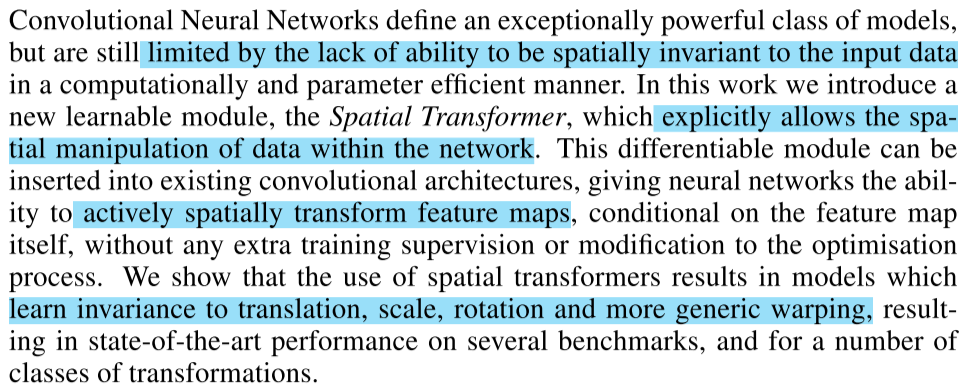
**STN阅读总结**

15331416 赵寒旭

**Abstract**



卷积神经网络定义了一个强大的分类模型，但是缺乏应对输入数据空间变换的能力。我们介绍了一种新的可习得的模型——空间变换，它明确允许了网络中数据的空间操作。这一可微的模型可以被插入目前的卷积结构，使神经网络能够对feature map进行基于其自身的空间变换而不需要对optimisation process进行任何额外的监督或修改。空间变换模型能够学习到平移，放缩，旋转和更多通用变换的不变性，提高网络的性能。

Q: spatial invariance具体含义是否指即使一个物体在空间上发生了变换，但是物体的属性仍然不变？比如原来的图像是数字1，即使经过平移旋转缩放等一系列空间变换，仍然保持为数字1并应该被网络识别。

试解：空间不变性对应着图像处理的经典手段：平移、缩放和旋转，他们同属于空间变换，并可以通过坐标矩阵的仿射变换来实现。

仿射变换矩阵：

1. 平移

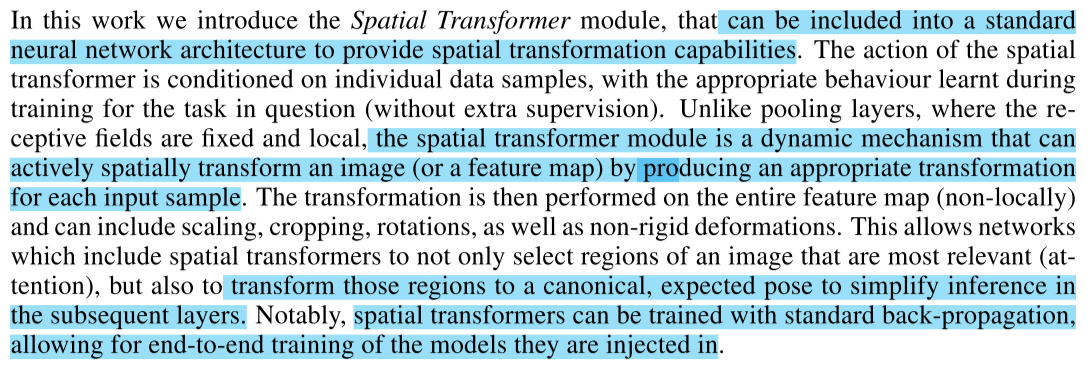
2. 缩放

3. 旋转

绕原点顺时针旋转度，坐标仿射矩阵：

需要做Normalization把坐标调整到使绕图像中心旋转。

**1. Introduction**

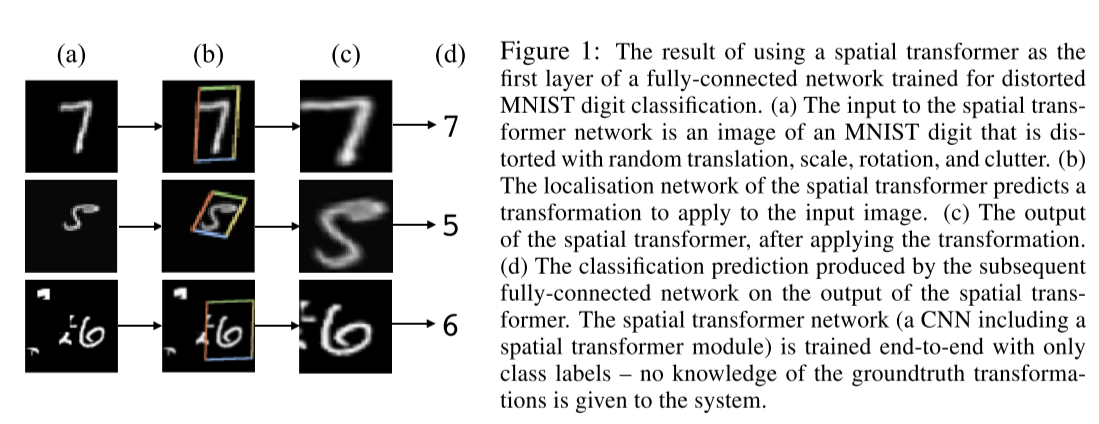


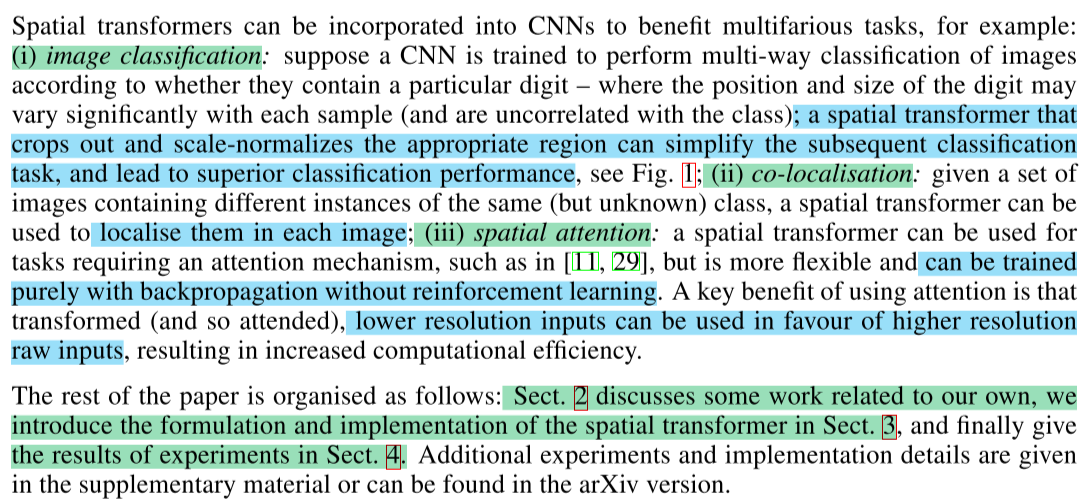
空间变换器模块可以被包含在标准神经网络结构中以提供空间变换能力。

空间变换器模块是一种动态机制，可以通过对每个输入样本产生合适的变换对一个图像（或者一个feature map）进行主动的空间变换。

包含空间变换的网络将不仅选择一个图片中最合适的区域，还可以把这些区域变换到一个标准的我们所期待的样子以简化后续层的推断。

空间变换可以被标准反向传播训练，允许包含这一模块的整个模型进行端到端的训练。





空间变换网络插入CNN中，对多种任务均有益：

（1）image classification

空间变换可以把输入剪切并缩放至合适的区域以简化后续的分类任务，提升分类性能。

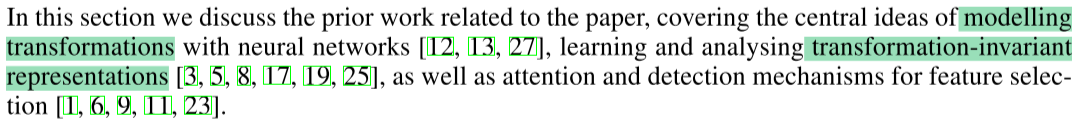
（2）co-localisation

给定一组包含一类物体不同实例的图像，空间变换器可以在每张图像中定位它们。

（3）spatial attention

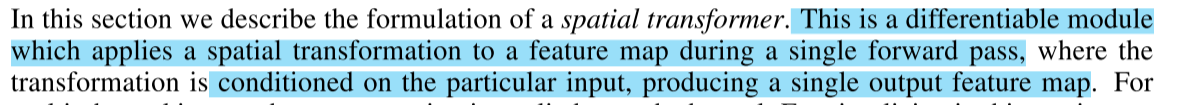
空间变换器可以被用于需要attention机制的任务中，它更加灵活并且可以只用反向传播进行训练。

**2. Related Work**

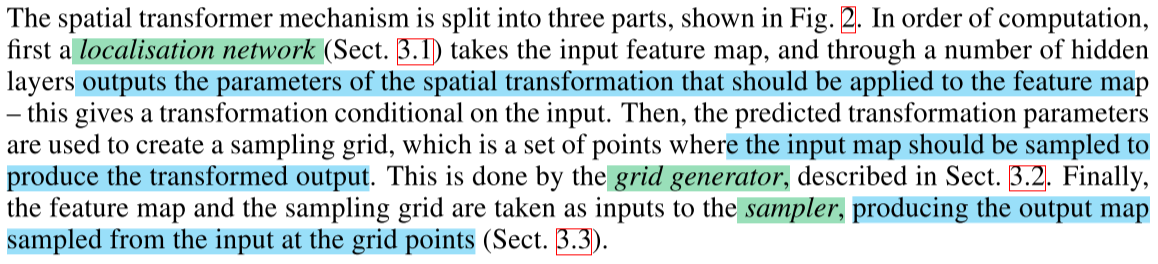


这一节是和本文有关的一些网络模型的讨论。

**3. Spatial Transformers**



这是一个在单个前向传递中把空间变换应用到feature map上的可微模型，其中变换基于特定输入产生单个输出feature map。

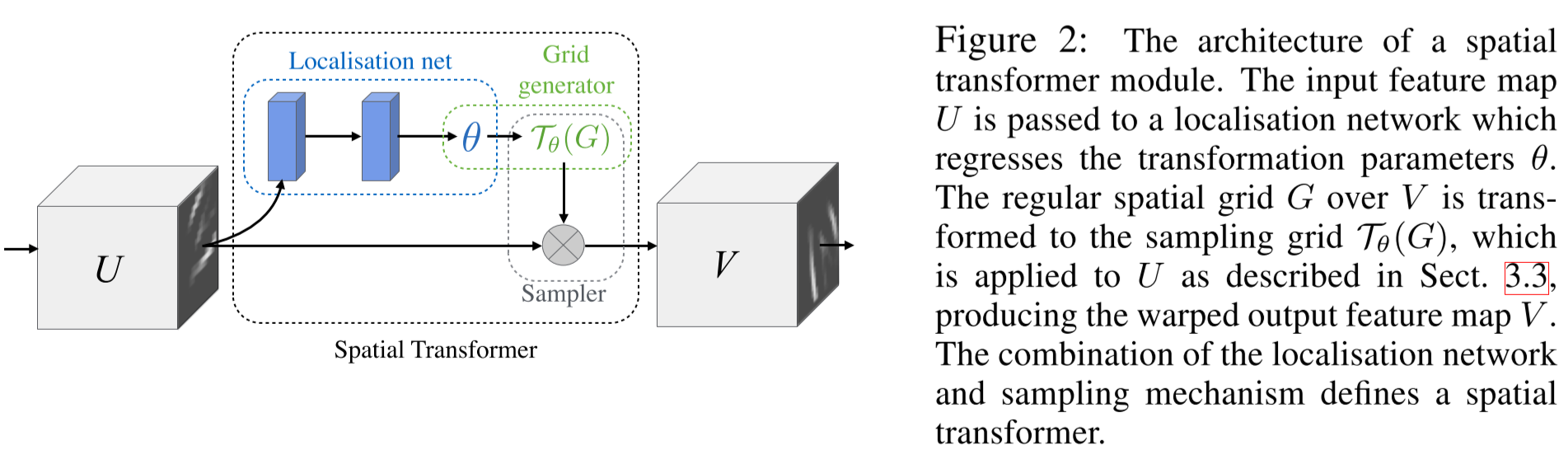


空间变换机制被分成三个部分：

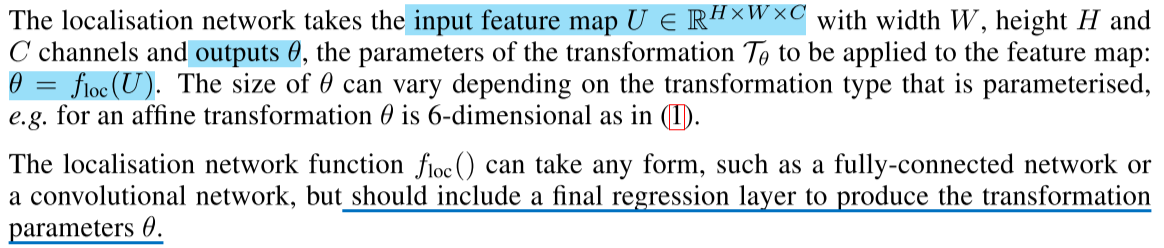
localization network: 输入feature map通过一系列隐藏层后输出要被应用于feature map的空间变换参数。

grid generator: 根据变换参数确定在输入feature map和输出feature map上的映射关系。

sampler: 结合输入的feature map和映射关系，获得变换后的输出。



**3.1 Localisation Network**



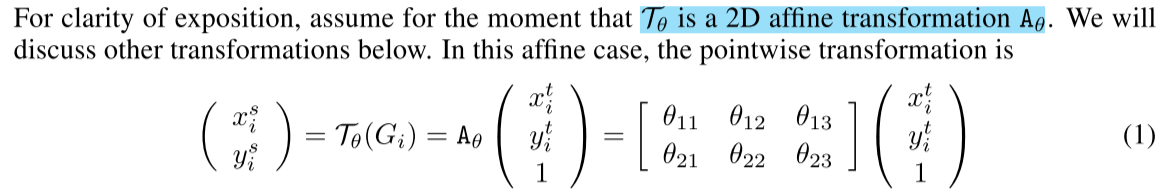
input: feature map （width W, height H, C channels）

output:

localisation网络函数可以是任何形式，如全连接网络或卷积网络，但最后一定有一个回归层用于生成变换参数。

的形式可以根据需要而变化，以2D仿射变换为例，就是一个2\*3的向量输出。

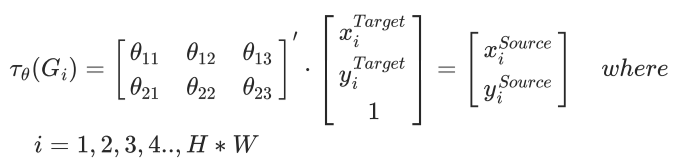
**3.2 Parameterised Sampling Grid**



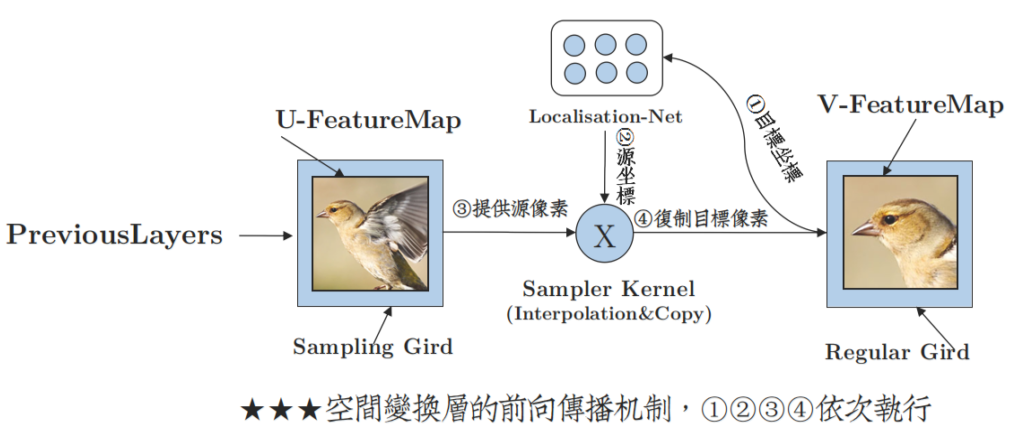
考虑逆向仿射变换，先根据仿射变换输出的大小，生成输出的坐标网格点，再对该坐标位置矩阵中的点进行仿射变换。

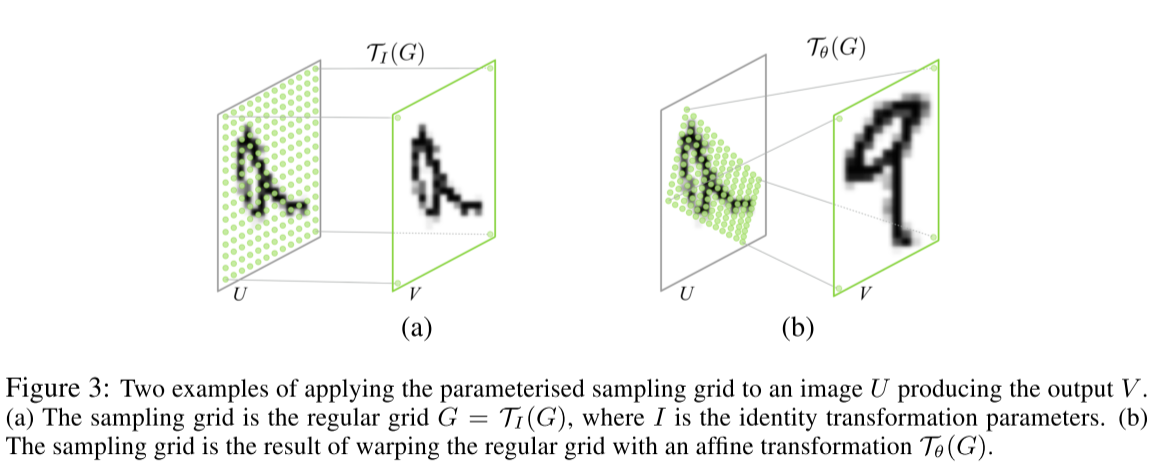
此时仿射系数为的逆矩阵，经仿射变换后可以得到V中的坐标点在U中的对应位置（可能非整数），再通过在输入图像中进行插值得到此坐标点的值。

得到U中坐标点的值后，则可将其复制到V中，得到仿射变换结果。



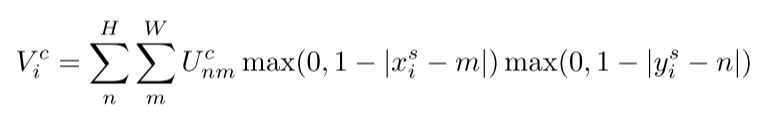
具体过程参考下图：





**3.3 Differentiable Image Sampling**

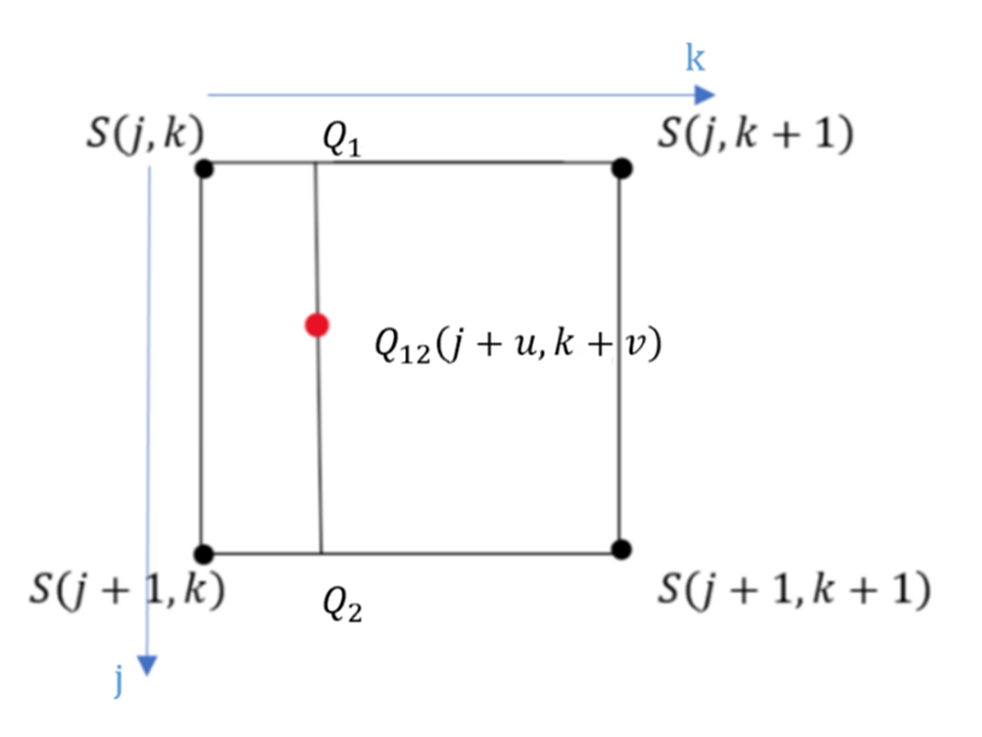
对双线性插值的情况有插值等式如下：

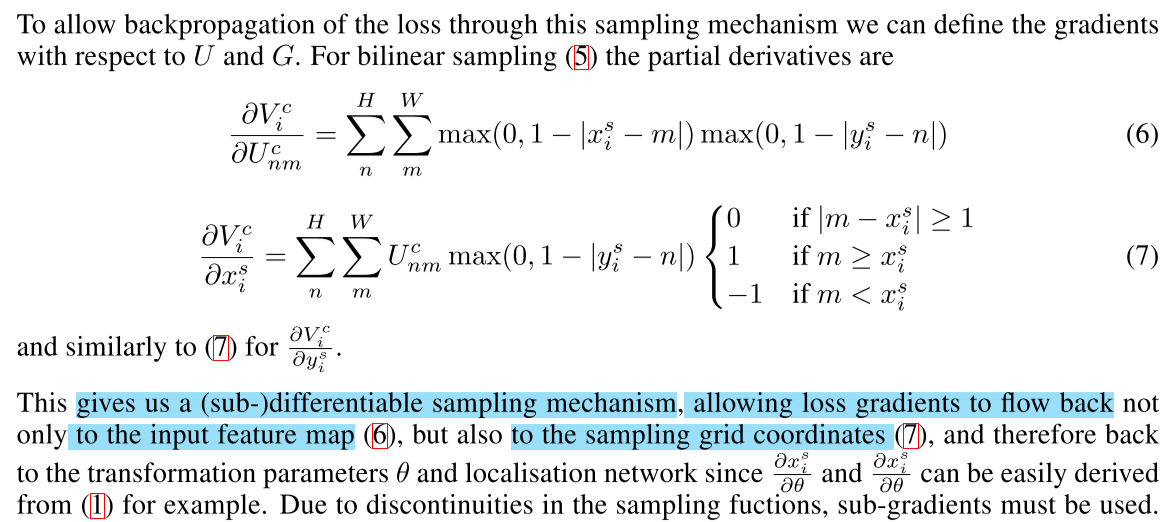


V中变换到U中，选取在U中的邻近点求和。

Q：此处不理解，按照双线性插值方法直接选择临近的四个点进行插值就可以了，而这个式子好像是循环选邻近点再求和。

双线性插值：

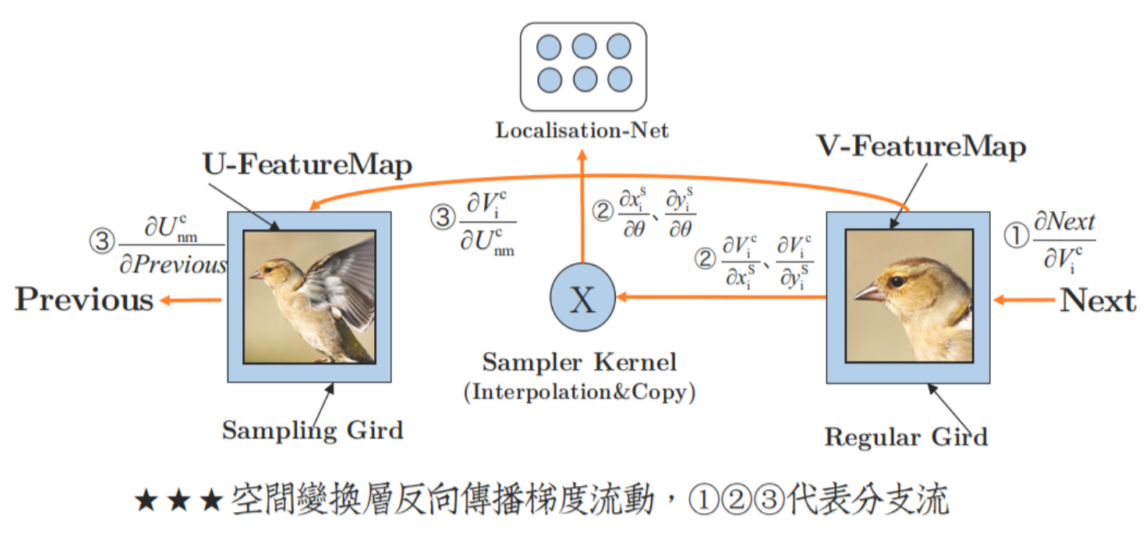




此处的采样核函数是不连续的，不能直接如下求导：

应该分两步，先对和求局部梯度、，后有：

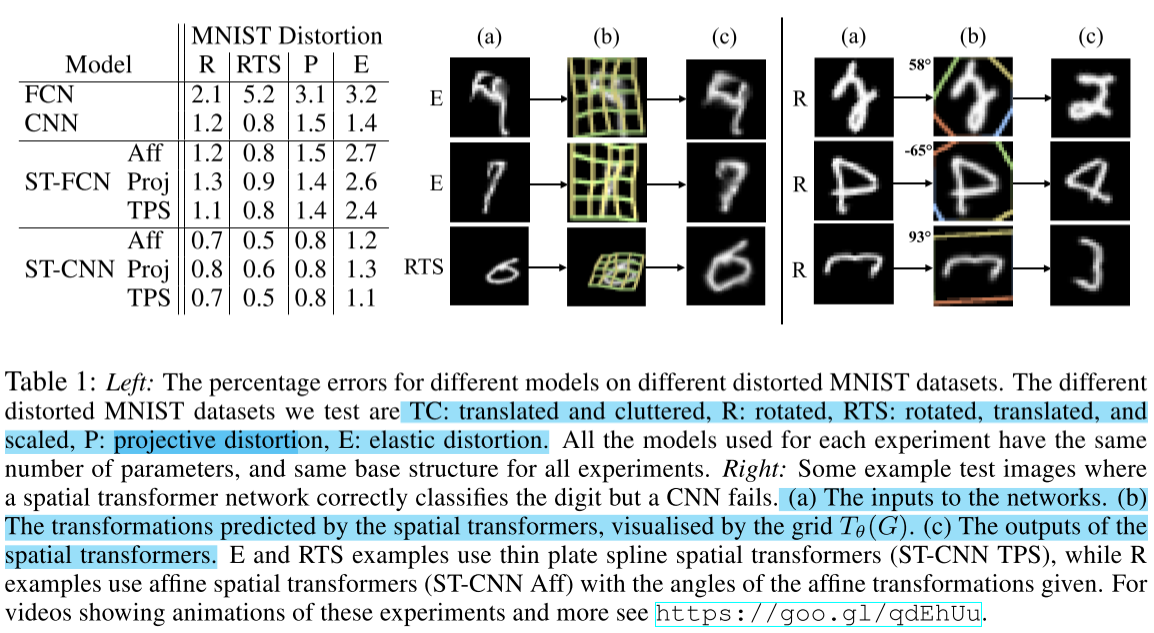
梯度流动过程参考下图：



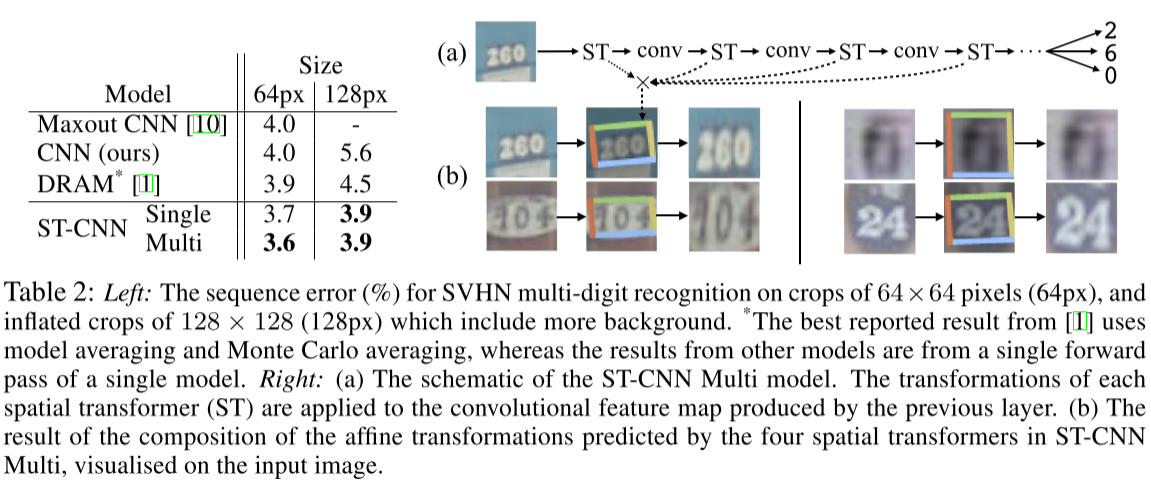
**3.4 Spatial Transformer Networks**

**4. Experiments**

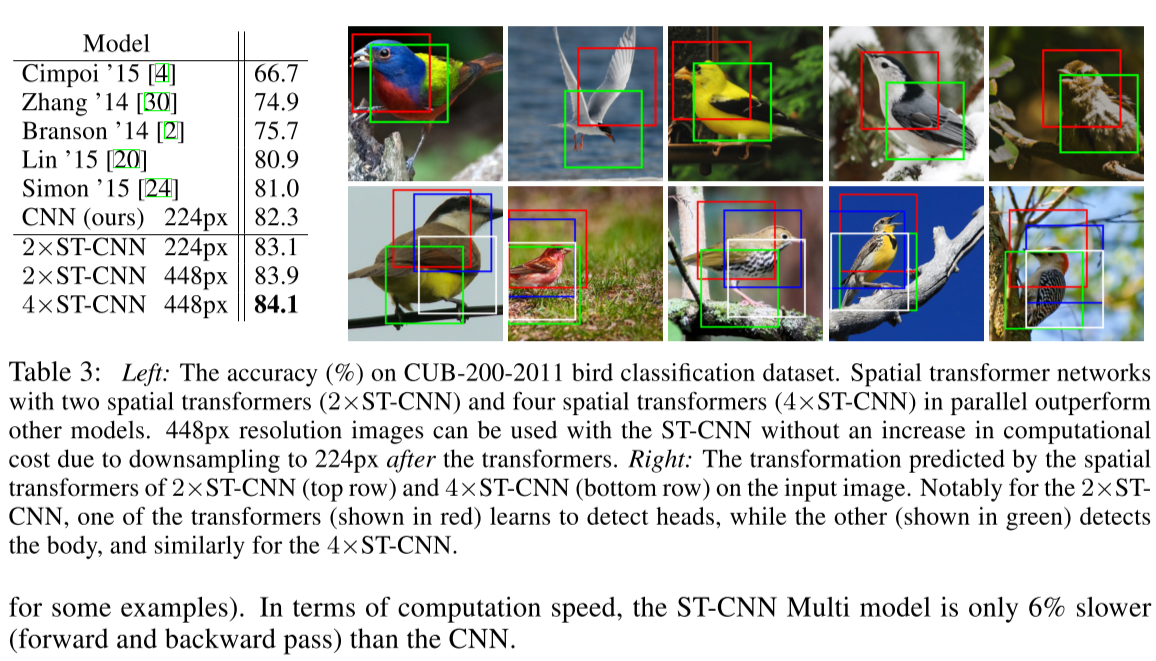
**4.1 Distorted MNIST**



**4.2 Street View House Numbers**



**4.3 Fine-Grained Classification**



**5. Conclusion**

