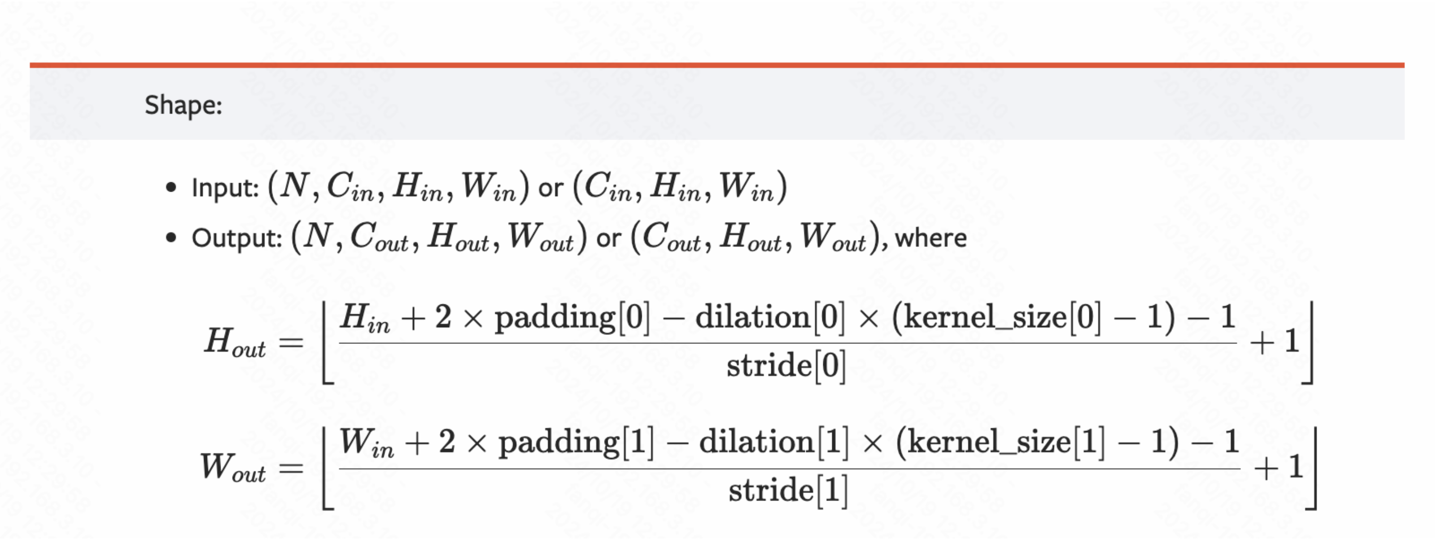
作业说明

1. 选择题的第三题和应用题的第一题卷积核数目都是一
2. 卷积输出尺寸计算中如果有无法整除的向下取整即可
3. Alexnet的文献中输入层的尺寸是224*224 但是实际在当时使用时为227*227。这次作业中alexnet的输入尺寸按照227x227，模型结构按照 https://github.com/BVLC/caffe/blob/master/models/bvlc_alexnet/deploy.prototxt

详细解释说明如下：

卷积网络设计的时候尽量设计成可以整除，如果不能整除则会造成特征信息丢失。如果遇到不能整除的，计算输出大小的时候就是向下取整

https://pytorch.org/docs/stable/generated/torch.nn.Conv2d.html



Shape:

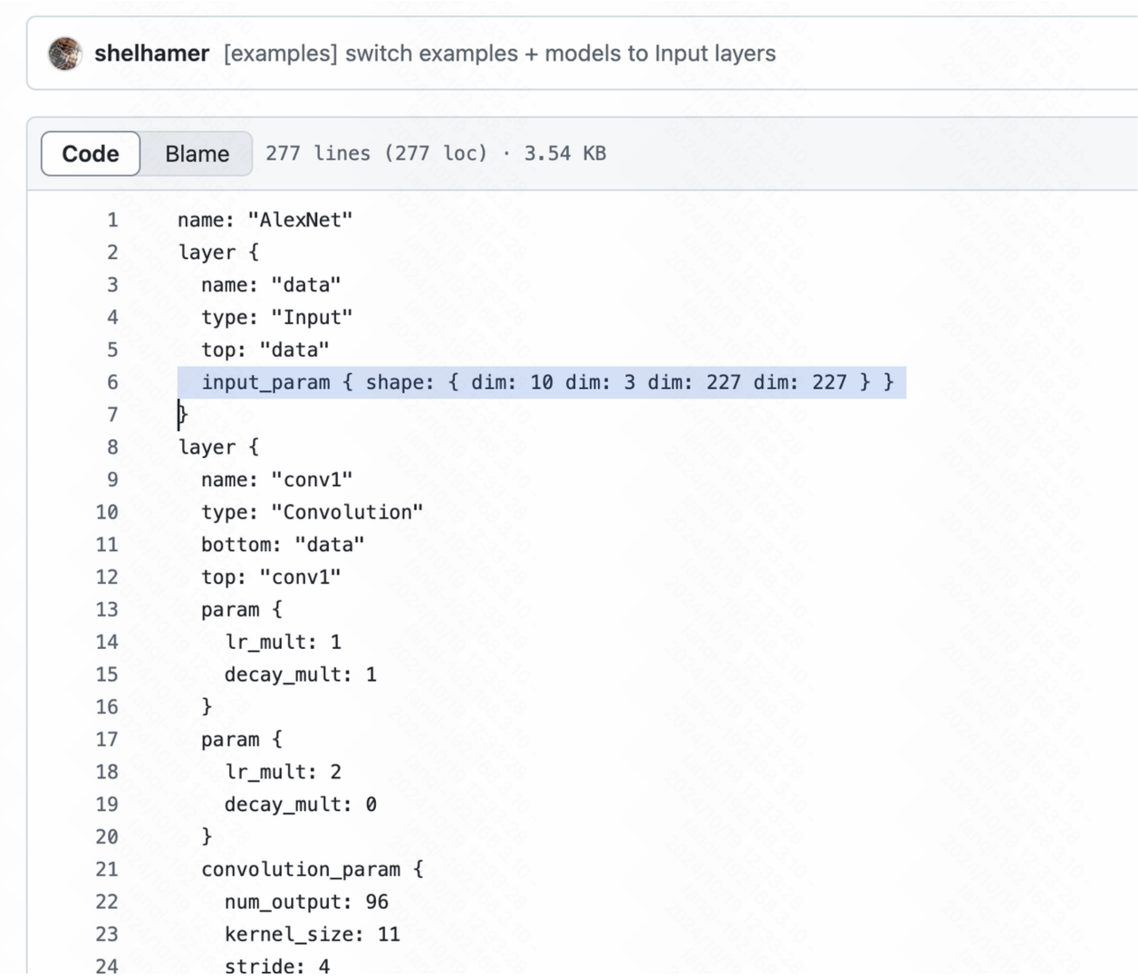
- Input: $(N, C_{in}, H_{in}, W_{in})$ or (C_{in}, H_{in}, W_{in})
- Output: $(N, C_{out}, H_{out}, W_{out})$ or $(C_{out}, H_{out}, W_{out})$, where

$$H_{out} = \left\lfloor \frac{H_{in} + 2 \times \text{padding}[0] - \text{dilation}[0] \times (\text{kernel\_size}[0] - 1) - 1}{\text{stride}[0]} + 1 \right\rfloor$$

$$W_{out} = \left\lfloor \frac{W_{in} + 2 \times \text{padding}[1] - \text{dilation}[1] \times (\text{kernel\_size}[1] - 1) - 1}{\text{stride}[1]} + 1 \right\rfloor$$

在AlexNet 的原始论文中提到图像的输入尺寸是 224x224 ，但AlexNet 的Caffe 代码实现中，输入尺寸是 227x227 。 这是由于在第一个卷积层中使用了步长为4 的卷积核，如果输入尺寸是 224x224 ，则输出尺寸无法被整除。 为了避免这种情况，代码实现将输入尺寸调整为 227x227 。

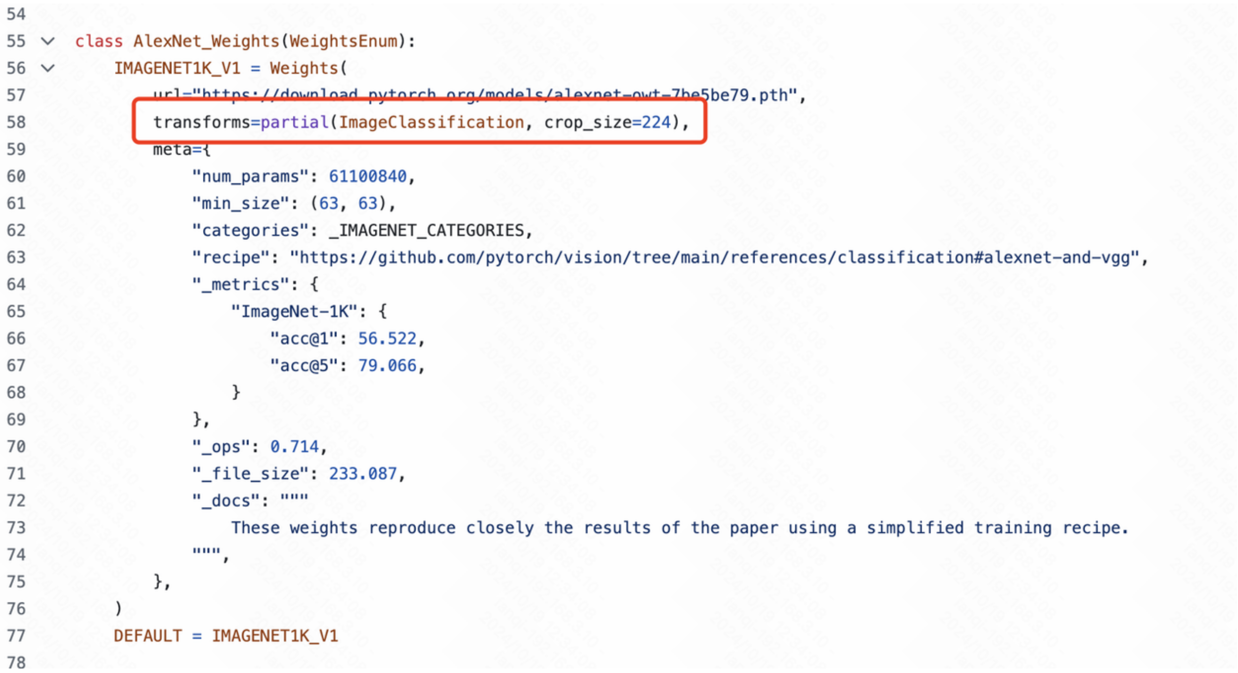
https://github.com/BVLC/caffe/blob/master/models/bvlc_alexnet/deploy.prototxt


shelhamer [examples] switch examples + models to Input layers
Code
Blame
277 lines (277 loc) · 3.54 KB

```
name: "AlexNet"
layer {
  name: "data"
  type: "Input"
  top: "data"
  input_param { shape: { dim: 10 dim: 3 dim: 227 dim: 227 } }
}
layer {
  name: "conv1"
  type: "Convolution"
  bottom: "data"
  top: "conv1"
  param {
    lr_mult: 1
    decay_mult: 1
  }
  param {
    lr_mult: 2
    decay_mult: 0
  }
  convolution_param {
    num_output: 96
    kernel_size: 11
    stride: 4
```

而目前pytorch的实现过程中通常直接将输入设计成224*224

https://github.com/pytorch/vision/blob/main/torchvision/models/alexnet.py



```
class AlexNet_Weights(WeightsEnum):
    IMAGENET1K_V1 = Weights(
        url="https://download.pytorch.org/models/alexnet-owt-7be5be79.pth",
        transforms=partial(ImageClassification, crop_size=224),
        meta={
            "num_params": 61100840,
            "min_size": (63, 63),
            "categories": _IMAGENET_CATEGORIES,
            "recipe": "https://github.com/pytorch/vision/tree/main/references/classification#alexnet-and-vgg",
            "_metrics": {
                "ImageNet-1K": {
                    "acc@1": 56.522,
                    "acc@5": 79.066,
                }
            },
            "_ops": 0.714,
            "_file_size": 233.087,
            "_docs": """
                These weights reproduce closely the results of the paper using a simplified training recipe.
            """,
        },
    )
    DEFAULT = IMAGENET1K_V1

```