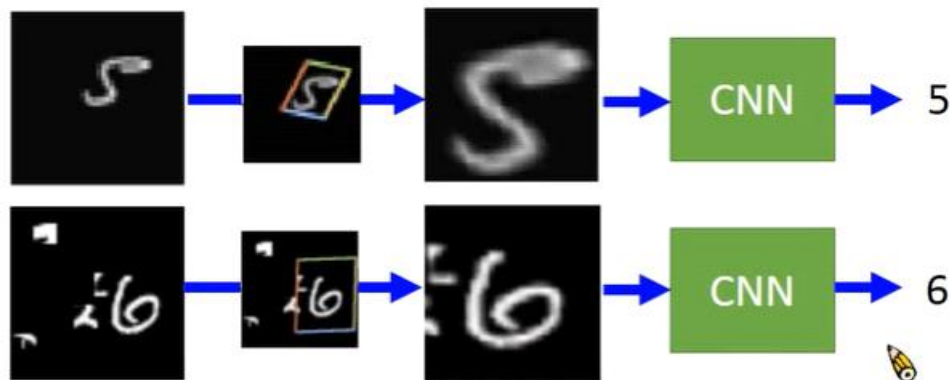


Spatial Transformer Layer

2021年4月1日 18:58

CNN is not invariant to scaling and rotation.



这个layer 也是一个nn 这个nn 和cnn 一起 train, 也可以 transform cnn所有layer

- How to transform an image/feature map

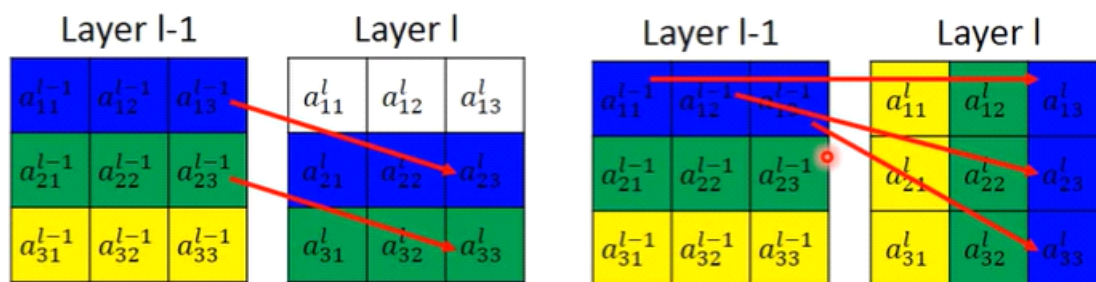


$$\text{General layer: } a_{nm}^l = \sum_{i=1}^5 \sum_{j=1}^5 w_{nm,ij}^l a_{ij}^{l-1}$$

$$\text{If we want translate as above: } a_{nm}^l = a_{(n-1)m}^{l-1}$$

给weight做不同的设计，可以达到旋转缩放平移

- How to transform an image/feature map



$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$$

Index of layer l

6 parameters to describe the affine transformation

先做一个线性变换（找到要识别的图像）再继续运算

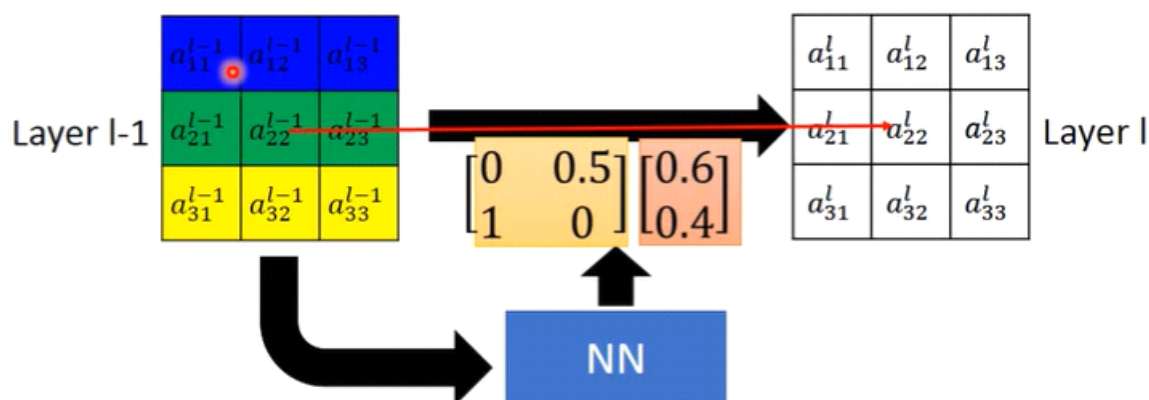
产生问题：每个图片的参数都应该是不同的，这是怎么达到的呢

Spatial Transformer Layer

$$\begin{bmatrix} 1.6 \\ 2.4 \end{bmatrix} = \begin{bmatrix} 0 & 0.5 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \end{bmatrix} + \begin{bmatrix} 0.6 \\ 0.4 \end{bmatrix}$$

Index of layer $l-1$ Index of layer l

6 parameters to describe the affine transformation



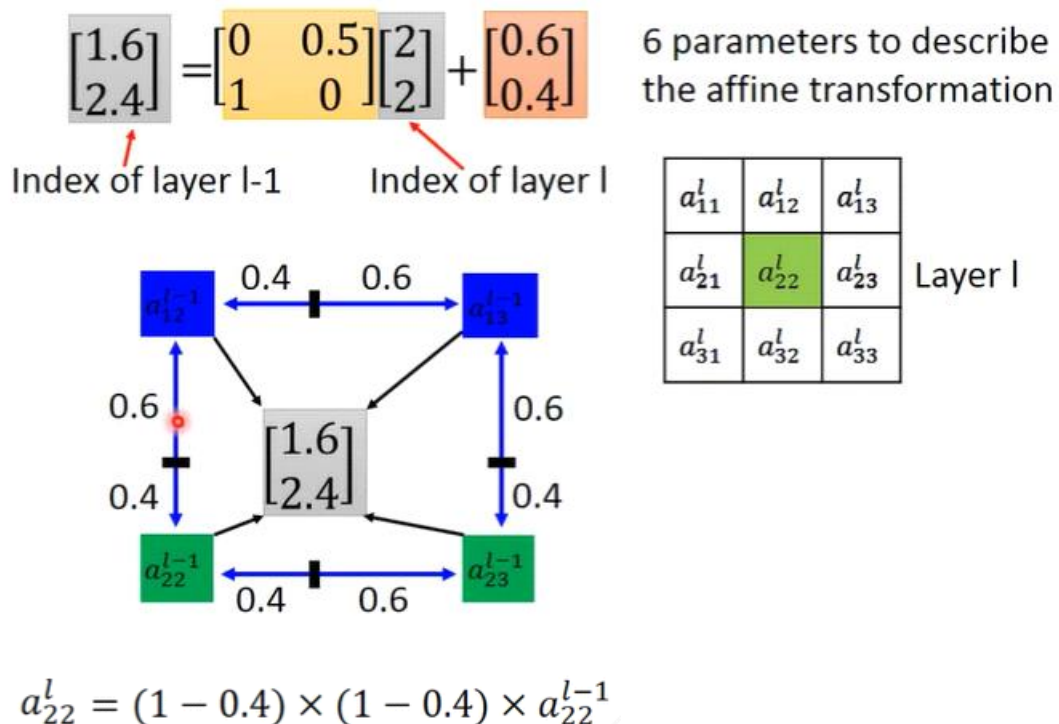
得到没有的index怎么办：取一个最近的。

what 's the problem?

可以用gradient descent解吗？
 不可以的，Gradient is always 0.
 Have no way to train the NN.

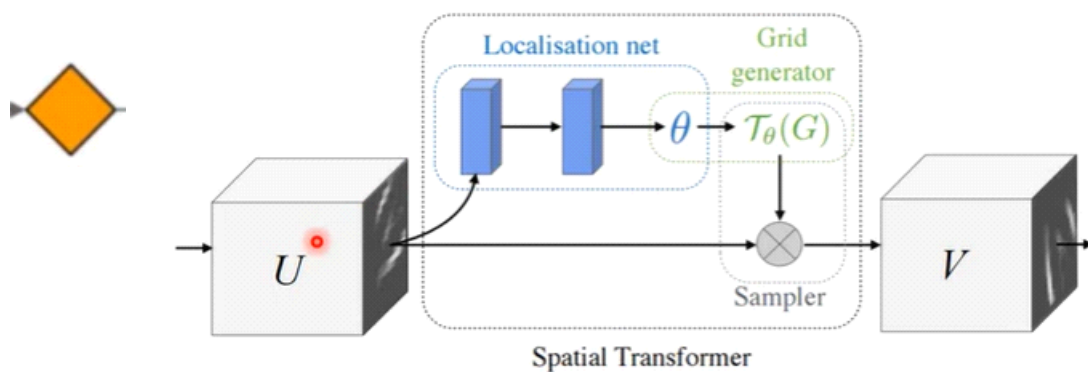
So how to solve it?

Interpolation

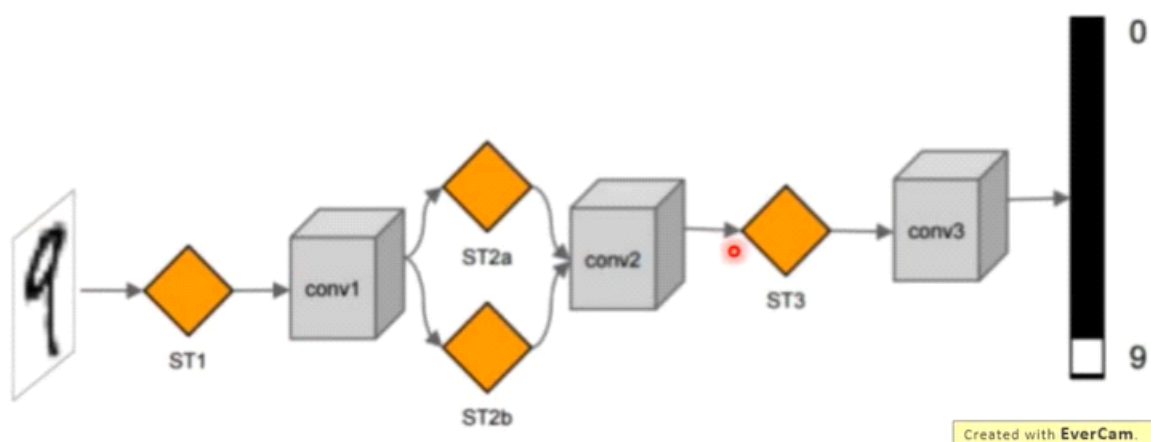


这样就可以微分啦！

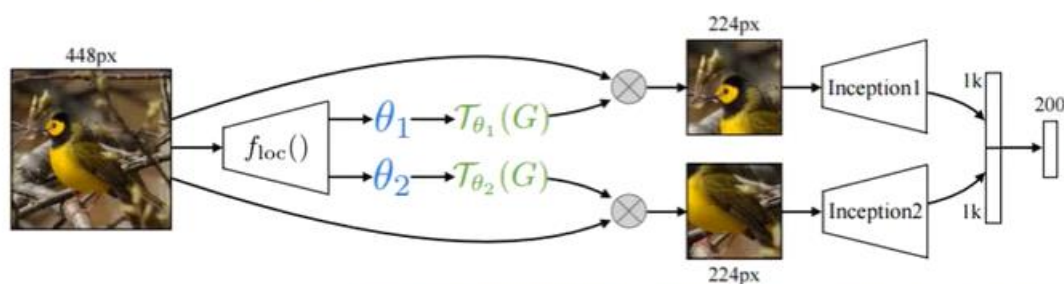
原始paper



Localisation net 就是前面提到的6 个参数的NN



Bird Recognition $\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} e \\ f \end{bmatrix}$



Model	
Cimpoi '15 [4]	66.7
Zhang '14 [30]	74.9
Branson '14 [2]	75.7
Lin '15 [20]	80.9
Simon '15 [24]	81.0
CNN (ours) 224px	82.3
2×ST-CNN 224px	83.1
2×ST-CNN 448px	83.9
4×ST-CNN 448px	84.1

