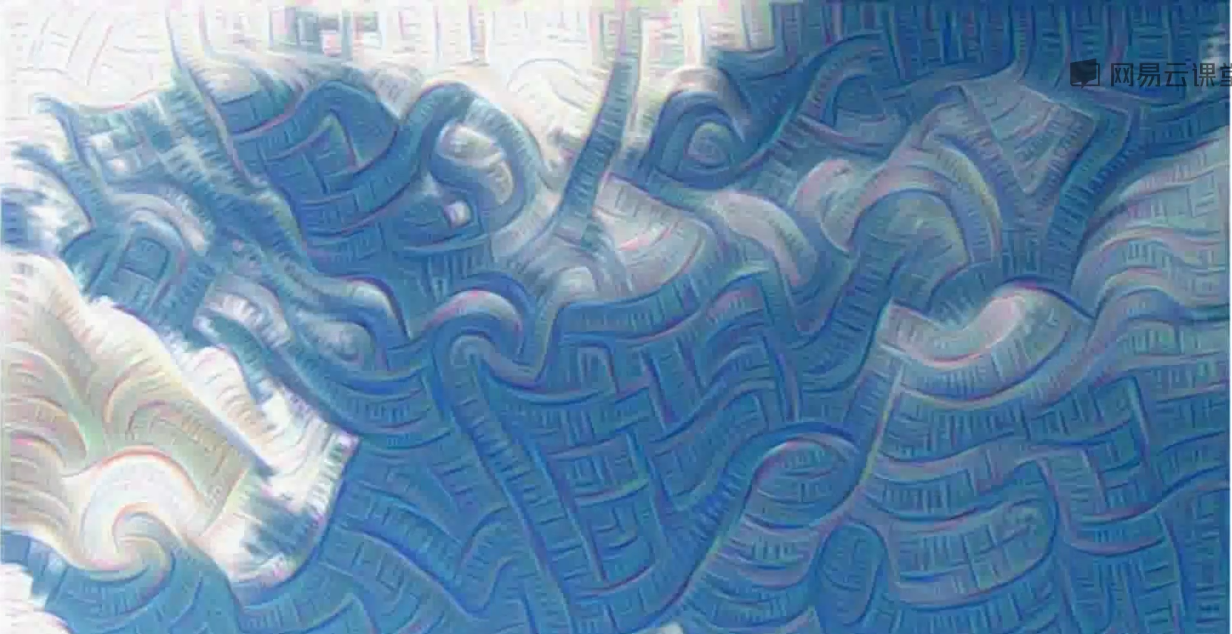
不是合成一个图像来最大化一个特定神经元，而是最大化某神经元的激活来改变图片



在计算梯度前，进行图片抖动时一个正则化，在较高层应用DeepDream

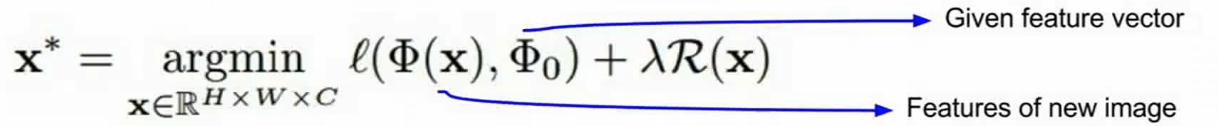


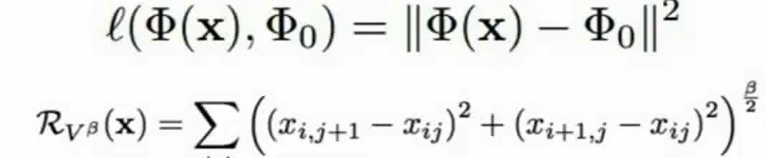
在较低层应用DeepDream



# 特征反演

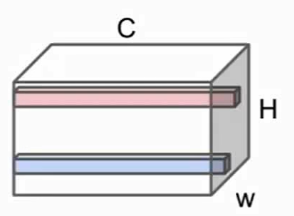
将图像经过网络，记录一个图像的特征值，通过最小化捕获到的特征向量的距离，来尝试合成一个新的与之前计算过的图像特征相匹配的图像

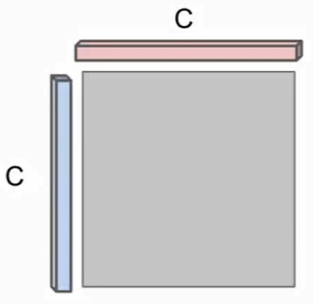




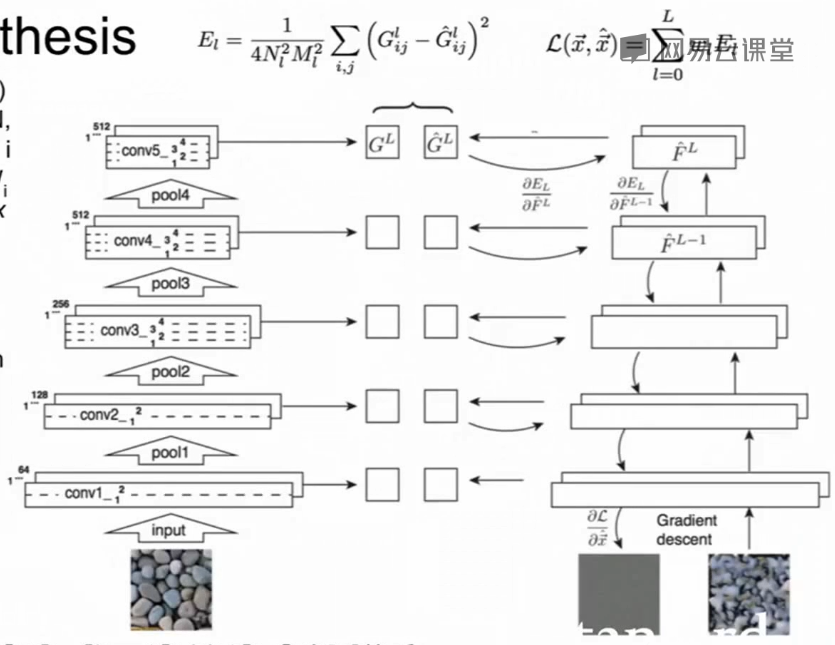
# 纹理合成

格拉姆矩阵，在某一层卷积层后给出了C\*H\*W的特征，可以看作H\*W的网格，在每个网格有C维数据。取两个网格的C维数据构成一个矩阵。如果某个ij数较大，则表示在分别两个特征向量上i，j会被同时激活，对h\*w的每一个网格都进行计算，并取它们的平均值。最终得到格拉姆矩阵

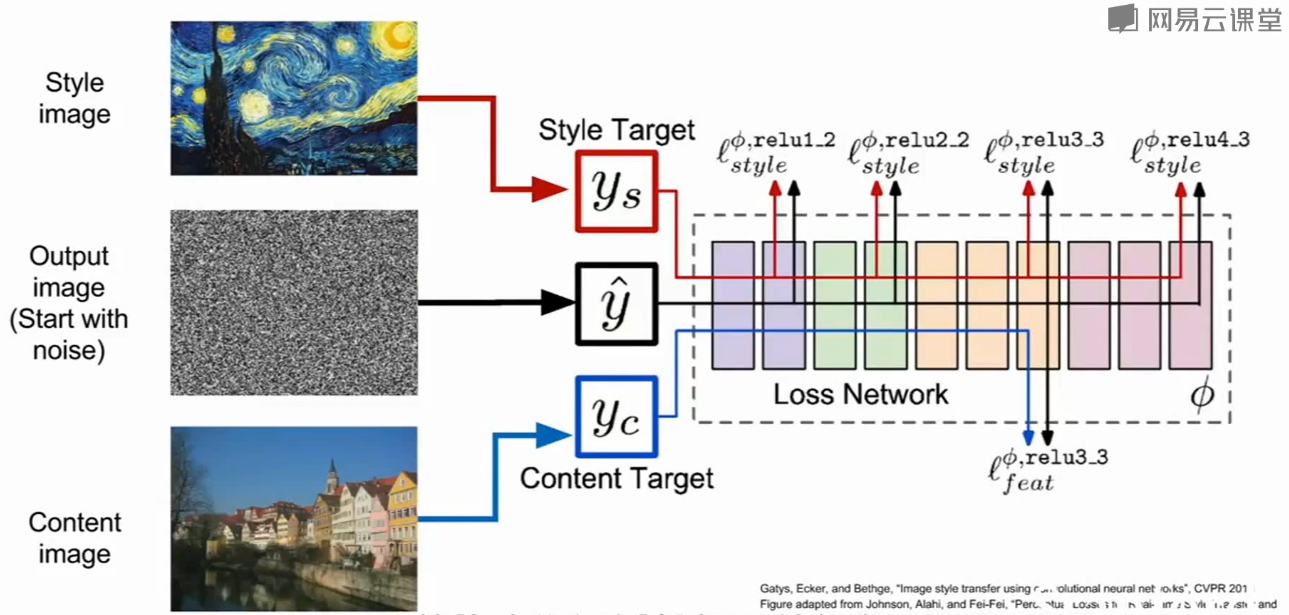




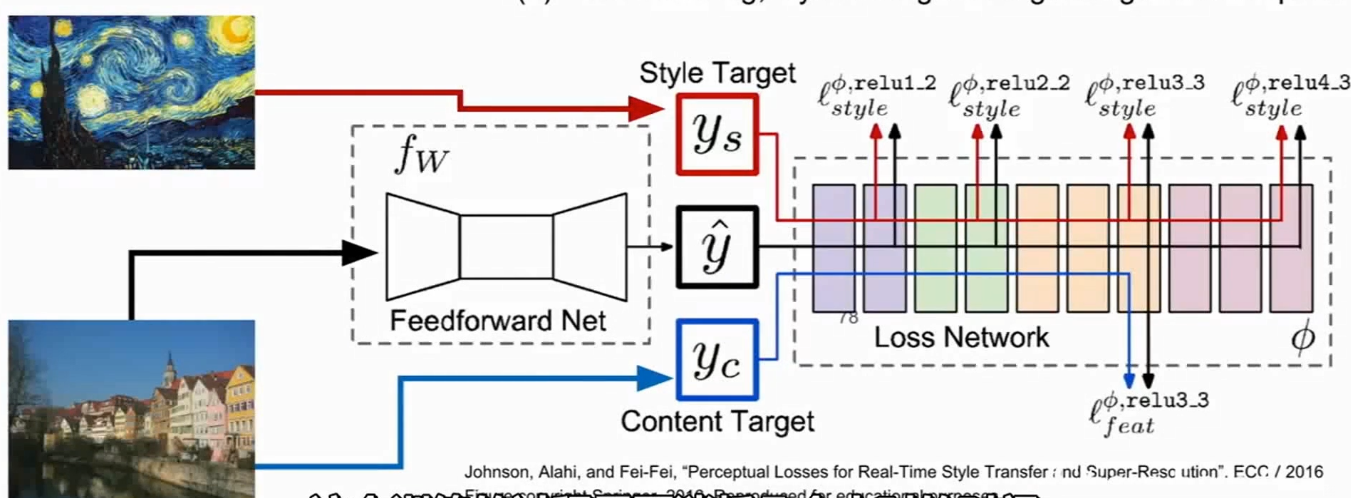
首先对于输入图片，计算已经预训练好的网络每一层的格拉姆矩阵，用随机噪声生成新的图片，将新的图片，将它通过神经网络，计算各层上的格拉姆矩阵，计算某层上输入图像的格拉姆矩阵和生成图像上的格拉姆矩阵之间的L2损失。然后进行反向传播计算生成图像的像素值梯度，然后进行梯度上升



将特征匹配的特征反演法和格拉姆矩阵匹配结合在一起可以完成风格迁移



直接训练一个前向网络用于风格迁移，通过计算内容图像和风格图像的损失更新权重

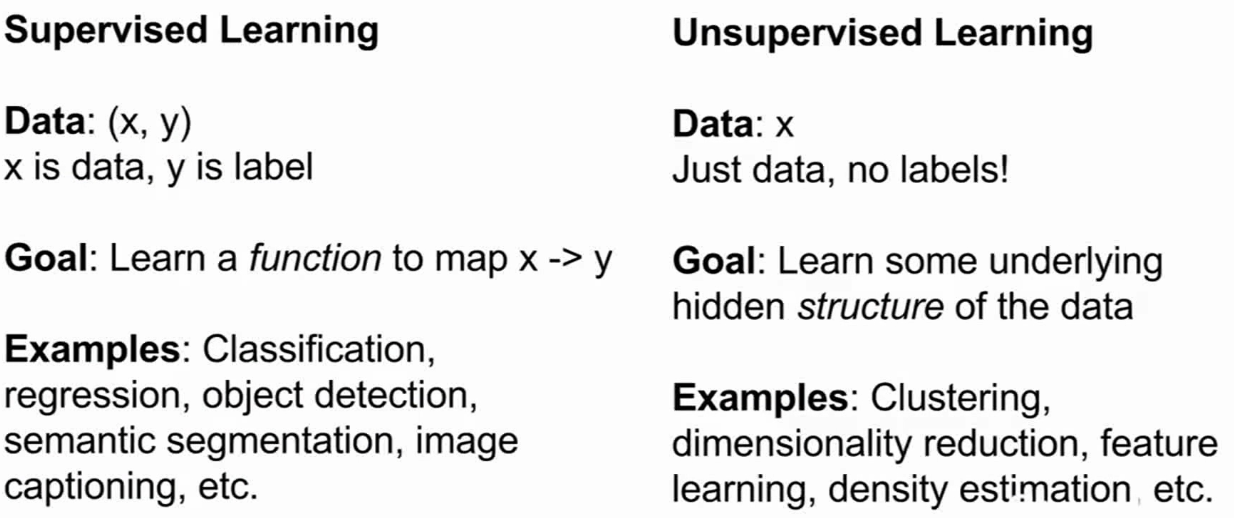


# 生成模型

生成模型包括pixelRNNs，pixelCNNs以及变分自编码器VAE还有生成式对抗网络。

无监督学习的例子：聚类、降维

无监督学习与监督学习的对比。

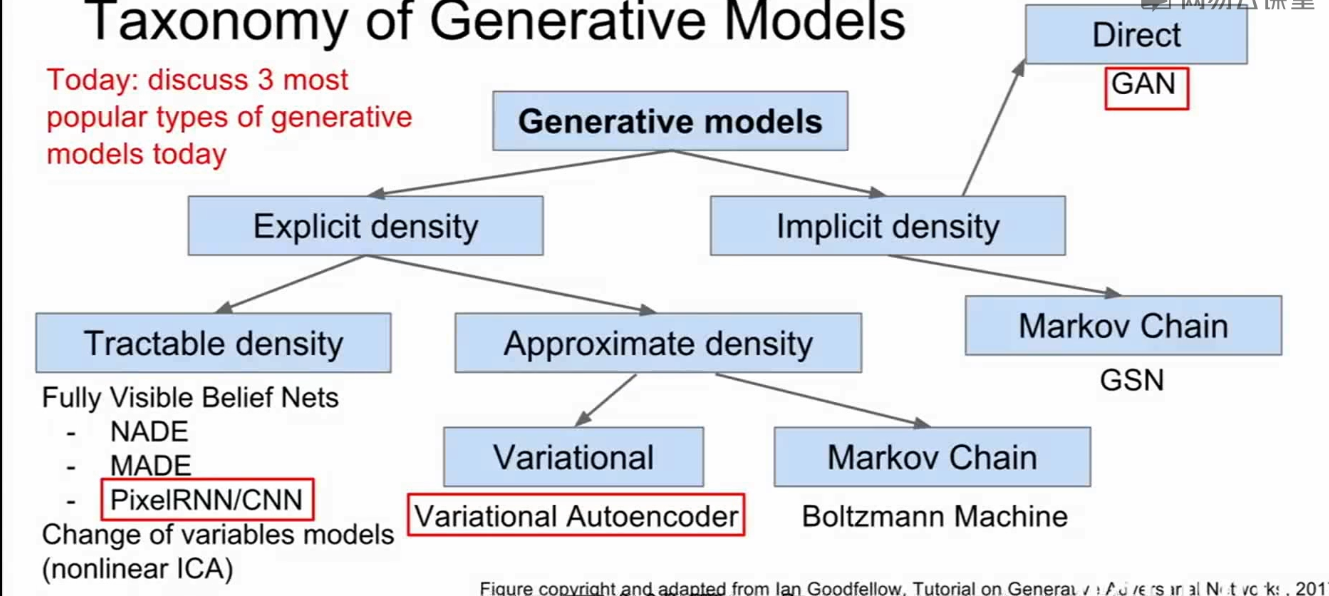


生成模型，该任务中，在给定训练数据的情况下，我们的目标是从相同的数据分布中生成新的样本

可以使用生成模型做显式的密度估计，显示地定义并求解出目标模型p-model。

也可以做隐式的密度估计，从p-model中的样本学习而不需要显示的定义它。

下图为生成模型的分类



pixelsRNN和pixelsCNN以及变分自动编码器，都是显式密度模型，GANs是隐式密度估计方法