# Op升级工作概况

## channel_last格式输入支持

### 背景

Paddle之前仅部分OP支持channel_last(NHWC)格式输入。TensorFlow目前CV模型默认均为channel_last格式输入，但支持NHWC和NCHW两种格式输入。以 `fluid.layers.conv2d_transpose(input, num_filters, filter_size=None, ..., data_format='NCHW')` 为例，用户通过设置 `data_format` 参数指定输入和输出的格式。

### 主要思路

#### conv_transpose

（1）use_cudnn=False，前向计算：gemm + col2im(conv2d_transpose) or col2vol(conv3d_transpose)。这种算法是把卷积运算转换成两个矩阵相乘。以conv2d_transpose前向计算为例，循环处理每一个batch：

- input_shape: (h, w, i_c) ，filter_shape: (i_c, o_c, k_h, k_w)
- resize and slice：input_slice：(h * w, i_c/g)，filter_slice: (i_c/g, o_c/g * k_h * k_w)
- col_matrix = filter_sliceˆT * input_sliceˆT => (o_c/g * k_h *k_w, h w)
- col2im：(o_c/g * k_h * k_w, h * w) => (o_h, o_w, o_c/g)

原始实现中col2im是根据(o_c/g, k_h, k_w, h, w)形状的输入，得到一个(o_c/g, o_h, o_w)形状的输出。所以对于前向，修改的核心是col2im（conv2d_transpose）函数中数据的读写。相同的数据，以NCHW和NHWC格式存储时，改变数据索引计算方式即可。假设以NCHW格式存储下，某一个数据的坐标是（nid, cid, hid, wid），它的索引计算是idx=((nid * C + cid) * H + hid) * W + wid。对应到NHWC格式存储下的索引就是idx=((nid * H + hid) * W + wid) * C + cid。

(2) use_cudnn=True，参考了TF的做法，对NHWC格式数据先transpose为NCHW格式，之后按照原始实现计算得到NCHW格式的输出，再对输出transpose为NHWC格式。

#### 其他Op

group_norm、interpolate（包含resize_nearest、resize_bilinear、resize_trilinear3种算法），修改的也是计算逻辑中数据的读写部分，修改索引计算方式即可。

### 相关PR

- conv_transpose：PR20072
- group_norm：PR19614
- interpolate：PR19914

## 非对称padding

# 背景

Padding之前涉及到Paddlnng操作的OP，如conv， pool都只支持对称的Padding方式，不支持两侧Padding size不同。以 `fluid.layers.conv2d(input, num_filters, ..., padding=0)` 为例，原来 `padding` 参数只支持传入int或者包含2个值的list：

- padding=2，表示在输入的上下各填充2行0，左右各填充2列0
- padding=[padding_H, padding_W]，表示上下各填充padding_H行0，左右各填充padding_W列0

修改的需求：Op需支持上下或者左右可以进行非对称的padding。用户通过设置padding参数，指定上下左右padding的大小。

# 主要思路

以conv2d_transpose为例：
（1）use_cudnn=False，涉及修改的部分：python接口中支持长度为4的list；C++部分的修改主要是im2col和col2im函数中，获取padding参数中上下左右对应的padding大小。
（2）use_cudnn=True，由于cudnn不支持非对称padding，这种场景下，需要对输入先padding。例如padding=[1, 2, 0, 0]，表示上下需要分别padding 1行和2行0，下面比上面多padding了1行：

- 首先把input下面这边多出的1行补上：即对下面的边先padding 1行0，得到新的输入为input_trans
- 更新padding参数：padding=[1, 1, 0, 0]，依然是对称padding
- 新的padding参数和变换后的输入input_trans就可以调用原来的cudnn算法

# 相关PR

conv_transpose：PR20072